

Symantec™ VirtualStore Administrator's Guide

Linux

6.0

Symantec™ VirtualStore Administrator's Guide

The software described in this book is furnished under a license agreement and may be used only in accordance with the terms of the agreement.

Product version: 6.0

Document version: 6.0.1

Legal Notice

Copyright © 2012 Symantec Corporation. All rights reserved.

Symantec, the Symantec logo, Veritas, Veritas Storage Foundation, CommandCentral, NetBackup, Enterprise Vault, and LiveUpdate are trademarks or registered trademarks of Symantec corporation or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Symantec Corporation and its licensors, if any.

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. SYMANTEC CORPORATION SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be commercial computer software as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, "Rights in Commercial Computer Software or Commercial Computer Software Documentation", as applicable, and any successor regulations. Any use, modification, reproduction release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement.

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043
<http://www.symantec.com>

Technical Support

Symantec Technical Support maintains support centers globally. Technical Support's primary role is to respond to specific queries about product features and functionality. The Technical Support group also creates content for our online Knowledge Base. The Technical Support group works collaboratively with the other functional areas within Symantec to answer your questions in a timely fashion. For example, the Technical Support group works with Product Engineering and Symantec Security Response to provide alerting services and virus definition updates.

Symantec's support offerings include the following:

- A range of support options that give you the flexibility to select the right amount of service for any size organization
- Telephone and/or Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers software upgrades
- Global support purchased on a regional business hours or 24 hours a day, 7 days a week basis
- Premium service offerings that include Account Management Services

For information about Symantec's support offerings, you can visit our Web site at the following URL:

www.symantec.com/business/support/index.jsp

All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policy.

Contacting Technical Support

Customers with a current support agreement may access Technical Support information at the following URL:

www.symantec.com/business/support/contact_techsupp_static.jsp

Before contacting Technical Support, make sure you have satisfied the system requirements that are listed in your product documentation. Also, you should be at the computer on which the problem occurred, in case it is necessary to replicate the problem.

When you contact Technical Support, please have the following information available:

- Product release level

- Hardware information
- Available memory, disk space, and NIC information
- Operating system
- Version and patch level
- Network topology
- Router, gateway, and IP address information
- Problem description:
 - Error messages and log files
 - Troubleshooting that was performed before contacting Symantec
 - Recent software configuration changes and network changes

Licensing and registration

If your Symantec product requires registration or a license key, access our technical support Web page at the following URL:

www.symantec.com/business/support/

Customer service

Customer service information is available at the following URL:

www.symantec.com/business/support/

Customer Service is available to assist with non-technical questions, such as the following types of issues:

- Questions regarding product licensing or serialization
- Product registration updates, such as address or name changes
- General product information (features, language availability, local dealers)
- Latest information about product updates and upgrades
- Information about upgrade assurance and support contracts
- Information about the Symantec Buying Programs
- Advice about Symantec's technical support options
- Nontechnical presales questions
- Issues that are related to CD-ROMs or manuals

Support agreement resources

If you want to contact Symantec regarding an existing support agreement, please contact the support agreement administration team for your region as follows:

Asia-Pacific and Japan	customercare_apac@symantec.com
Europe, Middle-East, and Africa	semea@symantec.com
North America and Latin America	supportsolutions@symantec.com

Documentation

Product guides are available on the media in PDF format. Make sure that you are using the current version of the documentation. The document version appears on page 2 of each guide. The latest product documentation is available on the Symantec Web site.

<https://sort.symantec.com/documents>

Your feedback on product documentation is important to us. Send suggestions for improvements and reports on errors or omissions. Include the title and document version (located on the second page), and chapter and section titles of the text on which you are reporting. Send feedback to:

doc_feedback@symantec.com

About Symantec Connect

Symantec Connect is the peer-to-peer technical community site for Symantec's enterprise customers. Participants can connect and share information with other product users, including creating forum posts, articles, videos, downloads, blogs and suggesting ideas, as well as interact with Symantec product teams and Technical Support. Content is rated by the community, and members receive reward points for their contributions.

<http://www.symantec.com/connect/storage-management>

Contents

Technical Support	4
Chapter 1 Symantec VirtualStore	9
About Symantec VirtualStore	9
VirtualStore commands	9
Chapter 2 Veritas Replicator	11
About Veritas Replicator	11
What is VFR	11
Features of VFR	12
Chapter 3 Deploying Oracle with Clustered NFS	13
Tasks for deploying Oracle with CNFS	13
About deploying Oracle with CNFS	14
VCS service groups in a CNFS environment	14
Configuring the CNFS server for Oracle	15
Configuring Oracle for Direct NFS	18
Recommended mount options for NFS	19
About oranfstab	20
Verifying Oracle Direct NFS usage	21
Chapter 4 VirtualStore utilities for the Oracle database	25
About VirtualStore utilities for the Oracle database	25
Installing VirtualStore utilities for the Oracle database	26
Using the svsdbsnap command	26
Chapter 5 Administering iSCSI with VirtualStore	29
About iSCSI with VirtualStore	29
Prerequisites	29
svsiscsiadm manual page	29
Administering iSCSI with VirtualStore	30
Configuring the cluster for iSCSI	30
Creating targets	30

	Adding LUNs to targets	30
	Removing LUNs	31
	Removing targets	31
	Unconfiguring the cluster for iSCSI	32
	Create a clone with FileSnap	32
	Add iSCSI-backed VirtualStore storage shares to vCenter and ESX	32
	Online targets	32
	Offline targets	33
	Display LUN status	33
Chapter 6	Administering datastores with VirtualStore	35
	About administering datastores with VirtualStore	35
	About svsdastore utility	35
	Administering NFS datastores	36
	Glossary	39
	Index	45

Symantec VirtualStore

This chapter includes the following topics:

- [About Symantec VirtualStore](#)
- [VirtualStore commands](#)

About Symantec VirtualStore

Symantec VirtualStore (SVS) powered by Veritas Storage Foundation Cluster File System High Availability (SFCFSHA) serves as a highly scalable, highly available NAS solution optimized for deploying and hosting virtual machine. VirtualStore is built on top of Cluster File System (CFS), which provides high availability and linear scalability across the cluster.

