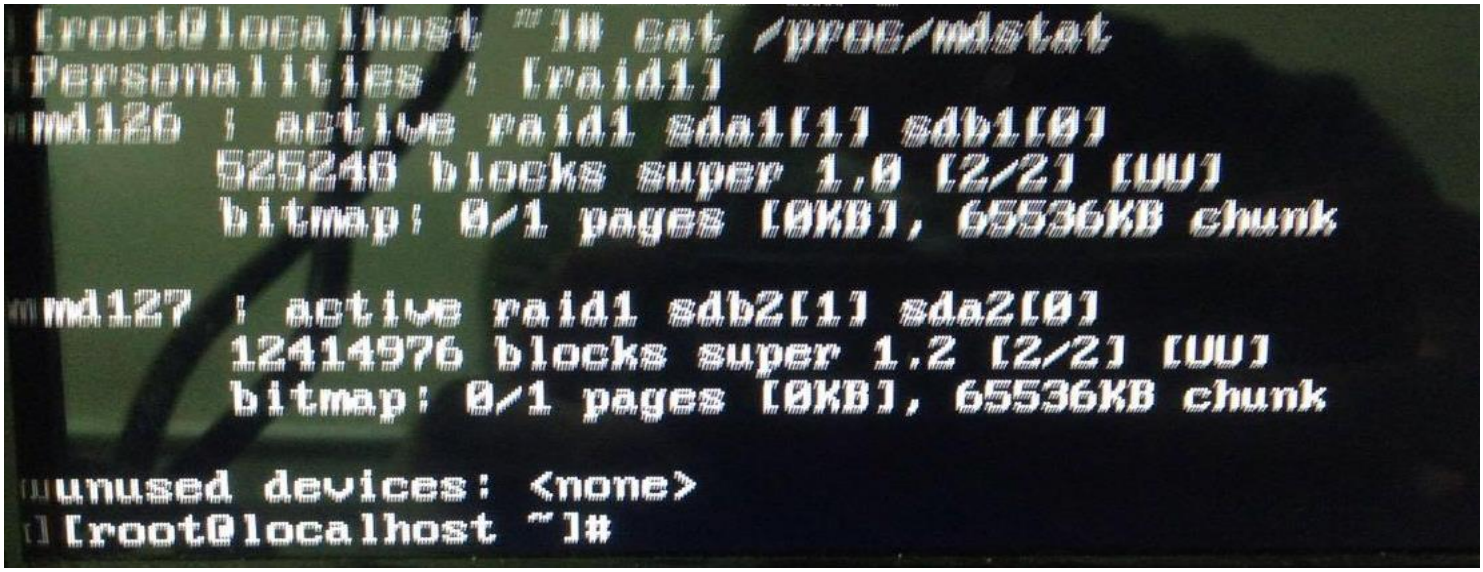
Manual addition of RAID1 partition

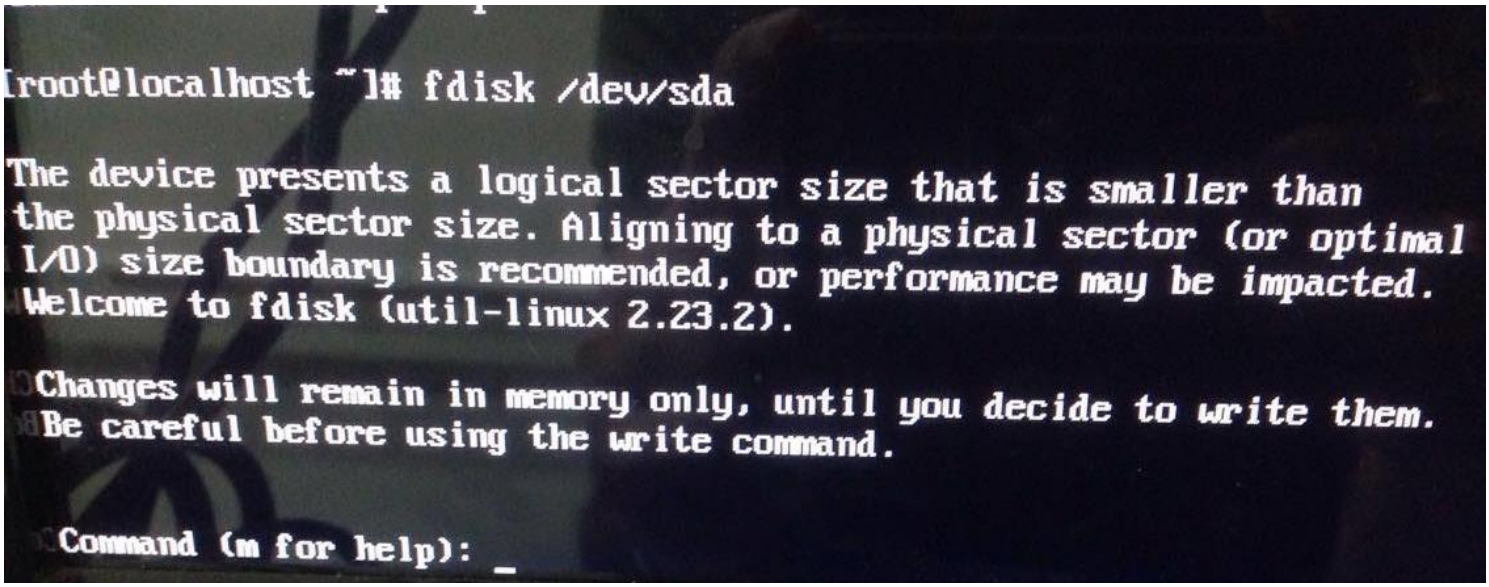
1st Step

Check first the status of RAID devices by using *cat /proc/mdstat*



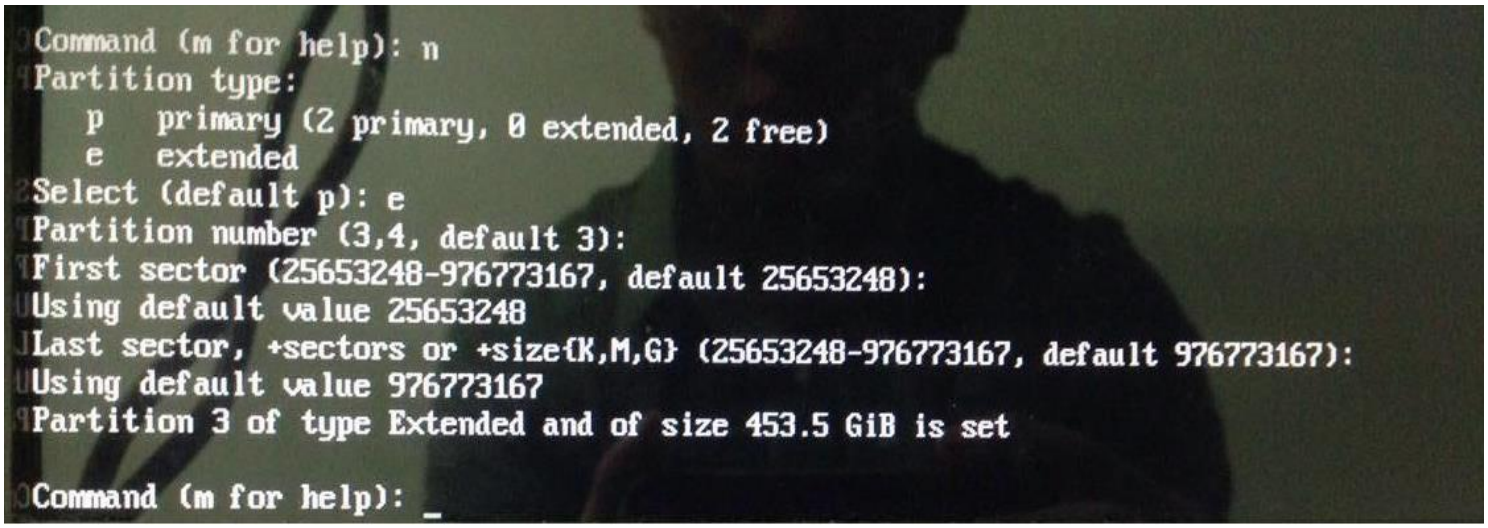

2nd Step

Use command *fdisk* for creating disk partitions including the available disk partition. To check the disk partitions, use the command *df.*



