Data ONTAP™ 6.5
System Administration
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 210-00475
December 2003

Copyright and trademark information

Copyright information

Copyright © 1994–2003 Network Appliance, Inc. All rights reserved. Printed in the U.S.A.

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

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

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

Portions of this product are derived from NetBSD, which is copyrighted 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.

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 and purchase of this product do not convey a license under any patent rights, trademark rights, or any other intellectual property rights of Network Appliance.

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

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

Trademark information

NetApp and the Network Appliance design are registered trademarks of Network Appliance, Inc. in the United States, Canada, and the European Union. Network Appliance is a registered trademark of Network Appliance, Inc. in Monaco and a trademark of Network Appliance, Inc. in the United States and Canada. FAServer is a registered trademark of Network Appliance, Inc. in the United States and the European Union. NetCache is a registered trademark of Network Appliance, Inc. in the United States, Canada, the European Union, and Japan. SnapCopy is a registered trademark of Network Appliance, Inc. in the European Union and a trademark of Network Appliance, Inc. in the United States. WAFL is a registered trademark of Network Appliance, Inc. in the United States, the European Union, and Canada. DataFabric, FilerView, NearStore, SecureShare, SnapManager, SnapMirror, and SnapRestore are registered trademarks of Network Appliance, Inc. in the United States. Data ONTAP is a trademark of Network Appliance, Inc. in the United States and Canada. Snapshot is a trademark of Network Appliance, Inc. in the United States and the European Union. NetApp–the Network Appliance Company is a registered trademark of Network Appliance, Inc. in the United States and other countries. ApplianceWatch, BareMetal, Center-to-Edge, ContentDirector, gFiler, MultiStore, SecureAdmin, Serving Data by Design, Smart SAN, SnapCache, SnapDrive, SnapVault, vFiler, and Web Filer are trademarks of Network Appliance, Inc. in the United States.

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

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

Table of Contents

Preface ix

Chapter 1 **About the MultiStore Software Product** 1

Understanding MultiStore 2

System administration for a filer running MultiStore 6

Chapter 2 **Managing MultiStore** 9

Establishing a vFiler 10
- Enabling or disabling the vFiler license 11
- Planning for vFiler creation 12
- Creating a vFiler 15
- Understanding the vFiler command family 19
- Setting up a vFiler manually 20

Administering vFiler resources from the hosting filer 22

Changing resources for vFilers 23

Adjusting the vFiler limit 27

Stopping and starting vFilers 29

Destroying a vFiler 31

Recreating a destroyed vFiler 33

Allowing or disallowing protocols for vFilers 34

Displaying vFiler status 36

Autosupport and syslogd messages about vFilers 37

Executing commands and setting options for vFilers 38

Effects of filer reboot on vFilers 42

Understanding volumes and qtrees on vFilers 43

Backing up vFilers 45

Understanding LUNs on vFilers 46

Networking considerations 50

Understanding SnapMirror on the hosting filer 51

Understanding SNMP when used with vFilers 52

Performance monitoring and statistics 53

Chapter 3 **Using vFilers in Different IPspaces**. 55

Understanding IPspaces 56

Advantages of using VLAN tagging for IPspaces 60

Requirements for using clusters and IPspaces 61

Case studies for IPspaces 63

Managing IPspaces on a filer 64
- Creating an IPspace 65
- Listing IPspaces on a filer 66
- Assigning an interface to an IPspace 67
- Destroying an IPspace 69

Creating a vFiler in a nondefault IPspace 70

Chapter 4 **File Access Using NFS and CIFS** 73

How to specify path names for NFS exports or CIFS shares. 74

Preparing the vFiler for NFS 75

Preparing the vFiler for CIFS. 76

Chapter 5 **Disk Space Management Using Quotas** 77

Understanding quotas. 78

Turning quotas on and off 79

How to resize quotas 82

Understanding the /etc/quotas file 83

Displaying quota status 84

Understanding quota reports 85

Messages from exceeded quota thresholds and soft quotas 87

Chapter 6 **Disaster Recovery and Data Migration** 89

Understanding disaster recovery and data migration 90

How to create and activate a disaster-recovery vFiler 91

How to move a vFiler. 92

Preparing the destination filer 93
Storage and network checklists 104
Creating the disaster-recovery vFiler 107
Activating the disaster-recovery vFiler 110
Reactivating the original vFiler 112
Migrating the vFiler 120

Chapter 7 **Virus Protection for CIFS** 125
Understanding virus scanning for vFilers 126
Configuring virus scanning for vFilers 127

Index 129

Preface

About this guide

This guide describes how to use the MultiStore™ software product on Network Appliance™ filers that run Data ONTAP™ 6.5 software and are Serving Data By Design™. It covers all NetApp®—the Network Appliance Company™ filer models.

Audience

This guide is for system administrators who are familiar with operating systems that run on the filer's clients, such as UNIX®, Windows 95™, Windows NT®, and Windows® 2000. It also assumes that you are familiar with how to configure the filer and how the NFS, CIFS, and HTTP protocols are used for file sharing or transfers. This guide doesn't cover basic system or network administration topics, such as IP addressing, routing, and network topology; it emphasizes the characteristics of the NetApp filer.

Terminology

When the term "filer" is used, the information applies to all NetApp filer models.

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 filer commands on either the system console or from any client computer that can access the filer 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 filer administrator can be performed by entering commands at the filer console, in configuration files, or through a Telnet session or a Remote Shell connection.

Another method of performing many common filer tasks is to use FilerView, the graphical management interface for viewing and managing a filer from a Web browser. FilerView comes with every filer, 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 filer with FilerView, and about FilerView Help, see the *Storage Management 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
Italic font	Words or characters that require special attention. Placeholders for information you must supply. For example, if the guide says to enter the `arp -d` *`hostname`* command, you enter the characters “arp -d” followed by the actual name of the host. 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.

Caution

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

WARNING

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

About the MultiStore Software Product 1

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 filer running MultiStore" on page 6

Understanding MultiStore

What MultiStore is

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

The storage resource assigned to a vFiler 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.

Prerequisites for using 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 filer.
- You use the filer to host data for multiple customers, such as clients of a service provider or different organizations within an enterprise.
- You use the SnapMirror™ feature to migrate data from one filer to another.

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 filer for easier administration.

  To consolidate the server, you can partition the filer into vFilers and then copy the data from the servers to the vFilers. If the vFilers are for CIFS users, you can set up the vFilers 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 vFilers are for 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 filer 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 filers. MultiStore, however, enables you to

install each vFiler in the appropriate domain while keeping all the data on the same physical filer.

- Because you can set up NIS and DNS servers for individual vFilers, after you consolidate the servers on one filer, network clients of the vFilers 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 filer to create many vFilers for client companies. Similarly, the information technology (IT) department of an enterprise can create vFilers for various organizations, or customers, within the enterprise.

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

For example, an SSP can create the following vFilers on a filer:

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

Although both CompanyA and CompanyB store data on the same filer, 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 filer 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 on page 55 for a more detailed discussion.

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

If you did not have MultiStore, you could use SnapMirror to copy data from one filer to another, but you would then need to edit the files' access control lists (ACLs), local user group definitions, user mapping information, and so on, before users could access the data. If the data being copied is stored on a vFiler, however, all user, group, and ACL information is encapsulated in the vFiler. You re-create a vFiler on the destination filer with the encapsulated information, and data can be served from the destination filer without further reconfiguration.

About the hosting filer

The filer on which you create vFilers is called the hosting filer. The storage and network resources used by the vFilers exist on the hosting filer.

About the default vFiler

Licensing MultiStore automatically causes Data ONTAP to create a default vFiler on the hosting filer. The name of the default vFiler is vFiler0.

Initially, vFiler0 owns all the resources on the filer. After you create vFilers, vFiler0 owns the resources that are not owned by the vFilers you create.

The default vFiler exists as long as the vFiler license is enabled. On a filer with the vFiler license enabled, you cannot destroy vFiler0.

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

Some limitations of MultiStore

MultiStore has the following limitations:

- You can create up to 32 vFilers on a filer (64 in a cluster), depending on the memory capacity of the hosting filer.

  A filer containing the maximum number of vFilers actually has 33 vFilers: vFiler0 plus the 32 vFilers that you, the system administrator, can create.

  See "Adjusting the vFiler limit" on page 27 for more information.
- Only Ethernet interfaces are supported for vFilers.
- Only the NFS, CIFS, and iSCSI file protocols are supported for vFilers.
- Not all Data ONTAP commands are supported on vFilers.

Hosting filer's access to data contained in vFilers

You, as the hosting filer administrator, do not have administrative, CIFS, or NFS access to the data owned by vFilers (other than the default vFiler, vFiler0). After you assign a qtree or volume to a vFiler, you lose access to the data in that qtree or volume.

Example: Before you create a vFiler 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, an attempt to mount the /vol/vol1 volume from the hosting filer results in the ".../vol/vol1 belongs to vFiler A, can't mount from vol0" error message.

To regain administrative control of storage owned by a vFiler, see "Administering vFiler resources from the hosting filer" on page 22.

Sample vFiler

A filer named toaster has the vFiler license enabled and is configured with three volumes, which are /vol/vol0, /vol/vol1, and /vol/vol2.

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

On toaster, you can create a vFiler 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 vFiler's *primary storage*, containing the vFiler's /etc directory.

Toaster is the hosting filer for vFiler1. vFiler0 uses all the IP addresses created on toaster except 123.123.123.1. It uses the following resources for storage:

- Space in the /vol/vol0 volume that is not in the /vol/vol0/qtree1 qtree
- The /vol/vol2 volume

The pseudo-root directory: From the point of view of CIFS clients, the root of the primary storage qtree (/vol/vol0/qtree1 in our example) is the root ("/")of vFiler1's file system. If "/" is shared, a CIFS client mapping to it can browse all of vFiler1's storage (all of /vol/vol0/qtree1and /vol/vol1 in this example) in a single tree. Network Appliance calls this mechanism the vFiler's "pseudo-root."

Note

Pseudo-root directories are not available to NFS clients. NFS clients must import discrete storage units as they are defined on the hosting filer. In this example, 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 in the vFiler's /etc/vfilerstorage file.

For more information, see "How to specify path names for NFS exports or CIFS shares" on page 74.

System administration for a filer running MultiStore

Types of system administration tasks

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

- Hosting-filer tasks, which are the same as those for a filer not supporting vFilers
- MultiStore management tasks, which are related to establishing vFilers and managing resources on vFilers
- Tasks related to using Data ONTAP features supported by vFilers

Hosting-filer tasks

You can perform tasks related to managing the resources on the hosting filer in the same way as you perform them on a filer without a vFiler license. 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 filer, you can manage MultiStore by

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

For more information about these tasks, see Chapter 2, “Managing MultiStore,” on page 9.

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

- Moving a vFiler from one hosting filer to another
- Creating a disaster-recovery vFiler

See Chapter 6, "Disaster Recovery and Data Migration," on page 89 for more information.

Tasks related to using Data ONTAP on vFilers

You can use Data ONTAP features on individual vFilers in the same way as you use them on a filer. Some of the tasks related to these features can be performed by vFiler administrators. These 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 filers and vFilers. For more information, see the *Storage Management Guide*.
- Managing iSCSI, LUNs, and initiator groups. For more information, see "Understanding LUNs on vFilers" on page 46.
- Managing NFS exports and CIFS shares. For more information, see "File Access Using NFS and CIFS" on page 73.
- Managing quotas. For more information, see "Disk Space Management Using Quotas" on page 77.
- Preparing vFilers for virus protection. For more information, see "Virus Protection for CIFS" on page 125.

