8.3.3 LVM Facts

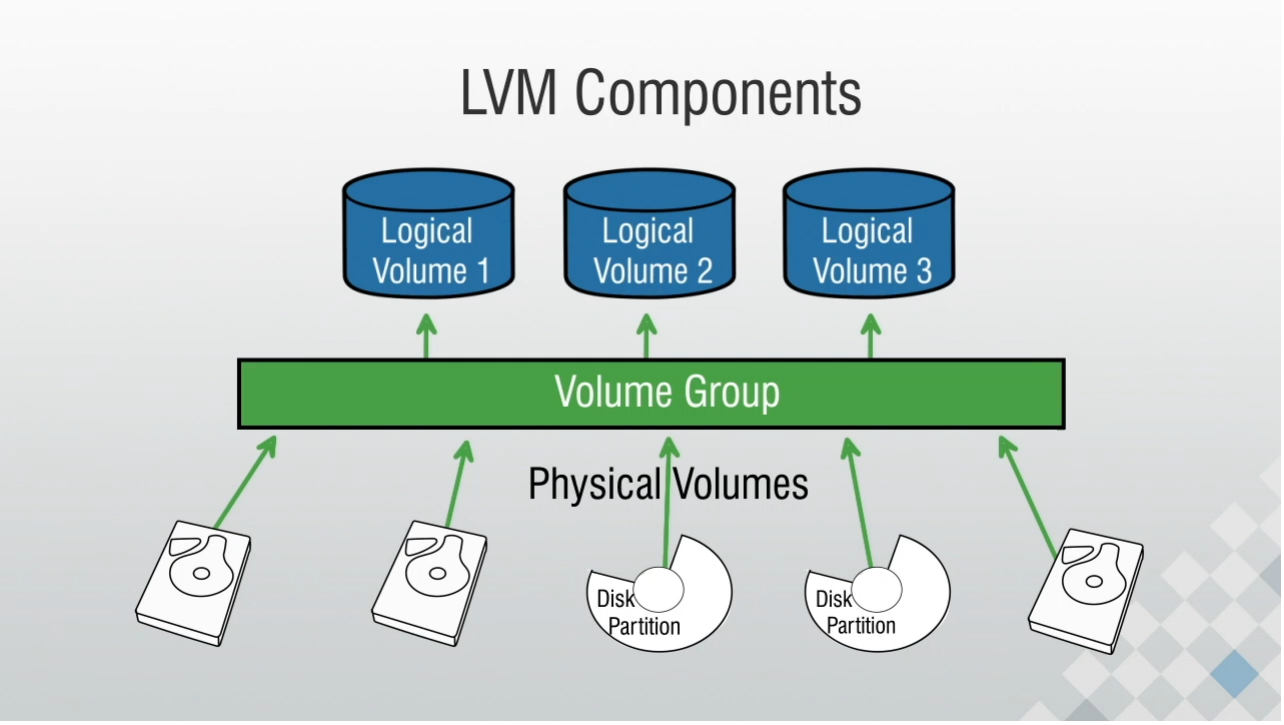
The Logical Volume Manager (LVM) provides an alternative method to managing partitions on a Linux system. LVM gives a system administrator more flexibility in allocating storage on a system.

This lesson covers the following topics:

- LVM components
- LVM commands
- Device mapper
- Persistent block device naming

LVM Components

As shown in the illustration and outlined in the table below, LVM has three components that must be configured to manage storage.



Component	Description
Physical volume	Physical volumes are physical block devices or other disk-like devices that are used by LVM as the building blocks for volume groups. Physical volumes can be: • A regular storages device such as a whole hard disk. • A partition on a hard disk. • A device created by the device mapper, like a RAID array.

Volume group	LVM combines physical volumes into storage pools known as volume groups. The volume group consists of all the space available on the physical volumes grouped together. • The storage space within a volume group can come from many different physical volumes on many different storage devices. • Additional hard disks or additional partitions can be added to the volume group at any time.
Logical volume	A volume group can be divided up into any number of logical volumes. • Logical volumes are the primary component that users and applications will interact with. • Logical volumes are functionally equivalent to a partitions on a physical disk. • Logical volumes can be formatted to accommodate a file system. • Logical volumes can be resized and moved while they are still mounted and running. • Logical volumes may be identified by using descriptive names (e.g., research or marketing) instead of physical disk names such as sda and sdb.

