
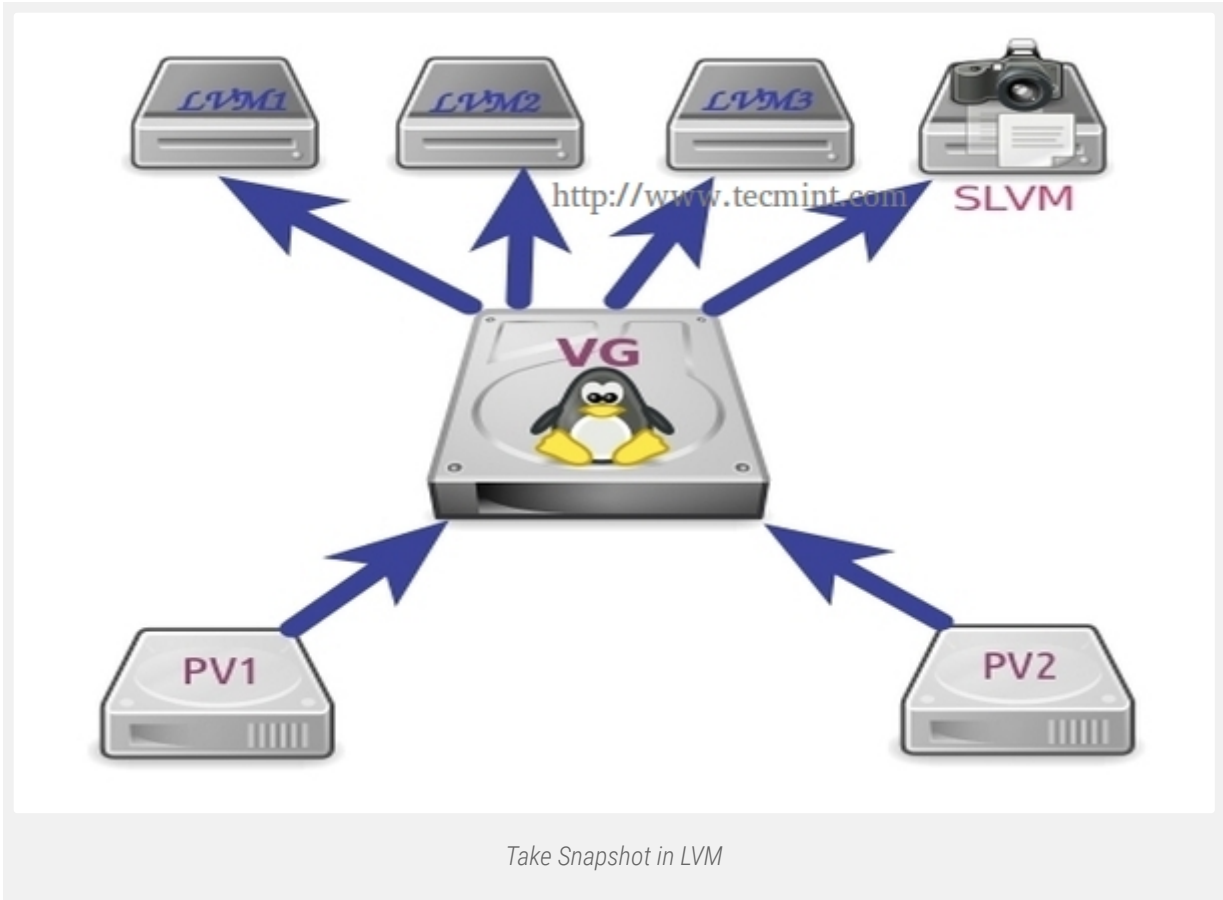


How to Take ‘Snapshot of Logical Volume and Restore’ in LVM – Part III

by Babin Lonston | Published: August 13, 2014 | Last Updated: January 7, 2015

 Download Your Free eBooks NOW - [10 Free Linux eBooks for Administrators](#) | [4 Free Shell Scripting eBooks](#)

LVM Snapshots are space efficient pointing time copies of lvm volumes. It works only with lvm and consume the space only when changes are made to the source logical volume to snapshot volume. If source volume has a huge changes made to sum of 1GB the same changes will be made to the snapshot volume. Its best to always have a small size of changes for space efficient. Incase the snapshot runs out of storage, we can use lvextend to grow. And if we need to shrink the snapshot we can use lvreduce.



