

Linux System Administration Project Documentation

Project Title: Linux System Administration - Essential Tasks

Author: Divya M

Project Overview:

This project demonstrates proficiency in Linux system administration by setting up and managing a Linux environment with essential configurations. It includes disk management, Logical Volume Management (LVM), user management, networking configurations, and file permissions

Objectives:

Implement disk mounting and LVM for efficient storage management.
Manage users and permissions to ensure security.
Configure network settings for seamless connectivity.
Use key Linux commands for troubleshooting and performance monitoring.

1. Introduction

Linux system administration is a crucial skill for managing servers and IT infrastructure. This project demonstrates essential Linux administration tasks, including disk mounting, LVM, file permissions, disk management, networking commands, and user management. It provides hands-on experience with real-world scenarios.

2. Disk Mounting and Management

Objective:

Learn how to mount and manage storage devices in Linux.

Disk mounting and management in Linux involve attaching storage devices and partitions to specific locations in the system, allowing users to access, manipulate, and organize their data effectively.

Disk Mounting and Management Commands in Linux

1. Checking Disk Information

- **lsblk** – Lists all block devices and their mount points.

- **blkid** – Displays filesystem type and UUID of partitions.
- **df -h** – Shows available and used disk space in a human-readable format.
- **mount | column -t** – Lists currently mounted filesystems.

2. Mounting a Disk

- **mount /dev/sdX1 /mnt** – Mounts partition /dev/sdX1 to the /mnt directory.
- **mount -t ext4 /dev/sdX1 /mnt** – Specifies the filesystem type when mounting.
- **mount -o rw,relatime /dev/sdX1 /mnt** – Mounts with specific options (read/write, etc.).
- **umount /mnt** – Unmounts the partition from /mnt.

3. Persistent Mounting (Editing /etc/fstab)

- **nano /etc/fstab** – Opens the file where persistent mounts are configured.
- Example entry for persistent mounting:

UUID=xxxx-yyy /mnt ext4 defaults 0 2

- **mount -a** – Applies all mounts listed in /etc/fstab.

Tasks:

- Identify available disks: **lsblk**, **fdisk -l**
- **Lsblk**

```
aws | Search [Alt+S] | United States (N. Virginia) | divya |

root@ip-172-31-28-56:~# pwd
/root
root@ip-172-31-28-56:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0        7:0      0  26.3M  1 loop /snap/amazon-ssm-agent/9881
loop1        7:1      0  73.9M  1 loop /snap/core22/1722
loop2        7:2      0  44.4M  1 loop /snap/snapd/23545
xvda         202:0    0    8G   0 disk
├─xvda1      202:1    0    7G   0 part /
├─xvda14     202:14   0     4M   0 part
├─xvda15     202:15   0   106M  0 part /boot/efi
└─xvda16     202:16   0   913M  0 part /boot
root@ip-172-31-28-56:~#
```

- If you don't have disk you can add one disk. go to aws console then create one volume and attached to particular instance.
- Lsblk

```
aws | Search [Alt+S] |

root@ip-172-31-28-56:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0        7:0      0  26.3M  1 loop /snap/amazon-ssm-agent/9881
loop1        7:1      0  73.9M  1 loop /snap/core22/1722
loop2        7:2      0  44.4M  1 loop /snap/snapd/23545
xvda         202:0    0    8G   0 disk
├─xvda1      202:1    0    7G   0 part /
├─xvda14     202:14   0     4M   0 part
├─xvda15     202:15   0   106M  0 part /boot/efi
└─xvda16     202:16   0   913M  0 part /boot
xvdx         202:5888  0     1G   0 disk
root@ip-172-31-28-56:~#
```

- Fdisk -l

```

Disk identifier: D7B2CFD6-F96A-42FB-B8E1-13FF6555EB07

Device            Start      End  Sectors  Size Type
/dev/xvda1        2099200 16777182 14677983   7G Linux filesystem
/dev/xvda14         2048    10239    8192    4M BIOS boot
/dev/xvda15        10240   227327   217088  106M EFI System
/dev/xvda16       227328  2097152  1869825  913M Linux extended boot

Partition table entries are not in disk order.

Disk /dev/xvdx: 1 GiB, 1073741824 bytes, 2097152 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
root@ip-172-31-28-56:~#

```

- Create a new partition: `fdisk /dev/xvdx`

```

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

Using default response p.
Partition number (1-4, default 1):
First sector (2048-2097151, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-2097151, default 2097151):

Created a new partition 1 of type 'Linux' and of size 1023 MiB.

Command (m for help): p
Disk /dev/xvdx: 1 GiB, 1073741824 bytes, 2097152 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
Disklabel type: dos
Disk identifier: 0xbc9f2647

Device      Boot Start      End  Sectors  Size Id Type
/dev/xvdx1          2048 2097151 2095104 1023M 83 Linux

Command (m for help): n
All space for primary partitions is in use.

Command (m for help): w

```

```

root@ip-172-31-28-56:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0        7:0      0 26.3M  1 loop /snap/amazon-ssm-agent/9881
loop1        7:1      0 73.9M  1 loop /snap/core22/1722
loop2        7:2      0 44.4M  1 loop /snap/snapd/23545
xvda        202:0     0   8G   0 disk
├─xvda1      202:1     0    7G   0 part /
├─xvda14     202:14    0    4M   0 part
├─xvda15     202:15    0 106M   0 part /boot/efi
└─xvda16     259:0     0 913M   0 part /boot
xvdx        202:5888  0    1G   0 disk
└─xvdx1      202:5889  0 1023M  0 part
root@ip-172-31-28-56:~# █

```

- Format the partition:
- `mkfs.ext4 /dev/xvdx1`

```

root@ip-172-31-28-56:~# mkfs.ext4 /dev/xvdx1
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 261888 4k blocks and 65536 inodes
Filesystem UUID: 4aaaa547-cc23-460b-a0db-ccf39cffa50b
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376

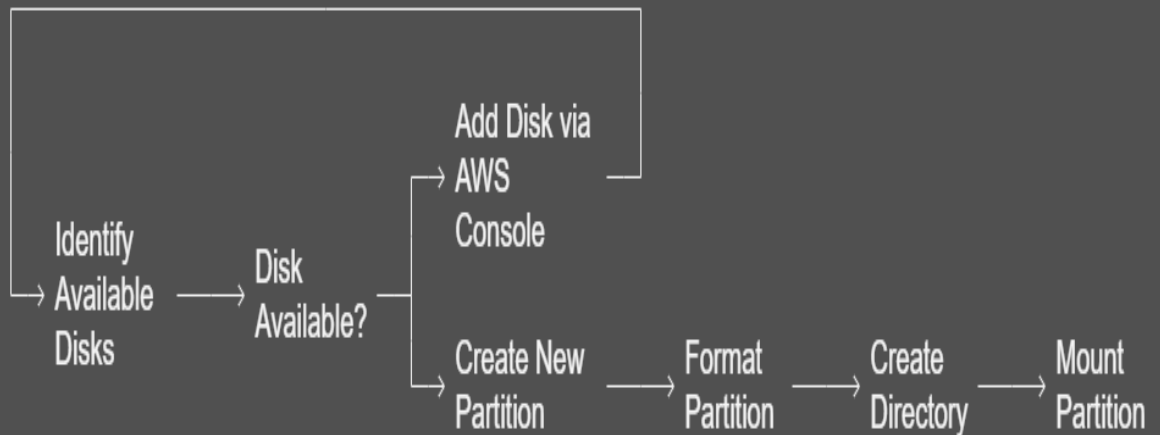
Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

root@ip-172-31-28-56:~# █

```

- create a one directory
- `mkdir /mnt`
- Mount the partition:

Disk Management in Linux

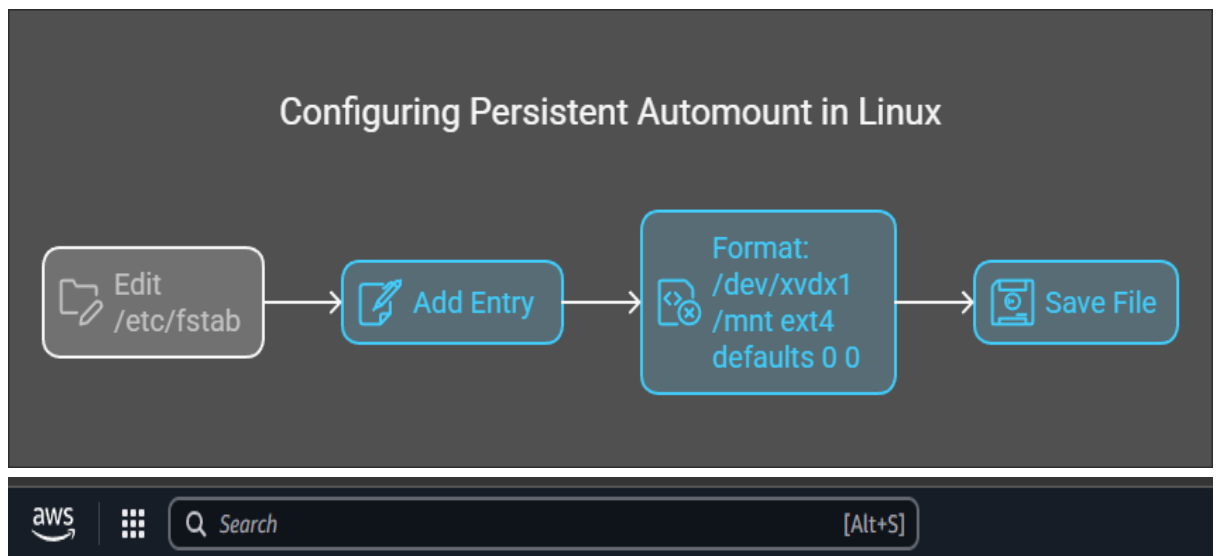


`mount /dev/sdX1 /mnt`

```
xvdx      202:5888  0    1G  0 disk
└─xvdx1   202:5889  0 1023M 0 part
root@ip-172-31-28-56:~# mount /dev/xvdx1 /mnt
root@ip-172-31-28-56:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0        7:0      0   26.3M 1 loop /snap/amazon-ssm-agent/9881
loop1        7:1      0   73.9M 1 loop /snap/core22/1722
loop2        7:2      0   44.4M 1 loop /snap/snapd/23545
xvda        202:0     0    8G  0 disk
├─xvda1     202:1     0    7G  0 part /
├─xvda14    202:14    0    4M  0 part
├─xvda15    202:15    0  106M  0 part /boot/efi
└─xvda16    259:0     0   913M  0 part /boot
xvdx        202:5888  0    1G  0 disk
└─xvdx1     202:5889  0 1023M 0 part /mnt
root@ip-172-31-28-56:~#
```

- Automount using `/etc/fstab`
- We need to the mount persistent use the `/etc/fstab`
- Vi `/etc/fstab`
- `/dev/xvdx1 /mnt ext4 defaults 0 0`

- Save the file



```

root@ip-172-31-28-56:~# sudo vi /etc/fstab
root@ip-172-31-28-56:~# cat /etc/fstab
LABEL=cloudimg-rootfs / ext4 discard,commit=30,errors=remount-ro 0 1
LABEL=BOOT /boot ext4 defaults 0 2
/dev/xvdx1 /mnt ext4 defaults 0 0
LABEL=UEFI /boot/efi vfat umask=0077 0 1
root@ip-172-31-28-56:~#
  
```

3. Logical Volume Management (LVM)

Objective:

Understand LVM concepts and create flexible storage solutions.

