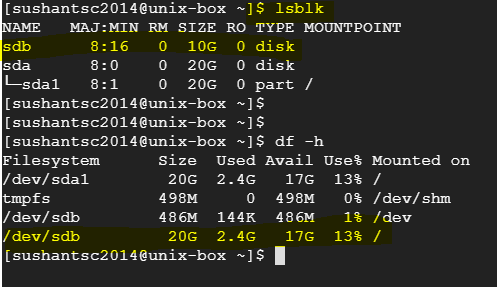
**Managing physical storage devices**

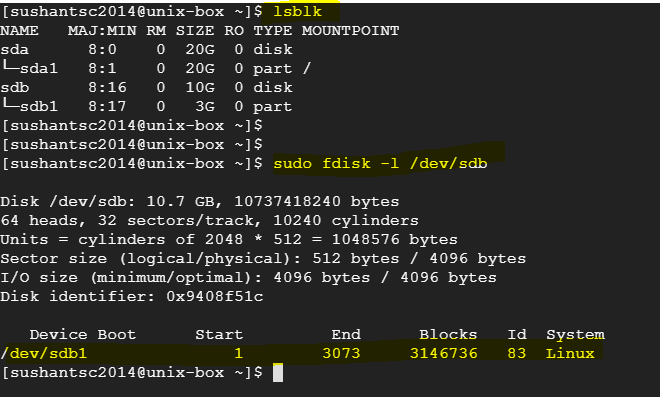
Attaching disk in GCP and mounting it

Command: mount /dev/sdb /

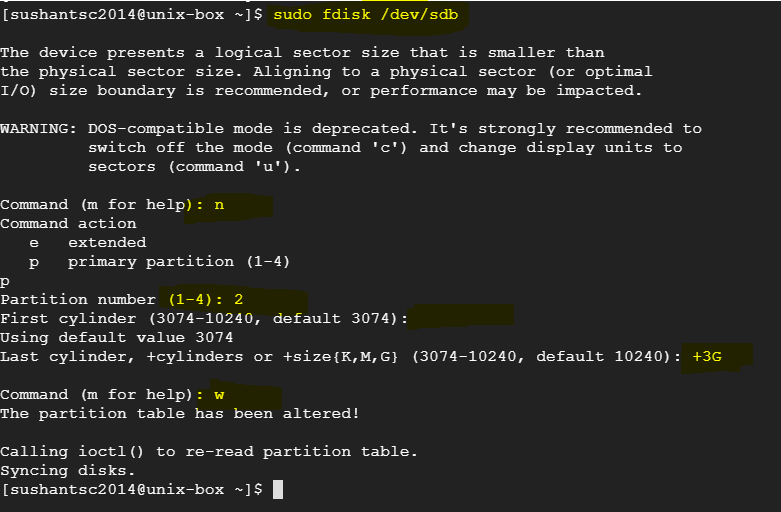


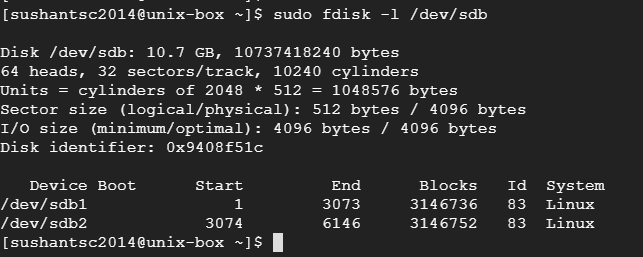
**DISK PARTITIONING-**

The process of dividing a disk into logical areas that can be worked with separately is called partitioning. Disk partitioning is done to subdivide the disk into pieces with broadly different purposes



Creating 2nd partition of 3 GB





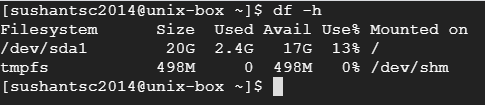
**File systems mounting**

To format the partitions with filesystems, you can make use of the **mkfs** command.