If we have accidentally deleted any file after creating a Snapshot we don't have to worry because the snapshot have the original file which we have deleted. It is possible if the file was there when the snapshot was created. Don't alter the snapshot volume, keep as it while snapshot used to do a fast recovery.

Snapshots can't be use for backup option. Backups are Primary Copy of some data's, so we cant use snapshot as a backup option.

Requirements

- [Create Disk Storage with LVM in Linux – PART 1](#)
- [How to Extend/Reduce LVM's in Linux – Part II](#)

My Server Setup

- Operating System – CentOS 6.5 with LVM Installation
- Server IP – 192.168.0.200

Step 1: Creating LVM Snapshot

First, check for free space in volume group to create a new snapshot using following 'vgs' command.

```
# vgs
# lvs
```

root@tecmin:~

```
[root@tecmin ~]# vgs
VG                #PV #LV #SN Attr   VSize  VFree
vg_tecmin         2   2   0 wz--n- 35.50g    0
vg_tecmin_extra   1   1   0 wz--n- 18.00g  8.00g
[root@tecmin ~]#
[root@tecmin ~]# lvs
LV                VG             Attr      LSize   Pool Origin Data%
LogVol00          vg_tecmin     -wi-ao---- 1.00g
LogVol01          vg_tecmin     -wi-ao---- 34.50g
tecmin_datas      vg_tecmin_extra -wi-a----- 10.00g
```

Check LVM Disk Space

You see, there is 8GB of free space left in above **vgs** output. So, let's create a snapshot for one of my volume named **tecmin_datas**. For demonstration purpose, I am going to create only 1GB snapshot volume using following commands.

```
# lvcreate -L 1GB -s -n tecmin_datas_snap /dev/vg_tecmin_extra/tecmin_datas
OR
# lvcreate --size 1G --snapshot --name tecmin_datas_snap /dev/vg_tecmin_extra/tecmin_datas
```

Both the above commands does the same thing:

- **-s** – Creates Snapshot
- **-n** – Name for snapshot

root@tecmin:~

```
[root@tecmin ~]#
[root@tecmin ~]# lvcreate -L 1GB -s -n tecmin_datas_snap /dev/vg_tecmin_extra/tecmin_datas
Logical volume "tecmin_datas_snap" created
[root@tecmin ~]#
```

Create LVM Snapshot

Here, is the explanation of each point highlighted above.

- Size of snapshot Iam creating here.
- Creates snapshot.
- Creates name for the snapshot.
- New snapshots name.
- Volume which we are going to create a snapshot.

If you want to remove a snapshot, you can use '**lvremove**' command.

```
# lvremove /dev/vg_tecmin_extra/tecmin_datas_snap
```

root@tecmin:~

```
[root@tecmin ~]#
[root@tecmin ~]# lvremove /dev/vg_tecmin_extra/tecmin_datas_snap
Do you really want to remove active logical volume tecmin_datas_snap? [y/n]: y
Logical volume "tecmin_datas_snap" successfully removed
[root@tecmin ~]#
```

Remove LVM Snapshot

Now, list the newly created snapshot using following command.

```
# lvs
```

root@tecmin~

```
[root@tecmin ~]# lvs
LV          VG      Attr      LSize   Pool Origin         Data%  Move Log
LogVol00    vg_tecmin -wi-ao--- 1.00g
LogVol01    vg_tecmin -wi-ao--- 34.50g
tecmin data snap vg_tecmin extra swi-a-s--- 1.00g   tecmin datas  0.00
tecmin datas vg_tecmin_extra owi-a-s--- 10.00g
[root@tecmin ~]#
```

Verify LVM Snapshot

You see above, a snapshot was created successfully. I have marked with an arrow where snapshots origin from where its created, Its **tecmin_datas**. Yes, because we have created a snapshot for **tecmin_datas** l-volume.

root@tecmin~

Terminal

```
[root@tecmin ~]# df -Th
Filesystem                                Type  Size  Used Avail Use% Mounted on
/dev/mapper/vg_tecmin-LogVol01             ext4   34G   2.2G   31G   7% /
tmpfs                                       tmpfs  939M    0   939M   0% /dev/shm
/dev/vda1                                  ext4   485M   39M   421M   9% /boot
/dev/mapper/vg_tecmin_extra-tecmin_datas  ext4   9.9G  642M   8.8G   7% /mnt/tecmin_datas
[root@tecmin ~]#
```