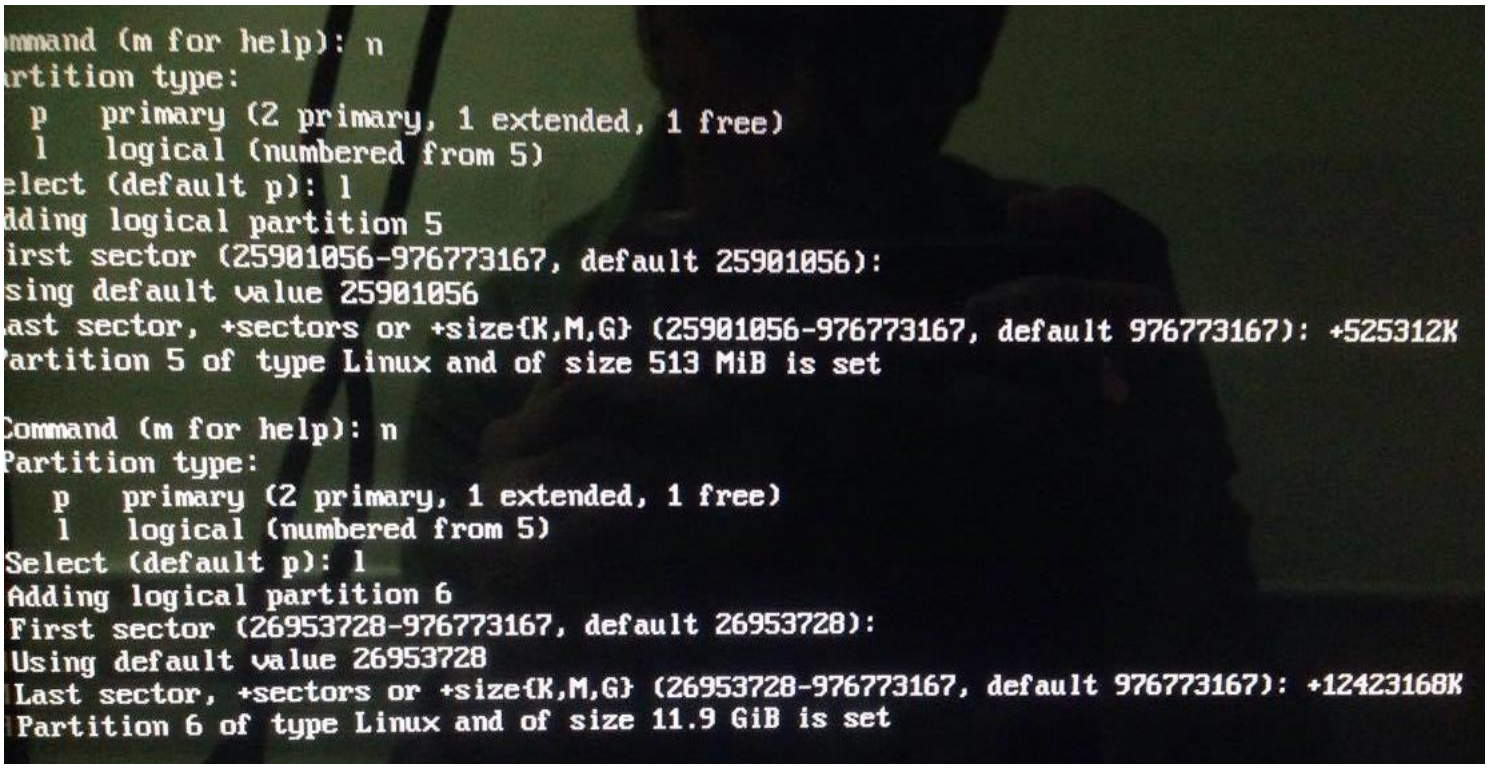
3rd Step

*Fdisk* has its own few commands that I used in creating partitions. To see all the *fdisk* commands type *m*.
Creating a new partition, I used the command *n*. The partition type to use is extended for me to be able to create 2 more partitions unlike the primary partitions which is only limited to 4. To automatically use the default sector I just leave out the first sector and use up all the remaining free space in disk partition /dev/sda by leaving out the last sector too.

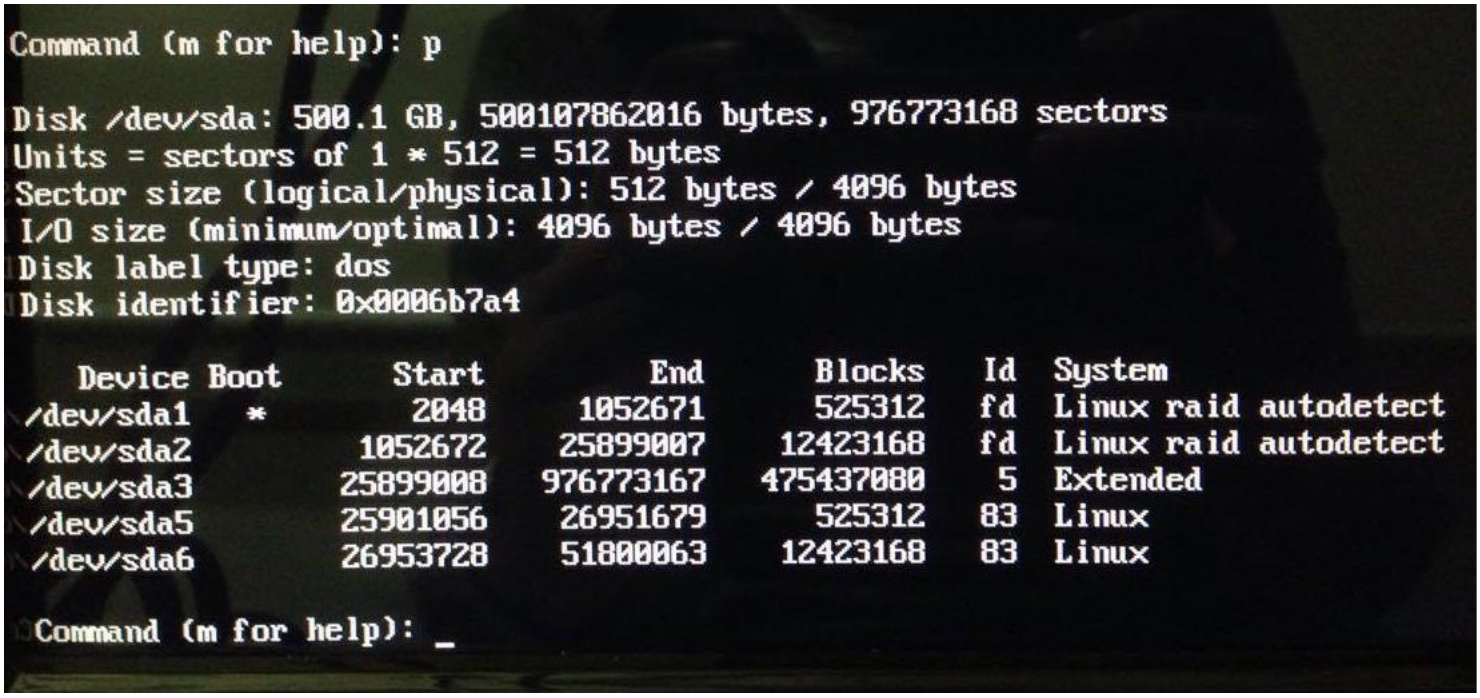



4th Step

Create 2 logical partitions, sda5 and sda6.

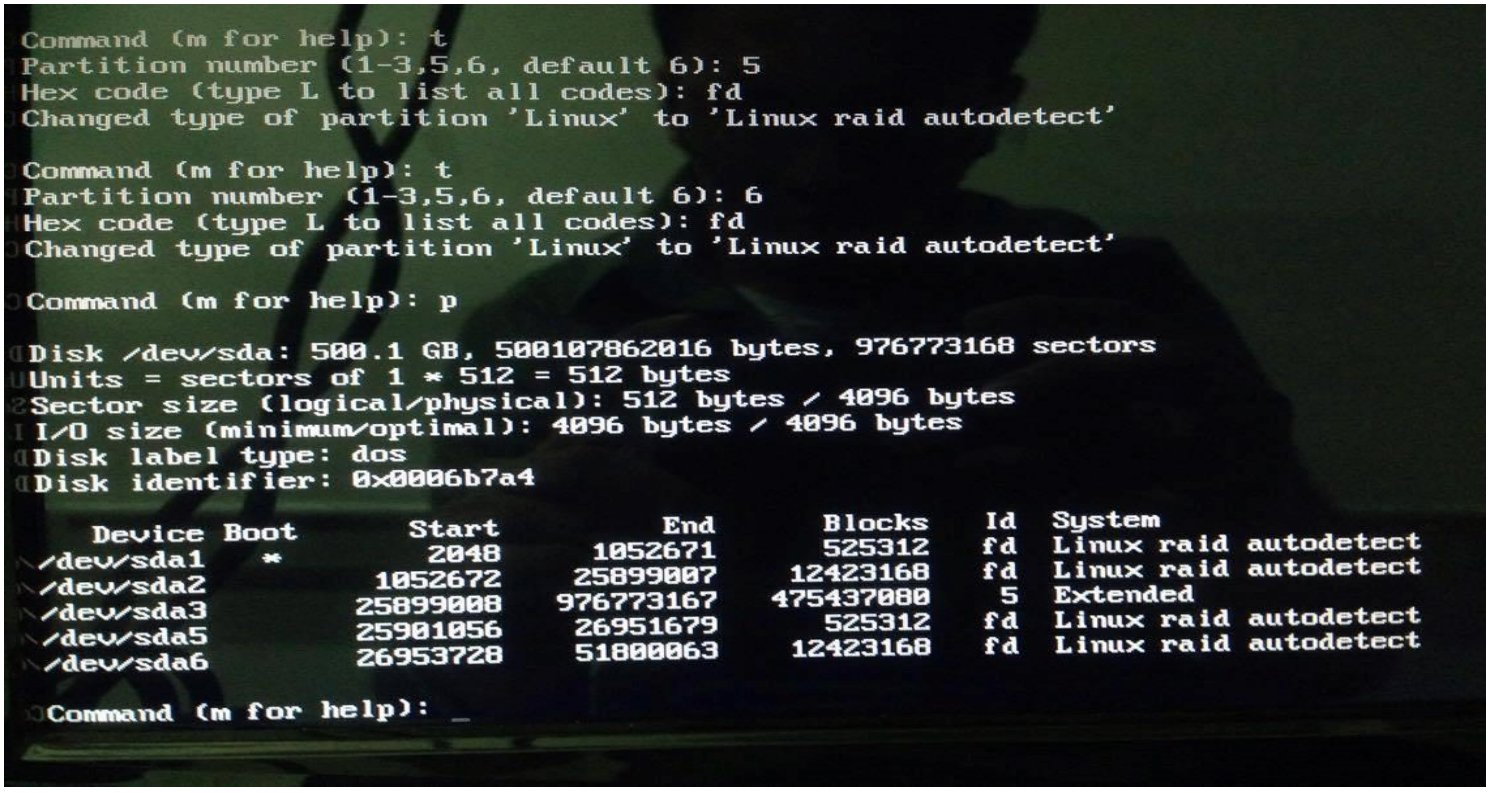


For me to check the partitions I just created I used the command *p* to print the partition table.

