



Ganpat University

॥ विद्या समाजोत्कर्षः ॥

Institute of
Computer
Technology

Name: Tushar Panchal

En.No: 21162101014

Sub: ITIM (IT Infrastructure & Management)

Branch: CBA

Batch:61

PRACTICAL 06

AIM : To understand the advanced concepts of LVM like changing its size, dynamic size allocation, etc

- Demonstrate how to increase the size of the volume group when the memory is exhausted in the volume group.**
- » **First, create the physical volumes, volume group and logical volume using the commands for the same :**

```

Red Hat
Activities Terminal
Feb 7 11:30
root@server:~ x
File Edit View Search Terminal Help
Welcome to fdisk (util-linux 2.32.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0xe10d57e7.

Command (m for help): n
Partition type
  p  primary (0 primary, 0 extended, 4 free)
  e  extended (container for logical partitions)
Select (default p):

Using default response p.
Partition number (1-4, default 1):
First sector (2048-10485759, default 2048):
Last sector, +sectors or +size(K,M,G,T,P) (2048-10485759, default 10485759): +1G

Created a new partition 1 of type 'Linux' and of size 1 GiB.

Command (m for help): n
Partition type
  p  primary (1 primary, 0 extended, 3 free)
  e  extended (container for logical partitions)
Select (default p):

Using default response p.
Partition number (2-4, default 2):
First sector (2099200-10485759, default 2099200):
Last sector, +sectors or +size(K,M,G,T,P) (2099200-10485759, default 10485759): +1G

Created a new partition 2 of type 'Linux' and of size 1 GiB.

Command (m for help): 

```



Activities Terminal Feb 7 11:34 root@servera:~

```
File Edit View Search Terminal Help
First sector (2899200-10485759, default 2099200):
Last sector, +sectors or +size{K,M,G,T,P} (2099200-10485759, default 10485759): +1G
Created a new partition 2 of type 'Linux' and of size 1 GiB.

Command (m for help): n
Partition type
  p  primary (2 primary, 0 extended, 2 free)
  e  extended (container for logical partitions)
Select (default p):

Using default response p.
Partition number (3,4, default 3):
First sector (4196352-10485759, default 4196352):
Last sector, +sectors or +size{K,M,G,T,P} (4196352-10485759, default 10485759): +1G
Created a new partition 3 of type 'Linux' and of size 1 GiB.

Command (m for help): p
Disk /dev/vdb: 5 Gib, 5368709120 bytes, 10485760 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xe1dd57e7

Device     Boot   Start    End Sectors Size Id Type
/dev/vdb1        2048 2099199 20997152  1G 83 Linux
/dev/vdb2      2099200 4196351 20997152  1G 83 Linux
/dev/vdb3      4196352 6293503 20997152  1G 83 Linux

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

[root@servera ~]#
```



Activities Terminal Feb 7 11:37 root@servera:~

```
[root@servera ~]# pvcreate /dev/vdb1 /dev/vdb2 /dev/vdb3
Physical volume "/dev/vdb1" successfully created.
Physical volume "/dev/vdb2" successfully created.
Physical volume "/dev/vdb3" successfully created.
[root@servera ~]# vgcreate tushar /dev/vdb1 /dev/vdb2 /dev/vdb3
Volume group "tushar" successfully created
[root@servera ~]#
[root@servera ~]# lvcreate -L +2G -n lvtushar tushar
Logical volume "lvtushar" created.
[root@servera ~]# mkdir /mnt/lvdir
[root@servera ~]# mkfs.xfs /dev/tushar/lvtushar
meta-data=/dev/tushar/lvtushar isize=512    agcount=4, agsize=131072 blks
          =                     sectsz=512  attr=2, projid32bit=1
          =                     crc=1    finobt=1, sparse=1, rmapbt=0
          =                     reflink=1
data     =                     .          sunits=0    swidth=0 blks
          =                     .          bsizes=4096   blocks=524288, imaxpct=25
          =                     .          bsize=4096   arclv_ci=0  ftype=1
naming   =version 2           bsize=4096   blocks=2560, version=2
log      =internal log        sectsz=512   sunit=0 blks, lazy_count=1
          =                     extsz=4096   blocks=0, rtextents=0
realtime =none
[root@servera ~]#
```



Activities Terminal ▾ Feb 7 11:45 root@servera:~

```
Welcome to fdisk (util-linux 2.32.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): t
Partition number (1-3, default 3):
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'.

Command (m for help): t
Partition number (1-3, default 3): 2
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'.

Command (m for help): t
Partition number (1-3, default 3): 1
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'.

Command (m for help): p
Disk /dev/vdb: 5 GiB, 5368709120 bytes, 10485760 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xe11d57e7

Device      Boot   Start     End  Sectors  Size Id Type
/dev/vdb1        2048 2099199 2097152   16 8e Linux LVM
/dev/vdb2    2099200 4196351 2097152   16 8e Linux LVM
/dev/vdb3    4196352 6293503 2097152   16 8e Linux LVM

Command (m for help):
```

» Now, create a PV and extend VG using that:



Activities Terminal ▾ Feb 7 11:47 root@servera:~

```
[root@servera ~]# fdisk /dev/vdb
Welcome to fdisk (util-linux 2.32.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): n
Partition type:
   p  primary (3 primary, 0 extended, 1 free)
   e  extended (container for logical partitions)
Select (default e): p
Selected partition 4
First sector (6293504-10485759, default 6293504):
Last sector, +sectors or +size[K,M,G,T,P] (6293504-10485759, default 10485759): +100M
Created a new partition 4 of type 'Linux' and of size 100 MiB.

Command (m for help): t
Partition number (1-4, default 4):
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'.

Command (m for help): w
The partition table has been altered.
Syncing disks.

[root@servera ~]# pvcreate /dev/vdb4
Physical volume "/dev/vdb4" successfully created.
[root@servera ~]# vgextend tushar /dev/vdb4
Volume group "tushar" successfully extended
[root@servera ~]#
```

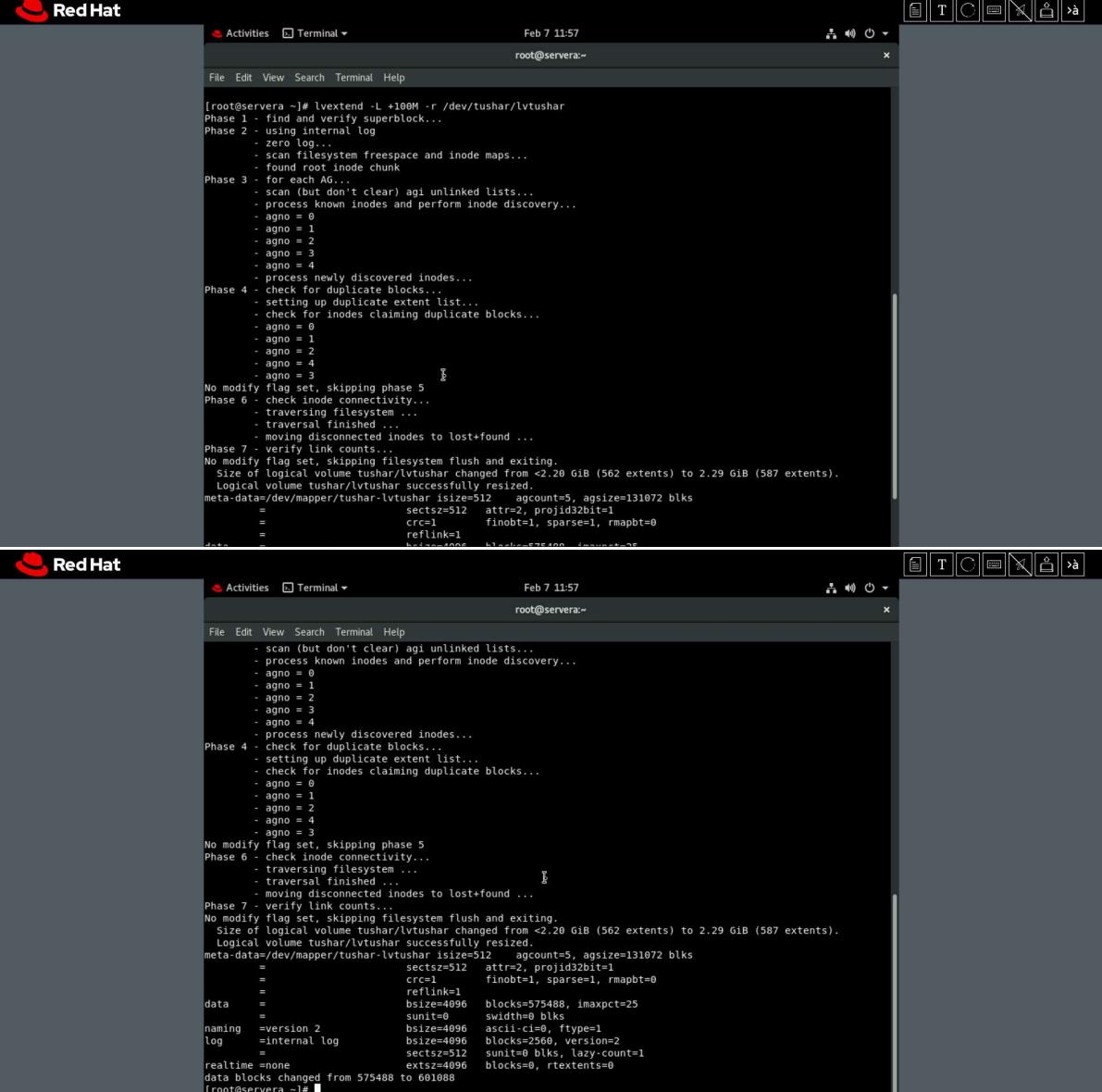
```
[root@servera ~]# pvcreate /dev/vdb4
Physical volume "/dev/vdb4" successfully created.
[root@servera ~]# vgextend tushar /dev/vdb4
Volume group "tushar" successfully extended
[root@servera ~]#
```

command: vgextend tushar /dev/vdb4

the command is extending the size of the Volume Group named "tushar" by adding the physical volume **/dev/vdb4** to it.

» Extend the logical volume now, using the command

lvextend :



```
[root@servera ~]# lvextend -L +100M -r /dev/tushar/lvtushar
Phase 1 - find and verify superblock...
Phase 2 - using internal log
- zero log...
- scan filesystem freespace and inode maps...
- found root inode chunk
Phase 3 - for each AG...
- scan (but don't clear) agi unlinked lists...
- process known inodes and perform inode discovery...
- agno = 0
- agno = 1
- agno = 2
- agno = 3
- agno = 4
- process newly discovered inodes...
Phase 4 - check for duplicate blocks...
- setting up duplicate extent list...
- check for inodes claiming duplicate blocks...
- agno = 0
- agno = 1
- agno = 2
- agno = 3
- agno = 4
- agno = 5
No modify flag set, skipping phase 5
Phase 6 - check inode connectivity...
- traversing filesystem ...
- traversal finished ...
- moving disconnected inodes to lost+found ...
Phase 7 - verify link counts...
No modify flag set, skipping filesystem flush and exiting.
Size of logical volume tushar/lvtushar changed from <2.20 GiB (562 extents) to 2.29 GiB (587 extents).
Logical volume tushar/lvtushar successfully resized.
meta-data=/dev/mapper/tushar-lvtushar isize=512  agcount=5, agsize=131072 blks
          =           sectsz=512  attr=2, projid32bit=1
          =           crc=1        finobt=1, sparse=1, rmapbt=0
          =           reflink=1
          =           btrfs=575488  imaxpct=25
data      =           sectsz=512  attr=2, projid32bit=1
          =           crc=1        finobt=1, sparse=1, rmapbt=0
          =           reflink=1
          =           sumit=0      swi32hd blks
naming   =version 2    bs=2048, isize=4096  firstc=0, ftype=1
log      =internal log  bs=2048, isize=4096  blocks=2860, version=2
          =           sectsz=512  sunit=0 blks, lazy-count=1
realtime =none          extsz=4096
data blocks changed from 575488 to 601088
[root@servera ~]#
```

command: `lvextend -L +100M -r /dev/tushar/lvtushar`

the command is extending the logical volume

/dev/tushar/lvtushar by adding 100 megabytes to its size and automatically resizing the filesystem on that logical volume to make use of the additional space.

» Check if the changes are successful using command

lvdisplay :

```

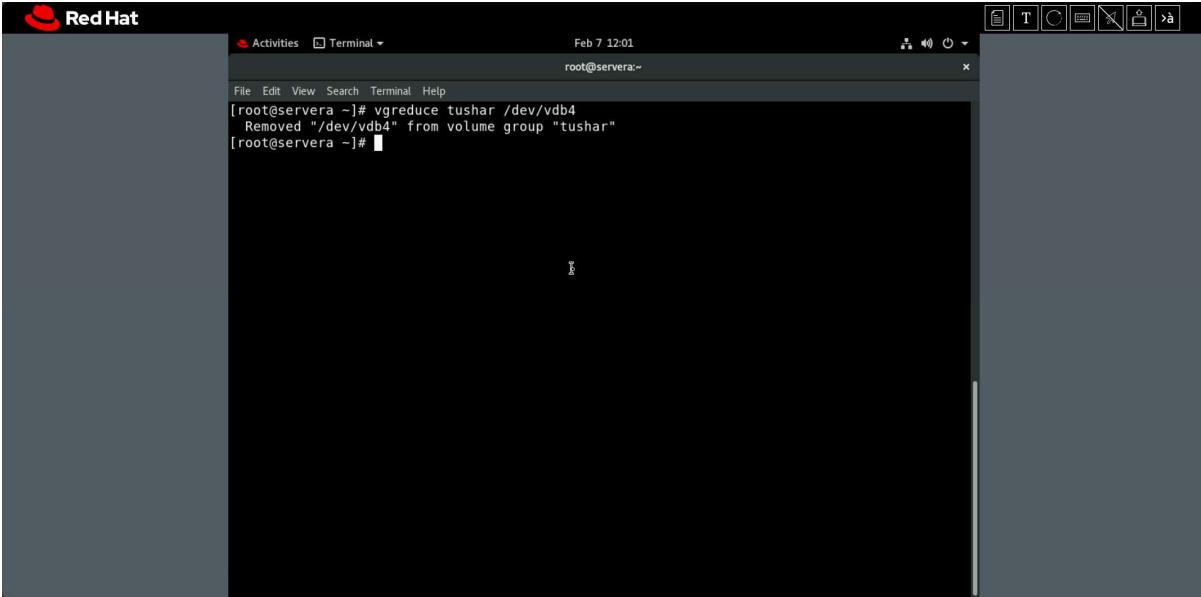
Activities Terminal * Feb 7 11:59
root@servera:~ x
File Edit View Search Terminal Help
- traversing filesystem ...
  - moving lost+found ...
  - moving disconnected inodes to lost+found ...
Phase 7 - verify link counts.
No modify flag set, skipping filesystem flush and exiting.
Size of logical volume tushar/lvtushar changed from <2.20 GiB (562 extents) to 2.29 GiB (587 extents).
Logical volume tushar/lvtushar successfully resized.
meta-data=/dev/mapper/tushar-lvtushar isize=512  agcount=5, agsize=131072 blks
          =         sectsz=512  attr=2, projid32bit=1
          =             crc=1   finobt=1, sparse=1, rmapbt=0
data      =         reflink=1
          =         bsize=4096 blocks=575488, imaxpct=25
          =             sunit=0  swidth=0 blks
naming    =version 2  bsize=4096 ascii-ci=0, ftype=1
log       =internal log bsize=4096 blocks=2560, version=2
          =             sectsz=512 sunit=0 blks, lazy-count=1
realtime  =none    extsz=4096 blocks=0, rtextents=0
data blocks changed from 575488 to 601088
[root@servera ~]# lvdisplay
...
Logical volume ...
LV Path           /dev/tushar/lvtushar
LV Name           lvtushar
VG Name           tushar
LV UUID           srIPig-oxNh-wJvj-BdJl-Ndmt-ZuBo-PFicRr
LV Write Access   read/write
LV Creation host, time servera.lab.example.com, 2024-02-07 11:36:49 -0500
LV Status         available
# open            0
LV Size           2.29 GiB
Current LE        587
Segments          3
Allocation        inherit
Read ahead sectors auto
- currently set to 8192
Block device     253:0
[root@servera ~]#

```

command : lvdisplay

- This command is used to display information about Logical Volumes.