Check LVM Snapshot Space

Let's add some new files into **tecmin_datas**. Now volume has some data's around 650MB and our snapshot size is 1GB. So there is enough space to backup our changes in snap volume. Here we can see, what is the status of our snapshot using below command.

```
# lvs
```

root@tecmin~

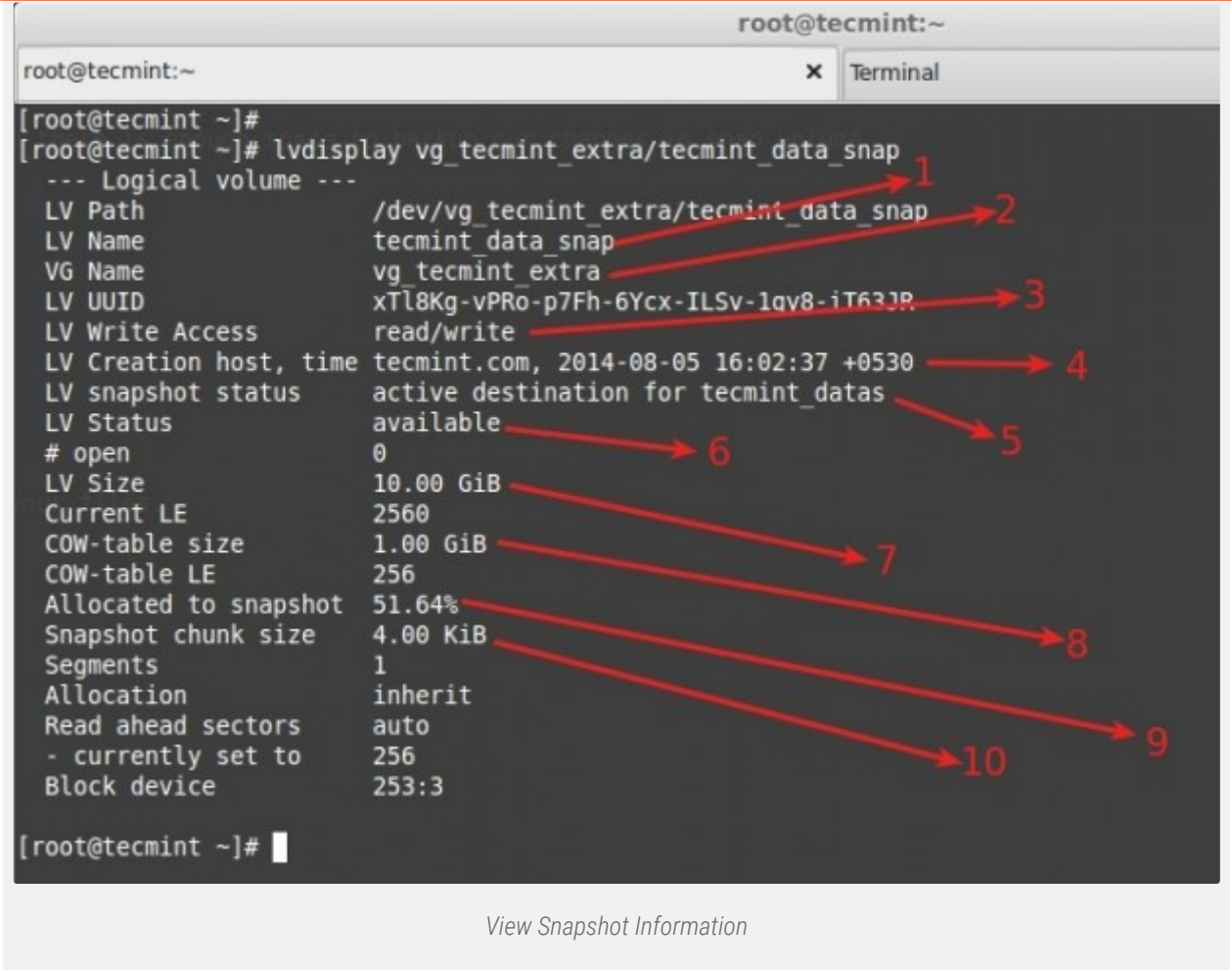
Terminal

```
[root@tecmin ~]# lvs
LV          VG      Attr      LSize   Pool Origin         Data%  Move Log
LogVol00    vg_tecmin -wi-ao--- 1.00g
LogVol01    vg_tecmin -wi-ao--- 34.50g
tecmin data snap vg_tecmin extra swi-a-s--- 1.00g   tecmin datas  51.64
tecmin datas vg_tecmin_extra owi-a-s--- 10.00g
[root@tecmin ~]#
```

Check Snapshot Status

You see, **51%** of snapshot volume was used now, no issue for more modification in your files. For more detailed information use command.