Logical Volume Management (LVM) in Linux is a flexible disk management system that allows users to dynamically allocate, resize, and manage storage more efficiently than traditional partitioning methods. LVM provides advanced features like snapshots, resizing, and easy disk management across multiple physical volumes.

Key Concepts in LVM

1. **Physical Volume (PV)** – A physical storage device (e.g., hard drive or partition) initialized for LVM.

2. **Volume Group (VG)** – A collection of physical volumes combined into a single storage pool.
3. **Logical Volume (LV)** – A flexible partition created within a volume group, used like a traditional partition.
4. **File System** – The format applied to a logical volume to store and manage files.

Essential LVM Commands

1. Initializing Physical Volumes

- **pvcreeate /dev/sdX** – Initializes a physical volume for use with LVM.
- **pvdisplay** – Displays information about physical volumes.
- **pvs** – Lists all physical volumes in a concise format.

2. Creating and Managing Volume Groups

- **vgcreate my_vg /dev/sdX /dev/sdY** – Creates a volume group named my_vg using multiple physical volumes.
- **vgextend my_vg /dev/sdZ** – Adds another physical volume to an existing volume group.
- **vgreduce my_vg /dev/sdX** – Removes a physical volume from a volume group.
- **vgdisplay** – Shows detailed information about volume groups.
- **vgs** – Lists all volume groups briefly.

3. Creating and Managing Logical Volumes

- **lvcreate -L 10G -n my_lv my_vg** – Creates a logical volume my_lv of 10GB in my_vg.
- **lvextend -L +5G /dev/my_vg/my_lv** – Expands my_lv by 5GB.
- **lvreduce -L -5G /dev/my_vg/my_lv** – Shrinks my_lv by 5GB.
- **lvremove /dev/my_vg/my_lv** – Deletes a logical volume.
- **lvdisplay** – Shows detailed information about logical volumes.

- `lvs` – Lists logical volumes in a short format.

4. Formatting and Mounting Logical Volumes

- `mkfs.ext4 /dev/my_vg/my_lv` – Formats the logical volume with the ext4 filesystem.
- `mount /dev/my_vg/my_lv /mnt` – Mounts the logical volume to /mnt.
- `umount /mnt` – Unmounts the volume.

5. Taking Snapshots

- `lvcreate -L 5G -s -n my_lv_snapshot /dev/my_vg/my_lv` – Creates a snapshot of a logical volume.
- `lvremove /dev/my_vg/my_lv_snapshot` – Deletes the snapshot.

6. Checking LVM Status

- `lsblk` – Displays all storage devices including LVM partitions.
- `df -h` – Shows available disk space.
- `vgscan` – Scans the system for available volume groups.

Tasks:

- Create Physical Volume (PV): `pvcreate /dev/sdX`

```
root@ip-172-31-82-94:~# pvcreate /dev/xvdx1
Physical volume "/dev/xvdx1" successfully created.
root@ip-172-31-82-94:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0         7:0    0 26.3M  1 loop /snap/amazon-ssm-agent/9881
loop1         7:1    0 73.9M  1 loop /snap/core22/1722
loop2         7:2    0 44.4M  1 loop /snap/snapd/23545
xvda         202:0    0   8G  0 disk
├─xvda1      202:1    0   7G  0 part /
├─xvda14     202:14   0   4M  0 part
├─xvda15     202:15   0 106M  0 part /boot/efi
└─xvda16     259:0    0  913M  0 part /boot
xvdx         202:5888  0   7G  0 disk
├─xvdx1      202:5889  0   2G  0 part
└─xvdx2      202:5890  0   3G  0 part
root@ip-172-31-82-94:~# pvs
PV          VG Fmt Attr PSize PFree
/dev/xvdx1  lvm2 --- 2.00g 2.00g
root@ip-172-31-82-94:~#
```

- Create Volume Group (VG):
- `vgcreate divya_vg /dev/xvdx1`

```
root@ip-172-31-82-94:~# vgcreate divya_vg /dev/xvdx1
Volume group "divya_vg" successfully created
root@ip-172-31-82-94:~# vgs
VG          #PV #LV #SN Attr   VSize  VFree
divya_vg    1   0   0 wz--n- <2.00g <2.00g
root@ip-172-31-82-94:~#
```

- Create Logical Volume (LV):
- `lvcreate -L 1G -n divya_lv divya_vg`

```
root@ip-172-31-82-94:~# lvcreate -L 1G -n divya_lv divya_vg
Logical volume "divya_lv" created.
root@ip-172-31-82-94:~# lvs
LV          VG          Attr      LSize Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
divya_lv    divya_vg  -wi-a----- 1.00g
root@ip-172-31-82-94:~# lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0                                7:0      0 26.3M 1 loop /snap/amazon-ssm-agent/9881
loop1                                7:1      0 73.9M 1 loop /snap/core22/1722
loop2                                7:2      0 44.4M 1 loop /snap/snapd/23545
xvda                                  202:0      0   8G 0 disk
├─xvda1                              202:1      0   7G 0 part /
├─xvda14                             202:14     0   4M 0 part
├─xvda15                             202:15     0 106M 0 part /boot/efi
└─xvda16                             259:0      0 913M 0 part /boot
xvdx                                  202:5888   0   7G 0 disk
├─xvdx1                              202:5889   0   2G 0 part
│   └─divya_vg-divya_lv              252:0      0   1G 0 lvm
└─xvdx2                              202:5890   0   3G 0 part
root@ip-172-31-82-94:~#
```

- Format and mount the LV:
- `mkfs.ext4 /dev/divya_vg/divya_lv`
- `mount:`
- `sudo mount /dev/divya_vg/divya_lv /mnt`

```

root@ip-172-31-82-94:~# mkfs.ext4 /dev/divya_vg/divya_lv
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: c6eb82e7-0003-43ab-92d3-6ab739e1f0a5
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

root@ip-172-31-82-94:~# mount /dev/divya_vg/divya_lv /mnt
root@ip-172-31-82-94:~# df -ht
df: option requires an argument -- 't'
Try 'df --help' for more information.
root@ip-172-31-82-94:~# df -ht
Filesystem                                Type      Size  Used Avail Use% Mounted on
/dev/root                                ext4       6.8G   1.7G   5.1G   26% /
tmpfs                                    tmpfs      479M     0   479M    0% /dev/shm
tmpfs                                    tmpfs      192M   892K   191M    1% /run
tmpfs                                    tmpfs      5.0M     0    5.0M    0% /run/lock
/dev/xvda16                             ext4       881M    76M   744M   10% /boot
/dev/xvda15                             vfat       105M    6.1M    99M    6% /boot/efi
tmpfs                                    tmpfs      96M    12K    96M    1% /run/user/1000
/dev/mapper/divya_vg-divya_lv            ext4       974M    24K   907M    1% /mnt
root@ip-172-31-82-94:~#

```

- if we need extend the logical volume so we need to first unmount
- command:
- `umount /mnt`

```

root@ip-172-31-82-94:~# umount /mnt
root@ip-172-31-82-94:~# df -hT
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/root       ext4       6.8G   1.7G   5.1G   26% /
tmpfs           tmpfs      479M     0   479M    0% /dev/shm
tmpfs           tmpfs      192M   892K   191M    1% /run
tmpfs           tmpfs      5.0M     0    5.0M    0% /run/lock
/dev/xvda16     ext4       881M    76M   744M   10% /boot
/dev/xvda15     vfat       105M    6.1M    99M    6% /boot/efi
tmpfs           tmpfs      96M    12K    96M    1% /run/user/1000
root@ip-172-31-82-94:~# lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0                               7:0      0 26.3M 1 loop /snap/amazon-ssm-agent/9881
loop1                               7:1      0 73.9M 1 loop /snap/core22/1722
loop2                               7:2      0 44.4M 1 loop /snap/snapd/23545
xvda                                202:0     0    8G  0 disk
├─xvda1                            202:1     0    7G  0 part /
├─xvda14                           202:14    0    4M  0 part
├─xvda15                           202:15    0 106M  0 part /boot/efi
└─xvda16                           259:0     0  913M  0 part /boot
xvdx                                202:5888  0    7G  0 disk
├─xvdx1                            202:5889  0    2G  0 part
│   └─divya_vg-divya_lv            252:0     0    1G  0 lvm
└─xvdx2                            202:5890  0    3G  0 part
root@ip-172-31-82-94:~#

```

- Extend an LV:
- `lvextend -L +2G /dev/divya_vg/divya_lv`
- mount command;

- **mount /dev/divya_vg/divya_lv /mnt**

```

root@ip-172-31-82-94:~# lvextend -L +512M /dev/divya_vg/divya_lv
  Size of logical volume divya_vg/divya_lv changed from 1.00 GiB (256 extents) to 1.50 GiB (384 extents).
  Logical volume divya_vg/divya_lv successfully resized.
root@ip-172-31-82-94:~# lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0                               7:0      0  26.3M  1 loop /snap/amazon-ssm-agent/9881
loop1                               7:1      0  73.9M  1 loop /snap/core22/1722
loop2                               7:2      0  44.4M  1 loop /snap/snapd/23545
xvda                                202:0    0    8G  0 disk 
├─xvda1                             202:1    0    7G  0 part /
├─xvda14                            202:14   0    4M  0 part 
├─xvda15                            202:15   0   106M  0 part /boot/efi
└─xvda16                            259:0    0   913M  0 part /boot
xvdx                                202:5888 0    7G  0 disk 
├─xvdx1                             202:5889 0    2G  0 part 
└─┬-divya_vg-divya_lv               252:0    0   1.5G  0 lvm 
  └─xvdx2                             202:5890 0    3G  0 part 
root@ip-172-31-82-94:~# mount /dev/divya_vg/divya_lv /mnt
root@ip-172-31-82-94:~# df -hT
Filesystem                                Type      Size  Used Avail Use% Mounted on
/dev/root                                ext4       6.8G   1.7G   5.1G  26% /
tmpfs                                    tmpfs      479M    0  479M   0% /dev/shm
tmpfs                                    tmpfs     192M  892K   191M   1% /run
tmpfs                                    tmpfs      5.0M    0   5.0M   0% /run/lock
/dev/xvda16                             ext4      881M    76M   744M  10% /boot
/dev/xvda15                             vfat     105M    6.1M    99M   6% /boot/efi
tmpfs                                    tmpfs      96M    12K    96M   1% /run/user/1000
/dev/mapper/divya_vg-divya_lv            ext4      974M    24K   907M   1% /mnt
root@ip-172-31-82-94:~#