Managing MultiStore 2

About this chapter

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

Topics in this chapter

This chapter discusses the following topics:

- "Establishing a vFiler" on page 10
- "Administering vFiler resources from the hosting filer" on page 22
- "Changing resources for vFilers" on page 23
- "Adjusting the vFiler limit" on page 27
- "Stopping and starting vFilers" on page 29
- "Destroying a vFiler" on page 31
- "Recreating a destroyed vFiler" on page 33
- "Allowing or disallowing protocols for vFilers" on page 34
- "Displaying vFiler status" on page 36
- "Autosupport and syslogd messages about vFilers" on page 37
- "Executing commands and setting options for vFilers" on page 38
- "Effects of filer reboot on vFilers" on page 42
- "Understanding volumes and qtrees on vFilers" on page 43
- "Backing up vFilers" on page 45
- "Understanding LUNs on vFilers" on page 46
- "Networking considerations" on page 50
- "Understanding SnapMirror on the hosting filer" on page 51
- "Understanding SNMP when used with vFilers" on page 52
- "Performance monitoring and statistics" on page 53

Establishing a vFiler

Tasks for establishing a vFiler

Before you can start protocol servers on a vFiler, you must establish a vFiler on your filer.

For detailed information

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

- "Enabling or disabling the vFiler license" on page 11
- "Planning for vFiler creation" on page 12
- "Creating a vFiler" on page 15
- "Understanding the vFiler command family" on page 19
- "Setting up a vFiler manually" on page 20

Enabling or disabling the vFiler license

Effects of enabling the vFiler license

Enabling the vFiler license has the following effects on the filer:

- You can use the `vfiler` and `ipspace` commands.
- Data ONTAP starts logging vFiler status information and sends vFiler status information using the autosupport feature.
- The `routed` daemon is disabled.
- The `ip.match_any_ifaddr` option is set to Off.
- The "vFiler limit" (the number of vFilers you can create on this filer, including vFiler0) is set to a default value between 3 and 11, depending on the memory capacity of the hosting filer. You can change this value; see "Adjusting the vFiler limit" on page 27 for more information.

Prerequisite for disabling the vFiler license

You can disable the vFiler license only when no vFilers other than vFiler0 are on the filer. If there are vFilers on the filer, you must destroy them before disabling the vFiler license.

Effects of disabling the vFiler license

Disabling the vFiler 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 vFiler license

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

Step	Action	
1	**If you want to...**	**Enter this command:**
	Enable the vFiler license	**`license vfiler=`*`code`***
	Disable the vFiler license	**`license vfiler=disable`**

Planning for vFiler creation

Prerequisites for vFiler creation

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

- You must create the units of storage (qtrees or volumes) before creating the vFiler.
- The storage unit for vFiler 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 must not be configured when you create the vFiler.

Storage guidelines

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

- The first storage unit assigned to the vFiler contains the vFiler configuration information. It is called the primary storage unit. Although you can remove storage units from a vFiler at any time after the vFiler is created, the primary storage unit must remain for as long as the vFiler exists.
- The primary storage unit has the same security characteristics it had before it was transferred to the vFiler. When you create a new vFiler, C$ share-level permissions are restricted to administrators only, but file-level security is not modified. The vFiler 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. Data in qtrees and volumes used as other storage units is preserved.
- A volume assigned to a vFiler must not be the filer's root volume. You can, however, assign qtrees in the root volume to a vFiler.
- If the vFiler administrator needs to create qtrees on the vFiler, assign volumes instead of qtrees to the vFiler. This is because qtrees can be created only at the root of a volume.
- If, in the future, you want to move from one filer to another the disks that are used for the vFiler's storage, assign volumes, not qtrees, to the vFiler.
- When managing NFS exports, CIFS shares, quotas, and options, vFiler administrators need to enter the complete path names of the storage resources used by the vFilers in commands and configuration files.

Therefore, choose volume and qtree names appropriately, so that you can share with the vFiler administrators the complete path names beginning with *filer_name*:/vol/*vol_name*.

Naming guidelines

When deciding the vFiler 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 vFiler0 is reserved.

Language guideline

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

Quotas guideline

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

Clustering considerations

Remember the following considerations for establishing vFilers on a cluster:

- You can create up to 32 vFilers on each member of a cluster, depending on the memory capacity of the hosting filers. See "Adjusting the vFiler limit" on page 27 for details.
- Both members of a cluster must have a vFiler license to take over its partner with a vFiler license.
- The vFilers hosted by the filers of the cluster are created and configured independently. That is, each filer can host a different number of vFilers, and the vFiler configuration on the filers can be different from each other.
- In takeover mode, the functioning filer takes over all vFilers created on the failed filer. These vFilers include the vFilers you create and vFiler0. Therefore, for vFilers on the failed filer to work correctly after the takeover,

each network interface used by a vFiler in a cluster must have a partner interface.

SAN considerations

When you create vFilers 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 filer, not on vFilers.
- iSCSI LUNs and initiator groups are supported on all vFilers and managed separately for each vFiler.
- When you create a vFiler on a filer on which iSCSI is licensed, the iSCSI service is automatically started on the vFiler.

  Note

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

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

Creating a vFiler

About this section

This section describes how to create a vFiler in the default IP address space (IPspace). For the definition of IPspace and information about creating vFilers in nondefault IPspaces, see Chapter 3, "Using vFilers in Different IPspaces," on page 55.

State of the vFiler after creation

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

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

The vFiler configuration information is stored in the vFiler's /etc directory. It contains a subset of files that are in the hosting filer'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

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

- Ensure that the network interface to be configured with a vFiler IP address is ready for vFiler creation.
- If iSCSI is licensed on the hosting filer and the storage to be allocated to the vFiler contains LUNs, unmap the LUNs.

  See the *Block Access Management Guide for iSCSI* for instructions.
- Assign and configure vFiler resources.

Ensuring that the network interface is ready

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

Step	Action	
1	**If the IP address for the vFiler 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 interface -alias address`
		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 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 interface 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 resources

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

Note

The procedure that follows assumes that you want to set up the vFiler for networking (and CIFS if necessary) immediately after creating the vFiler. If you want to defer setup, use the `-n` option of the `vfiler create` command, and then follow directions under "Setting up a vFiler manually" on page 20 when you are ready to do the setup. For more information on the `vfiler create` command, see the na_vfiler man page.

Step	Action
1	Enter the following command: **`vfiler create vfiler_name -i ip_address [ -i ip_address ] ... path [ path ] ...`** *vfiler_name* is the name of the vFiler. *ip_address* is an IP address. *path* 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. **Example:** The following example creates a vFiler using two IP addresses, one volume, and one qtree: `vfiler create vfiler1 -i 123.123.123.123 -i 123.123.123.124 /vol/vol1 /vol/vol2/qtree2`
2	Respond to prompts to set up the filer, and to set up CIFS if necessary.

Results of the setup process: The setup process does the following on the new vFiler:

- Starts NFS (if NFS is licensed on the hosting filer) and configures the vFiler's root volume to be exported to the vFiler's administrative host (using an entry in the /etc/exports file)
- Configures the vFiler'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 as a single tree (see "The pseudo-root directory" on page 5 for more information)
- Starts the iSCSI service (if iSCSI is licensed on the hosting filer)

What to do next

You, or the vFiler administrator, can now mount the root volume of the vFiler, edit /etc/exports to suit your needs and rerun the `exportfs` command. See Chapter 4, "File Access Using NFS and CIFS," on page 73 for more information.

Understanding the vFiler command family

Purpose of the vFiler command

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

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

Some `vfiler` commands support the *vfilertemplate* option. Wherever you see *vfilertemplate* in this manual or the na_vfiler man page, you can substitute a vFiler name, a comma-separated list of vFiler names, an IPspace declaration, or an asterisk as a wildcard. If you use the asterisk, the command takes effect on all vFilers, including vFiler0 (the physical filer), 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.

Setting up a vFiler 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 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 until you set it up manually.

The procedure for setting up a vFiler manually is similar to that for setting up a filer. You run the `setup` program, which enables you to

- Specify the administration host for the vFiler
- Specify the password for the root user on the vFiler
- 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 vFilers are in the same time zone as the hosting filer.

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

Setting up the vFiler

To set up the vFiler, complete the following steps.

Step	Action
1	Enter the following command: **`vfiler run vfiler_name setup`** **Result:** The `setup` command displays prompts for you to configure the vFiler. After you respond to all the prompts, configuration files, such as the /etc/exports file, are created in the /etc directory for the vFiler. **Note** Unlike the `setup` command for the filer, the `setup` command for vFilers does not cause NFS to start running. For more information about starting NFS, see Chapter 4, "File Access Using NFS and CIFS," on page 73. If the vFiler runs the CIFS protocol, go to the next step. Otherwise, you are done.
2	Enter the following command: **`vfiler run vfiler_name cifs setup`** **Result:** The `cifs setup` command displays prompts for you to configure CIFS on the vFiler. After you respond to all the prompts, CIFS starts running.

Administering vFiler resources from the hosting filer

Managing vFiler storage from the filer

If you, as the physical filer administrator, need to manage storage resources that belong to a vFiler, but you do not have administrative access to the vFiler, 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 for iSCSI*.

- Temporarily move the resources to the hosting filer (but you cannot move the vFiler's primary /etc volume).
- Temporarily destroy the vFiler.

  This returns ownership of all resources to the hosting filer. No user data is modified when you destroy a vFiler.

Once you have done what you need to do, you can either move the storage resources back to the vFiler or, if you destroyed the vFiler, restore the vFiler using the `vfiler create` *oldname* `-r` *oldfirstpath* command.

oldname is the name of the original vFiler and *oldfirstpath* is the first path that was specified in the original `vfiler create` command that created the vFiler. See the na_vfiler man page for more information.

Changing resources for vFilers

What you can change

You can add, move, or remove resources for vFilers. 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 resources has the following effects:

- After you add storage resources to a vFiler, the resources are moved from the hosting filer to a vFiler.
- After you remove storage resources from a vFiler, the resources are moved from the vFiler to the hosting filer.
- After you add an IP address to a vFiler, 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, the IP address becomes an unassigned IP address.
- After you move resources from one vFiler to another, the resources are removed from the source vFiler and added to the destination vFiler.

Note

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

Requirements for moving and removing resources

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

- The storage unit to be moved or removed cannot be the one containing the vFiler'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 iSCSI* 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 vFilers or removing an IP address from a vFiler:
  - Both the source vFiler and destination vFilers 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 vFilers

Remember the following considerations when you move resources between vFilers:

- If a storage unit that is to be moved from one vFiler to another contains LUNs, you must unmap the LUNs first. See the *Block Access Management Guide for iSCSI* for instructions.
- After you move a storage unit from one vFiler 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, and the user previously represented by this UID can no longer gain access the file.
- If you reassign a volume from one vFiler 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.
- If you reassign a qtree from one vFiler to another, Data ONTAP turns off quotas for the volume containing the qtree on both the source vFiler and the destination vFiler. After the qtree is moved, you can turn quotas on again for the volume.
- All network connections to the vFilers are terminated when resources are being moved.

Adding resources to a vFiler

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

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

Removing resources from a vFiler

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

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

Moving resources between vFilers

To move resources between vFilers, complete the following step.

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

Adjusting the vFiler limit

What the vFiler limit is

The vFiler limit sets the maximum number of vFilers that you can create on the hosting filer. In a cluster, the same limit applies to each cluster node.

Default limits

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

- 3 for filers with less than 1 GB of memory
- 5 for filers with between 1 GB and 3 GB of memory
- 11 for filers with 3 GB of memory or more

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 filer, 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 vFilers. In a cluster, a limit of 11 means that you can create a maximum of 10 vFilers 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 of up to 33 (up to 66 in a cluster), depending on the memory capacity of the hosting filer. The maximum limits are as follows:

- 11 for filers with less than 1 GB of memory
- 26 for filers with between 1 GB and 2GB of memory
- 33 for filers with over 2 GB of memory

Because the limit includes the hosting filer, vFiler0, which always exists, the number of vFilers you can create is one less than the limit.

Example: To raise the number of vFilers you can create to 15, enter the following command on the hosting filer:

```
vfiler limit 16
```

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

Note

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

Decreasing the limit

You can use the `vfiler limit` command to decrease the limit, setting the maximum number of vFilers lower than it now is.

Because the limit includes the hosting filer vFiler0, which already exists, the number of vFilers you can create in each case is one less than the limit.

Example: To reduce the number of vFilers you can create from 15 to 10, enter the following command on the hosting filer:

```
vfiler limit 11
```

Note

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

Caution

Before you reboot, make sure that the number of active vFilers on this filer (and on any cluster partner) is less than the limit. On reboot, the number of vFilers 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 vFilers are configured.