In addition, the SFCFSHA concepts and features are documented in *Veritas Storage Foundation Cluster File System High Availability Administrator's Guide* and can be used with SVS.

See the *Veritas Storage Foundation Cluster File System High Availability Administrator's Guide*.

VirtualStore commands

Table 1-1 describes the VirtualStore (SVS) commands.

Command	Description
svsdatastore	Datastore configuration command for VirtualStore. See “ About svsdatastore utility ” on page 35. See the <code>svsdatastore(1M)</code> manual page.

Table 1-1 describes the VirtualStore (SVS) commands. *(continued)*

Command	Description
svsdbsnap	<p>Creates, manages, and clones point-in-time copy of the Oracle database.</p> <p>See “About VirtualStore utilities for the Oracle database” on page 25.</p> <p>See the <code>svsdbsnap(1M)</code> manual page.</p>
svsiscsiadm	<p>Adds, deletes, and monitors iSCSI exports.</p> <p>See “About iSCSI with VirtualStore” on page 29.</p> <p>See the <code>svsiscsiadm(1M)</code> manual page.</p>
svsvmwadm	<p>Symantec VirtualStore (SVS) VMware register configuration command.</p> <p>See the <i>Symantec VirtualStore Installation and Configuration Guide</i>.</p> <p>See the <code>svsvmwadm(1M)</code> manual page.</p>

Veritas Replicator

This chapter includes the following topics:

- [About Veritas Replicator](#)
- [What is VFR](#)
- [Features of VFR](#)

About Veritas Replicator

Veritas Replicator from Symantec provides organizations with a comprehensive solution for heterogeneous data replication. As an option to Veritas Storage Foundation, Veritas Replicator enables cost-effective replication of data over IP networks, giving organizations an extremely flexible, storage hardware independent alternative to traditional array-based replication architectures. Veritas Replicator provides the flexibility of block-based continuous replication with Veritas Volume Replicator (VVR) and file-based periodic replication with Veritas File Replicator (VFR).

What is VFR

Veritas File Replicator (VFR) enables cost-effective periodic replication of data over IP networks, giving organizations an extremely flexible storage independent data availability solution for disaster recovery and off-host processing. With flexibility of scheduling the replication intervals to match the business requirements, Veritas File Replicator tracks all updates to the File System and replicates these updates at the end of the configured time interval. VFR leverages data deduplication provided by Veritas File System (VxFS) to reduce the impact that replication can have on scarce network resources. VFR is included, by default, with Symantec Virtual Store 6.0 on Linux and is available as an option with Veritas Storage Foundation and associated products on Linux.

Features of VFR

Veritas File Replicator (VFR) includes the following features:

- Supports file system level replication of application data.
- Supports reversible data transfer. The target of replication may become the source at runtime, with the former source system becoming a target.
- Provides efficiency of data transfer when transferring shared extents, so that the data is not sent multiple times over the network.
- Supports automatic recovery from the last good successfully replicated point in time image.
- Periodically replicates changes. The interval is configurable by the user.
- Supports deduplication to increase storage efficiency on the target system.

See the *Veritas Storage Foundation and High Availability Solutions Replication Administrator's Guide* for more information.

Deploying Oracle with Clustered NFS

This chapter includes the following topics:

- [Tasks for deploying Oracle with CNFS](#)
- [About deploying Oracle with CNFS](#)
- [Configuring the CNFS server for Oracle](#)
- [Configuring Oracle for Direct NFS](#)
- [Verifying Oracle Direct NFS usage](#)

Tasks for deploying Oracle with CNFS

If you are using Storage Foundation Database (SFDB) tools to set up an Oracle database with CNFS, complete these tasks in the order listed below:

Configure CNFS server for Oracle.

See [“About deploying Oracle with CNFS”](#) on page 14.

See [“Configuring the CNFS server for Oracle”](#) on page 15.

Configure Oracle for Direct NFS.

See [“Configuring Oracle for Direct NFS”](#) on page 18.

See [“Recommended mount options for NFS”](#) on page 19.

See [“About oranfstab”](#) on page 20.

Verify Oracle Direct NFS usage.

See [“Configuring Oracle for Direct NFS”](#)
on page 18.

About deploying Oracle with CNFS

Clustered Network File System (CNFS) is a solution to deliver active/active NFS serving over an underlying cluster file system. Each node in a CNFS cluster runs the complete Cluster Volume Manager-Cluster File System-Veritas Cluster Server (CVM-CFS-VCS) stack, and in addition, the CNFS server parallel application component. The CNFS server converts the NFS request from the client to POSIX file system requests and issues them to the underlying CFS instance. The CFS and the CVM instances coordinate to provide concurrent access to one or more file systems from all the cluster nodes. A cluster-wide volume and file system configuration allows for simplified management. Additionally, an integrated cluster volume manager presents every node in the cluster with the same logical view of shared device configurations.

Veritas Storage Foundation Cluster File System (CFS) from Symantec offers an efficient solution for providing active/active NFS serving at a fraction of the cost of high-end Network Attached Storage (NAS) heads and filers. CFS takes advantage of existing SAN infrastructure and scalability can be achieved both at the client connectivity layer and the backend storage layer. CFS is tuned to handle multiple types of workloads from access to large files to many clients accessing multiple small-sized files.

The Oracle Database 11g Direct NFS client integrates the NFS client functionality directly in the Oracle software. Through this integration, the I/O path between Oracle and the NFS server is optimized providing significantly better performance. In addition, the Direct NFS client simplifies and, in many cases, automates the performance optimization of the NFS client configuration for database workloads.

With CNFS as the database storage, you can use all the advanced features of Veritas Storage Foundation.