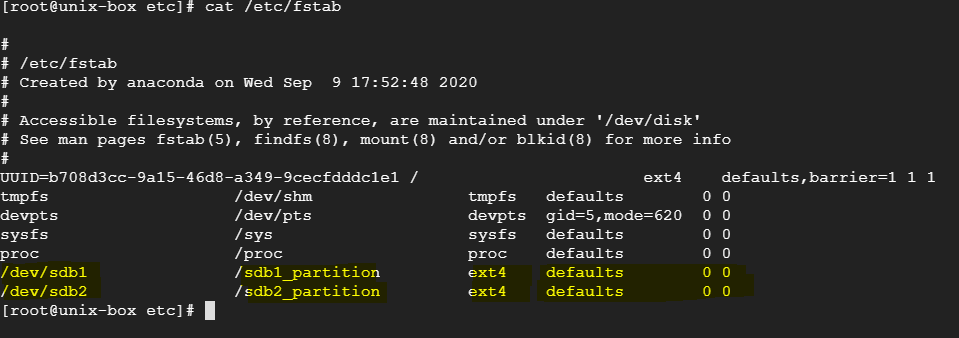
That is files will be of type ext4

mkfs.ext4 /dev/sdb1\_partition

mkfs.ext4 /dev/sdb2\_partition



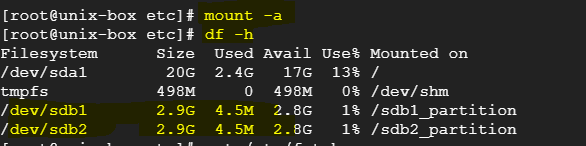
When you use ‘mount’ command- file systems are temporarily mounted. To permanent mount then you need to edit ‘/etc/fstab’ file



You need mention 6 fields

(use ‘man fstab’ to get to know field values)

Then use command ‘mount –a’. This command re-reads the filesystem table and mounts the filesystems that are not mounted. This file is always read when the Linux OS is booting.



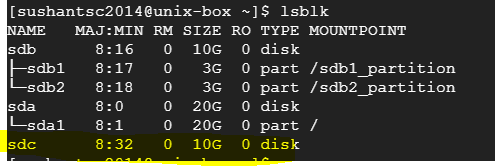
**Creating logical volumes**

Suppose, /sdb2\_partition is full and we don’t have space left on ‘/dev/sdb’ disk. And also, we have third disk ‘/dev/sdc’ (with partition /dev/sdc1)

So, we can create a virtual disk from using space from **/dev/sdb2** and **/dev/sdc2** (using space from two different disks.)

Physical volume (create from several disks) 🡪 volume group (group of physical volums) 🡪 logical volume

A volume group consists of one or more **physical volumes**. An entire hard disk or traditional partitions can be converted to physical volumes. These physical volumes can then be pooled together to constitute the volume group. The smallest unit of a volume group is called a **physical extent**and the size of a physical extent can be defined while creating the volume group.

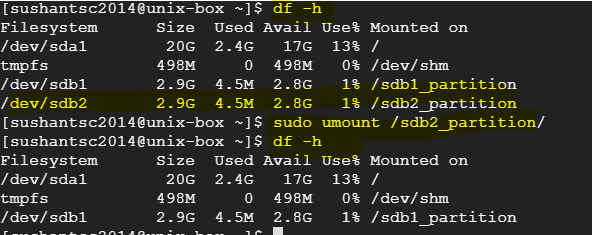


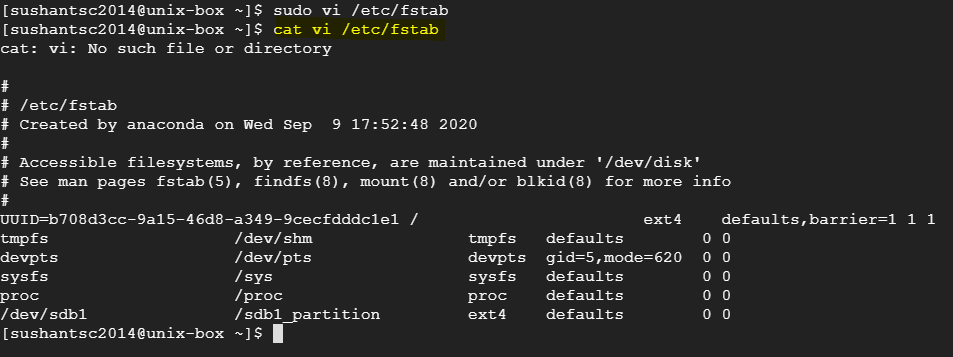
Create partition from disk C of full size

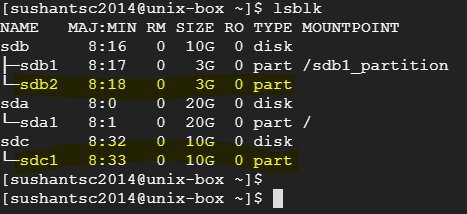


We will create logical volume from /dev/sdb2 and /dev/sdc1.

