

## 7.9 VxVM Snapshots

A VxVM snapshot is used to create a frozen image of a volume at a specific time. A common use for snapshots is to use the snapshot as a source for backups without directly affecting access to the main volume. In theory, it sounds like a VxFS filesystem snapshot. However, a VxVM snapshot is a complete mirror of the volume. This means that when we come to use the snapshot, we will not affect IO to the original volume. This also means that we can treat a VxVM snapshot like a split mirror. We can create a snapshot and then perform tasks such as application upgrades or batch processing. If something goes wrong with our software update, we can resynchronize the main volume from the snapshot, returning the volume to its former state.

Another advantage of using a VxVM snapshot is that there are times when we use raw volumes to store data with no underlying filesystem. A VxFS snapshot is (obviously) dependent on the VxFS filesystem being in place.

The process of creating a snapshot is relatively straightforward; we start by performing a `snapstart`:

```
root@hpeos003[] vxassist -g oral snapstart data3
root@hpeos003[] vxprint -g oral
root@hpeos003[] vxprint -g oral data3
TY NAME          ASSOC          KSTATE    LENGTH    PLOFFS    STATE      TUTILO    PUTILO
v  data3         fsgen         ENABLED   4194304  -         ACTIVE    -         -
pl data3-01      data3         ENABLED   4194304  -         SNAPDONE  -         -
sd ora_disk4-06 data3-01      ENABLED   2097152  0         -         -         -
sd ora_disk2-04 data3-01      ENABLED   2097152  0         -         -         -
pl data3-03      data3         ENABLED   4194304  -         ACTIVE    -         -
sv data3-S01     data3-03      ENABLED   2097152  0         -         -         -
sv data3-S02     data3-03      ENABLED   2097152  0         -         -         -
root@hpeos003[]
```

This is the part of the process that can take some time. The `snapstart` is setting up an additional plex containing a complete copy of the volume. We can choose which disk to store the snapshot by simply specifying the disk media name on the `vxassist` command line. Only when the state of the plex is `SNAPDONE` can we be sure that all data has been duplicated to the snapshot plex. There is an option to `vxassist` called `snapwait` (`vxassist -g oral snapwait data3`) that would simply wait until the snapshot plex has gone to a state of `SNAPDONE` before exiting. This can be useful if we run the `snapstart` in the background.

This plex will continue to be *in sync* with the rest of the volume (just like a mirror) until we are ready to perform our backup. When we are ready to *break off* the snapshot, we use the `snapshot` option:

```
root@hpeos003[] vxassist -g oral snapshot data3 SNAPdata3
root@hpeos003[] vxprint -g oral SNAPdata3
TY NAME          ASSOC          KSTATE    LENGTH    PLOFFS    STATE      TUTILO    PUTILO
v  SNAPdata3     fsgen         ENABLED   4194304  -         ACTIVE    -         -
pl data3-01      SNAPdata3     ENABLED   4194304  -         ACTIVE    -         -
sd ora_disk4-06 data3-01      ENABLED   2097152  0         -         -         -
sd ora_disk2-04 data3-01      ENABLED   2097152  0         -         -         -
root@hpeos003[]
```

Our new volume is called `SNAPdata3`. For this example, I created a filesystem in the original archive. It's a good idea to run the `fsck` command against the filesystem before mounting it to ensure that it is CLEAN. Once mounted, we can back up the data in the snapshot without affecting the original volume.

```
root@hpeos003[] fstyp /dev/vx/dsk/oral/SNAPdata3
vxfs
root@hpeos003[] fsck -F vxfs /dev/vx/dsk/oral/SNAPdata3
file system is clean - log replay is not required
root@hpeos003[]
root@hpeos003[] mkdir /SNAPdata3
root@hpeos003[] mount /dev/vx/dsk/oral/SNAPdata3 /SNAPdata3
root@hpeos003[] bdf
Filesystem          kbytes    used    avail  %used  Mounted on
/dev/vg00/lvol3    1335296 1073984 245026   81% /
```

```

/dev/vg00/lvol1      111637   56439   44034   56% /stand
/dev/vg00/lvol18    1638400 572157  999634  36% /var
/dev/vg00/lvol19    1024000   1752  958364   0% /var/mail
/dev/vg00/lvol17    1122304 820652  282882  74% /usr
/dev/vg00/lvol14     65536    2033   59593   3% /tmp
/dev/vg00/lvol16    1638400 991888  606321  62% /opt
/dev/vg00/lvol15     24576    5814   17626  25% /home
/dev/vx/dsk/oral/data3
      4194304 516287 3448147  13% /data3
/dev/vx/dsk/oral/SNAPdata3
      4194304 516287 3448147  13% /SNAPdata3
root@hpeos003[]

