Veritas Access 7.2.1.1 NetBackup Solutions Guide

Linux

7.2.1.1

July 2017

VERITAS™

Veritas Access NetBackup Solutions Guide

Last updated: 2017-07-10

Document version: 7.2.1.1 Rev 1

Legal Notice

Veritas Technologies LLC
500 E Middlefield Road
Mountain View, CA 94043

http://www.veritas.com

Technical Support

Technical Support maintains support centers globally. All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policies. For information about our support offerings and how to contact Technical Support, visit our website:

https://www.veritas.com/support

You can manage your Veritas account information at the following URL:

https://my.veritas.com

If you have questions regarding an existing support agreement, please email the support agreement administration team for your region as follows:

Worldwide (except Japan)	CustomerCare@veritas.com
Japan	CustomerCare_Japan@veritas.com

Documentation

Make sure that you have the current version of the documentation. Each document displays the date of the last update on page 2. The document version appears on page 2 of each guide. The latest documentation is available on the Veritas website:

https://sort.veritas.com/documents

Documentation feedback

Your feedback is important to us. Suggest improvements or report errors or omissions to the documentation. Include the document title, document version, chapter title, and section title of the text on which you are reporting. Send feedback to:

doc.feedback@veritas.com

You can also see documentation information or ask a question on the Veritas community site:

http://www.veritas.com/community/

Veritas Services and Operations Readiness Tools (SORT)

Veritas Services and Operations Readiness Tools (SORT) is a website that provides information and tools to automate and simplify certain time-consuming administrative tasks. Depending on the product, SORT helps you prepare for installations and upgrades, identify risks in your datacenters, and improve operational efficiency. To see what services and tools SORT provides for your product, see the data sheet:

https://sort.veritas.com/data/support/SORT_Data_Sheet.pdf

Contents

Chapter 1

Veritas Access integration with NetBackup

This chapter includes the following topics:

About Veritas Access

Veritas Access is a software-defined scale-out network-attached storage (NAS) solution for unstructured data that works on commodity hardware. Veritas Access provides resiliency, multi-protocol access, and data movement to and from the public and private cloud based on policies. You can reduce your storage costs by using low-cost disks and by storing infrequently accessed data in the cloud.

About Veritas Access as a NetBackup client

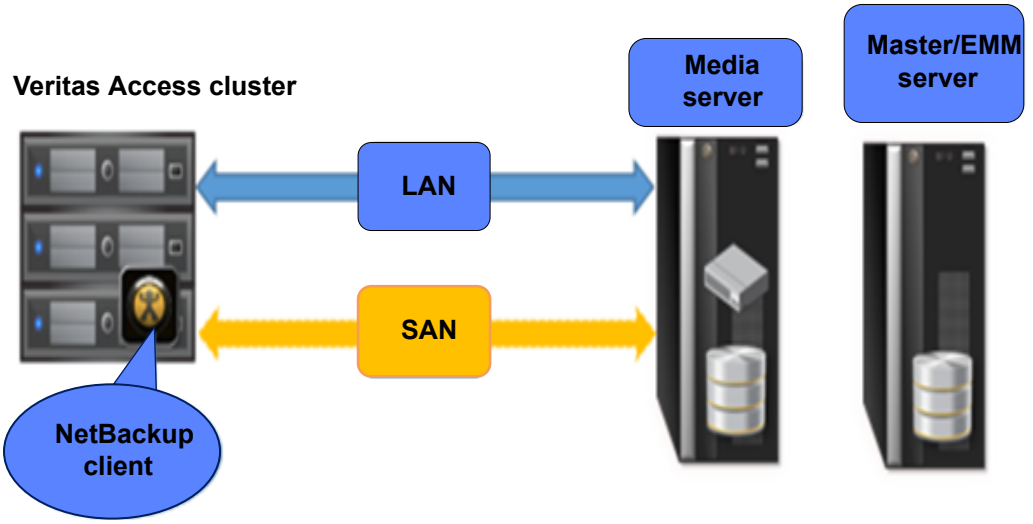
Veritas Access is integrated with Veritas NetBackup so that a NetBackup administrator can back up your Veritas Access file systems to NetBackup master or media servers and retain the data as per your company policy. Once data is backed up, a storage administrator can delete unwanted data from Veritas Access. The NetBackup master and media servers that run on separate computers from Veritas Access are licensed separately from Veritas Access.

You configure NetBackup domain information using any one of the following Veritas Access interfaces:

- CLISH
  The Veritas Access CLISH has a dedicated `Backup>` menu. From the `Backup>` menu, register the NetBackup client with the NetBackup domain. Information is saved in the `bp.conf` file on Veritas Access.
- GUI
  Settings > NetBackup Configuration
  See the online Help for how to configure NetBackup using the GUI.
- RESTful APIs
  See the *Veritas Access RESTful API Guide*.

Consolidating storage reduces the administrative overhead of backing up and restoring many separate file systems. Critical file data can be backed up and restored through the NetBackup client on Veritas Access.

Figure 1-1 Configuration of Veritas Access with NetBackup



About Veritas Access as backup storage for NetBackup

This document describes how Veritas Access fulfills the needs of NetBackup customers looking for a cost-effective solution for moving away from tape backups, yet retain the backed-up data for the long term.

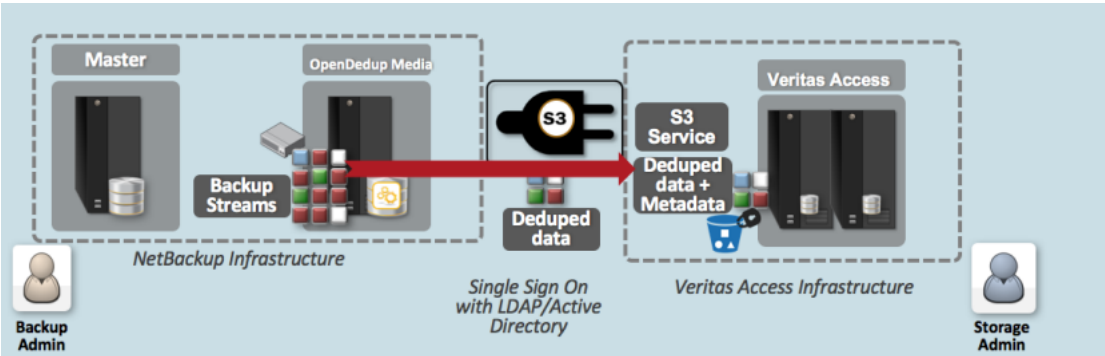
NetBackup is an enterprise-class heterogeneous backup and recovery application. It provides cross-platform backup functionality to a large variety of Windows, UNIX, and Linux operating systems.

Veritas Access is based on the rock-solid and industry-proven Veritas CFS stack. It offers an AWS-compatible S3 protocol as object storage for NetBackup.

Veritas Access is integrated with OpenDedup. OpenDedup is OpenSource software that lets you deduplicate your data to on-premises or cloud storage. OpenDedup installs on top of a NetBackup media server or Veritas Access; it performs data deduplication and stores deduplicated data on Veritas Access over S3.

Figure 1-2 shows how Veritas Access integrates with OpenDedup over S3 to store NetBackup backup streams as deduplicated data.

Figure 1-2

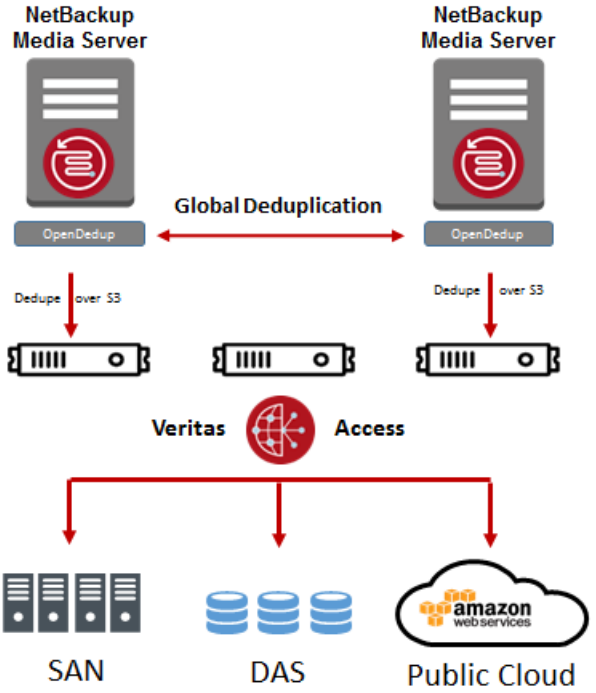


Use cases for long-term data retention

The following are the use cases for long-term data retention (LTR) with OpenDedup:

- Use Case 1: OpenStorage Technology (OST) and OpenDedup hosted on a NetBackup master and/or media server that sends deduplicated backup data to Veritas Access over the S3 protocol. Veritas Access can move this deduplicated data to Amazon Web Services (AWS) S3 or Glacier.
  See "Use case 1: Backing up deduplicated data (OpenDedup and NetBackup) using the S3 protocol to Veritas Access" on page 14.
- Use Case 2: OST hosted on a NetBackup master and/or media server sends backup data to OpenDedup hosted on Veritas Access, which deduplicates the data and sends this data over the S3 protocol to Veritas Access. Veritas Access moves this deduplicated data to AWS S3 or Glacier.
  See "Use case 2: Backing up data (NetBackup) and deduplicating the data (OpenDedup) on Veritas Access" on page 15.