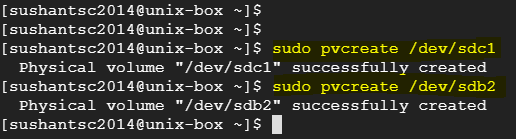
Unmount /dev/sdb2 (you can take backup of /dev/sdb2/partition). Remove entry from /etc/fstab file as well.

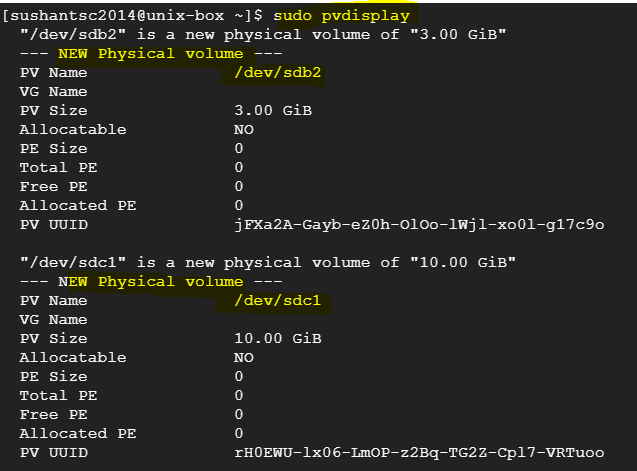






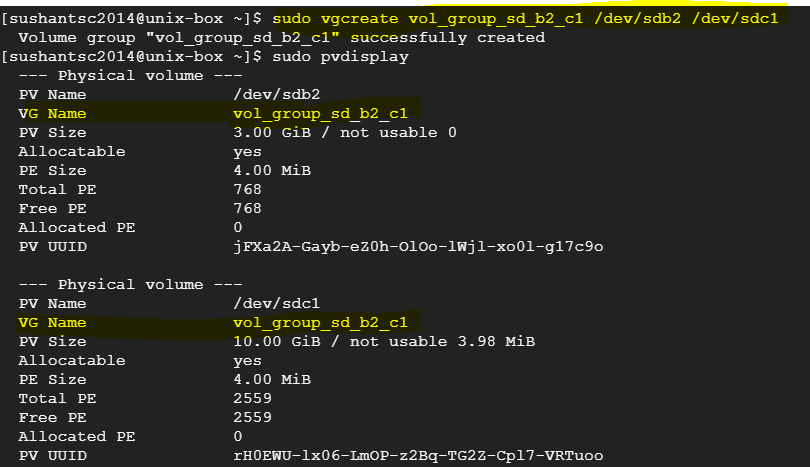
1. The /dev/sdb1 partition and /dev/sdc1 partition can be converted into **physical volumes** using the **pvcreate** command. **pvdisplay** is command to display Physical volumes created.

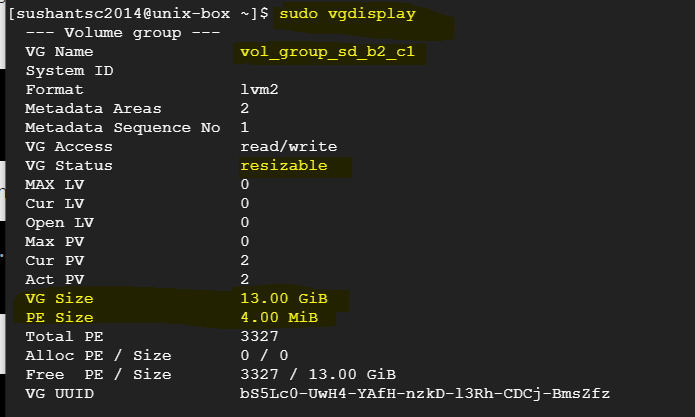




1. The next step is to create a volume group comprising the two physical volumes that were created in the previous step. The **vgcreate**command is used to create a volume group. It is necessary to give a name to the volume group. The **vgdisplay** command is used to display info about volume group. The **vgextend** command is used to add physical volume to existing volume group.