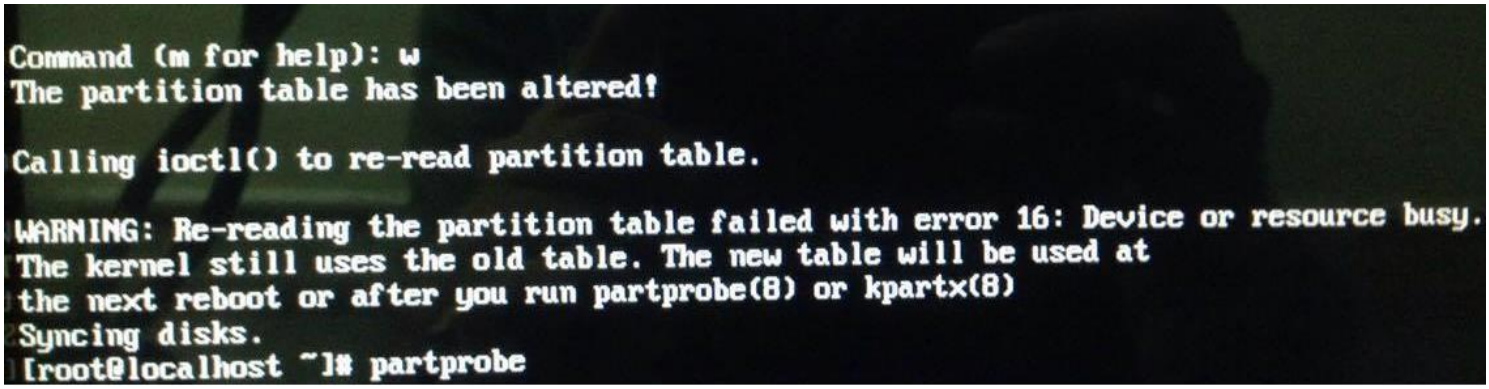



5th Step

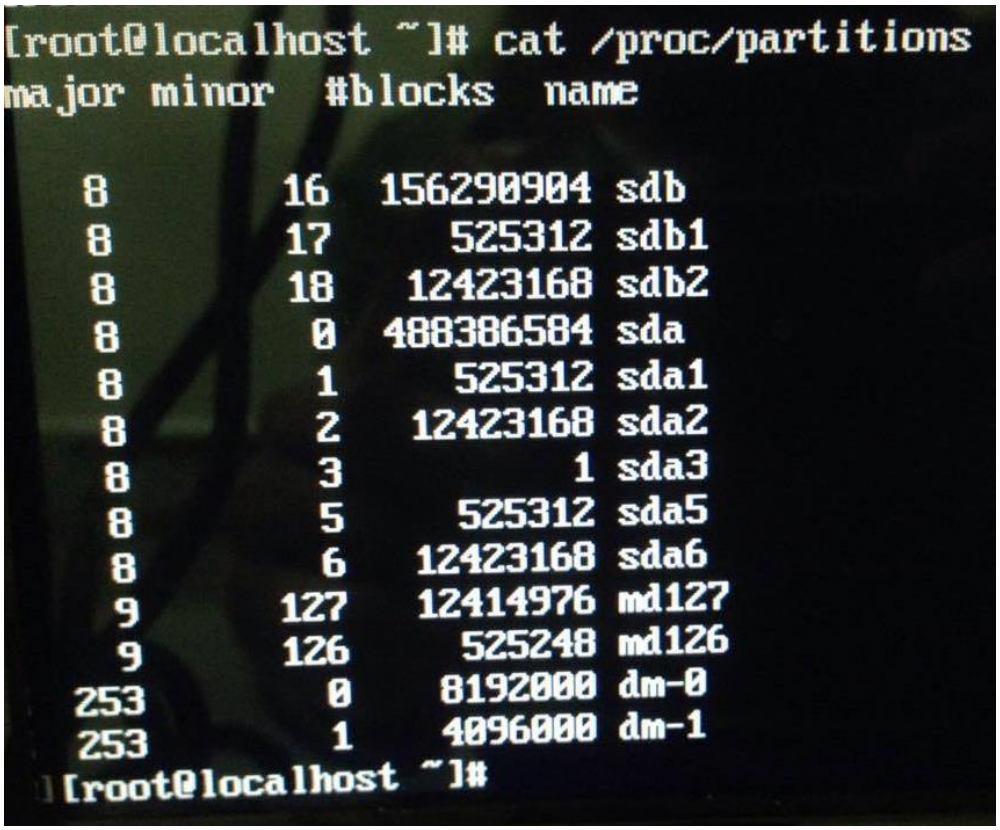
For me to be able to add the partition sda5 and sda6 to the RAID devices their partition type should be Linux raid autodetect. I used the *t* command then choose the partition number which is 5 & 6 then type *fd* in the hex code. Type L to see the hex codes for certain partition types. Then check again the partition table using *p.*



I used *w* to write table to the disk and exit. Instead of restarting the computer to see the results I use the command *partprobe* instead.

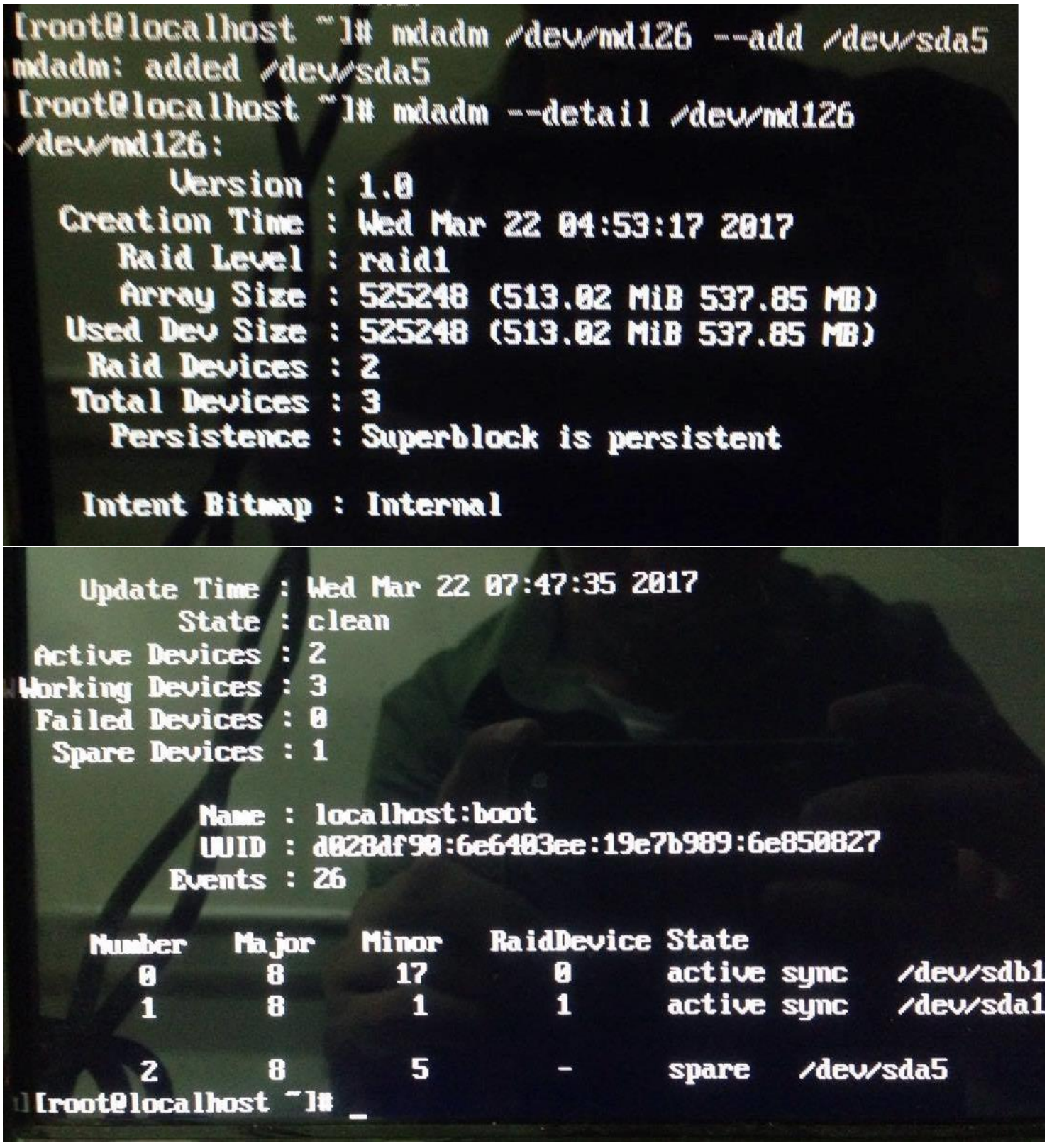



I used *cat /proc/partitions* to see the list of partitions available.