```

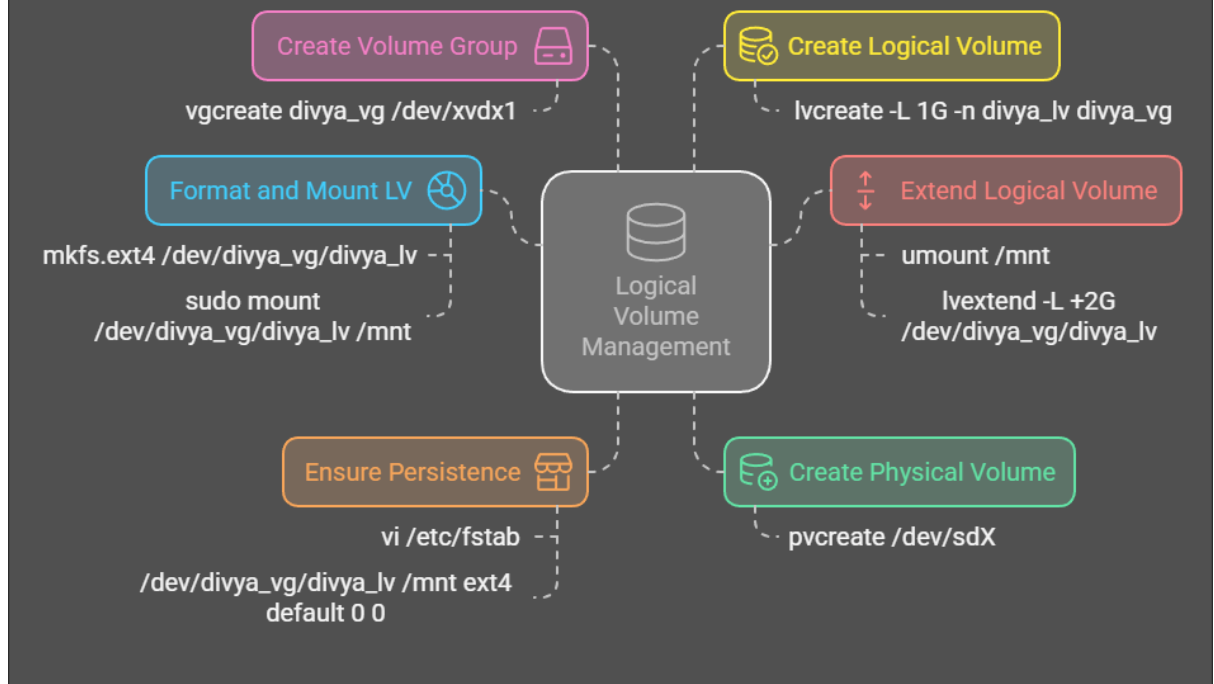
- if we need logical volume management persistence using add entry in /etc/fstab
- command: vi /etc/fstab
- /dev/divya_vg/divya_lv /mnt ext4 default 0 0
- Even system will reboot the mount point also there

```

root@ip-172-31-82-94:~# df -hT
Filesystem                                Type      Size  Used Avail Use% Mounted on
/dev/root                                ext4       6.8G   1.7G   5.1G  26% /
tmpfs                                    tmpfs      479M    0  479M   0% /dev/shm
tmpfs                                    tmpfs     192M  924K   191M   1% /run
tmpfs                                    tmpfs      5.0M    0   5.0M   0% /run/lock
/dev/xvda16                             ext4      881M    76M   744M  10% /boot
/dev/xvda15                             vfat     105M    6.1M    99M   6% /boot/efi
/dev/mapper/divya_vg-divya_lv            ext4      974M    24K   907M   1% /mnt
tmpfs                                    tmpfs      96M    12K    96M   1% /run/user/1000
root@ip-172-31-82-94:~# cat /etc/fstab
LABEL=cloudimg-rootfs / ext4 discard,commit=30,errors=remount-ro 0 1
LABEL=BOOT /boot ext4 defaults 0 2
/dev/mapper/divya_vg-divya_lv /mnt ext4 defaults 0 0
LABEL=UEFI /boot/efi vfat umask=0077 0 1
root@ip-172-31-82-94:~#

```

Linux Logical Volume Management: Steps and Commands



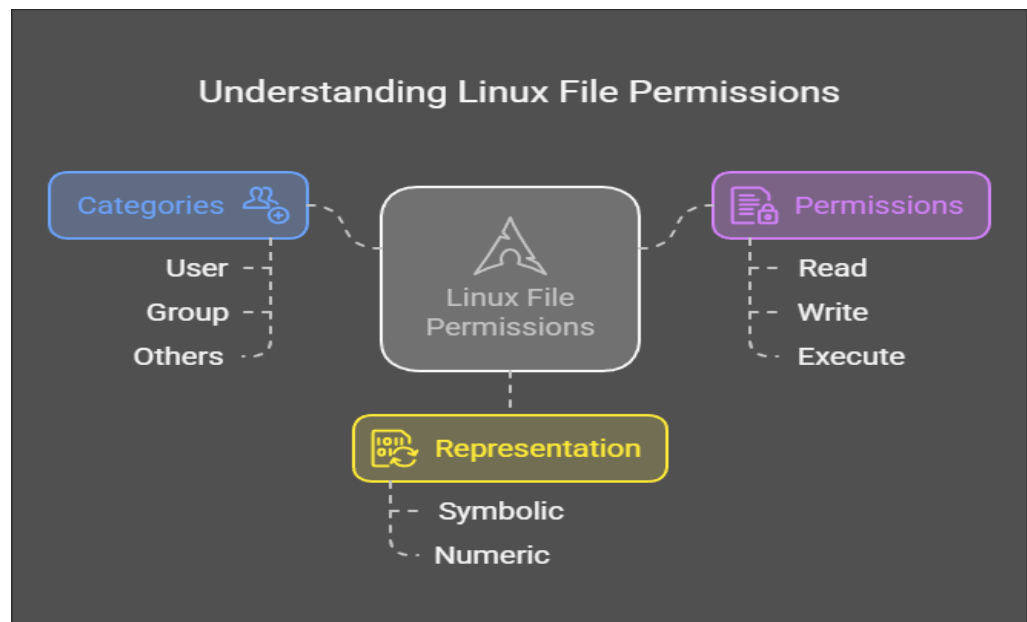
4. File Permissions and Management

Objective:

- File permission and management in Linux ensure that users and processes have appropriate access to files while maintaining security. Linux assigns permissions based on users, groups, and access modes—read, write, and execute.
- Understanding File Permissions in Linux
- Each file has three types of access permissions:
- User (Owner): Permissions for the file's owner.
- Group: Permissions for the group associated with the file.
- Others: Permissions for all other users.
- Each file also has three types of access rights:
- Read (r): Allows viewing the file's content.
- Write (w): Allows modifying the file's content.
- Execute (x): Allows running the file as a program/script.

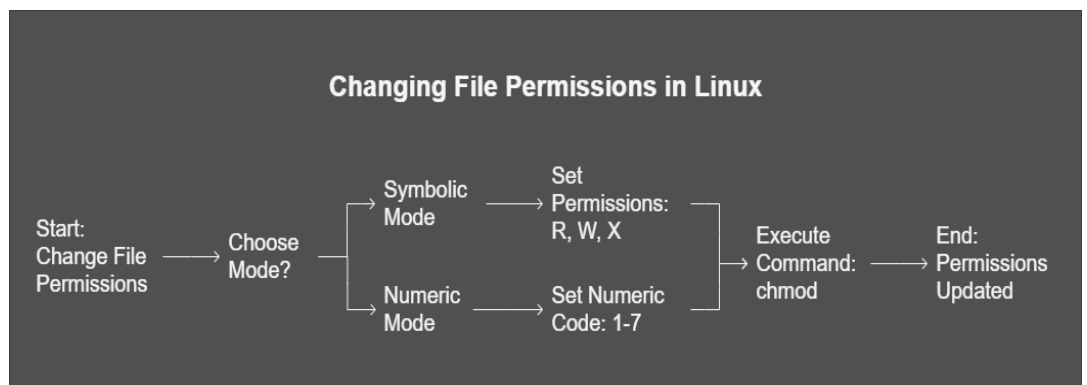
- Permissions are displayed using `ls -l`, like this:
- `-rw-r--r-- 1 user group 1234 May 21 file.txt`
- Where:
- `rw-` (Owner) = Read, Write.
- `r--` (Group) = Read-only.
- `r--` (Others) = Read-only.
- File Permission Commands
- 1. Checking File Permissions
- `ls -l` – Displays permissions of a file.
- `stat` – Provides detailed file information, including permissions.
- 2. Changing File Permissions
- `chmod 755` – Sets specific permissions using numeric mode (7=RWX, 5=RX).
- `chmod u+x` – Grants execute permission to the file owner (u=user).
- `chmod g-w` – Removes write permission from the group (g=group).
- `chmod o+r` – Gives read permission to others (o=others).
- 3. Changing Ownership
- `chown user` – Changes the owner of a file.
- `chown user:group` – Changes both user and group ownership.
- `chgrp group` – Changes the group ownership of a file.
- 4. Managing Special Permissions
- `chmod +s` – Sets the SetUID bit, allowing execution with owner's privileges.
- `chmod +g` – Sets the SetGID bit, making a file inherit the group of the directory.
- `chmod +t` – Sets the Sticky Bit, allowing only the owner to delete files.
- File Management Commands
- 5. Creating and Removing Files
- `touch` – Creates an empty file.
- `mkdir` – Creates a new directory.