/dev/sdb2 (3 GB) + /dev/sdc1 (10 GB) = volume group of 13 GB





1. The next step is to divide the volume group into a logical volume. A logical volume can be created using the **lvcreate**command as given below. We should give logical volume name and volume group. The **lvdisplay** command used to display info.

The **lvextend,lvreduce,lvresize** commands are used to alter size of logical volume. If we have free space available on existing logical volume (**vgdisplay** command will show free space), we can extend/resize logical volume. Mostly use **lvresize.**

sudo lvcreate -n logical\_vol\_sd\_b2\_c1 -L 12.8G vol\_group\_sd\_b2\_c1

-n : name of logical volume

-L : size of logical volume, we can make full size available or any size less than available (10+3 GB)

: mention volume group

sudo lvresize –r –L +2G /dev/vol\_group\_sd\_b2\_c1/logical\_vol\_sd\_b2\_c1 🡪 to add 2 GB in logical volume

-r: to re-size volume group as well.

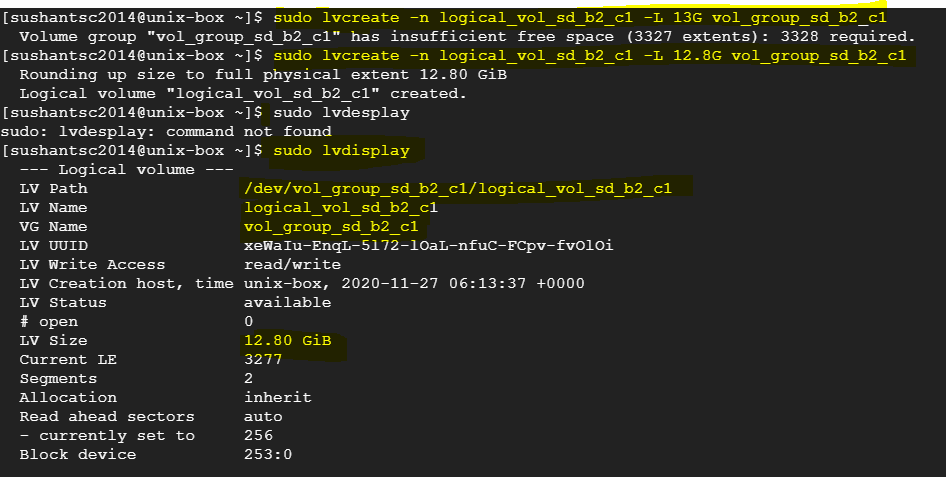
-L +2G : Add 2 GB

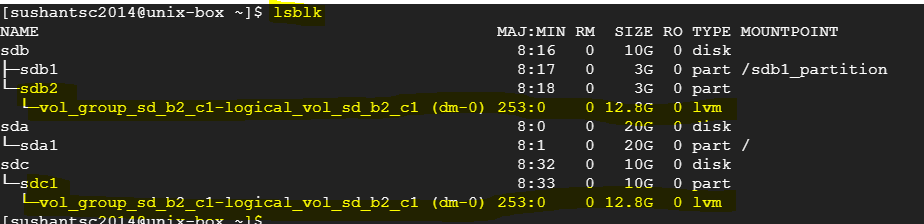
-l -2G : reduce by 2 GB

In **lvdispaly** command, we can find logical volume path created. **(/dev/<vol\_group\_name>/<logical\_vol\_name>)**