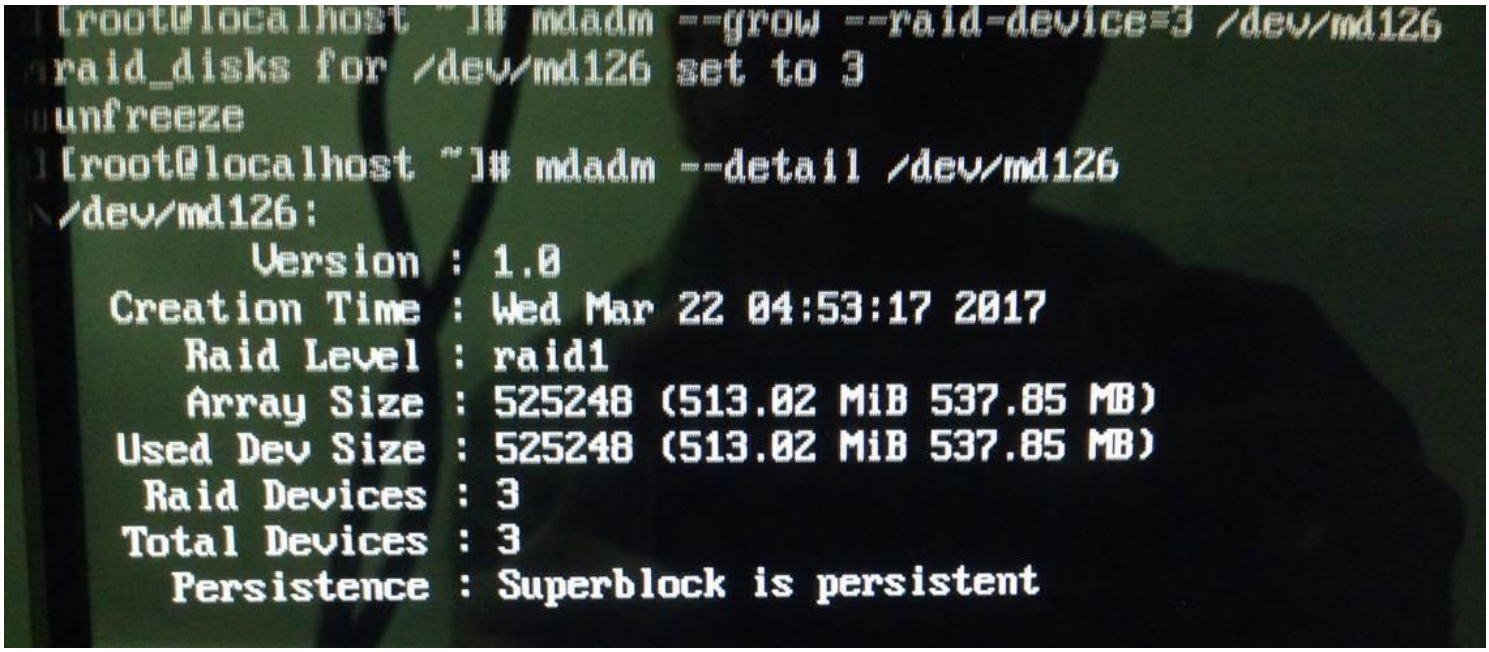
6th Step

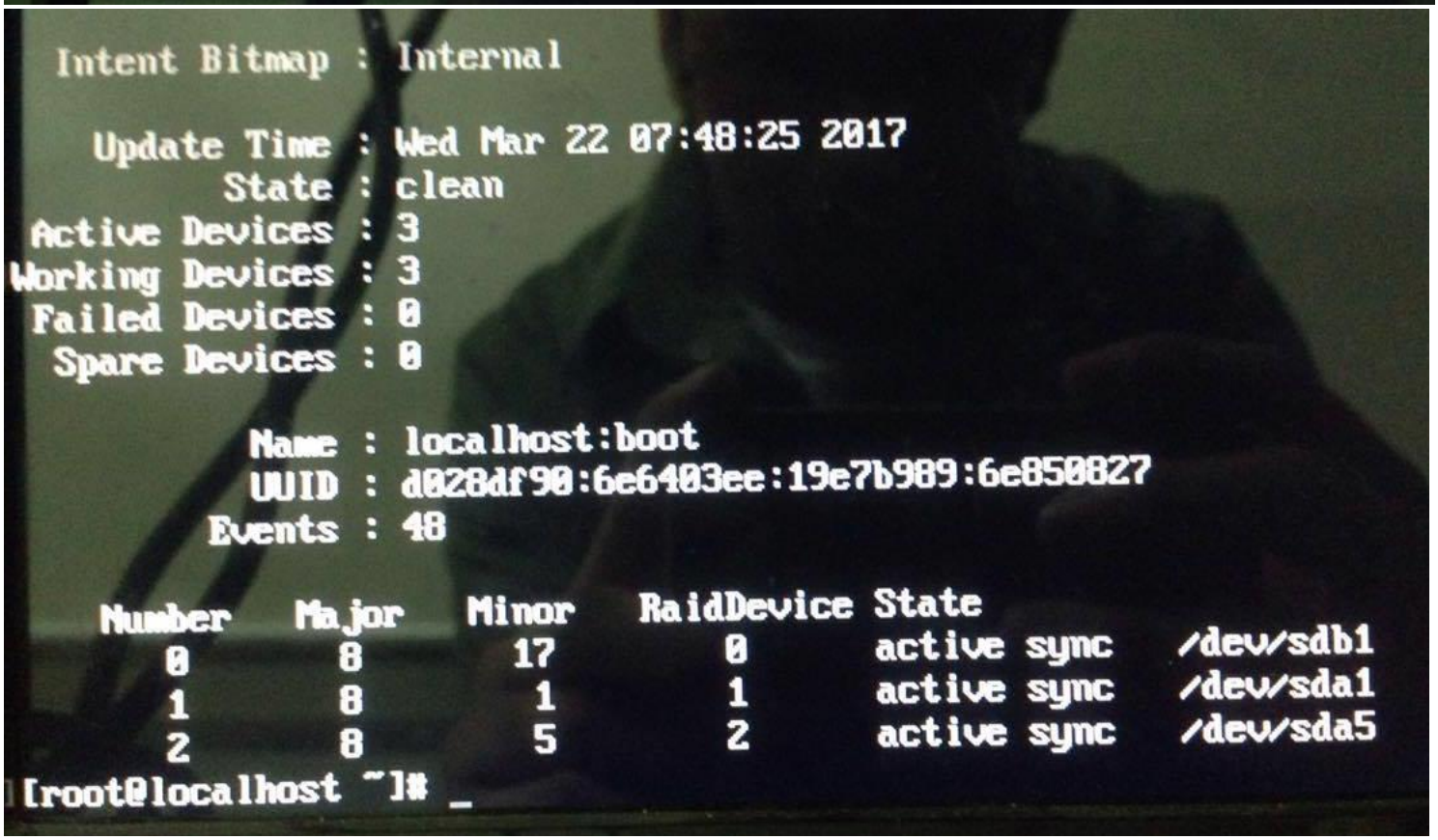
Looking at the output from *cat /proc/partitions*, I can now add my logical partitions to my existing RAID partitions using *mdadm --add.* Then check the status of the raid device by using *mdadm --detail.*



7th Step

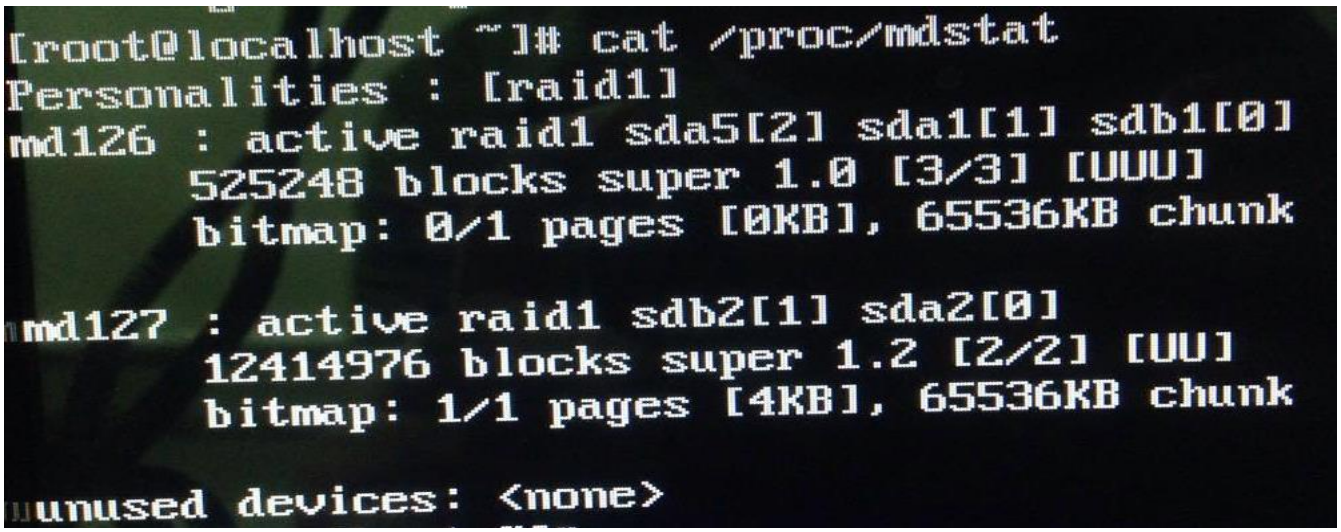
After checking the status of the raid device. I have to use *mdadm --grow --raid-device=3* to update the number of raid devices of the md126 from 2 to 3 for the sda5 to finally be added to the raid array.