2. Demonstrate how to remove a physical volume from a volume group.



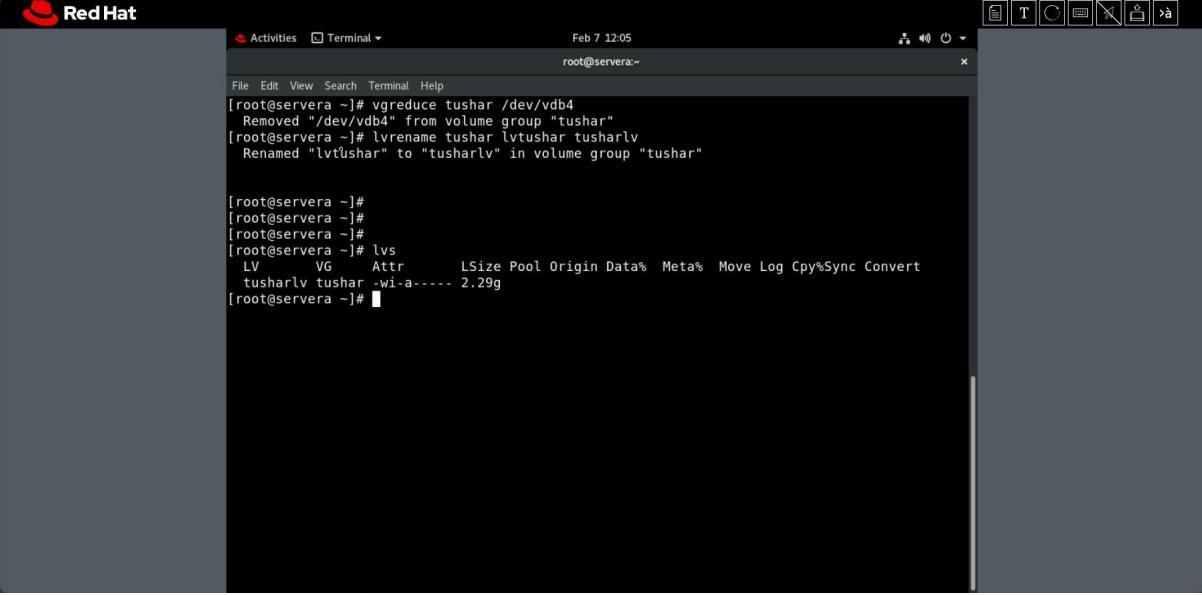
The screenshot shows a terminal window titled "Terminal" with the Red Hat logo. The window title bar includes "Activities", "Terminal", "Feb 7 12:01", and "root@servera:~". The terminal content shows the command [root@servera ~]# vgreduce tushar /dev/vdb4 followed by the output "Removed '/dev/vdb4' from volume group 'tushar'" and the prompt [root@servera ~]#.

command : **vgreduce tushar /dev/vdb4**

The **vgreduce** command in Linux is used to remove physical volumes from a Volume Group (VG) in the Logical Volume Manager (LVM) system. Let's break down the components of the command:

1. **vgreduce**: This is the command itself, used for reducing the size of a Volume Group by removing one or more physical volumes.
2. **tushar**: This is the name of the Volume Group from which you want to remove the physical volume. Replace "tushar" with the actual name of your Volume Group.
3. **/dev/vdb4**: This is the physical volume that you want to remove from the Volume Group named "tushar." Replace "/dev/vdb4" with the actual device or partition you want to remove.

3. Demonstrate how to rename LVM.



The screenshot shows a terminal window titled "Red Hat" with a dark theme. The command history and output are as follows:

```
[root@servera ~]# vgreduce tushar /dev/vdb4
Removed '/dev/vdb4' from volume group "tushar"
[root@servera ~]# lvrename tushar lvtushar tusharlv
Renamed "lvtushar" to "tusharlv" in volume group "tushar"

[root@servera ~]#
[root@servera ~]#
[root@servera ~]# lvs
  LV      VG Attr       LSize Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  tusharlv tushar -wi-a----- 2.29g
[root@servera ~]#
```

command : lvrename tushar lvtushar tusharlv

the command is renaming the logical volume from "lvtushar" to "tusharlv" within the Volume Group named "tushar." After running this command, the logical volume will be referred to as "tusharlv" instead of "lvtushar."

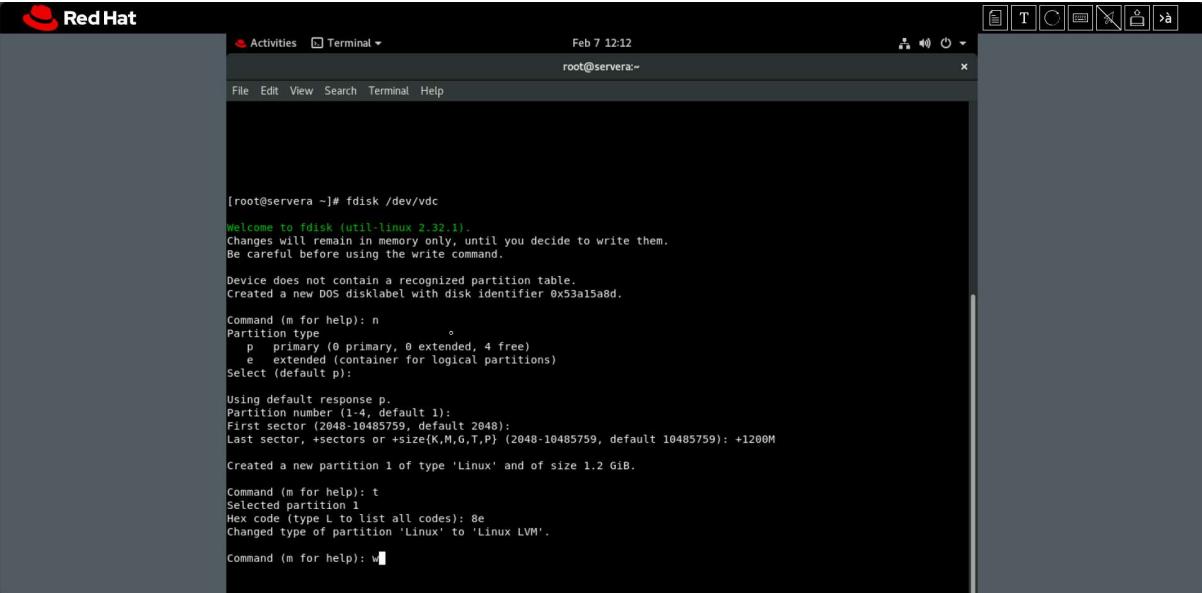
command : lvs

The **lvs** command is used in Linux to display information about Logical Volumes (LVs) within the Logical Volume Manager (LVM) system. This command provides a summary of various attributes and properties of the logical volumes in the specified Volume Group or across all Volume Groups.

4. Create a LVM- Create a partition : 1200M

- 1PE = 16MB**
- Volume Group name: dell**
- logical Vol: lenovo**
- Logical vol size: 20PE**
- Filesystem: xfs**
- Mount Point: /mnt/extend.**

» Create physical volume of size 1200 Megabytes and change the type to LVM :



```
[root@servera ~]# fdisk /dev/vdc
Welcome to fdisk (util-linux 2.32.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x53a15a8d.

Command (m for help): n
Partition type:
   p  primary (0 primary, 0 extended, 4 free)
   e  extended (container for logical partitions)
Select (default p):

Using default response p.
Partition number (1-4, default 1):
First sector (2048-10485759, default 2048):
Last sector, +sectors or +size{K,M,G,T,P} (2048-10485759, default 10485759): +1200M

Created a new partition 1 of type 'Linux' and of size 1.2 GiB.

Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'.

Command (m for help): w
```

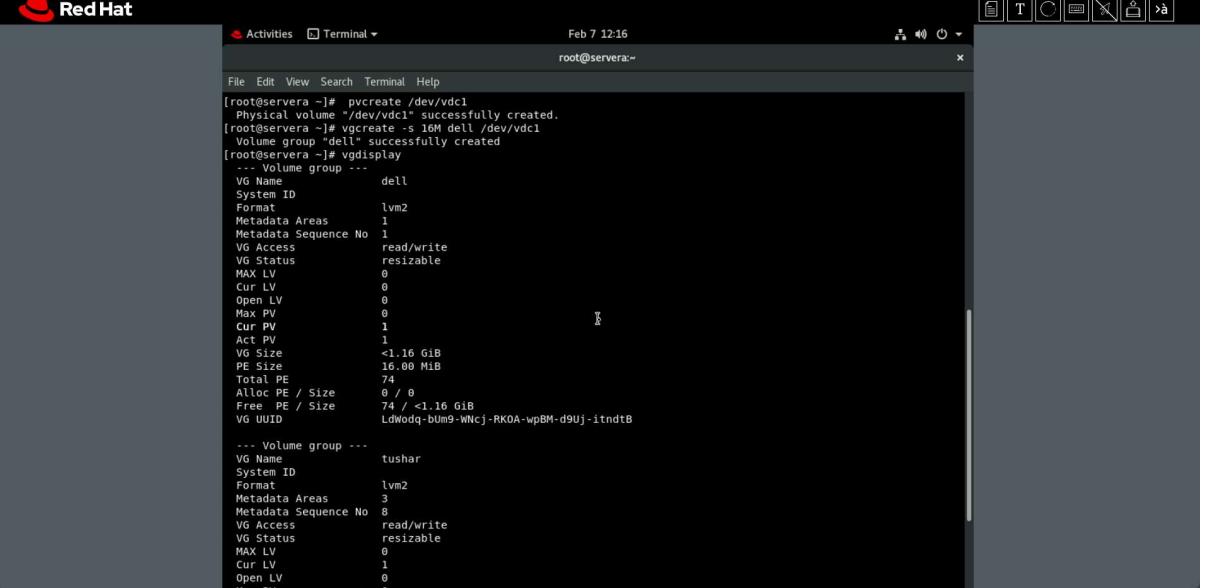
command: fdisk /dev/vdc

The **fdisk** command is a text-based utility in Linux used for disk partitioning. When you run **fdisk /dev/vdc**, you are accessing the disk partitioning tool for the specified device **/dev/vdc**

SubCommands of fdisk: n, p, t, w

1. **n**: Used to create a new partition.
2. **p**: Used to print the partition table.
3. **t**: Used to change the system ID (partition type) of a partition..
4. **w**: Used to write the changes to the disk and exit.

» Create physical volume and volume group:



```
[root@servera ~]# pvcreate /dev/vdc1
Physical volume "/dev/vdc1" successfully created.
[root@servera ~]# vgcreate -s 16M dell /dev/vdc1
Volume group "dell" successfully created
[root@servera ~]# vgdisplay
-- Volume group --
VG Name           dell
System ID         lvm2
Format            lvm2
Metadata Areas   1
Metadata Sequence No  1
VG Access        read/write
VG Status        resizable
MAX LV           0
Cur LV           0
Open LV          0
Max PV           0
Cur PV           1
Act PV           1
VG Size          <1.16 GiB
PE Size          16.00 MiB
Total PE          74
Alloc PE / Size  0 / 0
Free PE / Size   74 / <1.16 GiB
VG UUID          LdWodq-bUm9-WNcj-RKOa-wpBM-d9Uj-itndtB

-- Volume group --
VG Name           tushar
System ID         lvm2
Format            lvm2
Metadata Areas   3
Metadata Sequence No  8
VG Access        read/write
VG Status        resizable
MAX LV           0
Cur LV           1
Open LV          0
```

command: `pvcreate /dev/vdc1`

This command is used to initialize a physical volume (PV) on the specified device, in this case, **/dev/vdc1**.

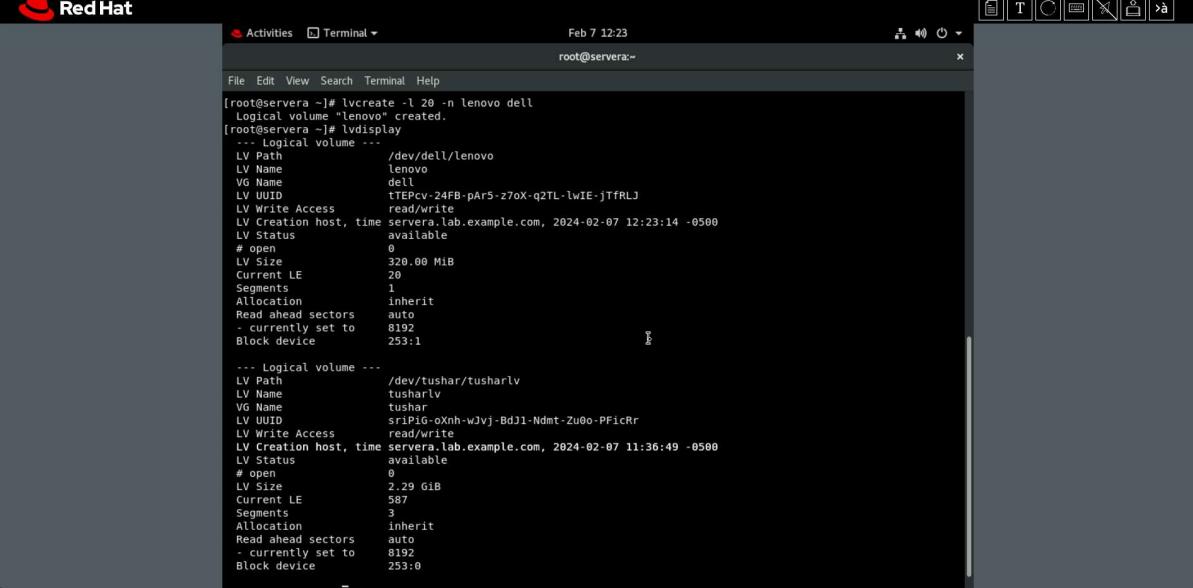
command: `vgcreate -s 16M dell /dev/vdc1`

This command creates a new Volume Group (VG) named "dell" using the physical volume **/dev/vdc1**. The **-s** option sets the extent size for the Volume Group. In this example, the extent size is set to 16 megabytes.

command: `vgdisplay`

- This command is used to display information about volume groups.

