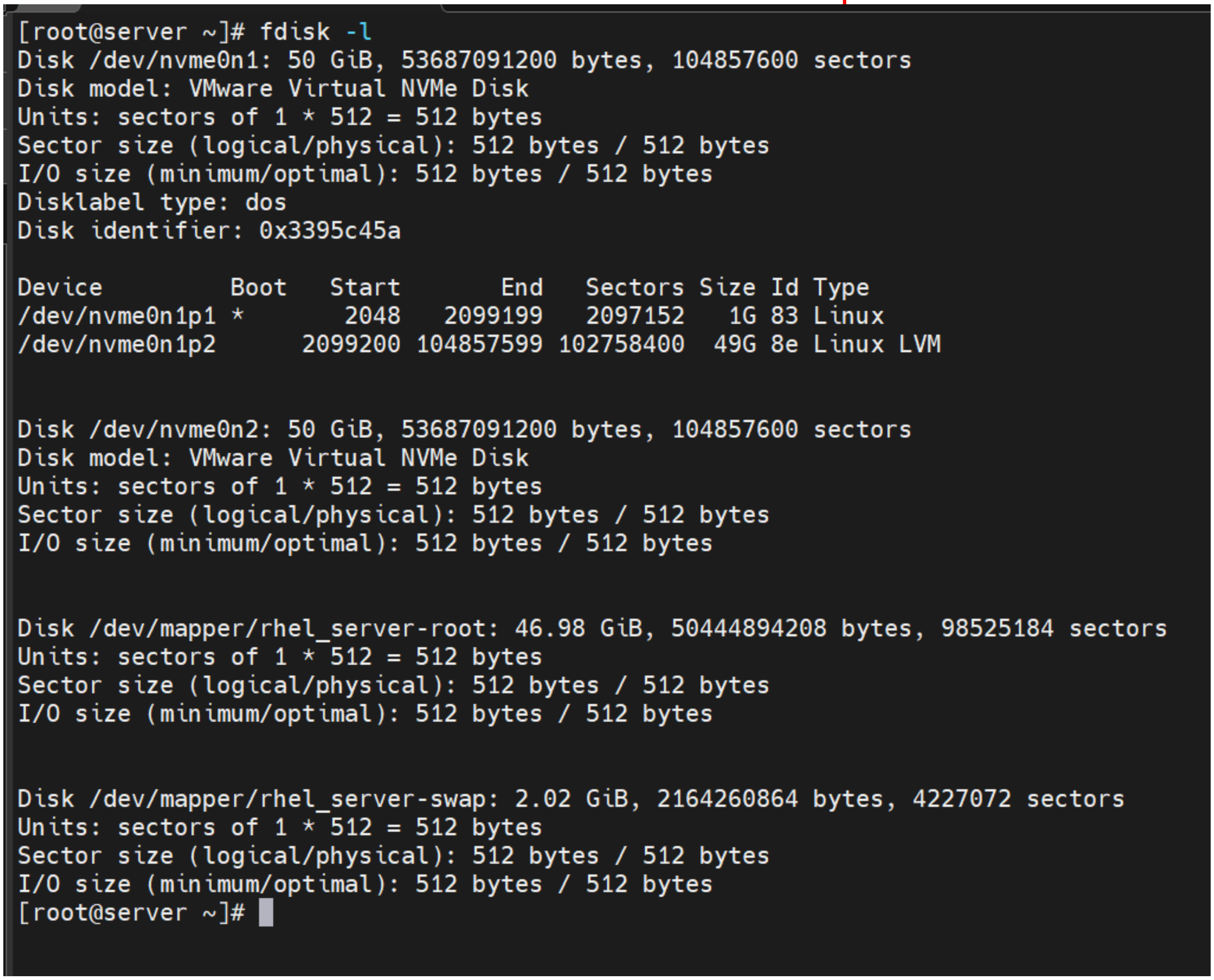
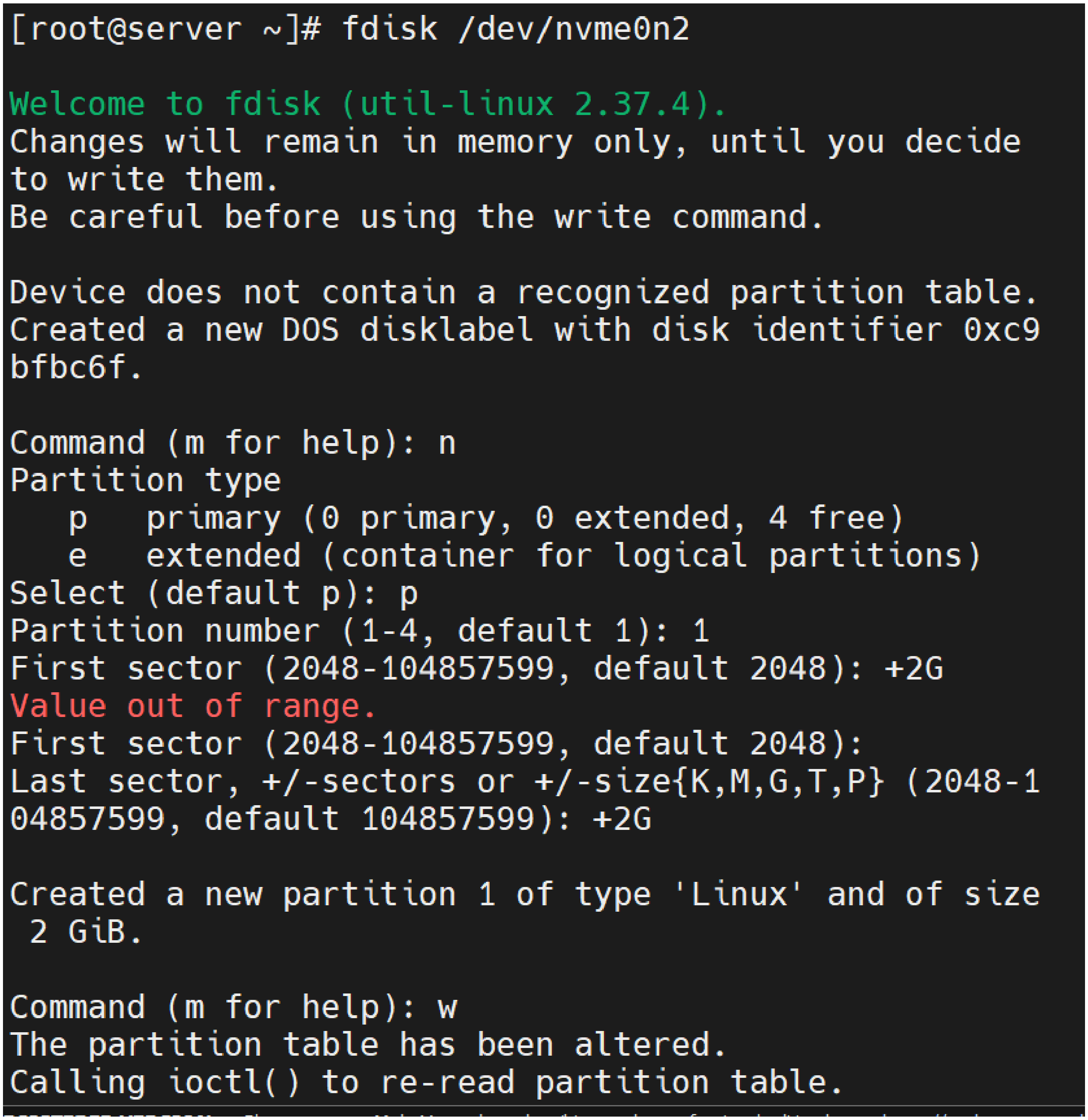
Lab 3

1. Use `fdisk -l` to locate information about the partition sizes.



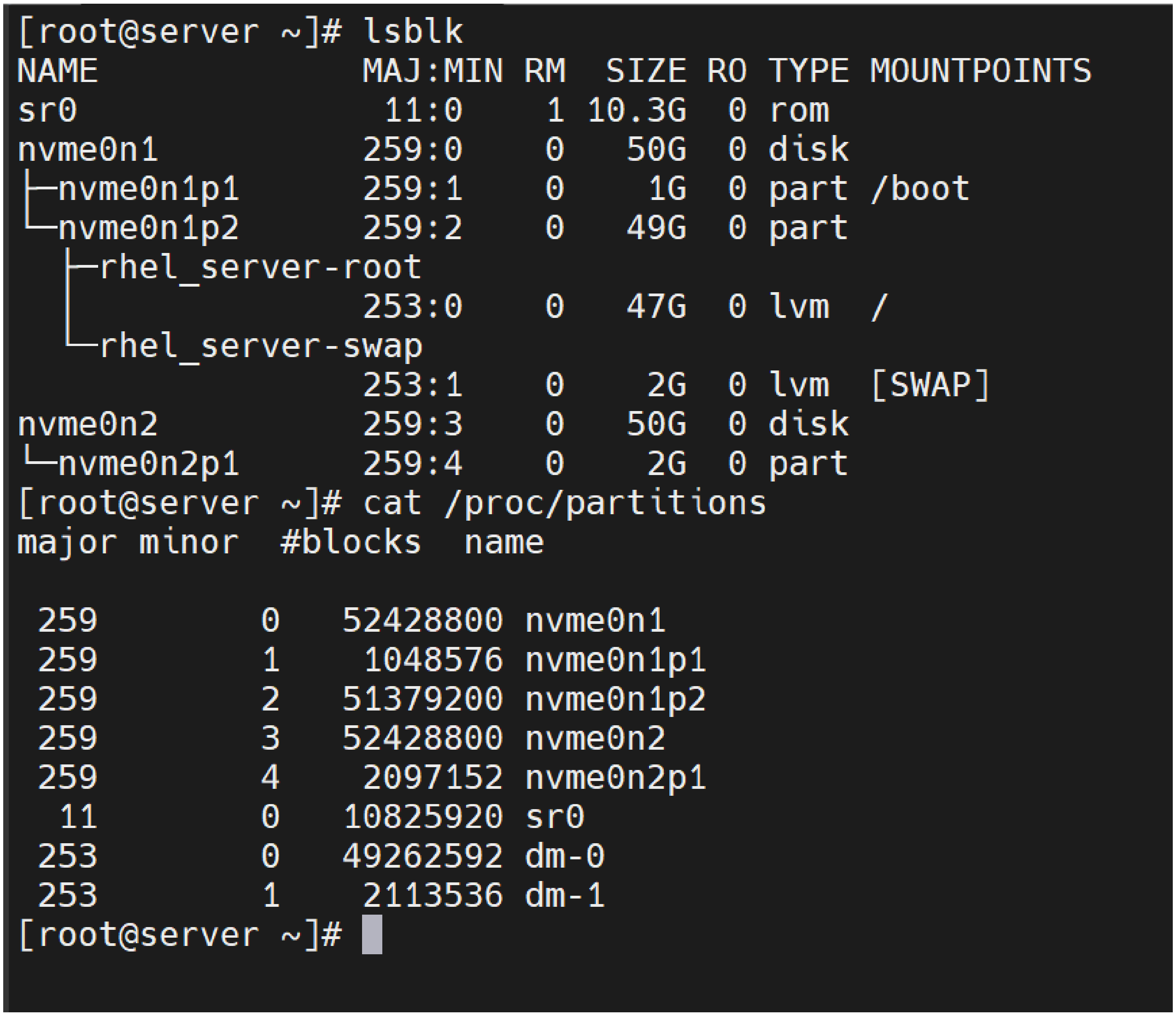
```
[root@server ~]# fdisk -l
Disk /dev/nvme0n1: 50 GiB, 53687091200 bytes, 104857600 sectors
Disk model: VMware Virtual NVMe Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x3395c45a

Device         Boot   Start       End   Sectors Size Id Type
/dev/nvme0n1p1 *       2048   2099199   2097152   1G 83 Linux
/dev/nvme0n1p2      2099200 104857599 102758400  49G 8e Linux LVM


Disk /dev/nvme0n2: 50 GiB, 53687091200 bytes, 104857600 sectors
Disk model: VMware Virtual NVMe Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/mapper/rhel_server-root: 46.98 GiB, 50444894208 bytes, 98525184 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


Disk /dev/mapper/rhel_server-swap: 2.02 GiB, 2164260864 bytes, 4227072 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
[root@server ~]# 
```

