Network Commands-

How to hide users from the GDM login screen/Login screen?

```
# Uncomment the line which says disable-user-list=true in /etc/gdm3/greeter.dconf-
defaults-
```

nano /etc/gdm3/greeter.dconf-defaults

```
# - Disable user list
disable-user-list=true
```

```
# one easy workaround if you want to avoid changing the uid of the user:
```

1. Open the terminal, and enter (replace user with the username you want to hide from the login screen):
2. sudo nano /var/lib/AccountsService/users/user
3. Add the following to the file:
4. [User]
5. Language=
6. XSession=gnome
7. SystemAccount=true
8. Switch user or log out to test if user is not listed anymore.

To list the interfaces-

```
ip link
ifconfig                         # View Active Network Interface Settings
ifconfig | grep inet             # View IP Addresses
ifconfig | grep netmask         # View Network Interface Masks
ifconfig -a                     # Show All/Every Network Interface Configuration
ifconfig -s                    # Display a Shortlist of Active Interfaces
ifconfig -v                    # Print the Verbose Output
ifconfig eth0/enp0s3          # About a Specific Network Interface
ifconfig | grep errors        # View Transmission Errors
ip a
ip add
lshw -C network
```

Enable/Disable(up/down) Network interfaces –

sudo ifconfig [interface_name] up

```
sudo ifconfig eth0 up
sudo ifconfig wlp6s0 up                  # wlp6s0 = adapter name
sudo ip link set eth0 up
```

sudo ifconfig [interface_name] down

```
sudo ifconfig eth0 down
sudo ifconfig wlp6s0 down

if=eth0
sudo ip link set $if down
```

Install ifconfig on Ubuntu/Debian-

```
sudo apt install net-tools

sudo apt install net-tools -y

# Install ifconfig on Centos-

sudo dnf install net-tools -y
```

How to Use the **ifconfig** Command-

```
ifconfig [-a] [-v] [-s] <interface> [[<AF>] <address>]
```

Where: interface - is the name of the network interface.

address - is the IP address that you want to assign.

Assign an IP address and Netmask to a Network Interface-

```
ifconfig [interface-name] [ip-address] netmask [subnet-mask]

ifconfig eth0 192.168.1.111 netmask 255.255.254.0

sudo ifconfig enp0s3 10.0.2.20                   # Assign an IP Address to an Interface

sudo ifconfig enp0s3 netmask 255.255.255.0      # Assign a Netmask to an Interface

sudo ifconfig enp0s3 broadcast 10.0.2.250      # Assign a New Broadcast IP

# sudo ifconfig [interface_name] [IP] netmask [netmask_addresss] broadcast [broadcast_address]

sudo ifconfig enp0s3 10.0.2.15 netmask 255.0.0.0 broadcast 10.0.2.255
```

IP Configuration with vi-

```
sudo vi /etc/network/interfaces
```

Now Add the following--

```
auto eth0
iface eth0 inet static
   address 192.168.1.100
   netmask 255.255.255.0
   gateway 192.168.1.1
dns-nameservers 8.8.8.8 8.8.4.4
pre-up /usr/local/sbin/start-iptables.sh
post-up /usr/local/sbin/backup-log.sh
```

Set an Alias IP-

```
sudo ifconfig [interface_name]:[alias_number] [alias_IP]
sudo ifconfig enp0s3:0 10.0.2.30
sudo ifconfig enp0s3:0 down            # To remove an alias IP
```

Enable Promiscuous Mode-

```
sudo ifconfig [interface_name] promisc
sudo ifconfig enp0s3 promisc        # Enable Promiscuous Mode
sudo ifconfig enp0s3 -promisc       # Disable Promiscuous Mode
```

Control network interfaces through Systemctl command-

```
sudo systemctl status NetworkManager.service
sudo systemctl stop NetworkManager.service
sudo systemctl disable NetworkManager.service
sudo systemctl restart NetworkManager.service
sudo systemctl enable NetworkManager.service
```

User related commands-

Adding a user-

You should have root/Admin Privileged for add users.

```
adduser username
adduser user
sudo adduser user
```

useradd Command-

```
useradd [OPTIONS] USERNAME
```

```
sudo useradd username
```

# When executed without any option, useradd creates a new user account using the default settings specified in the /etc/default/useradd file.