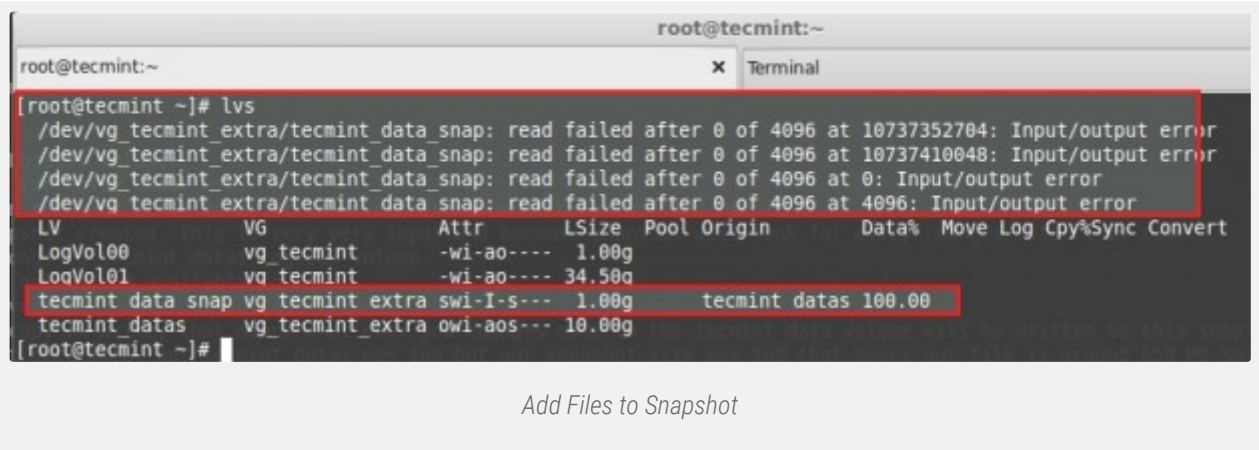
```
# lvdisplay vg_tecmin_extra/tecmin_data_snap
```



Again, here is the clear explanation of each point highlighted in the above picture.

- Name of Snapshot Logical Volume.
- Volume group name currently under use.
- Snapshot volume in read and write mode, we can even mount the volume and use it.
- Time when the snapshot was created. This is very important because snapshot will look for every changes after this time.
- This snapshot belongs to tecmint_datas logical volume.
- Logical volume is online and available to use.
- Size of Source volume which we took snapshot.
- Cow-table size = copy on Write, that means whatever changes was made to the tecmint_data volume will be written to this snapshot.
- Currently snapshot size used, our tecmint_datas was 10G but our snapshot size was 1GB that means our file is around 650 MB. So what its now in 51% if the file grow to 2GB size in tecmint_datas size will increase more than snapshot allocated size, sure we will be in trouble with snapshot. That means we need to extend the size of logical volume (snapshot volume).
- Gives the size of chunk for snapshot.

Now, let's copy more than 1GB of files in **tecmint_datas**, let's see what will happen. If you do, you will get error message saying 'Input/output error', it means out of space in snapshot.



If the logical volume become full it will get dropped automatically and we can't use it any more, even if we extend the size of snapshot volume. It is the best idea to have the same size of Source while creating a snapshot, **tecmint_datas** size was 10G, if I create a snapshot size of 10GB it will never over flow like above because it has enough space to take snap of your volume.