2. Use `fdisk` to add a new logical partition that is 2GB in size.Fdisk /dev/sdb



```
[root@server ~]# fdisk /dev/nvme0n2

Welcome to fdisk (util-linux 2.37.4).
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 0xc9
bfbc6f.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-104857599, default 2048): +2G
Value out of range.
First sector (2048-104857599, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-1
04857599, default 104857599): +2G

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

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

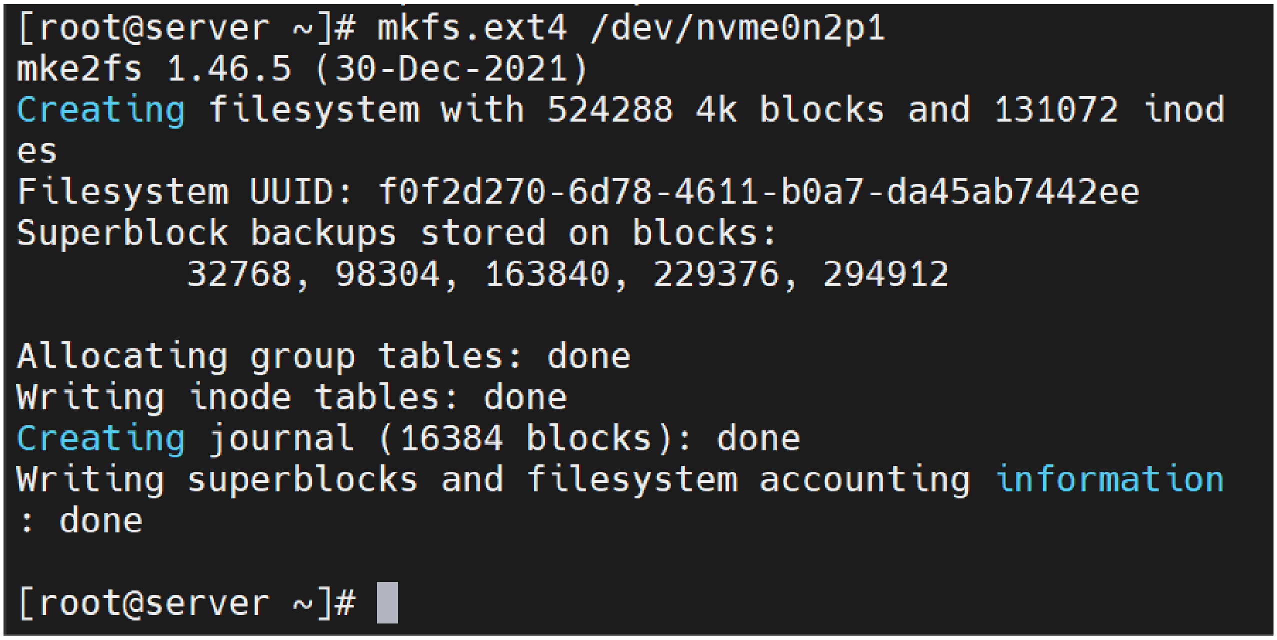
3. Did the kernel feel the changes? Display the content of /proc/partitions file? What didyou notice? How to overcome that?



```
[root@server ~]# lsblk
NAME             MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sr0               11:0    1 10.3G  0 rom
nvme0n1          259:0    0   50G  0 disk
├─nvme0n1p1      259:1    0    1G  0 part /boot
└─nvme0n1p2      259:2    0   49G  0 part
  ├─rhel_server-root
  │              253:0    0   47G  0 lvm  /
  └─rhel_server-swap
                 253:1    0    2G  0 lvm  [SWAP]
nvme0n2          259:3    0   50G  0 disk
└─nvme0n2p1      259:4    0    2G  0 part
[root@server ~]# cat /proc/partitions
major minor  #blocks  name

 259        0   52428800 nvme0n1
 259        1    1048576 nvme0n1p1
 259        2   51379200 nvme0n1p2
 259        3   52428800 nvme0n2
 259        4    2097152 nvme0n2p1
  11        0   10825920 sr0
 253        0   49262592 dm-0
 253        1    2113536 dm-1
[root@server ~]# 
```

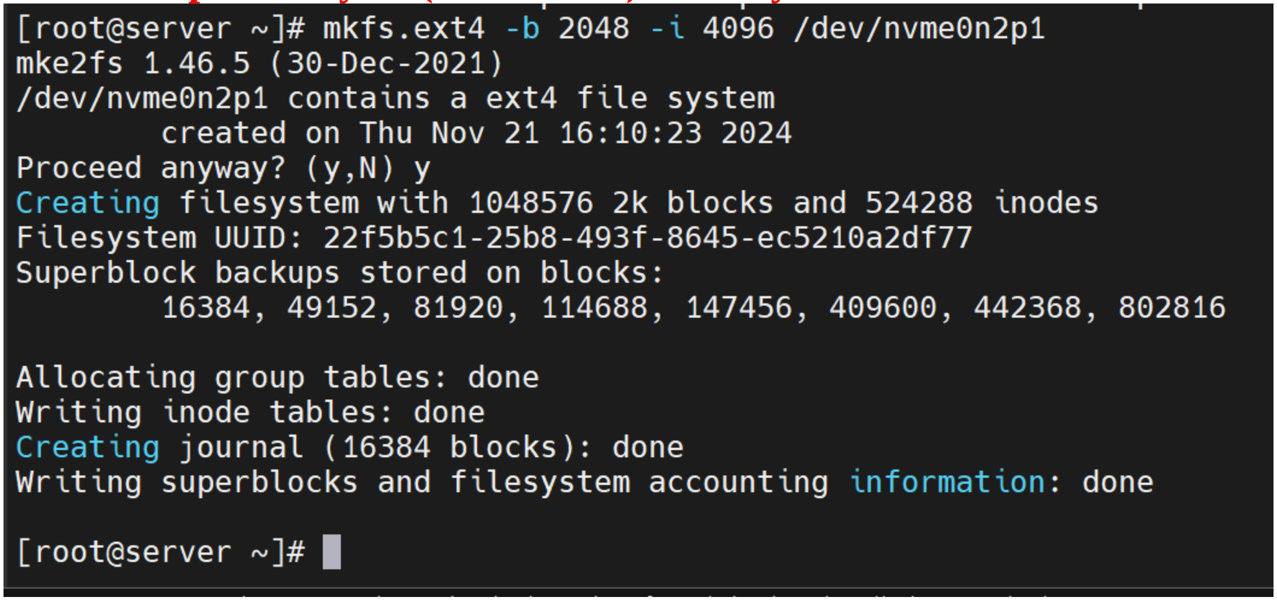
Yes feel the change and if else we can use partprobe

4. Make a new `ext4` file system on the new logical partition you just created.



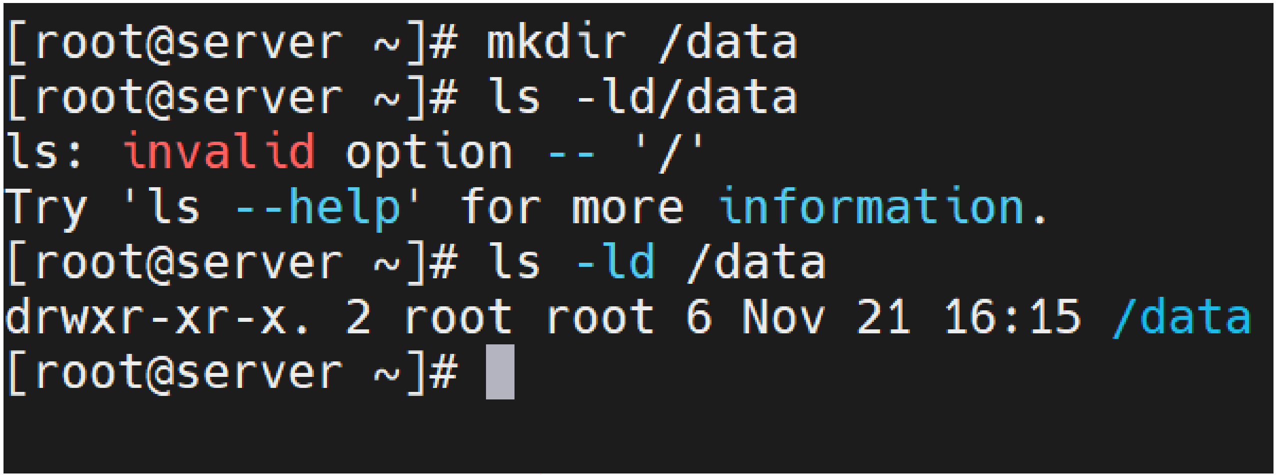
```
[root@server ~]# mkfs.ext4 /dev/nvme0n2p1
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 524288 4k blocks and 131072 inod
es
Filesystem UUID: f0f2d270-6d78-4611-b0a7-da45ab7442ee
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information
: done