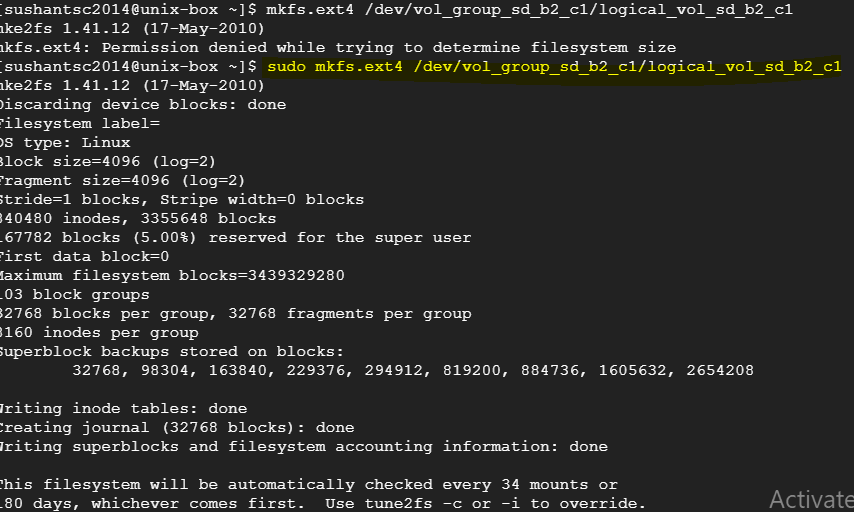
Logical volume created: /dev/vol\_group\_sd\_b2\_c1/logical\_vol\_sd\_b2\_c1

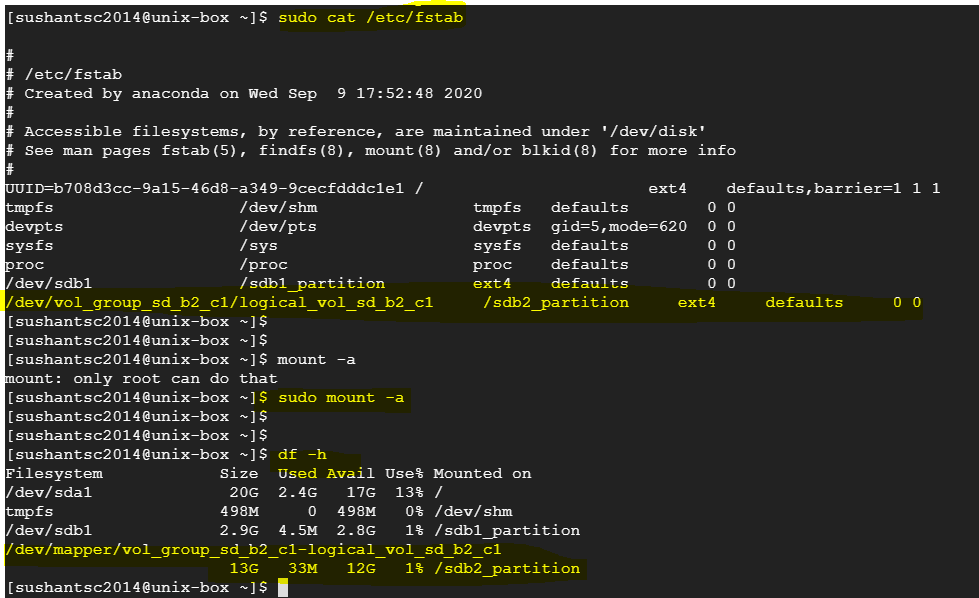
You can also verify running command ‘**lsblk’**



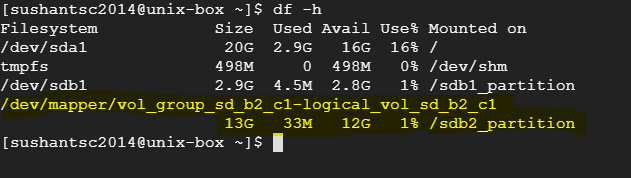


1. Mount this new disk. Format it with a file system and mount it. Mount on point /sdb2\_patition. Make entry in /etc/fstab and then command ‘mount –a’ to mount all unmounted disks. Run command ‘df –h’ to verify.





