CentOS Linux

System Administrator's Guide

Storage Management

Prepared By: Emad Hussain

Table of Contents

Prepared By: Emad Hussain

1. Standard Partitioning

The standard partitioning scheme typically refers to the use of traditional disk partitioning methods, such as using the MBR (Master Boot Record) or GPT (GUID Partition Table) partitioning schemes. These schemes divide the disk into multiple partitions to organize and manage the storage space.

1.1 Below are the steps to create a new partition from a newly attached disk drive:

1. First, check the available disks on your machine using “lsblk” command.

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sr0                  11:0    1 1024M  0 rom
```

2. Listed below, actual hard-disk “sda” is partition into sda(1) & sda(2), the sda(2) is further divided into different LV(s) using LVM system. This the default LVM created by OS while installation.

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk              <-- Actual Hard Disk (sda)
├─sda1                8:1    0    1G  0 part /boot        <-- (sda) is partitioned into sda(1),sda(2)
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]        <-- sda(2) is further partitioned into LVM(s)
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sr0                  11:0    1 1024M  0 rom
```

3. Let’s add the new hard-drive. I’m going to add 5 GB disk to my machine.
4. After adding new disk, list the available disk on your machine. The results will now show the newly added hard-drive.
5. If the new drive is not visible in available disks, you have to rescan all the “SCSI” hosts to make it visible.

   echo "- - -" | tee /sys/class/scsi_host/host*/scan

```
[root@sysadmin ~]# echo "- - -" | tee /sys/class/scsi_host/host*/scan
- - -
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0    5G  0 disk
sr0                  11:0    1 1024M  0 rom
```

6. Alternatively, you can use the “rescan-scsi-bus.sh” script contained in “sg3_utils” package to scan all the hosts.

   yum install sg3_utils

   rescan-scsi-bus.sh

7. By default, “fdisk” utility build the new drive using “DOS” partitioning scheme. Before proceeding with partitioning the disk, change the partition table of newly attached disk to “GPT”.

```
[root@sysadmin ~]# 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.

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x8dd337b5.

Command (m for help):
```

Press “g” to change the partition.

```
Command (m for help): g
Building a new GPT disklabel (GUID: 3052B452-8972-48B2-A130-EC008B9B2BF4)
```

Press “w” to save the changes.

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

```
[root@sysadmin ~]# blkid /dev/sdb
/dev/sdb: PTTYPE="gpt"
```

8. Create a partition from a newly attached disk, in this case “sdb” using “fdisk” utility.

fdisk /dev/sdb

```
[root@sysadmin ~]# 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.

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x8dd337b5.

Command (m for help):
```

Press ‘n’ to add a new partition to the selected disk.

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

Proceed with default ‘p’ option, now next select partition number (1-4), go ahead with default first sector value. Now, it prompts for the last sector value, the whole drive (5 GB) will be partitioned, if you don’t specify the last sector size. In this case, I’m partitioning with only “2 GB” out of total “5 GB”.

```
Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-10485759, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-10485759, default 10485759): +2G
Partition 1 of type Linux and of size 2 GiB is set
```

Press “p” to print available partitions for the selected drive (sdb).

```
Command (m for help): p

Disk /dev/sdb: 5368 MB, 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
Disk label type: dos
Disk identifier: 0x33762e31

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            2048     4196351     2097152   83  Linux
```

By default, the drive partitioned to a “Linux” filesystem type. However, there is still a need to format a partition to a filesystem type “ext4,xfs”.

Press ‘w’ to save for the changes.

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

9. Verify the the new created partition for “sdb” disk using “lsblk” command.

```
[root@sysadmin ~]# lsblk
NAME               MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                  8:0    0   60G  0 disk
├─sda1               8:1    0    1G  0 part /boot
└─sda2               8:2    0   59G  0 part
  ├─centos_vm-root 253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap 253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home 253:2    0 18.7G  0 lvm  /home
sdb                  8:16   0    5G  0 disk
└─sdb1               8:17   0    2G  0 part
sr0                 11:0    1 1024M  0 rom
```

10. Now format “sdb1” with the appropriate filesystem. I’m formatting with “xfs”.

    mkfs.xfs -L sysadmin_fs /dev/sdb1

```
[root@sysadmin ~]# mkfs.xfs -L sysadmin_fs /dev/sdb1
meta-data=/dev/sdb1              isize=512    agcount=4, agsize=131072 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0, sparse=0
data     =                       bsize=4096   blocks=524288, 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
```

11. Validate the formatted FS using “blkid” command.

    blkid /dev/sdb1

```
[root@sysadmin ~]# blkid /dev/sdb1
/dev/sdb1: LABEL="sysadmin_fs" UUID="72c64ae8-b7bf-4b97-9a8d-6728e6ff3969" TYPE="xfs"
PARTUUID="7cd853af-6b68-4b80-87c3-ca591804756c"
```

12. Let’s mount “sdb1” to a specific mount location.
13. You can mount partition on any location, here I’m going to create a custom directory and mount the partition on that custom mount point.

    mkdir -p /sysadmin_mountpoint

    mount /dev/sdb1 /sysadmin_mountpoint

```
[root@sysadmin ~]# mkdir -p /sysadmin_mountpoint
[root@sysadmin ~]# mount /dev/sdb1 /sysadmin_mountpoint
```

14. Verify the mount using “df -h” command.

```
[root@sysadmin ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
devtmpfs                  475M     0  475M   0% /dev
tmpfs                     487M     0  487M   0% /dev/shm
tmpfs                     487M  7.7M  479M   2% /run
tmpfs                     487M     0  487M   0% /sys/fs/cgroup
/dev/mapper/centos_vm-root   39G  1.6G   37G   5% /
/dev/mapper/centos_vm-home   19G   33M   19G   1% /home
/dev/sda1                1014M  169M  846M  17% /boot
tmpfs                      98M     0   98M   0% /run/user/0
/dev/sdb1                 2.0G   33M  2.0G   2% /sysadmin_mountpoint
```

15. To automatically mount the above the partition or drive on the specific mount location, we have to make configuration changes to the "/etc/fstab" file. The "fstab" file in Linux is used to define and configure the filesystems (partitions) that are automatically mounted during the system boot process.
16. Let's edit the file using "vi/nano" editor and the add following line.

    vi /etc/fstab

    /dev/sdb1 /sysadmin_mountpoint xfs defaults 1 2

```
/dev/mapper/centos_vm-root /                       xfs     defaults        0 0
UUID=02738ca7-6b46-4fb2-b4c3-093352dd8b6f /boot                   xfs     defaults        0 0
/dev/mapper/centos_vm-home /home                   xfs     defaults        0 0
/dev/mapper/centos_vm-swap swap                    swap    defaults        0 0
/dev/sdb1 /sysadmin_mountpoint          xfs     defaults        1       2
```

"/dev/sdb1": This is the device or partition that will be mounted. In this case, it refers to the device file /dev/sdb1.

"/sysadmin_mountpoint": This is the mount point, which is the directory where the filesystem will be attached. In this example, the filesystem will be mounted at the directory /sysadmin_mountpoint.

"xfs": This specifies the filesystem type. In this case, it is XFS, which is a high-performance filesystem commonly used in Linux.

"defaults": This field specifies the mount options for the filesystem. The "defaults" option implies a standard set of mount options, including allowing read and write access, enabling atime (access time) updates, and using default file permissions.

"1": This field determines whether the filesystem should be backed up by the dump utility. A value of 1 indicates that the filesystem will be backed up.

"2": This field specifies the order in which filesystems are checked during the boot process by the fsck (filesystem check) utility. A value of 2 means that the filesystem will be checked after the root filesystem but before other filesystems with a value of 0.

Note: It's a best practice to always specify a filesystem in the "fstab" with their "UUIDs" because "/dev/sdb1" for example can change to "/dev/sdc1" after a system reboot which the "fstab" file will not be able to read the supposed populated filesystem.

1.2 Below are the steps to create another partition from existing drive using “parted” utility:

1. First, check the available disks and partitions on your machine using “lsblk” command.

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0   15G  0 disk
└─sdb1                8:17   0   12G  0 part
sr0                  11:0    1 1024M  0 rom
```

2. Above shown, you can see the drive “sdb” has (15 GB) total space, and a partition of (12 GB) is already created. We’re now going to create another partition of available (3 GB) space.
3. Let’s create another partition using “parted”.

parted /dev/sdb

```
[root@sysadmin ~]# parted /dev/sdb
GNU Parted 3.1
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted)
```

Type ‘print’ to print the available partitions for drive “sdb”.