Stopping and starting vFilers

Reasons to stop a vFiler

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

Note

You cannot stop or start vFiler0.

What it means to stop and start a vFiler

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

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

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

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

Stopping vFilers

To stop vFilers, complete the following step.

Step	Action

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

Starting vFilers

To start vFilers, complete the following step.

Step	Action

1	Enter the following command:
	`vfiler start` ***`vfilertemplate`***
	Example: The filer supports two vFilers named vFiler1 and vFiler2. The following command starts all vFilers, 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

Effects of destroying a vFiler

Destroying a vFiler has the following effects:

- All resources associated with the vFiler are released to the hosting filer.
- No data is destroyed.
- All the vFiler's IP addresses are unconfigured and the corresponding entries in the filer's /etc/rc file are removed.
- If the vFiler was not in the same IPspace as the hosting filer, the IP addresses previously owned by the vFiler are not available for use after you destroy the vFiler.
- The effects on quotas are the same as when you move resources from a vFiler to the hosting filer. See "Considerations for moving resources between vFilers" on page 24 for more information about how moving the resources to the hosting filer affects quotas.
- Clients using LUNs experience an interruption of service.

  You must unmap all of a vFiler's LUNs before you can destroy it. The LUNs themselves are not destroyed. You can remap them from the hosting filer, or from another vFiler if you reallocate the storage to that vFiler. For information on mapping and unmapping LUNs, see the *Block Access Management Guide for iSCSI*.

Prerequisites for destroying a vFiler

The vFiler must be stopped before you can destroy it.

Destroying a vFiler

To destroy a vFiler, complete the following steps.

Step	Action
1	Unmap any LUNs that are mapped to the vFiler's storage. See the *Block Access Management Guide for iSCSI* for instructions.
2	Enter the following command to stop the vFiler: **`vfiler stop vfiler_name`**

Step	Action
3	Enter the following command: **`vfiler destroy [ -f ] vfiler_name`** 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 filer.

Recreating a destroyed vFiler

Why recreating a vFiler is easy

When you destroy a vFiler, it's resources are released and the associated configuration information is removed, but your user data remains intact; the data is not destroyed. You can recreate the original vFiler 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's primary storage unit (containing its /etc directory).

Additionally, when creating a vFiler, you can change the original name to a new name, and you can specify a different path name of the storage units.

To recreate a vFiler, complete the following steps.

Step	Action	
1	Specify the path to the original vFiler's primary storage unit. **`vfiler create vfilername -r origin_vfiler_path`** `vfilername` is the name of the new vFiler you are creating. `origin_vfiler_path` is the path of the original vFiler's /etc storage. **Example: `vfiler create vfiler1 -r /vol/vol0/qtree1`**	
	If you want to...	**Then...**
	Map the original path names to the new ones because the paths have been renamed	Enter the following command: **`vfiler create new_vfiler_name -r origin_vfiler -b old_vfiler_name -p /vol/src1:/vol/dst1,/vol/src2:/vol/dst2,...`**

Note

Although other components of the path name can be different, the actual names of the qtrees must not be changed. For example, /vol/orig1/qtree1 can be renamed to /vol/new/qtree1, but not to /vol/new/qtree2.

Allowing or disallowing protocols for vFilers

Protocols that vFilers can use

By default, vFilers can use the protocols that are licensed for the hosting filer. For example, if the hosting filer is licensed for CIFS and NFS, the vFilers created can also use the CIFS and NFS protocols. You can select those protocols that you allow or disallow on the vFilers.

Allowing and disallowing CIFS, NFS, or iSCSI

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

- CIFS
- NFS
- rsh
- iSCSI

To allow the CIFS, NFS, or both protocols on a vFiler, the CIFS, NFS, or both protocols must have an active license on the hosting filer.

Effects of disallowing CIFS, NFS, or iSCSI

If the CIFS, NFS, or iSCSI protocol is running and you disallow it, the protocol remains running until the filer reboots, but packets destined for the vFiler are ignored.

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

- 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 rsh

After you disallow rsh, attempts to connect to the vFiler using rsh fail, regardless of the `rsh.enable` option setting.

After you allow rsh again, you can connect to the vFiler using rsh if the `rsh.enable` option is On.

Allowing or disallowing rsh has no effect on the `rsh.enable` option, and you can turn the `rsh.enable` option on or off regardless of whether rsh is allowed.

Allowing or disallowing protocols for vFilers

To allow or disallow protocols for vFilers, 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: **`vfiler allow vfilertemplate proto=protocol ...`** *protocol* is `nfs`, `cifs`, `isci`, or `rsh`. **Example:** The following example allows the NFS and rsh protocols on the vFiler named vFiler1: `vfiler allow vfiler1 proto=nfs proto=rsh` You are done.	
3	To disallow a protocol, enter the following command: **`vfiler disallow vfilertemplate proto=protocol ...`** *protocol* is `nfs`, `cifs`, `iscsi`, or `rsh`. **Example:** The following command disallows the NFS and rsh protocols on the vFiler named vFiler1: `vfiler disallow vfiler1 proto=nfs proto=rsh`	

Displaying vFiler status

Status information you can display

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

- Whether the vFiler 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

  Technical support might need this ID when diagnosing problems related to vFilers.
- Protocols allowed
- Protocols disallowed

See the na_vFiler man page for more information.

Autosupport and syslogd messages about vFilers

About autosupport messages

A filer licensed for vFilers 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 vFilers are stopped or running, is logged in the /etc/messages file on the hosting filer.

All messages generated by vFilers are logged to the hosting filer's /etc/messages file. These messages are preceded with the name of the vFiler. For example, if the vFiler 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 vFilers

Methods for executing commands for vFilers

You can enter commands for vFilers in the following ways:

- From the context of the vFiler, you can enter commands directly.

  See "Entering commands from the vFiler" immediately following.
- From the hosting filer, you can use the `vfiler run` command to execute commands for the specified vFilers.

  See "Entering commands for vFilers from the hosting filer" on page 39.
- From the client of a vFiler, you can establish an rsh connection with the vFiler and then enter the commands.

  See "Entering a command for a vFiler through rsh" on page 40.

Note

Not all Data ONTAP commands can be executed for vFilers, 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 (see "Entering commands from the vFiler" on page 38) or enter the following command:

`vfiler run` *`vfilertemplate`* `?`

Special considerations and restrictions are spelled out in the "vFiler considerations" section of each command's man pages.

Entering commands from the vFiler

To enter commands for vFilers from the vFiler, complete the following step.

Step	Action
1	To run in the context of the vFiler, enter the following command: **`vfiler context`** ***`vfiler_name`*** **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.

Step	Action
3	To return to the context of the hosting filer, enter the command **`vfiler context vfiler0`**

Entering commands for vFilers from the hosting filer

To enter commands for vFilers from the hosting filer, complete the following step.

Step	Action
1	Enter the following command: **`vfiler run vfilertemplate command`** **Example:** `vfiler run vfiler1 useradmin`

Note

Not all Data ONTAP commands can be executed for vFilers. To see what commands are supported, enter **?** from the context of a vFiler (see "Entering commands from the vFiler" on page 38) or enter the following command:

`vfiler run vfilertemplate ?`

Prerequisites for entering commands through rsh from a vFiler client

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

- The rsh protocol must be allowed for the vFiler. By default, rsh is allowed.
- The rsh protocol must be enabled for the vFiler; that is, the `rsh.enable` option for the vFiler 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. In other words, the client must be one of the hosts specified by the `rsh.access` option for the vFiler.

Commands that you can enter through rsh

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

`?`	`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.

Entering a command for a vFiler through rsh

To enter a command for a vFiler 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 whose IP address is 123.123.123.1: `rsh 123.123.123.1 options`

Options that can be set for vFilers

Not all Data ONTAP options are supported on vFilers. 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`

- `wafl.nt_admin_priv_map_to_root`
- all `wafl.wcc_` options

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

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 filer reboot on vFilers

Effects of filer reboot on protocols

The disallow or allow state of a vFiler is persistent across reboots. For example, if you disallow NFS for a vFiler and then reboot the filer, NFS remains disallowed for the vFiler after the reboot.

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

Effect of filer reboot on the vFiler state

Rebooting the filer causes all stopped vFilers to enter the running state. For example, if you stop a vFiler and then reboot the filer, the vFiler starts running again after the reboot.

Understanding volumes and qtrees on vFilers

Effects of taking a vFiler volume offline

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

- All CIFS shares and NFS exports in the volume are deactivated.
- If the volume contains the /etc directory for a vFiler, the vFiler enters the stopped state.

  After you put the volume back online, the vFiler starts running again.
- All LUNs become unavailable.

Changes required after volumes are renamed

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

Who can create qtrees

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

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: 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 owns a volume, you can change the security styles or oplock settings for the volume and all qtrees on the volume from the vFiler.

- If a vFiler owns qtrees on a volume owned by the hosting filer, you can change the security styles or oplock settings from the vFiler only for the qtrees the vFiler owns.
- If the hosting filer owns a volume that contains qtrees assigned to vFilers, you can change the security styles or oplock settings from the hosting filer only for the qtrees the hosting filer owns.

Differences in qtree command output

The `qtree` command output is different in the following ways depending on where you enter the command:

- If you enter the `qtree` command from the hosting filer, the command displays information about all qtrees on the filer, whether the qtrees are owned by the hosting filer or vFilers.

  If you enter the `qtree` command from a vFiler, the command displays information about qtrees on that vFiler 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, 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 vFilers

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

```
vfiler run * qtree status
```

Backing up vFilers

Considerations

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

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

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

- From a client of a vFiler, you can back up all of that vFiler's data, but no other vFiler's data.
  - A CIFS client can mount "/" from a vFiler and see a virtual tree comprising all of that vFiler's storage units. (See "The pseudo-root directory" on page 5 for more information.)
  - A CIFS client can capture all of a vFiler's data, both CIFs and NFS.
  - An NFS client cannot see a virtual tree for the vFiler.
  - An NFS client can back up all of the vFiler's NFS data, but not its CIFS data.

  Backing up from a client (particularly a CIFS client) is a good way to back up an individual vFiler and keep the vFiler's data separate. It does not allow you to back up all vFilers at once.

Understanding LUNs on vFilers

What LUNs are

LUNs are portions of filer 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 vFilers

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

What is not supported on vFilers

FCP LUNs are supported only on the hosting filer, not on vFilers. For detailed information about FCP LUNs, see the *Block Access Management Guide for iSCSI*. This means

- You can create FCP igroups only on the hosting filer.
- FCP-connected hosts can access only those LUNs that are owned by the hosting filer.
- The `fcp` command does not recognize vFilers.
- You can create only iSCSI igroups on vFilers; you cannot create FCP igroups on vFilers.

iSCSI LUNs and igroups on vFilers

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

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

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

  For instructions for switching the console to the context of a vFiler, see “Executing commands and setting options for vFilers” on page 38.

- A vFiler is aware only of those LUNs in the storage units it owns. When executed on a vFiler, the `lun show` command displays only that vFiler'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 when you create the vFiler or later
  - Destroying a vFiler that owns storage that contains LUNs
  - Moving storage that contains LUNs from one vFiler to another, or between a vFiler and the hosting filer

  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 or replace it for disaster-recovery purposes. For more information, see Chapter 6, "Disaster Recovery and Data Migration," on page 89.

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

  For more information, see Chapter 6, "Disaster Recovery and Data Migration," on page 89.

The iSCSI service on vFilers

In general, the iSCSI service operates on individual vFilers, treating them as though each were a physical filer. But for 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 filer. This is because an iSCSI adapter on a filer has only one identity (there are no vFiler-specific adapter names) and so there is only one set of iSCSI sessions and statistics.

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

The following items are based on the hosting filer:

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

The following group is based on the individual vFiler:

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

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

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

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

  - Configure which iSNS server to use.

- Turn iSNS registration on or off.

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

For more information, see the na_isci man page.