- **rm** – Deletes a file.
- **rmdir** – Deletes an empty directory.
- **rm -r** – Deletes a directory with its contents.
- **6. Copying and Moving Files**
- **cp** – Copies files.
- **mv** – Moves or renames files.
- **cp -r** – Copies directories recursively.
- **7. Viewing and Editing Files**
- **cat** – Displays the file content.
- **nano** – Opens the file in a simple text editor.
- **vim** – Opens the file in a powerful text editor.
- **Understand file permissions and security in Linux.**
- **In linux file permissions like read,write,execute.**
- **There are three positions [known as permission gets]to represent permissions for each category**
- **Rwx|rwx|rwx**
| u | g | o |
U=user, g=group, o=others



- **Change file permission in two ways one is numeric mode another one is symbolic mode.**
- **Symbolic mode** **numeric mode**
- **R=read** **+**

- **W=write** -
- **X=execute** =
- **=no permission** 1,2,3,4,5,6,7
- **111-only execution,222-only write,333-write and execution,444-only read 555-read and execution 666-read and write 777-read and write, execution.**
- **File permission change use command: chmod [-R] permission file/dir.**



Tasks:

- **View file permissions: ls -l**
- **Modify permissions using chmod command: chmod permission file/directory**
- **Example: chmod 755 divya**
- **Chmod -R 644 hello**


```
aws | [Search] [Alt+S]

root@ip-172-31-88-111:~# ls
snap
root@ip-172-31-88-111:~# ls -l
total 4
drwx----- 3 root root 4096 Mar 17 15:19 snap
root@ip-172-31-88-111:~# touch divya
root@ip-172-31-88-111:~# mkdir hello
root@ip-172-31-88-111:~# ls -l
total 8
-rw-r--r-- 1 root root 0 Mar 17 15:23 divya
drwxr-xr-x 2 root root 4096 Mar 17 15:24 hello
drwx----- 3 root root 4096 Mar 17 15:19 snap
root@ip-172-31-88-111:~# chmod 755 divya
root@ip-172-31-88-111:~# ls -l
total 8
-rwxr-xr-x 1 root root 0 Mar 17 15:23 divya
drwxr-xr-x 2 root root 4096 Mar 17 15:24 hello
drwx----- 3 root root 4096 Mar 17 15:19 snap
root@ip-172-31-88-111:~# chmod -R 644 hello
root@ip-172-31-88-111:~# ls -l
total 8
-rwxr-xr-x 1 root root 0 Mar 17 15:23 divya
drw-r--r-- 2 root root 4096 Mar 17 15:24 hello
drwx----- 3 root root 4096 Mar 17 15:19 snap
root@ip-172-31-88-111:~#
```

Permission Breakdown

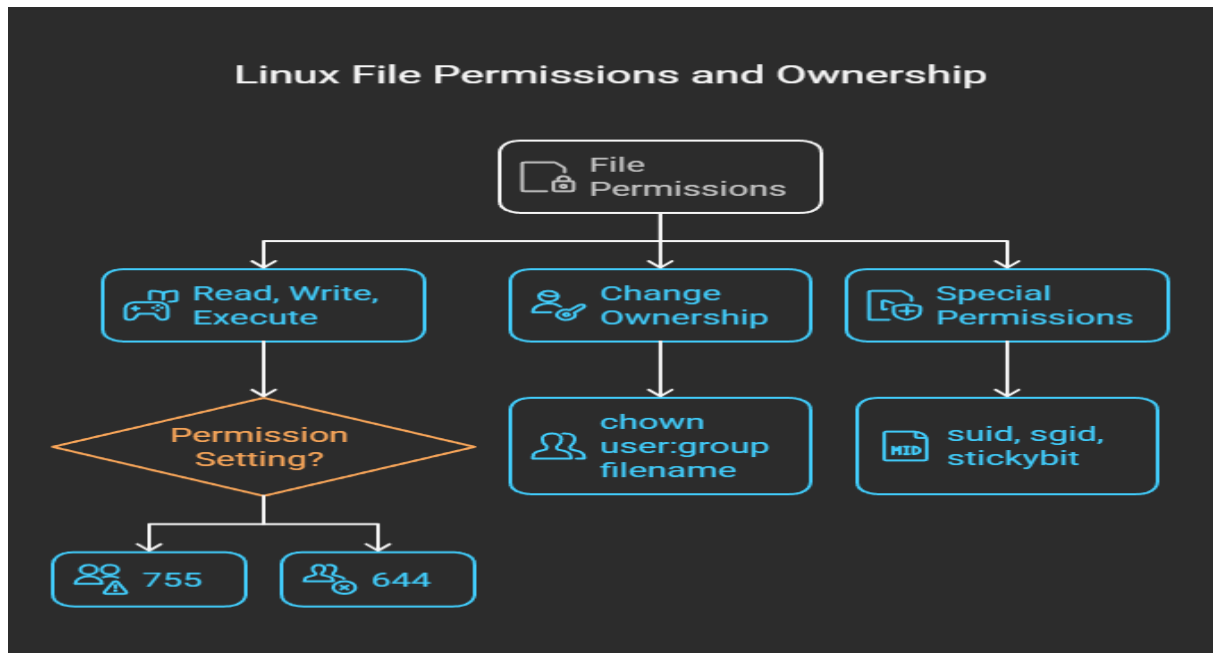
- Read (4), Write (2), Execute (1):
 - 755 = Owner (rwx), Group (rx), Others (rx).
 - 644 = Owner (rw), Group (r), Others (r).
- Change ownership: chown user:group filename
- Example: chown padhu:sun kernel.txt
- Here padhu is user, sun is group filename is kernel.txt

```

-rw-r--r--  1 root root   21 Mar 18 05:46 kernel.txt
drwx-----  3 root root 4096 Mar 18 05:39 snap/
root@ip-172-31-23-168:~# useradd padhu
root@ip-172-31-23-168:~# passwd padhu
New password:
Retype new password:
passwd: password updated successfully
root@ip-172-31-23-168:~# chown padhu kernel.txt
root@ip-172-31-23-168:~# ll
total 32
drwx-----  4 root  root 4096 Mar 18 05:46 ./
drwxr-xr-x 22 root  root 4096 Mar 18 05:39 ../
-rw-r--r--  1 root  root 3106 Apr 22  2024 .bashrc
-rw-r--r--  1 root  root  161 Apr 22  2024 .profile
drwx-----  2 root  root 4096 Mar 18 05:39 .ssh/
-rw-----  1 root  root  746 Mar 18 05:46 .viminfo
-rw-r--r--  1 padhu root   21 Mar 18 05:46 kernel.txt
drwx-----  3 root  root 4096 Mar 18 05:39 snap/
root@ip-172-31-23-168:~# groupadd sun
root@ip-172-31-23-168:~# chown padhu:sun kernel.txt
root@ip-172-31-23-168:~# ll
total 32
drwx-----  4 root  root 4096 Mar 18 05:46 ./
drwxr-xr-x 22 root  root 4096 Mar 18 05:39 ../
-rw-r--r--  1 root  root 3106 Apr 22  2024 .bashrc
-rw-r--r--  1 root  root  161 Apr 22  2024 .profile
drwx-----  2 root  root 4096 Mar 18 05:39 .ssh/
-rw-----  1 root  root  746 Mar 18 05:46 .viminfo
-rw-r--r--  1 padhu sun   21 Mar 18 05:46 kernel.txt
drwx-----  3 root  root 4096 Mar 18 05:39 snap/

```

- Some special file permission there in linux i.e, suid, sgid, stickybit .



- Set special permissions:
- SUID: `chmod u+s filename`
- SGID: `chmod g+s directory_name`
- Sticky Bit: `chmod +t directory_name`
- SUID(set user id): allows to execute a program with the privileges of the file's owner, not their own.
- SGID(set group id): enables a user to execute a program with the privileges of the file's group not their own
- STICKY BIT: when set on directory, it restricts deletion or renaming of files and subdirectories within that directory to only the owner, the directory owner, or root.
- Example SUID: `chmod u+s hello.txt`

```

root@ip-172-31-23-168:~# vi heelo.txt
root@ip-172-31-23-168:~# chown padhu heelo.txt
root@ip-172-31-23-168:~# chmod u+s heelo.txt
root@ip-172-31-23-168:~# ll
total 44
drwx----- 4 root root 4096 Mar 18 06:35 ./
drwxr-xr-x 22 root root 4096 Mar 18 05:39 ../
-rw----- 1 root root 183 Mar 18 05:58 .bash_history
-rw-r--r-- 1 root root 3106 Apr 22 2024 .bashrc
-rw-r--r-- 1 root root 161 Apr 22 2024 .profile
drwx----- 2 root root 4096 Mar 18 05:39 .ssh/
-rw----- 1 root root 1500 Mar 18 06:35 .viminfo
-rw-r--r-- 1 padhu root 70 Mar 18 06:20 file.txt
-rwSr--r-- 1 padhu root 17 Mar 18 06:35 heelo.txt
-rwxr-xr-x 1 padhu sun 21 Mar 18 05:46 kernel.txt*
drwx----- 3 root root 4096 Mar 18 05:39 snap/
root@ip-172-31-23-168:~# sudo su di
$ pwd
/root
$ whoami
di
$ vi heelo.txt
$ █