Step 2: Extend Snapshot in LVM

If we need to extend the snapshot size before overflow we can do it using.

```
# lvextend -L +1G /dev/vg_tecmint_extra/tecmin_data_snap
```


Now there was totally 2GB size for snapshot.

```
[root@tecmin ~]# lvextend -L +1G /dev/vg tecmint extra/tecmint data snap
Extending logical volume tecmint_data_snap to 2.00 GiB
Logical volume tecmint_data_snap successfully resized
```

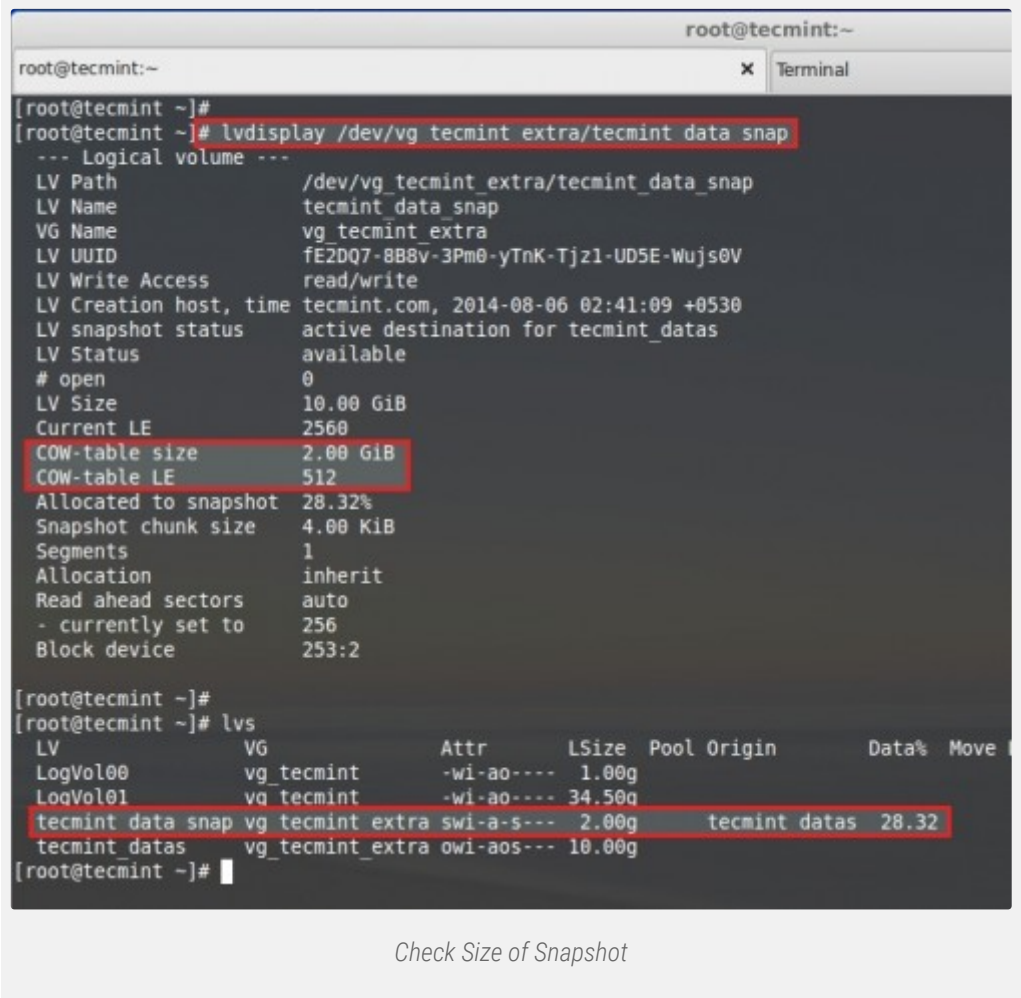
Extend LVM Snapshot

Next, verify the new size and COW table using following command.

```
# lvdisplay /dev/vg_tecmint_extra/tecmint_data_snap
```