Networking considerations

Effects of a disabled routed daemon

Enabling the vFiler 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 vFilers in the same IPspace share one routing table, you can manipulate the routing table by entering the `route` command from the hosting filer. The `route` command has the following syntax:

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

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

Command for changing routing tables in nondefault IPspaces

For more information about manipulating routing tables in nondefault IPspaces, see Chapter 3, "Using vFilers in Different IPspaces," on page 55.

About the /etc/dgateways file

Only the hosting filer has the /etc/dgateways file. vFilers do not maintain their own /etc/dgateways file.

IPSec on vFilers

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

You can configure each vFiler separately with unique security configuration. When you migrate the vFiler from one hosting filer to another (or replicate it for disaster-recovery purposes), the IPSec configuration is preserved so long as the vFiler'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 filer

Where to enter the snapmirror command

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

Considerations for using the snapmirror command

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

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

Understanding SNMP when used with vFilers

About SNMP

SNMP is not supported on individual vFilers; it is supported only on the hosting filer. You can enable SNMP on the hosting filer to collect data about vFilers.

About vFiler data in the standard MIB and NetApp custom MIB

In the standard MIB, all vFiler data is global. It pertains to the sum of data from all vFilers on the filer, 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.
- UDP statistics include data only from sockets in the default vFiler.
- Quota information is gathered for each volume. If the hosting filer or a vFiler owns a volume with quotas, quota information is provided for the hosting filer or vFiler owning the volume. If a vFiler owns qtrees in a volume that it does not own, no quota information is provided for the vFiler.

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

Performance monitoring and statistics

Commands for displaying filer statistics

The `sysstat` command displays filer statistics, which are the sum of statistics generated by all vFilers, including vFiler0. You cannot use the `sysstat` command to display statistics for particular vFilers.

The `uptime` command displays the uptime for the filer. You cannot use the `uptime` command to display the uptime for particular vFilers.

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 filer, specified vFilers, or the hosting filer, depending on the command format, as explained in the following table.

Command entered from the hosting filer		**Statistics displayed**
For NFS statistics	**For CIFS statistics**	
`nfsstat`	`cifs stat`	Statistics from all vFilers combined
`vfiler run` *`vfilertemplate`* `nfsstat`	`vfiler run` *`vfilertemplate`* `cifs stat`	Statistics from the specified vFilers
`vfiler run vfiler0 nfsstat`	`vfiler run vfiler0 cifs stat`	Statistics from the hosting filer

Using vFilers in Different IPspaces 3

About this chapter

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

Topics in this chapter

This chapter discusses the following topics:

- "Understanding IPspaces" on page 56
- "Advantages of using VLAN tagging for IPspaces" on page 60
- "Requirements for using clusters and IPspaces" on page 61
- "Case studies for IPspaces" on page 63
- "Managing IPspaces on a filer" on page 64
- "Creating a vFiler in a nondefault IPspace" on page 70

Understanding IPspaces

About IPspaces

An IPspace defines a distinct IP address space in which vFilers 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 participation in an IPspace

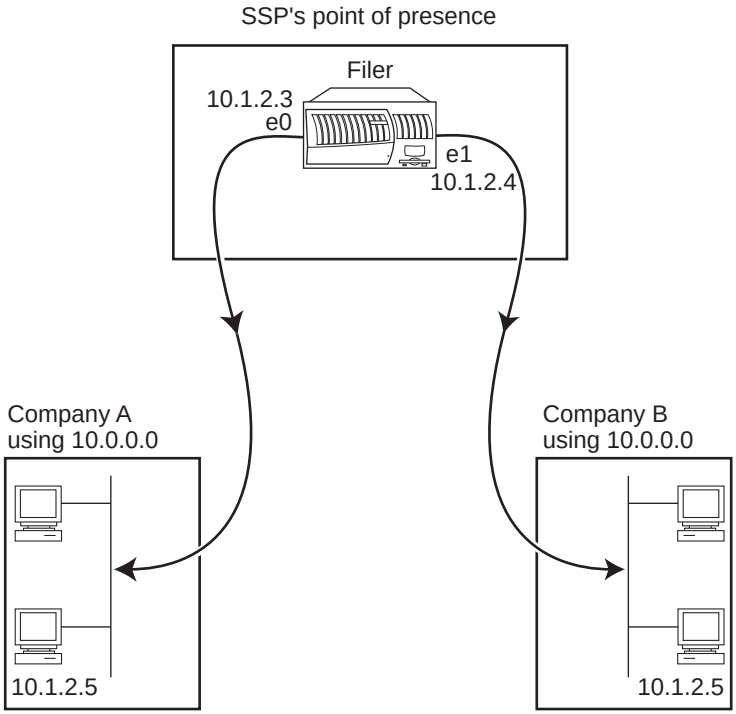
The following guidelines apply to vFiler participation in an IPspace:

- An IPspace can contain multiple vFilers, but a vFiler can belong only to one IPspace.
- Each vFiler in an IPspace must have an IP address that is unique within that IPspace, but a vFiler in one IPspace can have the same IP address as a vFiler in a different IPspace.
- Be careful when assigning IPspace to a vFiler because you cannot change the assignment without destroying the vFiler.

Typical application of IPspaces

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

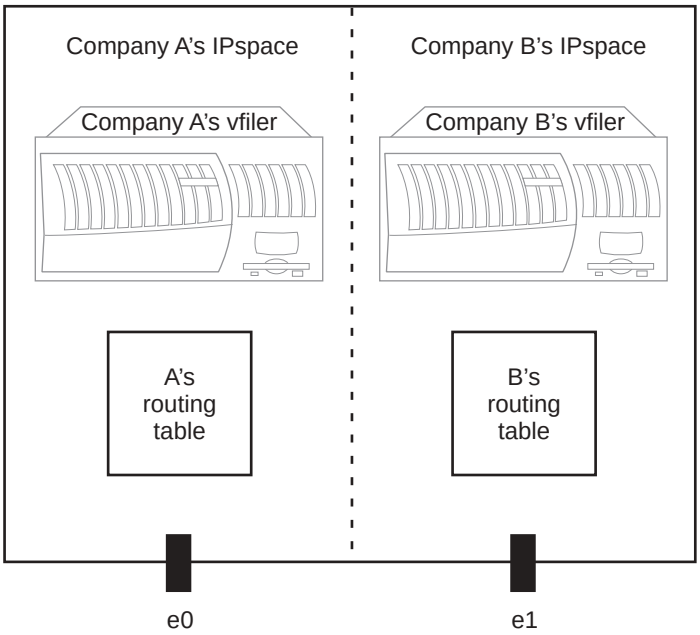
This deployment would work if both companies were using nonprivate IP address ranges; however, as shown in the following illustration, this is not the case.



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

- The two vFilers on the filer at the SSP site have conflicting IP addresses if both companies decide to use the same IP address for their respective vFilers.
- If the two companies agree on using different IP addresses for their vFilers, 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 filer 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 filer).

To alleviate these problems, two IPspaces are defined on the filer—one per vFiler. 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 vFilers 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 vFilers, without conflict.

How interfaces participate in an IPspace

If the filer is licensed for vFilers, 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 vFilers 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 for which the packet is intended.

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

Note

Broadcast packets are restricted to the vFilers within the destination IPspace.

Advantages of using VLAN tagging for IPspaces

Traffic separation

VLAN tagging for IPspaces provides traffic separation from each customer to the filer 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 filer. 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 filer.

Dedicating at least one physical interface per IPspace limits the number of IPspaces that can be set up on a filer to the number of physical interfaces available on the filer. 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 in an IPspace

VLANs inherently confine the broadcast domains. As a result, only vFilers belonging to a VLAN receive broadcasts intended for that VLAN, even if multiple vFilers 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 filers.

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

IPspace assignment requirement

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

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

Specifying partners in an asymmetric cluster setup

In an asymmetric cluster setup, vFiler-IPspace configuration on one cluster partner is different from the other; for example, one of the partners might not have any (or the same number of) vFilers configured in a specific IPspace, while the other partner does.

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 vFilers if the partner fails. In such a configuration, you might not have any or enough storage on this host to configure the same number of vFilers as the primary host. 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 filers, Filer1 and Filer2, are configured as shown in the following table.

Filer	vFiler (associated IP address)	IPspace	Interface name
Filer1	vFiler0 (1.1.1.1)	default	e0
	vFiler1 (2.1.1.1)	ips1	e1
	vFiler2 (3.1.1.1)	ips2	e2

Filer	vFiler (associated IP address)	IPspace	Interface name
Filer2	vFiler0 (1.1.1.2)	default	e0
			e4a
			e4b

To specify partner interfaces on Filer1, 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 Filer2, 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 Filer2 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 appliance at the SSP premises to their intranet.
- They want to ensure that the storage appliance is accessible only to the users they authorize.

Using the IPspace solution available on the filers, the SSP sets up an IPspace per customer, thus creating independent routing domains on a single filer. Additionally, the SSP uses either physical interfaces or VLAN tags to securely forward traffic to appropriate vFilers. 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 filers for each organization of the enterprise. As in the SSP approach, dedicating filers on an organizational basis can not only be expensive but also difficult to maintain. Setting up vFilers on a per-organization basis solves the problem. However, distinct IPspaces for these vFilers are usually not required. Because there is usually a level of trust between different organizations of the enterprise, network access to the various vFilers does not need to controlled. Therefore, all vFilers 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 vFilers using different default routes.

Managing IPspaces on a filer

Command for managing IPspaces

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

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 65
- "Listing IPspaces on a filer" on page 66
- "Assigning an interface to an IPspace" on page 67
- "Destroying an IPspace" on page 69

Creating an IPspace

Guidelines for creating an IPspace

The following guidelines apply to creating an IPspace on a filer:

- You can create up to 51 IPspaces per filer (101 in a cluster).
- You can use an alphanumeric string, 1 to 31 characters long, as the IPspace name.
- All IPspace names you create on a filer must be unique. However, the IPspace names on cluster partners must be the same.

Creating an IPspace

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

Step	Action
1	Enter the following command: **`ipspace create`** ***`ipspacename`*** *ipspacename* is the IPspace name that you want to create. **Example:** You can create an ipspace1 IPspace on a filer using the following command: `ipspace create ipspace1`

Listing IPspaces on a filer

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 `ipspace list` command on a filer that has three nondefault IPspaces, you see the following output:
	`Number of ipspaces configured: 4`
	`default-ipspace (e3 lo)`
	`ipspace1 (e2d)`
	`ipspace2 (e2c)`
	`ipspace3 (e10 e2b sf_vif)`
	Note
	In the preceding output, although loopback interface (lo) is shown in the default IPspace, all IPspaces support loopback traffic. The loopback traffic of each IPspace is completely isolated from the other IPspaces.

Assigning an interface to an IPspace

Requirement for assigning an interface

The interface you assign from one IPspace to another must not have an IP address configured on it.

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 in a nondefault IPspace	Remove the IP address configured for the interface from the vFiler. For more information about removing an IP address from a vFiler, see "Changing resources for vFilers" on page 23. You are done.
	The interface is in the default IPspace	Go to Step 2.

Step	Action	
	If...	**Then...**
	The interface is configured with an IP alias	**1.** Enter the following command to remove the IP alias: **`ifconfig interface -alias address`** *interface* is the name of the interface. *address* 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 interface 0.0.0.0`** *interface* 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 interface-name`** *ipspacename* is the IPspace name to which the interface will be assigned. *interface-name* 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 vFilers associated with the IPspace you want to destroy. Read "Listing IPspaces on a filer" on page 66 to identify any interfaces associated with the IPspace you want to destroy. Read "Managing IPspaces on a filer" on page 67 to assign network interfaces to another IPspace. Read "Displaying vFiler status" on page 36 and "Destroying a vFiler" on page 31 for procedures to identify any vFilers associated with the IPspace and to destroy those vFilers.
2	Enter the following command: **`ipspace destroy`** ***`ipspacename`*** *ipspacename* is the IPspace name to be destroyed.

Creating a vFiler in a nondefault IPspace

Guidelines for creating a vFiler in a nondefault IPspace

When creating a vFiler in a nondefault IPspace, you need to meet the same prerequisites and follow the same guidelines as you would when creating a vFiler in the default IPspace. For more information about the prerequisites and guidelines, see "Planning for vFiler creation" on page 12.