Figure 1-3


NetBackup
Media Server
NetBackup
Media Server
Global Deduplication
OpenDedup
OpenDedup
Dedupe over S3
Dedupe over S3
Veritas
Access
amazon
web services
SAN
DAS
Public Cloud

Chapter 2

System requirements

This chapter includes the following topics:

System requirements for OpenDedup installation

The system requirements for OpenDedup installation are:

- 4GB base memory + 256MB RAM per TB of unique storage
- 120 MB/s per CPU core
- 200 MB/s local disk speed
- 2K IOPS for local media server disk subsystem
- 0 .2 % local disk of logical storage on a NetBackup media server
- 0.2% local disk storage of unique data on a NetBackup media server

Supported configurations and versions

Table 2-1 Supported versions

OpenDedup	Veritas Access	Veritas NetBackup servers	OST
3.4.2.1	7.2.1.1	7.7.3 and 8.0	2.0

Download links:

Veritas Access: Veritas Access 7.2.1.1 DVD

OpenDedup:

https://sort.veritas.com/public/patchcentral/Linux/7.2/access/access-rhel6_x86_64-Patch-7.2.1.1sdfs.tar.gz

OpenStorage Technology (OST):

http://www.opendedup.org/downloads/ost-2.0.tar.gz

Chapter 3

Configuring Veritas Access backup over S3 with OpenDedup and NetBackup

This chapter includes the following topics:

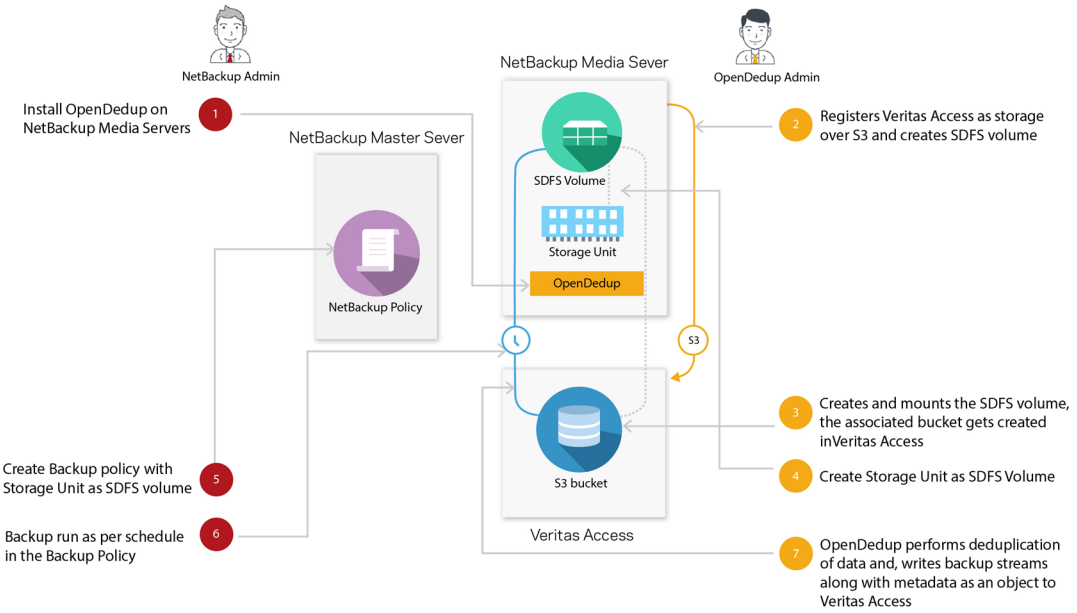
Benefits of using Veritas Access with NetBackup and OpenDedup

- Low-cost, flexible alternative for long-term data retention.
- Eliminate the need for cumbersome, time-consuming tape management.
- Cost-effective and resilient solution that is scale-out (linear performance) and elastic (grow/shrink on demand).

Workflow for OpenDedup

Figure 3-1 illustrates the workflow for OpenDedup for Veritas Access.

Figure 3-1 Workflow for OpenDedup



Use case 1: Backing up deduplicated data (OpenDedup and NetBackup) using the S3 protocol to Veritas Access

SDFS is a deduplicated file system.

To download and install the ost and sdfs rpms

1. On a standard NetBackup media server, run the following commands to install OST:

```
# wget http://www.opendedup.org/downloads/ost-2.0.tar.gz
# tar -xzvf ost-2.0.tar.gz
# cd dist
# ./media-install.sh
```

2. Uninstall the older `sdfs` rpm that is installed as a part of the ost installation.

```
# rpm -e sdfs
```

3. Download and install the Veritas supported OpenDedup rpm using the following commands:

```
# wget https://sort.veritas.com/public/patchcentral/Linux/7.2/
access/access-rhel6_x86_64-Patch-7.2.1.1sdfs.tar.gz
# tar zxvf access-rhel6_x86_64-Patch-7.2.1.1sdfs.tar.gz
# yum -y install fuse (This is optional as fuse may get installed as
part of step 1)
# rpm -ivh rpms/sdfs-3.4.2-1.x86_64.rpm
```

4. Restart the NetBackup service on the NetBackup media server.

```
# /etc/init.d/netbackup stop
# /etc/init.d/netbackup start
```

5. On the NetBackup master server, run the following commands:

```
# ./master-install.sh
# /etc/init.d/netbackup stop
# /etc/init.d/netbackup start
```

Use case 2: Backing up data (NetBackup) and deduplicating the data (OpenDedup) on Veritas Access

To download and install the ost and sdfs rpms

1. On a standard NetBackup media server, run the following commands to install the OST:

```
# wget http://www.opendedup.org/downloads/ost-2.0.tar.gz
# tar -xzvf ost-2.0.tar.gz
# cd dist
# ./media-install.sh
# /etc/init.d/netbackup stop
# /etc/init.d/netbackup start
```

2. On the NetBackup master server, run the following commands:

```
# ./master-install.sh
# /etc/init.d/netbackup stop
# /etc/init.d/netbackup start
```

3. On the server hosting the Veritas Access management console, download and install the Veritas OpenDedup rpm using the following commands:

```
# wget https://sort.veritas.com/public/patchcentral/Linux/7.2/
access/access-rhel6_x86_64-Patch-7.2.1.1sdfs.tar.gz
# tar zxvf access-rhel6_x86_64-Patch-7.2.1.1sdfs.tar.gz
# yum -y install fuse
# rpm -ivh rpms/sdfs-3.4.2-1.x86_64.rpm
```

Creating an S3 bucket on Veritas Access for storing deduplicated backup data from NetBackup

To create an S3 bucket on Veritas Access for storing deduplicated backup data from NetBackup

1. Log on to the Veritas Access GUI as the master user using the following URL:

   https://*Veritas Access Management console IP*:14161/

   You can obtain the Veritas Access Management console IP by logging on to the CLISH using the `su - master` command on the Veritas Access cluster.

2. Create a storage pool for the S3 buckets.

   Click **NAS Infrastructure** in the GUI navigation on the left.

   Select the disks that you want to use for the S3 bucket, and click the **Add to Storage Pool** button to invoke the wizard for storage pool creation.

   Follow the steps in the wizard for creating a new storage pool or adding the disks to an existing pool.

3. Click **Settings > Service Management > Configure Active Directory** to configure AD.

   Enter the required information, such as the **DNS Domain** , **DNS Name Servers**, **AD Domain**, **AD Domain Controller**, and the **AD Admin** and **Password**.

4. Click **Settings > S3 Management** to configure and enable the S3 server.

   Edit the default parameters that are required for the S3 server, such as the storage pool name, underlying S3 bucket layout, and the default size of the bucket.

5. Double-click **S3 Server Status** to start the S3 server.

6. Log out from the GUI, and log on again as an AD user.

   Generate the access key and the secret key for the Veritas Access S3 bucket.

   Save the access key and secret key in a safe location, as Veritas Access does not allow retrieval of a secret access key after initial creation.

7. Log out from the GUI, and log on again as the master user.

8. Registration of Amazon Web Services (AWS) is optional, and is only required in case you need to add an AWS cloud as a storage tier. Without this, backups are stored locally in Veritas Access S3 buckets.

   Click **Settings > Cloud Storage Registration > Add Cloud Subscription** to register the AWS cloud service.

   Enter information for the cloud service provider, name of subscription, access key, and secret key.