```
(parted) print
Model: VMware, VMware Virtual S (scsi)
Disk /dev/sdb: 16.1GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name  Flags
 1      1049kB  12.9GB  12.9GB  ext4
```

Proceed for creating a new partition. Type ‘mkpart’ to create a new partition.

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

Proceed with naming the new partition, you can leave it blank as well, next provide file system type, I’m opting for “ext4”. Provide the start & end sector values for creating a partition.

```
(parted) mkpart
Partition name?  []?
File system type?  [ext2]? ext4
Start? 12.9GB
End? 16.1GB
```

Type ‘print’ to make sure that partitioning has been done for drive “sdb”.

```
(parted) print
Model: VMware, VMware Virtual S (scsi)
Disk /dev/sdb: 16.1GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name  Flags
 1      1049kB  12.9GB  12.9GB  ext4
 2      12.9GB  16.1GB  3221MB
```

Type ‘quit’ to exit from parted utility.

4. Confirm using “lsblk” command.

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0   15G  0 disk
├─sdb1                8:17   0   12G  0 part
└─sdb2                8:18   0    3G  0 part
sr0                  11:0    1 1024M  0 rom
```

5. Now format “sdb2” with the appropriate filesystem. I’m formatting with “ext4”.

   mkfs.ext4 -L sysadmin_fs /dev/sdb2

6. Verify using “blkid” command

   blkid /dev/sdb2

```
[root@sysadmin ~]# blkid /dev/sdb2
/dev/sdb2: LABEL="sysadmin_fs2" UUID="f793dd40-0892-4aae-9e7c-bc93484006bb" TYPE="ext4"
PARTUUID="99747e8a-23cc-4453-863c-039cdcb1bd42"
```

7. Let’s mount “sdb2” to a specific mount location.
8. You can mount partition on any location, here I’m going to create a custom directory and mount the partition on that custom mount point.

   mkdir -p /sysadmin_mountpoint2

   mount /dev/sdb1 /sysadmin_mountpoint

9. Verify the mount using “df -h” command.

```
[root@sysadmin ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
devtmpfs                  475M     0  475M   0% /dev
tmpfs                     487M     0  487M   0% /dev/shm
tmpfs                     487M  7.7M  479M   2% /run
tmpfs                     487M     0  487M   0% /sys/fs/cgroup
/dev/mapper/centos_vm-root   39G  2.3G   36G   6% /
/dev/mapper/centos_vm-home   19G   33M   19G   1% /home
/dev/sda1                1014M  169M  846M  17% /boot
tmpfs                      98M     0   98M   0% /run/user/0
/dev/sdb1                  12G   41M   12G   1% /sysadmin_mountpoint
/dev/sdb2                 2.9G  9.0M  2.8G   1% /sysadmin_mountpoint2
```

10. To automatically mount the above the partition or drive on the specific mount location. You can follow the above steps to make entry in “fstab” file.

1.3 Below are the steps to delete partition from existing drive using "parted" utility:

1. First, check the available disks and partitions on your machine using "lsblk" command.

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0   15G  0 disk
├─sdb1                8:17   0   12G  0 part
└─sdb2                8:18   0    3G  0 part
sr0                  11:0    1 1024M  0 rom
```

2. Above shown, you can see the drive "sdb" has two partitions sdb(1),sdb(2). I'm going to delete sdb(2) using parted.
3. Please be careful while deleting the partitions as , this process can make your machine(s) corrupt.
4. Un-mount the partition before deleting it.

   umount /dev/sdb2
5. Let's delete partition sdb(2) using "parted".

   parted /dev/sdb

```
[root@sysadmin ~]# parted /dev/sdb
GNU Parted 3.1
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted)
```

Type 'print' to print the available partitions for drive "sdb".

```
(parted) print
Model: VMware, VMware Virtual S (scsi)
Disk /dev/sdb: 16.1GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name  Flags
 1      1049kB  12.9GB  12.9GB  ext4
 2      12.9GB  16.1GB  3221MB  ext4
```

Proceed for deleting partition sdb(2). Type 'rm <partition-number>' to delete a new partition.

```
(parted) rm 2
(parted)
```

Type 'print' to print the available partitions for drive "sdb".

```
(parted) print
Model: VMware, VMware Virtual S (scsi)
Disk /dev/sdb: 16.1GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name  Flags
 1      1049kB  12.9GB  12.9GB  ext4
```

Type 'quit' to exit from parted utility.

6. Confirm using “lsblk” command.

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0   15G  0 disk
└─sdb1                8:17   0   12G  0 part /sysadmin_mountpoint
sr0                  11:0    1 1024M  0 rom
```

7. You can see above, partition sdb(2) has been removed.

Note: Make sure to take backup before removing any partition. It is recommended to always perform this action using “parted utility” as this can be somewhat safer compared to other tools. Other tools, can make your OS stuck at next boot after deleting a partition.

1.4 Below are the steps to extend an existing partition without LVM using "growpart" utility: (Recommended)

1. First, check the available disks and partitions on your machine using "lsblk" command.

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0   10G  0 disk
└─sdb1                8:17   0    5G  0 part /sysadmin_mountpoint
sr0                  11:0    1 1024M  0 rom
```

2. Now increase the size of your drive from backend, and rescan the SCSI hosts using the steps given above. Here I've already done these steps, we have "sdb" drive of total size (10 GB). I'm going to increase the size of partition "sdb1" from (5 GB) to (10 GB) using growpart utility.
3. First make sure the growpart-utils are installed on your machine.

   yum install cloud-utils-growpart gdisk
4. Let's increase the size of "sdb1" to (10 GB).

   growpart /dev/sdb 1

```
[root@sysadmin ~]# growpart /dev/sdb 1
CHANGED: partition=1 start=2048 old: size=10485760 end=10487808 new: size=20967424 end=20969472
```

5. You need to use the "resize2fs or xfs_growfs" command to resize the filesystem after extending a partition to ensure that the filesystem recognizes and utilizes the additional space. Since in this case, partition "sdb1" FS was "xfs" so I use "xfs_growfs" for resizing. For ext(s) file systems, use "resize2fs"

   xfs_growfs /dev/sdb1

```
[root@sysadmin ~]# xfs_growfs /dev/sdb1
meta-data=/dev/sdb1              isize=512    agcount=4, agsize=327680 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0 spinodes=0
data     =                       bsize=4096   blocks=1310720, 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 1310720 to 2620928
```

6. Verify using "df -h" command.

```
[root@sysadmin ~]# df -h
Filesystem                 Size  Used Avail Use% Mounted on
devtmpfs                   475M     0  475M   0% /dev
tmpfs                      487M     0  487M   0% /dev/shm
tmpfs                      487M  7.7M  479M   2% /run
tmpfs                      487M     0  487M   0% /sys/fs/cgroup
/dev/mapper/centos_vm-root   39G  2.3G   36G   6% /
/dev/mapper/centos_vm-home   19G   33M   19G   1% /home
/dev/sda1                  1014M  169M  846M  17% /boot
tmpfs                       98M     0   98M   0% /run/user/0
/dev/sdb1                   10G  773M  9.3G   8% /sysadmin_mountpoint
```

1.5 Below are the steps to extend an existing partition without LVM using fdisk utility:

1. First, check the available disks and partitions on your machine using “lsblk” command.

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0    5G  0 disk
└─sdb1                8:17   0    2G  0 part /sysadmin_mountpoint
sr0                  11:0    1 1024M  0 rom
```

2. Here we have “sdb” drive of total size (5 GB).

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0    5G  0 disk
└─sdb1                8:17   0    2G  0 part /sysadmin_mountpoint
sr0                  11:0    1 1024M  0 rom
```

3. I’m going to enhance the hard drive size of my machine from backend. To do this, maybe you have to power off your machine.
4. You can see the drive size has been increased from (5 GB) to (10 GB).

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0   10G  0 disk
└─sdb1                8:17   0    2G  0 part /sysadmin_mountpoint
sr0                  11:0    1 1024M  0 rom
```

