NETAPP TECHNICAL REPORT

SnapVault Disk to Disk Backup on Windows Environment

Deepak S.N., NetApp
TR-3667

ABSTRACT

This technical report enumerates and explores SnapDrive integration with SnapVault for performing disk to disk backup on Windows environment. Best practices, architecture, and feature-based recommendations are provided to facilitate the construction of a system that meets performance and functionality expectations.

TABLE OF CONTENTS

1 PURPOSE AND SCOPE

This technical report describes how to protect, back up, restore, and copy data between storage systems that run Data ONTAP® software. Data ONTAP SnapVault® uses disk-to-disk technology to perform disk-to-disk backup and restore data. SnapDrive® is the host side application that integrates with Windows virtual disk service to create LUNs on the storage system, present these LUNs as local disks on the Window hosts, and create Data ONTAP Snapshot copies to create point-in-time image of data stored on the LUNs.

The paper explains how to configure and restore data using SnapDrive® and SnapVault features available in Data ONTAP in a Windows environment. It also provides multiple configuration designs on storage to achieve disk-to-disk back up and restore.

2 INTENDED AUDIENCE AND ASSUMPTIONS

This paper is intended for system and storage architects for designing NetApp storage appliances into IT environments for performing disaster and business continuity. For the concepts, methods, and procedures in this document to be useful for readers, the following assumption is made:

- The reader has a minimum general knowledge of NetApp hardware and software solutions, particularly in the area of Fibre Channel and iSCSI.
- The reader must be familiar with Data ONTAP features such as Snapshot, SnapVault, and SnapMirror® technologies.

3 INTRODUCTION

3.1 SNAPDRIVE FOR WINDOWS

SnapDrive for Windows is a host side program which automates storage provisioning tasks and simplifies the process of taking error free host consistent Snapshot copy on NetApp FAS storage systems. It provides Windows system and storage administrators access to storage provisioning functionality such as creating LUNs and performing a point-in-time data backup Snapshot copies, which are at file system and application data consistent. It also restores data from these Snapshots copies. SnapManager for Exchange, SQL, and SharePoint use the underlying functionality provided by SnapDrive for Windows for Snapshot and SnapVault copy management

3.2 DATA PROTECTION FEATURES ON DATA ONTAP

Data protection means backing up data and being able to recover it. To protect data, make copies so that you can restore the data even if the original is no longer available. Depending on data protection and backup requirements, Data ONTAP offers a variety of features and methods to assist against disaster loss of data. This paper discusses how to use SnapVault technology to protect data.

Table 1I) Data protection features.

Data Protection Feature	Description
Aggr copy	This is a fast block copy of data stored in aggregates. It enables you to quickly copy blocks of stored system data from one aggregate to another. For information about aggregates and aggr copy, see the Storage Management Guide.
Snapshot™	Backup within a volume. This feature allows you to manually or automatically create, schedule, and maintain multiple backups (also called Snapshot copies) of data on a volume. Snapshot copies use only a minimal amount of additional volume space, and do not have a performance cost. Snapshot copies are also used to create clones of FlexVol® volumes and Data ONTAP LUNs.

Data Protection Feature	Description
SnapRestore®	Fast, space efficient restoration of large volumes of data backed up to Snapshot copies. The SnapRestore feature performs on-request Snapshot recovery from Snapshot copies on an entire volume.
SnapMirror®	Volume-to-volume and qtree-to-qtree replication. This feature enables you to periodically make Snapshot copies of data on one volume or qtree, replicate that data to a partner volume or qtree usually on another storage system, and archive one or more iterations of that data as Snapshot copies. Replication on the partner volume or qtree results in quick availability and restoration of data, from the point of the last Snapshot copy, should the storage system containing the original volume or qtree be disabled.
SnapVault®	Centralized backup of multiple qtrees on multiple storage systems using Snapshot technology. This feature enables you to back up qtrees on multiple volumes and storage systems to a single SnapVault Secondary specialized for quick backup and restore of its sources. You can also install the Open Systems SnapVault agent on Windows NT, Windows 2000, Solaris, Linux, AIX, or HP-UX systems. This agent also allows SnapVault to back up from and restore data to these systems.
Tape backup dump and restore commands	Tape backup and restore. The dump and restore commands allow you to back up Snapshot copies to tape. The dump command takes a Snapshot copy of the volume and then copies that data to tape. Because the Snapshot copy, not the active file system, is backed up to tape, Data ONTAP can continue its normal functions while the tape backup takes place.
vol copy	Fast block-copy of data from one volume to another. The vol copy command enables you to quickly block-copy stored data from one volume to another.
SyncMirror® (active/active configuration required)	Continuous mirroring of data to two separate volumes. This feature allows you to mirror real-time data to matching volumes physically connected to the same storage system head. In case of irrecoverable disk error on one volume, the storage system automatically switches access to the mirrored volume. Active-Active configuration is required for this feature.
MetroCluster	SyncMirror functionality enhanced to provide continuous volume mirroring over 500-meter to 30-kilometer distances.

3.3 HOW SNAPVAULT WORKS

SnapVault is a disk-based storage backup feature of Data ONTAP that enables data stored on multiple storage systems to be backed up to a central, secondary storage system (SnapVault Secondary) as read-only Snapshot copies. SnapVault Secondary is a data storage system running Data ONTAP, such as a NearStore or a FAS system.

Initially, a complete copy of the data from the Primary is backed up to the SnapVault Secondary. This data is stored by the WAFL® file system, and Snapshot image of the data volume is created.

For the subsequent backups, SnapVault transfers only data blocks that have changed since the previous backup.

4 SNAPDRIVE WITH SNAPVAULT

SnapDrive software integrates with Windows Volume Manager for storage systems to serve as virtual storage devices for application data in Windows Server 2003 and 2008 environments. SnapDrive is dependent on the virtual disk service. The virtual disk service must be started on the host prior to installing SnapDrive. SnapDrive manages virtual disks (LUNs) on a storage system, making these LUNs available as local disks on Windows hosts. This allows Windows hosts to interact with the LUNs just as if they belonged to a directly attached Redundant Array of Independent Disks (RAID) system.

SnapDrive provides the following additional features:

- Enables online storage configuration, LUN expansion, and streamlined management.
- Integrates Data ONTAP Snapshot technology, which creates point-in-time images of data stored on LUNs.
- Works in conjunction with SnapMirror software to facilitate disaster recovery from either asynchronously or synchronously mirrored destination volumes.
- Enables SnapVault updates of qtrees to a SnapVault destination.
- Enables management of SnapDrive on multiple hosts.

SnapDrive 6.0 provides the following additional features:

- The older version SnapDrive required that CIFS shares be created for the volume on the storage system, before creating a new LUN. SnapDrive 6.0 eliminates this CIFS dependencies, and uses the HTTP/HTTPS protocols to communicate with the storage system for any storage activity.
- SnapDrive provides a new Storage System Explorer user interface from the MMC GUI, which allows the user to browse to the LUN locations on the storage system and perform LUN management operations. On Windows 2008 Server, SnapDrive MMC extends the Storage node of the Server Manager Snap-In. SnapDrive MMC 3.0 supports remote SnapDrive management functionalities on Windows XP and Microsoft Vista client machines, and allows managing SnapDrive instances running on Windows 2003 and Windows 2008 servers.

5 SNAPVAULT STORAGE LAYOUT REFERENCES

SnapVault is a disk-based storage backup feature of Data ONTAP that enables data stored on multiple storage systems to be backed up to a central SnapVault Secondary quickly and efficiently as read-only Snapshot copies.

The physical and logical setups to deploy SnapVault backup systems in a Data Center are as follows:

- SnapVault Primary Connected to SnapVault Secondary
- SnapVault Primary Connected to Many SnapVault Backup Systems
- SnapVault Primary and SnapVault Secondary with SnapMirror
- SnapVault Primary and SnapVault Secondary with Tape Drive
- SnapVault Primary and SnapVault Secondary with Active/Active Cluster Storage

- **SnapVault Primary Connected to SnapVault Secondary (Basic Configuration)**

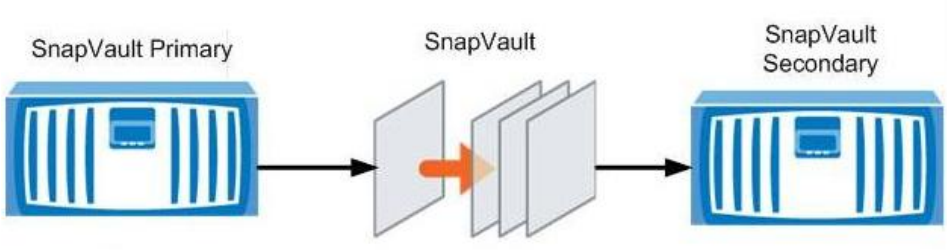

Figure 1) SnapVault Primary Connected to SnapVault Secondary.

- **SnapVault Primary Connected to Many SnapVault Backup Systems**

  In this setup, one SnapVault Primary is connected to many SnapVault Secondary systems. By this, you can utilize all the storage systems efficiently and also provide scalability for managing data.

  Figure 2 depicts a basic reference where a single SnapVault Primary A backs up Snapshot copies to two SnapVault Secondaries A and B.

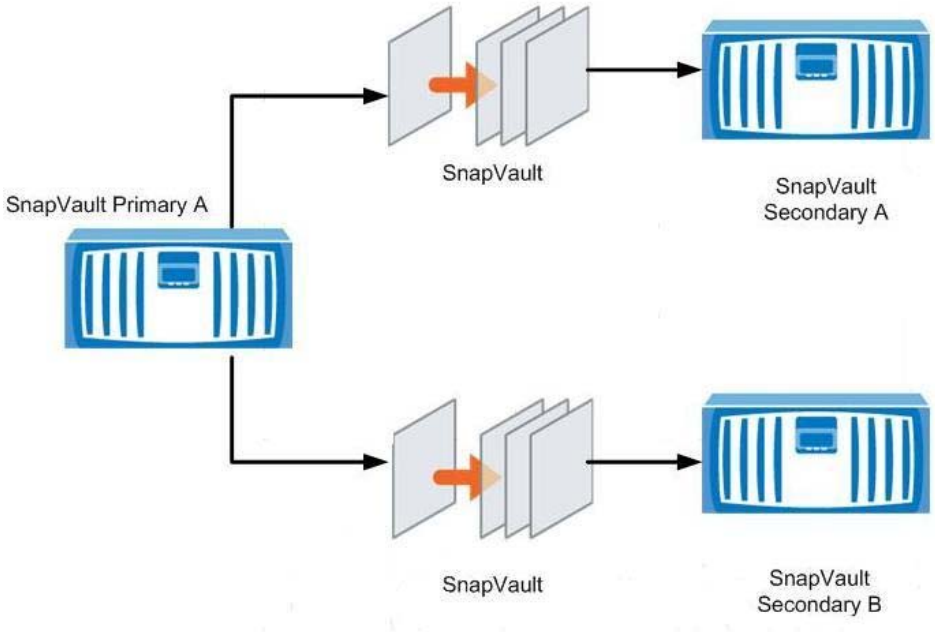

Figure 2) SnapVault Primary to many SnapVault Backup systems.