9. Activate the long-term data retention (LTR) policies.

   Click **Policies > LTR Policy**.

   Click **Activate** for either the **LTR On-Premises + Cloud** policy or the **LTR On-Premises** policy and provide the storage pool when prompted.

10. Provision the NetBackup bucket using the policy.

    Under **Quick Actions**, click **Provision for NetBackup**.

    Provide the bucket size, underlying layout of the bucket, the access key, and the secret key of the Veritas Access S3 server generated as the AD user.

    If you selected the **LTR On-Premises + Cloud** policy, add information such as which data should be moved to the AWS cloud tier, AWS region, cloud tier type (S3/Glacier), and when the data movement to the cloud should occur.

11. Monitor the progress of the task under **Recent Activity**.

    Make a note of the scale-out file system name that was used for the bucket creation.

12. Click **File Systems**.

    For the scale-out file system that is created, ensure that the **S3 Bucket** column displays **Yes** to indicate that the S3 bucket is enabled.

    You may need to wait for some time for this change to be reflected in the GUI.

13. Right-click the ellipses (additional options), and click **Generate LTR Script**.

14. Copy the LTR script to the host where OpenDedup is installed. It can be the host where the NetBackup media server is installed or the Veritas Access server.

15 Run the downloaded LTR script. The LTR script requires the Veritas Access S3 keys (access and secret key) as arguments that were generated as the AD user.

The LTR script creates the OpenDedup file system and prompts for the entry in the `/etc/hosts` file for the bucket to IP address mapping.

Output of LTR script execution:

```
[root@host1 ~]# sh LTRscript.sh <Access key> <Secret Key>
====================================
Insert the below details in /etc/hosts file
10.100.100.1 4f459a2d-736e-4be5-9c5a-f821fbc198fds3bucket.s3.access
====================================
Attempting to create SDFS volume ...
Volume [S3fs1497356186] created with a capacity of [10.00GB]
check [/etc/sdfs/S3fs1497356186-volume-cfg.xml] for configuration
details if you need to change anything
```

Note: The volume name highlighted above and its equivalent .xml file are used to mount and update the SDFS volume parameters in later steps.

16 Add the IP associated with the virtual hosted-style bucket name (generated from the LTR script) in the `/etc/hosts` file on the media server.

17 Mount the SDFS volume under `/opendedupe/volumes/` on the host where OpenDedup is installed.

```
# mkdir /opendedupe/volumes/filesystem_name
```

```
# mount -t sdfs filesystem_name /opendedupe/volumes/filesystem_name
```

The `mount` command mounts a bucket on the Veritas Access cluster or the NetBackup media server. The mount process might time out with an error. If it does, wait two minutes and try again.

Note: After mounting the SDFS volume, it will start listening on a specific port, usually starting from 6442. If OpenDedup is installed on Veritas Access, then ensure that the corresponding firewall rules are updated to allow traffic to this port.

Port information can be found using the `mount` command.

Example:

```
[root@host1 ~]# mount | grep opendedupe
sdfs:/etc/sdfs/S3fs1497346133-volume-cfg.xml:6443 on
/opendedupe/volumes/S3fs1497346133 type fuse
(rw,nosuid,nodev,allow_other,allow_other)
sdfs:/etc/sdfs/S3fs1497258807-volume-cfg.xml:6442 on
/opendedupe/volumes/pool1 type fuse
(rw,nosuid,nodev,allow_other,allow_other)
```

18 (Optional) Add the volume to `fstab` by adding the following line in: `/etc/fstab`.

```
filesystem_name /opendedupe/volumes/filesystem_name sdfs defaults 0 0
```

19 Update the URL tag in the `/etc/sdfs/ostconfig.xml` present on the NetBackup media server based on the following two cases:

Use case 1: OpenDedup on a NetBackup server

```
<URL>
http://localhost:6442/
</URL>
```

Use case 2: OpenDedup on Veritas Access

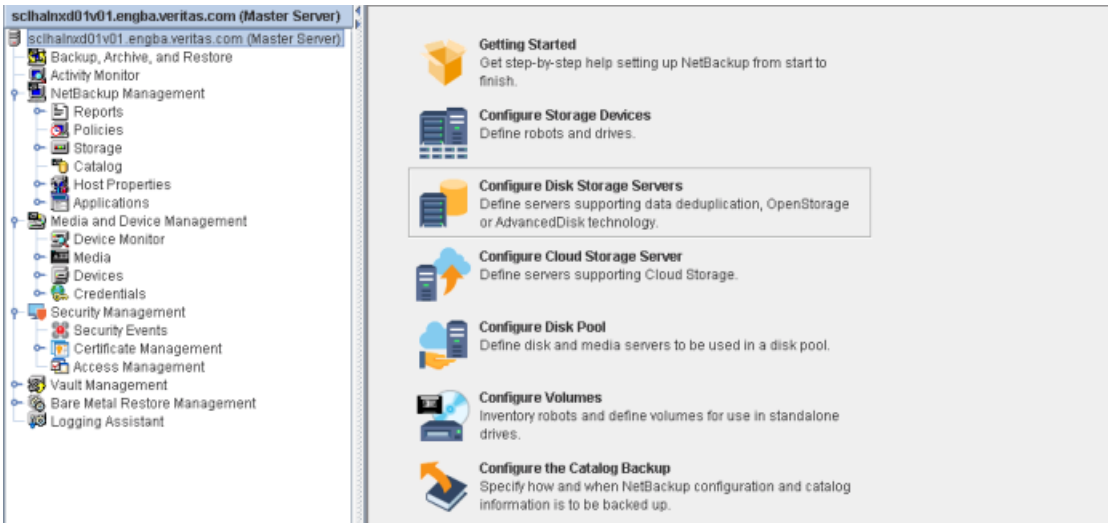
```
<URL>
http://<Veritas Access server hostname>:6442/
</URL>
```

Note: The Veritas Access server hostname should be DNS-resolvable and reachable from the NetBackup server. This should be the same node in the Veritas Access cluster where the SDFS volume is mounted in Step 17.

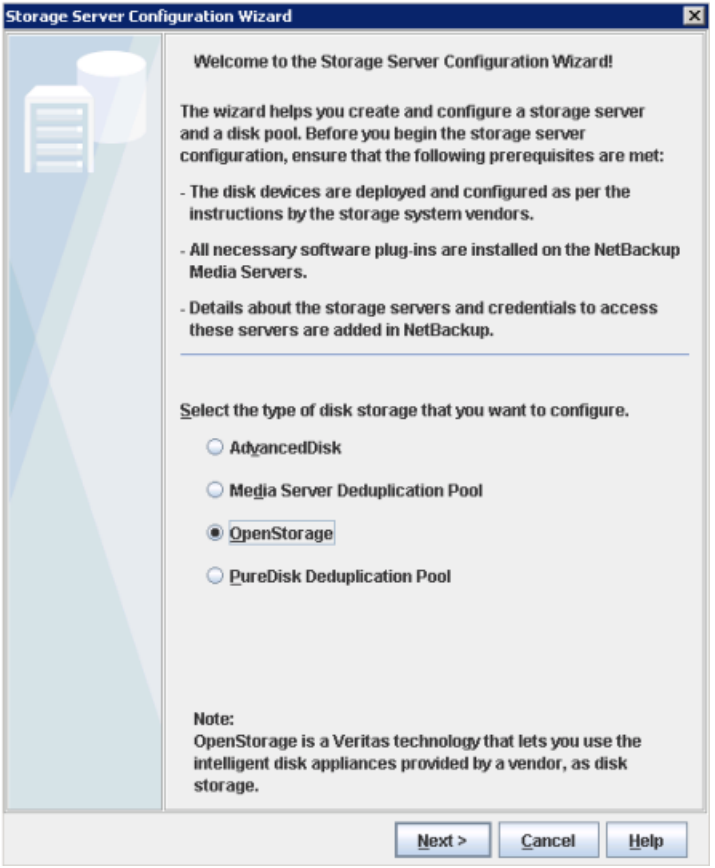
Creating an OST disk pool and STU in the NetBackup console

To create an OpenStorage Technology (OST) disk pool and storage unit (STU) in the NetBackup console

1. Log on to the NetBackup master server from the Java console.
2. Select **Configure Disk Storage Servers**.



3 Select the **OpenStorage** option from the **Select the type of disk storage that you want to configure** section of the dialog.



