Contents

[**LVM-PV, VG, LV** 2](#_Toc481192069)

[**LVM (Logical Volume Manager):** 2](#_Toc481192070)

[**Advantages:** 2](#_Toc481192071)

[**Physical Volume (PV):** 3](#_Toc481192072)

[**Volume Group (VG):** 3](#_Toc481192073)

[**Logical Volume (LV):** 4](#_Toc481192074)

# **LVM-PV, VG, LV**

## **LVM (Logical Volume Manager):**

* LVM is a tool that provides logical volume management for Linux kernel.
* It allows to manage disk storage space in effective/optimized manner.
* LVM is located within OS.
* Basically, it divides PVs (Physical Volumes) into PEs (Physical Extents). These PEs are mapped onto LEs (Logical Extents) and combined into VGs (Volume Groups). VGs are linked together into LVs (Logical Volumes) that act as virtual partition and that is managed by LVM.

### **Advantages:**

* A set of hard disk/drive is allocated to one or more physical volumes.
* It provides dynamic space allocation. Space can be allocated at any time instead of allocation at setup time.
* PV, VG, LV can be extended, removed or added at any time.

Logical Volume 2 GB

/home

Free 1 GB

Logical Volume 2 GB

/usr

Physical Hard Disk 2.5 GB

Physical Hard Disk 2.5 GB

Physical Hard Disk 5 GB

Physical Volume/Physical Partition

sdb1

Physical Volume/Physical Partition

sdb2

Volume Group 5 GB

Name of VG: vg\_vaibhav

## **Physical Volume (PV):**

* PV can be considered as Physical partition of hard disk/drive of a computer.
* LVM divides PV into chunk of data known as PEs. PE can be considered as smallest unit of data storage. Default size of PE is 4 MB.
* Hard disk notation: sda, sdb, sdc,etc. [sd: Standard disk]
* Physical Partition notation on hard disk: sda1, sda2,sda3,etc. [sda1:1st partition of hard disk sda, sda2: 2nd partition of hard disk sda]
* It is advisable to create physical partition before creating PV.
* To create PV,

Command: pvcreate <physical\_partition\_name>

E.g. pvcreate /dev/sdb1

E.g. pvcreate /dev/sdb1 /dev/sdb2

* To display PVs,

Command: pvs OR pvdisplay

* To remove PV,

Command: pvremove <physical\_partition\_name>

E.g. pvremove /dev/sdb1

## **Volume Group (VG):**

* In simple terms, combination of PVs = VG.
* PV must be created before creating VG.
* VG keeps PVs in a separate manner. So things are divided and get easily accessible. So multiple VGs are possible.
* PE and LE are of same size within a VG.
* To create VG,

Command: vgcreate <VG\_name> <PV>

E.g. vgcreate vg1 /dev/sdb

E.g. vgcreate vg2 /dev/sda1 /dev/sda2

* To display VG,

Command: vgs OR vgdisplay

* To extend VG,

Command: vgextend <VG\_name> <PV\_name to be extended>

E.g. vgextend vg1 /dev/sdb3

* To reduce particular OV from VG,

Command: vgreduce <VG\_name> <PV\_name to be reduced/deleted>

E.g. vgreduce vg1 /dev/sdb3

* To remove VG,

Command: vgremove <VG\_name>

E.g. vgremove vg\_vaibhav

## **Logical Volume (LV):**

* LV is just a logical partition that is created as per user’s requirement.
* LV can be spread across multiple PVs and VGs.
* LVM divides LV into chunk of data known as LEs. LE can be considered as smallest unit of data storage. Default size of LE is 4 MB.
* There are three kind of LVs:

1. Linear LV (Default): Aggregates space from one or more PVs into LV.
2. Stripped LV: Data is written across multiple PVs.
3. Mirrored LV: Keep mirror copy of LV on another PV.

* To create LV,

Command: lvcreate --size <LV\_size> --name <LV\_name> <VG\_name>

E.g. lvcreate –size 2G –name lv1 vg1

* To display LV,

Command: lvs OR lvdisplay

* To rename LV,

Command: lvrename <VG\_name> <old LV\_name> <new LV\_name>

E.g. lvrename vg1 oldlv1 newlv1

* To extend LV,

Command: lvextend -L <+size to be increased> <path of LV>

E.g., lvextend -L +2G /dev/vg\_vaibhav/var

* To remove LV,

Command: lvremove <LV\_name>

E.g. lvremove /dev/vg1/lv1