VCS service groups in a CNFS environment

In a Veritas Storage Foundation Cluster File System (CFS) cluster with Clustered Network File System (CNFS) configured with a single virtual IP, the following are the Veritas Cluster Server (VCS) service groups:

- **cvm**: This service group controls the Cluster Volume Manager (CVM) and the CFS shared resources. This group is automatically created during the configuration phase of CFS installation. This service group manages CVM and the basic CFS functionality that is provided through `vxfsckd`.

- `cfsnfssg`: This service group contains the CFS mount resources for the NFS share as well as the shared CFS mount resources that are needed for lock management. This service group consists of the NFS resource and the share resource apart from the `CVMVoldg` and the `CFSMount` resources.
- `vip1`: This service group contains the virtual IP and the NIC resources needed by NFS clients to connect. The virtual IP service group fails over from one node to another during system failover. Typically, more than one virtual IP is assigned per CNFS cluster.

`cvm` and `cfsnfssg` are configured as parallel service groups and are online on all the nodes. The `vip1` service group is configured as a failover service group.

See the *Veritas Cluster Server Administrator's Guide* for more information on service groups and service groups dependencies.

Configuring the CNFS server for Oracle

You can use the following procedure to configure a Clustered Network File System (CNFS) server for Oracle databases. The following example procedure assumes a two node Veritas Storage Foundation Cluster File System (CFS) 6.0 cluster with host names `cnfs-1` and `cnfs-2`.

To configure a CNFS server for Oracle database

- 1 Configure a shared disk group for Oracle.

```
[cnfs-1]# vxdg -s init oradg disk1 disk2 disk3 disk4
```

- 2 Create volumes for data files, archive logs, and CNFS locks.

```
[cnfs-1]# vxassist -g oradg make oranfsdata 100g \  
layout=stripe ncolumn=4 st_width=1m disk1 disk2 disk3 disk4  
[cnfs-1]# vxassist -g oradg make oranfsarch 10g  
[cnfs-1]# vxassist -g oradg make cnfs_locks 2g
```

Note: Symantec recommends a stripe volume with a stripe width of 1 MB for Oracle data files.

3 Create file systems for CNFS locks, data files, and archive logs.

```
[cnfs-1]# mkfs -t vxfs -o version=9,bsize=8192 \  
/dev/vx/rdisk/oradg/oranfsdata  
[cnfs-1]# mkfs -t vxfs /dev/vx/rdisk/oradg/oranfsarch  
[cnfs-1]# mkfs -t vxfs /dev/vx/rdisk/oradg/cnfs_locks
```

Note: Symantec recommends a file system block size of 8 KB for Oracle data files.

Note: To use the FileSnap feature, the file system must be disk layout Version 8 or later.

4 Configure Clustered NFS.

```
[cnfs-1]# cfsshare config -p nfs oranfsdg /cnfs_locks
```

5 Configure /oranfadata and /oranfsarch for NFS sharing.

```
[cnfs-1]# cfsshare add -p nfs -N "rw,no_wdelay,no_root_squash" \  
oradg oranfsdata /oranfsdata all=  
[cnfs-1]# cfsshare add -p nfs -N "rw,no_wdelay,no_root_squash" \  
oradg oranfsarch /oranfsarch all=
```

6 Add virtual IP (VIP).

```
[cnfs-1]# cfsshare addvip eth2 virtual_IP subnet_mask
```

Note: Symantec recommends adding one VIP for each node in the CNFS cluster to load balance the I/O among all the nodes in the cluster.

7 Display the CNFS configuration details.

```
[cnfs-1]# cfsshare display
```

SHARE	RESOURCE	MOUNTPOINT	SHARE OPTIONS
share1		/oranfsarch	rw,wdelay,no_root_squash
share2		/oranfsdata	rw,wdelay,no_root_squash

8 Display the VCS resource details.

```
[cnfs-1]# hastatus
```

group	resource	system	message
		cnfs-2	RUNNING
		cnfs-1	RUNNING
cfsnfssg		cnfs-1	ONLINE
cfsnfssg		cnfs-2	ONLINE
cfsnfssg_dummy		cnfs-1	OFFLINE
cfsnfssg_dummy		cnfs-2	OFFLINE
cvm		cnfs-1	ONLINE
cvm		cnfs-2	ONLINE
vip1		cnfs-1	OFFLINE
vip1		cnfs-2	ONLINE
vip2		cnfs-1	ONLINE
vip2		cnfs-2	OFFLINE
	app	cnfs-1	ONLINE
	app	cnfs-2	ONLINE
	cfsmount1	cnfs-1	ONLINE
	cfsmount1	cnfs-2	ONLINE
	cfsmount2	cnfs-1	ONLINE
	cfsmount2	cnfs-2	ONLINE
	cfsnfs_locks	cnfs-1	ONLINE
	cfsnfs_locks	cnfs-2	ONLINE
	cvmvoldg1	cnfs-1	ONLINE
	cvmvoldg1	cnfs-2	ONLINE
	nfs	cnfs-2	ONLINE
	nfs	cnfs-2	ONLINE
	share1	cnfs-1	ONLINE
	share1	cnfs-2	ONLINE
	share2	cnfs-1	ONLINE
	share2	cnfs-2	ONLINE
	vxfscd	cnfs-1	ONLINE
	vxfscd	cnfs-2	ONLINE

cvm_clus	cnfs-1	ONLINE
cvm_clus	cnfs-2	ONLINE
cvm_vxconfigd	cnfs-1	ONLINE
cvm_vxconfigd	cnfs-2	ONLINE
vip1	cnfs-1	OFFLINE
vip1	cnfs-2	ONLINE
nic1	cnfs-1	ONLINE
nic1	cnfs-2	ONLINE
vip2	cnfs-1	ONLINE
vip2	cnfs-2	OFFLINE
nic2	cnfs-1	ONLINE
nic2	cnfs-2	ONLINE

9 Verify that the NFS service is configured on all the nodes.

```
[cnfs-1]# chkconfig --list nfs  
nfs 0:off 1:off 2:on 3:on 4:on 5:on 6:off
```

10 Verify that the NFS service is running on all the nodes.

```
[cnfs-1]# service nfs status  
rpc.mountd (pid 4530) is running...  
nfsd (pid 4527 4526 4525 4524 ..... ) is running...  
rpc.rquotad (pid 4474) is running...
```

Configuring Oracle for Direct NFS

You can configure Oracle as a Direct Network File System (NFS) client by performing the steps in the following example procedure. In this example procedure, Oracle Database version 11.2.0.2 (single instance) is installed on a Linux box.