```

- Example SGID: `chmod g+s dir.txt`

```

root@ip-172-31-23-168:~# chmod g+s dir.txt
root@ip-172-31-23-168:~# ll
total 48
drwx----- 5 root root 4096 Mar 18 06:42 ./
drwxr-xr-x 22 root root 4096 Mar 18 05:39 ../
-rw----- 1 root root 183 Mar 18 05:58 .bash_history
-rw-r--r-- 1 root root 3106 Apr 22 2024 .bashrc
-rw-r--r-- 1 root root 161 Apr 22 2024 .profile
drwx----- 2 root root 4096 Mar 18 05:39 .ssh/
-rw----- 1 root root 1500 Mar 18 06:35 .viminfo
drwxr-sr-x 2 root sun 4096 Mar 18 06:42 dir.txt/
-rw-r--r-- 1 padhu root 70 Mar 18 06:20 file.txt
-rwSr--r-- 1 padhu root 17 Mar 18 06:35 heelo.txt
-rwxr-xr-x 1 padhu sun 21 Mar 18 05:46 kernel.txt*
drwx----- 3 root root 4096 Mar 18 05:39 snap/
root@ip-172-31-23-168:~# █

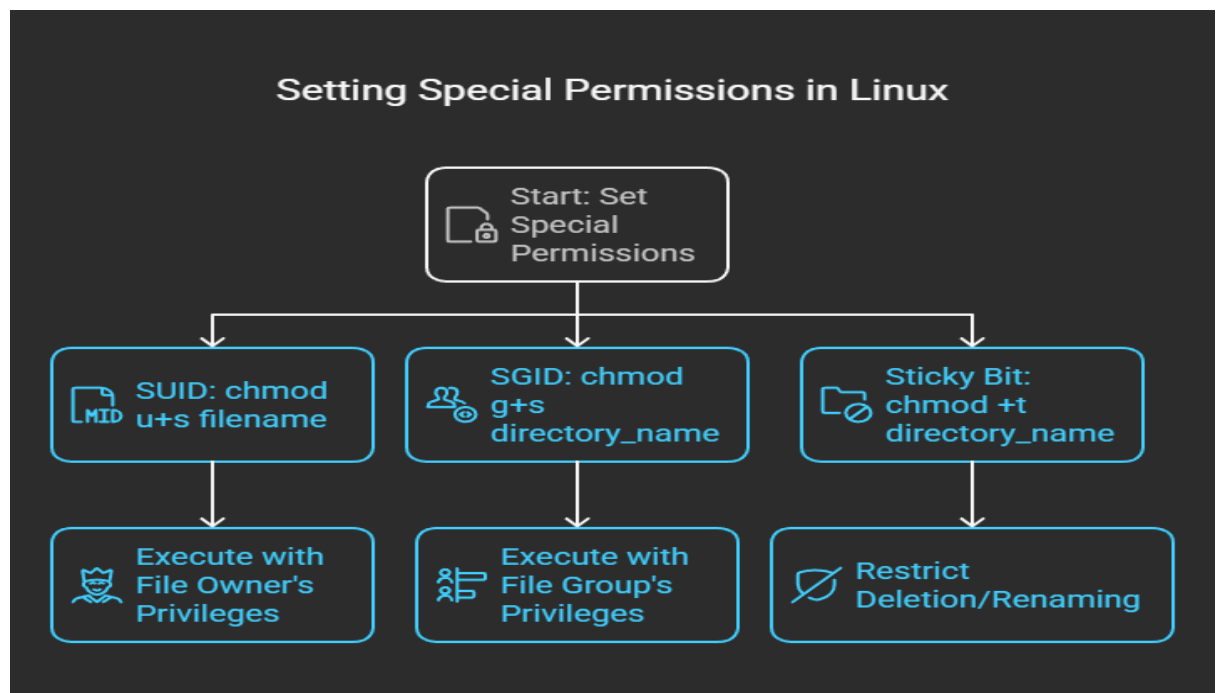
```

- Example STICKY BIT: `chmod +t directory.txt`

```

root@ip-172-31-23-168:~# chmod +t directory.txt
root@ip-172-31-23-168:~# ll
total 52
drwx----- 6 root  root 4096 Mar 18 06:48 ./
drwxr-xr-x 22 root  root 4096 Mar 18 05:39 ../
-rw----- 1 root  root  183 Mar 18 05:58 .bash_history
-rw-r--r-- 1 root  root 3106 Apr 22  2024 .bashrc
-rw-r--r-- 1 root  root  161 Apr 22  2024 .profile
drwx----- 2 root  root 4096 Mar 18 05:39 .ssh/
-rw----- 1 root  root 1500 Mar 18 06:35 .viminfo
drwxr-sr-x 2 root  sun  4096 Mar 18 06:42 dir.txt/
drwxr-xr-t 2 root  root 4096 Mar 18 06:48 directory.txt/
-rw-r--r-- 1 padhu root   70 Mar 18 06:20 file.txt
-rwSr--r-- 1 padhu root   17 Mar 18 06:35 heelo.txt
-rwxr-xr-x 1 padhu sun   21 Mar 18 05:46 kernel.txt*
drwx----- 3 root  root 4096 Mar 18 05:39 snap/
root@ip-172-31-23-168:~#

```



5. Disk Management

Objective:

Monitor and manage disk usage.

Disk management in Linux involves handling storage devices, partitions, and file systems to ensure efficient data organization and

system performance. Administrators use various commands to inspect, modify, format, and mount disks.

Essential Linux Disk Management Commands:

1. Checking Disk Information

- **lsblk** – Lists all block devices, showing disk partitions.
- **fdisk -l** – Displays partition details of all connected storage devices.
- **blkid** – Shows UUID and filesystem type of partitions.
- **df -h** – Displays disk space usage in a human-readable format.
- **du -sh** – Displays the size of a specific directory.

2. Managing Partitions

- **fdisk /dev/sdX** – Opens the partition tool for disk /dev/sdX.
- **parted /dev/sdX** – Another partitioning tool that supports GPT.
- **mkfs.ext4 /dev/sdX1** – Formats partition /dev/sdX1 with the ext4 filesystem.
- **tune2fs -m 5 /dev/sdX1** – Adjusts reserved space percentage on an ext4 partition.

3. Mounting and Unmounting Disks

- **mount /dev/sdX1 /mnt** – Mounts partition /dev/sdX1 to /mnt.
- **umount /mnt** – Unmounts the partition from /mnt.
- **mount -o rw,remount /dev/sdX1** – Remounts a filesystem with read/write access.

4. Managing Logical Volumes (LVM)

- **pvccreate /dev/sdX1** – Initializes a partition for use with LVM.
- **vgcreate my_vg /dev/sdX1** – Creates a volume group named my_vg.
- **lvcreate -L 10G -n my_lv my_vg** – Creates a logical volume my_lv of 10GB.
- **lvextend -L +5G /dev/my_vg/my_lv** – Increases the size of a logical volume by 5GB.
- **resize2fs /dev/my_vg/my_lv** – Resizes the filesystem after extending the volume.

5. Checking and Repairing Filesystems

- **fsck /dev/sdX1** – Checks and repairs filesystem errors.
- **e2fsck -p /dev/sdX1** – Runs automatic repair on an ext4 filesystem.
- **badblocks -v /dev/sdX1** – Scans for bad sectors on a disk.

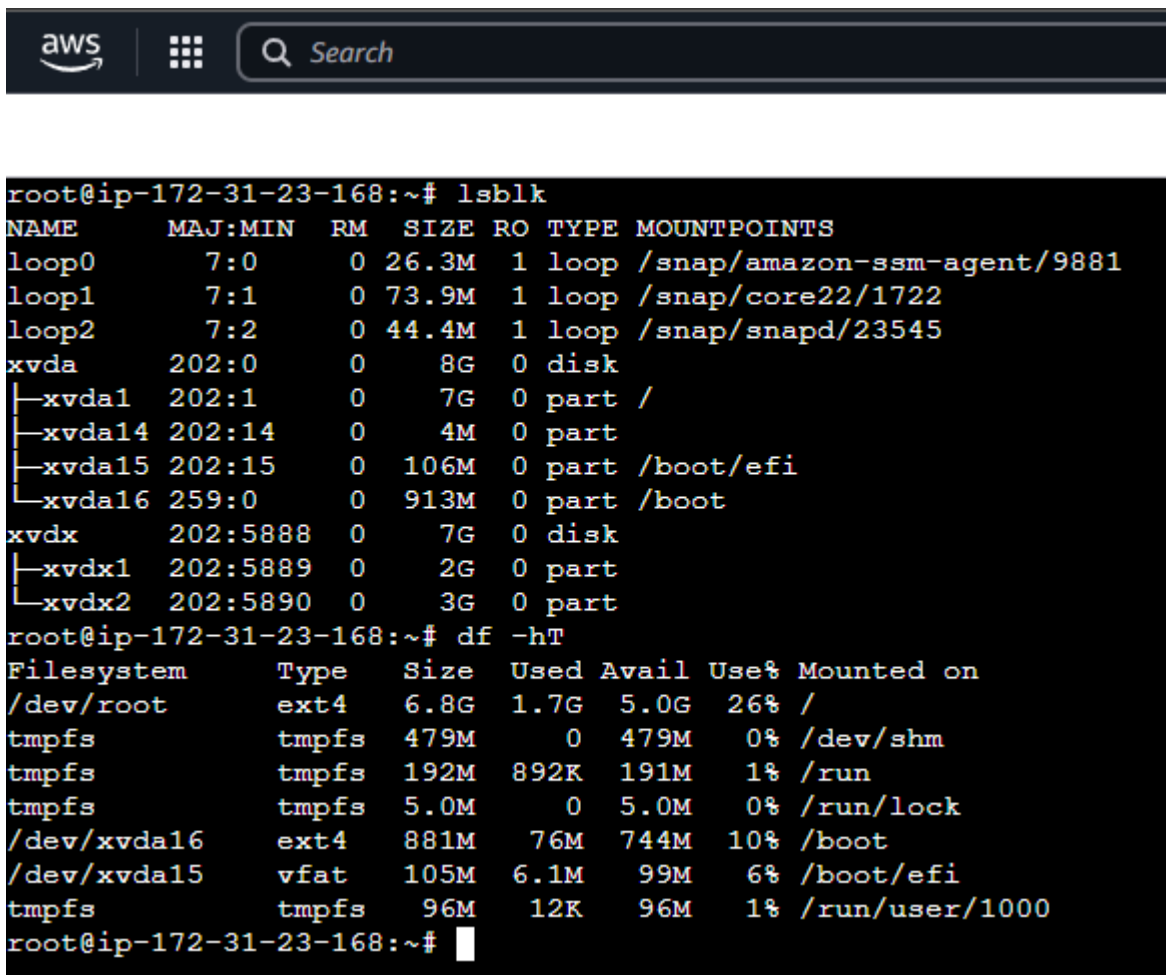
6. Viewing Disk I/O Performance

- iostat – Displays CPU and disk I/O statistics.
- iotop – Shows real-time disk usage per process.
- hdparm -tT /dev/sdX – Measures disk read performance.