» **Now, create logical volume using counts of PEs, instead of size as generally created.**



```
[root@servera ~]# lvcreate -l 20 -n lenovo dell
Logical volume "lenovo" created.
[root@servera ~]# lvdisplay
--- Logical volume ---
LV Path          /dev/dell/lenovo
LV Name          lenovo
VG Name          dell
LV UUID          1TEPcv-24FB-pAr5-z7oX-q2TL-lwIE-jTfRLJ
LV Write Access  read/write
LV Creation host, time servera.lab.example.com, 2024-02-07 12:23:14 -0500
LV Status        available
# open           0
LV Size          320.00 MiB
Current LE       20
Segments         1
Allocation       inherit
Read ahead sectors auto
- currently set to 8192
Block device    253:1

--- Logical volume ---
LV Path          /dev/tushar/tusharlv
LV Name          tusharlv
VG Name          tushar
LV UUID          srlPIG-oxnh-wJvj-BdJ1-Ndmt-Zu00-PFicRr
LV Write Access  read/write
LV Creation host, time servera.lab.example.com, 2024-02-07 11:36:49 -0500
LV Status        available
# open           0
LV Size          2.29 GiB
Current LE       587
Segments         3
Allocation       inherit
Read ahead sectors auto
- currently set to 8192
Block device    253:0
```

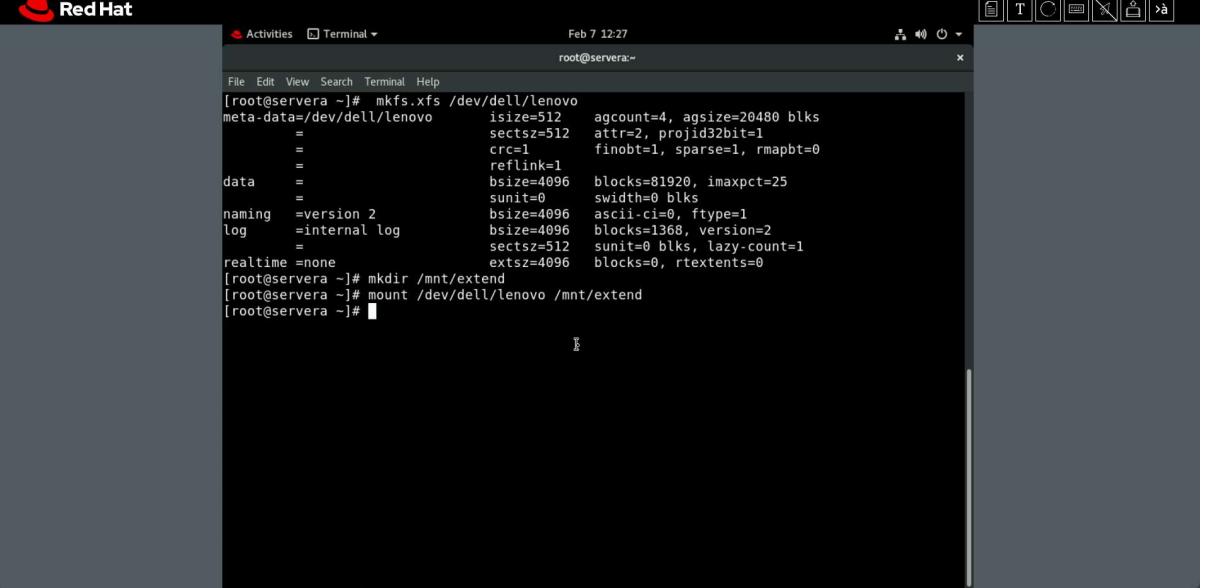
command : lvcreate -l 20 -n lenovo dell

The command is creating a logical volume named "lenovo" within the Volume Group "dell" with a size of 20 extents.

command : lvdisplay

The **lvdisplay** command in Linux is used to display detailed information about logical volumes within the Logical Volume Manager (LVM) system.

» Assign filesystem to LV and mount it at some directory.



The screenshot shows a terminal window titled "Activities Terminal" running as root on a Red Hat system. The terminal displays the following commands and their output:

```

[root@servera ~]# mkfs.xfs /dev/dell/lenovo
meta-data=/dev/dell/lenovo isize=512 agcount=4, agsize=20480 blks
          = sectsz=512 attr=2, projid32bit=1
          = crc=1 finobt=1, sparse=1, rmapbt=0
          = reflink=1
data     = bsize=4096 blocks=81920, imaxpct=25
          = sunit=0 swidth=0 blks
naming   =version 2 bsize=4096 ascii-ci=0, ftype=1
log      =internal log bsize=4096 blocks=1368, version=2
          = sectsz=512 sunit=0 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
[root@servera ~]# mkdir /mnt/extend
[root@servera ~]# mount /dev/dell/lenovo /mnt/extend
[root@servera ~]#

```

command: `mkfs.xfs /dev/dell/lenovo`

This command creates an XFS filesystem on the logical volume **/dev/dell/lenovo**. Replace "dell" with the actual name of your Volume Group and "lenovo" with the actual name of your logical volume.

command: `mkdir /mnt/extend`

This command creates a directory named "extend" in the "/mnt" directory. You can replace "/mnt/extend" with the desired path for your mount point.

command: `mount /dev/dell/lenovo /mnt/extend`

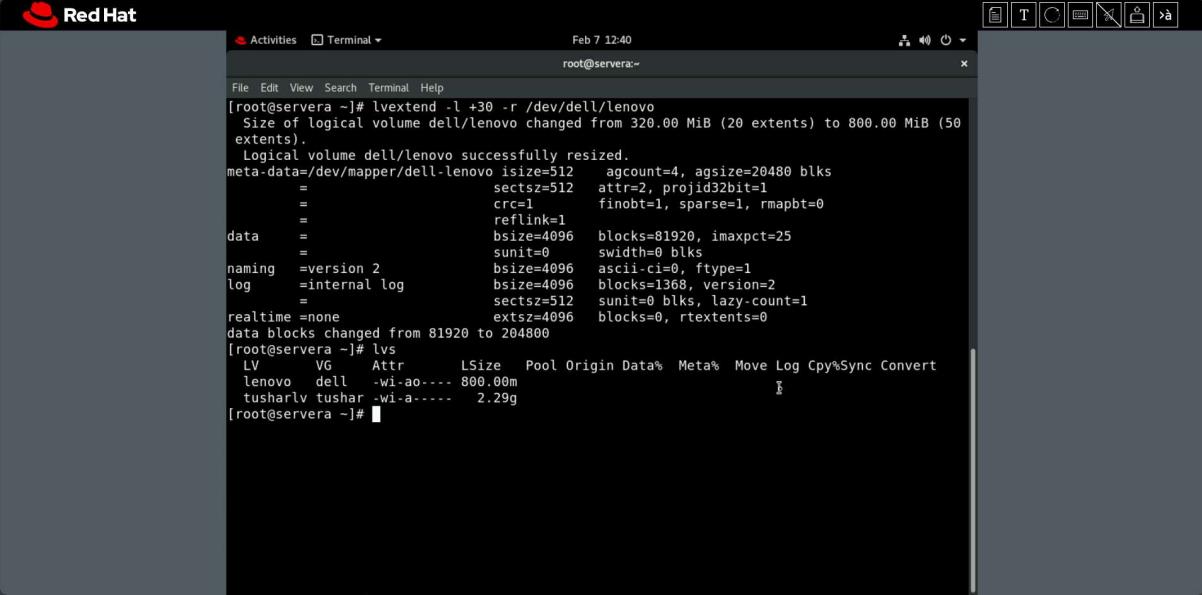
This command mounts the logical volume **/dev/dell/lenovo** to the directory "/mnt/extend". Now, the contents of the logical volume are accessible through the "/mnt/extend" directory.

➤ **Add an entry in fstab for turning its usage on after reboot.**

command: vim /etc/fstab

5. Set the size of the logical volume to 50PE under dell volume group.

- » **Size to increase = Desirable size - Current size (for PEs)**
- » **Here, size to increase = 50 PEs - 20 PEs = 30 PEs**



```
[root@servera ~]# lvextend -l +30 -r /dev/dell/lenovo
  Size of logical volume dell/lenovo changed from 320.00 MiB (20 extents) to 800.00 MiB (50 extents).
  Logical volume dell/lenovo successfully resized.
meta-data=/dev/mapper/dell-lenovo isize=512    agcount=4, agsize=20480 blks
          =         sectsz=512  attr=2, projid32bit=1
          =             crc=1   finobt=1, sparse=1, rmapbt=0
          =             reflink=1
data     =         bsize=4096  blocks=81920, imaxpct=25
          =             sunit=0  swidth=0 blks
naming   =version 2    bsize=4096  ascii-ci=0, ftype=1
log      =internal log  bsize=4096  blocks=1368, version=2
          =             sectsz=512  sunit=0 blks, lazy-count=1
realtime =none        extsz=4096  blocks=0, rtextents=0
data blocks changed from 81920 to 204800
[root@servera ~]# lvs
  LV   VG Attr  LSize  Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  lenovo dell -wi-ao---- 800.00m
  tusharlv tushar -wi-a----- 2.29g
[root@servera ~]#
```

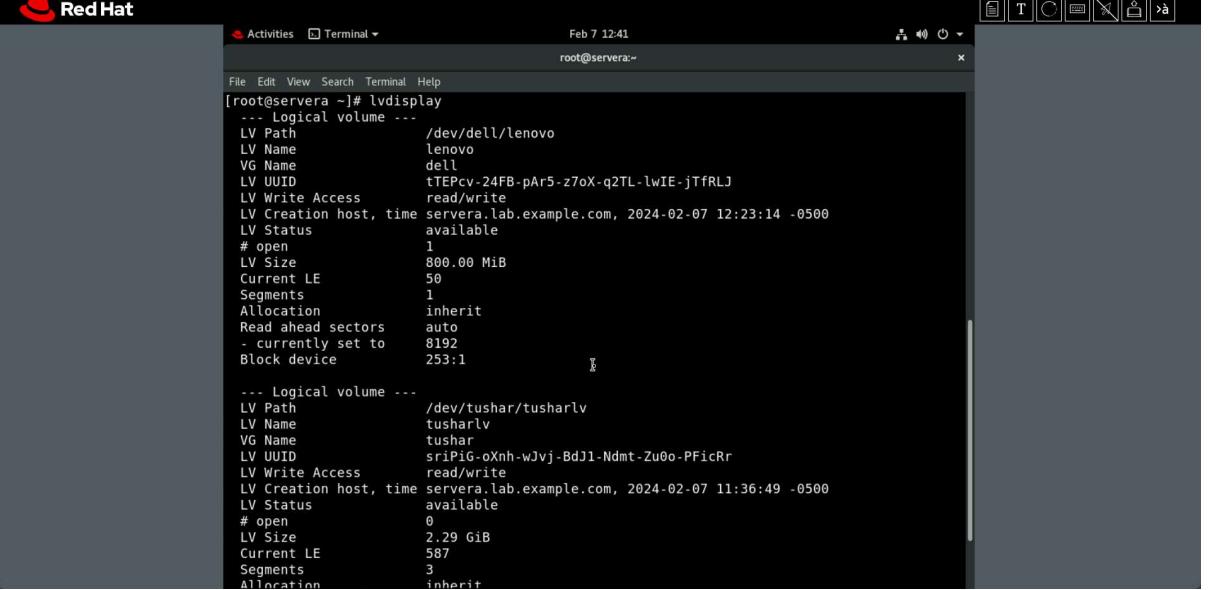
command: lvextend -l +30 -r /dev/dell/lenovo

the command is extending the size of the logical volume **/dev/dell/lenovo** by adding 30 logical extents, and it automatically resizes the filesystem to make use of the additional space

command: lvs

The **lvs** command is used in Linux to display information about Logical Volumes (LVs) within the Logical Volume Manager (LVM) system. This command provides a summary of various attributes and properties of the logical volumes in the specified Volume Group or across all Volume Groups.