Creating a vFiler in a nondefault IPspace

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

Note

You must follow these steps to configure IP addresses to be used by vFilers in nondefault IPspaces. Make sure you use the `-n` option to 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 65.
2	To assign an interface to be used by the vFiler in the newly created IPspace, complete the steps in "Assigning an interface to an IPspace" on page 67.
3	To verify that each interface to be used by the vFiler is ready to be configured, complete the steps in "Ensuring that the network interface is ready" on page 16.

Step	Action
4	To create the vFiler, enter the following command: **`vfiler create vfiler_name -n -s ipspace -i ip_address [ -i ip_address ] ... path [ path ] ...`** *ipspace* is the IPspace in which the vFiler IP addresses reside. *ip_address* is an IP address. *path* 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. **Example:** The following command creates a vFiler named vFiler1 in the IPspace named ipspace1, using the IP address 123.123.123.123 and the /vol/vol1 volume as resources: `vfiler create vfiler1 -n -s ipspace1 -i 123.123.123.123 /vol/vol1`
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: **`ifconfig interface alias address`** 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, enter the following command: **`vfiler run vfiler_name route add destination`** **Example:** The following example adds a route to the routing table used by the vFiler named vFiler1: `vfiler run vfiler1 route add 123.123.123.124`
9	For each interface used by the vFiler, add the following to the /etc/rc file on the hosting filer: ◆ 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 filer is part of a cluster, edit the /etc/rc file on each cluster partner to ensure that each interface the vFiler uses has a partner interface defined for it.

File Access Using NFS and CIFS 4

About this chapter

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

Topics in this chapter

This chapter discusses the following topics:

- "How to specify path names for NFS exports or CIFS shares" on page 74
- "Preparing the vFiler for NFS" on page 75
- "Preparing the vFiler for CIFS" on page 76

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 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 named vFiler1 uses the hosting filer's /vol/vol1 volume as its primary storage. To share the entire volume, and all other storage owned by the vFiler, 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 mechanism that makes this possible is known as "pseudo-root." See "The pseudo-root directory" on page 5 for more information.

Preparing the vFiler 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 has been created. At the end of the setup, NFS is running (if NFS is licensed on the physical filer). 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, enter the following command: **`nfs on`** ◆ From the hosting filer, enter the following command: **`vfiler run vfiler_name nfs on`** **Result:** The NFS protocol server starts running on the vFiler.

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, enter the following command: **`exportfs -a`** ◆ From the hosting filer, enter the following command: **`vfiler run vfiler_name exportfs -a`** ◆ From a vFiler client that is allowed to connect to the vFiler through rsh, enter the following command: **`rsh vfiler_name exportfs -a`**

Preparing the vFiler for CIFS

Different methods of CIFS management

From the hosting filer, you can use the `vfiler run` command format to issue `cifs` commands for vFilers. From the vFilers, 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 filer

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 command line (by means of the `vfiler context` command; see "Executing commands and setting options for vFilers" on page 38 for more information) or from the hosting filer (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 limit with CIFS

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

About local user accounts for vFilers

From the hosting filer, you can use the `useradmin` command to create local accounts for CIFS users of each vFiler. Each vFiler supports up to 96 local user accounts. The maximum number of vFiler user accounts per filer is 96 times the

maximum number of vFilers for that filer. (For information on the maximum number of vFilers for a particular filer, see “Adjusting the vFiler limit” on page 27.)

Disk Space Management Using Quotas 5

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.

Topics in this chapter

This chapter discusses the following topics:

- "Understanding quotas" on page 78
- "Turning quotas on and off" on page 79
- "How to resize quotas" on page 82
- "Understanding the /etc/quotas file" on page 83
- "Displaying quota status" on page 84
- "Understanding quota reports" on page 85
- "Messages from exceeded quota thresholds and soft quotas" on page 87

Understanding quotas

Types of quotas supported on vFilers

You can apply the same types of quotas—user, group, and tree quotas—on vFilers as you can on a filer.

Who manages quota specifications?

The vFiler administrator specifies the size of each quota in the vFiler's /etc/quotas file. That is, the vFiler 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 resides on a volume owned by the hosting filer, you, the hosting filer administrator, can also specify a quota for the qtree in the hosting filer's /etc/quotas file. The following example shows how the tree quota on the hosting filer affects a vFiler qtree:

Suppose the /vol/vol1/qtree1 qtree is a storage unit of the vFiler, and the /vol/vol1 volume is owned by the hosting filer.

In the /etc/quotas file of the vFiler, the vFiler administrator specifies that this qtree is limited to 20 GB of disk space. In the /etc/quotas file of the hosting filer, 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 filer 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 filer controls the usage of quotas on each volume it owns using the `quota allow` *`volume`* and `quota disallow` *`volume`* commands. Once the hosting filer allows quotas, you can turn quotas on or off using the `quota on` *`volume`* and `quota off` *`volume`* commands, respectively.

Turning quotas on and off

What happens when you create a vFiler

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

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

- Quotas are turned off *for the entire volume* /vol/vol1on 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 filer licensed for vFilers, the hosting filer 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 filer administrators can allow or disallow quotas for a volume.

Effects of disallowing quotas

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

- If quotas are currently turned off, you or the vFiler 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.

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 `quota allow` command, you can turn on quotas for the specified volume from a vFiler. After you enter the `quota disallow` command, vFilers are prevented from turning quotas on for the specified volume. If currently there are vFilers that have quotas on for the volume, quotas are turned off immediately for those vFilers.

How turning quotas on and off affects vFilers

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. 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. Quota on or off states are persistent and stay set after reboots.

Who can turn quotas on and off?

Hosting filer administrators and vFiler 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 filer, you can turn quotas on for the /vol/vol1 volume if vFiler0, which is the hosting filer, owns some storage space in the volume.

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

Turning quotas on or off from a vFiler

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

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

How to resize quotas

Method to resize quotas

You resize quotas on the vFiler in the same way as you resize quotas on a filer.

Which quota records are adjusted?

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

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

Understanding the /etc/quotas file

Format of the /etc/quotas file

The format of the /etc/quotas file for a vFiler is the same format as for a filer. 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.

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

Recommendation: A vFiler administrator should specify a path name in the Type field even though the quota is for a user or group in the filer'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 owns storage space.

Displaying quota status

To display quota status, complete the following steps.

Step	Action	
1	**If you manage the vFiler from...**	**Then...**
	The hosting filer	Enter the following command: `vfiler run vfilertemplate quota`
	The vFiler	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 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 vFilers

A quota report contains information for the vFiler 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 filer is included in the quota report.

About quota reports for the hosting filer

When you display a quota report for a hosting filer, it shows

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

How the filer's tree quota affects the vFiler'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 is restricted by the tree quota specified in the filer's /etc/quotas file, the K-Bytes and Files fields in the vFiler's quota report might not reflect the limits that are in effect for these reasons:

- If the vFiler 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 filer.
- If the vFiler imposes a tree quota on the qtree, the K-Bytes and Files fields in the quota report reflect the limits specified in the vFiler's /etc/quotas file, although the qtree is restricted by the tree quota imposed by the filer.

Example: The /vol/vol1/qtree1 qtree is owned by vFiler1. Suppose the filer's /etc/quotas file specifies that this qtree can use up to 100 GB of disk space, and the vFiler'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 names in the quota report

The `quota report` command syntax for a vFiler is the same as the syntax for a filer, except that it also takes the `-v` argument. If you use the `quota report -v` command, the report displays the vFiler 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 is exceeded, a warning message is logged to the filer console.

Example of the warning message

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

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

Disaster Recovery and Data Migration 6

About this chapter

This chapter covers two distinct but similar sets of tasks:

- Preparing a disaster-recovery, or backup, vFiler and activating it if a disaster occurs.
- Moving a vFiler from one physical filer to another (for example, if you are replacing an old filer with a new one). Moving a vFiler is sometimes called "migration."

Note

Both the source and destination filers must be running Data ONTAP 6.4 or later. However, the source and destination filers can be running different minor releases of Data ONTAP 6.4. For example, one filer can run 6.4.1 while the other filer runs 6.4.2.

Note

The procedures in this chapter do not cover migrating or replicating the hosting filer, 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 90
- "How to create and activate a disaster-recovery vFiler" on page 91
- "How to move a vFiler" on page 92
- "Preparing the destination filer" on page 93
- "Storage and network checklists" on page 104
- "Creating the disaster-recovery vFiler" on page 107
- "Activating the disaster-recovery vFiler" on page 110
- "Reactivating the original vFiler" on page 112
- "Migrating the vFiler" on page 120

Understanding disaster recovery and data migration

Similarities between disaster recovery and data migration

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

What the procedures do and do not cover: In both cases, the procedures that follow help you not only to move or copy the data (via SnapMirror), but also to configure all of the networking needed to serve the data. But they do not cover 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 filers.

Differences between disaster-recovery and data migration

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

- If you are preparing a disaster-recovery vFiler:
  - The original (source) vFiler remains intact and continues to serve data to its clients.
  - The new (destination) vFiler 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.
- If you are moving (migrating) a vFiler:
  - The original (source) vFiler is destroyed after the migration is complete.
  - The new (destination) vFiler becomes active and begins serving data as soon as the migration is complete.

How to create and activate a disaster-recovery vFiler

When to use this section

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

The procedures assume that the original vFiler will remain intact and running unless a disaster puts it out of action. If you want to destroy the original vFiler after the new vFiler is up and running, follow the procedures listed under "How to move a vFiler" on page 92 instead.

What to do

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

Stage	Description
1	Qualify and prepare the destination filer. Perform the "Storage checks and actions" on page 93 and the "Network checks and actions" on page 96.
2	Create the disaster-recovery vFiler. See "Creating the disaster-recovery vFiler" on page 107.

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

Stage	Description
1	Activate the disaster-recovery vFiler. See "Activating the disaster-recovery vFiler" on page 110.
2	To reactivate the original vFiler after the damage caused by the disaster has been repaired, follow directions under "Reactivating the original vFiler" on page 112.

How to move a vFiler

When to use this section

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

The procedures assume that you want to destroy the original vFiler once the new vFiler is up and running. If you want to leave the original vFiler intact and create a backup copy of it for disaster-recovery purposes, use the procedures listed under “How to create and activate a disaster-recovery vFiler” on page 91 instead.

What to do

To move a vFiler, follow this process.

Stage	Description
1	Qualify and prepare the destination filer. Perform the “Storage checks and actions” on page 93 and the “Network checks and actions” on page 96.
2	Move the vFiler. See “Migrating the vFiler” on page 120.

Preparing the destination filer

When to use this section

Perform the "Storage checks and actions" that follow and the "Network checks and actions" on page 96 whenever you intend to replicate a vFiler, whether you are moving the vFiler or preparing a disaster-recovery vFiler.

Storage checks and actions

Perform the following checks, and the accompanying actions if necessary, before moving (migrating) a vFiler or setting up a disaster-recovery vFiler. Photocopy the worksheet that follows this section and fill out the storage items as you go along.

Note

The storage checks are not necessary if you are going to use disk ownership technology to migrate the vFiler.

Step	Action
1	Check that the destination filer has enough storage space to hold the source vFiler's volumes. On the source filer, use the `vfiler status -r` command to see what volumes the vFiler is using. Then use the `df` 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 `df` on the destination filer to check. Fill in the `df` results on the worksheet. **Note** If the source and destination filers use different-sized disks and have different block sizes, you will need to adjust the `df` numbers accordingly.