- **SnapVault Primary and SnapVault Secondary with SnapMirror**

  Using this setup, you can back up data from SnapVault Secondary to a SnapMirror destination system, which is performed on a volume level for addition data protection (Figure 3). SnapMirror and SnapVault can co-exist with each other. If SnapVault Secondary stops functioning, data can stilll be accessed from the SnapMirror destination system. It can be also be converted to SnapVault Secondary for further updates from SnapVault Primary.

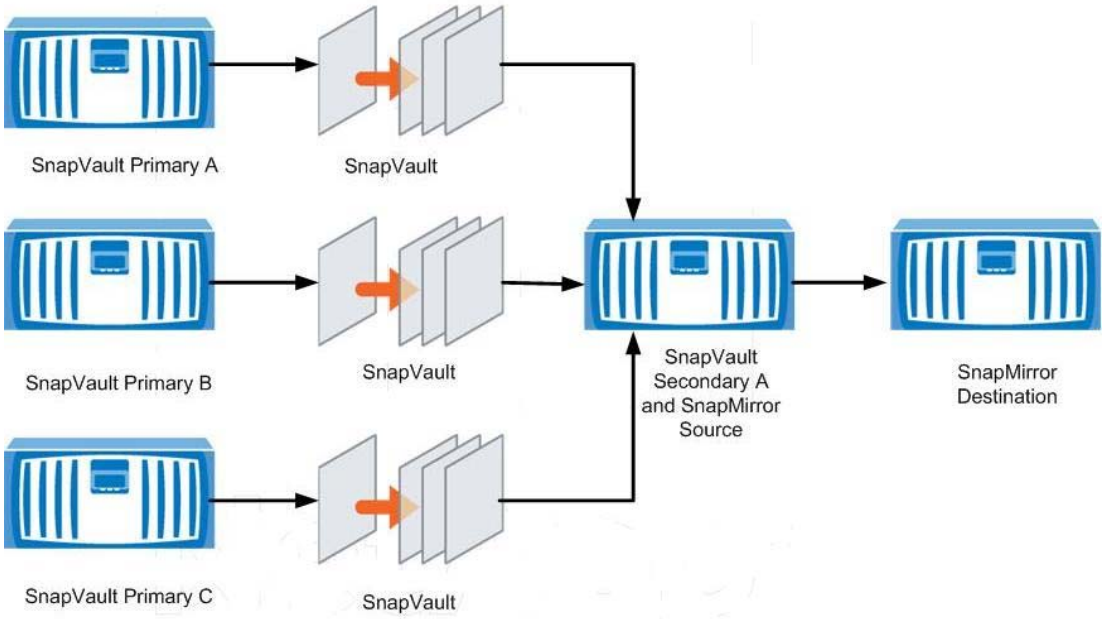

Figure 3) SnapVault Primary and SnapVault Secondary with SnapMirror.

- **SnapVault Primary and SnapVault Secondary with Tape Drive**

  Using this arrangement, you can offload data from the SnapVault Secondary to a tape drive (Figure 4). This provides moving less frequently used data from costly drive to low cost tape drives, and retains the recent data in the SnapVault Secondary if needed for restoration. It can also assist in providing a truly centalized tape back solution.

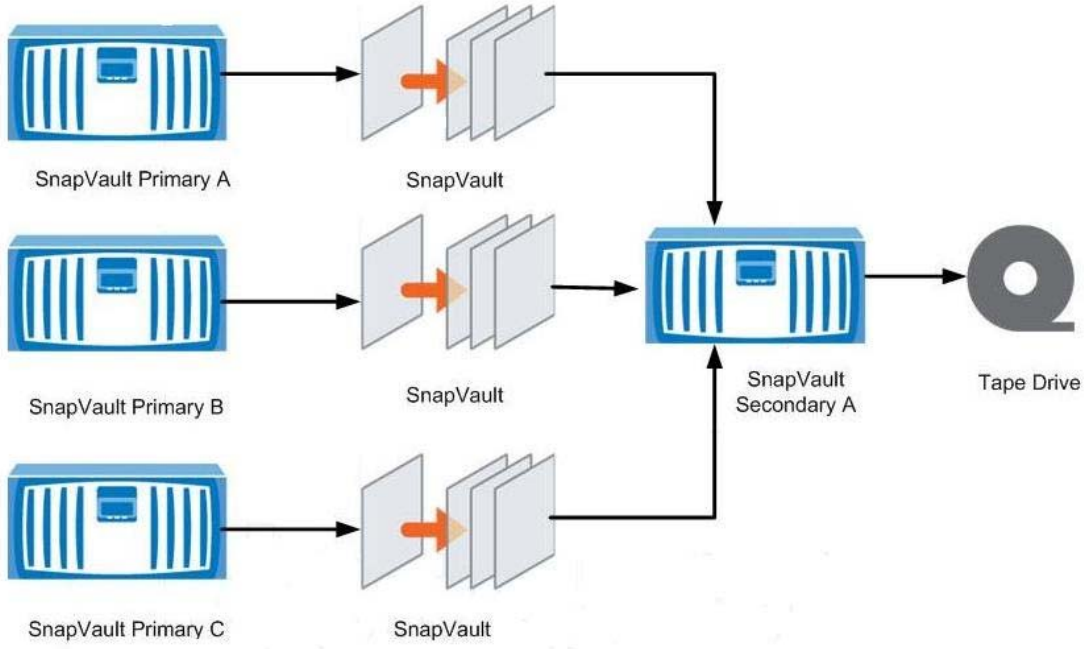

Figure 4) SnapVault Primary and SnapVault Secondary with tape Drive.

- **SnapVault Primary and SnapVault Secondary with Active/Active Cluster Storage**

  Figure 5 depicts the configuration of the SnapVault Primary and SnapVault Secondary on Active/Active cluster storage. On Node1 head you can configure the SnapVault Primary and on Node2 head SnapVault Secondary. If there is cluster failover, single storage Node 1 or 2 can manage both primary and SnapVault Secondary relationship and data backup is achieved. If you have FC drives on Node1 head, you can move the data to SATA drives on Node2 head which is a cheaper storage.

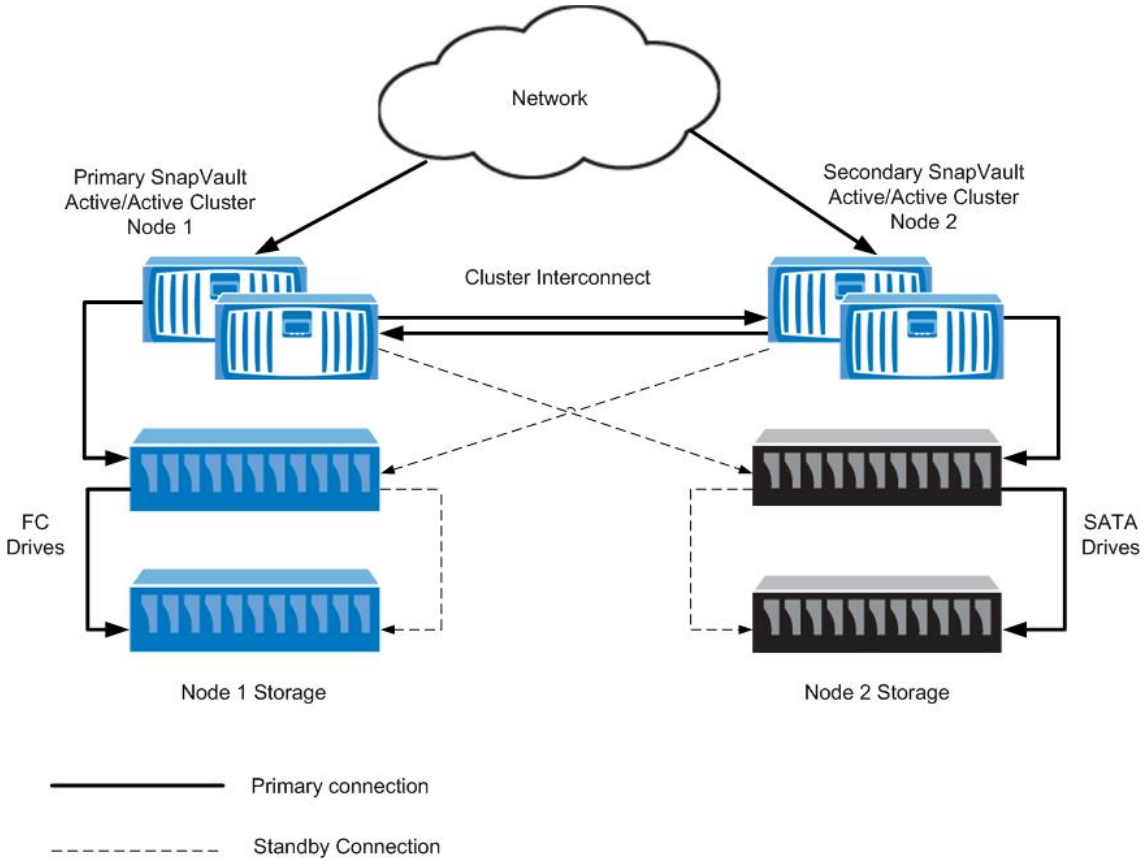

Figure 5) SnapVault Primary and SnapVault Secondary with Active/Active cluster storage.