» Now, check if the changes are successful



The screenshot shows a terminal window titled "Red Hat" with the title bar "Activities Terminal". The window is running as root on a server named "servera". The command "lvdisplay" is being run, displaying detailed information about two logical volumes: "lenovo" and "tusharlv".

```
[root@servera ~]# lvdisplay
  ... Logical volume ...
  LV Path          /dev/dell/lenovo
  LV Name          lenovo
  VG Name          dell
  LV UUID          tTEPcv-24FB-pAr5-z7oX-q2TL-lwIE-jTfRLJ
  LV Write Access  read/write
  LV Creation host, time servera.lab.example.com, 2024-02-07 12:23:14 -0500
  LV Status        available
  # open           1
  LV Size          800.00 MiB
  Current LE       50
  Segments         1
  Allocation       inherit
  Read ahead sectors auto
  - currently set to 8192
  Block device    253:1
  ...
  ... Logical volume ...
  LV Path          /dev/tushar/tusharlv
  LV Name          tusharlv
  VG Name          tushar
  LV UUID          sriPIG-oXnh-wJvj-BdJl-Ndmt-Zu0o-PFicRr
  LV Write Access  read/write
  LV Creation host, time servera.lab.example.com, 2024-02-07 11:36:49 -0500
  LV Status        available
  # open           0
  LV Size          2.29 GiB
  Current LE       587
  Segments         3
  Allocation       inherit
```

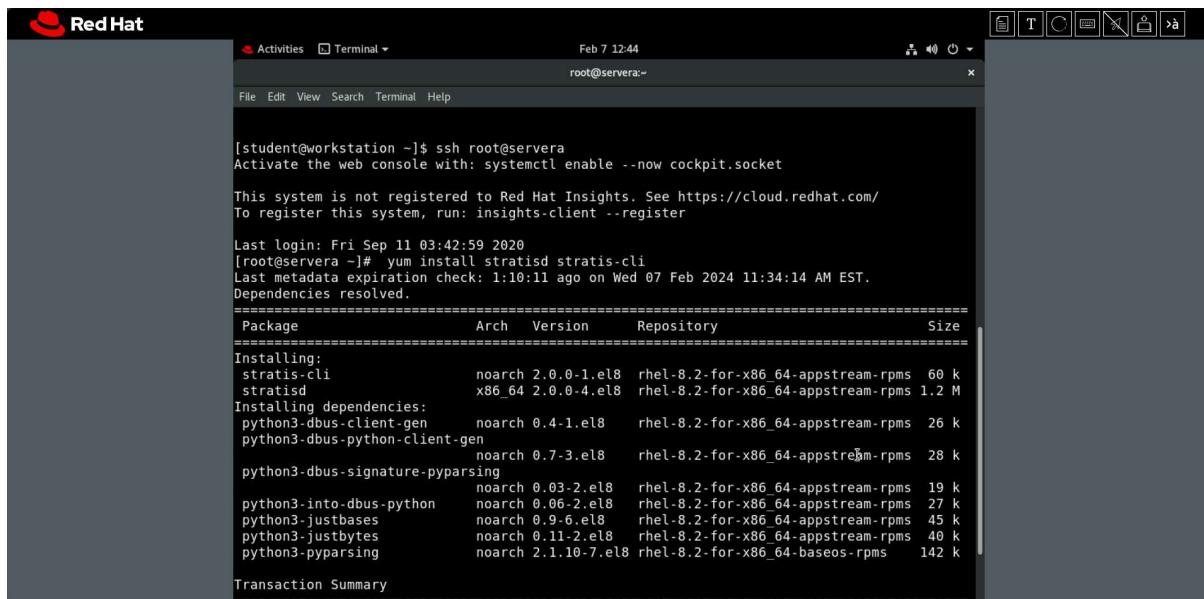
command: `lvdisplay`

- This command is used to display information about Logical Volumes.

6. Demonstrate how to manage the storage using Stratis (Perform the Guided exercise and Final lab of chapter 8).

➡ Guided Exercise-1: Managing Layered Storage with Stratis

- » Login with root user in servera from workstation via ssh and install stratisd and stratis-cli packages using yum package manager.**



The screenshot shows a Red Hat terminal window with the following output:

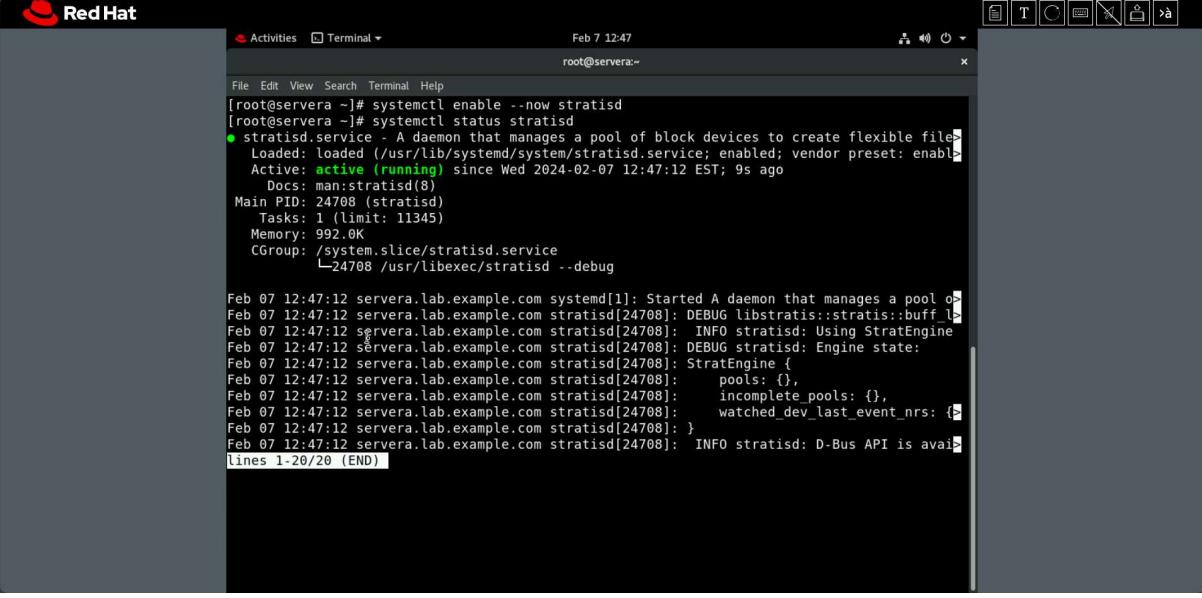
```
[student@workstation ~]$ ssh root@servera
Activate the web console with: systemctl enable --now cockpit.socket
This system is not registered to Red Hat Insights. See https://cloud.redhat.com/
To register this system, run: insights-client --register
Last login: Fri Sep 11 03:42:59 2020
[root@servera ~]# yum install stratisd stratis-cli
Last metadata expiration check: 1:10:11 ago on Wed 07 Feb 2024 11:34:14 AM EST.
Dependencies resolved.
=====
 Package           Arch   Version      Repository      Size
=====
Installing:
 stratisd          noarch 2.0.0-1.el8  rhel-8.2-for-x86_64-appstream-rpms 60 k
 stratisd          x86_64 2.0.0-4.el8  rhel-8.2-for-x86_64-appstream-rpms 1.2 M
Installing dependencies:
 python3-dbus-client-gen    noarch 0.4-1.el8  rhel-8.2-for-x86_64-appstream-rpms 26 k
 python3-dbus-python-client-gen noarch 0.7-3.el8  rhel-8.2-for-x86_64-appstream-rpms 28 k
 python3-dbus-signature-pyparsing noarch 0.03-2.el8  rhel-8.2-for-x86_64-appstream-rpms 19 k
 python3-into-dbus-python    noarch 0.06-2.el8  rhel-8.2-for-x86_64-appstream-rpms 27 k
 python3-justbases        noarch 0.9-6.el8  rhel-8.2-for-x86_64-appstream-rpms 45 k
 python3-justbytes         noarch 0.11-2.el8  rhel-8.2-for-x86_64-appstream-rpms 40 k
 python3-pyparsing          noarch 2.1.10-7.el8  rhel-8.2-for-x86_64-baseos-rpms 142 k
Transaction Summary
=====
```

commands :

`ssh root@servera`

`yum install stratisd stratis-cli`

» **Enable the stratis daemon service using systemctl command and check if it is running successfully.**



```
[root@servera ~]# systemctl enable --now stratisd
[root@servera ~]# systemctl status stratisd
● stratisd.service - A daemon that manages a pool of block devices to create flexible file
  Loaded: loaded (/usr/lib/systemd/system/stratisd.service; enabled; vendor preset: enable
  Active: active (running) since Wed 2024-02-07 12:47:12 EST; 9s ago
    Docs: man:stratisd(8)
   Main PID: 24708 (stratisd)
      Tasks: 1 (limit: 11345)
     Memory: 992.0K
       CGroup: /system.slice/stratisd.service
               └─24708 /usr/libexec/stratisd --debug

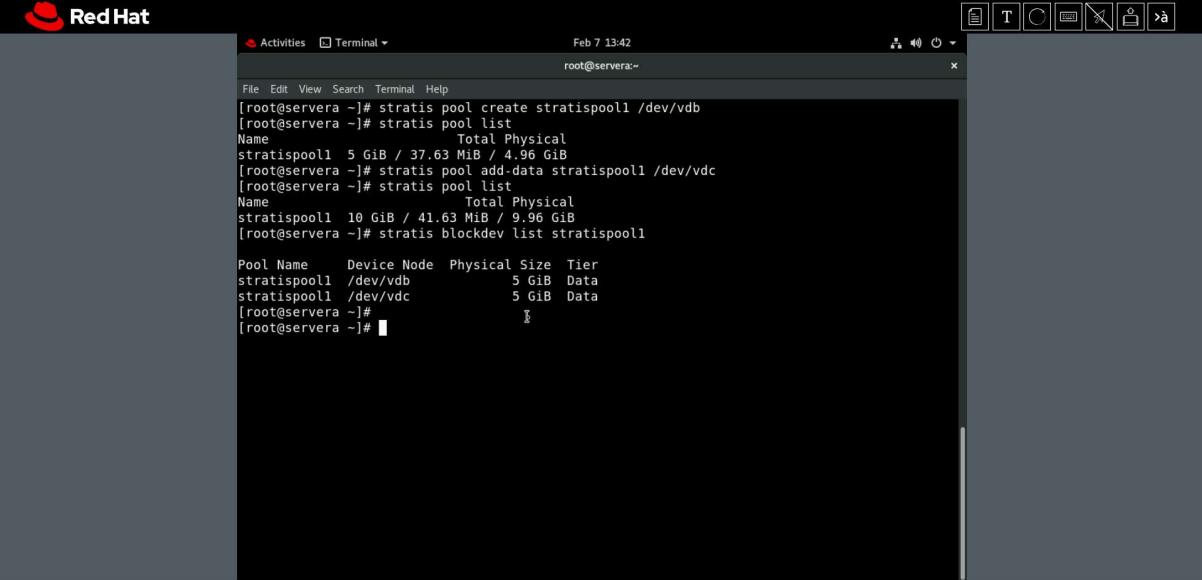
Feb 07 12:47:12 servera.lab.example.com systemd[1]: Started A daemon that manages a pool o>
Feb 07 12:47:12 servera.lab.example.com stratisd[24708]: DEBUG libstratis::stratis::buff_>
Feb 07 12:47:12 servera.lab.example.com stratisd[24708]: INFO stratisd: Using StratEngine
Feb 07 12:47:12 servera.lab.example.com stratisd[24708]: DEBUG stratisd: Engine state:
Feb 07 12:47:12 servera.lab.example.com stratisd[24708]: StratEngine {
Feb 07 12:47:12 servera.lab.example.com stratisd[24708]:   pools: {},
Feb 07 12:47:12 servera.lab.example.com stratisd[24708]:   incomplete_pools: {},
Feb 07 12:47:12 servera.lab.example.com stratisd[24708]:   watched_dev_last_event_nrs: {}
Feb 07 12:47:12 servera.lab.example.com stratisd[24708]: }
Feb 07 12:47:12 servera.lab.example.com stratisd[24708]: INFO stratisd: D-Bus API is avail>
lines 1-20/20 (END)
```

commands :

systemctl enable --now stratisd

systemctl status stratisd

» **Create stratispool1 named stratis pool with vdb block device included initially, check its details and add vdc block device also afterwards. Now, details can be verified using pool list option and block devices' details using blockdev list option.**



```

Red Hat Activities Terminal Feb 7 13:42 root@servera:~#
[root@servera ~]# stratis pool create stratispool1 /dev/vdb
[root@servera ~]# stratis pool list
Name          Total Physical
stratispool1  5 GiB / 37.63 MiB / 4.96 GiB
[root@servera ~]# stratis pool add-data stratispool1 /dev/vdc
[root@servera ~]# stratis pool list
Name          Total Physical
stratispool1  10 GiB / 41.63 MiB / 9.96 GiB
[root@servera ~]# stratis blockdev list stratispool1
Pool Name  Device Node  Physical Size  Tier
stratispool1 /dev/vdb      5 GiB Data
stratispool1 /dev/vdc      5 GiB Data
[root@servera ~]# 
[root@servera ~]# 

```

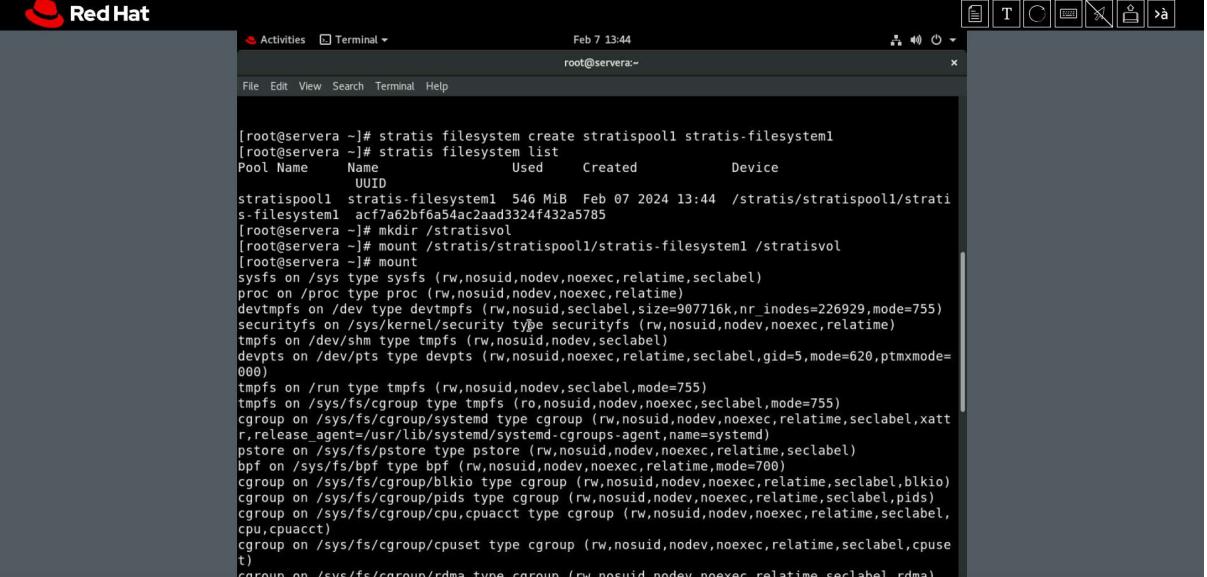
commands :

```

stratis pool create stratispool1 /dev/vdb
stratis pool list
stratis pool add-data stratispool1 /dev/vdc
stratis pool list
stratis blockdev list stratispool1

```

» Now, assign the filesystem to stratispool1 of name stratis-filesystem1 and check its details, and then after creating a directory to mount(/stratisvol here), mount it there.