Step	Action	
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 `vol add` command to add new disks to the destination volumes.
3	Check to see if the destination filer has the same volume structure as the source. The volumes to be used by the destination vFiler must have the same path names as those used by the source vFiler.	
4	**If...**	**Then...**
	The destination filer has volumes whose path names match those used by the source vFiler	Continue with Step 5.
	The destination filer does not have matching volume path names	Use the `vol create` command on the destination filer to create volumes whose names match those being used by the source vFiler, or `vol rename` to rename a volume.
5	Check to see if there are 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	Delete the matching qtrees in the destination volumes. To remove a qtree, delete it from a client as you would a directory or folder.
	There are no matching qtree names in the destination volumes	Continue with Step 7.
7	Check to see if quotas are being enforced from the hosting filer. Quotas enforced from the vFiler will be copied to the new vFiler, but quotas enforced from the hosting filer will not. To see where quotas are being enforced from, enter the command **`quota report`** from the hosting filer.	
8	**If...**	**Then...**
	The output of the `quota report` command shows that quotas for qtrees used by the vFiler are being enforced from the hosting filer	**a.** Print out a copy of the filer's quota file: /vol/vol0/etc/quotas. **b.** Copy the relevant entries into the destination filer's /vol/vol0/etc/quotas file.
	No quotas for qtrees used by the vFiler are being enforced from the hosting filer	Continue with Step 9.
9	Continue with "Network checks and actions" on page 96.	

Network checks and actions

Perform the following checks, and the accompanying actions if necessary, before moving (migrating) a vFiler or setting up a disaster-recovery vFiler. Photocopy the worksheet that follows this section and fill out the network items as you go along.

Step	Action
1	Check to see if the destination vFiler can take over the source vFiler's IP addresses. Use the command `ifconfig -a` on the source vFiler and on the destination filer to display information about all their network interfaces. You can reuse the source IP addresses and aliases on the destination vFiler if the destination vFiler will be on the same subnet as the source vFiler.

Step	Action	
2	**If...**	**Then...**
	The source and destination filers are on the same subnet **Note** This is the default case assumed by the `vfiler migrate` command.	Continue with Step 3.
	They are on different subnets	**a.** Obtain as many new IP addresses for the destination vFiler as are in use on the source vFiler. **Note** You might need to replicate subnet-separation arrangements that exist on the source vFiler. For example the source vFiler 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 `vfiler dr configure` command will prompt you for these addresses; see "Creating the disaster-recovery vFiler" on page 107. The `vfiler migrate` command will also prompt you for these addresses, but you might then need to run `setup` on the destination vFiler. See "If you have moved the vFiler to a different subnet or Windows domain" on page 124.

Step	Action	
3	Check whether the source vFiler uses the default IPspace. Use the command `ipspace list` on the source vFiler to display information about IPspaces and the interfaces assigned to them. For more information about IPspaces, see Chapter 3 on page 55.	
4	**If...**	**Then...**
	`ipspace list` reports `default-ipspace`	Continue with Step 5.
	`ipspace list` reports something other than `default-ipspace`	**a.** Use the command `ipspace create` *`ipspacename`* to create a corresponding IPspace with the same name on the destination filer. **b.** Use the command `ipspace assign` *`ipspacename interfacename`* 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 will have access to the same NIS servers as the source. **Note** You can skip this check if the source and destination vFilers are on the same subnet. Use the command `nis info` on the source vFiler to see what NIS servers are available to it. (The `ypwhich` command shows which server the filer is currently bound to.)	

Step	Action	
6	**If...**	**Then...**
	The destination vFiler can use the same NIS servers as the source vFiler **Note** This is the default case assumed by the `vfiler migrate` command.	Continue with Step 7.
	The destination vFiler cannot use same NIS servers	**a.** Find out which NIS servers are available to the destination filer. **b.** Make a note of the IP addresses of those servers on the worksheet. The `vfiler dr configure` command will prompt you for these addresses; see "Creating the disaster-recovery vFiler" on page 107. The `vfiler migrate` command will *not* prompt you for these addresses. If you move a vFiler to a different subnet, you might need to run `setup` on the destination vFiler. See "If you have moved the vFiler to a different subnet or Windows domain" on page 124.

Step	Action
7	Check to see if the destination vFiler will have access to the same DNS servers as the source. **Note** You can skip this check if the source and destination vFilers are on the same subnet. Use the command `dns info` on the source vFiler to see what DNS servers are available to it.

Step	Action	
8	**If...**	**Then...**
	The destination vFiler can use the same DNS servers as the source vFiler **Note** This is the default case assumed by the `vfiler migrate` command.	Continue with Step 9.
	The destination vFiler cannot use the same DNS servers	**a.** Find out which DNS servers are available to the destination filer. **b.** Make a note of the IP addresses of those servers on the worksheet. The vFiler `dr configure` command will prompt you for these addresses; see "Creating the disaster-recovery vFiler" on page 107. The `vfiler migrate` command will *not* prompt you for these addresses. If you move a vFiler to a different subnet, you might need to run `setup` on the destination vFiler. See "If you have moved the vFiler to a different subnet or Windows domain" on page 124.
9	Check to see if the destination vFiler 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 can use the same WINS servers and Windows security network as the source vFiler	Continue with Step 11.
	The destination vFiler 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 will be in. **b.** Make a note of this information on the worksheet. When you activate the disaster-recovery vFiler, you will need to configure it into the new domain. See "Activating the disaster-recovery vFiler" on page 110. If you move a vFiler into a different domain, you will need to configure it into the new domain. See "If you have moved the vFiler to a different subnet or Windows domain" on page 124.
11	Check to see if the destination vFiler can use the same trusted host for vFiler administration as the source vFiler.	

Step	Action	
12	**If...**	**Then...**
	The destination vFiler can use the same trusted host as the source vFiler	You are done.
	The destination vFiler 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, or after moving the vFiler. See "Creating the disaster-recovery vFiler" on page 107 or "If you have moved the vFiler to a different subnet or Windows domain" on page 124.

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 earlier in this section. (By photocopying them you will keep the originals clean for next time.)

STORAGE CHECKLIST:

How much disk space is used and available on the volumes? (See Step 1 of the "Storage checks and actions" on page 93.)

`df` of source filer's volumes: _____

`df` of destination filer's volumes: _____

Have you added a sufficient number of 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 vFilers, is each destination volume free of qtree names that duplicate source qtree names? (See Step 6 of the "Storage checks and actions".)_____

Have you copied filer-based quota information from the source to the destination filer's /etc/quotas file if necessary? (See Step 8 of the "Storage checks and actions".)_____

NETWORKING CHECKLIST:

Are there enough IP addresses available for the vFiler on the destination network? (See Step 2 of the "Network checks and actions" on page 96.)

_____

old interface:_____ new interface:_____

old interface:_____ new interface:_____

old interface:_____ new interface:_____

old interface:_____ new interface:_____

Check syntax carefully; interface names are case-sensitive.

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 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, you are ready to create the destination vFiler.

- If you are preparing a disaster-recovery vFiler, follow directions under "Creating the disaster-recovery vFiler" on page 107.

  In this case the original vFiler remains intact and running.

- If you are moving a vFiler from one physical filer to another, follow directions under "Migrating the vFiler" on page 120.

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

Creating the disaster-recovery vFiler

Before you start

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

- You have performed all the preparation steps listed under "Storage checks and actions" on page 93 and "Network checks and actions" on page 96.
- SnapMirror is licensed and enabled on both the source and the destination filer.
- The source and destination filers can communicate with each other over the network (for example, by means of DNS lookup or entries in /etc/hosts).
- The source filer can communicate with the destination filer via the `rsh` command.
- If any of the source qtrees or volumes contain LUNs (virtual disks), the destination filer is running a version of Data ONTAP that supports LUNs.
  - If the source vFiler owns iSCSI LUNs, the destination vFiler must be running a version of Data ONTAP that supports iSCSI LUNs on vFilers.
- The destination volumes are online.
- You know the source filer's administrative ID and password.

Creating the vFiler

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

Step	Action
1	*On the destination filer*, enter the command **`vfiler dr configure`** ***`source_vfiler@source_filer`***
2	Respond to the login prompt with a valid administrative ID and password for the source filer.
3	Respond to the IP address and binding prompts, using the addresses you entered on the "new interface" lines on the worksheet.
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.

Step	Action
5	Monitor progress by means of the following command: **`vfiler dr status source_vfiler@source_filer`** **Note** The `vfiler dr configure` command might take some time to complete, especially if a source qtree has many millions of inodes. When the `vfiler dr status` command reports all the storage units of the source vFiler as `snapmirrored`, proceed to the next step. **Note** The disaster-recovery vFiler now exists, but has not been started. See "What the vFiler dr configure command does" on page 109 for more information.
6	If you copied quota information into the destination filer's /etc/quotas (see Step 8 of "Preparing the destination filer" earlier in this chapter) activate the quotas on that filer. For each volume you are activating quotas on, use the following command: **`quota on volume`**
7	Edit the disaster-recovery vFiler's /etc/hosts.equiv file, adding the name of the trusted host for administering the disaster-recovery vFiler. (This is the host name 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, `adminhost joe_smith`
8	Add the path to the root volume and the name of the trusted host to the disaster-recover vFiler's /etc/exports file. **Example:** `/vol/vol0 access=adminhost, root=adminhost`

Step	Action
9	If the vFiler's storage units contain iSCSI LUNS, reconfigure iSCSI authentication. See the *Block Access Management Guide for iSCSI* for instructions.

What the vFiler dr configure command does

The `vfiler dr configure` command does the following:

- Checks that the destination filer is capable of receiving the source data.
- Configures and runs SnapMirror to copy the data from the source to the destination vFiler.
  - iSCSI LUNs (including the LUN maps) are copied from the source vFiler to the destination vFiler.
  - Initiator groups (igroups) and the iSCSI configuration, including node names and the iSCSI service state, are also copied to the destination vFiler.
  - iSCSI authentication is *not* copied to the destination vFiler.
- Saves the IP configuration information you supply (see Step 3 under "Creating the vFiler" on page 107).
- Saves the NIS and DNS server information you supply (see Step 4 under "Creating the vFiler" on page 107).
- Saves the quota information from the source vFiler's /etc/quotas file.
- Creates a special type of vFiler, known as a DR backup vFiler, on the destination filer.

  This vFiler is stopped and cannot be started except as described under "Activating the disaster-recovery vFiler" on page 110. Before activation, the vFiler will respond only to the `vfiler status`, `vfiler dr delete`, and `vfiler dr configure` commands. You cannot use `ifconfig` to configure its addresses.

Activating the disaster-recovery vFiler

To activate the vFiler in the event of a disaster, perform the following steps.

Step	Action
1	On the destination filer, 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. These include remounting volumes and qtrees.

What activation does

The `vfiler dr activate` command does the following:

- Breaks the SnapMirror relationships between the source and destination filers.
- Converts the vFiler into an ordinary vFiler that can be started and that responds to all the commands vFilers support.
- Brings LUNs online.
- Configures IP bindings on the destination vFiler 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 is removed from the destination /etc/rc.)
- 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.

Deleting the disaster-recovery vFiler

If you need to delete the disaster-recovery vFiler, you can do so, at any time after you have verified that the configuration process is complete, by running the following command on the destination filer:

`vfiler dr delete` *`source_vfiler@source_filer`*

This will also remove the SnapMirror relationships, and any other configuration information related to the disaster-recovery vFiler, from the source vFiler.

Reactivating the original vFiler

When to use this section

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

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

- If you have recovered the original filer, follow instructions under "Ways to reactivate the original vFiler" immediately following.
- If you are bringing up a replacement filer, follow directions under "Recreating the vFiler on a replacement filer" on page 116.

Unsupported platforms

Because open firmware (OFW) is not supported for the F85 and F87 filers, the instructions for reactivation of the original vFiler after a temporary failure do not apply.

Ways to reactivate the original vFiler

There are two ways to reactivate the original vFiler:

- If the filer was not damaged but suffered only a temporary failure, and you are comfortable using SnapMirror commands, follow the procedure "Reactivating the vFiler using SnapMirror commands".

  This method is a little more complicated but involves less data movement.
- If the filer was severely damaged, or you are not comfortable using SnapMirror commands, follow the procedure "Reactivating the vFiler using vFiler dr commands" on page 115.

  This method is less complicated but involves more data movement.

Reactivating the vFiler using SnapMirror commands: To reactivate the original vFiler on the original filer, perform the following steps.