[root@server ~]#
```

Bonus: Try creating the `ext4` filesystem with 2k blocks and one inode per every 4k(two blocks) of filesystem.



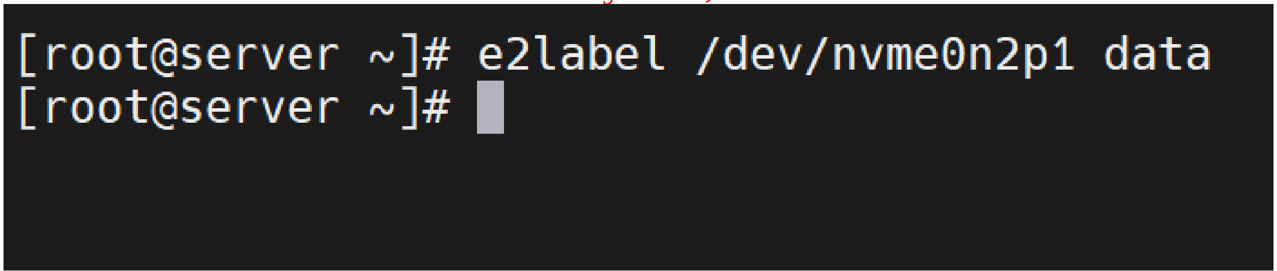
```
[root@server ~]# mkfs.ext4 -b 2048 -i 4096 /dev/nvme0n2p1
mke2fs 1.46.5 (30-Dec-2021)
/dev/nvme0n2p1 contains a ext4 file system
        created on Thu Nov 21 16:10:23 2024
Proceed anyway? (y,N) y
Creating filesystem with 1048576 2k blocks and 524288 inodes
Filesystem UUID: 22f5b5c1-25b8-493f-8645-ec5210a2df77
Superblock backups stored on blocks:
        16384, 49152, 81920, 114688, 147456, 409600, 442368, 802816

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

[root@server ~]#
```

5. Create a directory, name it `/data`.



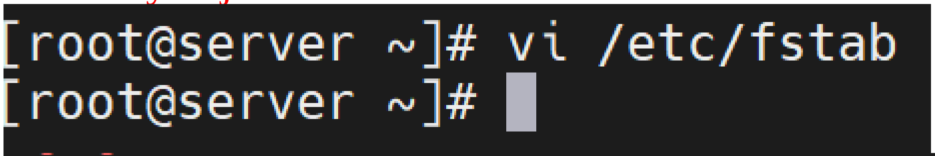
```
[root@server ~]# mkdir /data
[root@server ~]# ls -ld/data
ls: invalid option -- '/'
Try 'ls --help' for more information.
[root@server ~]# ls -ld /data
drwxr-xr-x. 2 root root 6 Nov 21 16:15 /data
[root@server ~]#
```

6. Add a label to the new filesystem, name it `data`.

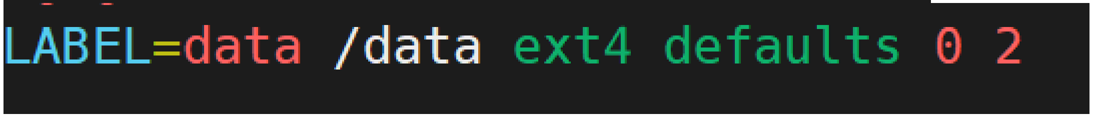


```
[root@server ~]# e2label /dev/nvme0n2p1 data
[root@server ~]#
```

7. Add a new entry to `/etc/fstab` for the new filesystem using the label you just create.

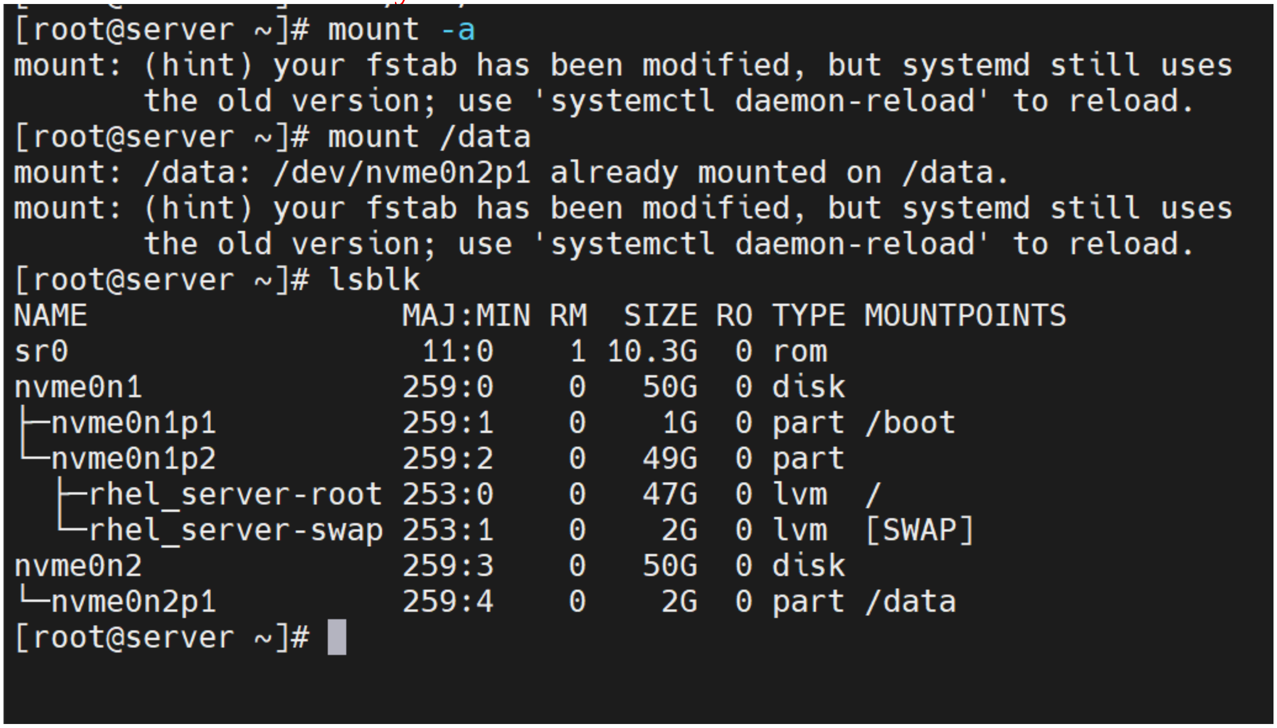