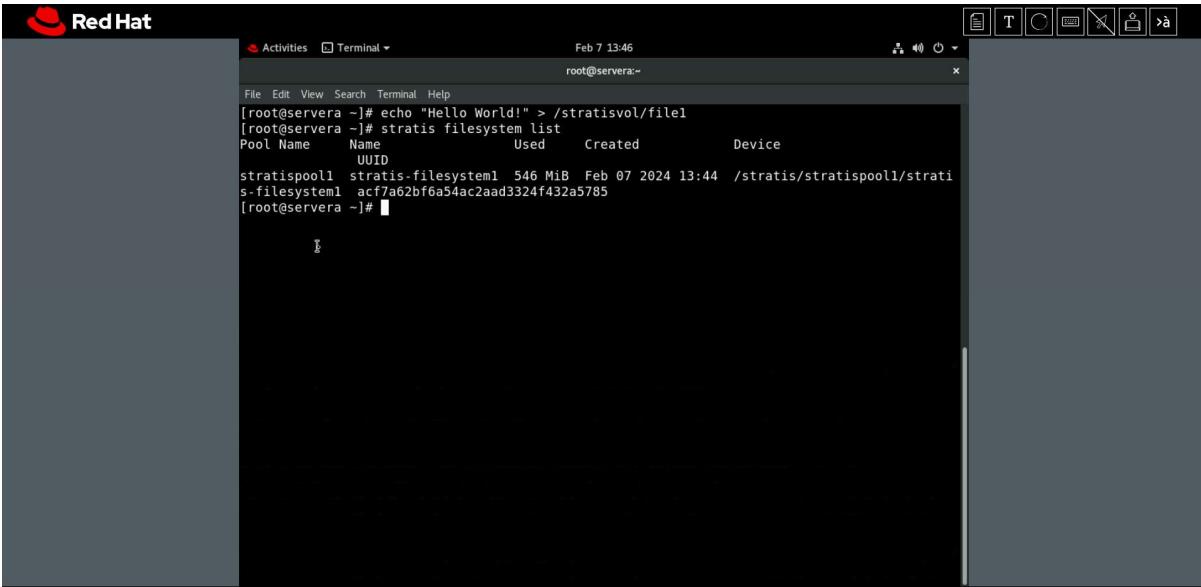


```
[root@servera -]# stratis filesystem create stratispool1 stratis-filesystem1
[root@servera -]# stratis filesystem list
Pool Name          Name           Used       Created      Device
          UUID
stratispool1 stratis-filesystem1 546 MiB Feb 07 2024 13:44 /stratis/stratispool1/stratis-filesystem1
[root@servera -]# mkdir /stratisvol
[root@servera -]# mount /stratis/stratispool1/stratis-filesystem1 /stratisvol
[root@servera -]# mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=907716k,nr_inodes=226929,mode=755)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,mode=755)
tmpfs on /sys/fs/cgroup type tmpfs (ro,nosuid,nodev,noexec,seclabel,mode=755)
cgroup on /sys/fs/cgroup/systemd type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,xattr,release_agent=/usr/lib/systemd/systemd-cgroups-agent,name=systemd)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime,seclabel)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
cgroup on /sys/fs/cgroup/blkio type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,blkio)
cgroup on /sys/fs/cgroup/pids type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,pids)
cgroup on /sys/fs/cgroup/cpu,cpuacct type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,cpu,cpuacct)
cgroup on /sys/fs/cgroup/cpuset type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,cpuset)
cgroup on /sys/fs/cgroup/rdma type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,rdma)
```

commands :

stratis filesystem create stratispool1 stratis-filesystem1
stratis filesystem list
mkdir /stratisvol
mount /stratis/stratispool1/stratis-filesystem1 /stratisvol
mount

» **Append Hello World text to a new file- file1 inside the /stratisvol directory and check the details of filesystem.**



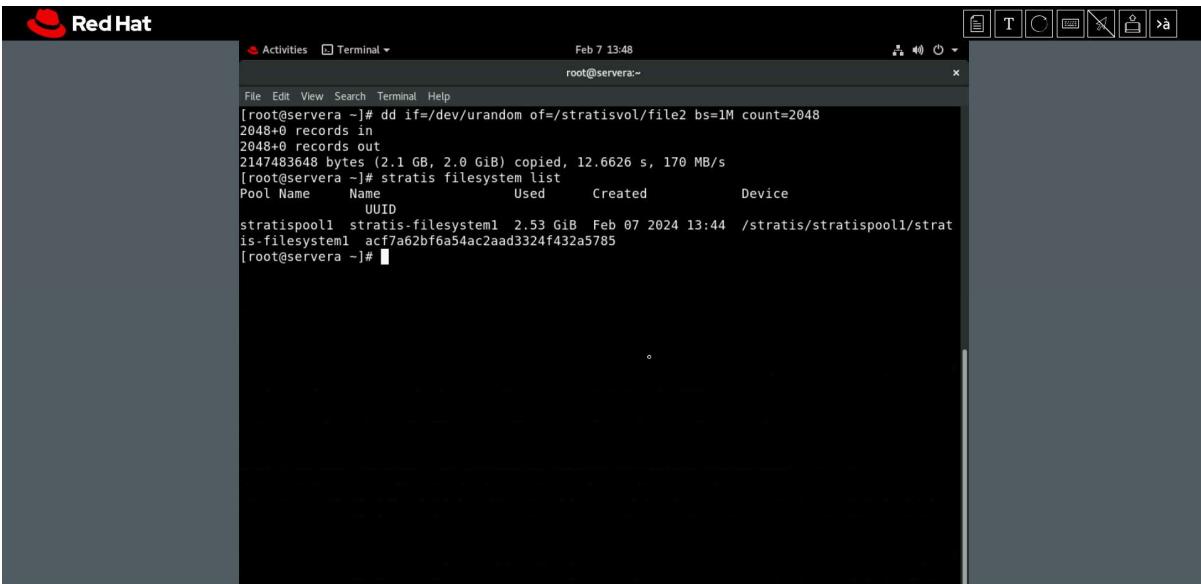
```
[root@servera ~]# echo "Hello World!" > /stratisvol/file1
[root@servera ~]# stratis filesystem list
Pool Name      Name          Used       Created      Device
           UUID
stratispool1  stratis-filesystem1  546 MiB  Feb 07 2024 13:44  /stratis/stratispool1/strati
s-filesystem1  acf7a62bf6a54ac2aad3324f432a5785
[root@servera ~]#
```

commands :

`echo "Hello World!" > /stratisvol/file1`

`stratis filesystem list`

» **Now, create a 2Gigabytes file of name file2 using dd command in that stratis filesystem directory.**



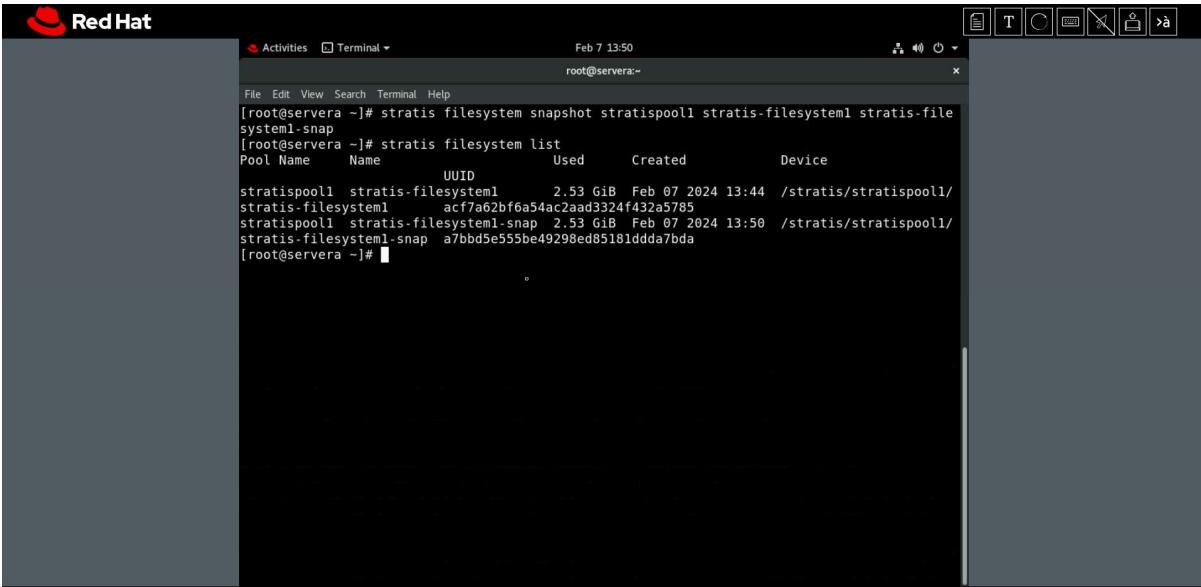
```
[root@servera ~]# dd if=/dev/urandom of=/stratisvol/file2 bs=1M count=2048
2048+0 records in
2048+0 records out
2147483648 bytes (2.1 GB, 2.0 GiB) copied, 12.6626 s, 170 MB/s
[root@servera ~]# stratis filesystem list
Pool Name      Name          Used       Created      Device
           UUID
stratispool1  stratis-filesystem1  2.53 GiB  Feb 07 2024 13:44  /stratis/stratispool1/strati
s-filesystem1  acf7a62bf6a54ac2aad3324f432a5785
[root@servera ~]#
```

commands :

`dd if=/dev/urandom of=/stratisvol/file2 bs=1M count=2048`

`stratis filesystem list`

» **Take a snapshot of the current stratis filesystem, which will create another filesystem as a backup of the current one.**



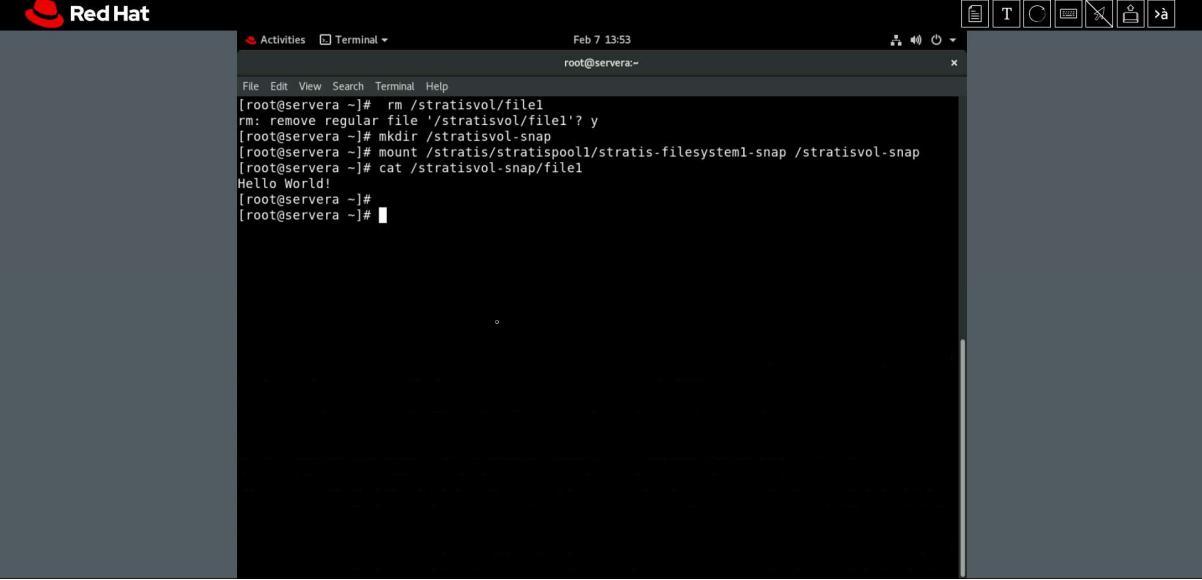
The screenshot shows a terminal window on a Red Hat desktop environment. The terminal output is as follows:

```
[root@servera ~]# stratis filesystem snapshot stratispool1 stratis-filesystem1 stratis-filesystem1-snap
[root@servera ~]# stratis filesystem list
Pool Name      Name          Used      Created      Device
              UUID
stratispool1  stratis-filesystem1    2.53 GiB Feb 07 2024 13:44  /stratis/stratispool1/
stratis-filesystem1-acf7a62bf6a54ac2aad3324f432a5785
stratispool1  stratis-filesystem1-snap  2.53 GiB Feb 07 2024 13:50  /stratis/stratispool1/
stratis-filesystem1-snap-a7bbd5e555be49298ed85181ddda7bda
[root@servera ~]#
```

commands :

**stratis filesystem snapshot stratispool1 stratis-filesystem1
stratis-filesystem1-snap**
stratis filesystem list

» **Now, remove file1 in /stratisvol and mount the snapshot to a different location and check if the file1 is there in it with required content.**



```

Red Hat Activities Terminal Feb 7 13:53 root@servera:~ 
File Edit View Search Terminal Help
[root@servera -]# rm /stratisvol/file1
rm: remove regular file '/stratisvol/file1'? y
[root@servera -]# mkdir /stratisvol-snap
[root@servera -]# mount /stratis/stratispool1/stratis-filesystem1-snap /stratisvol-snap
[root@servera -]# cat /stratisvol-snap/file1
Hello World!
[root@servera -]#
[root@servera -]# 

```

commands :

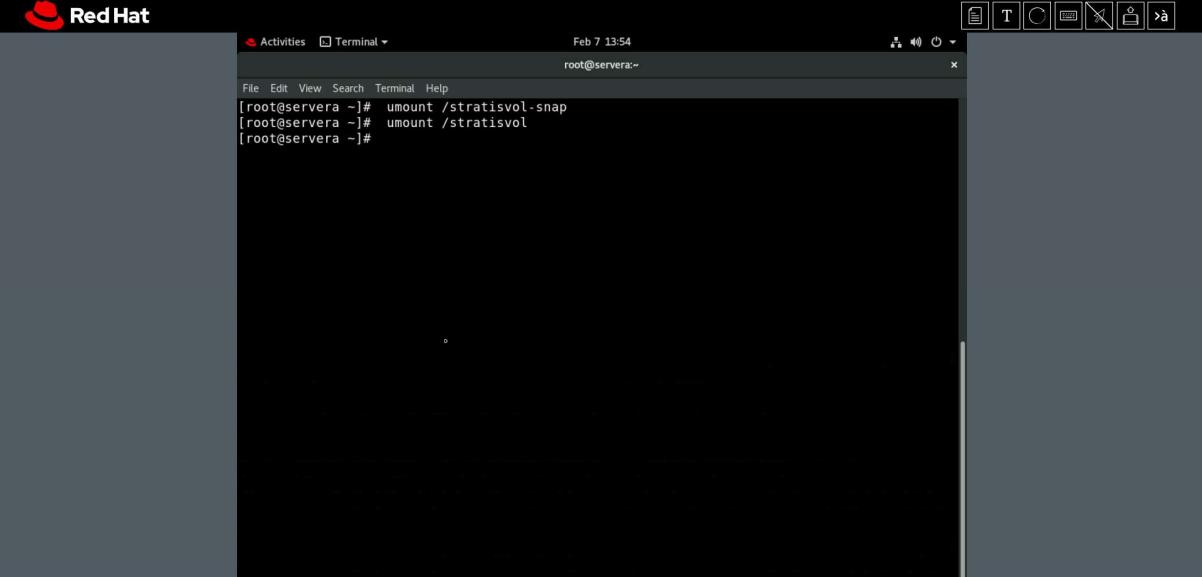
rm /stratisvol/file1

mkdir /stratisvol-snap

mount /stratis/stratispool1/stratis-filesystem1-snap/stratisvolsnap

cat /stratisvol-snap/file1

» After finishing, unmount the both stratis filesystems.