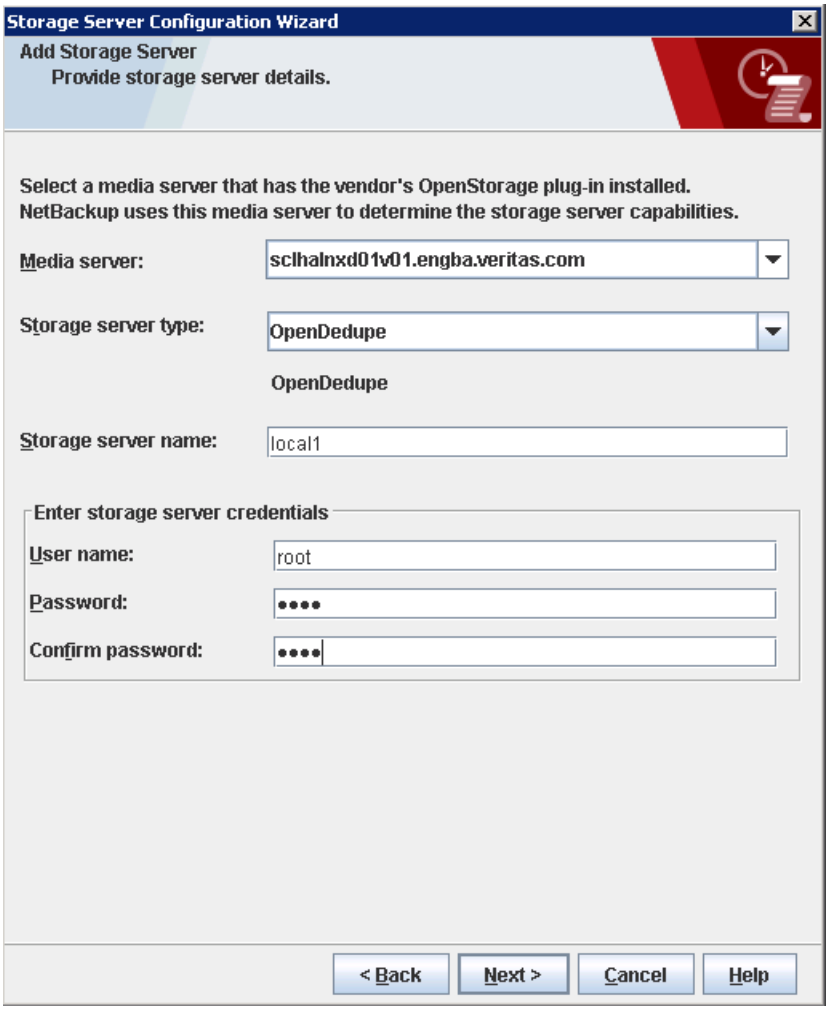
4 Add the following options to the **Storage Server Details**:

- **Storage server type**: OpenDedupe

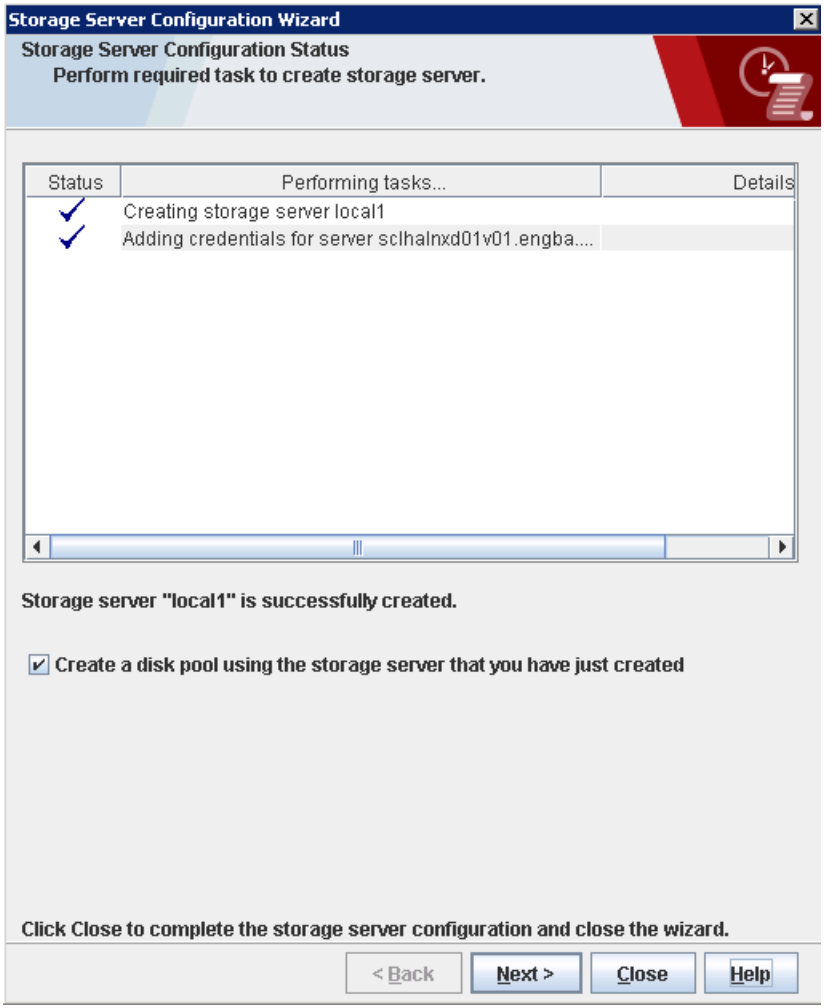
  Note: The **Storage server type** field is case-sensitive. **OpenDedupe** has to be entered exactly as shown in the screen shot.

- **Storage Server name**: The name in the <NAME></NAME> tag in the `/etc/sdfs/ostconfig.xml` file. This is `local` by default.
- **Username**: Anything can go in this field. It is not used.

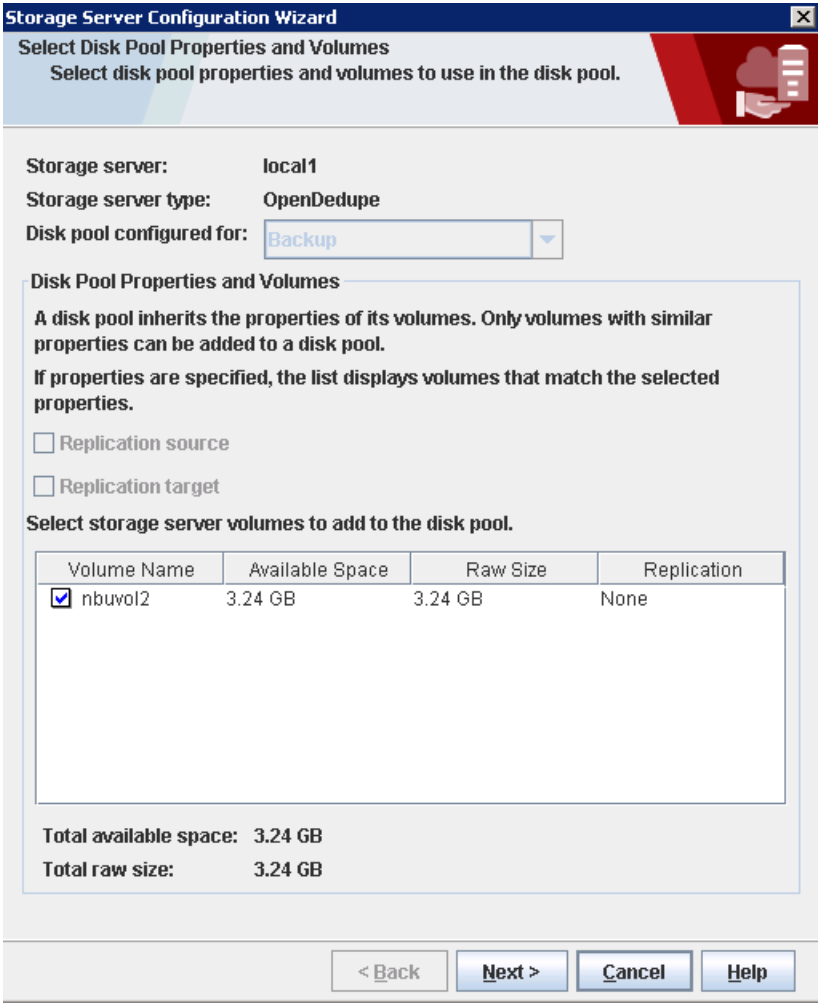
- **Password/Confirm Password**: Anything can go in this field as well.



5 Finish supplying entries for the storage configuration wizard and make sure **Create a disk pool using the storage server that you just created** is selected.