```
[root@server ~]# vi /etc/fstab
[root@server ~]#
```



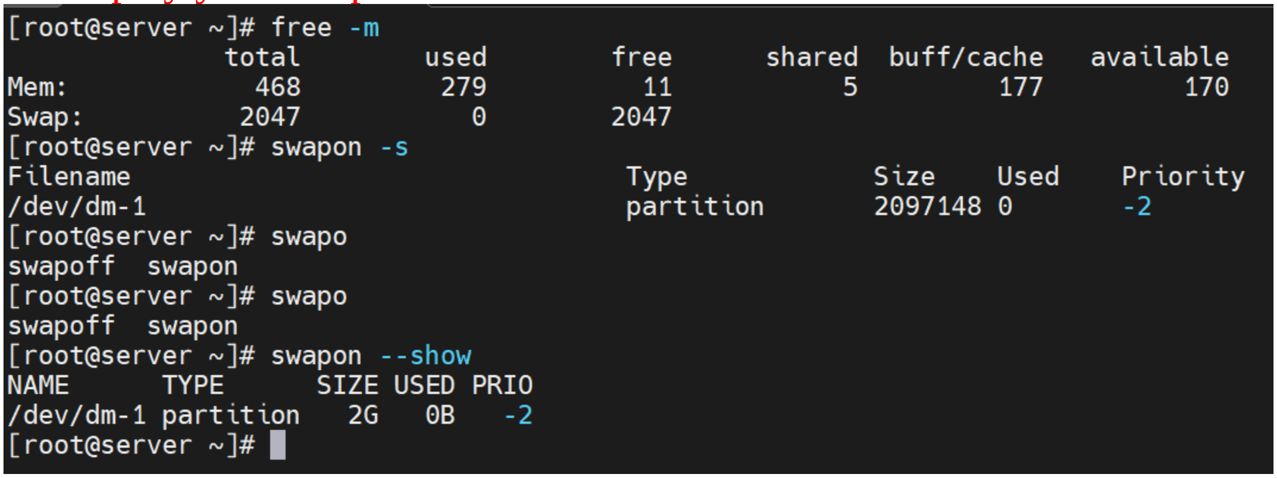
```
LABEL=data /data ext4 defaults 0 2
```

8. Mount the new filesystem



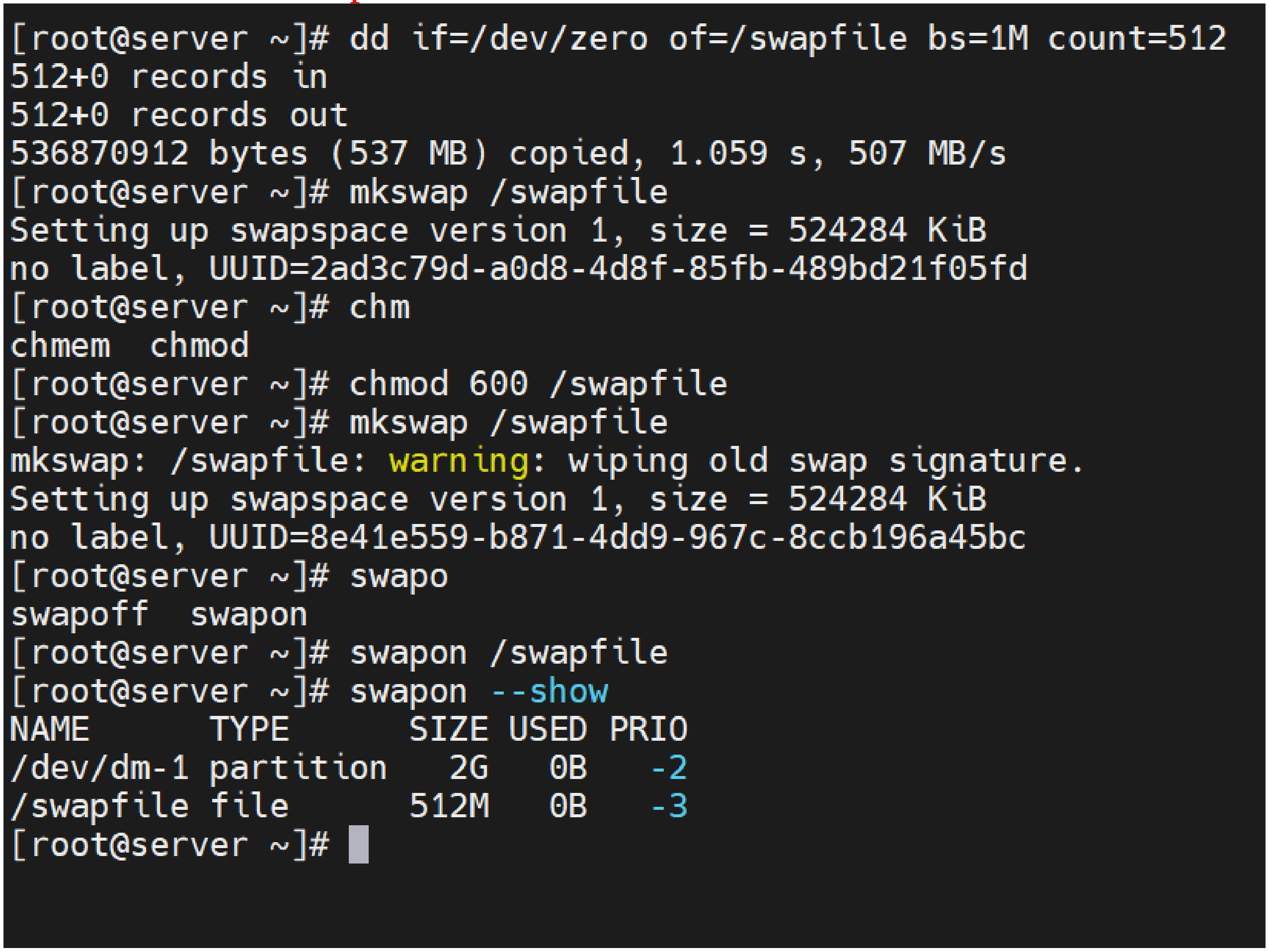
```
[root@server ~]# mount -a
mount: (hint) your fstab has been modified, but systemd still uses
       the old version; use 'systemctl daemon-reload' to reload.
[root@server ~]# mount /data
mount: /data: /dev/nvme0n2p1 already mounted on /data.
mount: (hint) your fstab has been modified, but systemd still uses
       the old version; use 'systemctl daemon-reload' to reload.
[root@server ~]# lsblk
NAME                     MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sr0                       11:0    1 10.3G  0 rom
nvme0n1                  259:0    0   50G  0 disk
├─nvme0n1p1              259:1    0    1G  0 part /boot
└─nvme0n1p2              259:2    0   49G  0 part
  ├─rhel_server-root 253:0    0   47G  0 lvm  /
  └─rhel_server-swap 253:1    0    2G  0 lvm  [SWAP]
nvme0n2                  259:3    0   50G  0 disk
└─nvme0n2p1              259:4    0    2G  0 part /data
[root@server ~]# 
```

9. Display your swap size.



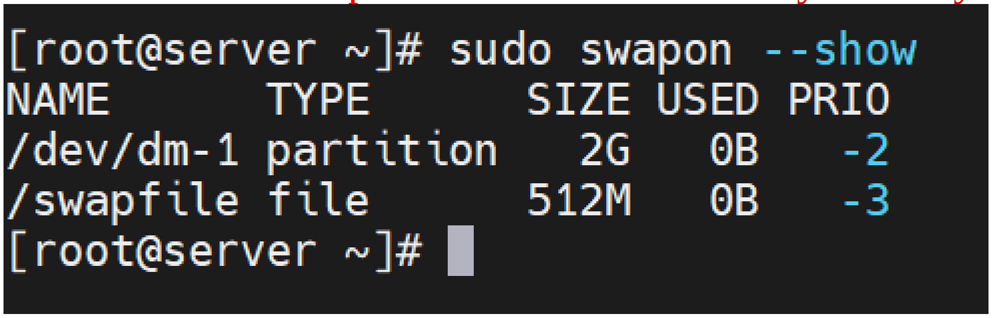
```
[root@server ~]# free -m
              total        used        free      shared  buff/cache   available
Mem:            468         279          11           5         177         170
Swap:          2047           0        2047
[root@server ~]# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dm-1                               partition       2097148 0       -2
[root@server ~]# swapo
swapoff  swapon
[root@server ~]# swapo
swapoff  swapon
[root@server ~]# swapon --show
NAME      TYPE      SIZE USED PRIO
/dev/dm-1 partition   2G   0B   -2
[root@server ~]#
```

10. Create a swap file of size 512MB.



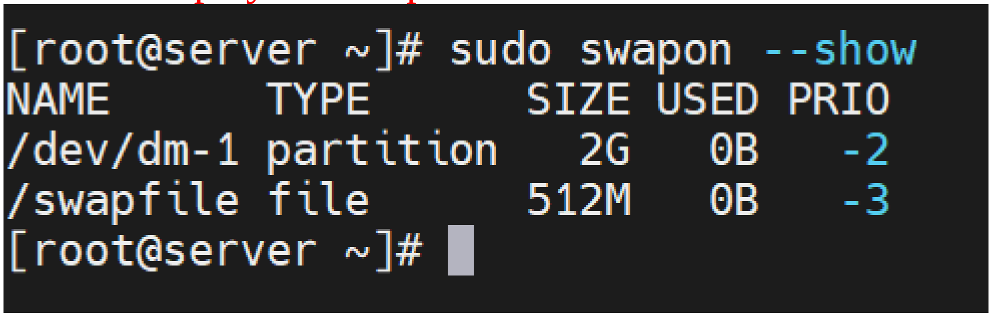
```
[root@server ~]# dd if=/dev/zero of=/swapfile bs=1M count=512
512+0 records in
512+0 records out
536870912 bytes (537 MB) copied, 1.059 s, 507 MB/s
[root@server ~]# mkswap /swapfile
Setting up swapspace version 1, size = 524284 KiB
no label, UUID=2ad3c79d-a0d8-4d8f-85fb-489bd21f05fd
[root@server ~]# chm
chmem  chmod
[root@server ~]# chmod 600 /swapfile
[root@server ~]# mkswap /swapfile
mkswap: /swapfile: warning: wiping old swap signature.
Setting up swapspace version 1, size = 524284 KiB
no label, UUID=8e41e559-b871-4dd9-967c-8ccb196a45bc
[root@server ~]# swapo
swapoff  swapon