5. You can also see the consumed space for sdb1.

```
[root@sysadmin sysadmin_mountpoint]# df -h .
Filesystem      Size  Used Avail Use% Mounted on
/dev/sdb1       2.0G  773M  1.3G  38% /sysadmin_mountpoint
```

6. Un-mount the partition

   umount /sysadmin_mountpoint

```
[root@sysadmin /]# umount /sysadmin_mountpoint
[root@sysadmin /]# df -h
Filesystem                 Size  Used Avail Use% Mounted on
devtmpfs                   475M     0  475M   0% /dev
tmpfs                      487M     0  487M   0% /dev/shm
tmpfs                      487M  7.6M  479M   2% /run
tmpfs                      487M     0  487M   0% /sys/fs/cgroup
/dev/mapper/centos_vm-root   39G  1.8G   37G   5% /
/dev/mapper/centos_vm-home   19G   33M   19G   1% /home
/dev/sda1                  1014M  169M  846M  17% /boot
tmpfs                        98M     0   98M   0% /run/user/0
```

7. Make sure to backup the data before extending a standard partition.
8. Extend the partition “sdb1” of disk “sdb” using “fdisk” utility.

   fdisk /dev/sdb

```
[root@sysadmin ~]# 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):
```

Press ‘p’ to print the available partitions for drive “sdb”.

```
Command (m for help): p

Disk /dev/sdb: 10.7 GB, 10737418240 bytes, 20971520 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: 0x00000000

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1               1    10485759     5242879+  ee  GPT

Command (m for help):
```

Press ‘d’ to delete the partition, if multiple partitions available, identify the partition number and then specify it while deleting.

```
Command (m for help): d
Selected partition 1
Partition 1 is deleted

Command (m for help):
```

Proceed for creating a new partition with new allocated space. Press ‘n’ to create a new partition.

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

Proceed with default ‘p’ option, now next select partition number (1-4), go ahead with default first sector value. Now, it prompts for the last sector value, the whole drive (10 GB) will be partitioned, if you don’t specify the last sector size. In this case, I’m partitioning with only “7 GB” out of total “10 GB” available.

```
Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p):
Using default response p
Partition number (1-4, default 1): 1
First sector (2048-20971519, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-20971519, default 20971519): +7G
Partition 1 of type Linux and of size 7 GiB is set
```

Press ‘w’ to save the changes made to disk.

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

9. Now mount the extended partition on the same mount point.

   mount /dev/sdb1 /sysadmin_mountpoint

10. You need to use the “resize2fs or xfs_growfs” command to resize the filesystem after extending a partition to ensure that the filesystem recognizes and utilizes the additional space. Since in this case, partition “sdb1” FS was “xfs” so I use “xfs_growfs” for resizing. For ext(s) file systems, use “resize2fs”

    xfs_growfs /dev/sdb1

```
[root@sysadmin ~]# xfs_growfs /dev/sdb1
meta-data=/dev/sdb1              isize=512    agcount=4, agsize=131072 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0 spinodes=0
data     =                       bsize=4096   blocks=524288, 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 524288 to 1835008
```

11. Verify the extended space using the “df -h” command.

```
[root@sysadmin ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
devtmpfs                  475M     0  475M   0% /dev
tmpfs                     487M     0  487M   0% /dev/shm
tmpfs                     487M  7.7M  479M   2% /run
tmpfs                     487M     0  487M   0% /sys/fs/cgroup
/dev/mapper/centos_vm-root   39G  1.8G   37G   5% /
/dev/mapper/centos_vm-home   19G   33M   19G   1% /home
/dev/sda1                1014M  169M  846M  17% /boot
tmpfs                      98M     0   98M   0% /run/user/0
/dev/sdb1                 7.0G  773M  6.3G  11% /sysadmin_mountpoint
```

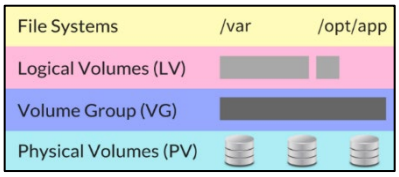
12. Above, it is noticed that the “sdb(1)” space has been extended from (5 GB) to (10 GB).

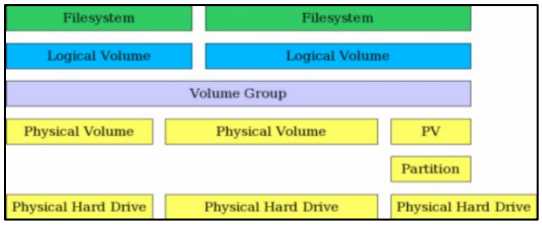
2. Linear LVM Partitioning

LVM stands for Logical Volume Management. This is an alternative method of managing storage systems than the traditional partition-based one. In LVM, instead of creating partitions, you create logical volumes, and then you can just as easily mount those volumes in your filesystem as you'd a disk partition. The main advantage of LVM is how easy it is to resize a logical volume or volume group. It abstracts away all the ugly parts (partitions, raw disks) and leaves us with a central storage pool to work with.

There are three main components to LVM:

1. Physical Volumes
2. Volume Groups
3. Logical Volumes
4. Physical Extents
5. Logical Extents





Physical Volumes:

Physical volumes are the raw materials or building blocks that are used to achieve the abstraction that is logical volumes. In simpler words, physical volumes are the logical unit of an LVM system. Physical volume(s) are created from block devices. A block device could be LUNS created from the storage array, the physical hard drive or even a partition on a disk. Creating and initializing a physical volume are the same thing. Both mean you're just preparing the building blocks (i.e., partitions, disks) for further operations.

Volume Groups:

Volume groups are collections of physical volumes. It is the next level of abstraction in LVM. Volume groups are the storage pool that combines the storage capacity of multiple raw storage devices.

Logical Groups:

A logical volume is like a partition, but instead of sitting on top of a raw disk, it sits on top of a volume group. You can format a logical volume with whichever filesystem you want. Mount it anywhere in the filesystem you want.

Physical Extents:

Physical extent represents a fixed-size block of storage in LVM. It refers to a contiguous portion of a physical disk or a RAID array that is allocated for use by LVM. The size of a physical extent is determined during the creation of a Volume Group (VG) in LVM. All physical extents within a Volume Group have the same size, and this size is typically set to a power of two, such as 4MB or 8MB.

When you create logical volumes (LVs) within a Volume Group, they are divided into smaller units called logical extents, which are mapped to physical extents on the underlying physical storage devices.

Logical Extents:

A logical extent represents a portion of a logical volume (LV) in LVM. It is an abstraction that allows LVM to manage the allocation and mapping of logical volumes to physical extents. The size of a logical extent is the same as the size of a physical extent in the associated Volume Group.

Logical extents are grouped together to form logical volumes, which can be formatted with a file system and mounted as a regular storage device within the operating system. The logical extents of a logical volume do not have to be contiguous on the physical storage; they can be spread across multiple physical devices, allowing for flexibility and management of storage resources.

2.1 Managing Physical Volumes (PV):

1. First, check the available disks on your machine using “lsblk” command.

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root 253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap 253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home 253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0   15G  0 disk
sdc                   8:32   0   20G  0 disk
sr0                  11:0    1 1024M  0 rom
```

2. Here, we have two physical drives “sdb” and “sdc” available right now. For now, we are going to create one PV from “sdb” drive.

   pvcreate /dev/sdb

```
[root@sysadmin ~]# pvcreate /dev/sdb
  Physical volume "/dev/sdb" successfully created.
[root@sysadmin ~]#
```

3. You can verify created PV using “pvs” command.