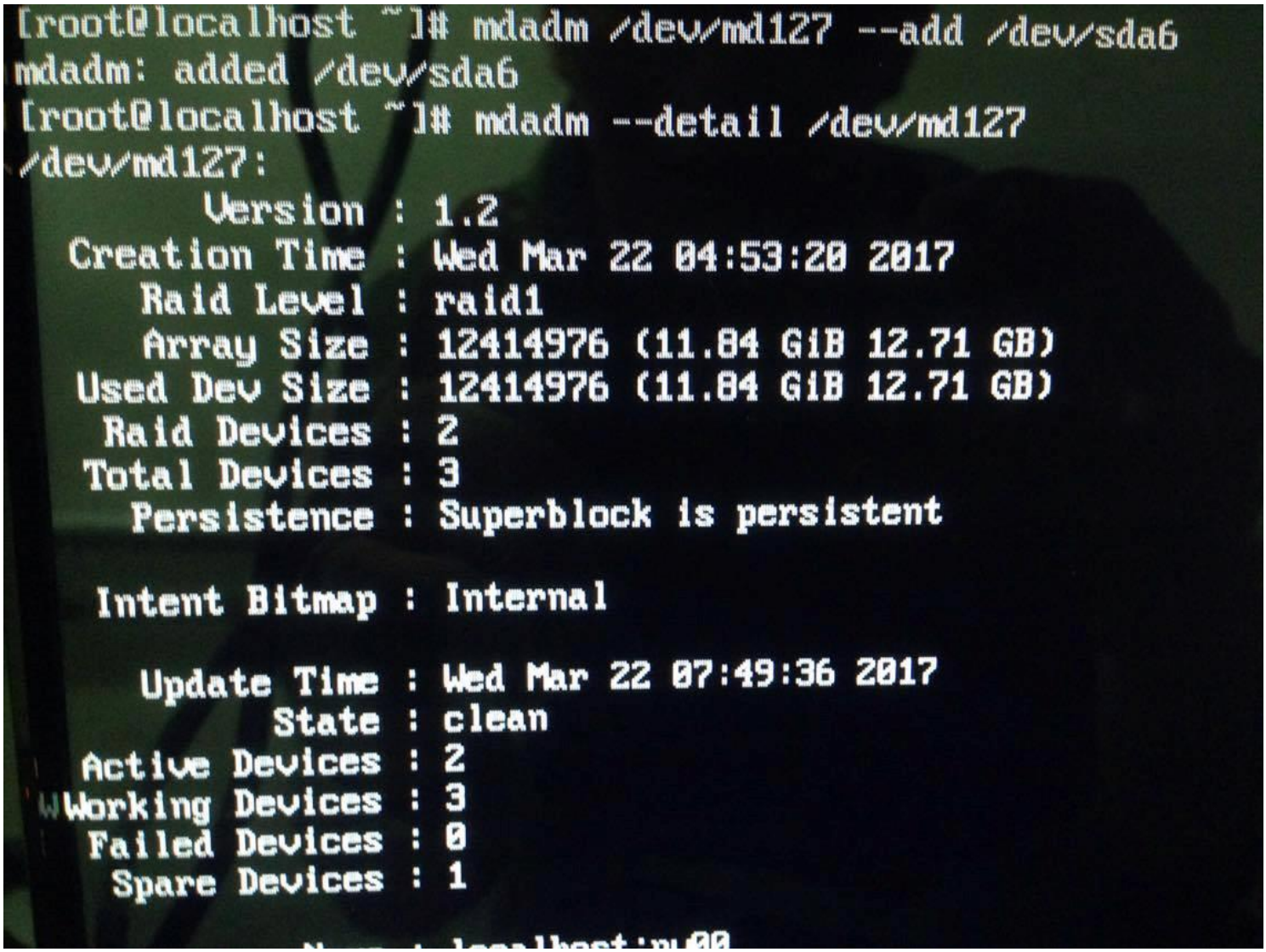



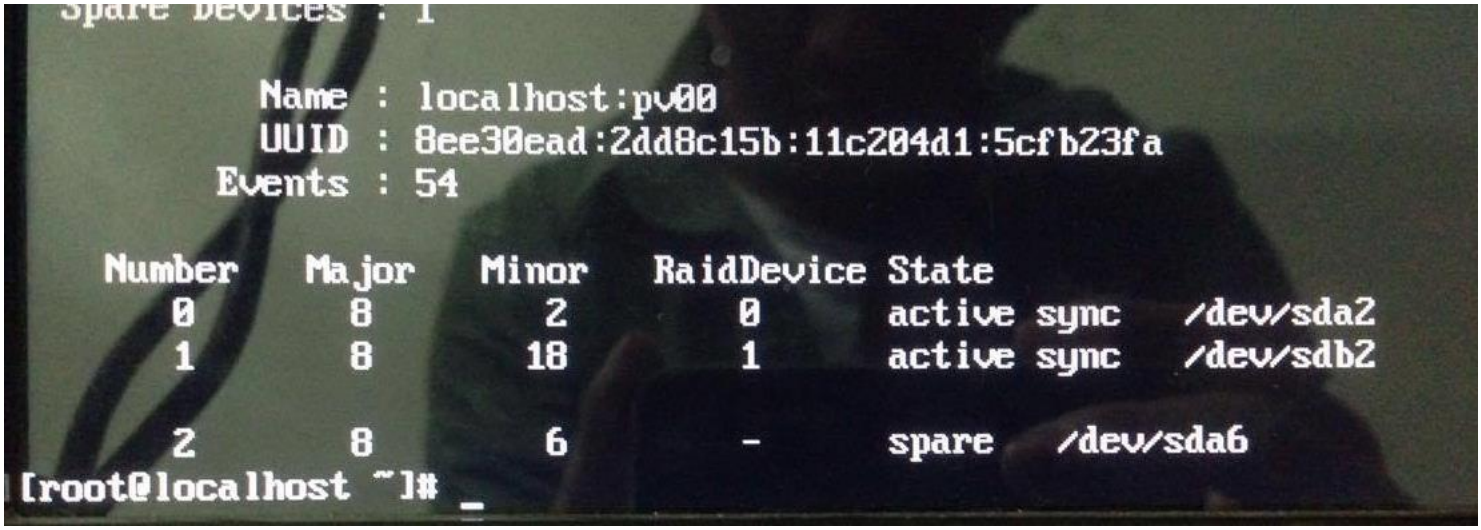
8th Step

Checking the status of the raid devices again using *cat /proc/mdstat*. Sda5 is now included in the raid array md126.

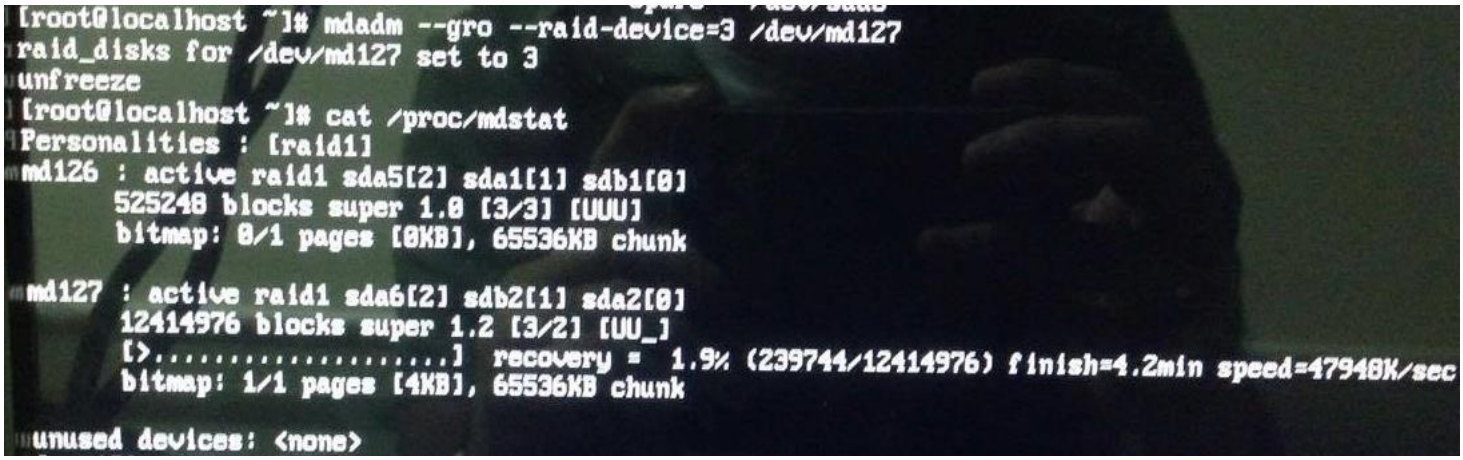



Adding logical partition sda6 to raid array md127 by repeating the same procedure and commands in adding the sda5 to raid array md126.

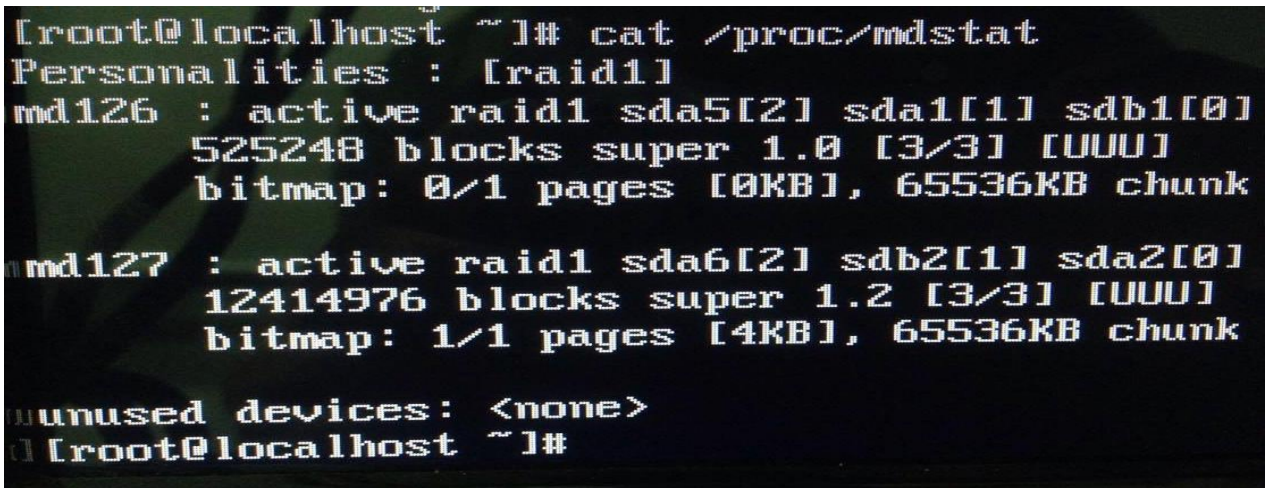





Checking again the status of raid devices by using *cat /proc/mdstat.* Entering the same command will output the status of the newly added raid array. If I use the command *watch* I will be able to monitor the status of the raid array that is being process.



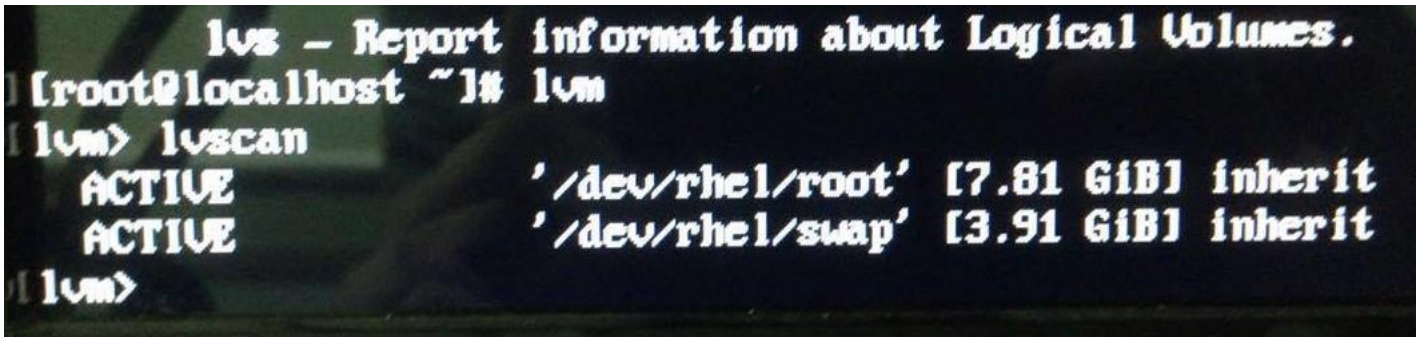

I have successfully added a new raid array to my raid devices.