To configure Oracle for Direct NFS

1 Mount the NFS file systems.

See [“Recommended mount options for NFS”](#) on page 19.

```
[orahost1]# mount -t nfs -o \
rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,noac,forcedirectio,vers=3
virtual_IP:/oranfsdata /oranfsdata
```

Note: Use the virtual IP (VIP) in the `mount` command.

2 Enable the Oracle Disk Manager (ODM) library that supports the Direct NFS client.

Note: You must replace the standard ODM library with one that supports the Direct NFS client to enable the Direct NFS client.

```
[orahost1]# cd $ORACLE_HOME/lib
[orahost1]# mv libodm11.so libodm11.so_bak
[orahost1]# ln -s libnfsodm11.so libodm11.so
```

3 Set up `oranfstab`.

See [“About oranfstab”](#) on page 20.

Recommended mount options for NFS

[Table 3-1](#) lists the mount options for Network File System (NFS) on Solaris, HP-UX, AIX, and Linux operating systems. The recommendations are for Oracle data files.

Table 3-1 Mount options for NFS

Operating System	Mount Options for Oracle Datafiles
Solaris	<code>rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,proto=tcp,noac,forcedirectio,vers=3,suid</code>
AIX (5L)	<code>cio,rw,bg,hard,nointr,rsize=1048576,wsiz=1048576,proto=tcp,noac,vers=3,timeo=600</code>

Table 3-1 Mount options for NFS (*continued*)

Operating System	Mount Options for Oracle Datafiles
HP-UX 11i v3	<code>rw,bg,vers=3,proto=tcp,noac,forcedirectio,hard,nointr,timeo=600,rsize=1048576,wsiz=1048576,suid</code>
Linux x86	<code>rw,bg,hard,nointr,rsiz=1048576,wsiz=1048576,tcp,actimeo=0,vers=3,timeo=600</code>
Linux x86-64	<code>rw,bg,hard,nointr,rsiz=1048576,wsiz=1048576,tcp,actimeo=0,vers=3,timeo=600</code>

About oranfstab

By default, Direct NFS attempts to serve mount entries that are found in `/etc/fstab`. You can use `oranfstab` to specify additional Oracle-specific options to Direct NFS. For example, you can use `oranfstab` to specify additional paths for a mount point. Additionally, a new Oracle-specific file `oranfstab` can be added to either `/etc` or to `$ORACLE_HOME/dbs`. When `oranfstab` is placed in `$ORACLE_HOME/dbs`, its entries are specific to a single database. However, when `oranfstab` is placed in `/etc`, it is global to all Oracle databases, and hence can contain mount points for all Oracle databases. Direct NFS determines mount point settings to NFS storage devices based on the configurations in `/etc/mtab`.

Direct NFS looks for the mount point entries in the following order:

- `$ORACLE_HOME/dbs/oranfstab`
- `/etc/oranfstab`
- `/etc/mtab`

Direct NFS uses the first matched entry as the mount point. In all cases, Oracle requires that mount points be mounted by the kernel NFS system even when being served through Direct NFS. Oracle verifies kernel NFS mounts by cross-checking entries in `oranfstab` with operating system NFS mount points. If a mismatch exists, then Direct NFS logs an informational message, and does not serve the NFS server.

Here is an example of `oranfstab`:

```
[orahost1]# cat $ORACLE_HOME/dbs/oranfstab
server: cnfs-1.engba.symantec.com
```

```
path: virtual_IP
export:/oranfsdata mount:/oranfsdata
export:/oranfsarch mount:/oranfsarch
```

Note: The IP address used in path is the virtual IP (VIP) address.

Verifying Oracle Direct NFS usage

Oracle records the use of Direct NFS in alert.log and also in internal catalog v\$dtnfs tables. [Table 3-2](#) lists the v\$tables available to find the state and the health of Direct NFS from Oracle side.

Table 3-2 v\$tables for Direct NFS information

Table Name	Description
v\$dtnfs_servers	Shows a table of servers accessed using Direct NFS
v\$dtnfs_channels	Shows a table of open network paths (or channels) to servers for which Direct NFS is providing files.
v\$dtnfs_files	Shows a table of files now open with Direct NFS
v\$dtnfs_stats	Shows a table of performance statistics for Direct NFS

To verify Oracle Direct NFS usage

1 Check alert.log for DNFS messages.

The following is an example of lines in alert.log from Oracle Database version 11.2.0.2 instance running with ODM:

```
Oracle Direct NFS ODM Library Version 3.0

ALTER DATABASE MOUNT
Direct NFS: channel id [0] path [10.182.110.126] to
filer [cnfs-1.engba.symantec.com] via local [] is UP
Direct NFS: channel id [1] path [10.182.110.126] to
filer [cnfs-1.engba.symantec.com] via local [] is UP
```

2 Check the DNFS server information from v\$dnfs_servers.

```
SQL> select * from v$dnfs_servers;
```

ID	SVRNAME	DIRNAME	MNTPORT	NFSPO	RTMAX	WTMAX
1	cnfs-1.engba.symantec.com	/oranfsdata1	33553	2049	1048576	1048576
2	cnfs-1.engba.symantec.com	/oranfsdata1	33553	2049	1048576	1048576