```

I am now going to make some changes in the original volume just to simulate normal online activity:

```

root@hpeos003[] cp /stand/vmunix /data3
root@hpeos003[] bdf
Filesystem      kbytes   used    avail %used Mounted on
/dev/vg00/lvol13 1335296 1073984 245026  81% /
/dev/vg00/lvol11 111637   56439   44034   56% /stand
/dev/vg00/lvol18 1638400 572157  999634  36% /var
/dev/vg00/lvol19 1024000   1752  958364   0% /var/mail
/dev/vg00/lvol17 1122304 820652  282882  74% /usr
/dev/vg00/lvol14  65536    2033   59593   3% /tmp
/dev/vg00/lvol16 1638400 991888  606321  62% /opt
/dev/vg00/lvol15  24576    5814   17626  25% /home
/dev/vx/dsk/oral/data3
      4194304 542255 3423802  14% /data3
/dev/vx/dsk/oral/SNAPdata3
      4194304 516287 3448147  13% /SNAPdata3
root@hpeos003[] ll /data3
total 52000
drwxr-xr-x  5 root      bin           11264 Oct 30 09:32 bin
dr-xr-xr-x 39 bin        bin           10240 Oct 31 20:26 lib
drwxr-xr-x  2 root      root          96 Feb 18 2003 lost+found
dr-xr-xr-x 10 bin        bin           1024 Feb 18 2003 newconfig
drwxr-xr-x  8 bin        bin           10240 Nov  9 16:56 sbin
drwxr-xr-x  4 root      sys            96 Nov 12 00:03 share
-rwxr-xr-x  1 root      sys          26590752 Nov 12 00:47 vmunix
root@hpeos003[]

```

Once my backup has finished, I can decide to do one of four things:

1. Get rid of the snapshot. We have no further use of the snapshot, so we can simply get rid of it. If there is a filesystem in the volume, we should `umount` the filesystem first:

```
# umount /SNAPdata3 # vxassist -g oral remove volume SNAPdata3
```

2. Re-associate the snapshot with the original volume. We may decide to do this, because we want the snapshot to be in sync with the original volume. Tomorrow evening, we can issue another `vxassist snapshot` in order to perform another backup. Some people would refer to this as *resyncing the snapshot*. If there is a filesystem in the volume, we should `umount` the filesystem first.

```
# umount /SNAPdata3 # vxassist -g oral snapback data3
```

3. Disassociate the snapshot from the original volume. We may want to use the snapshot on a long-term basis with no desire to ever re-associate it with the original volume.

```
# vxassist -g oral snapclear SNAPdata3
```

4. Resynchronize the snapshot *from* the snapshot. This is commonly referred as a *reverse resync*. In this situation, we will overwrite the original volume with the data in the snapshot. It is common practice to take a snapshot before performing a major software update. If that update fails, for whatever reason, we can resynchronize the original volume from the snapshot.