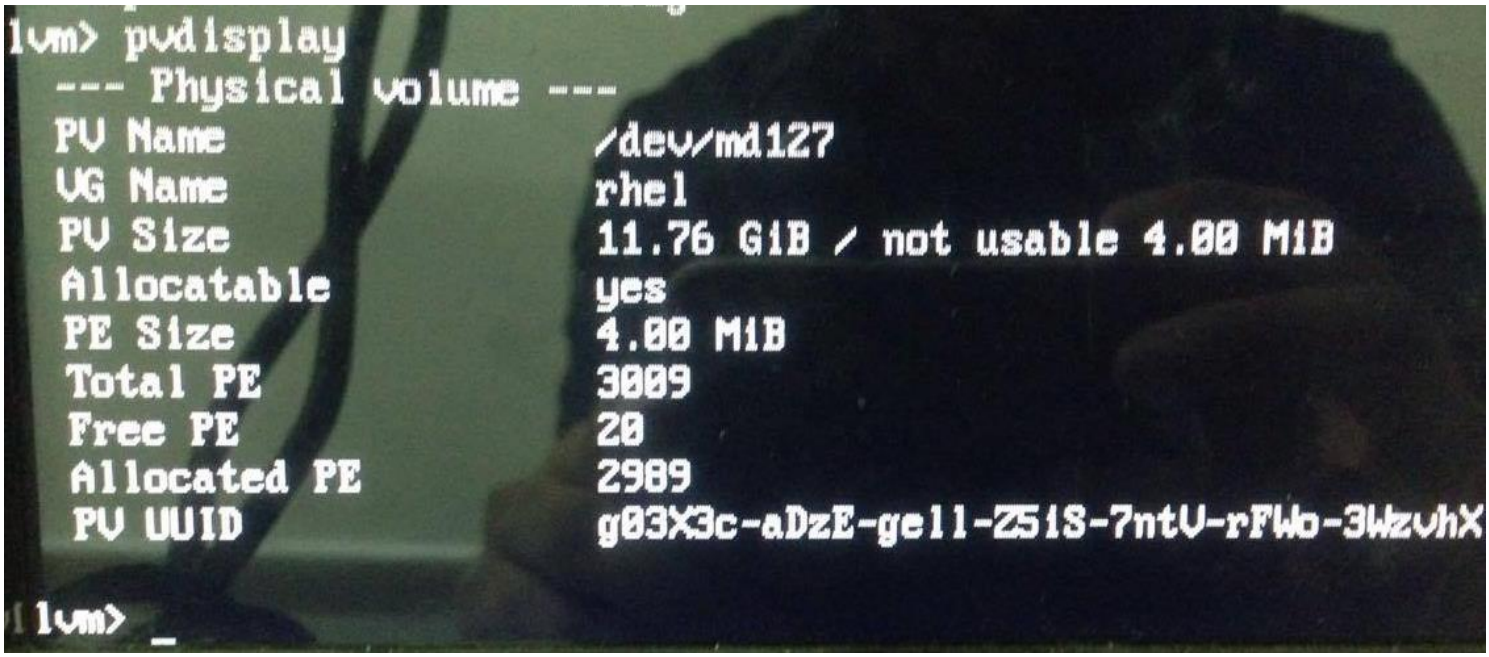
Manually expand the root directory from 8GB to 16GB using LVM

1st Step

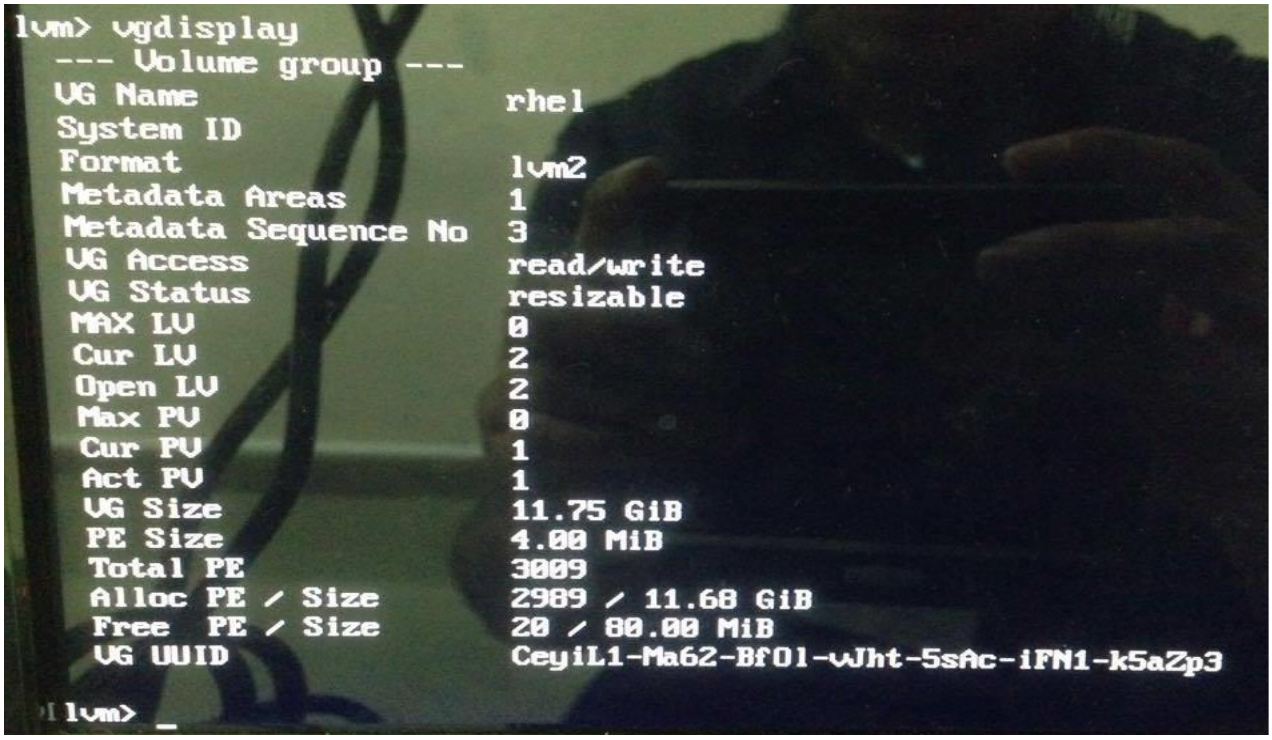
Enter the command *lvm* then check if there are existing logical volume by using the command *lvscan.*



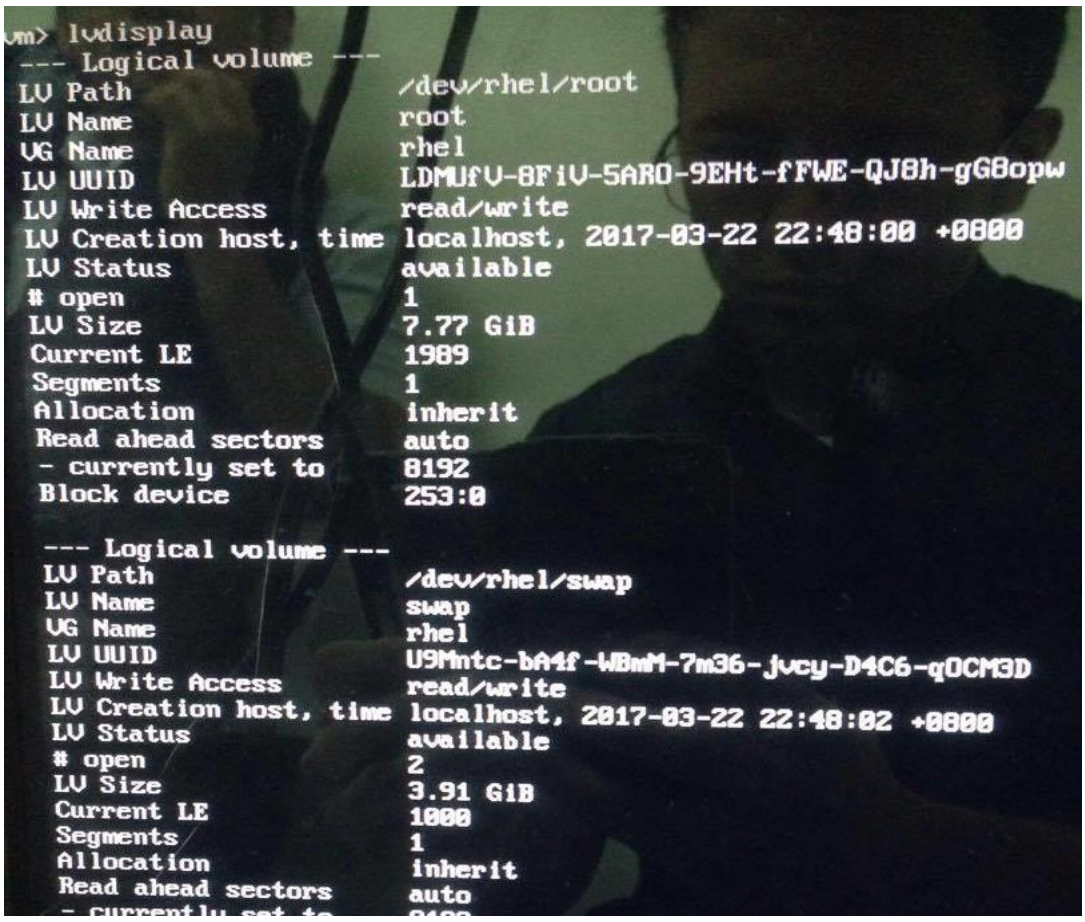

Check the physical volume by using *pvdisplay.*



Check the volume group by using the command *vgdisplay.*
Physical Extent number and size is 20 / 80.00 MiB here.