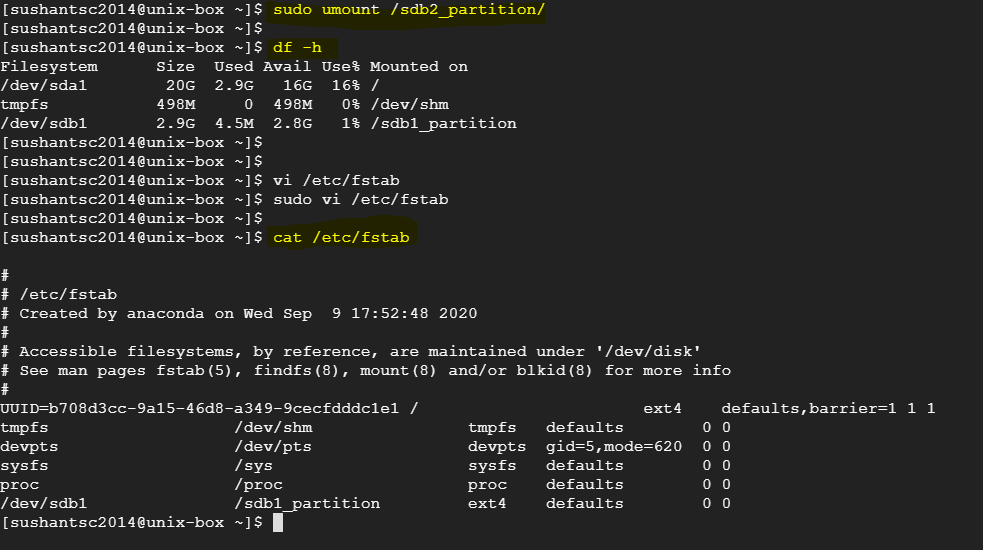
**Requirement**- Need additional 1 GB space on ‘/sdb2\_partitio’



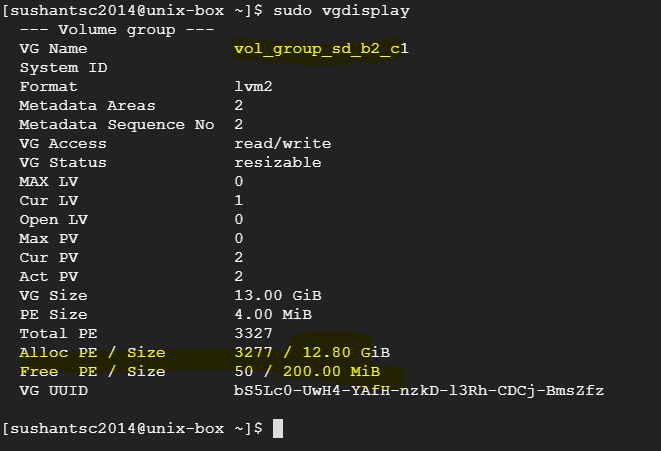
As we can see, ‘/sdb2\_partition’ has already 13 GB. And we want to take it up to 14 GB.

As we know, this is a logical volume space.

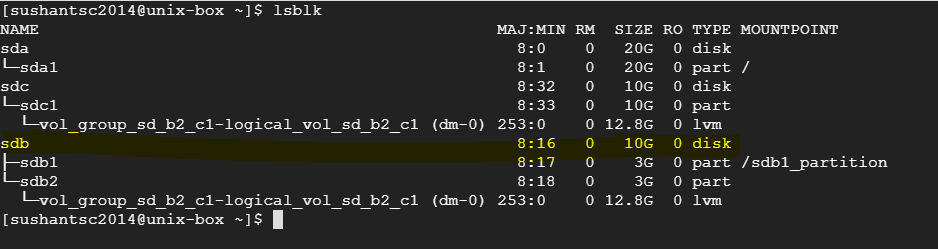
1. Let us first unmout this ‘/sdb2\_partition’. Remove entry from fstable as well.