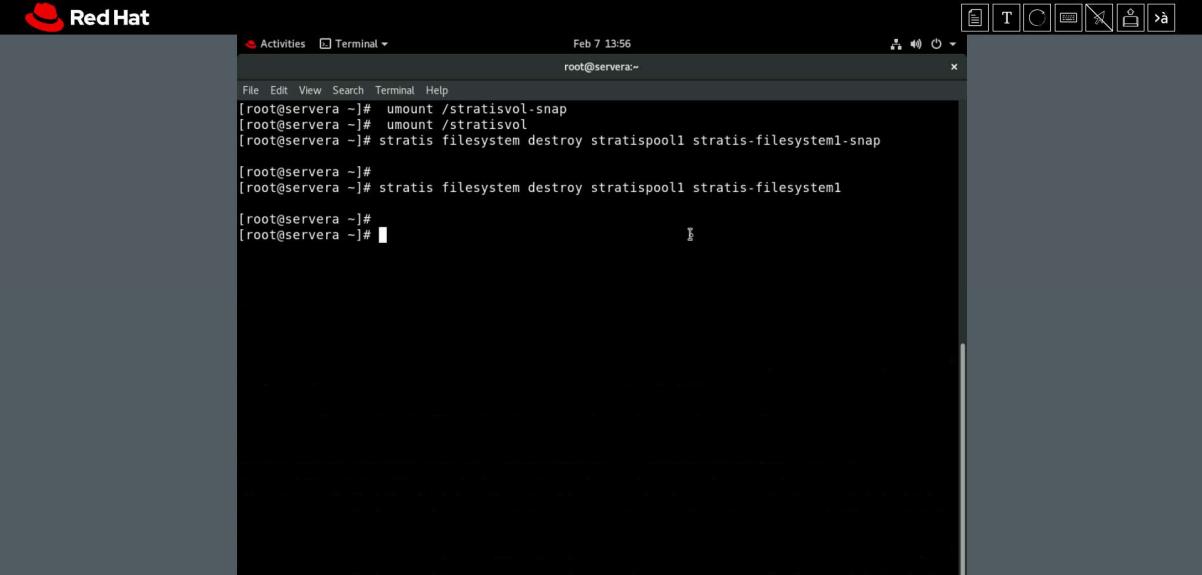
```
[root@servera ~]# umount /stratisvol-snap
[root@servera ~]# umount /stratisvol
[root@servera ~]#
```

commands :

umount /stratisvol-snap

umount /stratisvol

» Now, delete both the stratis filesystems.



```
[root@servera ~]# umount /stratisvol-snap
[root@servera ~]# umount /stratisvol
[root@servera ~]# stratis filesystem destroy stratispool1 stratis-filesystem1-snap
[root@servera ~]#
[root@servera ~]# stratis filesystem destroy stratispool1 stratis-filesystem1
[root@servera ~]#
[root@servera ~]#
```

commands :

stratis filesystem destroy stratispool1 stratis-filesystem1-snap

stratis filesystem destroy stratispool1 stratis-filesystem1

» Exit and finish the lab.

```
[root@servera ~]# exit
logout
Connection to servera closed.
[student@workstation ~]$ lab advstorage-stratis finsih
Missing or unrecognized command - finsih
This script controls the setup and grading of this lab.
Usage: lab advstorage-stratis COMMAND
      lab advstorage-stratis -h|--help

COMMAND is one of: start finish

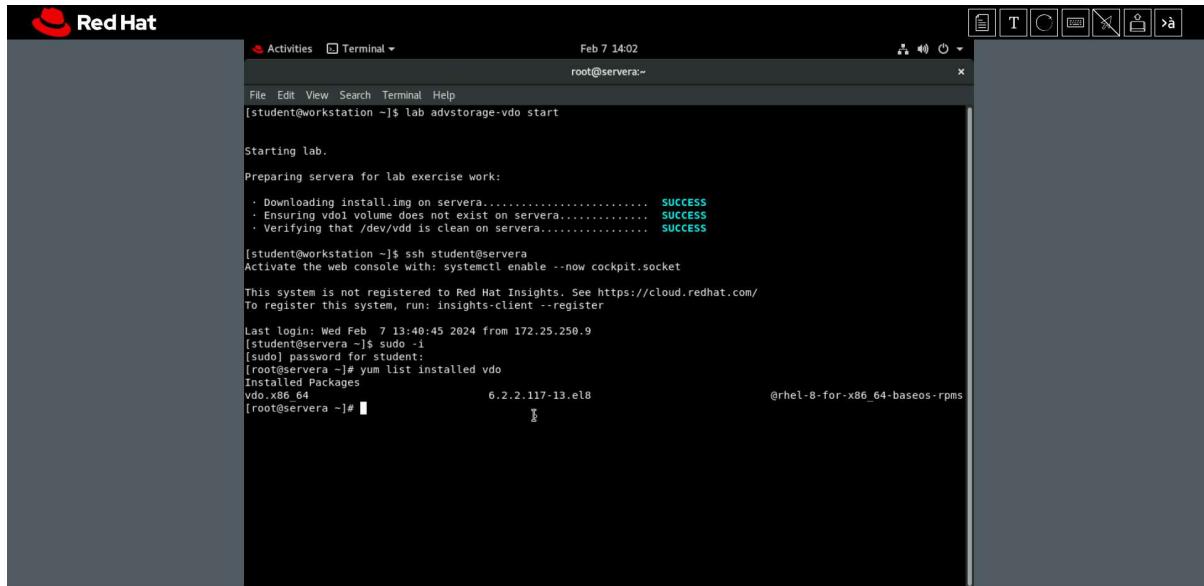
[student@workstation ~]$ lab advstorage-stratis finish >
Completing the lab on servera:
  • Removing stratis-filesystem1-snap from on servera..... SUCCESS
  • Removing stratis-filesystem1 from servera..... SUCCESS
  • Removing stratispool1 from servera..... SUCCESS
  • Removing /stratisvol from servera..... SUCCESS
  • Removing /stratisvol-snap from servera..... SUCCESS
  • Ensuring clean additional disks on servera..... SUCCESS

Lab finished.

[student@workstation ~]$
```

☞ **Guided Exercise-2 : Compressing and Deduplicating Storage with VDO :**

- » **Start the lab and login to servera from workstation via ssh.**
Check the existence of the installed package vdo using yum.



```

Red Hat Activities Terminal Feb 7 14:02
root@servera:~ x
File Edit View Search Terminal Help
[student@workstation ~]$ lab advstorage-vdo start
Starting lab.
Preparing servera for lab exercise work:
  . Downloading install.img on servera..... SUCCESS
  . Ensuring vd0l volume does not exist on servera..... SUCCESS
  . Verifying that /dev/vdd is clean on servera..... SUCCESS
[student@workstation ~]$ ssh student@servera
Activate the web console with: systemctl enable --now cockpit.socket
This system is not registered to Red Hat Insights. See https://cloud.redhat.com/
To register this system, run: insights-client --register
Last login: Wed Feb 7 13:40:45 2024 from 172.25.250.9
[student@servera ~]$ sudo -i
[sudo] password for student:
[root@servera ~]# yum list installed vdo
Installed Packages
vdo.x86_64           6.2.2.117-13.el8          @rhel-8-for-x86_64-baseos-rpms
[root@servera ~]# 

```

commands :

ssh root@servera
yum list installed vdo

» **Create a vdo1 named VDO with 50G logical size and check the same.**

```
[root@servera ~]# vdo create --name=vdo1 --device=/dev/vdd --vdoLogicalSize=50G
Creating VDO vdo1
    The VDO volume can address 2 GB in 1 data slab.
    It can grow to address at most 16 TB of physical storage in 8192 slabs.
    If a larger maximum size might be needed, use bigger slabs.
Starting VDO vdo1
Starting compression on VDO vdo1
VDO instance 0 volume is ready at /dev/mapper/vdo1
[root@servera ~]# vdo list
vdo1
[root@servera ~]#
```

commands :

```
vdo create --name=vdo1 --device=/dev/vdd --vdoLogicalSize=50G
vdo list
```

» **Check if deduplication and compression features of the VDO are enabled using vdo status command's stdout piped to stdin of grep command.**

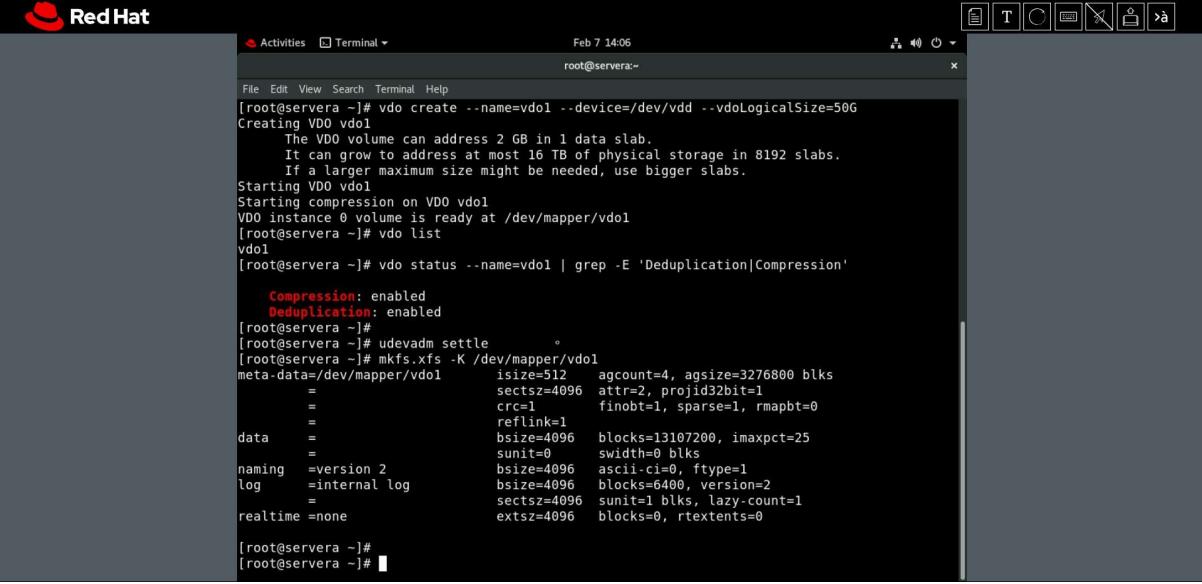
```
[root@servera ~]# vdo create --name=vdo1 --device=/dev/vdd --vdoLogicalSize=50G
Creating VDO vdo1
    The VDO volume can address 2 GB in 1 data slab.
    It can grow to address at most 16 TB of physical storage in 8192 slabs.
    If a larger maximum size might be needed, use bigger slabs.
Starting VDO vdo1
Starting compression on VDO vdo1
VDO instance 0 volume is ready at /dev/mapper/vdo1
[root@servera ~]# vdo list
vdo1
[root@servera ~]# vdo status --name=vdo1 | grep -E 'Deduplication|Compression'

  Compression: enabled
  Deduplication: enabled
[root@servera ~]#
[root@servera ~]#
```

commands :

```
vdo status --name=vdo1 | grep -E 'Deduplication|Compression'
```

» **Ensure the creation of file via **udevadm settle** and assign the **xfs** filesystem to the newly created VDO using **mkfs**.**



```

Activities Terminal Feb 7 14:06
root@servera:~ x
File Edit View Search Terminal Help
[root@servera ~]# vdo create --name=vdo1 --device=/dev/vdd --vdoLogicalSize=50G
Creating VDO vdo1
    The VDO volume can address 2 GB in 1 data slab.
    It can grow to address at most 16 TB of physical storage in 8192 slabs.
    If a larger maximum size might be needed, use bigger slabs.
Starting VDO vdo1
Starting compression on VDO vdo1
VDO instance 0 volume is ready at /dev/mapper/vdo1
[root@servera ~]# vdo list
vdo1
[root@servera ~]# vdo status --name=vdo1 | grep -E 'Deduplication|Compression'
    Compression: enabled
    Deduplication: enabled
[root@servera ~]#
[root@servera ~]# udevadm settle .
[root@servera ~]# mkfs.xfs -K /dev/mapper/vdo1
meta-data=/dev/mapper/vdo1 isize=512 agcount=4, agsize=3276800 blks
          = sectsz=4096 attr=2, projid32bit=1
          = crc=1         finobt=1, sparse=1, rmapbt=0
data     = reflink=1
          = bsize=4096   blocks=13107200, imaxpct=25
          = sunit=0      swidth=0 blks
naming   =version 2
log      =internal log
          = bsize=4096   ascii-ci=0, ftype=1
          = sectsz=4096  blocks=6400, version=2
          = sunit=1 blks, lazy-count=1
realtime =none
          = extsz=4096  blocks=0, rtextents=0

[root@servera ~]#
[root@servera ~]# 

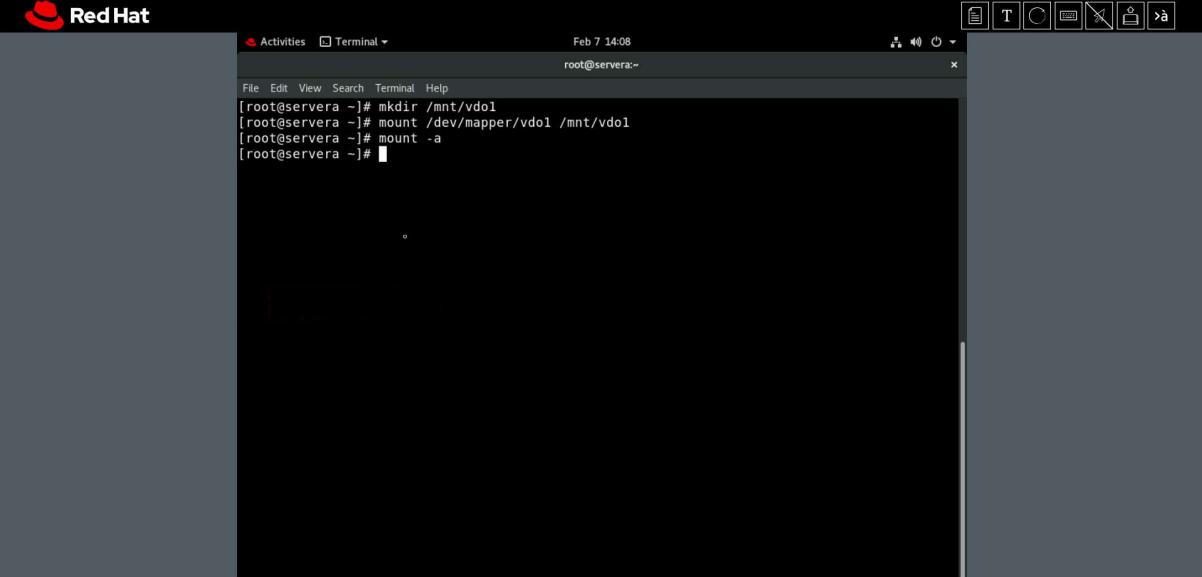
```

commands :

`udevadm settle`

`mkfs.xfs -K /dev/mapper/vdo1`

» Create a directory and mount the vdo volume to it.

