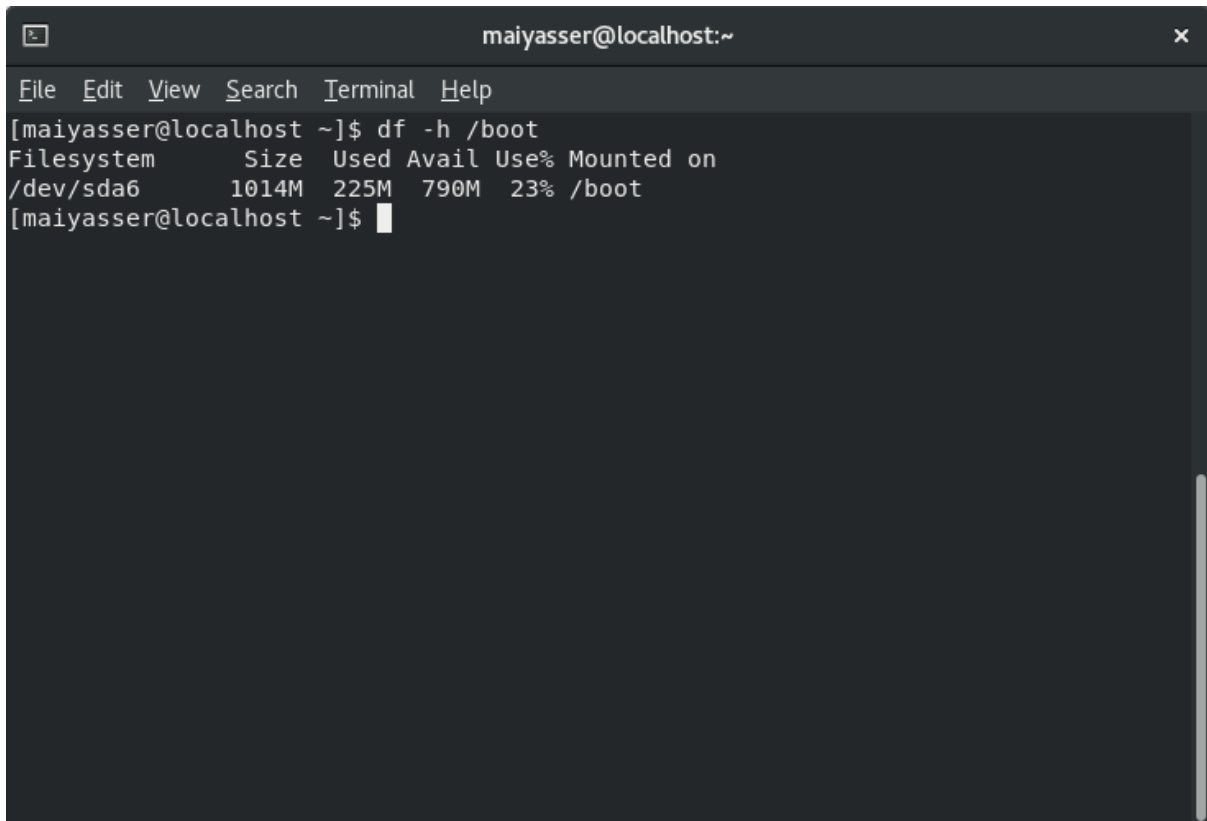


1- Find out what device node the /boot partition is



```
maiyasser@localhost:~  
File Edit View Search Terminal Help  
[maiyasser@localhost ~]$ df -h /boot  
Filesystem      Size  Used Avail Use% Mounted on  
/dev/sda6       1014M  225M  790M   23% /boot  
[maiyasser@localhost ~]$
```

2- Estimate the size in bytes of a level-zero dump for /boot

Sudo dump -S /boot

3- Back up the data of /boot to a dump file in /var/tmp/dumpfile

dump -0u -f /boot /var/tmp/dumpfile

4- Look in the /etc/dumpdates file and see how the dump command recorded the timestamp of the full backup

vi /etc/dumpdates

5- Use the restore command to view the contents of the dump file

restore -if /var/tmp/dumpfile

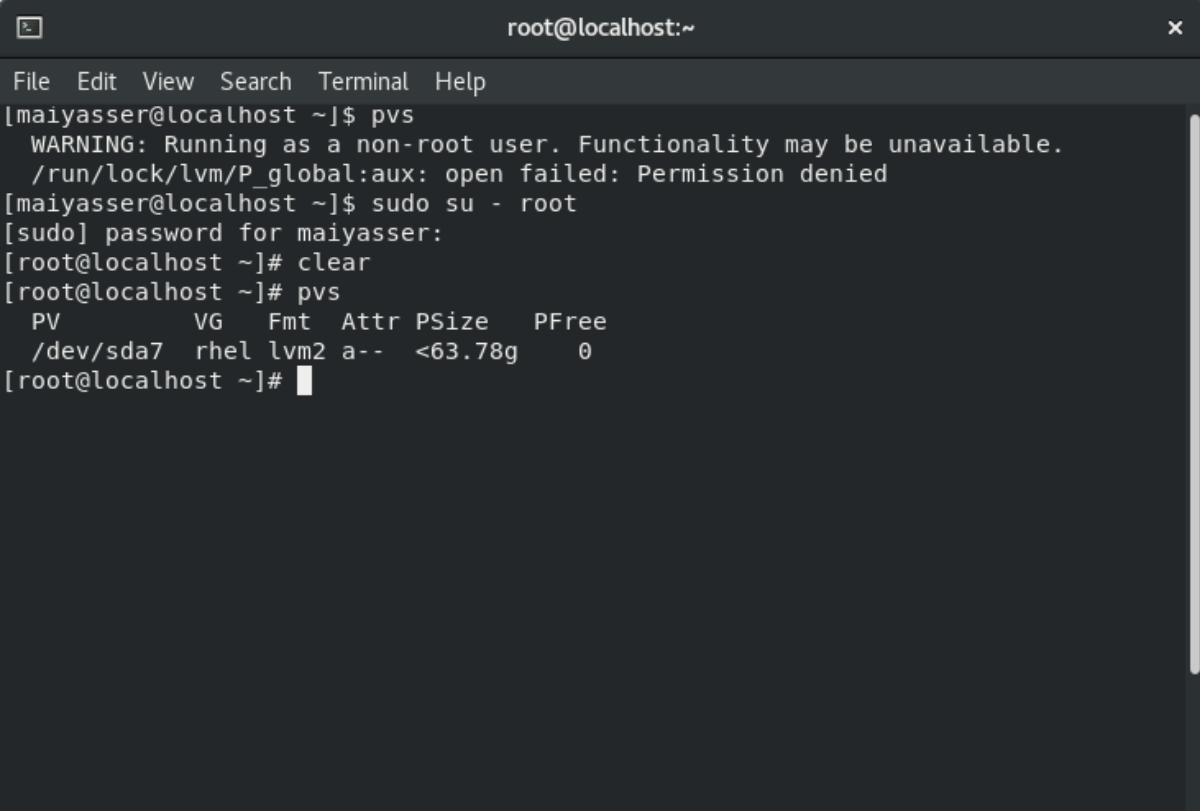
6- Use restore command in the interactive mode to extract /grub/splash.xpm.gz and /grub/grub.conf