To know the size of snap volume and usage %.

```
# lvs
```

A screenshot of a terminal window titled 'root@tecmin:~'. The terminal shows the output of two commands. The first command is 'lvdisplay /dev/vg tecmint extra/tecmint data snap', which displays detailed information about the logical volume 'tecmint_data_snap', including its path, name, VG, UUID, write access, creation time, snapshot status, size (10.00 GiB), and COW-table size (2.00 GiB). The second command is 'lvs', which shows a table of logical volumes. The 'tecmint data snap' entry is highlighted, showing its size as 2.00g and its usage as 28.32%.

```
root@tecmin:~
root@tecmin:~
[root@tecmin ~]#
[root@tecmin ~]# lvdisplay /dev/vg tecmint extra/tecmint data snap
--- Logical volume ---
LV Path                /dev/vg_tecmint_extra/tecmint_data_snap
LV Name                 tecmint_data_snap
VG Name                 vg_tecmint_extra
LV UUID                 fE2DQ7-8B8v-3Pm0-yTnK-Tjz1-UD5E-Wujs0V
LV Write Access         read/write
LV Creation host, time  tecmint.com, 2014-08-06 02:41:09 +0530
LV snapshot status      active destination for tecmint_datas
LV Status                available
# open                  0
LV Size                 10.00 GiB
Current LE              2560
COW-table size          2.00 GiB
COW-table LE            512
Allocated to snapshot   28.32%
Snapshot chunk size     4.00 KiB
Segments                1
Allocation              inherit
Read ahead sectors      auto
- currently set to      256
Block device            253:2

[root@tecmin ~]#
[root@tecmin ~]# lvs
LV          VG          Attr      LSize  Pool Origin     Data%  Move
LogVol00    vg_tecmint    -wi-ao--- 1.00g
LogVol01    vg_tecmint    -wi-ao--- 34.50g
tecmint data snap  vg_tecmint extra swi-a-s--- 2.00g  tecmint datas 28.32
tecmint datas  vg_tecmint_extra owi-aos--- 10.00g

[root@tecmin ~]#
```

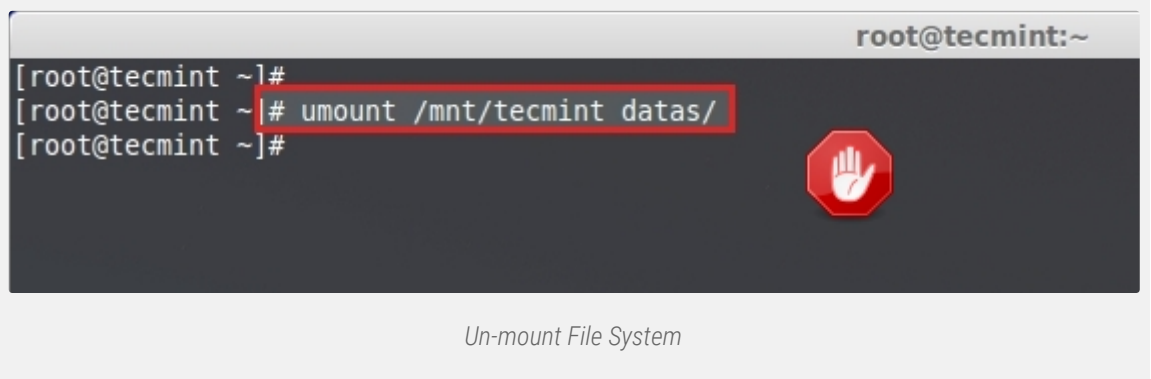
Check Size of Snapshot

But if, you have snapshot volume with the same size of Source volume we don't need to worry about these issues.

Step 3: Restoring Snapshot or Merging

To restore the snapshot, we need to un-mount the file system first.

```
# umount /mnt/tecmin_data/
```

A screenshot of a terminal window titled 'root@tecmin:~'. The terminal shows the execution of the 'umount /mnt/tecmin_data/' command. A red octagonal stop sign icon is visible on the right side of the terminal window.

```
root@tecmin:~
[root@tecmin ~]#
[root@tecmin ~]# umount /mnt/tecmin_data/
[root@tecmin ~]#
```

Un-mount File System

Just check for the mount point whether its unmounted or not.

```
# df -h
```

root@tecmin~

```
[root@tecmin ~]# df -h
Filesystem                Size      Used Avail Use% Mounted on
/dev/mapper/vg_tecmin-LogVol01  34G    2.2G    31G   7% /
tmpfs                      939M         0  939M   0% /dev/shm
/dev/vda1                  485M     39M   421M   9% /boot
[root@tecmin ~]#
```

Check File System Mount Points

Here our mount has been unmounted, so we can continue to restore the snapshot. To restore the snap using command **lvconvert**.

```
# lvconvert --merge /dev/vg_tecmin_extra/tecmin_data_snap
```

root@tecmin~

```
[root@tecmin ~]#
[root@tecmin ~]# lvconvert --merge /dev/vg_tecmin_extra/tecmin_data_snap
Merging of volume tecmin_data_snap started.
tecmin_datas: Merged: 70.9%
tecmin_datas: Merged: 86.7%
tecmin_datas: Merged: 100.0%
Merge of snapshot into logical volume tecmin_datas has finished.
Logical volume "tecmin_data_snap" successfully removed
[root@tecmin ~]#
```