6 Select the storage pool that was just created.

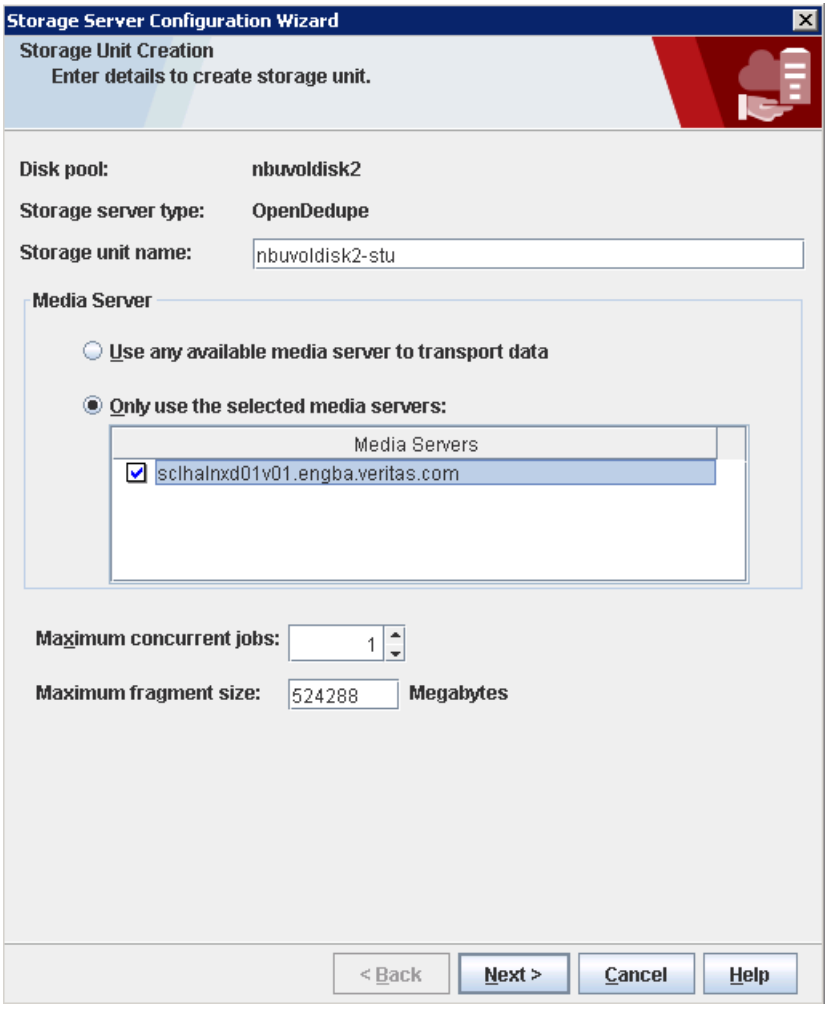


7 Add a disk pool name.

8 Finish the wizard entries and select **Create a storage unit using the disk pool that you just created**.

9 In the **Storage Unit Creation** page, select **Only use the selected media servers** and select the media server that the storage was created on. For maximum concurrent jobs select **8**.

Note: If you plan to run concurrent jobs for this STU, increase the **Maximum concurrent jobs** count to the desired value.



Setting up multiple NetBackup media servers in the same domain

To set up the OST connector on multiple NetBackup media servers in the same domain, additional steps must be taken on each NetBackup media server before adding the storage pools in NetBackup.

To set up multiple NetBackup media servers in the same domain

1. Follow the instructions for setting up the OST connector on each media server that uses the OST connector.

   See "Use case 1: Backing up deduplicated data (OpenDedup and NetBackup) using the S3 protocol to Veritas Access" on page 14.

2. Edit `/etc/sdfs/ostconfig.xml` and change the <name> tag to something unique in the NetBackup domain, such as the host name with an incremented number, for example:

   ```
   <NAME>hostname-0</NAME>
   ```

3. Follow the instructions in the "Creating an OST disk pool and STU in the NetBackup console" section and use the name in the <NAME> tag as the **Storage Server** name.

   See "Creating an OST disk pool and STU in the NetBackup console" on page 21.

   See "Use case 1: Backing up deduplicated data (OpenDedup and NetBackup) using the S3 protocol to Veritas Access" on page 14.

Setting up multiple SDFS volumes on a NetBackup media server

The OST connector supports multiple SDFS volumes on the same media server but additional steps are required to support this configuration.

To set up multiple SDFS volumes on a NetBackup media server

1. Follow the instructions for setting up the OST connector on each NetBackup media server that uses the OST connector.

   See "Use case 1: Backing up deduplicated data (OpenDedup and NetBackup) using the S3 protocol to Veritas Access" on page 14.

2 Edit the `/etc/sdfs/ostconfig.xml` and add a new <CONNECTION> tag inside of the <CONNECTIONS> tag for the new volume.

Add a name that is unique to the <NAME> tag and specify the new volume name in the <LSU_NAME> tag (pool1).

In the new <CONNECTION> tag, add the port number identified by running the `mount` command to the <URL> tag (http://localhost:6443/) as shown in the example output.

```
[root@host1 ~]# mount | grep opendedupe
sdfs:/etc/sdfs/S3fs1497346133-volume-cfg.xml:6443 on
/opendedupe/volumes/S3fs1497346133 type fuse
(rw,nosuid,nodev,allow_other,allow_other)
sdfs:/etc/sdfs/S3fs1497258807-volume-cfg.xml:6442 on
/opendedupe/volumes/pool1 type fuse
(rw,nosuid,nodev,allow_other,allow_other)
```

The following is a complete example of an `ostconfig.xml` file with two volumes.

```
<!-- This is the config file for the OST connector for opendedup and Netbackup -->
<CONNECTIONS>
<CONNECTION>
<!--NAME is the local server name that you will reference within Netbackup -->
<NAME>
local
</NAME>
<LSU_NAME>
svol4
</LSU_NAME>
<URL>
http://localhost:6442/
</URL>
<!--PASSWD - The password of the volume if one is required for this sdfs volume -->
<PASSWD>admin</PASSWD>
<!-
<SERVER_SHARE_PATH>
A_SUBDIRECTORY_UNDER_THE_MOUNT_PATH
</SERVER_SHARE_PATH>
-->
</CONNECTION>
<!-- Below is the new volume-->
<CONNECTION>
<!--NAME is the local server name that you will reference within Netbackup -->
<NAME>
```

```
hostname0
</NAME>
<LSU_NAME>
svol10
</LSU_NAME>
<URL>
http://localhost:6443/
</URL>
<!--PASSWD - The password of the volume if one is required for this sdfs volume -->
<PASSWD>admin</PASSWD>
<!--
<SERVER_SHARE_PATH>
A_SUBDIRECTORY_UNDER_THE_MOUNT_PATH
</SERVER_SHARE_PATH>
-->
</CONNECTION>
</CONNECTIONS>
```

Chapter 4

Configuring backup and restore using NetBackup policies

This chapter includes the following topics:

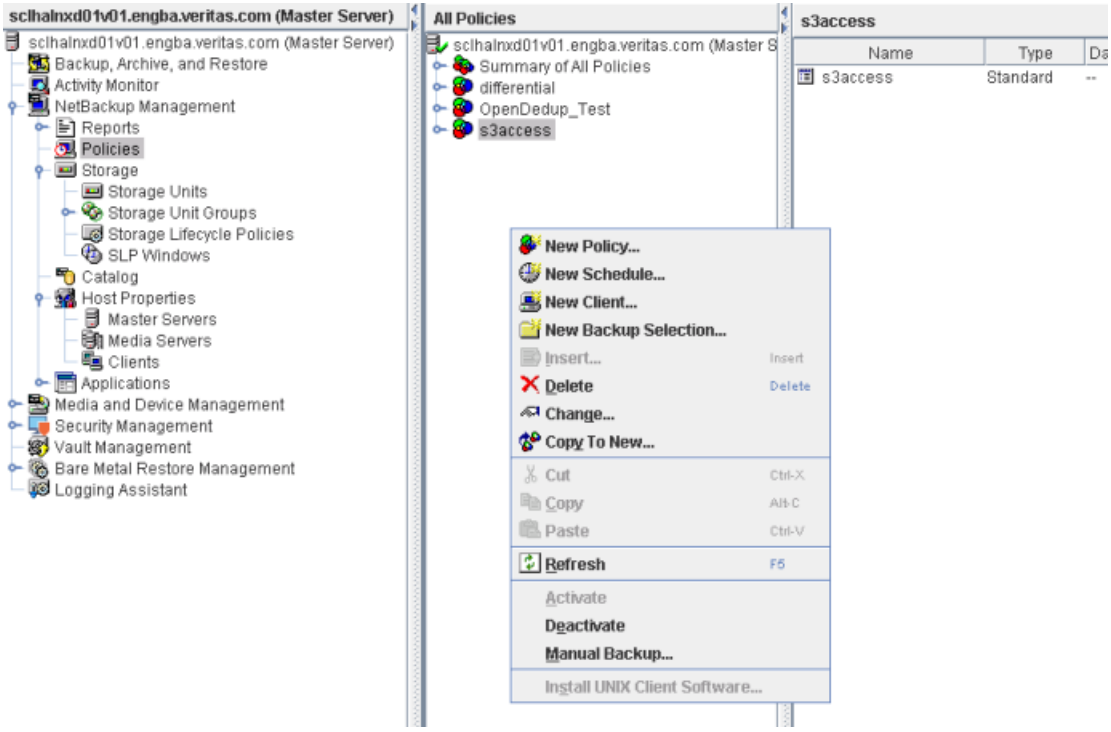
Backup and restore

After completing the configurations, the following are the backup and restore steps.