Tasks:

Steps

- Monitor usage with df and du.
- Partition and format new disks as needed.
- Check health with smartctl.
- Check disk usage: df -h



The screenshot shows an AWS terminal window with the AWS logo and a search bar at the top. The terminal output displays the results of two commands: `lsblk` and `df -hT`.

```
root@ip-172-31-23-168:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0         7:0    0 26.3M  1 loop /snap/amazon-ssm-agent/9881
loop1         7:1    0 73.9M  1 loop /snap/core22/1722
loop2         7:2    0 44.4M  1 loop /snap/snapd/23545
xvda         202:0    0   8G   0 disk
├─xvda1      202:1    0   7G   0 part /
├─xvda14     202:14   0   4M   0 part
├─xvda15     202:15   0 106M   0 part /boot/efi
└─xvda16     202:16   0 913M   0 part /boot
xvdx         202:5888 0   7G   0 disk
├─xvdx1      202:5889 0   2G   0 part
└─xvdx2      202:5890 0   3G   0 part
root@ip-172-31-23-168:~# df -hT
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/root        ext4      6.8G  1.7G  5.0G  26% /
tmpfs            tmpfs     479M    0  479M   0% /dev/shm
tmpfs            tmpfs     192M  892K  191M   1% /run
tmpfs            tmpfs     5.0M    0   5.0M   0% /run/lock
/dev/xvda16      ext4      881M   76M  744M  10% /boot
/dev/xvda15      vfat      105M   6.1M   99M   6% /boot/efi
tmpfs            tmpfs     96M   12K   96M   1% /run/user/1000
root@ip-172-31-23-168:~#
```

- Check inode usage: df -i

```

root@ip-172-31-23-168:~# df -hi
Filesystem      Inodes  IUsed  IFree  IUse% Mounted on
/dev/root        896K    80K   817K    9% /
tmpfs            120K      2   120K    1% /dev/shm
tmpfs            800K    624   800K    1% /run
tmpfs            120K      3   120K    1% /run/lock
/dev/xvda16       58K    601    57K    2% /boot
/dev/xvda15        0      0      0     - /boot/efi
tmpfs            24K     32    24K    1% /run/user/1000
root@ip-172-31-23-168:~#

```

- Display disk space usage by directory: `du -sh /path`



```

root@ip-172-31-23-168:~# du -h
4.0K    ./snap/amazon-ssm-agent/9881
4.0K    ./snap/amazon-ssm-agent/common
12K     ./snap/amazon-ssm-agent
16K     ./snap
8.0K    ./ssh
4.0K    ./directory.txt
4.0K    ./dir.txt
64K     .
root@ip-172-31-23-168:~# du -h directory.txt
4.0K    directory.txt
root@ip-172-31-23-168:~#

```

- Format a partition:
- `Sudo mkfs.ext4 /dev/xvdx2`


```

root@ip-172-31-23-168:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0         7:0    0 26.3M  1 loop /snap/amazon-ssm-agent/9881
loop1         7:1    0 73.9M  1 loop /snap/core22/1722
loop2         7:2    0 44.4M  1 loop /snap/snapd/23545
xvda         202:0    0   8G   0 disk
├─xvda1      202:1    0   7G   0 part /
├─xvda14     202:14   0   4M   0 part
├─xvda15     202:15   0 106M   0 part /boot/efi
└─xvda16     259:0    0 913M   0 part /boot
xvdx         202:5888  0   7G   0 disk
├─xvdx1      202:5889  0   2G   0 part
└─xvdx2      202:5890  0   3G   0 part
root@ip-172-31-23-168:~# mkfs.ext4 /dev/xvdx2
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 786432 4k blocks and 196608 inodes
Filesystem UUID: e835acc1-2f1f-42ec-8e32-6eec0a6ff419
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

```

- Check disk health:
- `sudo smartctl -a /dev/xvdx2`
- if incase smartcl is command not found you need to install
- `apt install smartmontools` # Or use yum/dnf based on your distro
- `smartctl -a /dev/xvdx2`

```

└─xvda15 202:15      0  106M  0 part /boot/efi
└─xvda16 259:0       0  913M  0 part /boot
xvdx      202:5888    0    7G   0 disk
└─xvdx1   202:5889    0    2G   0 part
└─xvdx2   202:5890    0    3G   0 part
root@ip-172-31-23-168:~# apt update
apt install smartmontools
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
143 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
smartmontools is already the newest version (7.4-2build1).
0 upgraded, 0 newly installed, 0 to remove and 143 not upgraded.
root@ip-172-31-23-168:~# smartctl -a /dev/xvdx2
smartctl 7.4 2023-08-01 r5530 [x86_64-linux-6.8.0-1021-aws] (local build)
Copyright (C) 2002-23, Bruce Allen, Christian Franke, www.smartmontools.org

/dev/xvdx2: Unable to detect device type
Please specify device type with the -d option.

Use smartctl -h to get a usage summary
root@ip-172-31-23-168:~#

```

- Monitor disk I/O: iostat, iotop
- Steps to Install
- Update the package list (optional but recommended to ensure you get the latest versions
- apt update
- Install sysstat for iostat command:
- apt install sysstat
- Install iotop command:
- apt install iotop
- Running the Commands
- After installation, you can use iostat to check disk I/O statistics. For example
- iostat -x 1
- For iotop, run it with sudo to monitor real-time I/O usage by processes
- sudo iotop

```
aws | [Search] [Alt+S]

Try: apt install <deb name>
root@ip-172-31-23-168:~# apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
143 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ip-172-31-23-168:~# apt install sysstat
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
sysstat is already the newest version (12.6.1-2).
sysstat set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 143 not upgraded.
root@ip-172-31-23-168:~# apt install iotop
Reading package lists... Done
Building dependency tree... Done
Progress: [ 80%] [#####]
The following NEW packages will be installed:
  iotop
0 upgraded, 1 newly installed, 0 to remove and 143 not upgraded.
Need to get 24.4 kB of archives.
After this operation, 111 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 iotop amd64
Fetched 24.4 kB in 0s (411 kB/s)
Scanning processes... [
root@ip-172-31-23-168:~# Outdated hypervisor (qemu) binaries on this host.
```

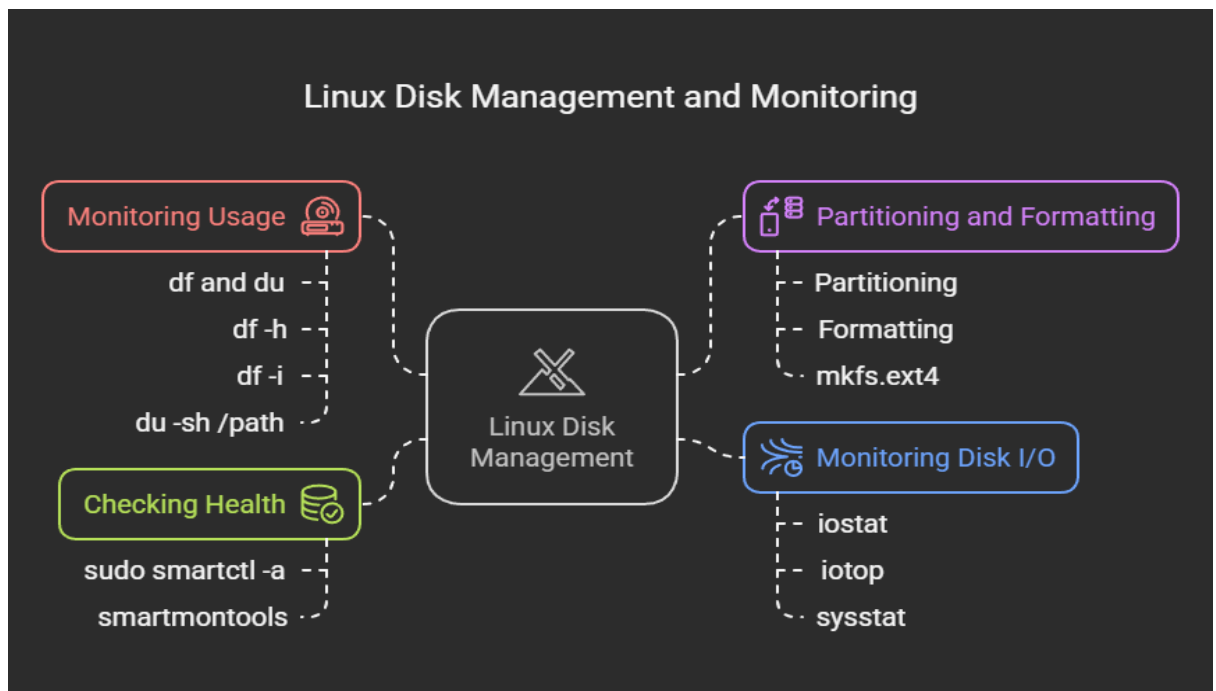
- command: iostat

```
root@ip-172-31-23-168:~# iostat
Linux 6.8.0-1021-aws (ip-172-31-23-168)      03/18/25      _x86_64_      (1 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.37    0.03   0.12   0.14   15.08   84.25

Device            tps    kB_read/s    kB_wrtn/s    kB_dscd/s    kB_read    kB_wrtn    kB_dscd
loop0              0.02         0.47         0.00         0.00        4178         0         0
loop1              0.01         0.12         0.00         0.00       1086         0         0
loop2              0.06         2.04         0.00         0.00      18189         0         0
loop3              0.00         0.00         0.00         0.00         14         0         0
xvda               3.43        57.07       111.47         0.00     508182     992506         0
xvdx               0.13         1.88         7.55         0.00     16744     67248         0

root@ip-172-31-23-168:~#
```



6. Networking Commands

Objective:

Understand basic networking commands.

Networking in Linux refers to the configuration and management of network connections, interfaces, and protocols to enable communication between systems. Linux provides powerful tools and commands for monitoring, troubleshooting, and managing networks.

Essential Linux Network Commands:

1. Checking Network Configuration

- **ifconfig** – Displays network interfaces and IP addresses (deprecated, replaced by **ip** command).
- **ip addr show** – Shows IP addresses assigned to interfaces.
- **ip link show** – Displays network interface details.
- **nmcli device status** – Checks the status of network devices.

2. Managing Network Interfaces

- **ifconfig eth0 down** – Disables a network interface.
- **ifconfig eth0 up** – Enables a network interface.
- **ip link set eth0 down** – Another way to disable an interface.
- **ip link set eth0 up** – Enables an interface using the **ip** command.

- **iwconfig** – Configures wireless interfaces.

3. Checking Network Connections

- **ping** – Tests network connectivity.
- **traceroute** – Shows the path packets take to a destination.
- **netstat -tunlp** – Displays active network connections and listening ports.
- **ss -tunlp** – An improved alternative to netstat.
- **curl -I** – Fetches HTTP headers of a webpage.

4. Managing Network Routing

- **route -n** – Displays the routing table (deprecated, use ip command).
- **ip route show** – Shows the routing table.
- **ip route add via** – Adds a static route.
- **ip route del** – Deletes a static route.

5. DNS Lookup & Host Resolution

- **nslookup** – Queries DNS servers (deprecated, use dig).
- **dig** – Performs a detailed DNS lookup.
- **host** – Resolves hostnames to IP addresses.