3 Check the DNFS channel information from v\$dnfs_channels.

```
SQL> select CH_ID, SVR_ID, SENDS, RECVS, PINGS from v$dnfs_channels;
```

CH_ID	SVR_ID	SENDS	RECVS	PINGS
0	1	0	0	0
0	1	65	130	0
1	1	44	88	0
1	1	47	94	0

4 Check the DNFS files information from v\$dnfs_files.

```
SQL> select * from v$dnfs_files;
```

FILENAME	FILESIZE	PNUM	SVR_ID
/oranfsdata1/rw_clone/control01.ctl	16072704	15	1
/oranfsdata1/rw_clone/control02.ctl	16072704	15	1
/oranfsdata1/rw_clone/control03.ctl	16072704	15	1
/oranfsdata1/rw_clone/bench.dbf	838877184	10	1
/oranfsdata1/rw_clone/sysaux.dbf	838877184	10	1
/oranfsdata1/rw_clone/undo1.dbf	838877184	10	1
/oranfsdata1/rw_clone/item_1000	1996505088	10	1

5 Check the DNFS statistics information from v\$dnfs_stats.

```
SQL> select PNUM, NFS_READ, NFS_WRITE, NFS_COMMIT, NFS_MOUNT from v$dnfs_stats;
```

PNUM	NFS_READ	NFS_WRITE	NFS_COMMIT	NFS_MOUNT
10	135	201	0	0
11	0	201	0	0
12	0	191	0	0
13	0	198	0	0
14	86	813	0	0
15	426	1293	1	1

VirtualStore utilities for the Oracle database

This chapter includes the following topics:

- [About VirtualStore utilities for the Oracle database](#)
- [Installing VirtualStore utilities for the Oracle database](#)
- [Using the `svsdbsnap` command](#)

About VirtualStore utilities for the Oracle database

Symantec provides a tool to create a point-in-time copy of the database called DBSNAP. DBSNAP is a point-in-time copy of all the database files taken using the `vxfilesnap` command. FileSnap is a space-optimized copy of a file in the same name space stored in the same file system. DBAs can use this facility to create a space optimized point-in-time copy of the database. This database copy can be used to create a clone database on any host which has access to this file system. DBSNAP is a true database backup copy of the database which can be used for database point-in-time recovery.

The `svsdbsnap` command provides ability to create, manage, and clone point-in-time copy of Oracle database. These point-in-time copies are called DBSNAP images. This command runs as Oracle DBA from the ORACLE host. This command queries database, retrieves information about data files, control files and redo log by querying the database. Therefore the database needs to be online for creating DBSNAP. This command can also be used to restore the primary database from the DBSNAP image.

Installing VirtualStore utilities for the Oracle database

To install VirtualStore utilities for the Oracle database

- 1 Login as root.
- 2 Set umask 002.
- 3 Create the following directory:

```
# mkdir /opt/VRTSdbsnap
```
- 4 Change to the /opt/VRTSdbsnap directory:

```
# cd /opt/VRTSdbsnap
```
- 5 Download the DBSNAP utilities from go.symantec.com/virtualstoreutilities.
- 6 Unzip the DBSNAP file using the `gzip` command. This provides a file by name `dbsnap-MM-DD-YY.tar`.
- 7 Extract the TAR image:

```
# tar xvf dbsnap-MM-DD-YY.tar
```
- 8 Ensure that you add the `/opt/VRTSdbsnap/bin` to your `PATH` for the Oracle login user.
- 9 Ensure that you add the `/opt/VRTSdbsnap/man` to your `MANPATH` for the Oracle login user.

Using the `svsdbsnap` command

This section provides many examples how to use the `svsdbsnap` command.

See the `svsdbsnap` (1M) manual page for more detailed information.

To create FileSnap for a database (auto generated name)

- ◆ Create FileSnap for a database:

```
$ svsdbsnap -o create -P $ORACLE_HOME/dbs/initORCL.ora
DBSNAP by name ORCL_SNAP_2011-04-01:13:29:30 created successfully
Program Completed
```

To create a named FileSnap for a database

- ◆ Create a named FileSnap for a database:

```
$ svsdbsnap -o create -P $ORACLE_HOME/dbs/initORCL.ora -n new_snap
DBSNAP by name new_snap created successfully
Program Completed
```

To display the FileSnap copies of a database

- ◆ Display the FileSnap copies of a database:

```
$ svsdbsnap -o display -P $ORACLE_HOME/dbs/initORCL.ora
```

```
-----
NAME                                STATUS    ARCH_DEST
-----
new_snap                            VALID     /oranfsdata2/primary/ARCH
ORCL_SNAP_2011-04-01:13:29:30      VALID     /oranfsdata2/primary/ARCH
-----
```

To remove a FileSnap copy of a database

- ◆ Remove a FileSnap copy of a database:

```
$ svsdbsnap -o remove -P $ORACLE_HOME/dbs/initORCL.ora -n \
ORCL_SNAP_2011-04-01:13:29:30
```

To create a clone database from the FileSnap copy by new_snap

- ◆ Create a clone database from the FileSnap copy by the name new_snap:

```
$ svsdbsnap -o create -P $ORACLE_HOME/dbs/initORCL.ora -n \
new_snap -c testdb
Creating Clone database with ORACLE_SID = testdb
Using new_snap
Clone create control file created.
Creating Clone pfile /new11gr2/home/dbs/inittestdb.ora
Clone database is mounted.
Clone Database Opened Successfully
Database CLONE using DBSNAP new_snap created successfully
Program Completed
```

To restore the primary database from the FileSnap copy

- ◆ Restore the primary database from the FileSnap copy:

```
$ svdsbsnap -o restore -D -P $ORACLE_HOME/dbs/initORCL.ora -n new_snap  
RESTORING DATABASE  
Restoring Data files from /oranfsdata1/primary/.DBSNAP/new_snap  
Program Completed
```

Administering iSCSI with VirtualStore

This chapter includes the following topics:

- [About iSCSI with VirtualStore](#)
- [Prerequisites](#)
- [svsiscsiadm manual page](#)
- [Administering iSCSI with VirtualStore](#)

About iSCSI with VirtualStore

The iSCSI with VirtualStore feature provides a mechanism to simplify the administration of exporting iSCSI LUNs backed by files residing on the VirtualStore file system.