[root@server ~]# swapon /swapfile
[root@server ~]# swapon --show
NAME      TYPE      SIZE USED PRIO
/dev/dm-1 partition   2G   0B   -2
/swapfile file      512M   0B   -3
[root@server ~]#
```

11. Add the swap file to the virtual memory of the system.



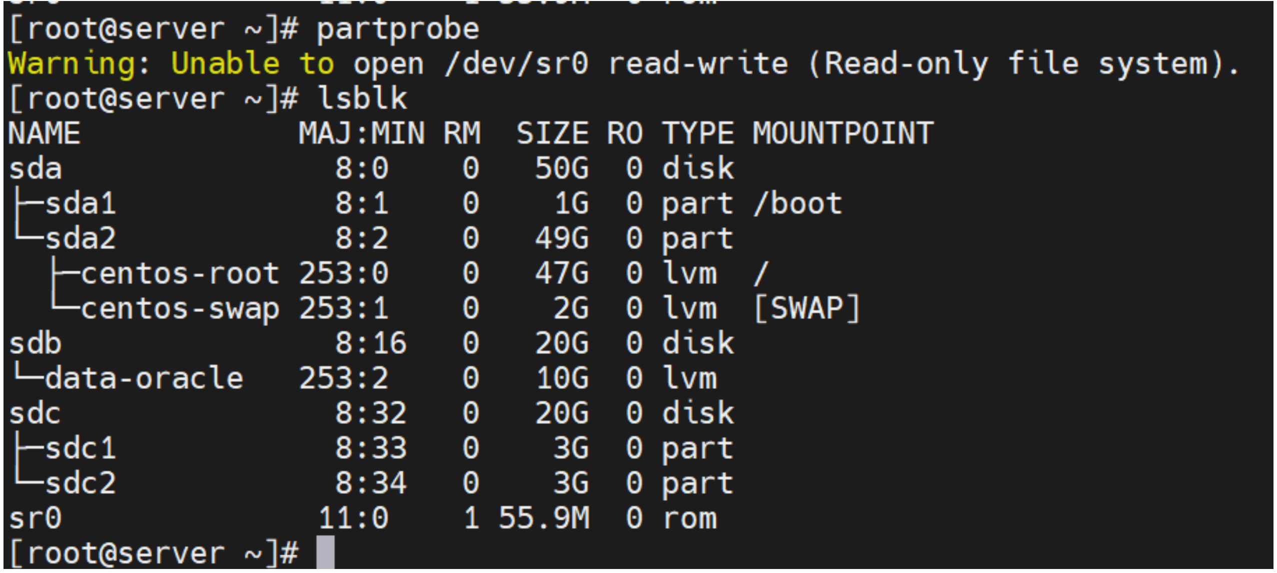
```
[root@server ~]# sudo swapon --show
NAME      TYPE      SIZE USED PRIO
/dev/dm-1 partition   2G   0B   -2
/swapfile file      512M   0B   -3
[root@server ~]# 
```

12. Display the swap size.



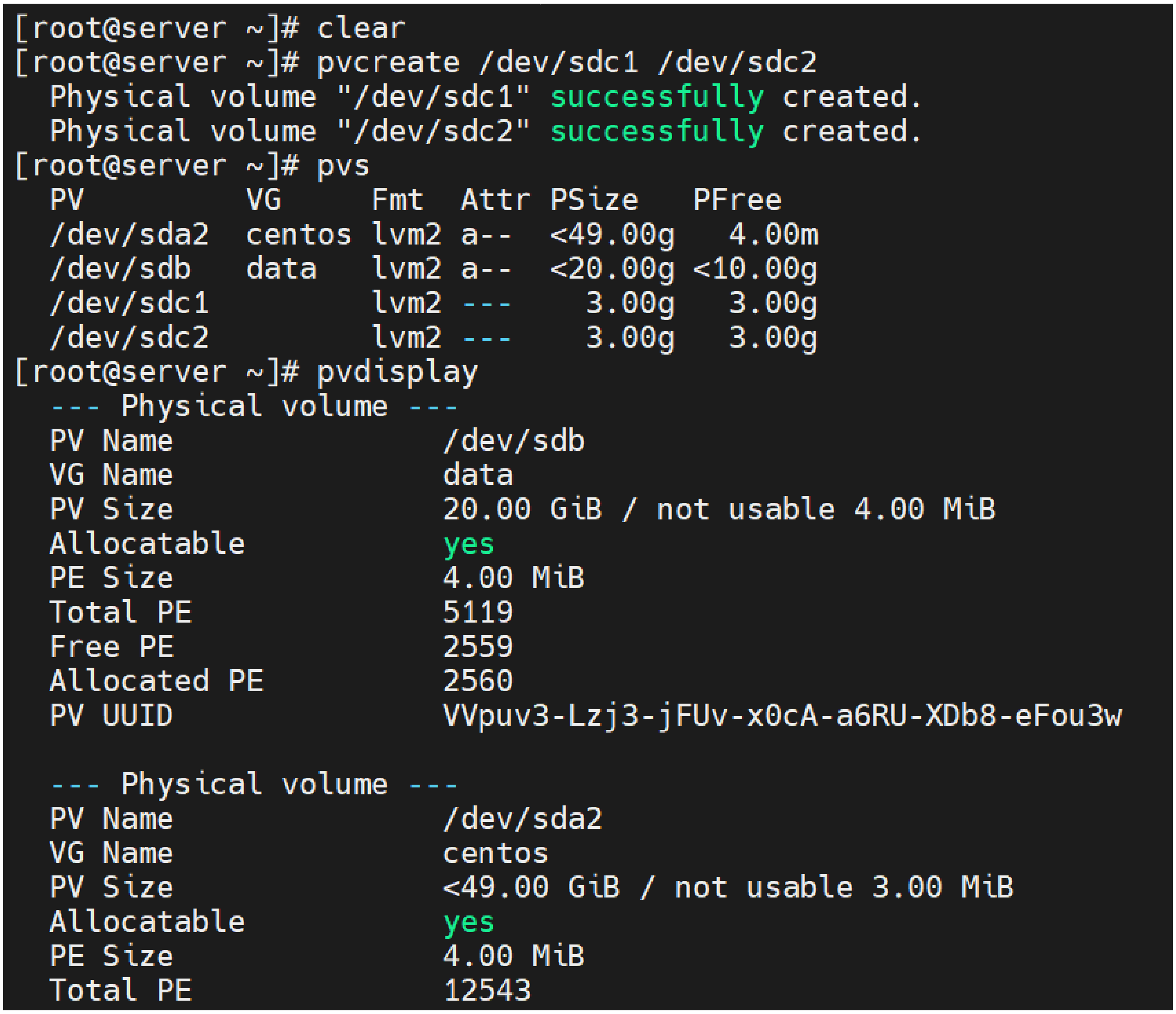
```
[root@server ~]# sudo swapon --show
NAME      TYPE      SIZE USED PRIO
/dev/dm-1 partition   2G   0B   -2
/swapfile file      512M   0B   -3
[root@server ~]# 
```

13. 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.



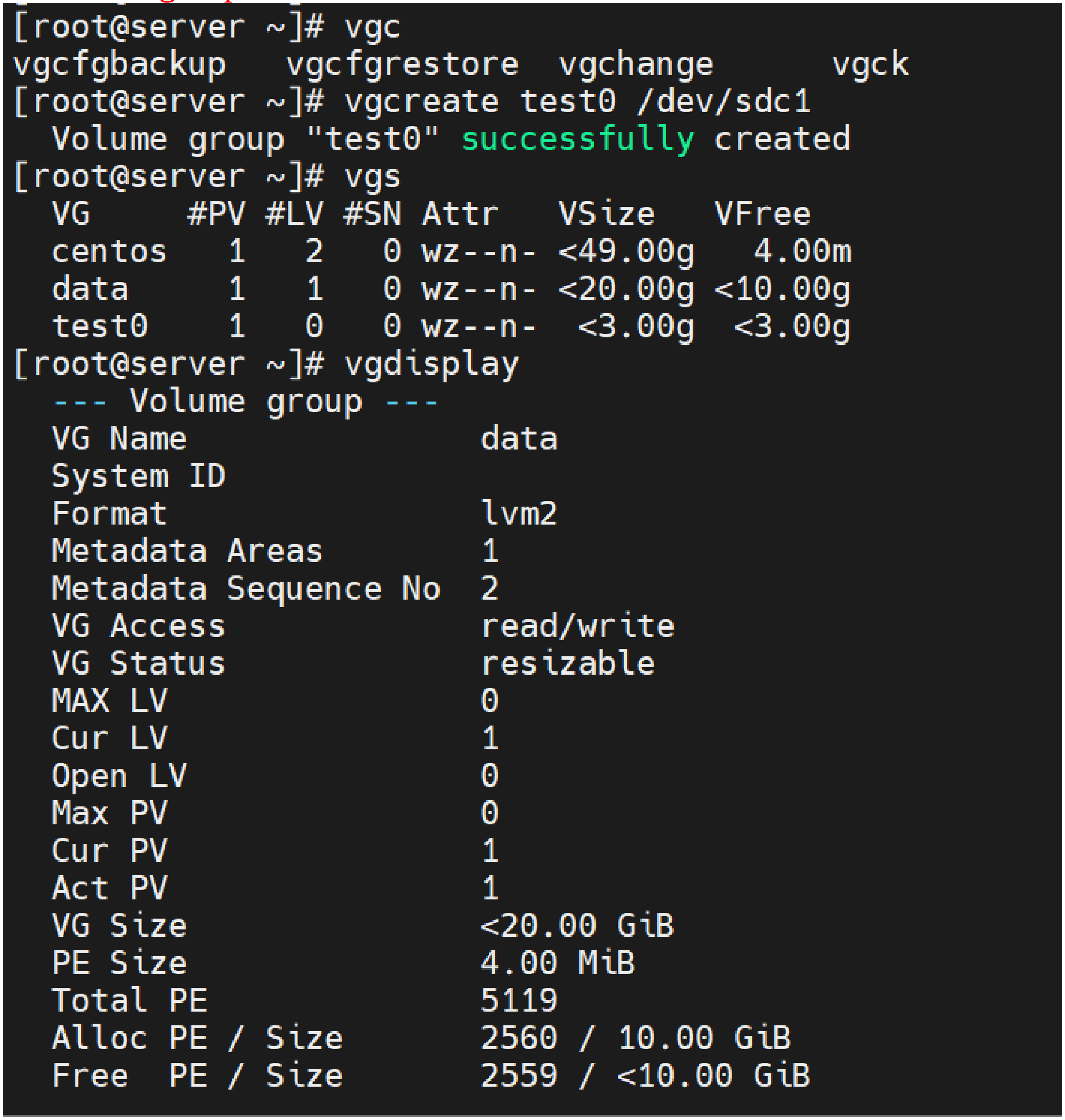
```
[root@server ~]# partprobe
Warning: Unable to open /dev/sr0 read-write (Read-only file system).
[root@server ~]# lsblk
NAME              MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                 8:0    0   50G  0 disk
├─sda1              8:1    0    1G  0 part /boot
└─sda2              8:2    0   49G  0 part
  ├─centos-root 253:0    0   47G  0 lvm  /
  └─centos-swap 253:1    0    2G  0 lvm  [SWAP]
sdb                 8:16   0   20G  0 disk
└─data-oracle     253:2    0   10G  0 lvm
sdc                 8:32   0   20G  0 disk
├─sdc1              8:33   0    3G  0 part
└─sdc2              8:34   0    3G  0 part
sr0                11:0    1 55.9M  0 rom
[root@server ~]# 
```

14. 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 asphysical volumes.



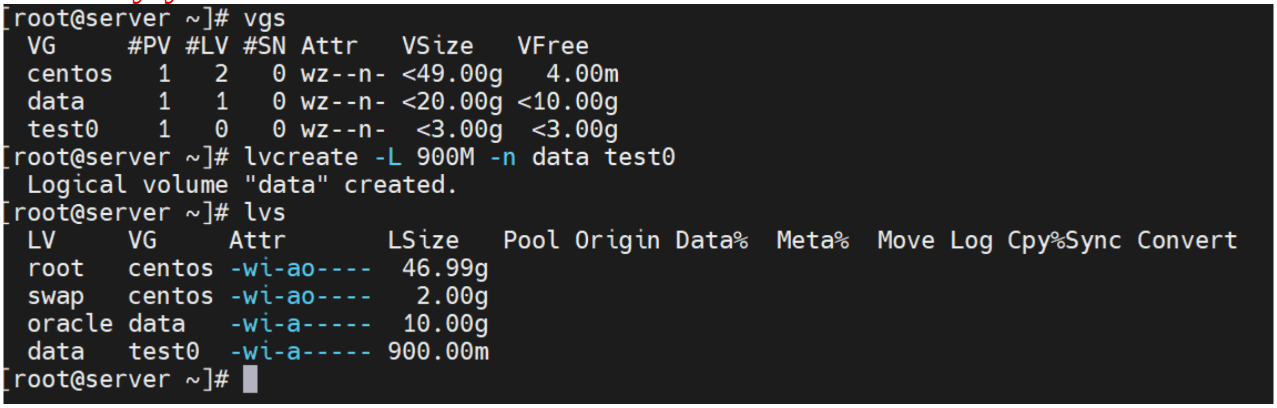
```
[root@server ~]# clear
[root@server ~]# pvcreate /dev/sdc1 /dev/sdc2
  Physical volume "/dev/sdc1" successfully created.
  Physical volume "/dev/sdc2" successfully created.
[root@server ~]# pvs
  PV         VG     Fmt  Attr PSize   PFree
  /dev/sda2  centos lvm2 a--  <49.00g   4.00m
  /dev/sdb   data   lvm2 a--  <20.00g <10.00g
  /dev/sdc1         lvm2 ---    3.00g   3.00g
  /dev/sdc2         lvm2 ---    3.00g   3.00g
[root@server ~]# pvdisplay
  --- Physical volume ---
  PV Name               /dev/sdb
  VG Name               data
  PV Size               20.00 GiB / not usable 4.00 MiB
  Allocatable           yes
  PE Size               4.00 MiB
  Total PE              5119
  Free PE               2559
  Allocated PE          2560
  PV UUID               VVpuv3-Lzj3-jFUv-x0cA-a6RU-XDb8-eFou3w