Policy creation

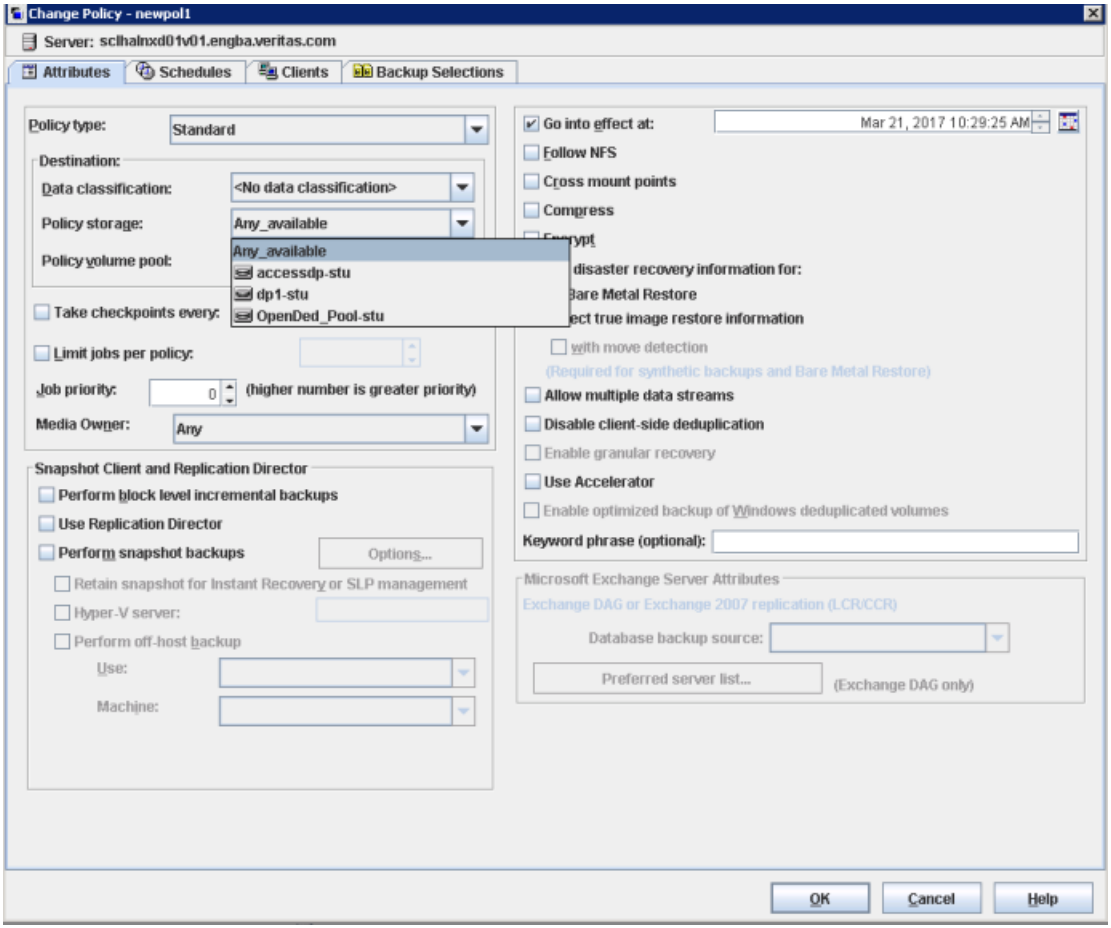
To create policies

1 Right-click on **Policies** within the NetBackup console and click on **New Policy**.



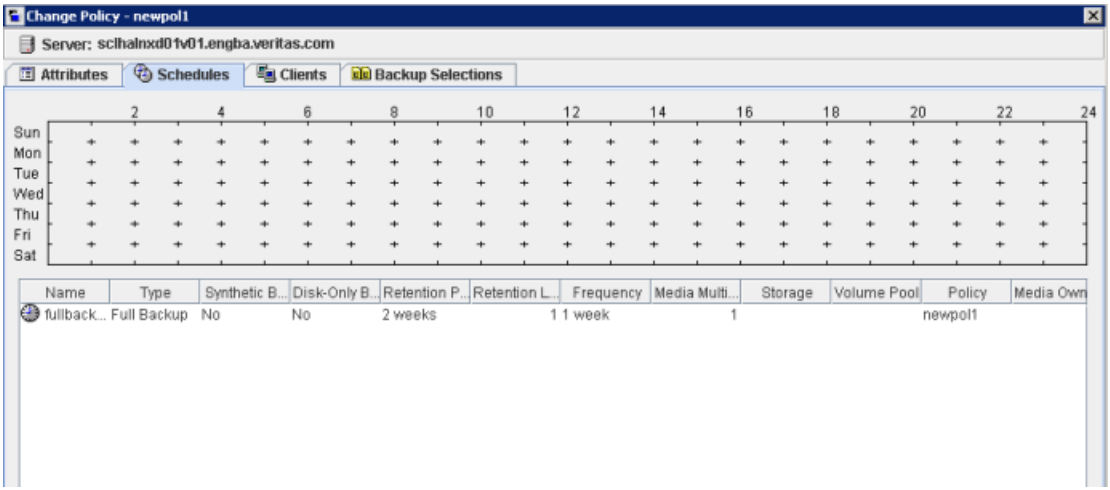
2 Provide the following information for policy creation.

- Policy name
- From the **Attributes** tab, select the appropriate storage unit under **Policy storage**.



Note: The **Policy Storage** selection should be the storage unit created for OpenDedup earlier.

3 Under the **Schedule** tab, enter the name of the schedule. For example, **fullbackup**.



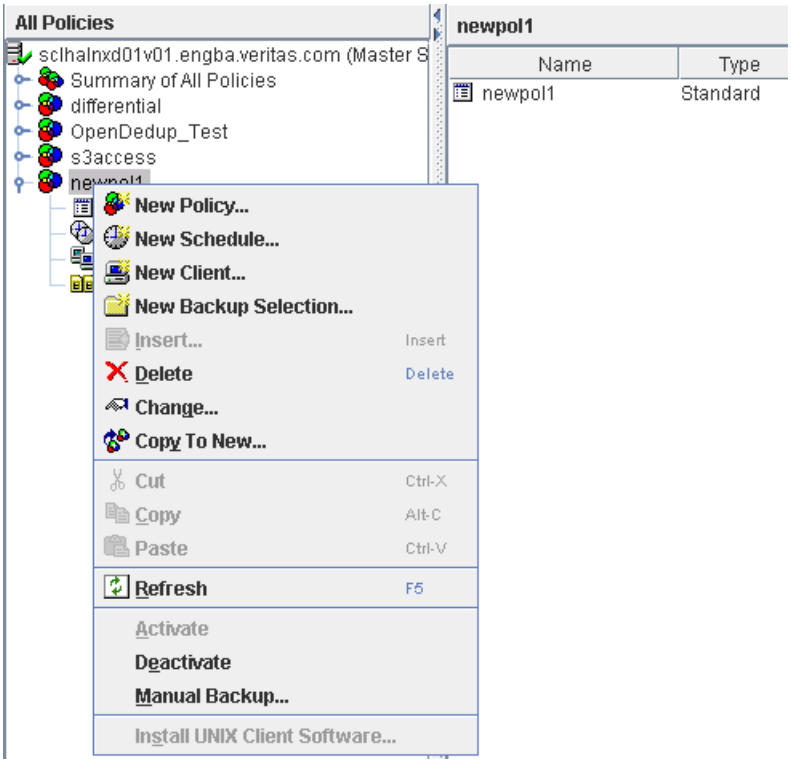
4 Provide client information under the **Clients** tab.

5 Provide the folders that need to be backed up under **Backup Selections**.

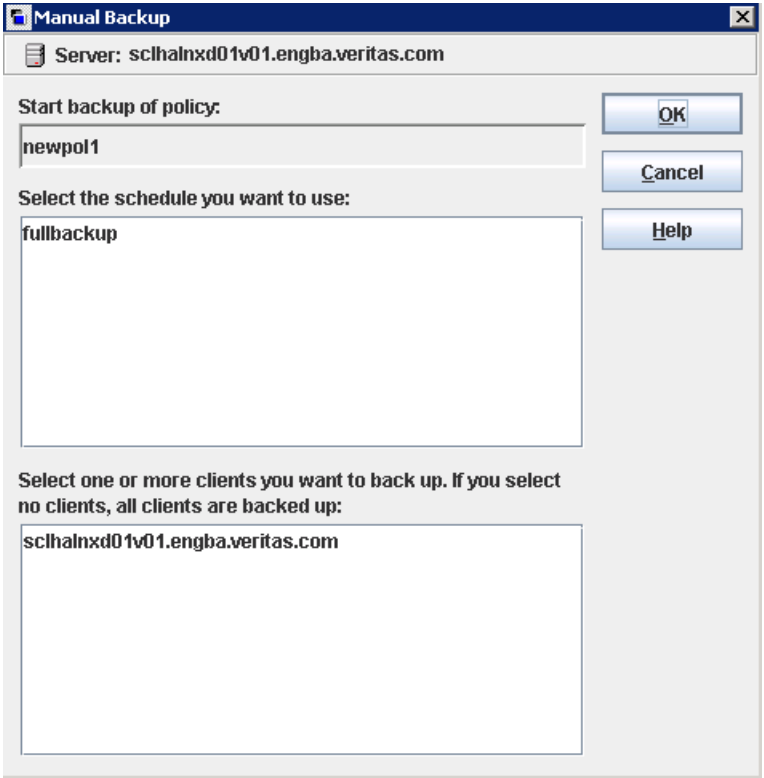
Running a backup policy manually

To run a backup policy manually

1 Once the policy is created, right-click on the name of the policy that you want to run under **Summary of All Policies**, and click on **Manual Backup**.