Check the logical volume by using the *lvdisplay.*



2nd Step

I use *fdisk* using a disk partition where there is still a free primary partition which is sdb.

```
[root@localhost ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.


Command (m for help): p

Disk /dev/sdb: 160.0 GB, 160041885696 bytes, 312581808 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 label type: dos
Disk identifier: 0x00050205

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1   *        2048     1052671      525312   fd  Linux raid autodetect
/dev/sdb2         1052672    25726975    12337152   fd  Linux raid autodetect

Command (m for help):
```

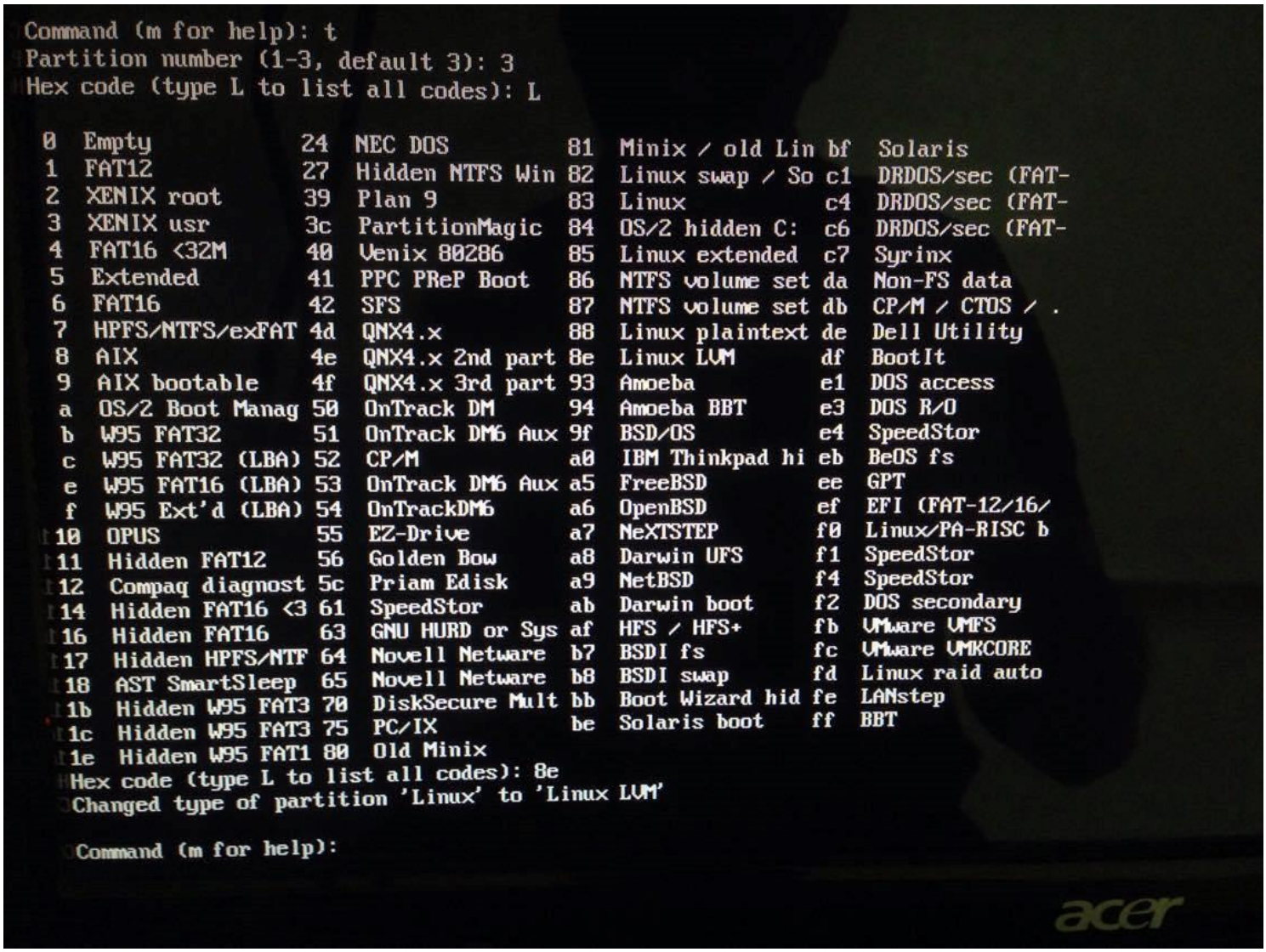
3rd Step

I created a new primary partition for me to be able to create a physical volume.

```
Command (m for help): n
Partition type:
   p   primary (2 primary, 0 extended, 2 free)
   e   extended
Select (default p): p
Partition number (3,4, default 3): 3
First sector (25726976-312581807, default 25726976):
Using default value 25726976
Last sector, +sectors or +size{K,M,G} (25726976-312581807, default 312581807): +8G
Partition 3 of type Linux and of size 8 GiB is set

Command (m for help): p

Disk /dev/sdb: 160.0 GB, 160041885696 bytes, 312581808 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 label type: dos
Disk identifier: 0x00050205

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1   *        2048     1052671      525312   fd  Linux raid autodetect
/dev/sdb2         1052672    25726975    12337152   fd  Linux raid autodetect
/dev/sdb3        25726976    42504191     8388608   83  Linux

Command (m for help):
```

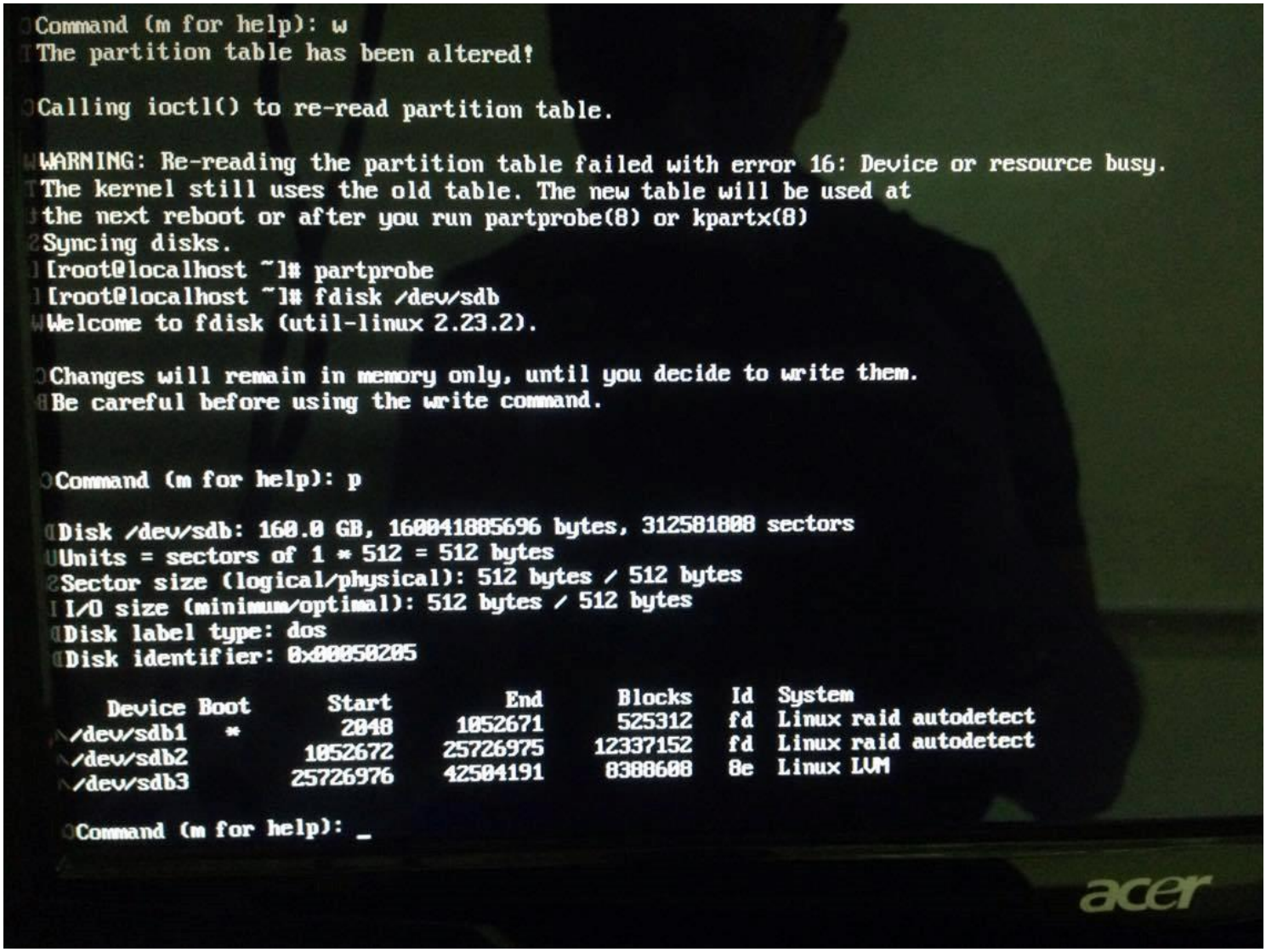
4th Step

Change the partition type of the newly created primary partition to Linux LVM using the *fdisk* command *t.* Enter *L* command to list the partition types and their hex codes.



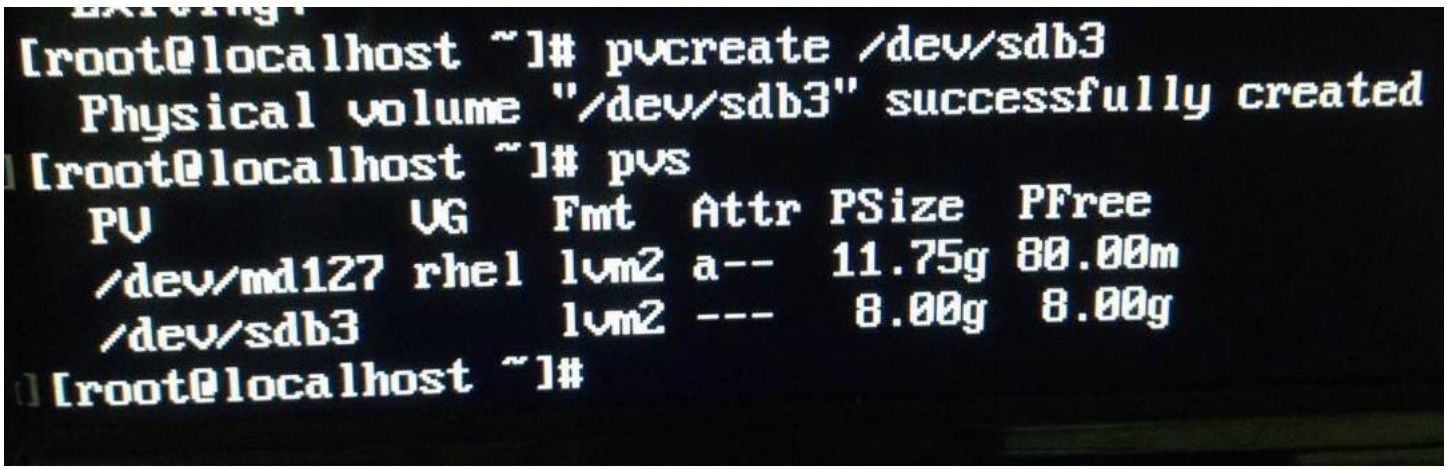
5th Step

I used the *fdisk* command *w* to write to the disk and quit. Then entered command *partprobe* so that I won't reboot the computer to see the changes I made. To check the partition type again of the newly created partition I used the *fdisk* and *p* command.