6. Port and Firewall Management

- **iptables -L** – Lists firewall rules.
- **iptables -A INPUT -p tcp --dport 80 -j ACCEPT** – Allows traffic on port 80.
- **firewall-cmd --list-all** – Lists firewall settings (for systems using firewalld).
- **ufw status** – Shows firewall status (for systems using ufw).

7. Network Traffic Monitoring

- **tcpdump -i eth0** – Captures network packets on a specified interface.
- **iftop** – Displays live bandwidth usage per connection.
- **nmap -sP** – Scans for active hosts on a network.

Tasks:

Commands

- **Check IP address:**
- **ip addr**

```

root@ip-172-31-80-214:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enX0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc fq_codel state UP group default qlen 1000
    link/ether 12:a9:6c:6e:b5:4b brd ff:ff:ff:ff:ff:ff
    inet 172.31.80.214/20 metric 100 brd 172.31.95.255 scope global dynamic enX0
        valid_lft 3370sec preferred_lft 3370sec
    inet6 fe80::10a9:6cff:fe6e:b54b/64 scope link
        valid_lft forever preferred_lft forever
root@ip-172-31-80-214:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enX0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc fq_codel state UP group default qlen 1000
    link/ether 12:a9:6c:6e:b5:4b brd ff:ff:ff:ff:ff:ff
    inet 172.31.80.214/20 metric 100 brd 172.31.95.255 scope global dynamic enX0
        valid_lft 3360sec preferred_lft 3360sec
    inet6 fe80::10a9:6cff:fe6e:b54b/64 scope link
        valid_lft forever preferred_lft forever
root@ip-172-31-80-214:~#

```

- **Explanation:** Shows network interfaces. lo is the loopback interface (127.0.0.1), and eth0 has an IP of 172.31.80.214.
- **Ping a host:**
- **ping google.com -c 4**

```

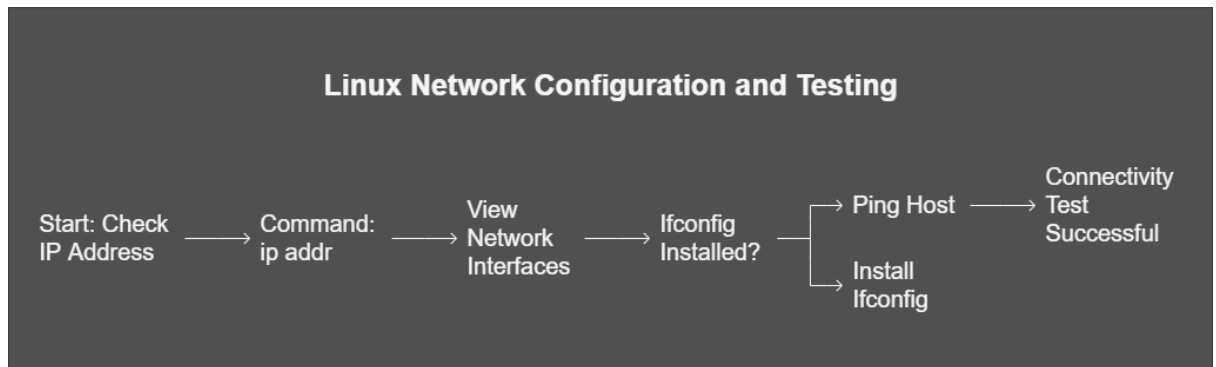
root@ip-172-31-80-214:~# ping google.com -c 4
PING google.com (172.253.122.102) 56(84) bytes of data:
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=1 ttl=106 time=2.07 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=2 ttl=106 time=2.03 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=3 ttl=106 time=2.05 ms
64 bytes from bh-in-f102.1e100.net (172.253.122.102): icmp_seq=4 ttl=106 time=1.95 ms

--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 1.946/2.021/2.067/0.045 ms
root@ip-172-31-80-214:~#

```

- **Explanation:** Tests connectivity to google.com. The -c 4 limits it to 4 pings. Latency is around 15 ms with no packet loss.
- **View network interfaces:**
- **Ifconfig**

- In case this command is not found, so you need to install it.



How to Install ifconfig (net-tools)

Since you're logged in as root (indicated by root@ip-172-31-80-214), you can directly run the installation command without needing sudo.

Step-by-Step Instructions

- **Update the Package List** (optional but recommended):
Ensure your package list is up-to-date to avoid issues with outdated repositories.
- **apt update**

```
Fetchd 32.8 MB in 6s (5309 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
143 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ip-172-31-80-214:~#
```

- **Install net-tools:**
Run the command suggested in the error message:
- **apt install net-tools**

```

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.31.80.214 netmask 255.255.240.0 broadcast 172.31.95.255
    ether 02:00:17:12:34:56 txqueuelen 1000 (Ethernet)
    RX packets 54321 bytes 1234567 (1.2 MB)
    TX packets 12345 bytes 987654 (987.6 KB)

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 100 bytes 8000 (8.0 KB)
    TX packets 100 bytes 8000 (8.0 KB)

```

- ifconfig

```

root@ip-172-31-80-214:~# ifconfig
enx0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
    inet 172.31.80.214 netmask 255.255.240.0 broadcast 172.31.95.255
    inet6 fe80::10a9:6cff:fe6e:b54b prefixlen 64 scopeid 0x20<link>
    ether 12:a9:6c:6e:b5:4b txqueuelen 1000 (Ethernet)
    RX packets 25892 bytes 34758081 (34.7 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 4002 bytes 448716 (448.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 164 bytes 18585 (18.5 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 164 bytes 18585 (18.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@ip-172-31-80-214:~#

```

- Explanation: Displays interface details. eth0 shows the IP, netmask, and traffic stats. Note: ifconfig may require the net-tools package.
- Restart networking (Ubuntu):
- sudo systemctl restart networking
- The error Unit networking.service not found indicates that the networking.service is not available on your system. This is common in newer Linux distributions, especially those using systemd, like Ubuntu 16.04 and later, where traditional networking.service has been replaced by other network management tools such as NetworkManager or systemd-networkd. The exact solution depends on your system's network configuration.
- systemctl status systemd-networkd


```

root@ip-172-31-80-214:~# systemctl status systemd-networkd
● systemd-networkd.service - Network Configuration
   Loaded: loaded (/usr/lib/systemd/system/systemd-networkd.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-03-19 13:17:12 UTC; 18s ago
   TriggeredBy: ● systemd-networkd.socket
     Docs: man:systemd-networkd.service(8).
           man:org.freedesktop.network1(5).
    Main PID: 2553 (systemd-network)
      Status: "Processing requests..."
        Tasks: 1 (limit: 1130)
      FD Store: 0 (limit: 512)
     Memory: 1.5M (peak: 1.8M)
        CPU: 33ms
    CGroup: /system.slice/systemd-networkd.service
            └─2553 /usr/lib/systemd/systemd-networkd

Mar 19 13:17:12 ip-172-31-80-214 systemd[1]: Starting systemd-networkd.service - Network Configuration...
Mar 19 13:17:12 ip-172-31-80-214 systemd-networkd[2553]: lo: Link UP
Mar 19 13:17:12 ip-172-31-80-214 systemd-networkd[2553]: lo: Gained carrier
Mar 19 13:17:12 ip-172-31-80-214 systemd-networkd[2553]: enX0: Link UP
Mar 19 13:17:12 ip-172-31-80-214 systemd-networkd[2553]: enX0: Gained carrier
Mar 19 13:17:12 ip-172-31-80-214 systemd-networkd[2553]: enX0: Gained IPv6LL
Mar 19 13:17:12 ip-172-31-80-214 systemd-networkd[2553]: Enumeration completed
Mar 19 13:17:12 ip-172-31-80-214 systemd[1]: Started systemd-networkd.service - Network Configuration.

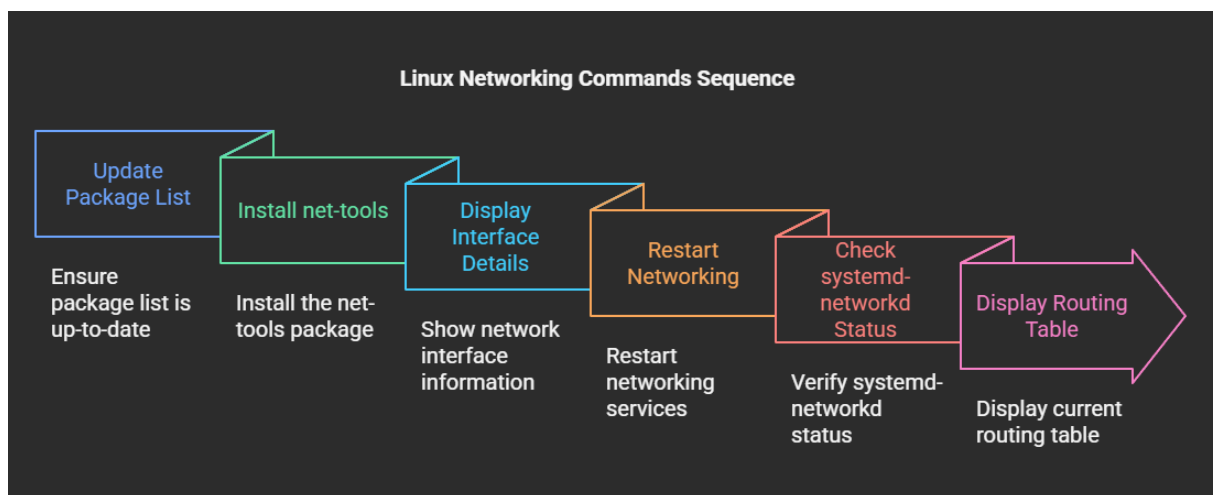
```

- Display routing table:
- route -n

```

Last login: Wed Mar 19 12:59:58 2025 from 18.208.107.29
ubuntu@ip-172-31-80-214:~$ sudo su -
root@ip-172-31-80-214:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         172.31.80.1    0.0.0.0         UG    100    0        0 enX0
172.31.0.2      172.31.80.1    255.255.255.255 UGH    100    0        0 enX0
172.31.80.0     0.0.0.0        255.255.240.0   U    100    0        0 enX0
172.31.80.1     0.0.0.0        255.255.255.255 UH    100    0        0 enX0
root@ip-172-31-80-214:~#

```



7. User Management

Objective:

Manage users and groups in Linux.

User management in Linux is all about handling users and their permissions, ensuring that different people or processes can access the system securely. It involves adding, removing, modifying users, and managing their group memberships.