  --- Physical volume ---
  PV Name               /dev/sda2
  VG Name               centos
  PV Size               <49.00 GiB / not usable 3.00 MiB
  Allocatable           yes
  PE Size               4.00 MiB
  Total PE              12543
```

15. 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.



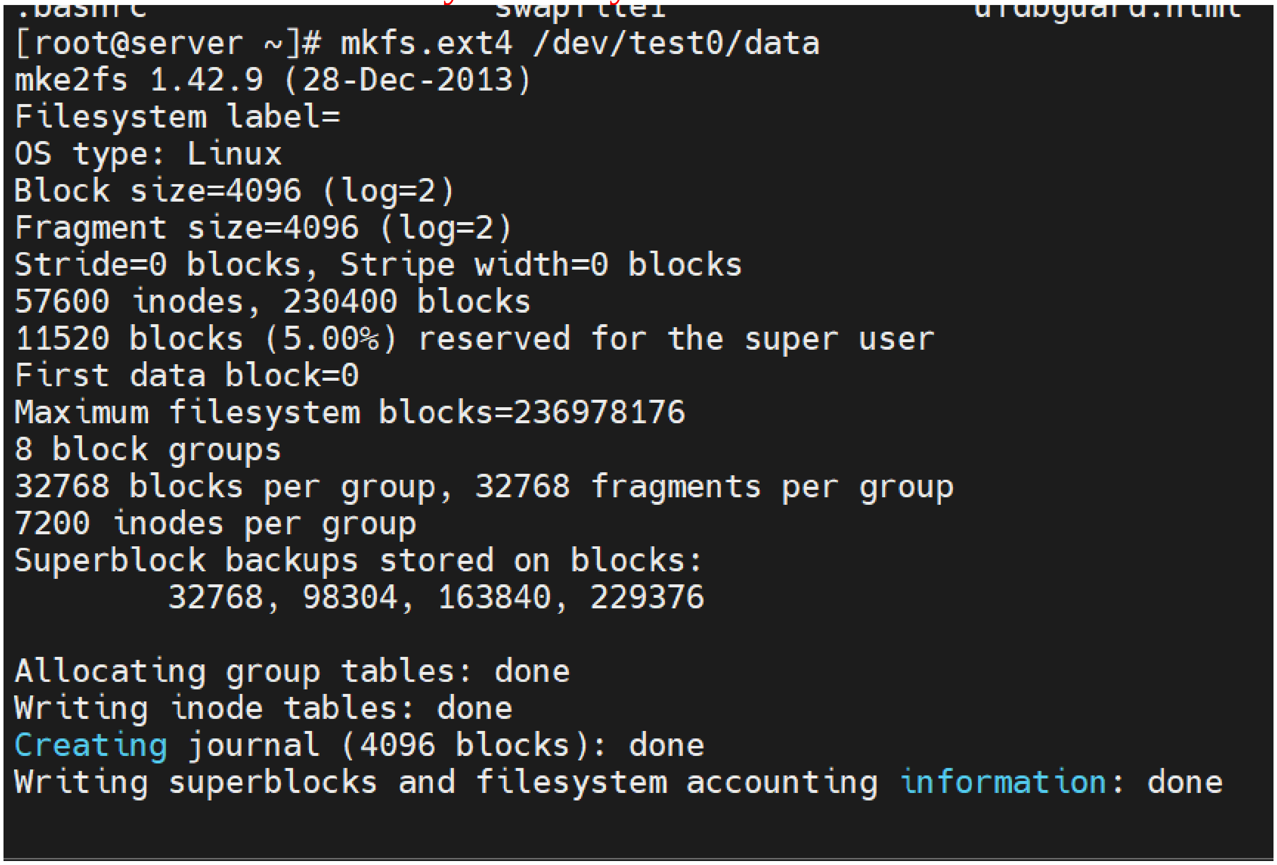
```
[root@server ~]# vgc
vgcfgbackup   vgcfgrestore  vgchange      vgck
[root@server ~]# vgcreate test0 /dev/sdc1
  Volume group "test0" successfully created
[root@server ~]# vgs
  VG     #PV #LV #SN Attr   VSize   VFree
  centos   1   2   0 wz--n- <49.00g   4.00m
  data     1   1   0 wz--n- <20.00g <10.00g
  test0    1   0   0 wz--n-  <3.00g  <3.00g
[root@server ~]# vgdisplay
  --- Volume group ---
  VG Name               data
  System ID
  Format                lvm2
  Metadata Areas        1
  Metadata Sequence No  2
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                1
  Open LV               0
  Max PV                0
  Cur PV                1
  Act PV                1
  VG Size               <20.00 GiB
  PE Size               4.00 MiB
  Total PE              5119
  Alloc PE / Size       2560 / 10.00 GiB
  Free  PE / Size       2559 / <10.00 GiB
```

16. Create a small logical volume (LV) called data that uses about 30 percent of the availablespace 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.



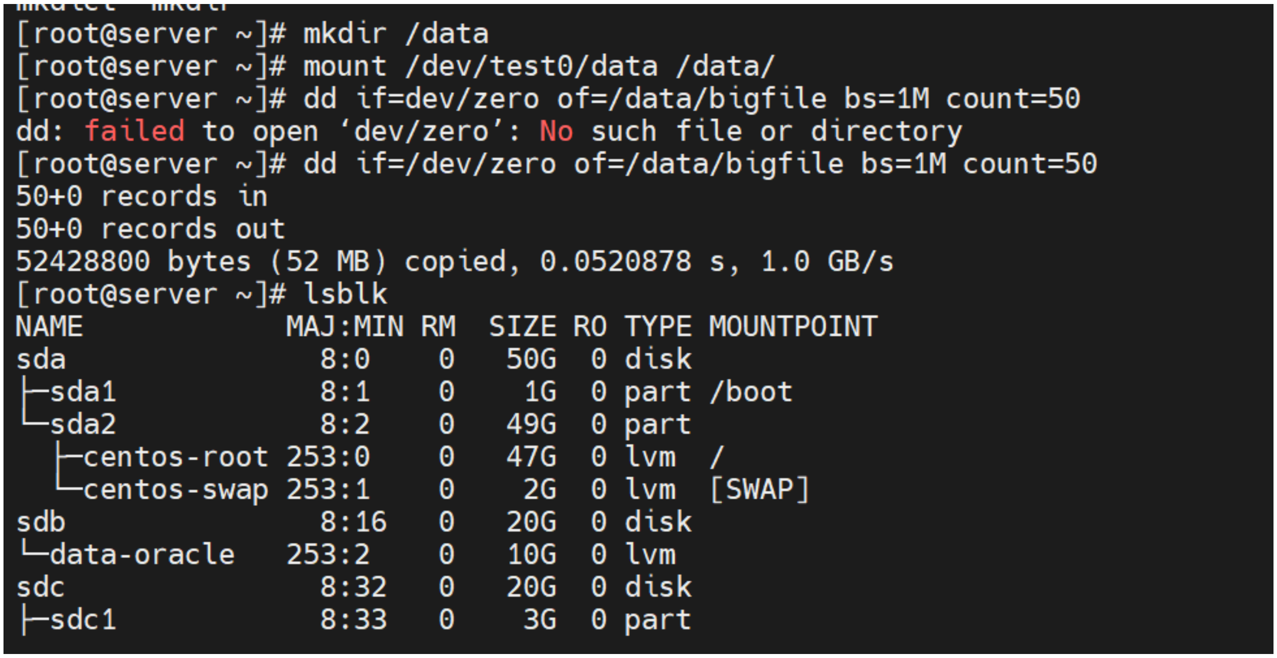
```
[root@server ~]# vgs
  VG     #PV #LV #SN Attr   VSize   VFree
  centos   1   2   0 wz--n- <49.00g   4.00m
  data     1   1   0 wz--n- <20.00g <10.00g
  test0    1   0   0 wz--n-  <3.00g  <3.00g
[root@server ~]# lvcreate -L 900M -n data test0
  Logical volume "data" created.
[root@server ~]# lvs
  LV     VG     Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  root   centos -wi-ao----  46.99g
  swap   centos -wi-ao----   2.00g
  oracle data   -wi-a-----  10.00g
  data   test0  -wi-a----- 900.00m
[root@server ~]#
```

17. Create an xfs filesystem on your new LV.



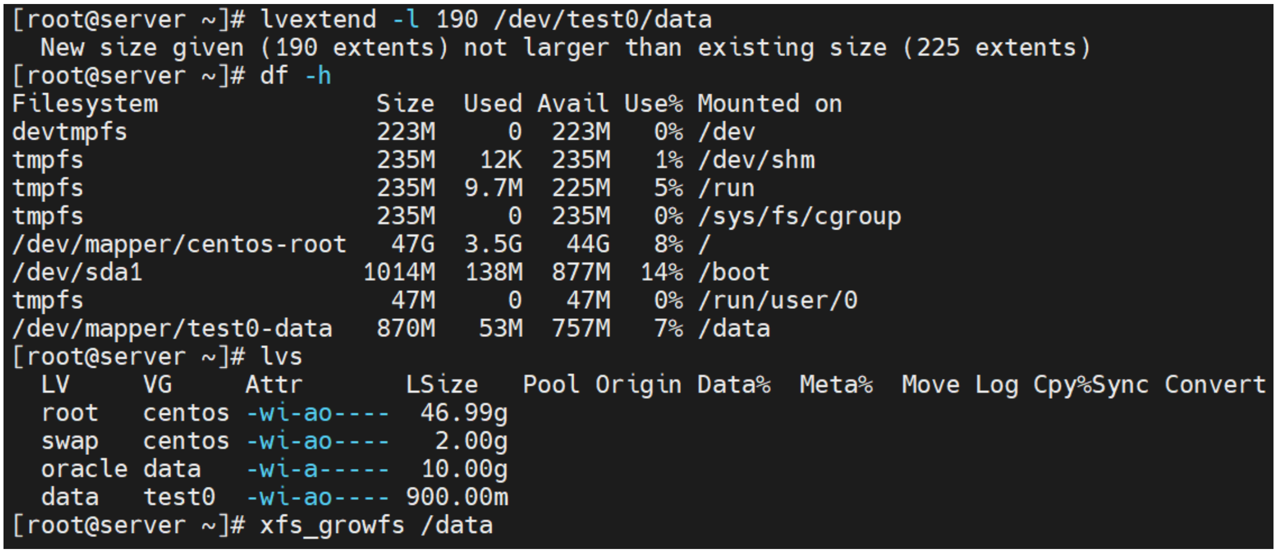
```
[root@server ~]# mkfs.ext4 /dev/test0/data
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
57600 inodes, 230400 blocks
11520 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=236978176
8 block groups
32768 blocks per group, 32768 fragments per group
7200 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done
```

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



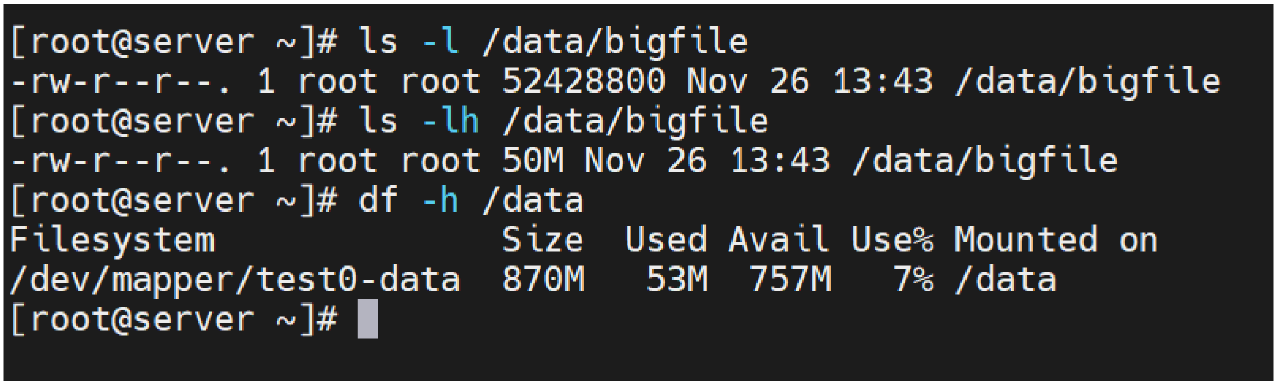
```
[root@server ~]# mkdir /data
[root@server ~]# mount /dev/test0/data /data/
[root@server ~]# dd if=dev/zero of=/data/bigfile bs=1M count=50
dd: failed to open ‘dev/zero’: No such file or directory
[root@server ~]# dd if=/dev/zero of=/data/bigfile bs=1M count=50
50+0 records in
50+0 records out
52428800 bytes (52 MB) copied, 0.0520878 s, 1.0 GB/s
[root@server ~]# lsblk
NAME               MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                  8:0    0   50G  0 disk
├─sda1               8:1    0    1G  0 part /boot
└─sda2               8:2    0   49G  0 part
  ├─centos-root    253:0    0   47G  0 lvm  /
  └─centos-swap    253:1    0    2G  0 lvm  [SWAP]
sdb                  8:16   0   20G  0 disk
└─data-oracle      253:2    0   10G  0 lvm
sdc                  8:32   0   20G  0 disk
├─sdc1               8:33   0    3G  0 part
```

19. 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, enlarge the filesystem of the LV.



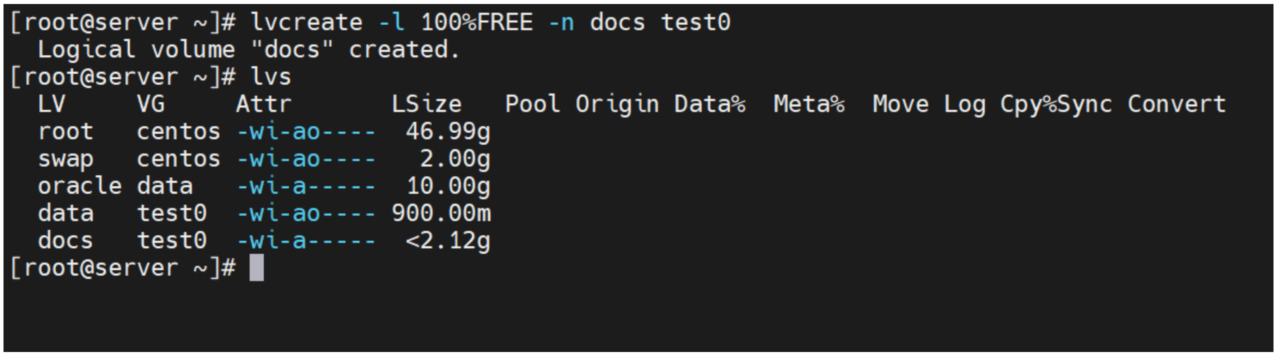
```
[root@server ~]# lvextend -l 190 /dev/test0/data
  New size given (190 extents) not larger than existing size (225 extents)
[root@server ~]# df -h
Filesystem               Size  Used Avail Use% Mounted on
devtmpfs                 223M     0  223M   0% /dev
tmpfs                    235M   12K  235M   1% /dev/shm
tmpfs                    235M  9.7M  225M   5% /run
tmpfs                    235M     0  235M   0% /sys/fs/cgroup
/dev/mapper/centos-root   47G  3.5G   44G   8% /
/dev/sda1               1014M  138M  877M  14% /boot
tmpfs                     47M     0   47M   0% /run/user/0
/dev/mapper/test0-data   870M   53M  757M   7% /data
[root@server ~]# lvs
  LV     VG     Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  root   centos -wi-ao----  46.99g
  swap   centos -wi-ao----   2.00g
  oracle data   -wi-a-----  10.00g
  data   test0  -wi-ao---- 900.00m
[root@server ~]# xfs_growfs /data
```

20. 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.



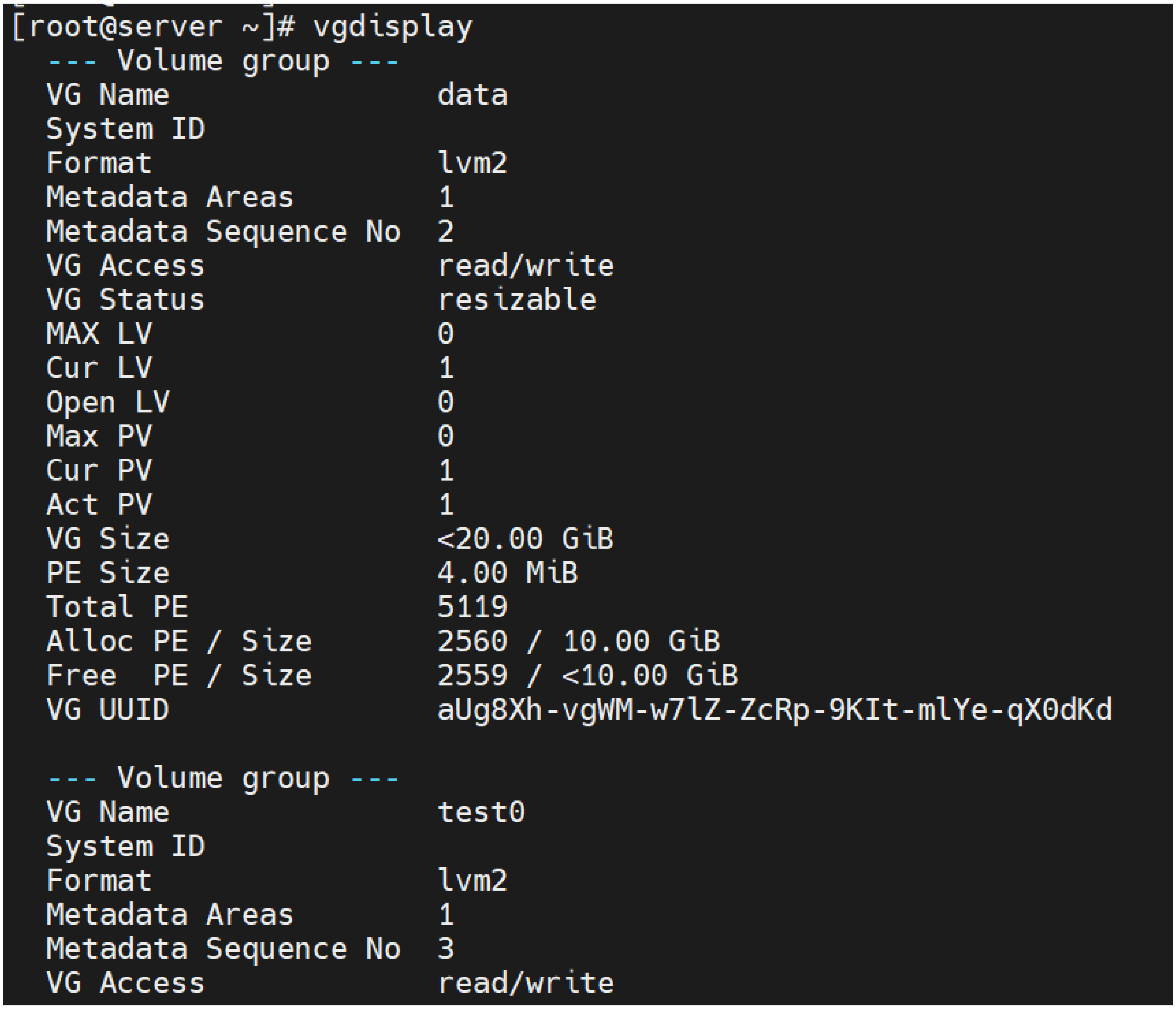
```
[root@server ~]# ls -l /data/bigfile
-rw-r--r--. 1 root root 52428800 Nov 26 13:43 /data/bigfile
[root@server ~]# ls -lh /data/bigfile
-rw-r--r--. 1 root root 50M Nov 26 13:43 /data/bigfile
[root@server ~]# df -h /data
Filesystem              Size  Used Avail Use% Mounted on
/dev/mapper/test0-data  870M   53M  757M   7% /data
[root@server ~]#
```

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

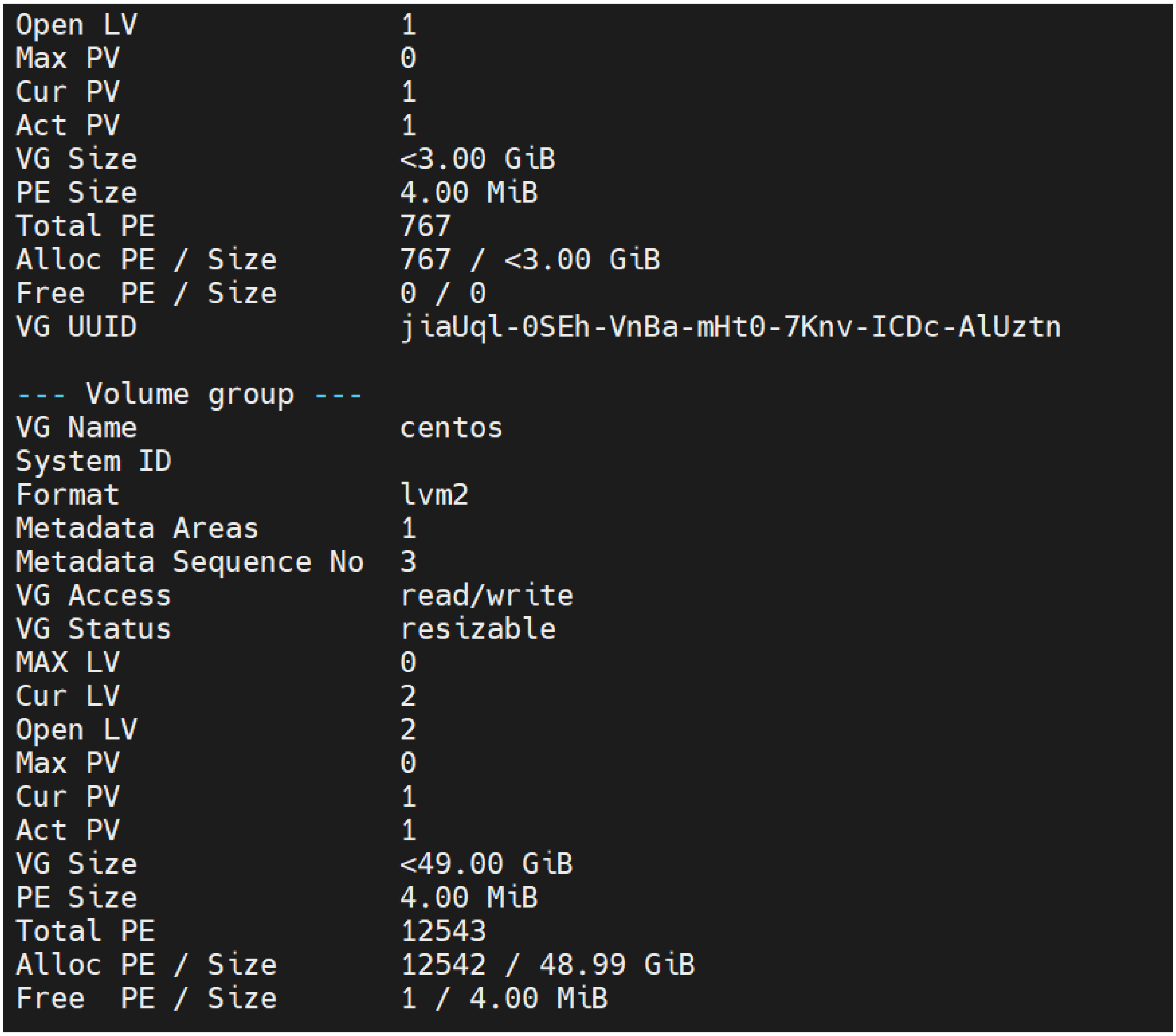