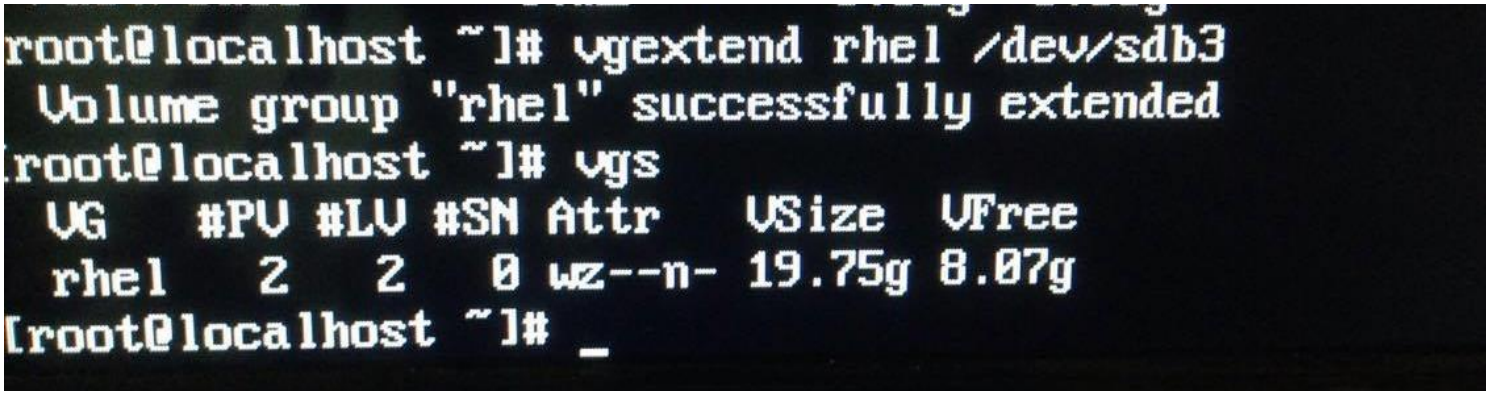

7th Step

I created a physical volume using the primary partition I just created. The command is *pvcreate*, then I used the *pvs* command to check the physical volumes.

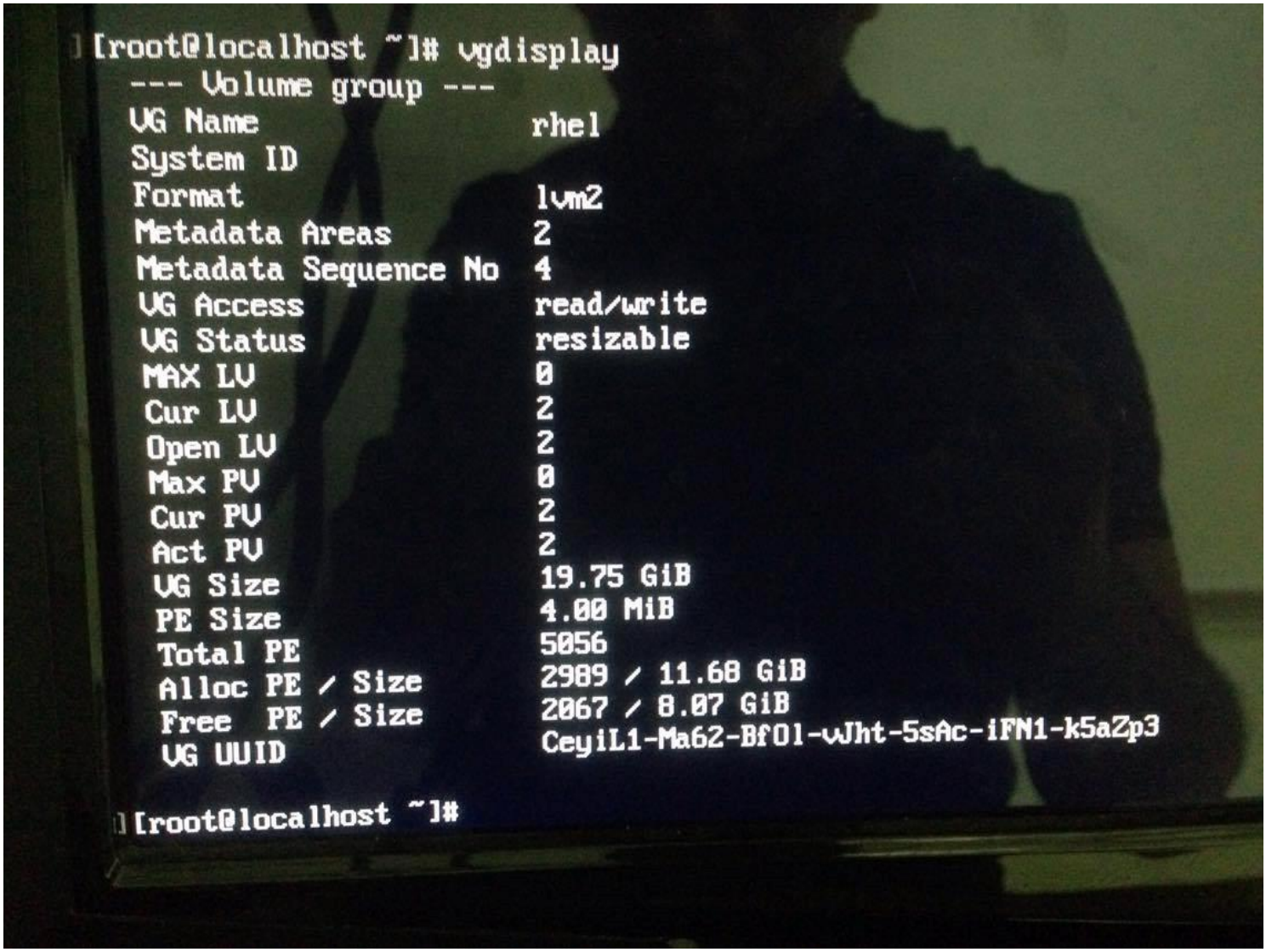


8th Step

I used the command *vgextend* to allocate size of the physical volume that I created which I will be using to expand the size of my logical volume root.




I used the command *vgdisplay* to check the number and size of the physical extents.

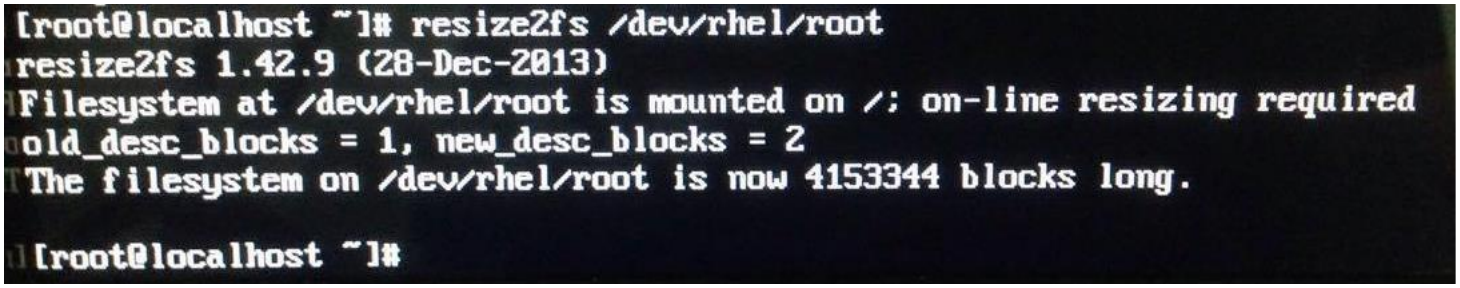


9th Step

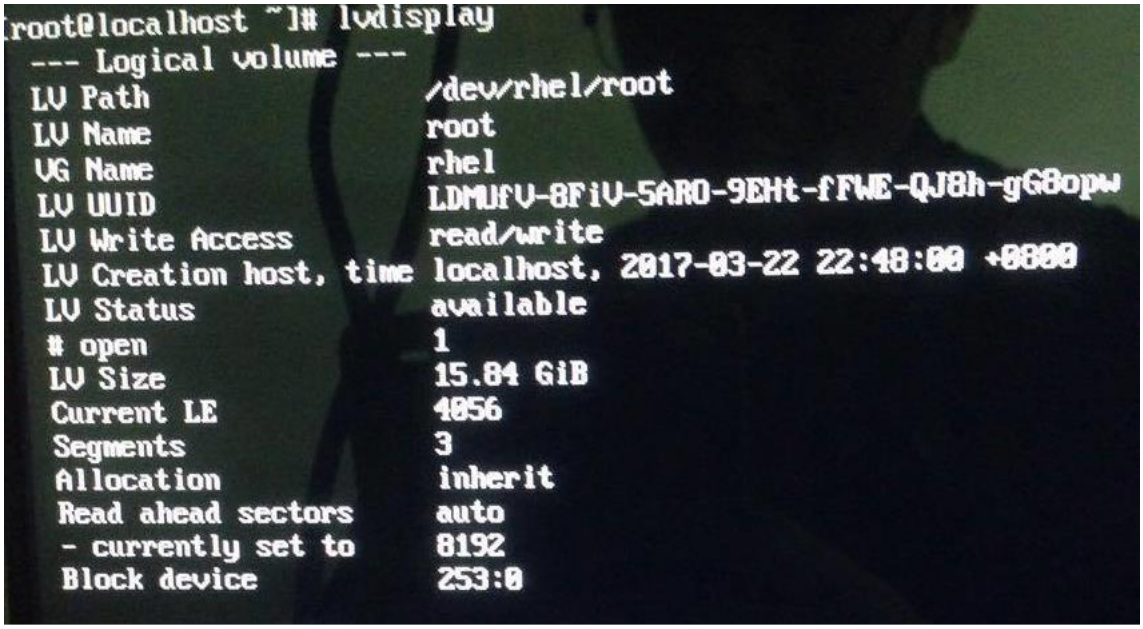
Now for me to extend the size of my logical volume root I had to use the command *lvextend -l.* The *-l* option is for using the PE or physical extent number of the free PE in the volume group.


10th Step

The resize2fs program will resize ext2, ext3, or ext4 file systems. It can be used to enlarge or shrink an unmounted file system located on device. If the filesystem is mounted, it can be used to expand the size of the mounted filesystem, assuming the kernel supports on-line resizing.




To check the logical volume root I used the command *lvdisplay.*



Checking the volume group, physical volume and logical volume using *pvs, vgs* and *lvs.*