LVM Commands

The following table describes common LVM commands:

Command	Description	Examples
pvcreate	Initializes physical volume for later use by the LVM.	**pvcreate /dev/sdb** creates a physical volume from the second hard disk in the system. **pvcreate /dev/sdd1** creates a physical volume from the first partition on the fourth hard disk in the system.
pvscan	Scans all disks for physical volumes and displays all found physical volumes on the system and their associated volume groups.	
vgcreate	Creates a new volume group.	**vgcreate system /dev/sdb** creates a volume group named system and adds the /dev/sdb physical volume to the group. **vgcreate backup /dev/sdd** creates a volume group named backup and adds the /dev/sdd physical volume to the group.
vgextend		

	Adds one or more physical volumes to an existing volume group, increasing its available storage space.	**vgextend system /dev/sdc** adds the /dev/sdc physical volume to the existing system volume group.

lvcreate	Creates a new logical volume from the space available in a volume group. Options include:
	• **-L** specifies the size. Use the following size suffixes:
	  ◦ **K** for kilobytes
	  ◦ **M** for megabytes
	  ◦ **G** for gigabytes
	  ◦ **T** for terabytes
	  ◦ **P** for petabytes
	  ◦ **E** for exabytes
	• **-n** specifies the name.
		lvcreate -L 20G -n data system creates a 20 GB logical volume named data in the system volume group.
		lvcreate -L 2T -n Storage1 backup creates a 2 TB logical volume named Storage1 in the backup volume group.
		lvcreate -L 1T -n Storage2 backup creates a 1 TB logical volume named Storage2 in the backup volume group.

lvscan	Scans all known volume groups in the system for logical volumes and displays the result.	
lvextend	Extends the size of a logical volume. The **-L** option is used to specify the new size of the volume.	**lvextend -L 30G data** extends the data logical volume to a total of 30 Gigabytes. **lvextend -L +10G data** extends the data logical volume by another 10 Gigabytes.

You can also use the following commands to manage an LVM-based storage configuration:

- Use **pvmove** to move the data from one physical volume to another physical volume.
- Use **pvremove** to remove the LVM label from a device so that it will no longer be recognized as a physical volume.
- Use **vgreduce** to remove a physical volume from an existing volume group. Before running vgreduce, you must use pvmove to shift the data to another physical volume.
- Use **vgremove** to delete a volume group. Before you can use vgremove, you must first remove all logical volumes that have been defined in the volume group.
- Use **lvreduce** to reduce the size of a logical volume.
- Use **lvremove** to remove a logical volume from the system.

After your logical volumes have been created, you need to create file systems on them and then mount them:

- You create a file system using mkfs, just as with traditional partitions. Use the following syntax: **mkfs -t** ***file_system*** **/dev/***volume_group/logical_volume*
- You mount a logical volume using the mount command, just as you would to mount file systems on traditional partitions. Use the following syntax: **mount -t** ***file_system***

/dev/*volume_group*/*logical_volume* /*mount_point*

Device Mapper

The device mapper is a Linux kernel framework for mapping physical block devices onto higher-level virtual block devices.

- The functions of the device mapper are used by LVM to create logical volumes and software RAIDs.
- Logical volumes and RAIDs can be found in the /dev/mapper directory.
- The **mdadm** utility can be used to manage and monitor software RAID devices.

Persistent block device naming

If a Linux system has more than one SATA, SCSI or IDE disk controller, the order in which their corresponding device nodes are added may change between bootups. When this happens, device names like /dev/sda and /dev/sdb may change on each boot. Persistent block device naming solves this issue.

- Block devices are given a persistent name using the devices ID, UUID or path name
    - The device's hardware serial number is used to create a persistent name which is added as a symbolic link in the /dev/disk/by-id directory.
    - The UUID of the file system on the device is used to create a persistent name which is added as a symbolic link in the /dev/disk/by-uuid directory.
    - The shortest physical path as determined by the sysf pseudo file system is used to create a persistent name which is added as a symbolic link in the /dev/disk/by-path directory.
- If device mapper multi-pathing is used, a persistent name will be added to the /dev/disk/by-multipath directory.
- Persistent names that are referenced in scripts will always point to the same disk device.