```
[root@server ~]# lvcreate -l 100%FREE -n docs test0
  Logical volume "docs" created.
[root@server ~]# lvs
  LV     VG     Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  root   centos -wi-ao----  46.99g
  swap   centos -wi-ao----   2.00g
  oracle data   -wi-a-----  10.00g
  data   test0  -wi-ao---- 900.00m
  docs   test0  -wi-a-----  <2.12g
[root@server ~]#
```

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

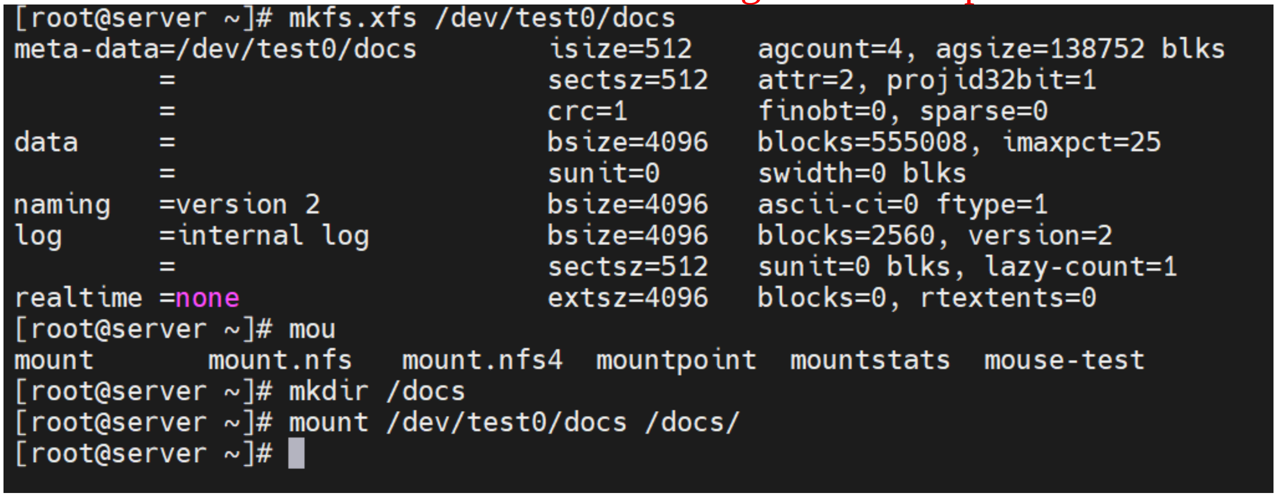