restore -if /var/tmp/dumpfile /grub/splash.xpm.gz /grub/grub.conf

7- Use the fdisk command to create 2 Linux LVM (0x8e) partitions using "unpartitioned" space on your hard disk. These partitions should all be the same size; to speed up the lab, do not make them larger than 300 MB each. Make sure to write the changes to disk by using the w command to exit the fdisk utility. Run the partprobe command after exiting the fdisk utility.

8- Initialize your Linux LVM partitions as physical volumes with the pvcreate command. You can use the pvdisplay command to verify that the partitions have been initialized as physical volumes.

pvcreate -ff /dev/sda7

A terminal window titled 'root@localhost:~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the following commands and output:

```
[maiyasser@localhost ~]$ pvs
WARNING: Running as a non-root user. Functionality may be unavailable.
/run/lock/lvm/P_global:aux: open failed: Permission denied
[maiyasser@localhost ~]$ sudo su - root
[sudo] password for maiyasser:
[root@localhost ~]# clear
[root@localhost ~]# pvs
  PV          VG      Fmt  Attr PSize  PFree
  /dev/sda7   rhel  lvm2 a--  <63.78g  0
[root@localhost ~]#
```

9- Using only one of your physical volumes, create a volume group called test0. Use the vgdisplay command to verify that the volume group was created.

vgcreate test0 /dev/sda7

10- Create a small logical volume (LV) called data that uses about 30 percent of the available space of the test0 volume group. Look for VG Size and Free PE/Size in the output of the vgdisplay command to assist you with this. Use the lvdisplay command to verify your work.

lvcreate -n data -l 30%free test0

11- Create an ext2 filesystem on your new LV.

sudo mkfs.ext2 /dev/test0/data

12- Make a new directory called /data and then mount the new LV under the /data directory. Create a "large file" in this volume.

mkdir /data1

mount /dev/test0/data /data1

touch /data1/file1

blkid /dev/test0/data

13- Enlarge the LV that you created in Sequence 1 (/dev/test0/data) by using approximately 25 percent of the remaining free space in the test0 volume group. Then, use the ext2online command to enlarge the filesystem of the LV.

lvextend -r -l 25%free /dev/test0/data

14- Verify that the file /data/bigfile still exists in the LV. Run the df command and check to verify that more free disk space is now available on the LV.

df -hT

15- Use the remaining extents in the test0 volume group to create a second LV called docs.

lvcreate -n docs -l 100%free test0

16- Run the vgdisplay command to verify that there are no free extents left in the test0 volume group.

vgdisplay

17- Create an ext2 filesystem on the new LV, make a mount point called /docs and mount the docs LV using this mount point.

mkfs.xfs /dev/test0/docs

mkdir /docs

mount /dev/test0/docs /docs/

18- Add all of the remaining unused physical volumes that you created in Sequence 1 to the test0 volume group.

vgextend test0 /dev/sda7

19- If you run vgdisplay again, there now should be free extents (provided by the new physical volumes) in the test0 volume group. Extend the docs LV and underlying filesystem to make use of all of the free extents of the test0 volume group.

lvextend -l 100%free /dev/test0/docs

Verify your actions before moving on to the RAID sequence, disassemble your LVM-managed volumes by taking the following actions:

Remove any /etc/fstab entries you created.

umount /dev/test0/data

lvremove /dev/test0/data

umount /dev/test0/docs

lvremove /dev/test0/docs

vgchange -an test0 (this deactivates the volume group)

vgremove test0 (this deletes the volume group)

20- Run the fdisk command and convert the Linux LVM (0x8e) partitions that were created in above into Linux raid auto (0xfd) partitions. Save your changes and run the partprobe command

21- Initialize your RAID array (RAID 0)

mdadm --create /dev/md0 --level=0 --raid-devices=2

22- Format the RAID device with an ext3 filesystem

mdadm --create /dev/md0 --level=0 --raid-devices=2 /dev/nvme0n2p2 /dev/nvm02p3

23- Use the /data directory as a mount point for the /dev/md0 RAID device. Use the df command to check the size of the filesystem.

gedit /etc/fstab

mount -a

df