# The command adds an entry to the /etc/passwd, /etc/shadow, /etc/group and /etc/gshadow files.

```
sudo useradd -m username
#Use the -m (--create-home) option to create the user home directory as /home/username
```

Creating a User with Specific Home Directory-

# Here is an example showing how to create a new user named username with a home directory of /opt/username

```
sudo useradd -m -d /opt/username username  # -d define directory of user
```

Creating a User with Specific User ID-

```
sudo useradd -u 1500 username      #  you can verify user id by running - id -u username
```

Creating a User with Specific Group ID-

The following example shows how to create a new user named username and set the login group to users type:

```
sudo useradd -g users username         # verify user id by running-  id -gn username
```

Creating a User and Assign Multiple Groups-

The following command creates a new user named username with primary group users and secondary groups wheel and docker

```
sudo useradd -g users -G wheel, docker username

sudo useradd -g users -G wheel, developers akkc
```

Creating a User with Specific Login Shell-

```
# to create a new user named username with /usr/bin/zsh as a login shell type:

sudo useradd -s /usr/bin/zsh username

grep username /etc/passwd
```

Creating a User with Custom Comment-

```
# we are creating a new user named username with text string Test User Account as a comment:

 sudo useradd -c "Test User Account" username

grep username /etc/passwd
```

Creating a User with an Expiry Date-

```
# to create a new user account named username with an expiry time set to January 22 2019 you would run:

sudo useradd -e 2024-01-22 username      # -e (--expiredate)

sudo chage -l username
```

Creating a System User-

```
sudo useradd -r username          # Use the -r (--system) option to create a system user account
```

Changing the Default useradd Values-

```
useradd -D              # The default useradd options can be viewed and changed using the -D, --defaults

#  to change the default login shell from /bin/sh to /bin/bash

sudo useradd -D -s /bin/bash

sudo useradd -D | grep -i shell
```

Set user password-

```
sudo passwd username
```

Adding the user to the Sudo Group-

```
groups user    # to know the group of users

usermod -aG sudo user

sudo usermod -aG sudo user
```

Specifying Explicit User Privileges in /etc/sudoers

```
sudo visudo
```

add the below line to output of sudo visudo -

```
user ALL= (ALL: ALL) ALL
```

Deleting a User-

```
sudo deluser user
```

you want to delete the user's home directory when the user is deleted, you can issue the following command -

```
sudo deluser --remove-home user
```

If you previously configured sudo privileges for the user you deleted, you may want to remove the relevant line again-

```
sudo visudo

newuser ALL=(ALL: ALL) ALL    # DELETE THIS LINE
```

Group related commands-

Creating a Group in Linux –

sudo groupadd [OPTIONS] GROUPNAME   # syntax for the **groupadd** command

```
sudo groupadd mygroup

sudo groupadd -r mysystemgroup          # Use the -r (--system) to create system group
```

# If the **group exist** and to make the command exit successfully, use the -f (--force) option

```
sudo groupadd -f mygroup
```

# Creating a Group **with Specific GID**, # Use the -g (--gid

```
sudo groupadd -g 1010 mygroup
```

Delete a Group-

```
sudo groupdel [OPTIONS] GROUPNAME
sudo groupdel mygroup
sudo groupdel groupname
```

listing all groups-

```
getent group
getent group | grep mygroup    # to get specific group
```

# When used with the -o (--non-unique) option the groupadd command allows you to create a group with **non-unique GID**:

```
groupadd -o -g 1010 mygroup
```

Creating a System Group with Password-

```
groupadd -p grouppassword mygroup
```

How to Add an Existing User to a Group-

two types of groups in Linux operating systems-

The Primary group and Secondary or supplementary group.

```
sudo usermod -a -G groupname username
sudo usermod -a -G group user
sudo usermod -a -G sudo akkc              # to add the user akkc to the sudo group
```

Add an Existing User to Multiple Groups-

```
sudo usermod -a -G group1, group2 username
```

Remove a User from a Group-

```
sudo gpasswd -d username groupname
```

Change a User’s Primary Group-

```
sudo usermod -g groupname username

sudo usermod -g docker akkc
```

File Permission-

The basic Linux permissions model works by associating each system file with an owner and a group and assigning permission access rights for three different classes of users:

The file owner.

The group members.

Others (everybody else).

File ownership, file permissions, user and/or group ownership of a given file, directory, or symbolic link can be **changed using the chown, chmod and chgrp commands**.

Three file permissions types apply to each class of users: -

The read permission.

The write permission.

The execute permission.