You can explore the first three options in your own time (option 2 is the most common). We look at the last option, the *reverse resync*, here. This option is used to return the original volume to its state at the time of the snapshot. We use this option only when we have to return the original volume to its previous state, e.g., a software update has failed and the data in the volume is unusable. To start this process, we will `umount` both the snapshot volume and the original volume.

```
root@hpeos003[] umount /data3
```

We will now use the `resyncfromreplica` option to return the original volume to its previous state.

```
root@hpeos003[] vxassist -g oral -o resyncfromreplica snapback SNAPdata3
```

```
root@hpeos003[]
```

```
root@hpeos003[] vxprint -g oral data3
```

TY	NAME	ASSOC	KSTATE	LENGTH	PLOFFS	STATE	TUTIL0	PUTILO
v	data3	fsgen	ENABLED	4194304	-	ACTIVE	-	-
pl	data3-01	data3	ENABLED	4194304	-	SNAPDONE	-	-
sd	ora_disk4-06	data3-01	ENABLED	2097152	0	-	-	-
sd	ora_disk2-04	data3-01	ENABLED	2097152	0	-	-	-
pl	data3-03	data3	ENABLED	4194304	-	ACTIVE	-	-
sv	data3-S01	data3-03	ENABLED	2097152	0	-	-	-
sv	data3-S02	data3-03	ENABLED	2097152	0	-	-	-

```
root@hpeos003[]
```

Again, this process can take some time because we are copying all the data from the snapshot volume back to the original volume. Just to confirm that the volume has returned to its original state, I will mount the filesystem and check for the existence of the changes made previously.

```
root@hpeos003[] mount /dev/vx/dsk/oral/data3 /data3
```

```
root@hpeos003[] ll /data3
```

```
total 64
drwxr-xr-x  5 root      bin           11264 Oct 30 09:32 bin
dr-xr-xr-x 39 bin       bin           10240 Oct 31 20:26 lib
drwxr-xr-x  2 root      root          96 Feb 18 2003 lost+found
dr-xr-xr-x 10 bin       bin           1024 Feb 18 2003 newconfig
drwxr-xr-x  8 bin       bin           10240 Nov  9 16:56 sbin
drwxr-xr-x  4 root      sys            96 Nov 12 00:03 share
```

```
root@hpeos003[] bdf /data3
```

Filesystem	kbytes	used	avail	%used	Mounted on
/dev/vx/dsk/oral/data3	4194304	516287	3448147	13%	/data3

```
root@hpeos003[] vxprint -g oral data3
```

TY	NAME	ASSOC	KSTATE	LENGTH	PLOFFS	STATE	TUTIL0	PUTILO
v	data3	fsgen	ENABLED	4194304	-	ACTIVE	-	-
pl	data3-01	data3	ENABLED	4194304	-	SNAPDONE	-	-
sd	ora_disk4-06	data3-01	ENABLED	2097152	0	-	-	-
sd	ora_disk2-04	data3-01	ENABLED	2097152	0	-	-	-
pl	data3-03	data3	ENABLED	4194304	-	ACTIVE	-	-
sv	data3-S01	data3-03	ENABLED	2097152	0	-	-	-
sv	data3-S02	data3-03	ENABLED	2097152	0	-	-	-

```
root@hpeos003[]
```

As you can see, all appears well. The snapshot and the original volume are now in sync. If I want to get rid of the snapshot when it is in this state, I can use the `snapabort` command:

```
root@hpeos003[] vxassist -g oral snapabort data3
```

```
root@hpeos003[] vxprint -g oral data3
```

TY	NAME	ASSOC	KSTATE	LENGTH	PLOFFS	STATE	TUTIL0	PUTILO
v	data3	fsgen	ENABLED	4194304	-	ACTIVE	-	-
pl	data3-03	data3	ENABLED	4194304	-	ACTIVE	-	-
sv	data3-S01	data3-03	ENABLED	2097152	0	-	-	-
sv	data3-S02	data3-03	ENABLED	2097152	0	-	-	-

```
root@hpeos003[]
```

The volume has now returned to its original state before we started snapshots.

If you are going to use snapshots often, it is worth considering using the option known as Persistent FastResync. This utilizes a special plex known as a Data Change Object (DCO) and DCO log plex. This works in a similar way to a Dirty Region Log for a mirror volume. To add a DCO to a snap volume, we can use the `vxassist addlog` or the `vxdc` command. This can significantly increase speed of snapshot resync.