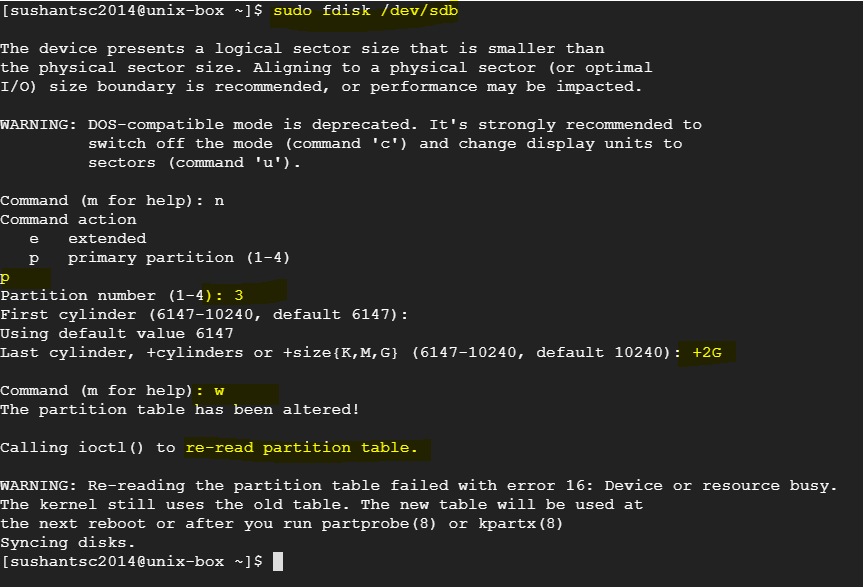


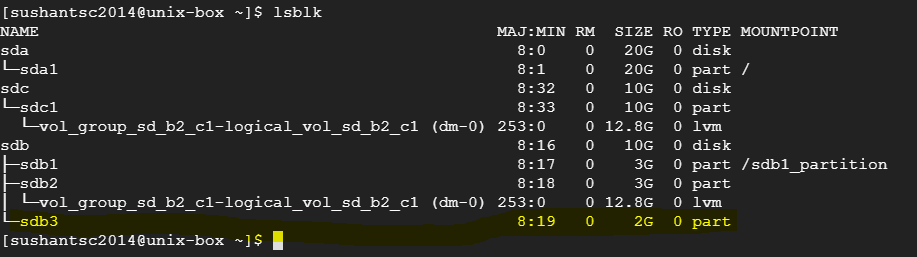
1. We can use command ‘lvresize’, but we should have enough free space on volume group. So, let’s check volume group info ‘vol\_group\_sd\_b2\_c1’. As we can see, there is no free space available.



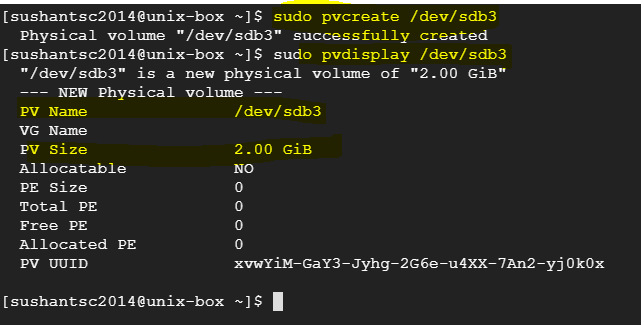
1. We need to create physical volume and add it to above group to resize logical volume. So we will create new partiion /sdb3 in ‘/dev/sdb’ of 2 GB. We have 4 GB left after two partitions of 3 GB each.





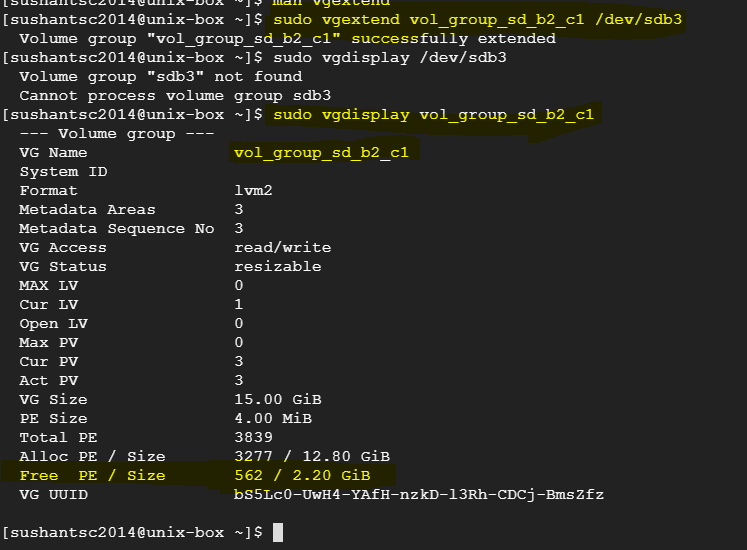


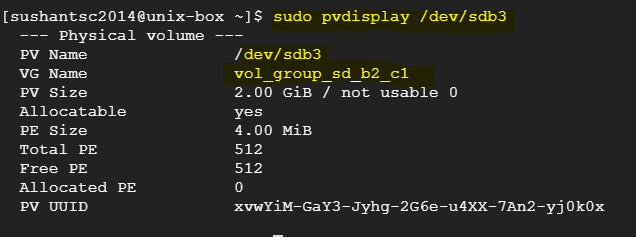
1. Create physical volume out of /dev/sdb3