Prerequisites

- Ensure that you have disk layout Version 7 or later.

svsiscsiadm manual page

The `svsiscsiadm` command adds, deletes, and monitors iSCSI exports. This utilizes the iSCSI target driver implementation that is shipped with the operating system.

See the `svsiscsiadm(1M)` manual page.

Administering iSCSI with VirtualStore

This section describes how to administer iSCSI.

Configuring the cluster for iSCSI

To configure the cluster for iSCSI

- ◆ Configure the cluster for iSCSI:

```
# svsiscsiadm config iqn_prefix
```

For example:

```
# svsiscsiadm config iqn.2007:07:com.symantec.storage
```

Creating targets

To creating targets

- 1 Create targets:

```
# svsiscsiadm create target -a ACL CFMountResource
```

For example:

```
# svsiscsiadm create target -a 10.172.139.31 1.2.3.4 cfmount2
```

- 2 Display the current iSCSI exports:

```
# svsiscsiadm list
```

Adding LUNs to targets

To create a LUN and a target with default options

- 1 Create a LUN and a target with default options:

```
# svsiscsiadm create lun Path_Of_LUNBackingFile Size_Of_LUN
```

For example:

```
# svsiscsiadm create lun /mnt0/target1/lun1 1G
```

- 2 Display the current iSCSI exports:

```
# svsiscsiadm list
```

To create a second target against the same target

- 1 Create a second target against the same target:

```
# svsiscsiadm create lun [-t TargetID] LUNBackingFileSize
```

For example:

```
# svsiscsiadm create lun -t 1 /mnt0/target1/lun2 1G
```

- 2 Display the current iSCSI exports:

```
# svsiscsiadm list
```

Removing LUNs

To remove LUNs

- ◆ Remove LUNs:

```
# svsiscsiadm remove lun TargetID LUN_ID
```

For example:

```
# svsiscsiadm remove lun 2 1
```

Removing targets

To remove targets

- 1 Remove targets:

```
# svsiscsiadm remove target TargetID
```

For example:

```
# svsiscsiadm remove target 1
```

Repeat this step if there is more than 1 target.

- 2 Display the current iSCSI exports:

```
# svsiscsiadm list
```

Unconfiguring the cluster for iSCSI

To unconfigure the cluster for iSCSI

- ◆ Unconfigure the cluster for iSCSI:

```
# svsiscsiadm unconfig
```

Create a clone with FileSnap

To create a clone with FileSnap

- ◆ Create a clone with FileSnap:

```
# svsiscsiadm create lun -s PATH_Of_LUNBackingFile \  
PATH_Of_LUNBackingFileSNAP
```

For example:

```
# svsiscsiadm create lun -s /mnt1/target1/lun1 /mnt1/target1/lun1_snap
```

Add iSCSI-backed VirtualStore storage shares to vCenter and ESX

To add iSCSI-backed VirtualStore storage shares to vCenter and ESX

- ◆ Refer to *VMware iSCSI SAN Configuration Guide* http://www.vmware.com/pdf/vsphere4/r40/vsp_40_iscsi_san_cfg.pdf for more information on "Configuring iSCSI Initiators and Storage" and "Add iSCSI Storage".

Online targets

To online a target

- ◆ Online a target:

```
# svsiscsiadm online TargetID
```

For example:

```
# svsiscsiadm online 1
```


Offline targets

To offline a target

- ◆ Offline a target:

```
# svsiscsiadm offline TargetID
```

For example:

```
# svsiscsiadm offline 1
```

To force offline a target

- ◆ Force offline a target:

```
# svsiscsiadm offline -f TargetID
```

For example:

```
# svsiscsiadm offline -f 1
```

Display LUN status

To display LUN status

- ◆ Display LUN status:

```
# svsiscsiadm list
```

Sample output:

```
Target 1: iqn.2011-07.com.symantec:svst1 /vxfsshare
      1: /vxfsshare/lun23          23G *
      2: /vxfsshare/lun22          22G *
```

To get cluster wide status use

- ◆ To get cluster wide status use:

```
# svsisciadm list -s
```

Sample output:

```
Target 1: iqn.2011-07.com.symantec:svst1 /vxfsshare
      1: /vxfsshare/lun23          23G *
          fssolspr13                <Online>
          fssolspr14                <Online>
      2: /vxfsshare/lun22          22G *
          fssolspr13                <Online>
          fssolspr14                <Online>
```

Administering datastores with VirtualStore

This chapter includes the following topics:

- [About administering datastores with VirtualStore](#)
- [About svsdastore utility](#)
- [Administering NFS datastores](#)

About administering datastores with VirtualStore

The `svsdastore` utility shipped with VirtualStore provides end-to-end provisioning of NFS datastores. It takes a bunch of disks and Virtual IP information and sets up a datastore that can be easily added to VMware ESX via NFS. The utility also allows you to do basic operations such as grow, shrink, and delete the configured NFS datastores. The utility hides all the internal details of the VirtualStore stack and provides an simplified abstraction for the user.

About svsdastore utility

Provides end-to-end provisioning of NFS datastores. The `svsdastore` command allows you to do basic operations such as grow, shrink, delete on the NFS datastore.

The `svsdastore` command does the following:

- It takes a mount point and resizes (grow and shrink) the NFS datastore associated with it.
- It takes a bunch of disks and creates an NFS datastore that can be added to ESX.

- It takes a mount point and deletes the NFS datastore associated with it.
- It takes an IP address, netmask, device, and set it up as a virtual IP (VIP).
- It displays the complete CNFS configuration (mount points and virtual IP addresses).