```
[root@sysadmin ~]# pvs
  PV         VG        Fmt  Attr PSize   PFree
  /dev/sda2  centos_vm lvm2 a--  <59.00g  4.00m
  /dev/sdb             lvm2 ---   15.00g 15.00g
```

4. Now we’ll be partition “sdc” into sdc(1) & sdc(2). You can use above steps to create a partition using fdisk or any other tool.

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root 253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap 253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home 253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0   15G  0 disk
sdc                   8:32   0   20G  0 disk
├─sdc1                8:33   0   10G  0 part
└─sdc2                8:34   0    5G  0 part
sr0                  11:0    1 1024M  0 rom
```

5. Here above we have created sdc(1) of (10 GB) & sdc(2) of (5 GB) out of total (20 GB) space available.
6. Create two PV(s) from these newly created partitions.

   pvcreate /dev/sdc1 /dev/sdc2

```
[root@sysadmin ~]# pvcreate /dev/sdc1 /dev/sdc2
  Physical volume "/dev/sdc1" successfully created.
  Physical volume "/dev/sdc2" successfully created.
```

7. You can verify available physical volumes using “pvs” command.

```
[root@sysadmin ~]# pvs
  PV         VG        Fmt  Attr PSize   PFree
  /dev/sda2  centos_vm lvm2 a--  <59.00g  4.00m
  /dev/sdb             lvm2 ---   15.00g 15.00g
  /dev/sdc1            lvm2 ---   10.00g 10.00g
  /dev/sdc2            lvm2 ---    5.00g  5.00g
```

8. You can also remove a PV using “pvremove” command.

```
[root@sysadmin ~]# pvremove /dev/sdc2
  Labels on physical volume "/dev/sdc2" successfully wiped.
```

9. You can verify available physical volumes using “pvs” command.

```
[root@sysadmin ~]# pvs
  PV         VG        Fmt  Attr PSize   PFree
  /dev/sda2  centos_vm lvm2 a--  <59.00g  4.00m
  /dev/sdb             lvm2 ---   15.00g 15.00g
  /dev/sdc1            lvm2 ---   10.00g 10.00g
```

10. Here, you can see sdc(2) has been removed.

2.2 Managing Volume Groups (VG):

1. Let’s create a volume group from two PV(s) (sdb,sdc1)

   vgcreate <vg-name> <pv-1> <pv-2>

```
[root@sysadmin ~]# vgcreate sysadmin_VG_1 /dev/sdb /dev/sdc1
  Volume group "sysadmin_VG_1" successfully created
```

2. You can list all the volume groups using “vgs” command.

```
[root@sysadmin ~]# vgs
  VG            #PV #LV #SN Attr   VSize   VFree
  centos_vm       1   3   0 wz--n- <59.00g  4.00m
  sysadmin_VG_1   2   0   0 wz--n-  24.99g 24.99g
```

3. You can also extend the existing volume group with a new PV. I’m adding “sdc(2)” to our existing volume group we just created “sysadmin_VG_1”

   vgextend <vg-name> <pv-1> <pv-2>

```
[root@sysadmin ~]# vgextend sysadmin_VG_1 /dev/sdc2
  Volume group "sysadmin_VG_1" successfully extended
```

4. List all the volume groups to verify extended VG, using “vgs” command.

```
[root@sysadmin ~]# vgs
  VG            #PV #LV #SN Attr   VSize   VFree
  centos_vm       1   3   0 wz--n- <59.00g   4.00m
  sysadmin_VG_1   3   0   0 wz--n- <29.99g <29.99g
```

5. Here above, you can see volume group “sysadmin_VG_1” has been extended successfully.
6. You can also reduce a volume group using “vgreduce” command

   vgreduce <vg-name> <pv-1> <pv-2>

```
[root@sysadmin ~]# vgreduce sysadmin_VG_1 /dev/sdc2
  Removed "/dev/sdc2" from volume group "sysadmin_VG_1"
```

7. List all the volume groups to verify reduced VG, using “vgs” command.

```
[root@sysadmin ~]# vgs
  VG            #PV #LV #SN Attr   VSize   VFree
  centos_vm       1   3   0 wz--n- <59.00g  4.00m
  sysadmin_VG_1   2   0   0 wz--n-  24.99g 24.99g
```

8. The whole volume group can be removed using “vgremove” command
   vgremove <vg-name>

2.3 Managing Logical Volumes (LV):

Case I: Creating new LV from Scratch

1. Let’s create a logical volume from VG. I’m creating LV of size 8 GB on our VG “sysadmin_VG_1”
   lvcreate -L <size> -n <lv-name> <vg-name>

```
[root@sysadmin ~]# lvcreate -L 8GB -n sysadmin_LV_1 sysadmin_VG_1
  Logical volume "sysadmin_LV_1" created.
```

2. You can list all the volume groups using “vgs” command.

```
  VG            #PV #LV #SN Attr   VSize   VFree
  centos_vm       1   3   0 wz--n- <59.00g  4.00m
  sysadmin_VG_1   2   1   0 wz--n-  24.99g 16.99g
```

3. Verify the created LV using “lvs” command.

```
[root@sysadmin ~]# lvs
  LV            VG            Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  home          centos_vm     -wi-ao---- <18.70g
  root          centos_vm     -wi-ao---- <38.30g
  swap          centos_vm     -wi-ao----   2.00g
  sysadmin_LV_1 sysadmin_VG_1 -wi-a-----   8.00g
```

4. Format the newly created LVM.
   mkfs.ext4 /dev/<vg-name>/<lv-name>

```
[root@sysadmin ~]# mkfs.ext4 /dev/sysadmin_VG_1/sysadmin_LV_1
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
524288 inodes, 2097152 blocks
104857 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2147483648
64 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

5. You can verify FS type using “blkid” command

```
[root@sysadmin ~]# blkid /dev/sysadmin_VG_1/sysadmin_LV_1
/dev/sysadmin_VG_1/sysadmin_LV_1: UUID="03d3c368-054d-4356-a68a-790eb4d05f24" TYPE="ext4"
```

6. Mount the LV on the any location
   mount <lv-name> <mount-location>
7. Verify using “df -h” command

```
[root@sysadmin ~]# df -h
Filesystem                                Size  Used Avail Use% Mounted on
devtmpfs                                  475M     0  475M   0% /dev
tmpfs                                     487M     0  487M   0% /dev/shm
tmpfs                                     487M  7.7M  479M   2% /run
tmpfs                                     487M     0  487M   0% /sys/fs/cgroup
/dev/mapper/centos_vm-root                 39G  1.6G   37G   5% /
/dev/sda1                                1014M  169M  846M  17% /boot
/dev/mapper/centos_vm-home                 19G   33M   19G   1% /home
tmpfs                                      98M     0   98M   0% /run/user/0
/dev/mapper/sysadmin_VG_1-sysadmin_LV_1  7.8G   36M  7.3G   1% /sysadmin_mountpoint
```

Case II: Extend an existing LV by increasing capacity of an existing partition of a drive (PV):

1. First verify the LV that needs to be extended is part of which volume group using “lvs”.

```
[root@sysadmin ~]# lvs
  LV            VG           Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  home          centos_vm    -wi-ao---- <18.70g
  root          centos_vm    -wi-ao---- <38.30g
  swap          centos_vm    -wi-ao----   2.00g
  sysadmin_LV_1 sysadmin_VG_1 -wi-ao----  24.99g
```

2. I want to increase the space of “sysadmin_LV_1” which is a part of volume group “sysadmin_VG_1”.
3. Let’s check, if “sysadmin_VG_1” volume group has free space available using “vgs” & “pvs” command.

```
[root@sysadmin ~]# vgs
  VG            #PV #LV #SN Attr   VSize   VFree
  centos_vm       1   3   0 wz--n- <59.00g 4.00m
  sysadmin_VG_1   2   1   0 wz--n-  24.99g    0
```

```
[root@sysadmin ~]# pvs
  PV         VG            Fmt  Attr PSize   PFree
  /dev/sda2  centos_vm     lvm2 a--  <59.00g 4.00m
  /dev/sdb   sysadmin_VG_1 lvm2 a--  <15.00g    0
  /dev/sdc1  sysadmin_VG_1 lvm2 a--  <10.00g    0
  /dev/sdc2                lvm2 ---    5.00g 5.00g
```

4. Here above you can see that our VG has no space left. We have to increase the space of any PV (drive/partition) that is a part of our VG. Above “sdb” or “sdc1” both are part of our VG “sysadmin_VG_1”. I’m extending the space of “sdc” from backend.
5. I have increased the space of “sdc” from (20 GB) to (30 GB). Let’s verify using “lsblk” command.