```
[root@server ~]# vgdisplay
  --- Volume group ---
  VG Name               data
  System ID
  Format                lvm2
  Metadata Areas        1
  Metadata Sequence No  2
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                1
  Open LV               0
  Max PV                0
  Cur PV                1
  Act PV                1
  VG Size               <20.00 GiB
  PE Size               4.00 MiB
  Total PE              5119
  Alloc PE / Size       2560 / 10.00 GiB
  Free  PE / Size       2559 / <10.00 GiB
  VG UUID               aUg8Xh-vgWM-w7lZ-ZcRp-9KIt-mlYe-qX0dKd

  --- Volume group ---
  VG Name               test0
  System ID
  Format                lvm2
  Metadata Areas        1
  Metadata Sequence No  3
  VG Access             read/write
```



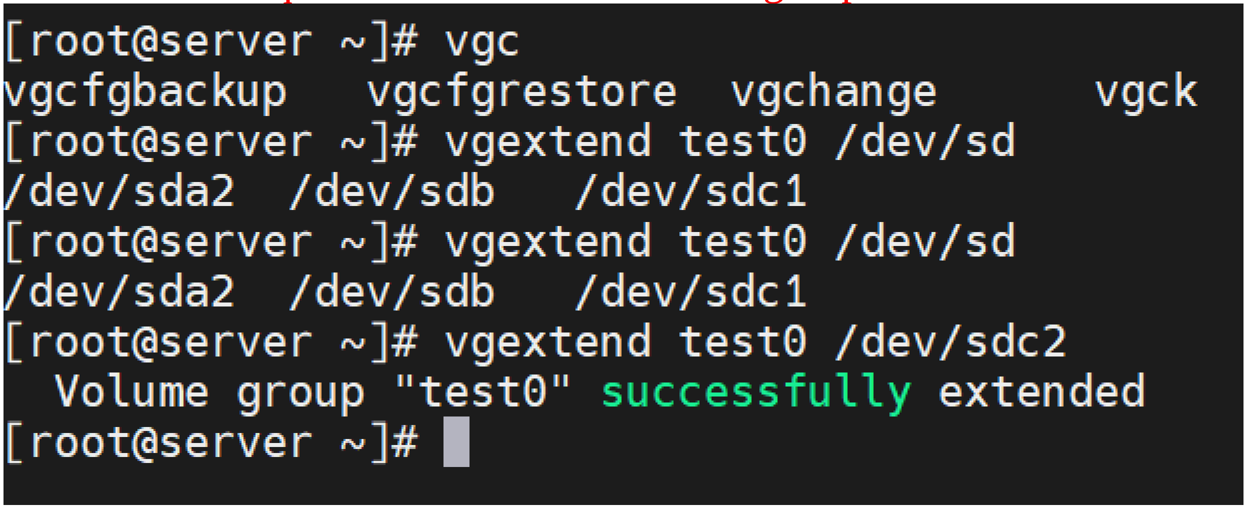
```
Open LV               1
Max PV                0
Cur PV                1
Act PV                1
VG Size               <3.00 GiB
PE Size               4.00 MiB
Total PE              767
Alloc PE / Size       767 / <3.00 GiB
Free  PE / Size       0 / 0
VG UUID               jiaUql-0SEh-VnBa-mHt0-7Knv-ICDc-AlUztn

--- Volume group ---
VG Name               centos
System ID
Format                lvm2
Metadata Areas        1
Metadata Sequence No  3
VG Access             read/write
VG Status             resizable
MAX LV                0
Cur LV                2
Open LV               2
Max PV                0
Cur PV                1
Act PV                1
VG Size               <49.00 GiB
PE Size               4.00 MiB
Total PE              12543
Alloc PE / Size       12542 / 48.99 GiB
Free  PE / Size       1 / 4.00 MiB
```

23. Create an xfs filesystem on the new LV, make a mount point called /docs and mount the docs LV using this mount point.



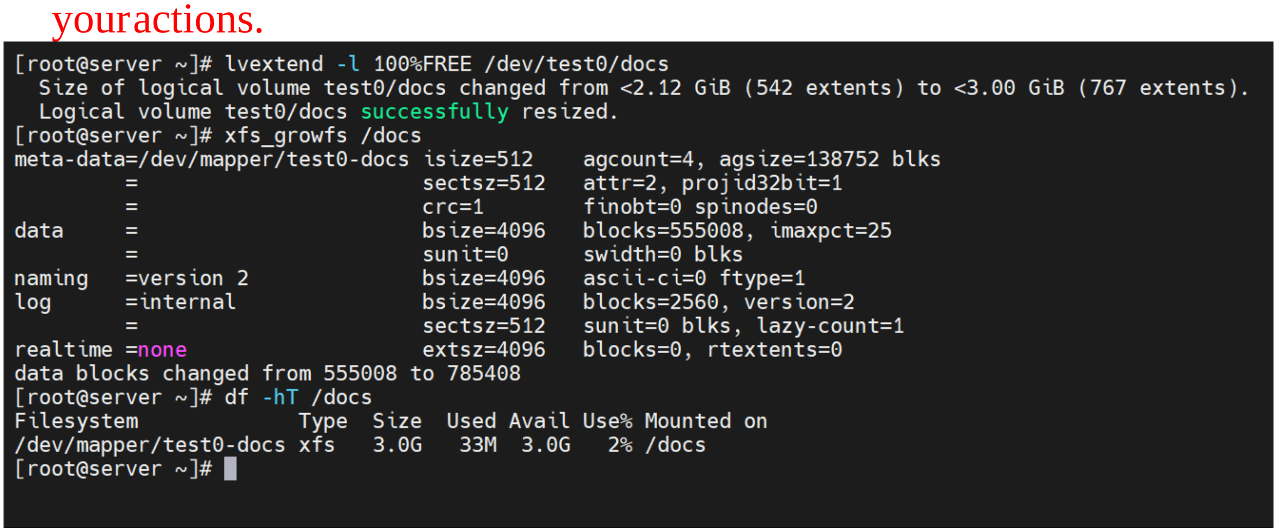
```
[root@server ~]# mkfs.xfs /dev/test0/docs
meta-data=/dev/test0/docs        isize=512    agcount=4, agsize=138752 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0, sparse=0
data     =                       bsize=4096   blocks=555008, 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
[root@server ~]# mou
mount        mount.nfs    mount.nfs4   mountpoint   mountstats   mouse-test
[root@server ~]# mkdir /docs
[root@server ~]# mount /dev/test0/docs /docs/
[root@server ~]#
```

24. Add all of the remaining unused physical volumes that you created in Sequence 1 to thetest0 volume group.



```
[root@server ~]# vgc
vgcfgbackup   vgcfgrestore  vgchange      vgck
[root@server ~]# vgextend test0 /dev/sd
/dev/sda2  /dev/sdb   /dev/sdc1
[root@server ~]# vgextend test0 /dev/sd
/dev/sda2  /dev/sdb   /dev/sdc1
[root@server ~]# vgextend test0 /dev/sdc2
  Volume group "test0" successfully extended
[root@server ~]# 
```

25. 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. Verify youractions.



```
[root@server ~]# lvextend -l 100%FREE /dev/test0/docs
  Size of logical volume test0/docs changed from <2.12 GiB (542 extents) to <3.00 GiB (767 extents).
  Logical volume test0/docs successfully resized.
[root@server ~]# xfs_growfs /docs
meta-data=/dev/mapper/test0-docs isize=512    agcount=4, agsize=138752 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0 spinodes=0
data     =                       bsize=4096   blocks=555008, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=1
log      =internal               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 555008 to 785408
[root@server ~]# df -hT /docs
Filesystem             Type  Size  Used Avail Use% Mounted on
/dev/mapper/test0-docs xfs   3.0G   33M  3.0G   2% /docs
[root@server ~]# 
```