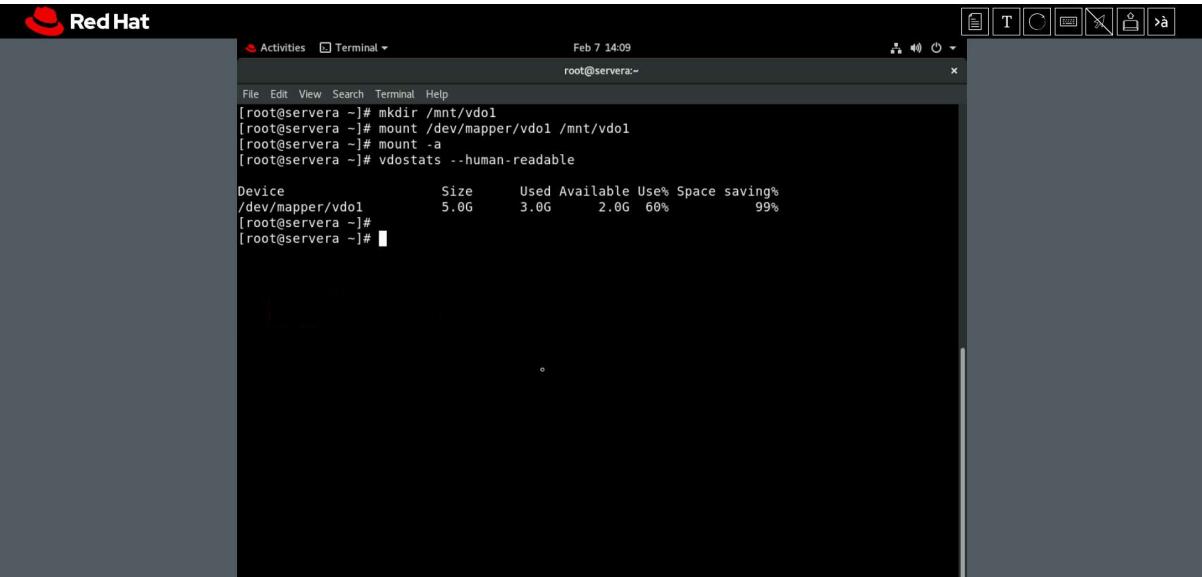


```
[root@servera ~]# mkdir /mnt/vdo1
[root@servera ~]# mount /dev/mapper/vdo1 /mnt/vdo1
[root@servera ~]# mount -a
[root@servera ~]#
```

commands :

```
mkdir /mnt/vdo1
mount /dev/mapper/vdo1 /mnt/vdo1
mount -a
```

» Check the human readable details of the cdo and create 2 copies of /root/install.img file there. And check the increase in volume size.



```
[root@servera ~]# mkdir /mnt/vdo1
[root@servera ~]# mount /dev/mapper/vdo1 /mnt/vdo1
[root@servera ~]# mount -a
[root@servera ~]# vdostats --human-readable
Device           Size     Used Available Use% Space saving%
/dev/mapper/vdo1    5.0G    3.0G     2.0G  60%      99%
[root@servera ~]#
[root@servera ~]#
```

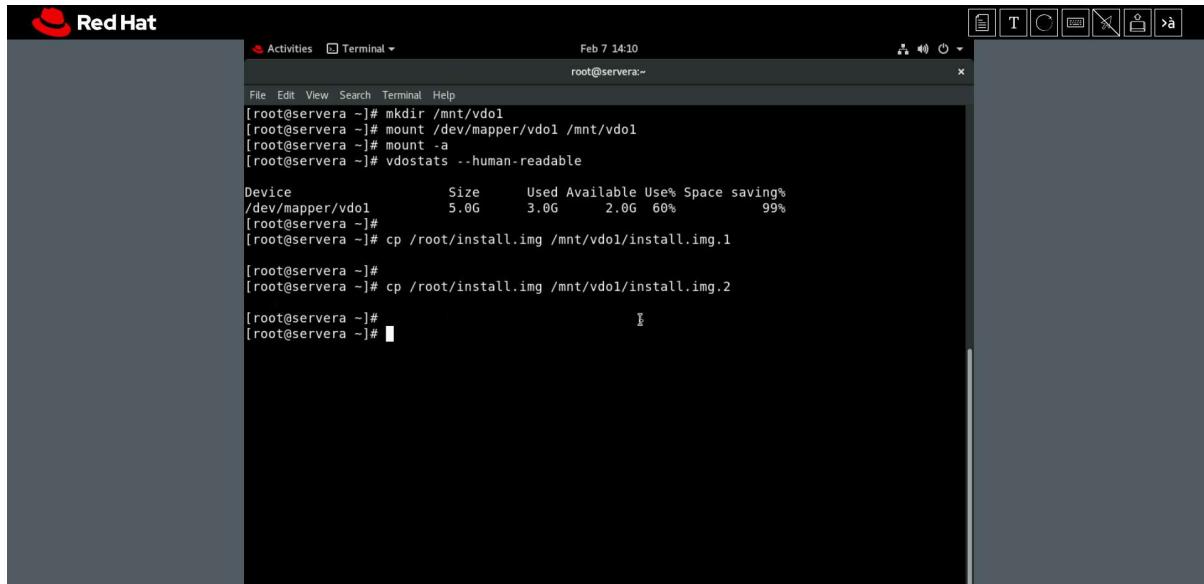
commands :

```
vdostats --human-readable
```

```
[root@servera ~]# cp /root/install.img /mnt/vd01/install.img.1
cp: overwrite '/mnt/vd01/install.img.1'? y
[root@servera ~]# vdostats --human-readable
Device           Size   Used Available Use% Space saving%
/dev/mapper/vd01      5.0G    3.5G     1.5G  70%       67%
[root@servera ~]# clear
```

commands :

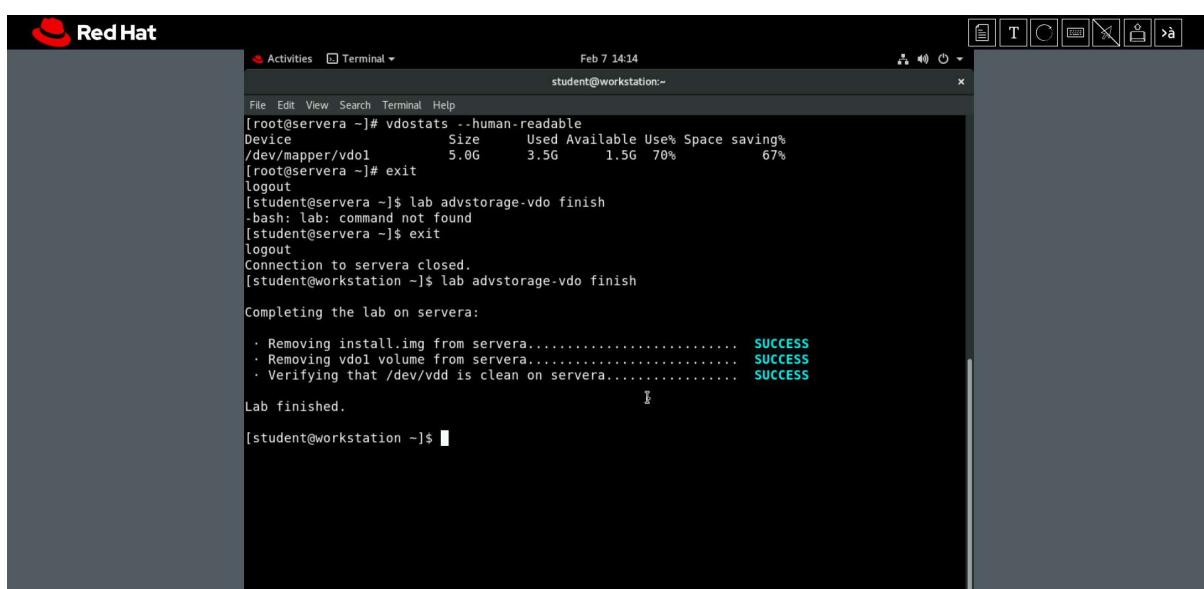
cp /root/install.img /mnt/vd01/install.img.1



commands :

cp /root/install.img /mnt/vd01/install.img.2

» Now, exit and end the lab.



→ **Lab : Implementing Advanced Storage Features (using stratis) :**

» **Login to serverb as student user from workstation via ssh and switch to root user. Install the stratisd and stratis-cli packages using yum package manager.**

```

Red Hat
Activities Terminal Feb 7 14:19
root@serverb:~>

File Edit View Search Terminal Help
· Ensuring labvdo volume does not exist on serverb..... SUCCESS
· Ensuring clean additional disks on serverb..... SUCCESS

[student@workstation ~]$ ssh student@serverb
Activate the web console with: systemctl enable --now cockpit.socket

This system is not registered to Red Hat Insights. See https://cloud.redhat.com/
To register this system, run: insights-client --register

Last login: Tue Sep 1 08:19:05 2020 from 172.25.250.9
[student@serverb ~]$ sudo -i
[sudo] password for student:
[root@serverb ~]# yum install stratisd stratis-cli
Last metadata expiration check: 0:33:52 ago on Wed 07 Feb 2024 01:44:45 PM EST.
Dependencies resolved.

=====
Package           Arch   Version      Repository      Size
=====
Installing:
stratis-cli      noarch  2.0.0-1.el8   rhel-8.2-for-x86_64-appstream-rpms  60 k
stratisd         x86_64  2.0.0-4.el8   rhel-8.2-for-x86_64-appstream-rpms  1.2 M
Installing dependencies:
python3-dbus-client-gen  noarch  0.4-1.el8   rhel-8.2-for-x86_64-appstream-rpms  26 k
python3-dbus-python-client-gen  noarch  0.7-3.el8   rhel-8.2-for-x86_64-appstream-rpms  28 k
python3-dbus-signature-pyparsing  noarch  0.03-2.el8   rhel-8.2-for-x86_64-appstream-rpms  19 k
python3-into-dbus-python  noarch  0.06-2.el8   rhel-8.2-for-x86_64-appstream-rpms  27 k
python3-justbytes  noarch  0.9-1.el8   rhel-8.2-for-x86_64-appstream-rpms  45 k
python3-pybytes    noarch  0.11-2.el8   rhel-8.2-for-x86_64-appstream-rpms  46 k
python3-pyparsing   noarch  2.1.10-7.el8  rhel-8.2-for-x86_64-baseos-rpms  142 k

Transaction Summary
=====
Install 9 Packages

Total download size: 1.6 M
Installed size: 5.0 M
Is this ok [y/N]: y

```

commands :

```

ssh root@serverb
sudo -i
yum install stratisd stratis-cli

```

» **Enable the stratis daemon service using systemctl.**

```

[root@serverb ~]#
[root@serverb ~]# systemctl enable --now stratisd
[root@serverb ~]#

```

commands :

```

systemctl enable --now stratisd

```

» **Create a labpool named stratis pool using the block device /dev/vdb and check the status.**

```
[root@serverb ~]# stratis pool create labpool /dev/vdb
[root@serverb ~]# stratis pool list

Name          Total Physical
labpool      5 GiB / 37.63 MiB / 4.96 GiB
[root@serverb ~]#
[root@serverb ~]# █
```

commands :

stratis pool create labpool /dev/vdb
stratis pool list

» **Now, add vdc block device to the pool and check the status.**

```
[root@serverb ~]# stratis pool add-data labpool /dev/vdc
[root@serverb ~]# stratis pool list

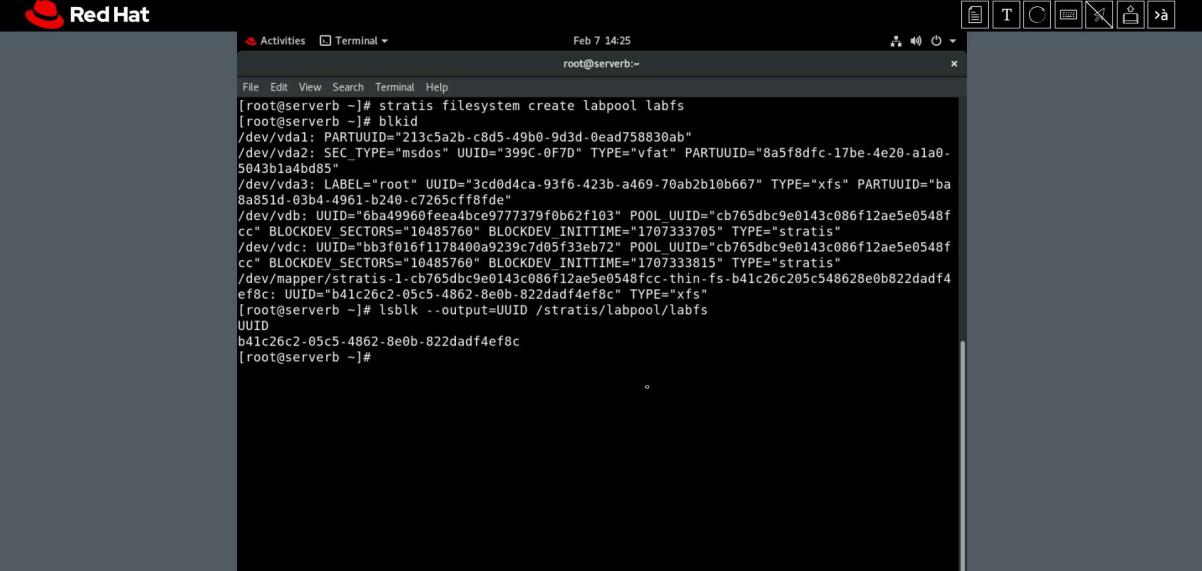
Name          Total Physical
labpool      10 GiB / 41.63 MiB / 9.96 GiB
[root@serverb ~]# stratis blockdev list labpool

Pool Name  Device Node  Physical Size  Tier
labpool    /dev/vdb        5 GiB   Data
labpool    /dev/vdc        5 GiB   Data
[root@serverb ~]#
[root@serverb ~]# █
```

commands :

stratis pool add-data labpool /dev/vdc
stratis pool list
stratis blockdev list labpool