```
[root@sysadmin ~]# lsblk
NAME                            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                               8:0    0   60G  0 disk
├─sda1                            8:1    0    1G  0 part /boot
└─sda2                            8:2    0   59G  0 part
  ├─centos_vm-root              253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap              253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home              253:3    0 18.7G  0 lvm  /home
sdb                               8:16   0   15G  0 disk
└─sysadmin_VG_1-sysadmin_LV_1   253:2    0   25G  0 lvm
sdc                               8:32   0   40G  0 disk
├─sdc1                            8:33   0   10G  0 part
│ └─sysadmin_VG_1-sysadmin_LV_1 253:2    0   25G  0 lvm
└─sdc2                            8:34   0   20G  0 part
sr0                              11:0    1 1024M  0 rom
```

If the changes is not reflect, follow the step 5 of SECTION 1.1, to rescan all the SCSI hosts.

6. Since partition “sdc(1)” is part of Volume Group “sysadmin_VG_1”. So, we have to extend this partition first to make the free space available to VG. I’m using “grow-part” utility for this purpose.

   growpart /dev/sdc 1

```
[root@sysadmin ~]# growpart /dev/sdc 1
NOCHANGE: partition 1 is size 20971520. it cannot be grown
```

7. Above listed error, I’m unable to increase the size sdc(1) even though I’ve increased my drive (sdc) space. This is because, the unallocated space resides at the end of drive. In our case right after partition “sdc(2)”. To increase “sdc1” (PV) we have to delete “sdc2” to make the unallocated space available right after “sdc1” or we can attach a new disk drive and create PV from it, make that PV a part of our VG “sysadmin_VG_1”. This whole picture is better illustrated below:

8. In this case, since I have no any data on my sdc(2) partition. I’m going to delete it or I can make a PV from it also. But, proceeding with deletion using parted.

```
[root@sysadmin ~]# parted /dev/sdc
GNU Parted 3.1
Using /dev/sdc
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) print
Model: VMware, VMware Virtual S (scsi)
Disk /dev/sdc: 42.9GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      1049kB  10.7GB  10.7GB  primary
 2      10.7GB  32.2GB  21.5GB  primary

(parted) rm 2
(parted) print
Model: VMware, VMware Virtual S (scsi)
Disk /dev/sdc: 42.9GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      1049kB  10.7GB  10.7GB  primary
```

9. Verify using “lsblk” command

```
[root@sysadmin ~]# lsblk
NAME                          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                             8:0    0   60G  0 disk
├─sda1                          8:1    0    1G  0 part /boot
└─sda2                          8:2    0   59G  0 part
  ├─centos_vm-root            253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap            253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home            253:3    0 18.7G  0 lvm  /home
sdb                             8:16   0   15G  0 disk
└─sysadmin_VG_1-sysadmin_LV_1 253:2    0   25G  0 lvm
sdc                             8:32   0   40G  0 disk
└─sdc1                          8:33   0   10G  0 part
  └─sysadmin_VG_1-sysadmin_LV_1 253:2    0   25G  0 lvm
sr0                            11:0    1 1024M  0 rom
```

10. Now extend “sdc1” using growpart utility.

```
[root@sysadmin ~]# growpart /dev/sdc 1
CHANGED: partition=1 start=2048 old: size=20971520 end=20973568 new: size=83883999 end=83886047
```

11. List all the disks and partitions again.

```
[root@sysadmin ~]# lsblk
NAME                            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                               8:0    0   60G  0 disk
├─sda1                            8:1    0    1G  0 part /boot
└─sda2                            8:2    0   59G  0 part
  ├─centos_vm-root              253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap              253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home              253:3    0 18.7G  0 lvm  /home
sdb                               8:16   0   15G  0 disk
└─sysadmin_VG_1-sysadmin_LV_1   253:2    0   25G  0 lvm
sdc                               8:32   0   40G  0 disk
└─sdc1                            8:33   0   40G  0 part
  └─sysadmin_VG_1-sysadmin_LV_1 253:2    0   25G  0 lvm
sr0                              11:0    1 1024M  0 rom
```

12. Now, we have to resize our PV (sdc1) to make the free space available.

    pvresize /dev/sdc1

```
[root@sysadmin ~]# pvresize /dev/sdc1
  Physical volume "/dev/sdc1" changed
  1 physical volume(s) resized or updated / 0 physical volume(s) not resized
```

13. Confirm the new free space using “pvs” command.

```
[root@sysadmin ~]# pvs
  PV         VG            Fmt  Attr PSize   PFree
  /dev/sda2  centos_vm     lvm2 a--  <59.00g  4.00m
  /dev/sdb   sysadmin_VG_1 lvm2 a--  <15.00g     0
  /dev/sdc1  sysadmin_VG_1 lvm2 a--  <40.00g 30.00g
```

14. Here, it is confirmed that free space has been available.
15. Let’s confirm the same available to VG.

```
[root@sysadmin ~]# vgs
  VG            #PV #LV #SN Attr   VSize   VFree
  centos_vm       1   3   0 wz--n- <59.00g  4.00m
  sysadmin_VG_1   2   1   0 wz--n-  54.99g 30.00g
```

16. Proceed for extending the size of LV of “sysadmin_VG_1”

    lvresize -L <+|-><size> <vg-name>/<lv-name>

17. Let’s first confirm our current LV size.

```
NAME                            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                               8:0    0   60G  0 disk
├─sda1                            8:1    0    1G  0 part /boot
└─sda2                            8:2    0   59G  0 part
  ├─centos_vm-root              253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap              253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home              253:3    0 18.7G  0 lvm  /home
sdb                               8:16   0   15G  0 disk
└─sysadmin_VG_1-sysadmin_LV_1   253:2    0   25G  0 lvm
sdc                               8:32   0   40G  0 disk
└─sdc1                            8:33   0   40G  0 part
  └─sysadmin_VG_1-sysadmin_LV_1 253:2    0   25G  0 lvm
sr0                              11:0    1 1024M  0 rom
```

18. Current LV size is 25 GB. Let’s extend it to 55 GB.

```
[root@sysadmin ~]# lvresize -L +30GB sysadmin_VG_1/sysadmin_LV_1
  Size of logical volume sysadmin_VG_1/sysadmin_LV_1 changed from 24.99 GiB (6398 extents) to 54.99 GiB (1407
8 extents).
  Logical volume sysadmin_VG_1/sysadmin_LV_1 successfully resized.
```

19. You need to use the resize the filesystem after extending a LVM to ensure that the filesystem recognizes and utilizes the additional space. For LVM, “resize2fs” will be used.

    resize2fs /dev/<vg-name>/<lv-name>

```
[root@sysadmin ~]# resize2fs /dev/sysadmin_VG_1/sysadmin_LV_1
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/sysadmin_VG_1/sysadmin_LV_1 is mounted on /sysadmin_mountpoint; on-line resizing required
old_desc_blocks = 4, new_desc_blocks = 7
The filesystem on /dev/sysadmin_VG_1/sysadmin_LV_1 is now 14415872 blocks long.
```

20. Verify using “df -h”

```
[root@sysadmin ~]# df -h
Filesystem                                Size  Used Avail Use% Mounted on
devtmpfs                                  475M     0  475M   0% /dev
tmpfs                                     487M     0  487M   0% /dev/shm
tmpfs                                     487M  7.7M  479M   2% /run
tmpfs                                     487M     0  487M   0% /sys/fs/cgroup
/dev/mapper/centos_vm-root                 39G  1.6G   37G   5% /
/dev/mapper/centos_vm-home                 19G   33M   19G   1% /home
/dev/sda1                                1014M  169M  846M  17% /boot
tmpfs                                      98M     0   98M   0% /run/user/0
/dev/mapper/sysadmin_VG_1-sysadmin_LV_1   55G   52M   52G   1% /sysadmin_mountpoint
```

Case III: Extend an existing LV by adding another drive to a machine:

1. First verify the LV that needs to be extended is part of which volume group using “lvs”.

```
[root@sysadmin ~]# lvs
  LV            VG            Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  home          centos_vm     -wi-ao---- <18.70g
  root          centos_vm     -wi-ao---- <38.30g
  swap          centos_vm     -wi-ao----   2.00g
  sysadmin_LV_1 sysadmin_VG_1 -wi-ao----  54.99g
```

2. I want to increase the space of “sysadmin_LV_1” which is a part of volume group “sysadmin_VG_1”.
3. Let’s check, if “sysadmin_VG_1” volume group has free space available using “vgs” & “pvs” command.