5.1 SNAPVAULT QTREE, VOLUME, AND LUN LAYOUT

This section describes the various logical configurations that can be designed for backing up SnapVault Primary Snapshot copies to SnapVault Secondary. The following configurations describes, and can be used as a reference during initial configuration for disk-disk backup:

- Qtree to Qtree backup
- Qtree to different Qtree backup
- Volume to qtree backup

QTREE TO QTREE BACKUP

Qtree is the smallest granularity for SnapVault recovery. There can be more than one qtree existing per volume in SnapVault relationsip. You can place block level LUNs (FC or iSCSI) inside qtrees for application data.

Figure 6 depicts the backup scenario where a single `/vol/privault` volume with two qtrees (`qtree_c` and `qtree_d`) on the SnapVault Primary and `/vol/secvault` volume with two qtrees `qtree_c` and `qtree_d` on SnapVault Secondary.

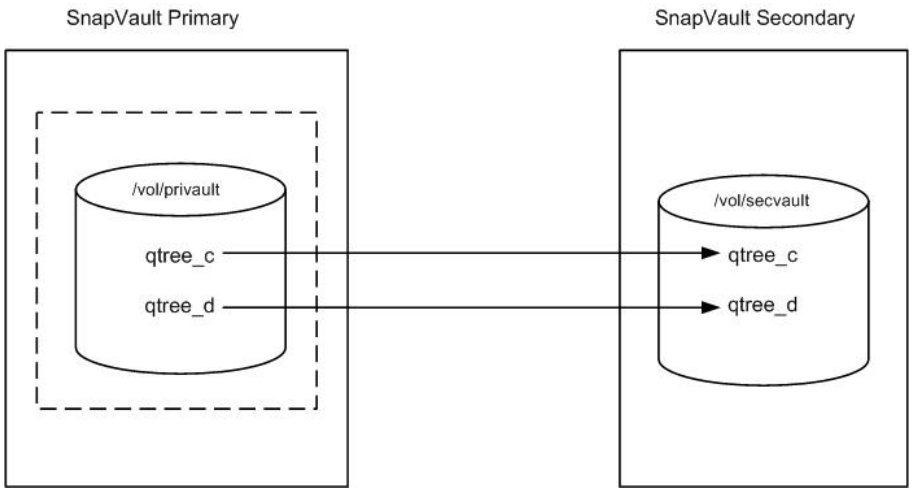

Figure 6) Qtree to Volume qtree backup.

Advantages

- If you have configured qtree-to-qtree backup configuration between two volumes, the restoration operation becomes easier. Example: If `you back up a qtree_c` on SnapVault primary that contains Oracle database and logs that the LUNs created to `qtree_c` on SnapVault secondary. If you need to perform a restore to primary `qtree_c`, you only need to restore LUNs under `qtree_c` and not the entire `/vol/privault` volume. By initially designing you can improve the recovery objective time.
- You can configure a group LUNs of similar applications residing on SnapVault Primary qtree to be backed up to a SnapVault Secondary qtree. In this configuration a storage administrator can know the qtrees in which the applications backed up data resides on the SnapVault Secondary. In addition, troubleshooting becomes easier when narrowing down any errors caused.

Disadvantages

- SnapVault Secondary polls for any new data that is updated on the SnapVault Primary qtrees after initial baseline data is transferred between primary and SnapVault Secondary qtrees which may be triggered by SnapVault Secondary schedule or manually by the user. When all the data is transferred, SnapVault will take volume level Snapshot copy.

QTREE TO DIFFERENT QTREE BACKUP

Using this arrangement, you can have different backup relationships established between primary and secondary qtrees.

Figure 7 depicts a single `/vol/privault` volume, which contains two `qtree_c` and `qtree_d` qtrees which can be placed on two different volumes `/vol/oracle` and `/vol/Sql` on SnapVault Secondary.

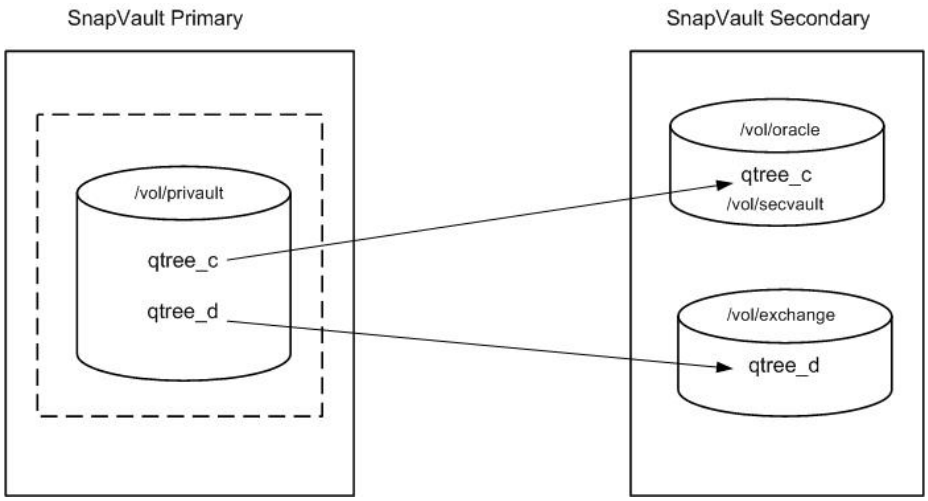

Figure 7) Qtree to different volumes qtree backup.

Advantages

- Space management on SnapVault Secondary becomes easier by separating SnapVault Primary qtrees relationship with different volumes on the SnapVault Secondary volumes. For example, on `qtree_c` if you have multiple Oracle databases LUNs and on `qtree_d` if you have multiple SQL databases LUNs under a single `/vol/privault` volume on primary system, for efficient space and Snapshot management you can create backup qtrees `qtree_c` and `qtree_d` on different volumes `/vol/oracle` and `/vol/Sql` has shown in Figure 7.
- Restoration of data from SnapVault Secondary to SnapVault Primary qtrees improves since your LUNs are evenly disturbed across volumes residing in different aggreates and it makes easier to find data blocks.

VOLUME TO QTREE BACKUP

Using this arrangement, you can back up the entire content of the SnapVault Primary volume like Qtrees, CIFS, and NFS mount points, files and directories and Block access FC or iSCSI LUNs (Figure 8). A single schedule can be established between primary and SnapVault Secondary to back up all the Snapshot copies, which eases management.

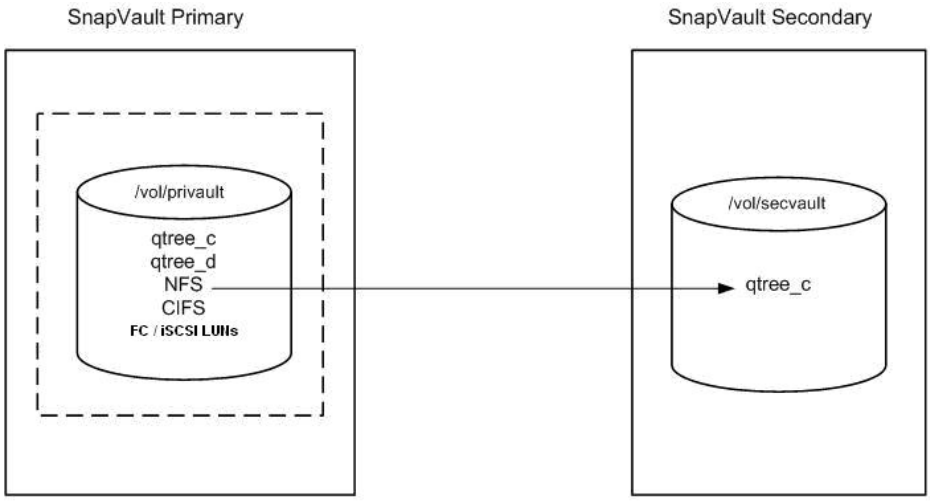

Figure 8) Volume to volume Backup.

Advantages

- SnapVault management between primary and secondary systems is simplified since you can perform a single initial base line transfer of the `/vol/privault` volume to `/vol/secvault` which copies the entire data on the volume.
- On the SnapVault Secondary, a single schedule is created for querying the primary for any updated new data on the `/vol/privault` volume for transferring the data to the `/vol/secvault` secondary volume. This allows lesser Snapshot copies creation and management for restoring purpose.

Disadvantages

- SnapVault will only restore to a qtree on the SnapVault Primary, due to this, all qtree security and quota management will be lost.
- After performing a restore rearranging the data back is a complex task.

Note: It is not recommended to use this type of configuration, since SnapVault will only perform Qtree to Qtree restores and not a directly to a volume.

6 SNAPVAULT BACKUP CONFIGURATIONS

This section describes initial configurations to be performed on the SnapVault Primary and Secondary for performing disk-disk backup. It is assumed user has the knowledge of Data ONTAP fundamentals like creating flex volumes, aggregates, igroup, LUNS, and so on, on the storage using CLI or GUI using the Filer View application.

6.1 SNAPVAULT BACKUP INITIAL DATA GATHERING

Figure 9 depicts a basic SnapVault design which is an example for performing SnapVault configurations on NetApp FAS storage systems. Refer to this figure through out the paper for all the examples during configuring backup or restore options.

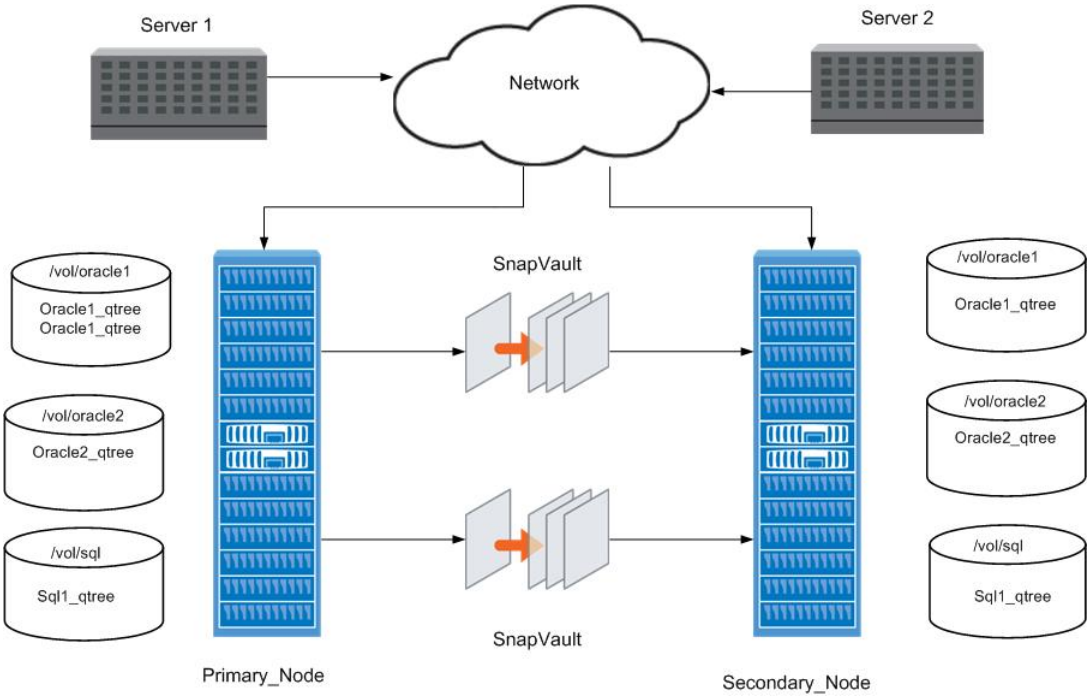

Figure 9) SnapVault configuration on storage and servers nodes.