» **Assign and create labfs named filesystem to labpool and get its UUID using blkid or lsblk command.**



```
[root@serverb ~]# stratis filesystem create labpool labfs
[root@serverb ~]# blkid
/dev/vda1: PARTUUID="213c5a2b-c8d5-49b0-9d3d-0ead758830ab"
/dev/vda2: SEC_TYPE="msdos" UUID="399C-0F7D" TYPE="vfat" PARTUUID="8a5f8dfc-17be-4e20-a1a0-5043b1a4bd85"
/dev/vda3: LABEL="root" UUID="3cd0d4ca-93f6-423b-a469-70ab2b10b667" TYPE="xfs" PARTUUID="ba8a851d-03b4-4961-b240-c7265cff8fde"
/dev/vdb: UUID="6ba49960fea4bce9777379f0b62f103" POOL_UUID="cb765dbc9e0143c086f12ae5e0548fcc" BLOCKDEV_SECTORS="10485760" BLOCKDEV_INITTIME="1767333705" TYPE="stratis"
/dev/vdc: UUID="bb3f016f178400a9239c7d05f33eb72" POOL_UUID="cb765dbc9e0143c086f12ae5e0548fcc" BLOCKDEV_SECTORS="10485760" BLOCKDEV_INITTIME="1707333815" TYPE="stratis"
/dev/mapper/stratis-1-cb765dbc9e0143c086f12ae5e0548fcc-thin-fs-b41c26c205c548628e0b822dadf4ef8c: UUID="b41c26c2-05c5-4862-8e0b-822dadf4ef8c" TYPE="xfs"
[root@serverb ~]# lsblk --output=UUID /stratis/labpool/labfs
UUID
b41c26c2-05c5-4862-8e0b-822dadf4ef8c
[root@serverb ~]#
```

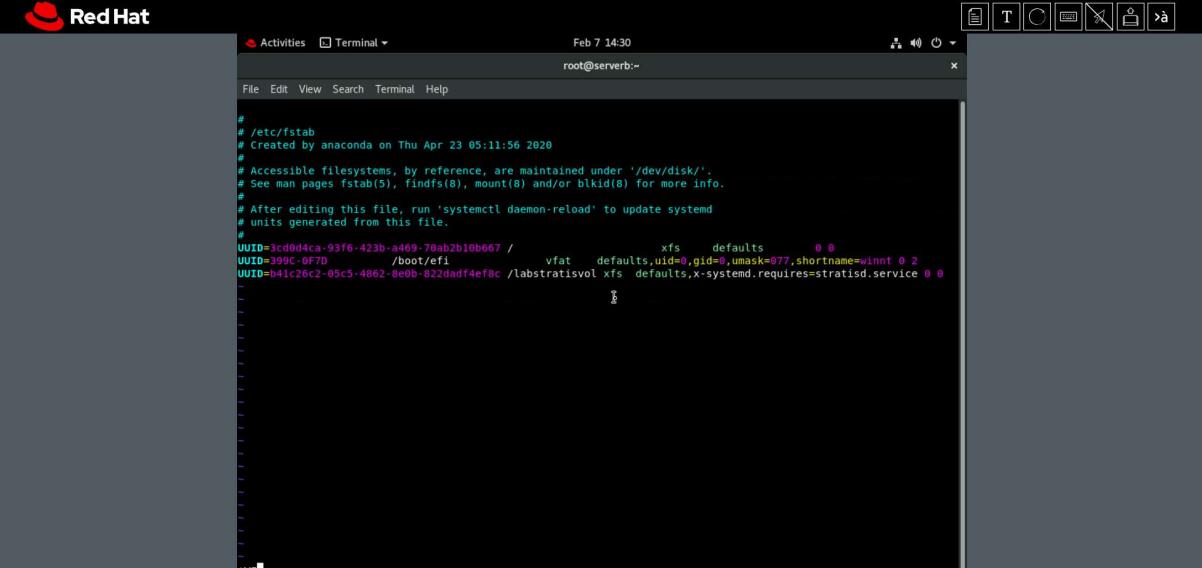
commands :

stratis filesystem create labpool labfs

lsblk --output=UUID /stratis/labpool/labfs

blkid (alternative to lsblk long command)

» **Create its entry in fstab using text editor like vim.**



```
# /etc/fstab
# Created by anaconda on Thu Apr 23 05:11:56 2020
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=3cd0d4ca-93f6-423b-a469-70ab2b10b667 /          xfs    defaults      0 0
UUID=399C-0F7D   /boot/efi   vfat   defaults,uid=0,gid=0,umask=077,shortname=winnt 0 2
UUID=b41c26c2-05c5-4862-8e0b-822dadf4ef8c /labstratisvol xfs  defaults,x-systemd.requires=stratisd.service 0 0
-
```

commands :

vim /etc/fstab

» **Now, create a directory and mount the stratis filesystem there, monut it and create a Hello World text containing file in it.**

```
[root@serverb ~]# mkdir /labstratisvol
[root@serverb ~]# mount /labstratisvol
[root@serverb ~]# echo "Hello World!" > /labstratisvol/labfile1
[root@serverb ~]#
[root@serverb ~]#
```

commands :

```
mkdir /labstratisvol
mount /labstratisvol
echo "Hello World!" > /labstratisvol/labfile1
```

» **Check the status before and after creating a 2GB file in stratis filesystem.**

```
[root@serverb ~]# dd if=/dev/urandom of=/labstratisvol/labfile2 bs=1M count=2048
2048+0 records in
2048+0 records out
2147483648 bytes (2.1 GB, 2.0 GiB) copied, 12.2015 s, 176 MB/s
[root@serverb ~]# stratis filesystem list
Pool Name    Name    Used    Created        Device          UUID
labpool      labfs   2.68 GiB Feb 07 2024 14:25  /stratis/labpool/labfs  b41c26c205c548628e0b822dadf4ef8c
[root@serverb ~]#
```

commands :

```
stratis filesystem list
dd if=/dev/urandom of=/labstratisvol/labfile2 bs=1M count=2048
stratis filesystem list
```

» **Create a snapshot of the current stratis filesystem and check the status.**

```
[root@serverb ~]# stratis filesystem snapshot labpool labfs labfs-snap
[root@serverb ~]# stratis filesystem list
Pool Name    Name    Used    Created        Device          UUID
labpool      labfs   2.68 GiB Feb 07 2024 14:25  /stratis/labpool/labfs          b41c26c205c548628e0b822dadf4ef8c
labpool      labfs-snap 2.68 GiB Feb 07 2024 14:36  /stratis/labpool/labfs-snap  b947e335c3a146bda1dd8e42ad19f586
[root@serverb ~]#
```

commands :

```
stratis filesystem snapshot labpool labfs labfs-snap
stratis filesystem list
```

- » **To check, remove the labfile1 and create another directory and mount the snapshot of stratis filesystem to it. After that, check if the backup of labfile1 exists and has similar contents.**

```
[root@serverb ~]# rm /labstratisvol/labfile1
rm: remove regular file '/labstratisvol/labfile1'? y
[root@serverb ~]# mkdir /labstratisvol-snap
[root@serverb ~]# mount /stratis/labpool/labfs-snap /labstratisvol-snap
[root@serverb ~]# cat /labstratisvol-snap/labfile1
Hello World!
[root@serverb ~]#
```

commands :

```
rm /labstratisvol/labfile1
mkdir /labstratisvol-snap
mount /stratis/labpool/labfs-snap /labstratisvol-snap
cat /labstratisvol-snap/labfile1
```

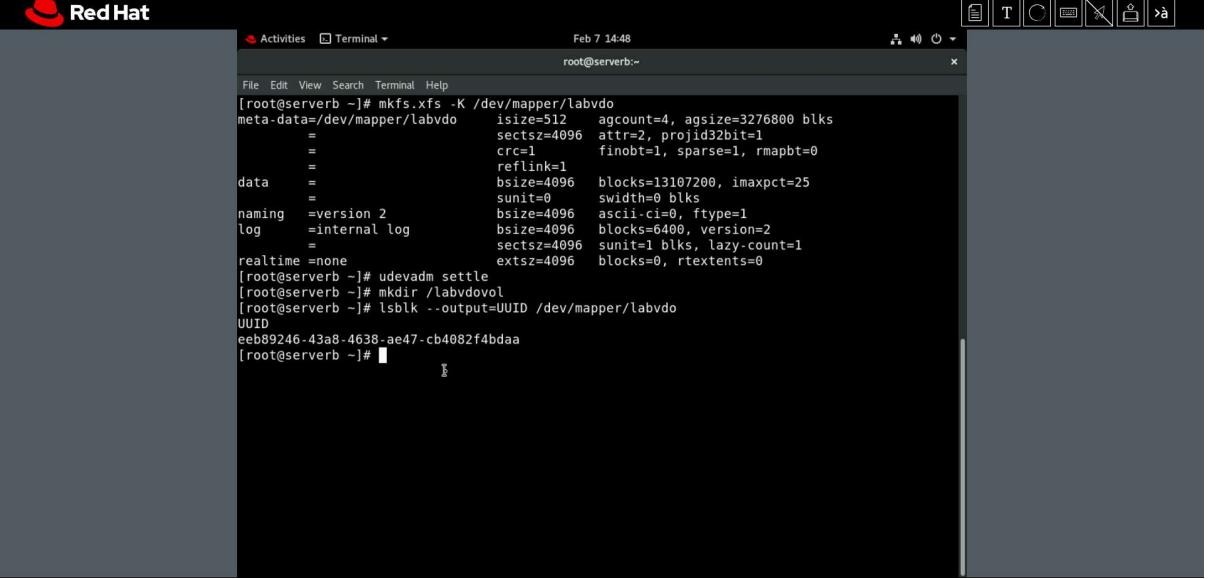
- » **Create the VDO volume of name labvdo with 50GB logical size and check the status.**

```
[root@serverb ~]# vdo create --name=labvdo --device=/dev/vdd --vdoLogicalSize=50G
Creating VDO labvdo
    The VDO volume can address 2 GB in 1 data slab.
    It can grow to address at most 16 TB of physical storage in 8192 slabs.
    If a larger maximum size might be needed, use bigger slabs.
Starting VDO labvdo
Starting compression on VDO labvdo
VDO instance 0 volume is ready at /dev/mapper/labvdo
[root@serverb ~]# vdo list \
> ^C
[root@serverb ~]# vdo list
labvdo
[root@serverb ~]# █
```

commands :

```
vdo create --name=labvdo --device=/dev/vdd --vdoLogicalSize=50G
vdo list
```

» **Assign the xfs filesystem to labvdo, check the status and get its UUID.**



```

Activities Terminal Feb 7 14:48
root@serverb:~#
File Edit View Search Terminal Help
[root@serverb ~]# mkfs.xfs -K /dev/mapper/labvdo
meta-data=/dev/mapper/labvdo isize=512 agcount=4, agsize=3276800 blks
          = sectsz=4096 attr=2, projid32bit=1
          = crc=1 finobt=1, sparse=1, rmapbt=0
data     = reflink=1
          = bsize=4096 blocks=13107200, imaxpct=25
          = sunit=0 swidth=0 blks
naming   =version 2 bsize=4096 ascii-ci=0, ftype=1
log      =internal log bsize=4096 blocks=6400, version=2
          = sectsz=4096 sunit=1 blks, lazy-count=1
realtime =none extsz=4096
[root@serverb ~]# udevadm settle
[root@serverb ~]# mkdir /labvdovol
[root@serverb ~]# lsblk --output=UUID /dev/mapper/labvdo
UUID
eeb89246-43a8-4638-ae47-cb4082f4bdaa
[root@serverb ~]#

```

commands :

mkfs.xfs -K /dev/mapper/labvdo
udevadm settle
mkdir /labvdovol
lsblk --output=UUID /dev/mapper/labvdo

» **Create an entry of vdo volume in /etc/fstab using vim.**

```
# /etc/fstab
# Created by anaconda on Thu Apr 23 05:11:56 2020
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=3cd0d4ca-93f6-423b-a469-79ab2b10b667 / xfs defaults 0 0
UUID=399c-0F7D /boot/efi vfat defaults,uid=0,gid=0,umask=077,shortname=winnt 0 2
UUID=b41c26c2-95c5-4862-8e0b-82dadff4ef8c /labvdovol xfs defaults,x-systemd.requires=stratisd.service 0 0
UUID=e6b89246-43a8-4638-ae47-cb4082740dd9 /labvdovol xfs defaults,x-systemd.requires=stratisd.service 0 0
```

commands :

vim /etc/fstab

» **Mount the labvdo volume to the directory previously created.**

```
[root@serverb student]# mount /labvdovol/
[root@serverb student]#
```

commands :

mount /labvdovol/

» **Now, create two copies of the /root/install.img file in the vdo volume and check the status before and after the creation.**

```
[root@serverb ~]# vdostats --human-readable
Device          Size      Used   Available  Use%  Space saving%
/dev/mapper/labvdo    5.0G     3.0G      2.0G  60%        99%
[root@serverb ~]# cp /root/install.img /labvdovol/install.img.1
[root@serverb ~]# cp /root/install.img /labvdovol/install.img.2
[root@serverb ~]# vdostats --human-readable
Device          Size      Used   Available  Use%  Space saving%
/dev/mapper/labvdo    5.0G     3.0G      2.0G  60%        99%
[root@serverb ~]#
```

commands :

```
vdostats --human-readable
cp /root/install.img /labvdovol/install.img.1
cp /root/install.img /labvdovol/install.img.2
vdostats --human-readable
```

» **Reboot serverb and grade the lab and ensure the successful completion of all the operations performed.**

```
[root@serverb ~]# exit
logout
Connection to serverb closed.
[student@workstation ~]$ lab advstorage-review grade

Grading the student's work on serverb:
· Verifying the stratis file system on serverb..... PASS
· Verifying the stratis snapshot on serverb..... PASS
· Verifying the files in the stratis file system on serverb... PASS
· Verifying the files in stratis snapshot on serverb..... PASS
· Verifying the VDO volume on serverb..... PASS
· Verifying the files in VDO volume on serverb..... PASS

Overall lab grade..... PASS

[student@workstation ~]$ lab advstorage-review finish

Completing the lab on serverb:
· Removing labfs-snap from on serverb..... SUCCESS
· Removing labfs from serverb..... SUCCESS
· Removing labpool from serverb..... SUCCESS
· Removing /labstratisvol from serverb..... SUCCESS
· Removing /labstratisvol-snap from serverb..... SUCCESS
· Removing install.img from serverb..... SUCCESS
· Removing labvdo volume from serverb..... SUCCESS
· Ensuring clean additional disks on serverb..... SUCCESS
· Restoring original /etc/fstab on serverb..... SUCCESS

Lab finished.

[student@workstation ~]$ █
```