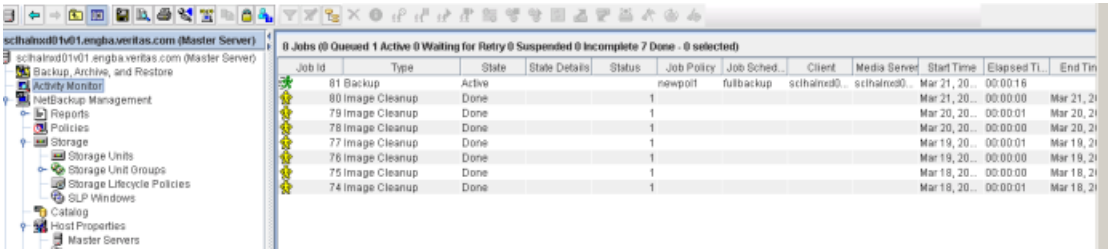


2 Select the schedule that you want to use and click **OK**.



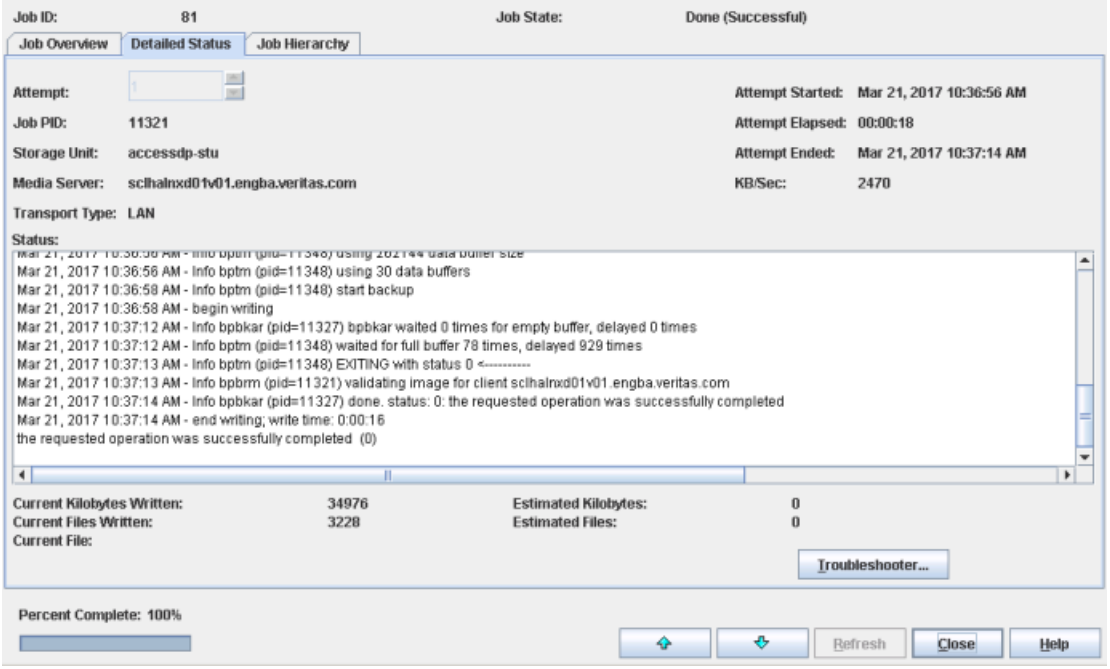
This starts the manual backup with the policy.

3 To verify the status of the backup, go to **Activity Monitor**.



4 Select the appropriate job from the displayed jobs.

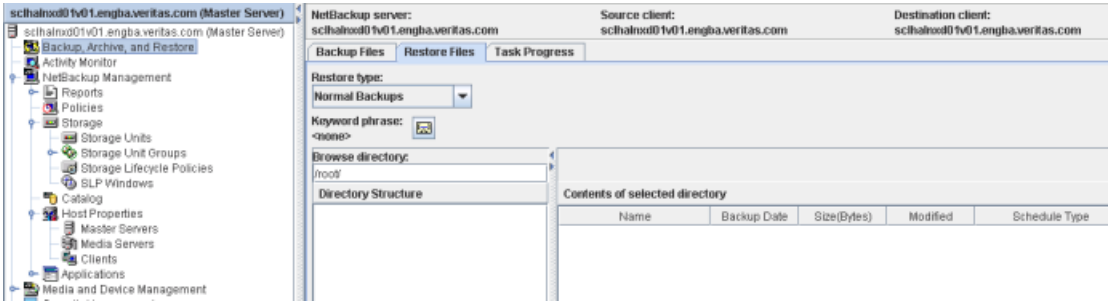
5 Click on the **Detailed Status** tab in the new window to check on the status of the backup.