The following examples assume that you are configuring backups on a non clustered FAS systems named Primary_Node which is configured as SnapVault Primary and Secondary_Node as SnapVault Secondary.

- On Primary_Node, which is a SnapVault Primary, there are three volumes `/vol/oracle1`, `/vol/oracle2` and `/vol/Sql` created.
- On the `/vol/oracle1` and `/vol/oracle2` volumes, there are `Oracle1_qtree` and `Oracle2_qtree` qtrees, and on `/vol/Sql` there is `Sql1_qtree` created on Primary_Node system which participates in backup.
- Two LUNs are created on each of the `Oracle1_qtree`, `Oracle2_qtree` and `Sql1_qtree` qtrees.
- On Secondary_Node, which is a SnapVault Secondary, we need to create only the `/vol/oracle1`, `/vol/oracle2` and `/vol/Sql` volumes which holds the backup Snapshot copies from all qtrees which are defined on the Primary_Node system.
- `Oracle1_qtree`, `Oracle2_qtree` Qtrees and `Sql1_qtree` on the Secondary_Node are created by Data ONTAP during initial SnapVault start command. See Table 2, for more details.
- LUNs under the `Oracle1_qtree` and `Oracle2_qtree` Qtree in Primary_Node are backed up to `Oracle1_qtree` and `Oracle2_qtree` on Secondary_Node system. The LUNs are exposed to Server1 for Oracle applications as documented in the below table.

- LUNs under Qtree `Sql1_qtree` in Primary_Node are backed up to `Sql1_qtree` on Secondary_Node system. The LUNs are exposed to Server2 for SQL application listed in Table 2.

Table 2 lists the logical design which explains SnapVault backup relationships. The below sections are explained keeping the name convention defined in Table 2 and Table 3.

Table 2) SnapVault storage configuration (1).

Storage Names	Roles	Volumes	Qtree
Primary_Node	SnapVault Primary	/vol/oracle1	Oracle1_qtree
		/vol/oracle2	Oracle2_qtree
		/vol/Sql	Sql1_qtree
Secondary_Node	SnapVault Secondary	/vol/oracle1	Oracle1_qtree
		/vol/oracle2	Oracle2_qtree
		/vol/Sql	Sql1_qtree

Table 3) SnapVault storage configuration (2).

Server Names	Roles	LUNs Exposed with Paths
Server 1	Oracle Database	/vol/oracle1/Oracle1_qtree/sales_ora /vol/oracle1/Oracle1_qtree/sales_ora_log /vol/oracle2/Oracle2_qtree/payroll_ora /vol/oracle2/Oracle2_qtree/payroll_ora_log
Server 2	Sql Database	/vol/Sql/Sql1_qtree/sales_mails /vol/Sql/Sql1_qtree/sales_mails_log

6.2 PERFORMING INITIAL CONFIGURATION ON STORAGE

The following commands are to be executed in the Primary_Node and Secondary_Node:

LICENSE SNAPVAULT AND ENABLE ON

```
Primary_Node> license add XXXXXX

Primary_Node>options snapvault.enable on

Primary_Node>options snapvault.access host=Secondary_Node

Secondary_Node> license add XXXXXX

Secondary_Node>options snapvault.enable on

Secondary_Node>options snapvault.access host=Primary_Node
```

Note: The system must be able to resolve the host names to an IP address in the `/etc/hosts` file, else the system needs to be running DNS or NIS.

SCHEDULE SNAPSHOT COPIES ON THE PRIMARY_NODE

To take a restartable consistent Snapshot copy of the LUNs, it is recommended to perform the operation using SnapDrive application installed on the Windows host. SnapDrive is tightly integrated with the operating system. As part of SnapDrive Snapshot copy process, the file system (NTFS) is flushed to the disk and hence the disk image in the Snapshot copy is consistent. It is recommended not to run SnapVault and Snap schedules on the same volumes that are part of SnapVault relation. Snap schedule will fail if SnapVault schedule conflicts, and hence proper planning is necessary when using both together. You will also end up having more number of Snapshots which violate maximum number for a volume that aren't needed.

Before taking Snapshot copies on Oracle and SQL LUNs (defined in Table 2 and Table 3), first put the Oracle and SQL databases into hot backup mode to quiesce I/O for a few seconds. Then, using SnapDrive for Windows either through CLI or GUI, perform a Snapshot on the LUNs. This provides a restartable backup copy. SnapVault is also integrated with SnapManager for Windows to perform the above tasks.

Assume we have created two volumes namely /vol/oracle1 and /vol/Sql with four LUNs exposed to Server1 and two LUNs to Server2 carved out from /vol/oracle1 and /vol/Sql volumes as shown in Table 2 and Table 3. Before scheduling Snapshot copies in the Server1 and Server2, the following observations must be considered when defining schedule policy:

- SnapVault performs disk-to-disk backup by identifying the changed blocks from last backup Snapshot copy. The initial baseline data transfer is performed by executing the SnapVault start command. This transfers the entire data in the LUNs from SnapVault Primary to the SnapVault Secondary. Once data is transferred completely a baseline Snapshot is taken in the SnapVault Primary and SnapVault Secondary.
- For the next update, based on the SnapVault schedule on SnapVault Secondary, it initates the transfer of changed data blocks on SnapVault Primary. If a new Snapshot copy exists in SnapVault Primary's volume, it compares it against the last Snapshot (the baseline Snapshot), produces all the changed data blocks, and transfers it to SnapVault Secondary. Once data is transferred, a Snapshot of the SnapVault Secondary volume is taken.
- To automate the SnapVault backup process, we can use the SnapVault schedule within Data ONTAP to take Snapshot copies of the LUNs every hour, nightly or weekly. It also retains Snapshot copies of interest which provides better Recovery Point Objective (RPO) and Recovery Time Objective (RTO) when disaster strikes. The drawback of using the Data ONTAP SnapVault schedule to take Snapshot copies, is they are not consistent because it is not integrated with applications. For example, it will not freeze the I/O during Snapshot creation. Due to this, when a restore is performed you will not end up with a restartable copy. To overcome this problem, perform Snapshot management by using SnapDrive which integrates with applications to take consistent Snapshot copies. Windows scheduler can be called to perform the scheduling.
- A Snapshot copy is the basic unit of transfer and is taken on the volume level. It is very important to group similar qtrees with the same characteristic into the same volume on the Primary and Secondary, and also to decide scheduling policy. You must carefully plan how many Snapshots are required to be taken and retained on the Primary and Secondary_Node, otherwise the space management and maximum number of Snapshot copies per volume bases will be violated. This can be explained with the below scenario where we have used the naming conventions as defined in the Section 5.1.

Scenario based example

Oracle_qtree1 holds `sales_ora` and `sales_ora_log` LUNS and `Oracle_qtree2` holds `payroll_ora` and `payroll_ora_log` LUNs on the SnapVault Primary.

It is observed using the **`-l`** SnapVault status command, the `Last Transfer Size` on `sales_ora` is 10GB and on `payroll_ora` the last transfer size is 500MB.

If both qtrees are backed up to the same volume on the Secondary_Node, then there are higher chances of 500MB changed data transfer arriving faster and 10GB changed data arriving slower.

Both transfers are to the same Secondary_Node volume. SnapVault will not take a Snapshot unless all data is transferred. This condition is known as Quiescing with slow transfer.

To overcome this problem, you can group all the like LUNs in the qtree with similar transfer data size on a separate qtree on the Secondary_Node.

Snapshot is always taken at the volume level. It consumes space if you have space reserved on your LUNs as shown in the Figure 9 in Section 6 where Primary_Node `Oracle1_qtree` is backed to Secondary_Node `Oracle1_qtree` in `/vol/oracle1` volume and Primary_Node `Oracle2_qtree` on volume `/vol/oracle2` is backed to different qtree `Oracle2_qtree` on `/vol/Oracle2` volume on Secondary_Node.

All Windows systems have a in-built scheduler that can be used to schedule regular jobs.

An example of a Perl script that schedules Snapshots copies to occur at hourly, nightly and weekly intervals is described below.

To execute this script, install the active Perl for Windows which is freely availble for download. Copy the entire script to a notepad and save with `.pl` extension.

Note: If you place the `.pl` in a directory other than `c:\perl\bin` set the path under System Properties. To do this, navigate to the Advanced tab, click Environment Variables, and edit the path file to point to `c:\perl\bin`.