```
[root@sysadmin ~]# vgs
  VG             #PV #LV #SN Attr   VSize   VFree
  centos_vm        1   3   0 wz--n- <59.00g 4.00m
  sysadmin_VG_1    2   1   0 wz--n-  54.99g    0
[root@sysadmin ~]# pvs
  PV         VG            Fmt  Attr PSize   PFree
  /dev/sda2  centos_vm     lvm2 a--  <59.00g 4.00m
  /dev/sdb   sysadmin_VG_1 lvm2 a--  <15.00g    0
  /dev/sdc1  sysadmin_VG_1 lvm2 a--  <40.00g    0
```

4. Here above you can see that our VG has no space left. We have to attached another hard drive to our machine.
5. Below you can see that I have attached disk “sdd” of space (5 GB).

```
[root@sysadmin ~]# lsblk
NAME                          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                             8:0    0   60G  0 disk
├─sda1                          8:1    0    1G  0 part /boot
└─sda2                          8:2    0   59G  0 part
  ├─centos_vm-root            253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap            253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home            253:3    0 18.7G  0 lvm  /home
sdb                             8:16   0   15G  0 disk
└─sysadmin_VG_1-sysadmin_LV_1 253:2    0   55G  0 lvm  /sysadmin_mountpoint
sdc                             8:32   0   40G  0 disk
└─sdc1                          8:33   0   40G  0 part
  └─sysadmin_VG_1-sysadmin_LV_1 253:2    0   55G  0 lvm  /sysadmin_mountpoint
sdd                             8:48   0    5G  0 disk
sr0                            11:0    1 1024M  0 rom
```

   If the changes is not reflect, follow the step 5 of SECTION 1.1, to rescan all the SCSI hosts.
6. Let’s create a PV from this newly attached drive.

   pvcreate /dev/sdd

```
[root@sysadmin ~]# pvcreate /dev/sdd
  Physical volume "/dev/sdd" successfully created.
[root@sysadmin ~]#
```

7. List all the PV(s) to confirm.

```
[root@sysadmin ~]# pvs
  PV         VG            Fmt  Attr PSize   PFree
  /dev/sda2  centos_vm     lvm2 a--  <59.00g 4.00m
  /dev/sdb   sysadmin_VG_1 lvm2 a--  <15.00g    0
  /dev/sdc1  sysadmin_VG_1 lvm2 a--  <40.00g    0
  /dev/sdd                 lvm2 ---    5.00g 5.00g
```

8. Now extend the VG “sysadmin_VG_1” using the newly created PV (sdd)

   vgextend sysadmin_VG_1 /dev/sdd

9. Confirm by listing the available VG(s) and check the free space.

```
[root@sysadmin ~]# vgextend sysadmin_VG_1 /dev/sdd
  Volume group "sysadmin_VG_1" successfully extended
[root@sysadmin ~]# vgs
  VG            #PV #LV #SN Attr   VSize   VFree
  centos_vm       1   3   0 wz--n- <59.00g  4.00m
  sysadmin_VG_1   3   1   0 wz--n- <59.99g <5.00g
```

21. Proceed for extending the size of LV of “sysadmin_VG_1”

    lvresize -L <+|-><size> <vg-name>/<lv-name>

22. Let’s first confirm our current LV size.

```
[root@sysadmin ~]# lsblk
NAME                            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                               8:0    0   60G  0 disk
├─sda1                            8:1    0    1G  0 part /boot
└─sda2                            8:2    0   59G  0 part
  ├─centos_vm-root              253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap              253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home              253:3    0 18.7G  0 lvm  /home
sdb                               8:16   0   15G  0 disk
└─sysadmin_VG_1-sysadmin_LV_1   253:2    0   55G  0 lvm  /sysadmin_mountpoint
sdc                               8:32   0   40G  0 disk
└─sdc1                            8:33   0   40G  0 part
  └─sysadmin_VG_1-sysadmin_LV_1 253:2    0   55G  0 lvm  /sysadmin_mountpoint
sdd                               8:48   0    5G  0 disk
sr0                              11:0    1 1024M  0 rom
```

23. Current LV size is 55 GB. Let’s extend it to 60 GB.

```
[root@sysadmin ~]# lvresize -L +4.9GB sysadmin_VG_1/sysadmin_LV_1
  Rounding size to boundary between physical extents: 4.90 GiB.
  Size of logical volume sysadmin_VG_1/sysadmin_LV_1 changed from 54.99 GiB (14078 extents) to 59.89 GiB (153
33 extents).
  Logical volume sysadmin_VG_1/sysadmin_LV_1 successfully resized.
```

24. You need to use the resize the filesystem after extending a LVM to ensure that the filesystem recognizes and utilizes the additional space. For xfs LVM, “xfs_growfs” will be used. For ext4 LVM, “resize2fs” will be used.

    resize2fs /dev/<vg-name>/<lv-name>

```
[root@sysadmin ~]# resize2fs /dev/sysadmin_VG_1/sysadmin_LV_1
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/sysadmin_VG_1/sysadmin_LV_1 is mounted on /sysadmin_mountpoint; on-line resizing required
old_desc_blocks = 7, new_desc_blocks = 8
The filesystem on /dev/sysadmin_VG_1/sysadmin_LV_1 is now 15700992 blocks long.
```

25. Verify using “df -h”

```
[root@sysadmin ~]# df -h
Filesystem                              Size  Used Avail Use% Mounted on
devtmpfs                                475M     0  475M   0% /dev
tmpfs                                   487M     0  487M   0% /dev/shm
tmpfs                                   487M  7.7M  479M   2% /run
tmpfs                                   487M     0  487M   0% /sys/fs/cgroup
/dev/mapper/centos_vm-root               39G  1.6G   37G   5% /
/dev/mapper/centos_vm-home               19G   33M   19G   1% /home
/dev/sda1                              1014M  169M  846M  17% /boot
tmpfs                                    98M     0   98M   0% /run/user/0
/dev/mapper/sysadmin_VG_1-sysadmin_LV_1   59G   52M   57G   1% /sysadmin_mountpoint
```

Case IV: Extend an existing LV by reducing another LV of the same VG:

Reduction:

1. First verify the LV that needs to be reduced is part of which volume group using “lvs”.

```
[root@sysadmin ~]# lvs
  LV            VG            Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  home          centos_vm     -wi-ao---- <18.70g
  root          centos_vm     -wi-ao---- <38.30g
  swap          centos_vm     -wi-ao----   2.00g
  sysadmin_LV_1 sysadmin_VG_1 -wi-a-----  49.89g
  sysadmin_LV_2 sysadmin_VG_1 -wi-a-----  10.00g
```

2. I want to decrease the space of “sysadmin_LV_2” which is a part of volume group “sysadmin_VG_1”.
3. Let’s check, if “sysadmin_VG_1” volume group has free space available using “vgs” command.

```
[root@sysadmin ~]# vgs
  VG            #PV #LV #SN Attr   VSize   VFree
  centos_vm       1   3   0 wz--n- <59.00g  4.00m
  sysadmin_VG_1   3   2   0 wz--n- <59.99g 96.00m
```

4. Here above you can see that our VG has almost no space left. What we are going to do is to decrease the space the “sysadmin_LV_2” and increase space of “sysadmin_LV_1” of the same VG.
5. First verify the all the drives and partitions.

```
[root@sysadmin ~]# lsblk
NAME                          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                             8:0    0   60G  0 disk
├─sda1                          8:1    0    1G  0 part /boot
└─sda2                          8:2    0   59G  0 part
  ├─centos_vm-root            253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap            253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home            253:4    0 18.7G  0 lvm  /home
sdb                             8:16   0   15G  0 disk
└─sysadmin_VG_1-sysadmin_LV_1 253:2    0 49.9G  0 lvm
sdc                             8:32   0   40G  0 disk
└─sdc1                          8:33   0   40G  0 part
  ├─sysadmin_VG_1-sysadmin_LV_1 253:2    0 49.9G  0 lvm
  └─sysadmin_VG_1-sysadmin_LV_2 253:3    0   10G  0 lvm
sdd                             8:48   0    5G  0 disk
└─sysadmin_VG_1-sysadmin_LV_2 253:3    0   10G  0 lvm
sr0                            11:0    1 1024M  0 rom
```

6. Here current size of “sysadmin_LV_2” is (10 GB). I’m reducing it to (5 GB). If the LVM is mounted make sure to unmount it first before actual reduction. In my case it is already unmounted.