The Cluster manager software must be up and the `cfsscuster config` command should have been run before you run the `cfsshare` command. Only a privileged user can run this command.

See the `svsdatastore(1M)`, `cfsscuster(1M)`, and `cfsshare(1M)` manual pages.

Administering NFS datastores

To create a new datastore with disk `disk_0` and `disk_1`

- ◆ Create a new datastore with disk `disk_0` and `disk_1`:

```
# svsdatastore create disk_0 disk_1
```

To create a new datastore with disk `disk_0` and `disk_1` with a size of 10G

- ◆ Create a new datastore with disk `disk_0` and `disk_1` with a size of 10G:

```
# svsdatastore create -s 10g disk_0 disk_1
```

To create a new datastore with disk `disk_0` and `disk_1` with a size of 10G on the mount point

- ◆ Create a new datastore with disk `disk_0` and `disk_1` with a size of 10G on the mount point:

```
# svsdatastore create -s 10g -m /mntpt disk_0 disk_1
```

To delete the datastore associated with mount point

- ◆ Delete the datastore associated with mount point:

```
# svsdatastore delete -m /mntpt
```

To resize the datastore associated with mount point to the size of 15G

- ◆ Resize the datastore associated with mount point to the size of 15G:

```
# svsdatastore resize -m /mntpt -s 15g
```

To add a virtual IP "10.192.111.222" with the netmask "255.255.240.0" on network interface "NIC"

- ◆ Add a virtual IP "10.192.111.222" with the netmask "255.255.240.0" on network interface "NIC":

```
# svodatastore addvip -i 10.192.111.222 -n 255.255.240.0 -e bge0
```

To remove Virtual IP "10.192.111.222" from the configuration

- ◆ Remove Virtual IP "10.192.111.222" from the configuration:

```
# svodatastore deletevip -i 10.192.111.222
```

To display the datastore configuration

- ◆ Display the datastore configuration:

```
# svodatastore display
CNFS metadata filesystem : /locks

# MOUNTPOINT          SIZE    SHARE  OPTIONS
/defragvol            250G   rw,no_root_squash

# Virtual IP          STATE
10.209.87.147        ONLINE on swlx65
```


Glossary

ACL (access control list)	The information that identifies specific users or groups and their access privileges for a particular file or directory.
agent	A process that manages predefined Veritas Cluster Server (VCS) resource types. Agents bring resources online, take resources offline, and monitor resources to report any state changes to VCS. When an agent is started, it obtains configuration information from VCS and periodically monitors the resources and updates VCS with the resource status.
allocation unit	A group of consecutive blocks on a file system that contain resource summaries, free resource maps, and data blocks. Allocation units also contain copies of the super-block.
API	Application Programming Interface.
asynchronous writes	A delayed write in which the data is written to a page in the system's page cache, but is not written to disk before the write returns to the caller. This improves performance, but carries the risk of data loss if the system crashes before the data is flushed to disk.
atomic operation	An operation that either succeeds completely or fails and leaves everything as it was before the operation was started. If the operation succeeds, all aspects of the operation take effect at once and the intermediate states of change are invisible. If any aspect of the operation fails, then the operation aborts without leaving partial changes.
BLI (block-level incremental) backup	A Veritas backup capability that does not store and retrieve entire files. Instead, only the data blocks that have changed since the previous backup are backed up.
boot disk	A disk that is used for the purpose of booting a system.
boot disk group	A private disk group that contains the disks from which the system may be booted.
buffered I/O	A mode of I/O operation (where I/O is any operation, program, or device that transfers data to or from a computer) that first transfers data into the Operating System buffer cache.
bootdg	A reserved disk group name that is an alias for the name of the boot disk group.
cluster mounted file system	A shared file system that enables multiple hosts to mount and perform file operations on the same file. A cluster mount requires a shared storage device that can be accessed by other cluster mounts of the same file system. Writes to the

shared device can be done concurrently from any host on which the cluster file system is mounted. To be a cluster mount, a file system must be mounted using the `mount -o cluster` option.

Cluster Services	The group atomic broadcast (GAB) module in the SFCFS stack provides cluster membership services to the file system. LLT provides kernel-to-kernel communications and monitors network communications.
contiguous file	A file in which data blocks are physically adjacent on the underlying media.
CVM (cluster volume manager)	The cluster functionality of Veritas Volume Manager.
CVM Master	The cluster volume manager has a master node that records changes to the volume configuration.
data block	A block that contains the actual data belonging to files and directories.
data synchronous writes	A form of synchronous I/O that writes the file data to disk before the write returns, but only marks the inode for later update. If the file size changes, the inode will be written before the write returns. In this mode, the file data is guaranteed to be on the disk before the write returns, but the inode modification times may be lost if the system crashes.
defragmentation	The process of reorganizing data on disk by making file data blocks physically adjacent to reduce access times.
direct extent	An extent that is referenced directly by an inode.
direct I/O	An unbuffered form of I/O that bypasses the kernel's buffering of data. With direct I/O, the file system transfers data directly between the disk and the user-supplied buffer.
discovered direct I/O	Discovered Direct I/O behavior is similar to direct I/O and has the same alignment constraints, except writes that allocate storage or extend the file size do not require writing the inode changes before returning to the application.
encapsulation	A process that converts existing partitions on a specified disk to volumes. If any partitions contain file systems, <code>/etc/fstab</code> entries are modified so that the file systems are mounted on volumes instead. Encapsulation is not applicable on some systems.
extent	A group of contiguous file system data blocks treated as a single unit. An extent is defined by the address of the starting block and a length.
extent attribute	A policy that determines how a file allocates extents.
external quotas file	A quotas file (named <code>quotas</code>) must exist in the root directory of a file system for quota-related commands to work. Support for group quotas requires a group quotas file is named <code>quotas.grp</code> .