The following is a sample script and not supported by NetApp. This script provides logic for automating Snap shot copies with Snap drive for providing disk to disk backup through Snap Vault. Before you run the script you need to put application in hot backup mode and flush all the data in the file system cache to disk, which takes consistent Snapshot copies of the LUNs exposed to the application.

```
#This perl script takes hourly/nightly/weekly snap shot copies using SDW based
on the retention value.

# This perl script makes hourly / nightly / weekly Snapshots copies using Snap
Drive for Windows

# First parameter – Snapshot name

# Second parameter – Mount Point or Drive Letter on which Snapshot as to be
taken

# Third parameter – Number of Snapshots to retain

use strict;

use Getopt::Long;

my (@snap_shots);

my $extn = 0;

my $tmp_file = "temp_file.$$";

&parse_input();

#Process command line arguments from the batch file

sub parse_input{

    my ($Snapshot_name, $mountpoint_list, $retension_value);

GetOptions('s=s'=>\$Snapshot_name,'d=s'=>\$mountpoint_list,'n=i'=>\$retension_va
lue);

 if ( defined ($Snapshot_name && $mountpoint_list)){

        &snap_create($Snapshot_name, $mountpoint_list, $retension_value);

    } else {

        print "Invalid arguments:\n";

        &usage();

    }

}

#Usage infomation

sub usage {

    print "usage: snp_archive.pl -s <Snapshot_name> -d <mountpoint_list> -n
<retension value>\n";

}

#To create snap shot

sub snap_create{

 my ($sname, $mpoint, $rvalue) = @_;
```

```
    my $max = $$;
    my $tmp_name = $sname.'.'.$extn;
 @snap_shots = &get_snaplist($sname, $mpoint);
    my $sn = @snap_shots;
 if (grep(/$sname\.0/,@snap_shots)){
        for (my $i = $sn; $i > 0; $i--) {
            my $cmd = "sdcli snap rename -d $mpoint -o $snap_shots[$i-1] -n
$sname.$i";
            system("$cmd");
        }
my $cmd1 = "sdcli snap create -s $tmp_name -D $mpoint";
        system("$cmd1");
    } else {
        my $cmd = "sdcli snap create -s $tmp_name -D $mpoint";
        system("$cmd");
    }
@snap_shots = &get_snaplist($sname, $mpoint);
    my $sn = @snap_shots;
  #delete snap shot if it is exeeding the retension value
    if ($sn > $rvalue){
 my @items = splice(@snap_shots,$rvalue);
        foreach my $file (@items) {
            my $cmd ="sdcli snap delete -d $mpoint -s $file";
            system("$cmd");
        }
    }
}
#To get snap shot list from storage
sub get_snaplist {
    my ($snap_name, $mount_point) = @_;
    my @snap_list;
    my $cmd = "sdcli snap list -d $mount_point";
    system("$cmd 1>$tmp_file 2>&1");

    open(IN,"$tmp_file") || die("Unable open file:$!\n");
    my @lines = <IN>;

    foreach my $tmp (@lines) {
```

```
        if($tmp =~ m/($snap_name\.\d)/g){
            push(@snap_list,$1);
        }
    }
    close(IN);
    unlink($tmp_file) or die "Can't unlink $tmp_file: $!";
    return (@snap_list);
}
```

The script detailed below, is a `bat` file where the following three parameters must be supplied for the Perl script to automate the entire Snapshot management:

- Snapshot copy name
- File system mount point name or drive letter
- Number of copies to be retained on Primary storage

Copy the `bat` file contents to a notepad and name it with a `.bat` extension. This `bat` file will be later used in the Windows Task Scheduler for automating purpose. Place this `bat` file and the Perl file in the same directory.

The name of each Snapshot copy is appended with a number subscript, starting from 0 (0 is the latest). This scheme fits well with the Snapshot naming convention used by SnapVault. Using this naming convention, you can transfer these copies to the SnapVault secondary storage system automatically.

```
@cd C:\Perl\bin

@c:\perl\bin\perl.exe c:\perl\bin\snap_record.pl -s Snapshot_name -d mount point
/ drive letter -n number of Snapshot copies to retain
```

Based on the parameters defined, the Perl script defined above executes according to the following logic:

1. `-s` Snapshot name: `sv_hourly`
2. `-d` mount point or drive letter: `e:`
3. `-n` number of Snapshots copies to retain in the SnapVault primary storage:`12`

Creating a Basic Task using Windows Task Scheduler

The following procedure describes how to add the above `bat` file to a Windows 2008 Server Task Scheduler where Snap Drive 5.0 or above and Active Perl software is installed and `.pl` script and `bat` file is placed inside Perl installed directory.

This procedure describes how to create schedules for running the `bat` file every hour. Similarly you can schedule a different `bat` file to run once a day or once a week based on the scheduling process you have defined in the SnapVault Primary and Secondary.

In this example, hourly Snapshot copies named `sv_hourly.x` and latest 20 copies retained on the SnapVault Primary (named `sv_hourly.0` through `sv_hourly.11`) are created. Similarly, you can have a different `bat` file created with different sets of parameters.

1. Open Task Scheduler.
2. Right-click the Microsoft folder and select Create Basic Task (Figure 10).

   The Create Basic Task Wizard appears (Figure 11).

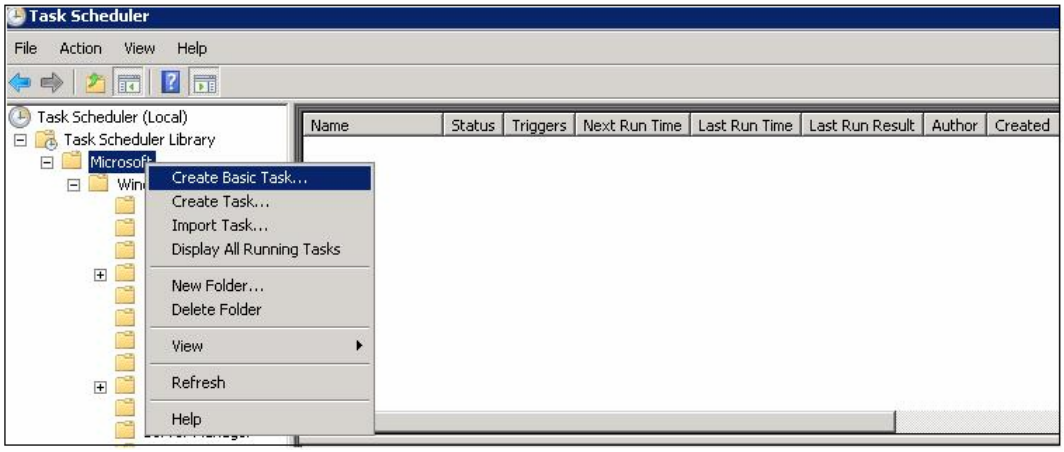

Figure 10) Create Basic Task.

3. In the Create Basic Task Wizard, enter the name and description of task and click Next (Figure 11).

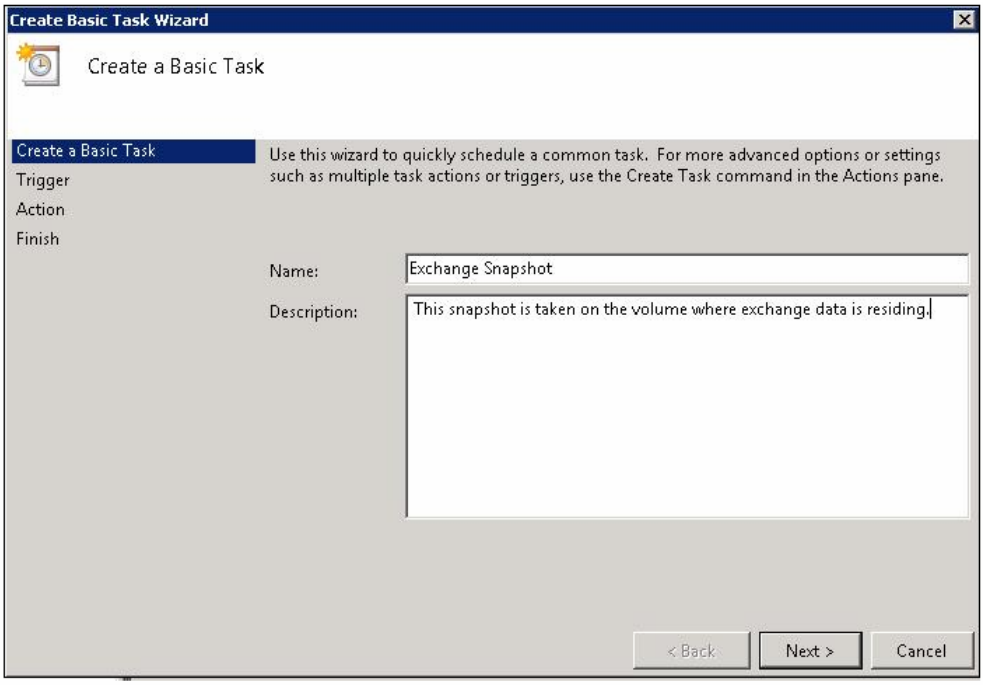

Figure 11) Create Basic Task-Set Name and Description.

4. In Task Trigger Options, select when you want the task to start, and click Next (Figure 12).

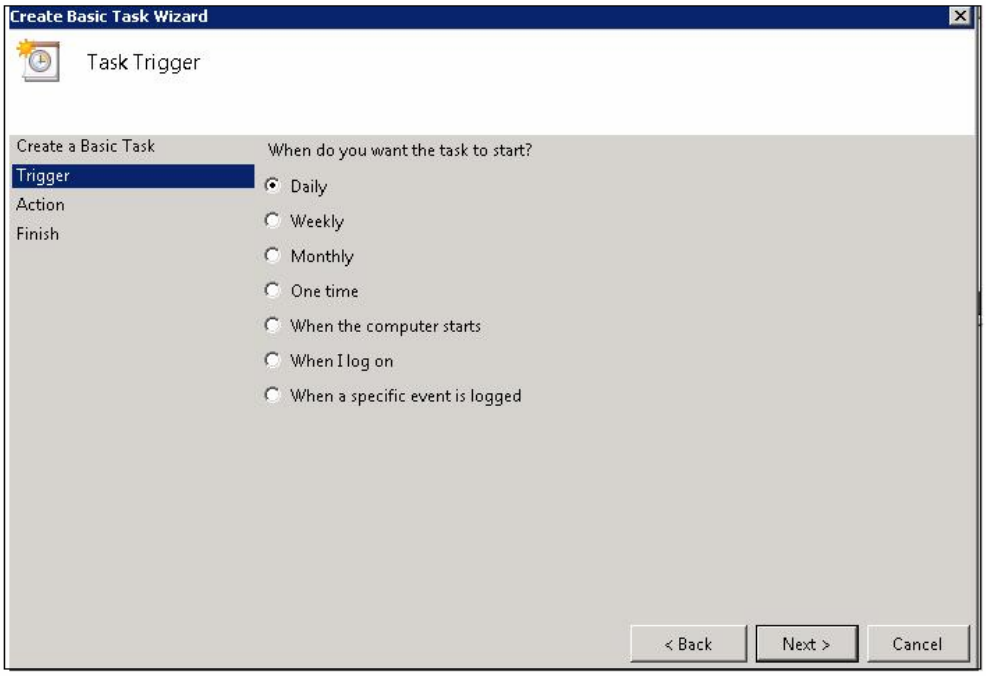

Figure 12) Create Basic Task-Set Schedule.