Key Commands for User Management:

1. Creating a User

- **useradd** – Creates a new user.
- **passwd** – Sets or changes the password for a user.

2. Modifying a User

- **usermod -aG** – Adds a user to a group.
- **usermod -l** – Changes a username.
- **usermod -d /new/home/directory** – Updates the user's home directory.

3. Deleting a User

- **userdel** – Removes a user account.
- **userdel -r** – Deletes a user and their home directory.

4. Managing Groups

- **groupadd** – Creates a new group.
- **groupdel** – Deletes a group.
- **groupmod -n** – Renames a group.

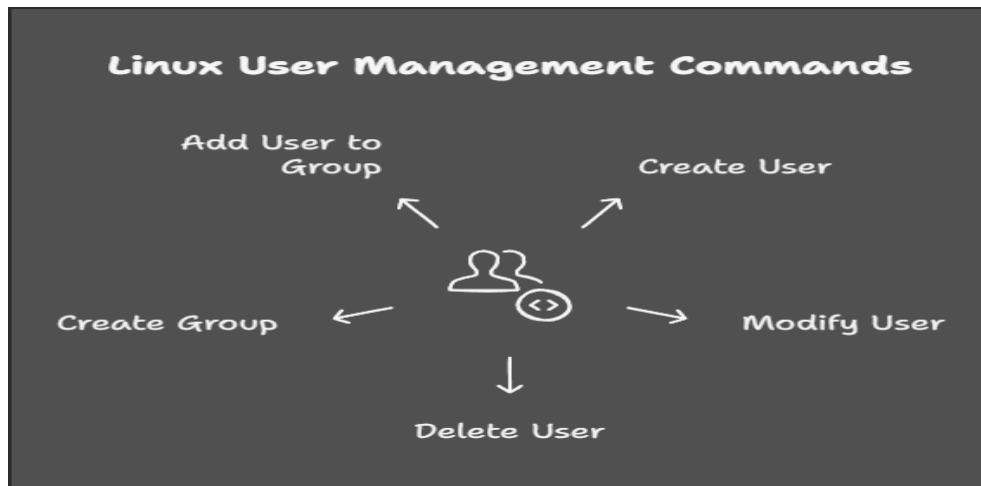
5. Checking User Information

- **id** – Displays the user ID (UID) and group ID (GID).
- **who** or **w** – Shows who is currently logged in.
- **cat /etc/passwd** – Lists all users in the system.

6. Changing User Privileges

- **sudo** – Runs a command as the superuser.
- **visudo** – Edits the sudoers file to manage user privileges.

Linux user management helps maintain security and organization, ensuring users only access what they need. Need help with a specific command?

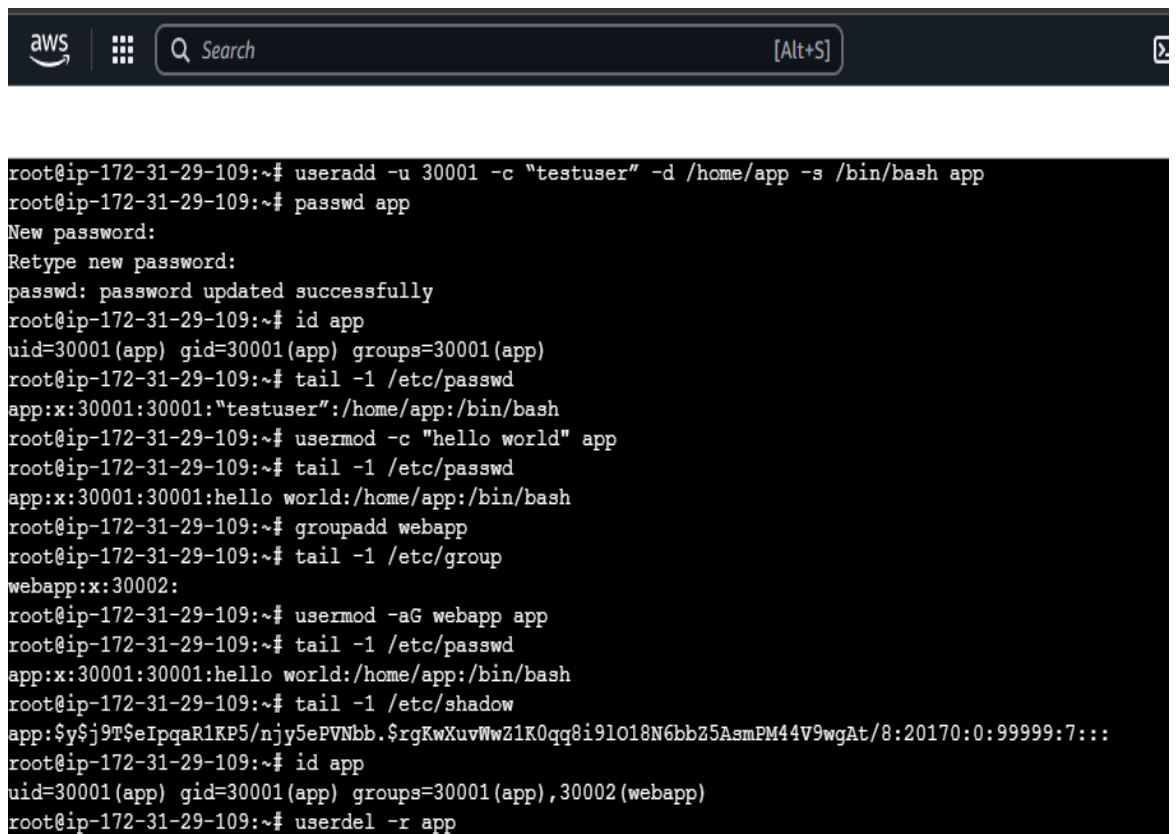


Tasks:

Add a user:

- **Useradd options username**
- **Options:**
- **-u=uid**
- **-g=primary group id**
- **-G= supplementary groups**
- **-c=gicous(comment)**
- **-d=home directory**
- **-s=shell**
- **Set user password**
- **Particular user we need to set password.so here is command**
- **Passwd username.**
- **For example:**
- **Useradd -u 30001 -c "testuser" -d /home/app -s /bin/bash app**
- **Passwd app**
- **Verify with id username.**
- **Id app**
- **Modify user details: usermod -c "New User" username**
- **Example:**

- Usermod -c "hello world" app
- Create a group: groupadd groupname
- Example :
- Groupadd webapp
- Add user to a group: usermod -aG groupname username
- Example:
- Usermod -aG webapp app
- Delete a user: userdel -r username
- Example:
- Userdel -r app

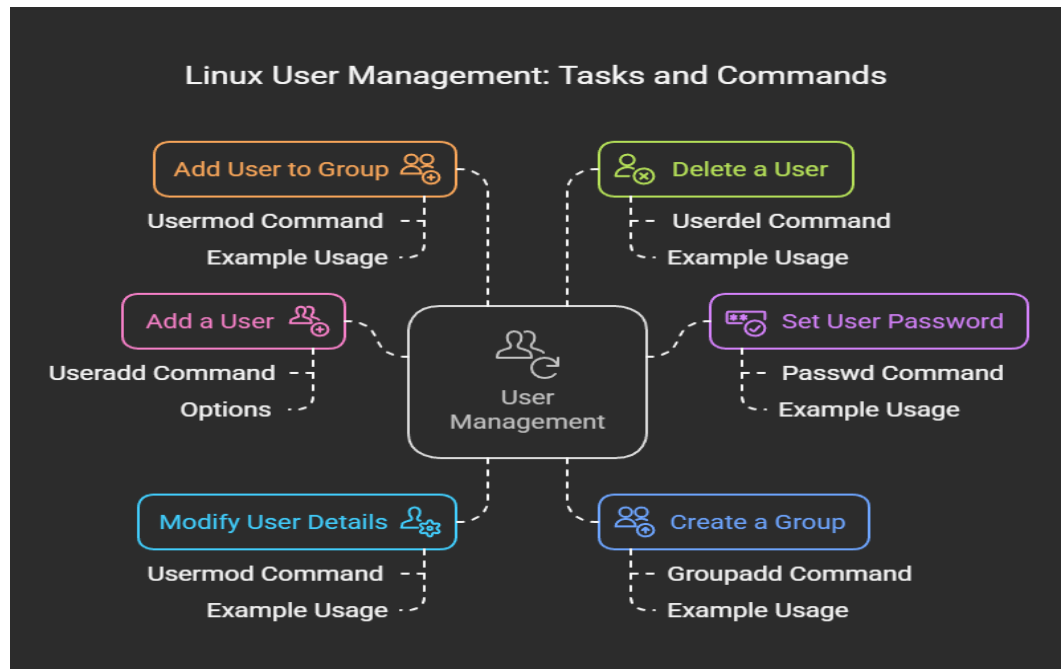


```

aws  [Search] [Alt+S]

root@ip-172-31-29-109:~# useradd -u 30001 -c "testuser" -d /home/app -s /bin/bash app
root@ip-172-31-29-109:~# passwd app
New password:
Retype new password:
passwd: password updated successfully
root@ip-172-31-29-109:~# id app
uid=30001(app) gid=30001(app) groups=30001(app)
root@ip-172-31-29-109:~# tail -1 /etc/passwd
app:x:30001:30001:"testuser":/home/app:/bin/bash
root@ip-172-31-29-109:~# usermod -c "hello world" app
root@ip-172-31-29-109:~# tail -1 /etc/passwd
app:x:30001:30001:hello world:/home/app:/bin/bash
root@ip-172-31-29-109:~# groupadd webapp
root@ip-172-31-29-109:~# tail -1 /etc/group
webapp:x:30002:
root@ip-172-31-29-109:~# usermod -aG webapp app
root@ip-172-31-29-109:~# tail -1 /etc/passwd
app:x:30001:30001:hello world:/home/app:/bin/bash
root@ip-172-31-29-109:~# tail -1 /etc/shadow
app:$y$j9T$IpqaR1KP5/njy5ePVNbb.$rgKwXuvWwZ1K0qq8i9l018N6bbZ5AzmPM44V9wgAt/8:20170:0:99999:7:::
root@ip-172-31-29-109:~# id app
uid=30001(app) gid=30001(app) groups=30001(app),30002(webapp)
root@ip-172-31-29-109:~# userdel -r app

```



8. Conclusion

Mastering Linux system administration is an invaluable skill for IT professionals. This project covers critical administrative tasks, equipping you with practical experience in managing storage, file permissions, networking, and user accounts. By systematically documenting and reporting these activities, you ensure better troubleshooting, enhanced security, and efficient system performance.