file system block	The fundamental minimum size of allocation in a file system. This is equivalent to the fragment size on some UNIX file systems.
fileset	A collection of files within a file system.
fixed extent size	An extent attribute used to override the default allocation policy of the file system and set all allocations for a file to a specific fixed size.
fragmentation	The on-going process on an active file system in which the file system is spread further and further along the disk, leaving unused gaps or fragments between areas that are in use. This leads to degraded performance because the file system has fewer options when assigning a file to an extent.
GB (gigabyte)	2^{30} bytes or 1024 megabytes.
hard limit	The hard limit is an absolute limit on system resources for individual users for file and data block usage on a file system.
heartbeat	Heartbeat messages are sent over the private link to obtain information on cluster membership changes. If a node does not send a heartbeat for 16 seconds, it is removed from the membership. The command <code>lltconfig</code> is used for information on the various heartbeat parameters. The low latency transport (LLT) module provides communication services across the cluster.
indirect address extent	An extent that contains references to other extents, as opposed to file data itself. A single indirect address extent references indirect data extents. A double indirect address extent references single indirect address extents.
indirect data extent	An extent that contains file data and is referenced via an indirect address extent.
inode	A unique identifier for each file within a file system that contains the data and metadata associated with that file.
inode allocation unit	A group of consecutive blocks containing inode allocation information for a given fileset. This information is in the form of a resource summary and a free inode map.
intent logging	A method of recording pending changes to the file system structure. These changes are recorded in a circular intent log file.
internal quotas file	VxFS maintains an internal quotas file for its internal usage. The internal quotas file maintains counts of blocks and indices used by each user and group.
KB (kilobyte)	2^{10} bytes or 1024 bytes.
large file	A file larger than one terabyte. VxFS supports files up to 8 exabytes in size.
large file system	A file system larger than one terabytes. VxFS supports file systems up to 8 exabytes in size.
latency	For file systems, this typically refers to the amount of time it takes a given file system operation to return to the user.

local mounted file system	A file system mounted on a single host. The single host mediates all file system writes to storage from other clients. To be a local mount, a file system cannot be mounted using the <code>mount -o cluster</code> option.
metadata	Structural data describing the attributes of files on a disk.
MB (megabyte)	2 ²⁰ bytes or 1024 kilobytes.
mirror	A duplicate copy of a volume and the data therein (in the form of an ordered collection of subdisks). Each mirror is one copy of the volume with which the mirror is associated.
multi-volume file system	A single file system that has been created over multiple volumes, with each volume having its own properties.
MVS (multivolume support)	
node	One of the hosts in a cluster.
node abort	A situation where a node leaves a cluster (on an emergency basis) without attempting to stop ongoing operations.
node join	The process through which a node joins a cluster and gains access to shared disks.
OLT (object location table)	The information needed to locate important file system structural elements. The OLT is written to a fixed location on the underlying media (or disk).
page file	A fixed-size block of virtual address space that can be mapped onto any of the physical addresses available on a system.
preallocation	A method of allowing an application to guarantee that a specified amount of space is available for a file, even if the file system is otherwise out of space.
primary fileset	The files that are visible and accessible to the user.
quotas	Quota limits on system resources for individual users for file and data block usage on a file system.
quotas file	The <code>quotas</code> commands read and write the external <code>quotas</code> file to get or change usage limits. When quotas are turned on, the quota limits are copied from the external <code>quotas</code> file to the internal <code>quotas</code> file.
reservation	An extent attribute used to preallocate space for a file.
SFCFS (Storage Foundation Cluster File System)	
SFCFS Primary	There is a primary node for each file system in the cluster responsible for updating metadata in the file system.

shared disk group	A disk group in which the disks are shared by multiple hosts (also referred to as a cluster-shareable disk group).
shared volume	A volume that belongs to a shared disk group and is open on more than one node at the same time.
snapshot file system	An exact copy of a mounted file system at a specific point in time. Used to do online backups.
snapped file system	A file system whose exact image has been used to create a snapshot file system.
soft limit	The soft limit is lower than a hard limit. The soft limit can be exceeded for a limited time. There are separate time limits for files and blocks.
Storage Checkpoint	A facility that provides a consistent and stable view of a file system or database image and keeps track of modified data blocks since the last Storage Checkpoint.
structural fileset	The files that define the structure of the file system. These files are not visible or accessible to the user.
super-block	A block containing critical information about the file system such as the file system type, layout, and size. The VxFS super-block is always located 8192 bytes from the beginning of the file system and is 8192 bytes long.
synchronous writes	A form of synchronous I/O that writes the file data to disk, updates the inode times, and writes the updated inode to disk. When the write returns to the caller, both the data and the inode have been written to disk.
TB (terabyte)	2 ⁴⁰ bytes or 1024 gigabytes.
transaction	Updates to the file system structure that are grouped together to ensure they are all completed.
throughput	For file systems, this typically refers to the number of I/O operations in a given unit of time.
unbuffered I/O	I/O that bypasses the kernel cache to increase I/O performance. This is similar to direct I/O, except when a file is extended; for direct I/O, the inode is written to disk synchronously, for unbuffered I/O, the inode update is delayed.
VCS (Veritas Cluster Server)	
volume	A virtual disk which represents an addressable range of disk blocks used by applications such as file systems or databases.
volume set	A container for multiple different volumes. Each volume can have its own geometry.
vxfs	The Veritas file system type. Used as a parameter in some commands.
VxFS	The Veritas File System.

VxVM

The Veritas Volume Manager.

Index

A

About

Symantec VirtualStore 9

about deploying Oracle with CNFS 14

C

configuring CNFS server for Oracle 15

configuring Oracle for DNFS 18

D

deploying Oracle

 CNFS environment 13–15, 18–21

deploying Oracle with CNFS 13

 configuring CNFS server 15

 configuring Oracle for DNFS 18–20

 mount options for NFS 19

 oranfstab 20

 VCS service groups 14

 verifying DNFS usage 21

S

Symantec VirtualStore

 about 9