5. In the Action page, select Start a program and click Next (Figure 13).

   The Start a Program page appears (Figure 14).

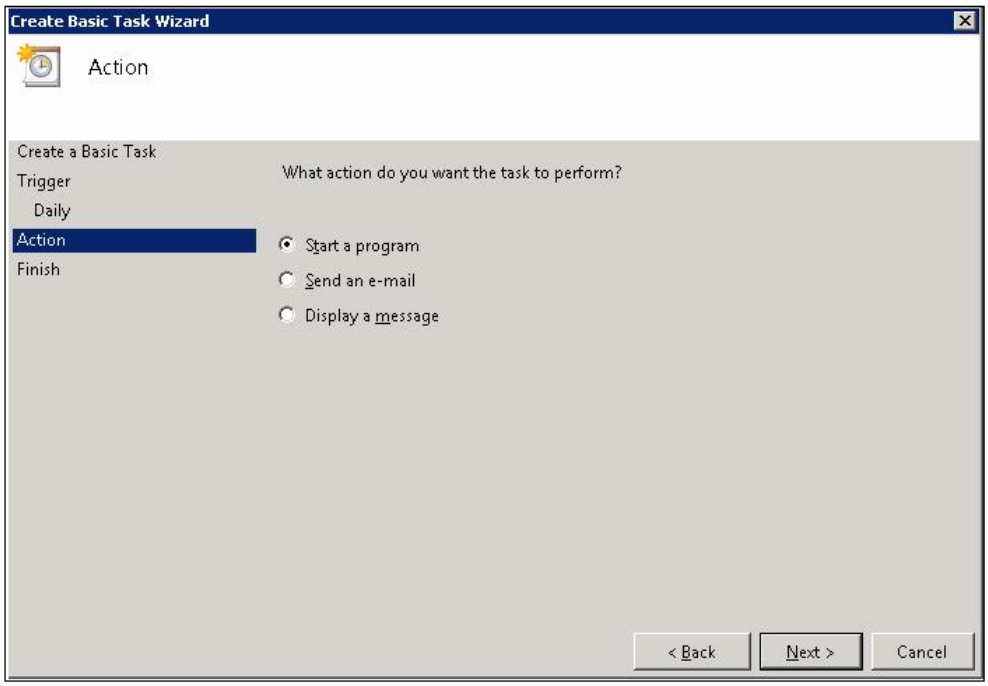

Figure 13) Create Basic Task- Set Action.

6. In the Open window, on the Start a Program page, navigate to the directory where you have stored the bat file (for example, `config.bat`), and select it (Figure 14).

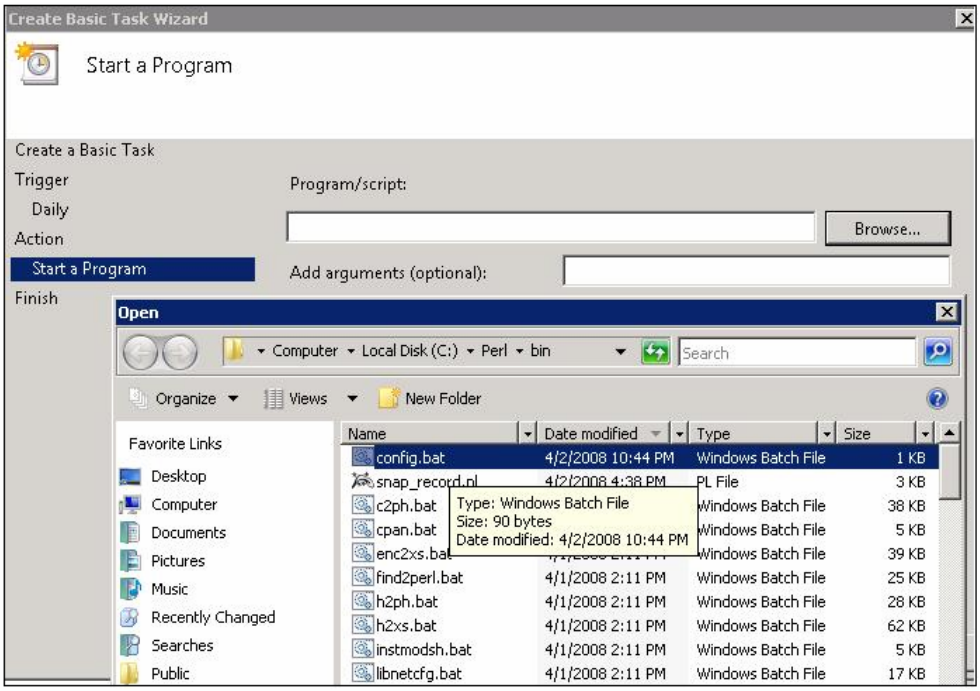

Figure 14) Create Basic Task- Select the Program.

7. In the Summary page, review the settings for the task created and click Finish (Figure 15).

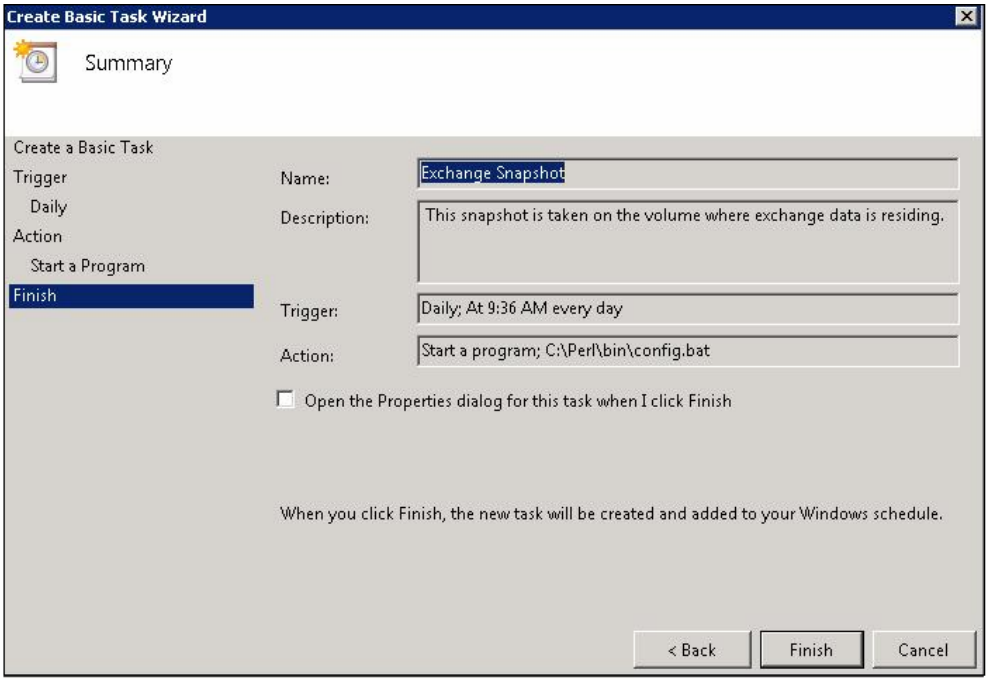

Figure 15) Create Basic Task-Summary.

8. To set duration to repeat the tasks, do the following:
   a) Right-click the schedule created and click Properties.
   b) Click the Triggers tab, and click Edit.

      The Edit Trigger page appears (Figure 16).

c) In the Edit Trigger page, select the Repeat task every checkbox.

d) Set the time duration to repeat the task.

e) Click Ok.

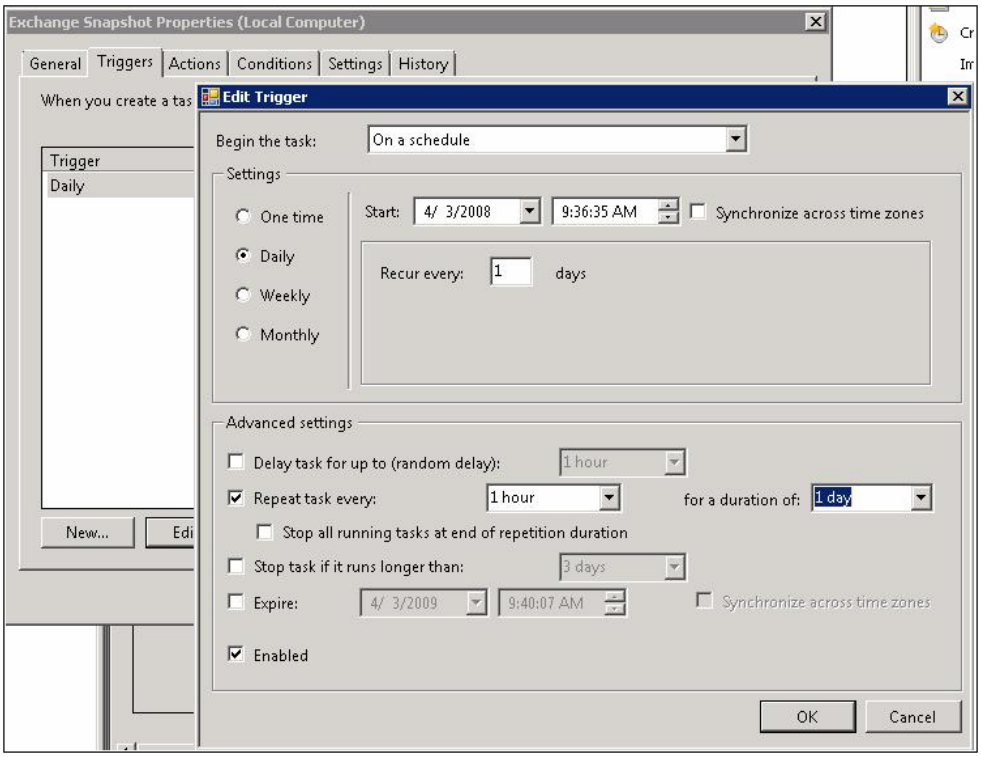

Figure 16) Create Basic Task- Edit Trigger.