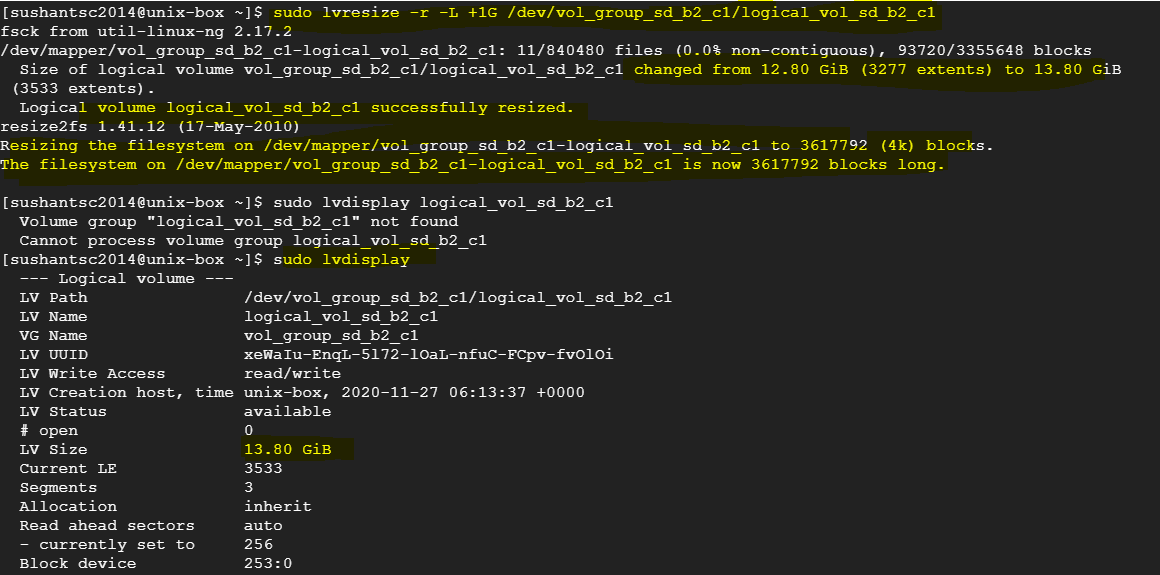
1. Now, we have volume group ‘vol\_group\_sd\_b2\_c1’. We will add ‘/dev/sdb3’ partition to this volume group.

Check if ‘/dev/sdb3’ is part of ’vol\_group\_sd\_b2\_c1’ group.





1. Now we can see additional 2 GB space available on volume group ’vol\_group\_sd\_b2\_c1’. We can use ‘lvresize’ command to resize logical volume. As per our requirement we need only 1 GB of additional space.

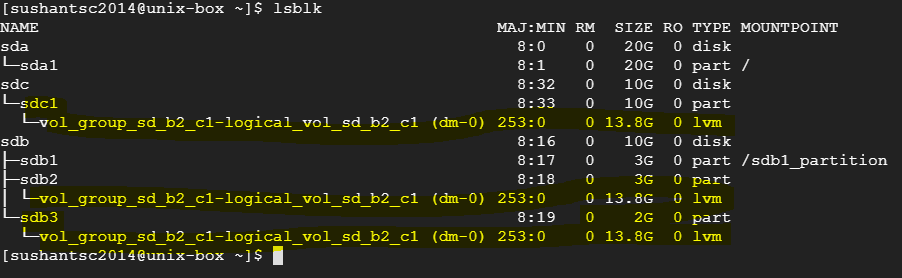


Command: lvresize –r –L +1G <logical\_volume\_name>

-r : file syatem need to resize as well.

-L: provide size in terms of MB,GB etc

+1G : add 1 GB, -1G : reduce by 1 GB



1. Mount the logical volume