```
[root@sysadmin ~]# lvs
  LV            VG            Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  home          centos_vm     -wi-ao---- <18.70g
  root          centos_vm     -wi-ao---- <38.30g
  swap          centos_vm     -wi-ao----   2.00g
  sysadmin_LV_1 sysadmin_VG_1 -wi-a-----  49.89g
  sysadmin_LV_2 sysadmin_VG_1 -wi-a-----  10.00g
```

7. Let’s check the FS for any errors. Depending on the filesystem type, you can use commands like “fsck.ext4” or “xfs_repair”

   fsck.ext4 /dev/<vg-name>/<lv-name>

```
[root@sysadmin ~]# fsck.ext4 /dev/sysadmin_VG_1/sysadmin_LV_2
e2fsck 1.42.9 (28-Dec-2013)
/dev/sysadmin_VG_1/sysadmin_LV_2: clean, 11/655360 files, 83137/2621440 blocks
```

8. You need to use the resize the filesystem after extending a LVM to ensure that the filesystem recognizes and utilizes the additional space. For xfs LVM, “xfs_growfs” will be used. For ext4 LVM, “resize2fs” will be used.

   resize2fs /dev/<vg-name>/<lv-name> <size in MB(s)>

```
[root@sysadmin ~]# resize2fs /dev/sysadmin_VG_1/sysadmin_LV_2 5012M
resize2fs 1.42.9 (28-Dec-2013)
Resizing the filesystem on /dev/sysadmin_VG_1/sysadmin_LV_2 to 1283072 (4k) blocks.
The filesystem on /dev/sysadmin_VG_1/sysadmin_LV_2 is now 1283072 blocks long.
```

9. Let's decrease the size of LV.

   lvresize -L <-><size> <vg-name>/<lv-name>

```
[root@sysadmin ~]# lvresize -L -5GB sysadmin_VG_1/sysadmin_LV_2
  WARNING: Reducing active logical volume to 5.00 GiB.
  THIS MAY DESTROY YOUR DATA (filesystem etc.)
Do you really want to reduce sysadmin_VG_1/sysadmin_LV_2? [y/n]: y
  Size of logical volume sysadmin_VG_1/sysadmin_LV_2 changed from 10.00 GiB (2560 extents) to 5.00 GiB (1280
extents).
  Logical volume sysadmin_VG_1/sysadmin_LV_2 successfully resized.
```

10. Let's check, if "sysadmin_VG_1" volume group has free space available using "vgs" command.

```
[root@sysadmin ~]# vgs
  VG             #PV #LV #SN Attr   VSize   VFree
  centos_vm        1   3   0 wz--n- <59.00g 4.00m
  sysadmin_VG_1    3   2   0 wz--n- <59.99g 5.09g
```

11. Here above you can see that our VG has now 5 GB free space available. Let's allocate it to the "sysadmin_LV_1" of the same VG.

    lvresize -L <+><size> <vg-name>/<lv-name>

```
[root@sysadmin ~]# lvresize -L +5.09GB sysadmin_VG_1/sysadmin_LV_1
  Rounding size to boundary between physical extents: 5.09 GiB.
  Size of logical volume sysadmin_VG_1/sysadmin_LV_1 changed from 49.89 GiB (12773 extents) to <54.99 GiB (14
077 extents).
  Logical volume sysadmin_VG_1/sysadmin_LV_1 successfully resized.
```

12. You need to use the resize the filesystem after extending a LVM to ensure that the filesystem recognizes and utilizes the additional space. For xfs LVM, "xfs_growfs" will be used. For ext4 LVM, "resize2fs" will be used. Perform diskcheck if needed.

    resize2fs /dev/<vg-name>/<lv-name>

```
[root@sysadmin ~]# e2fsck -f /dev/sysadmin_VG_1/sysadmin_LV_1
e2fsck 1.42.9 (28-Dec-2013)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/sysadmin_VG_1/sysadmin_LV_1: 11/3270400 files (0.0% non-contiguous), 251390/13079552 blocks
[root@sysadmin ~]#
[root@sysadmin ~]# resize2fs /dev/sysadmin_VG_1/sysadmin_LV_1
resize2fs 1.42.9 (28-Dec-2013)
Resizing the filesystem on /dev/sysadmin_VG_1/sysadmin_LV_1 to 14414848 (4k) blocks.
The filesystem on /dev/sysadmin_VG_1/sysadmin_LV_1 is now 14414848 blocks long.
```

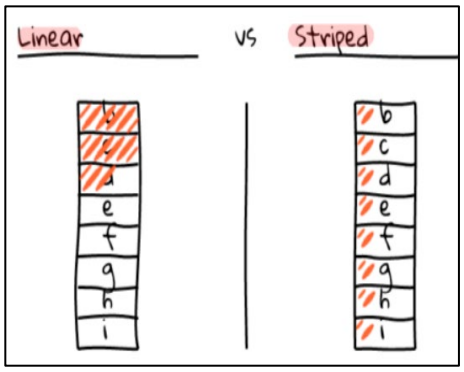
13. Remount all the LVM(s)
14. Verify using "df -h"

```
[root@sysadmin ~]# df -h
Filesystem                                Size  Used Avail Use% Mounted on
devtmpfs                                  475M     0  475M   0% /dev
tmpfs                                     487M     0  487M   0% /dev/shm
tmpfs                                     487M  7.7M  479M   2% /run
tmpfs                                     487M     0  487M   0% /sys/fs/cgroup
/dev/mapper/centos_vm-root                 39G  1.6G   37G   5% /
/dev/sda1                                1014M  169M  846M  17% /boot
/dev/mapper/centos_vm-home                 19G   33M   19G   1% /home
tmpfs                                      98M     0   98M   0% /run/user/0
/dev/mapper/sysadmin_VG_1-sysadmin_LV_1    55G   53M   52G   1% /sysadmin_mountpoint
/dev/mapper/sysadmin_VG_1-sysadmin_LV_2   4.8G   33M  4.5G   1% /sysadmin_mountpoint1
```

15. Above you can see that LVM_1 space has been increased from 50GB to 55GB and LVM_2 space has been decreased from 10GB to 5 GB

Note: Working with LVM reductions can result in data-loss or LVM corruption. It is always recommended to backups all the important data when reducing the LVM(s).

3. Stripped LVM Partitioning



A linear logical volume consists of a single physical volume or a concatenation of multiple physical volumes. It's the simplest form of logical volume in LVM, where the data is stored sequentially across the underlying physical volumes.

A striped logical volume, also known as a RAID-0 logical volume, distributes data across multiple physical volumes in a striped manner. The data is divided into chunks or stripes, which are then distributed evenly across the physical volumes. This provides improved performance by allowing simultaneous read and write operations across multiple disks.

The stripe size refers to the amount of data that is written to a single physical volume (PV) before moving on to the next one. It determines the size of each individual stripe within the striped LV.

3.1 Steps to create stripped LV from VG:

1. First, check the available disks on your machine using "lsblk" command.