SCHEDULE SNAPSHOT UPDATE USING SNAPDRIVE

You can control SnapVault update through Snap Drive 5.0 and higher from the host without having a SnapVault schedule configured on the Secondary_Node storage, however Snapshot space management activites for the storage administrator will be increased. This is because you cannot define a retention period and you must manually monitor the volume space and number of Snapshot copies to be retained on the Secondary_Node SnapVault secondary. Or you can also manually script the whole logic which is defined in the above Perl script for Primary_Node SnapVault primary.

Scenario based example

If you need to update `sv_hourly.3` Snapshot which is taken on `sales_mails` and `sales_mails_log` on `/vol/Sql/Sql1_qtree` on the Primary_Node to be SnapVault update to the Secondary_Node immediately due to power outage or if you sense any physical or logical storage problem then you can run the following command on server 2.

```
C:\ sdcli snapvault archive  -a sv_hourly.3 -Ds f:sv_hourly.3
```

This command will create a `sv_hourly.3` Snapshot on the Secondary_Node storage. Ensure that the `sv_hourly.3` Snapshot does not already exist since SnapDrive will create it.

Table 4 and Table 5 define the different set of policies that can be considered as references for designing schedule policies. By proper planning retention policy, you can provide better RPO and RTO.

Table 4) Hourly Policy

Snapshot Name	Hourly Policy (24 hours clock)	Retention Policy
sv_hourly	• 0 – 23 • 8 - 18 • 8 - 20	• 23 [24 hrs] • 12 [12 hrs] • 6 [6 hrs]

Table 5) Nightly and Weekly Policy

Snapshot Name	Nightly Policy (24 hours clock)	Days Policy	Retention Policy
• sv_nightly	• 23 • 20 • 18	• Mon- Fri • Mon – Sun	• 6[1 weeks] • 12 [2 weeks] • 30 [1 month] • 60 [2 months]
• sv_weekly	• 23 • 20 • 18	• Sat • Fri	• 6[1 weeks] • 12 [2 weeks] • 30 [1 month] • 60 [2 months]

Scenario based example

The policies schedules listed in Table 4 and Table 5 as described as follows:

- `sv_hourly` with hourly policy set to 0 – 23 takes Snapshot copies on the volume every hour from morning 1am to 11 pm and the retention policy set to 23 which will hold 23 Snapshots on the storage.
- `sv_nightly` with nightly policy set at 20 takes Snapshot copies at every day at 8 pm and retention policy set to 6 holds the last one week's backup.
- `sv_weekly` with weekly policy set to on 23 takes Snapshot copies at 11 pm every Monday-Sunday and retention policy set to 30 holds last 1 month Snapshots data.

SCHEDULE SNAPSHOTS ON THE SECONDARY_NODE

Before creating SnapVault schedule on the Secondary_Node, it is assumed you have created flex volume and planned all qtrees naming conventions that are needed to be created for backing up datasets from Primary_Node to Secondary_Node. The naming conventions used in this section are defined in Section 5.1.

The SnapVault schedule checks for the specified Snapshot name on the primary qtree for updates based on the schedule created on the SnapVault Secondary. If it finds any new Snapshot copy, it transfers the changed data blocks to destination volume on Secondary_Node and then creates a Snapshot copy.

Before setting up SnapVault schedule, consider the following:

- When setting up SnapVault schedule, it is recommended to disable all the local snap schedules. It is mandatory to have the same Snapshot naming convention on both the SnapVault Primary and SnapVault Secondary, else data transfer will not occur. Following the same naming convention also enables the storage administrator at the time of restoration to recognise which Snapshots belong to which qtrees and are resding on what volumes.
- When defining the SnapVault Snapshot retention period during scheduling, always retain few Snapshots copies to be locally available at the SnapVault Primary (defined by Snap Drive schedule policy). This allows to efficiently manage space on SnapVault Secondary. You can perform a local restore, which provides better turn around restoration time.

Scenario based example

- There are three schedules for hourly, nightly and weekly for `oracle1`, `oracle2` and `Sql` volumes on the Secondary_Node. You can decide on schedule polices which best suits your requirement. You can refer to Table 4 and Table 5 which defines various policies.

Commands to achieve SnapVault schedule are as follows.

Hourly SnapVault schedule

```
Secondary_Node> snapvault snap sched -x oracle1 sv_hourly 23@mon-sun@0-23

Secondary_Node > snapvault snap sched -x oracle2 sv_hourly 23@mon-sun@0-23

Secondary_Node > snapvault snap sched -x Sql sv_hourly 23@mon-sun@0-23
```

Nightly SnapVault schedule

```
Secondary_Node >snapvault snap sched -x oracle1 sv_nightly 6@mon-sun@20

Secondary_Node >snapvault snap sched -x oracle2 sv_nightly 6@mon-sun@20

Secondary_Node >snapvault snap sched -x Sql sv_nightly 6@mon-sun@20
```

Weekly SnapVault schedule

```
Secondary_Node >snapvault snap sched -x oracle1 sv_weekly 6@sat@20

Secondary_Node >snapvault snap sched -x oracle2 sv_weekly 6@sat@20

Secondary_Node >snapvault snap sched -x Sql sv_weekly 6@sat@20
```

The following are the Important considerations when performing scheduled backup:

- The time set on servers and storages systems (Server 1,Server 2 and Primary_Node , Secondary_Node) should be synchronized, so that the schedules match correctly. If the systems are in different time zones, the schedule must be adjusted accordingly.
- The Snapshot copy names should be identical, because the SnapVault secondary storage system looks for these names during the transfer.
- Snapshot copies that are scheduled on the secondary storage system with the **`snapvault snap sched -x`** command are created five minutes after the hour that you specify. This delay is usually sufficient to give the script that is running on the Windows host enough time to create Snapshot copies before the secondary storage system is updated from them.
- In this example, more Snapshot copies are retained on the secondary system, because typically it has a larger storage space.
- SnapVault transfer status on the secondary storage system can be monitored by using the **`snapvault status`** command.

SNAPVAULT INITIAL BASELINE TRANSFER

After you have decided the logical design, configure schedules on both Primary_Node and Secondary_Node systems, and check that SnapVault is enabled and running. Before SnapVault starts transferring incremental data to Secondary_Node qtrees, you must perform an initial baseline transfer of the LUNs residing on Primary_Node qtrees which provides information on where to store the backed Snapshot copies on the Secondary_Node. At this time on the Primary_Node SnapDrive will be taking Snapshot copies, but no data will be transferred to the Secondary_Node.

Before setting up initial baseline transfer on the Secondary_Node, you must size volumes for backing up data and also plan bandwidth consumption for each Snapshot copy transfer.

- When managing the space on the secondary volume where backup data is stored on qtrees it is important to decide on how many Snapshot copies should be maintained and also estimate rate of data change for each qtree residing on the volume. This will provide you a sizing figure to plan for space on the volume.
- You can use **`snap delta`** command which reports the rate of change between Snapshot copies. The command compares all copies in a volume, or just the copies specified.

Scenario based example

Consider `sales_ora` and `payroll_ora` LUNs of 500GB size configured on the Primary_Node `Oracle1_qtree` and `Oracle2_qtree` qtrees on `/vol/oracle1` volume of 3TB.

The default setting of volume `/vol/oracle1` will reserve 20% for SnapReserve and if you create two LUNs of 500GB each with default space reserve option will occupy 2TB space on the volume with available space 500GB left.

If the rate of data on `sales_ora` LUN changes 100% and on the `payroll_ora` LUN only 10% changed at this stage a Snapshot named `sv_hourly.0` is taken on the `/vol/oracle1` volume.

After certain timeline the `sales_ora` LUN overwrites by 100%. At this point available space on the `/vol/oracle1` will be less than 500GB due to which Data ONTAP does not allow you take any new Snapshot copies on `payroll_ora` or `sales_ora` LUNs even if `sales_ora` LUN changed only 10% due to in-advent space.

Hence placing the LUNs in two different volumes will better utilize the Snapshots space on the Primary_Node and Secondary_Node volumes as described in Section 5.1 where like characteristic LUNs are placed in two different volumes which is shown in Section 6.1 under Figure 9 and Table 4 and Table 5. It is recommended to place single qtree in a single volume.

If you have a bandwidth of 1GBps which is 100MBps. The **`-k`** options provides the threshold value which can be set on individually on each SnapVault relationship. These settings are applicable during initial baseline transfer as well as subsequent update transfers from Primary_Node volumes. If the SnapVault settings are changed in the middle of transfers, the settings will only be applied during the next update. Hence, the network bandwidth can be better utilized.

Scenario based example

A SnapVault transfer triggered by schedules set on `Oracle1_qtree` and `Oracle2_qtree` residing on same Oracle volume on the primary Node at the same time to the same volume on the Secondary_Node which has network bandwidth of 100MBps.

On SnapVault, start configuration **`-k`** option for `Oracle1_qtree` is set to 80MB since it holds larger capacity LUNs and rate of data changed for subsequent updates is higher. On `Oracle2_qtree` the **`-k`** option is set to 20MB since it holds smaller capacity LUNs and so rate of data changed is lesser.

Since `Oracle1_qtree` transfers more data it is advisable to have higher threshold limit set which in this case 80MB and `Oracle2_qtree` transfers less changed data blocks it is set with 20MB pipe for a given 100MB link.

This allows for a better bandwidth management and the Secondary_Node Volume also takes the Snapshot copies without any delay which overcomes the quiescent state.

If there is a delay in Snapshot update or during the initial baseline copy with **`-t`** option value and specify the number of retry counts before failing the transfers to Primary_Node volumes.

Scenario based example

If there is a network outage or delay during subsequent updates to `Oracle1_qtree` by default SnapVault will try twice and then fail. You can override this count by specifying **`-t`** options with 4 where it tries four times before deciding to fail the transfer.

Note: You can allow **`-k,`** and **`-t`** options to default setting which allows SnapVault to manage bandwidth threshold and retry transfers to the maximum on the systems.

Use the following commands to perform initial baseline transfer:

```
Secondary_Node> snapvault start -k 80000 -t 4 -S Primary_Node:
/vol/oracle1/Oracle1_qtree Secondary_Node: /vol/oracle/Oracle1_qtree

Secondary_Node> snapvault start -k 20000 -t 4 -S Primary_Node:
/vol/oracle2/Oracle2_qtree Secondary_Node: /vol/oracle2/Oracle2_qtree

Secondary_Node> snapvault start -k 20000 -t 4 –S Primary_Node:
/vol/Sql/Sql1_qtree Secondary_Node: /vol/Sql/Sql1_qtree
```

No other steps are needed for here since all the scheduling is already configured on both Primary_Node and Secondary_Node volumes.