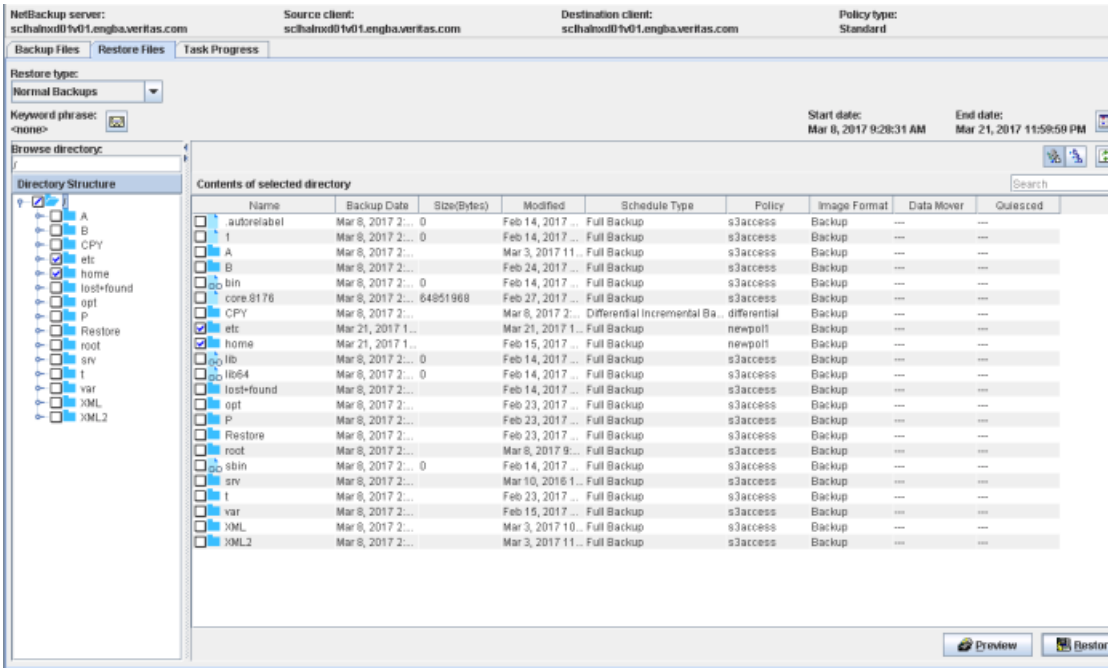
Restoring backed up files

To restore backed up files

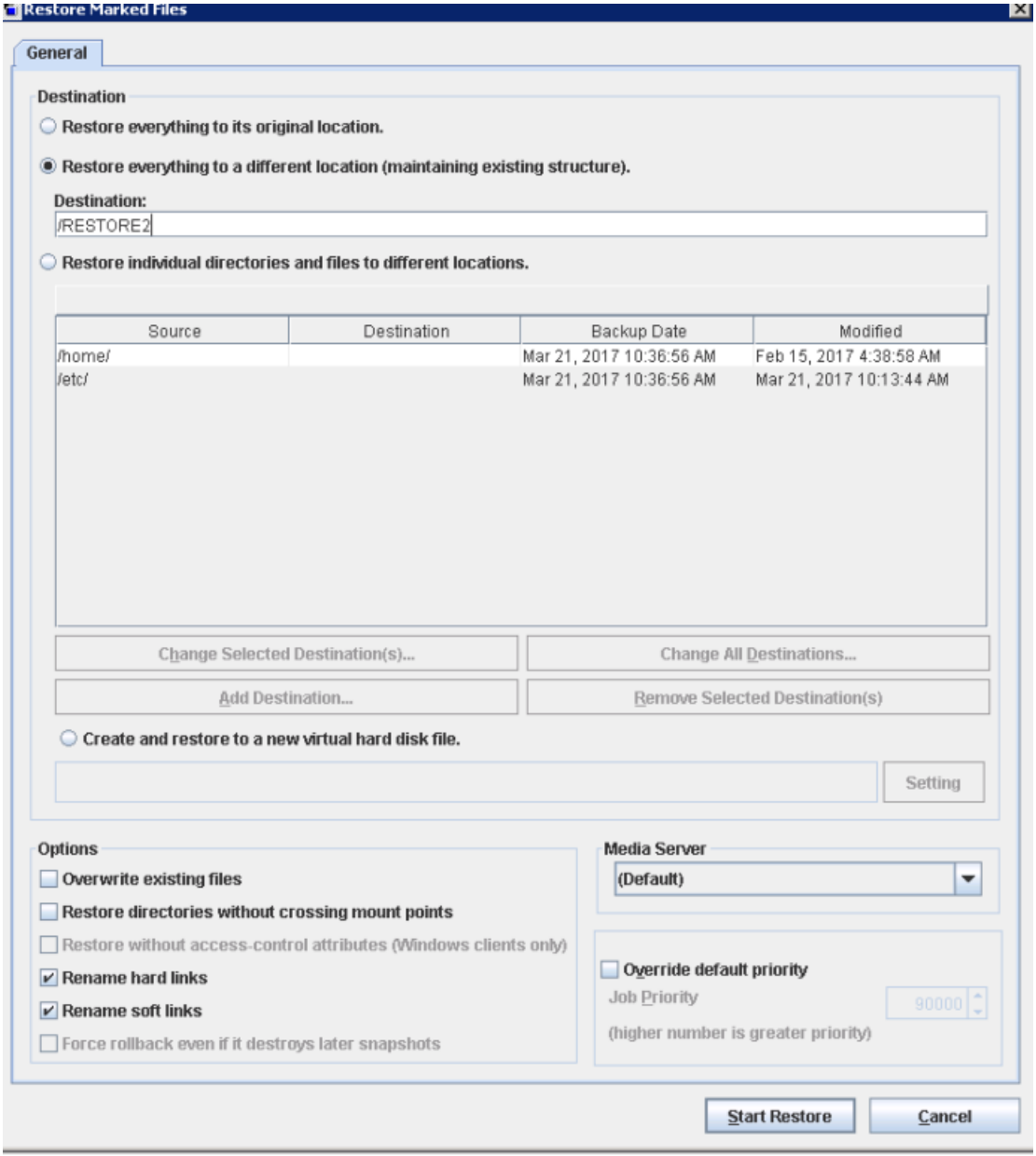
1. Create a directory where you want to restore the backed up files.
2. Go to the **Restore Files** tab under **Backup, Archive, Restore**.



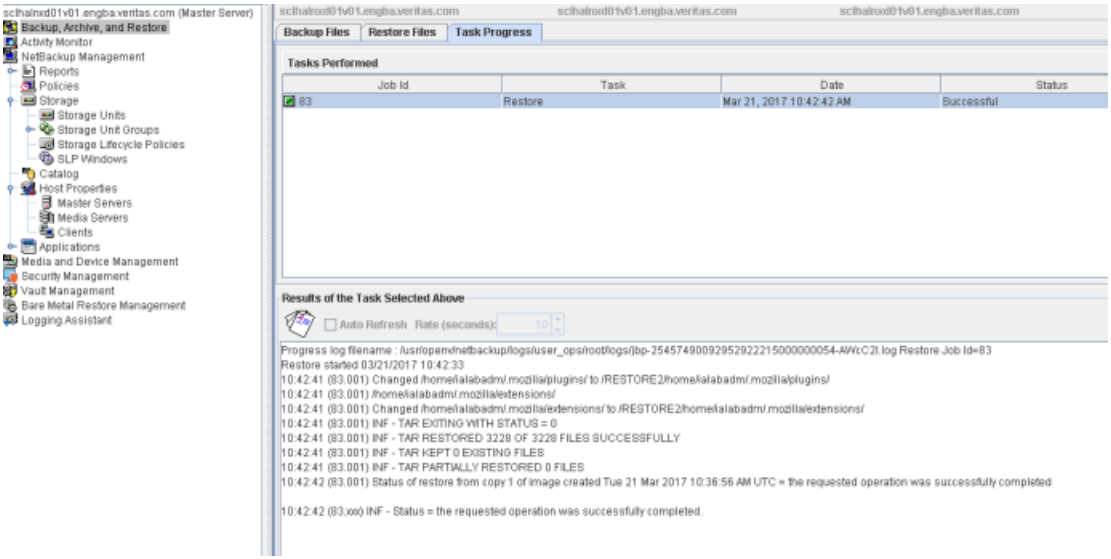
3. Go to the browse directory and select the appropriate files to restore and click **Restore**.



4 Provide the location where the files should be restored, and click on the **Start Restore** button.



5 To view the progress of the restore operation, click **Yes** on the **Restore Initiated** window.



Chapter 5

Troubleshooting

This chapter includes the following topics:

Unmounting the SDFS volume before restarting Veritas Access or the NetBackup media server

Before restarting Veritas Access or the NetBackup media server, create a backup copy of the SDFS volume and unmount the SDFS volume.

To perform a clean unmount of the SDFS volume

1. Create a backup copy of the SDFS volume .xml file in the `/etc/sdfs` directory.
2. Unmount the SDFS volume and wait for the `jsvc` process to exit before restarting Veritas Access.

Log locations for troubleshooting

Veritas Access S3 logs

- `/opt/VRTSnas/log/portald.log`
- `/opt/VRTSnas/log/portald_access.log`

SDFS logs

SDFS creates its logs under `/var/logs/sdfs/<volume-name>-volume-cfg.xml.log`. Errors can be identified in this log file.

OST plug-in logs

The OpenDedup OST plug-in log can be found in `/tmp/logs/opendedup.log`.

NetBackup logs

Pertinent OST-related errors and logging are trapped in the `bptm` log. NetBackup logging for `bptm` can be enabled by creating the `bptm` logging directory:

```
mkdir /usr/openv/netbackup/logs/bptm
```

Veritas Access support debug information upload command

```
CLISH> support debuginfo upload path
```

Additional resources

See the following documentation for more information on Veritas Access, OpenDedup, and Veritas NetBackup:

- *Veritas Access Installation Guide* for the supported NetBackup clients and the OpenDedup ports.
- *Veritas Access Troubleshooting Guide* for setting the NetBackup client log levels and debugging options.
- Veritas NetBackup product documentation on the SORT website.
- OpenDedup product documentation on the OpenDedup website.

Helper script for generating access/secret keys

Create the access and the secret keys using the Veritas Access helper script:

- Location of the helper script:
  `/opt/VRTSnas/scripts/utils/objectaccess/objectaccess_client.py`
- The Veritas Access helper script can be used from any client system that has Python installed.
- To run the script, your S3 client needs to have the `argparse` and `requests` Python modules.

If these modules are missing, install both these modules using `pip` or `easy_install`.

- Add the ADMIN_URL name in your `/etc/hosts` file.
  where the *ADMIN_URL* is admin.<*cluster_name*> and the port is 8144. This url should point to the Veritas Access management console IP address.
- Create the access and the secret key using the Veritas Access helper script by providing the user name, password, and ADMIN_URL (check the online Help of the Veritas Access helper script for all of the provided operations like `list key` and `delete key`).
  Create a secret key:

```
clus_01:~ # ./objectaccess_client.py --create_key
--server admin.clus:8144 --username localuser1 --password root123
--insecure
UserName                : localuser1
AccessKeyId             : Y2FkODU2NTU2MjVhYzV
Status                  : Active
SecretAccessKey         : ODk0YzQxMDhkMmRjM2M5OTUzNjI5OWIzMDgyNzY
```

  The <*localuser1*> is the local user created on both the Veritas Access cluster nodes with same unique ID.
  List a secret key for the specified user:

```
clus_01:~ # ./objectaccess_client.py --list_key --server
admin.clus:8144 --username localuser2 --password root123 --insecure
```

  Delete a secret key for the specified user:

```
clus_01:~ # ./objectaccess_client.py --delete_key
ZTkyNDdjZTViM2EyMWZ --server admin.clus:8144 --username localuser2
--password root123 --insecure
```

- If the object server is enabled without the `SSL` option, you need to add the `--insecure` option.

```
clus_01 ~# ./objectaccess_client.py --server
admin.clus:8144 --username <uname> --create_key --insecure
```