```
[root@sysadmin ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                   8:0    0   60G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   59G  0 part
  ├─centos_vm-root  253:0    0 38.3G  0 lvm  /
  ├─centos_vm-swap  253:1    0    2G  0 lvm  [SWAP]
  └─centos_vm-home  253:2    0 18.7G  0 lvm  /home
sdb                   8:16   0   10G  0 disk
sdc                   8:32   0   10G  0 disk
sdd                   8:48   0   20G  0 disk
sr0                  11:0    1 1024M  0 rom
```

2. Here above, we have three disks "**sdb,sdc,sdd**" are available for use.
3. Let create physical volume from these drives and verify using "pvs" command.

pvcreate /dev/sd[b-d]

```
[root@sysadmin ~]# pvcreate /dev/sd[b-d]
  Physical volume "/dev/sdb" successfully created.
  Physical volume "/dev/sdc" successfully created.
  Physical volume "/dev/sdd" successfully created.
```

```
[root@sysadmin ~]# pvs
  PV         VG        Fmt  Attr PSize   PFree
  /dev/sda2  centos_vm lvm2 a--  <59.00g  4.00m
  /dev/sdb             lvm2 ---   10.00g 10.00g
  /dev/sdc             lvm2 ---   10.00g 10.00g
  /dev/sdd             lvm2 ---   20.00g 20.00g
```

4. Create volume group from these PV(s) and verify newly created VG using "vgs" command.

vgcreate sysadmin_VG_1 /dev/sd[b-d]

```
[root@sysadmin ~]# vgcreate sysadmin_VG_1 /dev/sd[b-d]
  Volume group "sysadmin_VG_1" successfully created
[root@sysadmin ~]# vgs
  VG            #PV #LV #SN Attr   VSize   VFree
  centos_vm       1   3   0 wz--n- <59.00g   4.00m
  sysadmin_VG_1   3   0   0 wz--n- <39.99g <39.99g
```

5. Let's create stripped LV from newly created VG "sysadmin_VG_1". The below command creates a LV of 27G from VG total space of 40 GB, with used stripes(drives) = 3, stipe-size = 1MB each.

lvcreate --stripes 3 --stripesize 1M -L 27G -n sysadmin_SLV_1 sysadmin_VG_1

```
[root@sysadmin ~]# lvcreate --stripes 3 --stripesize 1M -L 27G -n sysadmin_SLV_1 sysadmin_VG_1
  Logical volume "sysadmin_SLV_1" created.
[root@sysadmin ~]# lvs
  LV            VG           Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  home          centos_vm    -wi-ao---- <18.70g
  root          centos_vm    -wi-ao---- <38.30g
  swap          centos_vm    -wi-ao----   2.00g
  sysadmin_SLV_1 sysadmin_VG_1 -wi-a-----  27.00g
```

6. You can check the devices and extents used by stripped LV using "lvdisplay -v <lv-name>".

```
[root@sysadmin ~]# lvdisplay -m /dev/sysadmin_VG_1/sysadmin_SLV_1
  --- Logical volume ---
  LV Path                /dev/sysadmin_VG_1/sysadmin_SLV_1
  LV Name                sysadmin_SLV_1
  VG Name                sysadmin_VG_1
  LV UUID                N1Li08-7Haf-29Ue-lzw6-fmGE-Yns7-vZ4w1n
  LV Write Access        read/write
  LV Creation host, time sysadmin, 2023-06-03 04:40:35 +0500
  LV Status              available
  # open                 0
  LV Size                27.00 GiB
  Current LE             6912
  Segments               1
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     12288
  Block device           253:3

  --- Segments ---
  Logical extents 0 to 6911:
    Type                striped
    Stripes             3
    Stripe size         1.00 MiB
    Stripe 0:
      Physical volume   /dev/sdb
      Physical extents  0 to 2303
    Stripe 1:
      Physical volume   /dev/sdc
      Physical extents  0 to 2303
    Stripe 2:
      Physical volume   /dev/sdd
      Physical extents  0 to 2303
```

7. Let's format the stripped LV with ext4 FS.

mkfs.ext4 /dev/<vg-name>/<lv-name>

```
[root@sysadmin ~]# mkfs.ext4 /dev/sysadmin_VG_1/sysadmin_SLV_1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=256 blocks, Stripe width=768 blocks
1769472 inodes, 7077888 blocks
353894 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2155872256
216 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

8. You can verify using blkid command.

```
[root@sysadmin ~]# blkid /dev/sysadmin_VG_1/sysadmin_SLV_1
/dev/sysadmin_VG_1/sysadmin_SLV_1: UUID="4531ae5e-d3cb-49b6-8a8a-b1345f6003c4" TYPE="ext4"
[root@sysadmin ~]#
```

9. We can also create linear LV along with stripped LV on same VG.

lvcreate -L 12.988GB -n sysadmin_LLV_1 sysadmin_VG_1

```
[root@sysadmin ~]# lvcreate -L 12.988GB -n sysadmin_LLV_1 sysadmin_VG_1
  Rounding up size to full physical extent <12.99 GiB
  Logical volume "sysadmin_LLV_1" created.
```

10. Let's mount the stripped LV created above to a mount location.

mount -t ext4 /dev/<vg-name>/<lv-name>

```
Filesystem                                Size  Used Avail Use% Mounted on
devtmpfs                                  475M     0  475M   0% /dev
tmpfs                                     487M     0  487M   0% /dev/shm
tmpfs                                     487M  7.7M  479M   2% /run
tmpfs                                     487M     0  487M   0% /sys/fs/cgroup
/dev/mapper/centos_vm-root                 39G  1.6G   37G   5% /
/dev/sda1                                1014M  169M  846M  17% /boot
/dev/mapper/centos_vm-home                 19G   33M   19G   1% /home
tmpfs                                      98M     0   98M   0% /run/user/0
/dev/mapper/sysadmin_VG_1-sysadmin_SLV_1   27G   45M   26G   1% /stripped_mount
```

3.2 Steps to extend a stripped LV.

To increase the size of striped logical volume, there must be enough free space on the underlying physical volumes that make up the volume group to support the stripe. For example, if you have a two-way stripe that that uses up an entire volume group, adding a single physical volume to the volume group will not enable you to extend the stripe. Instead, you must add at least two physical volumes to the volume group.

1. First, verify VG free space and list all the LVM using “lvs” command.

```
[root@sysadmin /]# vgs
  VG            #PV #LV #SN Attr   VSize   VFree
  centos_vm       1   3   0 wz--n- <59.00g   4.00m
  sysadmin_VG_1   3   1   0 wz--n- <39.99g <12.99g
[root@sysadmin /]# lvs
  LV            VG            Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  home          centos_vm     -wi-ao---- <18.70g
  root          centos_vm     -wi-ao---- <38.30g
  swap          centos_vm     -wi-ao----   2.00g
  sysadmin_SLV_1 sysadmin_VG_1 -wi-ao----  27.00g
```

2. Let’s try extending the stripped LV.

   lvextend /dev/<vg-name>/<lv-name> -L <size>

```
[root@sysadmin ~]# lvextend /dev/sysadmin_VG_1/sysadmin_SLV_1 -L +3GB
  Using stripesize of last segment 1.00 MiB
  Insufficient suitable allocatable extents for logical volume sysadmin_SLV_1: 3 more required
```

3. You can see above that, we can’t able to extend. Based on required extents we need at-least two new physical disks.
4. Let’s extend our VG by attaching another 2 PV(s) to our volume group. I have already attached a new physical disk (sde,sdf) of 30GB,5GB respectively to our machine.

   pvcreate /dev/sd[e-g]

   vgextend <vg-name> /dev/sd[e-g]

```
[root@sysadmin ~]# pvcreate /dev/sd[e-f]
  Physical volume "/dev/sde" successfully created.
  Physical volume "/dev/sdf" successfully created.
[root@sysadmin ~]# vgextend sysadmin_VG_1 /dev/sd[e-f]
  Volume group "sysadmin_VG_1" successfully extended
```

5. Let’ check the LV segments first.

   lvdisplay -m /dev/sysadmin_VG_1/sysadmin_SLV_1 | tail -17

```
[root@sysadmin ~]# lvdisplay -m /dev/sysadmin_VG_1/sysadmin_SLV_1 | tail -17

  --- Segments ---
  Logical extents 0 to 6911:
    Type                striped
    Stripes             3
    Stripe size         1.00 MiB
    Stripe 0:
      Physical volume   /dev/sdb
      Physical extents  0 to 2303
    Stripe 1:
      Physical volume   /dev/sdc
      Physical extents  0 to 2303
    Stripe 2:
      Physical volume   /dev/sdd
      Physical extents  0 to 2303
```

6. Now, try extending the stripped LV again.

   lvextend /dev/<vg-name>/<lv-name> -L +<size>

```
[root@sysadmin ~]# lvextend /dev/sysadmin_VG_1/sysadmin_SLV_1 -L +10G
  Using stripesize of last segment 1.00 MiB
  Rounding size (9472 extents) up to stripe boundary size for segment (9474 extents).
  Size of logical volume sysadmin_VG_1/sysadmin_SLV_1 changed from 27.00 GiB (6912 extents) to <37.01 GiB (9474 extents)
.
  Logical volume sysadmin_VG_1/sysadmin_SLV_1 successfully resized.
```

7. Now LV has been successfully extended.

Note: You can also reduce stripped LV size, but since data resides on different PV(s), so it's not wise to reduce this type of logical volumes.