7 SNAPDRIVE CONNECT AND SNAPVAULT RESTORE OPTIONS

SnapDrive currently supports creating local backups of Snapshot copies and performing local restore only. It cannot perform SnapVault restores from Secondary_Node. Also it is not possible to add any policy, such as Backup and then mirror, or only Backup, and so on.

To perform end-to-end backup and restore functionality, SnapDrive has to integrate with SnapManager and Protection Manager to work on the datasets created by SnapDrive by assigning a policy (Backup to SnapVault Secondary system). It also provides remote restore in accordance to policy settings configured by the storage administrator.

The following methods can be used to connect and restore the backed up data from Secondary_Node system to server:

- Connecting Data from SnapVault Secondary System using SnapDrive
- Restoring Data from SnapVault Secondary System
- SnapDrive Data Restore from SnapVault Primary System
- The naming conventions used in the examples are described in Section 5.1. To simplify the scenarios explanation, a single qtree is used for all scenarios.

7.1 CONNECTING DATA FROM SNAPVAULT SECONDARY SYSTEM USING SNAPDRIVE

In this scenario, there is network connectivity between Server1 (which plays an Oracle role and has SnapDrive 5.0 higher version running) and Secondary_Node (which has Data ONTAP 7.2 or higher running). You can perform two kinds of granularity recovery- one at LUN level, and other at the entire Volume level.

This does not perform a complete restoration of backup data. However, if you need to partially recover only some part of data, perform the following methods:

- LUN Level Restoration
- Volume Level Restoration

LUN LEVEL RESTORATION

If you are certain about the data corruption on the LUNS and the time at which this happaned, it is advantageous to recover only from those LUNs instead of restoring the full volumes to the Primary_Node. This reduces time, and also the network bandwidth. The commands to perform this task are described below.

Scenario based example

- If the partial data corruption on the oracle database `sales_ora` where an `sv_hourly.2` Snapshot copy was taken and is required:

1. You can perform a LUN clone of `sales_ora` to a new qtree, and not to the same qtree where the parent LUN resides by the following command:

   ```
   Secondary_Node> lun clone create /vol/oracle1/backup_sales_ora -b
   /vol/oracle1/oracle1_qtree/sales_ora sv_hourly.2
   ```

2. After this operation you can go the Server1 and connect using a GUI or the following CLI command:

   ```
   C:\ sdcli disk connect -p btc-ppe-filer173:/vol/oracle1/backup_sales_ora -d
   q: -s sv_hourly.2 -IG iscsi viaRPC.iqn.1991-05.com.microsoft:iscsi -dtype
   dedicated
   ```

VOLUME LEVEL RESTORATION

Volume cloning allows you to perform a recovery of the LUNs residing in the qtrees on the volume, to a specific time when the Snapshot copywas taken.

Scenario based example

If there is a virus attack on the Oracle or SQL databases residing on all the LUNs in a volume which as partially corrupted databases. To recover the databases spread on all the LUNs, it is advantageous to clone the entire volume to recover partial data across the LUNs.

All the LUNs are residing on volume oracle1 which contains one qtree with the two `sales_ora` and `sales_ora_log` LUNs to different volume `backup_oracle1` with the Snapshot name `sv_hourly.3`.

To perform volume cloning and connecting to the Server1 through the SnapDrive, do as follows:

1. Perform a vol clone by typing:

   ```
   Secondary_Node> vol clone create backup_oracle1 -s none -b oracle1
   sv_hourly.3
   ```

2. After volume cloning, all the LUNs will be offline. Bring the LUNs online by typing:

   ```
   Secondary_Node> lun online /vol/backup_oracle1/oracle1_qtree/sales_ora

   Secondary_Node> lun online /vol/backup_oracle1/oracle1_qtree/sales_ora_log
   ```

3. After this operation, you can go the Server1 and connect to both LUNs using a GUI or CLI.

4. The following SnapDrive command mounts the LUNs `sales_ora` and `sales_ora_log` to drive letter `p:` and `q:` on the Server1. If you mount to the same server where original LUNs are mounted, it is advised to mount with different drive letters.

   ```
   C:\ sdcli disk connect -p Secondary_Node:/vol/backup_oracle1/sales_ora -d p:
   -s sv_hourly.2 -IG iscsi viaRPC.iqn.1991-05.com.microsoft:iscsi -dtype
   dedicated

   C:\ sdcli disk connect -p Secondary_Node:/vol/backup_oracle1/sales_ora_log -
   d q: -s sv_hourly.2 -IG iscsi viaRPC.iqn.1991-05.com.microsoft:iscsi -dtype
   dedicated
   ```

   Note: Since you will be accessing the clone LUN across the WAN link which resides on the SnapVault Secondary, the performance may degrade. To permanently reuse cloned LUN data, it is better to perform a SnapVault restore to SnapVault Primary and then connect using SnapDrive on Windows which will be locally accesed. To perform SnapVault restore to Primary_Node System follow the instructions specified in Section 7.3.

DISCONNECTING AND DESTROYING THE CLONE VOLUME

1. After the recovery operation is completed, disconnect the LUNs from Server1 using SnapDrive. The following commands should be run in Server1. Server Side:

   ```
   C:\>sdcli disk disconnect -d p:

   C:\>sdcli disk disconnect -d q:
   ```

2. Once disconnected you can destroy the `backup_oracle1` FlexClone volume on the storage by typing:

   ```
   Secondary_Node> vol offline backup_oracle1

   Secondary_Node> vol destroy backup_oracle1
   ```

3. To destroy the cloned LUN copy on secondary storage, type:

   ```
   Secondary_Node> lun destroy -f /vol/oracle1/backup_sales_ora
   ```

Note: If you have enabled RSH on the storage, you can run all the above commands from Server1.

7.2 RESTORING DATA FROM SNAPVAULT SECONDARY SYSTEM

If there is no network connection from the Server1 to the Secondary_Node system, you can always restore the backup to Primary_Node system which can then be accessed by the Server1 for partial or full recovery. Before performing the restore, check if the required Snapshot is not retained in the Primary_Node. If it exists, then you can perform a local SnapRestore. Restoring from the Primary_Node system should be

considered only when the required Snapshot is not found in the Primary_Node which is decided by retention policy configured on the storage which is explained in the Section 5. SnapVault always restores data to a qtree only on SnapVault Primary and also you cannot restore to the same qtree path.

Scenario based example

This scenario describes how to restore `Oracle2_qtree` qtree on `oracle2` volume residing on Secondary_Node system to Primary_Node system to a different qtree path.

- Use the following commands on the Primary_Node storage system to restore data to a new qtree `backup_oracle2_qtree` to the same parent volume or to different volume. In this example we will restore to the same parent volume.
- `Backup_oracle2_qtree` should not be created manually or it should exist in the Secondary _Node. SnapVault restore operation will create the required `sv_hourly.25` Snapshot.

The following command runs only on primary storage system. SnapVault creates a new `Backup_oracle2_qtree` qtree on the Oracle volume, and transfers the backup data from Secondary_Node system.

```
Primary_Node>snapvault restore -s sv_hourly.25 –S
Secondary_Node:/vol/oracle2/oracle2_qtree Primary_Node:/vol/oracle/Backup_oracle2_qtree
```

When the transfer is completed, you can use SnapDrive to connect to the LUNs which are restored under the `Backup_oracle2_qtree`.

```
C:\ sdcli disk connect –p
Primary_Node:/vol/oracle2/Backup_oracle2_qtree/payroll_ora -d f: -s sv_hourly.2
-IG iscsi viaRPC.iqn.1991-05.com.microsoft:iscsi -dtype dedicated
```

```
C:\ sdcli disk connect -p
Primary_Node:/vol/oracle2/Backip_oracle2_qtree/payroll_ora_log -d g: -s
sv_hourly.2 -IG iscsi viaRPC.iqn.1991-05.com.microsoft:iscsi -dtype dedicated
```

Note: If you do not have space in the storage to perform the restore to a different volume, you can disconnect all the LUNs, destroy the qtree, and then perform a restore with the same qtree name and then use SnapDrive to connect the restored LUNs.

7.3 SNAPDRIVE DATA RESTORE FROM SNAPVAULT PRIMARY SYSTEM

If the required Snapshot copy is available on the Primary_Node system, it is faster to restore this data by performing a local SnapRestore, since it does not require any SnapVault operations. You can also perform LUN or Volume level cloning to recover data.

Scenario based example

This scenario describes how to restore the `sales_ora` LUN residing in the `oracle1_qtree` qtree in the Primary_Node:

- If the `sv_hourly_4` which is still in Primary_Node has a defined retention policy configured and if the `sales_ora` LUN is mapped Server1, execute the following command on the Server1 where LUN `sales_ora` is mapped:

  ```
  C:\> sdcli snap restore -d e: -s sv_hourly_4
  ```

- To partially recover data from the `sales_ora` LUN which is mounted on E:\ drive on the Server1, you can use the local `sv_hourly.4` Snapshot copy and use the following mount command to mount to s:\ drive and perform a recover:

  ```
  C:\Users\Administrator>sdcli snap mount -r iscsi -k e -s sv_hourly.4 -d s
  ```

8 CONCLUSION

SnapDrive for Windows is a tool for storage provisioning and for Snapshot management on Windows systems connecting to NetApp FAS series storage systems. Data ONTAP has several

unique features for data protection, including Snap Mirror and SnapVault. This technical report details how to use SnapDrive for Windows and SnapVault together to backup and recover data.

9 REFERENCES

- Data ONTAP Data Protection Guide:
  http://now.netapp.com/NOW/knowledge/docs/ontap/rel724/html/ontap/onlinebk/5snapv27.htm
- SnapVault Best Practice Guide:
  http://www.netapp.com/us/library/technical-reports/TR-3487.html
- Data protection portal:
  http://www.netapp.com/solutions/data_protection.html
- NearStore product information:
  http://www.netapp.com/products/nearstore/
- SnapVault product overview:
  http://www.netapp.com/solutions/data_protection-br.html
- Data Protection Strategies for NetApp Storage Controllers:
  http://www.netapp.com/library/tr/3066.pdf
- Best Practices Guide for Tape with NearStore Appliances:
  http://www.netapp.com/library/tr/3149.pdf