Restore LVM Snapshot

After the merge is completed, snapshot volume will be removed automatically. Now we can see the space of our partition using **df** command.

```
# df -Th
```

root@tecmin~

Terminal

```
[root@tecmin ~]#
[root@tecmin ~]# df -Th
Filesystem                Type      Size      Used Avail Use% Mounted on
/dev/mapper/vg_tecmin-LogVol01  ext4    34G    2.2G    31G   7% /
tmpfs                      tmpfs    939M         0  939M   0% /dev/shm
/dev/vda1                  ext4    485M     39M   421M   9% /boot
/dev/mapper/vg_tecmin_extra-tecmin_datas  ext4    9.9G    434M   9.0G   5% /mnt/tecmin_datas
[root@tecmin ~]#
```

Check Size of Snapshot

After the snapshot volume removed automatically. You can see the size of logical volume.

```
# lvs
```

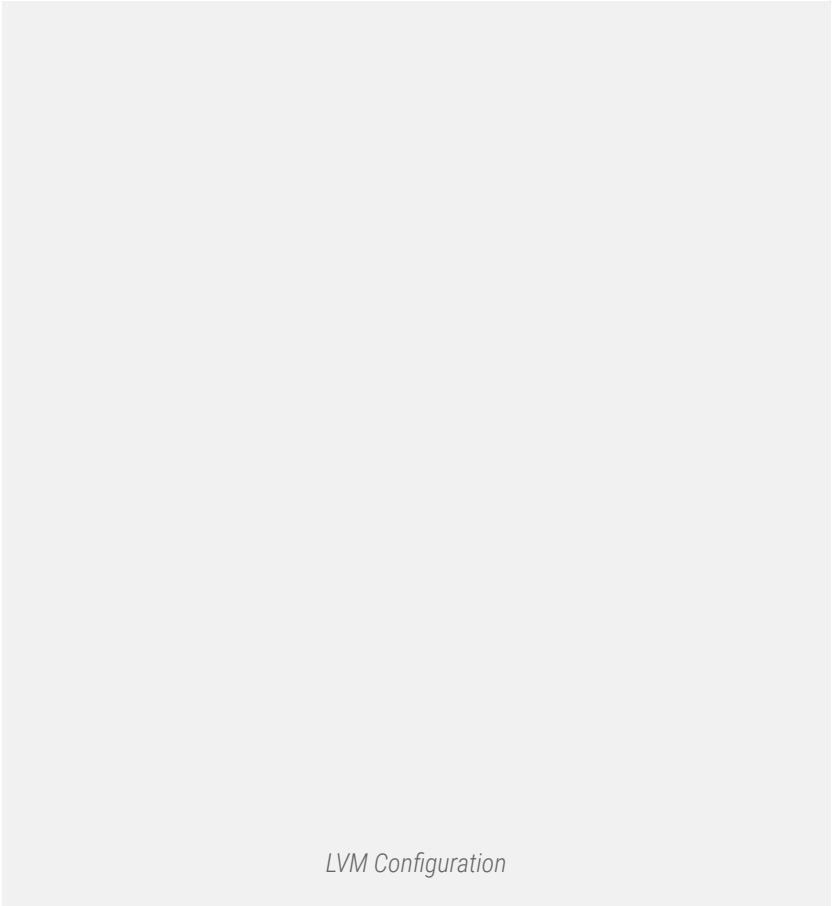


Important: To Extend the Snapshots automatically, we can do it using some modification in conf file. For manual we can extend using lvextend.

Open the lvm configuration file using your choice of editor.

```
# vim /etc/lvm/lvm.conf
```

Search for word autoextend. By Default the value will be similar to below.



Change the **100** to **75** here, if so auto extend threshold is **75** and auto extend percent is **20**, it will expand the size more by **20 Percent**

If the snapshot volume reach **75%** it will automatically expand the size of snap volume by **20%** more. Thus,we can expand automatically. Save and exit the file using **wq!**.

This will save snapshot from overflow drop. This will also help you to save more time. LVM is the only Partition method in which we can expand more and have many features as thin Provisioning, Striping, Virtual volume and more Using thin-pool, let us see them in the next topic.