Step	Action
1	Boot the original filer 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: **`setenv no-vfiler-ips? true`** This ensures that the filer does not try to bind IP addresses already being used by the disaster-recovery vFiler. When the filer boots, the original vFiler 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, enter the following command on the *original filer*: **`snapmirror resync -S`** ***`disaster_recovery_filer:/pathname production_filer:/pathname`*** **Example:** `snapmirror resync -S drfiler:/vol/vfiler1/qtree1 prfiler:/vol/vfiler1/qtree` **Troubleshooting:** If the `snapmirror resync` command fails, telling you that there are no matching snapshots, you have probably accidentally deleted the snapshots 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.
4	Find out if the `snapmirror.access` option on the *disaster-recovery filer* is set to `legacy`. Enter the following command on the disaster-recovery filer: **`options snapmirror`**

Step	Action	
5	**If...**	**Then...**
	The `options snapmirror` command returns `snapmirror.access legacy`	Edit the file /etc/snapmirror.allow and add the host name of the *original filer* if it is not already there.
	The `options snapmirror` command returns a list of host names, and the name of the original filer is not in the list	Use the `options snapmirror` command to add the host name of the *original filer.* **Example:** **`options snapmirror.access host=fridge,toaster,prfiler`**
6	Run `setup` *on the disaster-recovery vFiler* and unconfigure its IP addresses.	
7	Update the data on the original vFiler. For each volume and qtree owned by the original vFiler, enter the following command *on the original filer*: **`snapmirror update -S`** ***`disaster_recovery_filer:/pathname production_filer:/pathname`*** **Example:** `snapmirror update -S drfiler:/vol/vfiler1/qtree1 prfiler:/vol/vfiler1/qtree1`	
8	Stop SnapMirror transfers to the disaster-recovery vFiler. For each volume and qtree owned by the original vFiler, enter the following command *on the original filer*: **`snapmirror quiesce`** ***`pathname`*** **Example:** `snapmirror quiesce /vol/vfiler1/qtree1` **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: **`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, enter the following command *on the original filer*: **`snapmirror break`** ***`pathname`*** **Example:** `snapmirror break /vol/vfiler1/qtree1`
11	*On the original vFiler*, run `setup` to configure the vFiler'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 o*n the original vFiler* 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 *on the original vFiler.*
14	Destroy the disaster-recovery vFiler. *On the disaster-recovery filer*, enter the following command: **`vfiler destroy`** ***`vfilername`*** *vfilername* is the name of the disaster-recovery vFiler.
15	Recreate the disaster-recovery vFiler. From the *disaster-recovery filer*, perform the procedure "Creating the disaster-recovery vFiler" on page 107.

Reactivating the vFiler using vFiler dr commands: To reactivate the original vFiler on the original filer, perform the following steps.

Step	Action
1	Boot the original filer 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: **`setenv no-vfiler-ips? true`** This ensures that the filer does not try to bind IP addresses already being used by the disaster-recovery vFiler.
3	At the `OK` prompt, enter **`boot.`**
4	Destroy the original vFiler. Enter the following command *on the original filer*: **`vfiler destroy`** ***`vfilername`*** *vfilername* is the name of the original vFiler.
5	Follow the procedure for "Creating a replacement vFiler on the original filer or its replacement" on page 117.

Recreating the vFiler on a replacement filer

To recreate the original vFiler on a replacement filer, perform the following steps.

Step	Action
1	Boot the replacement filer.
2	Follow the procedure for "Creating a replacement vFiler on the original filer or its replacement" on page 117.

Creating a replacement vFiler on the original filer or its replacement

Follow these steps to create a replacement vFiler on the original (production) filer or its replacement. Perform this procedure only after completing the preparatory steps under "Recreating the vFiler on a replacement filer" on page 116, or under "Reactivating the vFiler using vFiler dr commands" on page 115.

Step	Action
1	Prepare the new vFiler: Perform the preparation steps under "Preparing the destination filer" on page 93. **Note** Remember that you are performing these checks to inventory the resources of the *original filer or its replacement*.
2	Create the new vFiler, following directions under "Creating the vFiler" on page 107. **Note** Perform those steps on the *original filer or its replacement*. After performing those steps, you have a production vFiler and a backup vFiler, as you did before the disaster, but their roles are reversed: the vFiler you have just created is now the backup, and what was originally the disaster-recovery vFiler is still managing the live data. This might or might not suit your needs. ◆ If you want to put things back the way they were before the disaster, continue with the next step. ◆ If you don't care about the roles of the two vFilers, you are done.
3	Run `setup` *on the disaster-recovery* vFiler and unconfigure its IP addresses.
4	Update the data on the new vFiler. For each volume and qtree owned by the new vFiler, enter the following command *on the original filer or its replacement:* **`snapmirror update -S`** ***`disaster_recovery_filer:/pathname production_filer:/pathname`*** **Example:** `snapmirror update -S drfiler:/vol/vfiler1/qtree1 prfiler:/vol/vfiler1/qtree1`

Step	Action
5	Activate the new vFiler, following directions under "Activating the disaster-recovery vFiler" on page 110. **Note** Perform these steps on the *original filer or its replacement*.
6	Destroy the disaster-recovery vFiler by entering the following command *on the disaster-recovery hosting filer*: **`vfiler destroy vfilername`** *vfilername* is the name of the original vFiler
7	Recreate the disaster-recovery vFiler. From the *original filer or its replacement*, perform the procedure "Creating the disaster-recovery vFiler" on page 107.

Migrating the vFiler

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 93 and "Network checks and actions" on page 96.

  Note
  The storage checks are not necessary if you are going to use disk ownership technology to migrate the vFiler.

- SnapMirror is licensed and enabled on both the source and the destination filer.
- The source and destination filers are on the same subnet.

  Note
  If the filers are not on the same subnet, you will need to run `setup` (and `cifs setup` if necessary) on the new vFiler after the migration, to configure authority servers as specified on your worksheet. See "If you have moved the vFiler to a different subnet or Windows domain" on page 124 for details.

- The source and destination filers can communicate with each other over the network (for example by means of DNS lookup or entries in /etc/hosts).
- The source filer can communicate with the destination filer via the `rsh` command.
- If any of the source qtrees or volumes contain LUNs (virtual disks), the destination filer is running a version of Data ONTAP that supports virtual disks.
  - If the source vFiler owns iSCSI LUNs, the destination vFiler must be running a version of Data ONTAP that supports iSCSI LUNs on vFilers.
- The destination volumes are online.
- You know the source filer's administrative ID and password.

How migrating a vFiler affects clients

If you migrate a vFiler to another filer 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, an initiator hides this brief disruption from applications accessing the LUNs. To the applications, access continues without interruption.

For information on moving a vFiler to a filer on a different subnet, see "If you have moved the vFiler to a different subnet or Windows domain" on page 124.

Caution

The procedure that follows destroys the original vFiler after replicating it on the destination filer. If you want to leave the original vFiler intact and create a backup copy of it for disaster-recovery purposes, use the procedures listed under "How to create and activate a disaster-recovery vFiler" on page 91 instead.

Migrating the vFiler

To migrate the filer, perform the following steps.

Step	Action

1	*On the destination filer*, enter the following command:
	`vfiler migrate start source_vfiler@source_filer`
	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:
	`vfiler migrate source_vfiler@source_filer`
	If you do this, skip Step 4 and Step 5.

2	Respond to the login prompt with a valid administrative ID and password for the source filer.
3	Respond to the IP address and binding prompts, using the addresses you entered on the "new interface" lines on the worksheet.

Step	Action
4	Monitor the progress of the migration with the following command: **`vfiler migrate status`** ***`source_vfiler@source_filer`*** **Note** The `vfiler dr migrate` command might take some time to complete, especially if a source qtree has many millions of inodes.
5	When the `status` command reports that SnapMirror has replicated all the storage units of the source vFiler, complete the migration by entering the following command: **`vfiler migrate complete`** ***`source_vfiler@source_filer`*** **Note** If you want to cancel the migration, you can do so now by issuing the following command on the destination filer: **`vfiler migrate cancel`** ***`source_vfiler@source_filer`*** This destroys the destination vFiler and removes the SnapMirror and other migration-related configuration information from the source vFiler. **Note** If the `vfiler migrate complete` command reports that the process cannot complete, wait a few minutes and then enter the same `vfiler migrate complete` command again.
6	If you copied quota information into the destination filer's /etc/quotas file (see Step 8 of "Preparing the destination filer" earlier in this chapter) activate the quotas on that filer. For each volume you are activating quotas on, use the following command: **`quota on`** ***`volume`***
7	Make adjustments on the clients. These will include remounting qtrees, but volume-level mounts should be intact.

Step	Action
8	If you have moved the vFiler to a different subnet, follow the steps in "If you have moved the vFiler to a different subnet or Windows domain" on page 124.
9	If the vFiler's storage units contain iSCSI LUNS, reconfigure iSCSI authentication. See the *Block Access Management Guide for iSCSI* for instructions.

What the vFiler migrate commands do

The `vfiler migrate start` command does the following:

- Checks that the destination filer is capable of receiving the source data.
- Configures and runs SnapMirror to copy the data from the source to the destination vFiler.
- Saves the IP configuration and binding information you supply (see Step 3 in "Creating the vFiler" on page 107).
- Saves the quota information from the source vFiler's /etc/quotas file.

The `vfiler migrate complete` command does the following:

- Stops the source vFiler.
- Updates the data on the destination vFiler.
- Breaks the SnapMirror relationships.
- Configures IP bindings on the destination vFiler 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 is removed from the destination /etc/rc.)
- Configures any quota information saved by the `vfiler migrate start` command.
- Destroys the source vFiler.
- Brings LUNs online using migrated LUN maps and igroups.

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

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

Perform the following steps *on the destination vFiler.*

Step	Action
1	To configure NIS and DNS servers, run `setup`.
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	To change the trusted host, do the following: **a**. Edit the vFiler's /etc/hosts.equiv file, adding the name of the trusted host for administering the vFiler. (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 `adminhost joe_smith` **b**. Add the path to the root volume and the name of the trusted host to the vFiler's /etc/exports file. **Example:** `/vol/vol0 access=adminhost, root=adminhost`
5	Make adjustments on the clients. These will include remounting volumes and qtrees.

Virus Protection for CIFS 7

About this chapter

This chapter discusses how to perform virus scanning for vFilers that run the CIFS protocol.

Topics in this chapter

This chapter discusses the following topics:

- "Understanding virus scanning for vFilers" on page 126
- "Configuring virus scanning for vFilers" on page 127

Understanding virus scanning for vFilers

Who can configure virus scanning

You can specify how virus scanning works on files owned by the hosting filer and files owned by vFilers. vFiler administrators, however, cannot configure virus scanning for vFilers.

Filers with which virus scanners can be registered

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

Virus scanning for vFiler0

Virus scanning works for vFiler0 in the same way as it does for a filer not licensed for vFilers. See the *Data Protection Guide* for how to configure virus scanning for a filer.

Requirements for virus scanning for a vFiler other than vFiler0

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

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

Effect of virus scanner availability on CIFS access

CIFS clients of the vFiler are not allowed to open any files on the vFiler if all the following conditions apply:

- Virus scanning is enabled.
- No virus scanner is available for the vFiler.
- For the vFiler, the `mandatory_scan` option for `vscan` is set to On.

Configuring virus scanning for vFilers

Configuring virus scanning

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

Step	Action
1	Enter the following command to specify the virus scanner for scanning files on a vFiler: **`vfiler run vfilertemplate vscan options use_host_scanners { on | off }`** Set the `use_host_scanners` option to On if you allow the vFiler to use the virus scanner registered with the hosting filer. The vFiler uses the virus scanner registered with the hosting filer only when the virus scanner registered with the vFiler is unavailable. Set the `use_host_scanners` option to Off if you do not want to allow the vFiler to use the virus scanner registered with the hosting filer. **Note** The `use_host_scanners` option is applicable only to a vFiler you created. You cannot set it on vFiler0 or a filer.
2	Enter the following command to enable or disable virus scanning: **`vfiler run vfilertemplate vscan { on | off }`**

Index

Symbols

/etc directory
    configuration files in 15
    effects of vFiler creation on 12
/etc/dgateways file 50
/etc/exports file
    changing path names in after renaming volume 43
    exporting all file systems in 75
/etc/messages file 37
/etc/quotas file
    format of 83
    language for encoding 13
/etc/snapmirror.conf file 51
/etc/usermap.cfg file, language for encoding 13

A

activating disaster-recovery vFiler 110
adding and removing vFilerresources, effects of 23
addresses relative to IPspace 58
administering resources from the hosting filer 22
after a disaster
    reactivating original vFiler 112
    summary of tasks 91
allowing or disallowing
    CIFS or NFS, effects of 34
    iSCSI, effects of 34
    protocols, steps for 35
    rsh, effects of 34
assigning interfaces to IPspaces 67
autosupport messages for vFilers 37

B

backing up vFilers 45
broadcast packets in an IPspace 59

C

checklists for migration, disaster recovery
    networking 105
    storage 104
CIFS
    local user accounts 76
    path names for shares 74
    prefdc command (domain controller) 110
    statistics 53
    virus scanning 126
cifs
    path names 74
    preparing for 76
    setup command, syntax for 21
    setup command, when to run 20
    stat command 53
    virus scanning 126
clients
    effect of vFiler move on 120
clusters
    considerations when used with IPspaces 61
    licensing vFilers for 13
    naming IPspaces for 61
commands for vFilers
    add 25
    allow 35
    context 38
    create 18
    destroy 32
    disallow 35
    dr activate 110
    dr configure 107
    dr delete 111
    executable from hosting filer 38
    migrate 121
    migrate complete 122
    migrate start 121
    move 26
    prerequisites for entering through rsh 39
    remove 25
    run 38
    setup 20, 75
    start 30
    status 93
    stop 29

ways to execute 38
consolidating multiple servers 2
context of vFiler
switching to 38
creating disaster-recovery vFiler 107
creating vFilers
in nondefault IPspaces 70
options with default settings 15
prerequisites for 12
required status of network interface 16
results of 15

D

daemon
disabling the routed 11
effects of disabling the routed 50
enabling the routed 11
data migration 89
explanation 90
using vFilers for 3
data migration between filers 3
Data ONTAP tasks 6
decreasing the limit for vFilers per filer 28
default IPspace, interfaces in 58
default limits for vFilers per filer 27
default vFiler, definition of 4
deleting disaster-recovery vFiler 111
destination filer
network checks for migration 96
preparing for migration or disaster 93
storage checks for migration 93
destroying IPspaces 69
destroying vFilers
effects on quotas 31
meaning of 31
prerequisites for 31
steps for 31
disabling vFiler license
effects of 11
prerequisites for 11
disallowing or allowing
CIFS or NFS, effects of 34
iSCSI, effects of 34
protocols, steps for 35
rsh, effects of 34
disaster recovery 89
explanation 90
network checks 96
quotas 95
reactivating original vFiler 112
storage checks 93
summary of tasks 91
disaster-recovery vFiler
activating 110
creating 107
deleting 111
preparing to create 93
disks for vFilers, moving 12
displaying
quota reports 85
quota status 84, 95
distinct IP address space 56
DNS servers
and migration, disaster recovery 101
and vFiler 3
domains
multiple 2

E

enabling vFiler license, effects of 11
entering commands from the vFiler 38
enterprises
using IPspaces 63
using vFilers for 3
establishing a vFiler, major steps for 10

F

FCP LUNs 46
filer data migration 3
filer partitioning 2
filer reboot, effects on vFilers 42
filer resources
partitioning 3
FilerView, configuring resources with 17

H

hosting filer

access to data on vFilers 4
administering vFiler from 22, 39
administration tasks 6
cifs commands on 76
definition of 4
format of quota reports on 85
inventorying for vFiler replication 93
managing vFiler storage from 22
managing vFilers from 6
quotas not copied to new vFiler 95

I

incoming packets for an IPspace 59
increasing the limit for vFilers per filer 27
interfaces, assigning to IPspaces 67
IP addresses
and migration, disaster recovery 96
assigned to vFilers 16
configuring 16
in different IPspaces 70
planning for when creating vFilers 12
removing from a vFiler 23
removing from an interface 67
unconfiguring 118
IP alias, removing 16
IPSec, configuring vFiler for 50
ipspace command 64
IPspaces
and migration, disaaster recovery 98
assigning interfaces to 67
case studies of 63
considerations when used with clusters 61
creating 65
creating vFilers in nondefault 70
destroying 69
destroying vFilers in different 31
guidelines for using 56
incoming and outgoing packets in 59
interfaces in 58
listing 66
managing using the ipspace command 64
names for in a cluster 61
restricting broadcast packets in 59
routing tables for 50, 56, 58
typical applications of 56
used with VLAN tagging 60
iSCSI and vFilers 48
iSCSI LUNs 46
iSCSI service on vFilers 48

L

language, guidelines for 13
license for MultiStore, enabling or disabling 11
limit (vFilers per filer)
adjusting 27
listing IPspaces 66
local user accounts 76
LUNs on vFilers 46

M

managing filer
backups and data recovery 6
Snapshot and SnapMirror 6
volumes, disks, and RAID groups 6
managing vFilers from vFilers 7
mandatory_scan option 126
maximum number of vFilers 4
messages, autosupport for vFilers 37
MIBs
NetApp custom 52
standard 52
migrating (moving) a vFiler
summary of tasks 92
migrating a vFiler 120
effect on clients 120
migrating data, using vFilers for 3
migrating filer data 3
moving a vFiler 89
network checks 96
preparation 93
quotas 95
storage checks 93
summary of tasks 92
moving or removing resources, requirements for 23
moving resources, meaning of 23
multiple security domains 2
multiple server consolidation 2

N

naming storage resources in vFilers 12
naming vFilers, guidelines for 13
NetApp custom MIBs 52
network checks
    for migration, disaster recovery 96
network interfaces
    configuring down 17
    for vFilers 4
    removing an IP alias from 16
network resources
    adding 24
    base IP address 16
    changing 23
    considerations for moving between vFilers 24
    IP alias 16
    planning for when creating vFilers 12
    removing 25
    requirements for moving and removing 23
    steps for moving between vFilers 25
    types of 2
NFS
    path names for exporting 74
    starting the protocol 75
    statistics 53
nfsstat command 53
NIS and DNS servers
    and vFiler 3
NIS servers
    and migration, disaster recovery 99
    and vFiler 3
no-vfiler-ips? variable
    setting 116

O

oplock settings for qtrees on vFilers, changing 43
options
    after vFiler creation 15
    settable on vFilers 40
    specifying path names for 41
options snapmirror command 113
options snapmirror.access command 114
original vFiler
    reactivating after disaster 112
outgoing packets for an IPspace 59

P

packets
    broadcast in an IPspace 59
    incoming for an IPspace 59
partitioning filer resources 3
path names, for NFS exports and CIFS shares 74
performance, monitoring 53
primary storage unit, definition of 12
production (original) vFiler
    reactivating after disaster 112
protocols
    allowing or disallowing 34
    supported for vFilers 34
pseudo-root, definition of 5

Q

qtree command output, vFilers and hosting filer 44
qtrees
    assigning to vFilers 12
    creating on vFiler0 43
    creating on vFilers 43
    oplock settings for 43
    requirement for replication on new vFiler 94
    security styles for 43
    who can create on vFilers 43
quota report command 95
quotas
    displaying status of 84
    effects of destroying vFilers on 31
    guideline for 13
    information in standard MIB 52
    messages from exceeding thresholds 87
    prerequisite for turning on and off 79
    reports on 85
    resizing 82
    seeing where enforced from 95
    tree 85
    turning on and off 80
    types supported for vFilers 78
    user and group 83
    vFiler names in report 85, 86
    where user and group quotas take effect 83

which are copied to new vFiler 95
who specifies 78

R

reactivating original vFiler after disaster 112
rebooting filer, effects on vFilers 42
removing IP addresses from an interface 67
removing IP aliases from an interface 16
replacement vFiler
creating after disaster 116
resizing quotas 82
resources
administering from the hosting filer 22
assigning 17
changing 23
guidelines for assigning 12
network 2
storage 2
restricting filer traffic 3
routed daemon
disabling 11
effects of disabling 50
enabling 11
routing table
for the filer 50
for vFilers in default IPspace 50
rsh
access to vFiler from clients 39
allowing or disallowing on vFilers 34
enabling 39
entering commands for vFilers through 38
rsh.access option 39
rsh.enable option 35, 39
running state of vFilers 15

S

sample vFiler 5
scanners, virus 126
secure routing and IPspace 58
server consolidation 2
Server Manager 76
service providers, using vFilers for data hosting 3
setting up vFilers
introduction to 20, 75
steps for 20
setup command for vFilers
purpose of 20, 75
running to unconfigure IP addresses 118
syntax for 21
SnapMirror
license requirement 90
on hosting filer 51
options snapmirror.access command 114
using to reactivate vFiler 112
snapmirror break command 115
snapmirror command
considerations for 51
where to enter 51
snapmirror quiesce command 114
snapmirror resync command 113
snapmirror update command 114, 118
snapmirror.access option 113
snapmirror.allow file 114
SNMP
on the hosting filer 52
using with vFilers 52
SSPs, and IPspaces 63
standard MIB 52
starting vFilers
meaning of 29
step for 30
state of vFiler
after vFiler creation 15
depends on role after replication 90
displaying 36
effect of filer reboot on 42
stopping vFilers 29
storage area network (SAN) guidelines 14
storage checks
for migration, disaster recovery 93
storage resources
adding 24
administering from the hosting filer 22
assigning qtrees 12
assigning volumes 12
changing 23
considerations for moving between vFilers 24
moving between vFilers 25
naming 12

planning for when creating vFilers 12
primary unit 12
removing 25
requirements for moving and removing 23
types of 2
subnets, and moving vFiler 124
syslogd messages 37
sysstat command 53

T

TCP statistics in standard MIB 52
thresholds, quota 87
timezone 20
traffic
restricting on filer 3
trusted host
and migration, disaster recovery 102
changing after migration 124
changing on disaster-recovery vFiler 108

U

UDP statistics in standard MIB 52
uptime command 53
User Manager 76
UUID for vFiler 36

V

vFiler
activating backup for disaster-recovery 110
add command, syntax for 25
administering resources from hosting filer 22
allow command, syntax for 35
changing resources for 23
command family, path name in 19
command family, purpose of 19
commands, entering from hosting filer 38, 39
commands, entering from vFiler context 38
commands, entering through rsh 40
create command, syntax for 18
creating backup for disaster-recovery 89
creating disaster-recovery 107
default 4
destroy command, syntax for 32
disallow command, syntax for 35
displaying status 36
group in NetApp custom MIB 52
limit per hosting filer 4, 27
LUNs and 46
migrating 120
move command, syntax for 26
moving 121
moving (migrating) 89, 120
moving (summary of tasks) 92
moving to different subnet 124
moving to different Windows domain 124
preparing backup for disaster reovery 91
reactivating original after disaster 112
reactivating using vFiler dr commands 115
reactivating via SnapMirror commands 112
recreating 33
remove command, syntax for 25
replacing on original filer 117
run command, purpose of 38
server consolidation, using for 2
setup 20, 75
start command, syntax for 30
states, effect of filer reboot on 42
states, stopped or running 29
stop command, syntax for 29
vFiler context command 38
vFiler create command 33
vFiler dr activate command
what it does 110
vFiler dr configure command
what it does 109
vFiler dr delete command 111
vFiler migrate command 121
vFiler migrate complete command
what it does 123
vFiler migrate start command
what it does 123
vFiler status command 36, 37, 93
vFiler0
definition of 4
included in vFiler limit 27
virus scanning for 126
vfilertemplate option, meaning of 19
vfilertemplate, defined 19

virus scanning
configuring 126
for vFiler0 126
registering scanners for 126
requirements for 126
VLAN tagging, used with IPspaces 60
volumes
requirement for replication on new vFiler 94
volumes in vFilers
assigning 12
effects of renaming 43
taking offline 43

W

Windows domain
and migration, disaster recovery 102
changing 110
Windows domain controller
changing 110
WINS server
and migration, disaster recovery 102
changing 110
worksheet for migration, disaster recovery
networking 105
storage 104