To view the file permissions, use the ls command:

ls -l file_name

```
-rw-r--r-- 12 linuxize users 12.0K Apr 28 10:10 file_name
|[-][-][-]-   [------] [---]
|| | ||    |      |
|| | ||    |      +-----------> 7. Group
|| | ||     +------------------> 6. Owner
|| | | +-------------------------> 5. Alternate Access Method
|| |  +----------------------------> 4. Others Permissions
||  +-------------------------------> 3. Group Permissions
```

```
| +---------------------------------> 2. Owner Permissions

+-----------------------------------> 1. File Type
```

Changing File permissions-

Symbolic (Text) Method-

chmod [OPTIONS] [ugoa…][-+=]perms…[,…] FILE..

The first set of flags ([ugoa…]), users flags, defines the users' classes for which the permissions to the file are changed.

u - The file owner.

g - The users who are members of the group.

o - All other users.

a - All users, identical to ugo.

When the users' flag is omitted, it defaults to a.

The second set of flags ([-+=]), the operation flags, defines whether the permissions are to be removed, added, or set:

- - Removes the specified permissions.

+ - Adds specified permissions.

= - Changes the current permissions to the specified permissions. If no permissions are given after the = symbol, all permissions from the specified user class are removed.

The permissions (perms...) are explicitly set using either zero or one or more of the following letters: r, w, x, X, s, and t. Use a single letter from the set u, g, and o when copying permissions from one to another users' class.

When setting permissions for more than one user classes ([,…]), use commas (without spaces) to separate the symbolic modes.

# Give the members of the group permission to execute the file, but not to read and write to it:

chmod g=x filename

# Remove the write permission for all users:

chmod a-w filename

# Remove the read, write, and execute permission for all users except the file's owner:

chmod og-rwx filename

chmod og= filename

# Give read, write and execute permission to the file's owner, read permissions to the file's group, and no permissions to all other users:

chmod u=rwx,g=r,o= filename

Numeric Method-

The syntax of the chmod command when using the symbolic mode

chmod [OPTIONS] NUMBER FILE...

The permission number can consist of three or four digits, ranging from 0 to 7.

When 3 digits number is used, the first digit represents the permissions of the file's owner, the second one the file's group and the last one all other users.

The write, read, and execute permissions have the following number value:

r (read) = 4

w (write) = 2

x (execute) = 1

no permissions = 0

The permissions digit of a specific user class is the sum of the values of the permissions for that class.

Each digit of the permissions number may be a sum of 4, 2, 1 and 0:

0 (0+0+0) – No permission.

1 (0+0+1) – Only execute permission.

2 (0+2+0) – Only write permission.

3 (0+2+1) – Write and execute permissions.

4 (4+0+0) – Only read permission.

5 (4+0+1) – Read and execute permission.

6 (4+2+0) – Read and write permissions.

7 (4+2+1) – Read, write, and execute permission.

For example, if the permission number is set to 750 it means that the file’s owner has read, write and execute permission, file’s group has read and execute permissions, and other users have no permissions:

Owner: rwx=4+2+1=7

Group: r-x=4+0+1=5

Others: r-x=0+0+0=0

When the 4 digits number is used, the first digit has the following meaning:

setuid=4

setgid=2

sticky=1

no changes = 0

The next three digits have the same meaning as when using 3 digits number. If the first digit is 0 it can be omitted, and the mode can be represented with 3 digits. The numeric mode 0755 is the same as 755.

# Give the file’s owner read and write permissions and only read permissions to group members and all other users:

```
chmod 644 dirname

chmod 750 dirname/fileName

chmod 1777 dirname
```

Never Use chmod 777-

Setting 777 permissions to a file or directory means that it will be readable, writable and executable by all users and may pose a huge security risk.

For example, if you recursively change the permissions of all files and subdirectories under the /var/www directory to 777, any user on the system will be able to create, delete or modify files in that directory.

If you experience permission issues with your web server, instead of recursively setting the permission to 777, change the file’s ownership to the user running the application and set the file’s permissions to 644 and directory’s permissions to 755.

Ubuntu/Linux install OpenSSH server-

The procedure to install a ssh server in Ubuntu Linux is as follows:

1. Open the terminal application for Ubuntu desktop.
2. For remote Ubuntu server you must use BMC or KVM or IPMI tool to get console access.
3. Type command:

   ```
   $ sudo apt-get install openssh-server
   ```

4. Enable the ssh service by typing:

   ```
   $ sudo systemctl enable ssh
   ## OR enable and start the ssh service immediately ##
   $ sudo systemctl enable ssh –now
   ```

5. Start the ssh service by typing:

   ```
   $ sudo systemctl start ssh
   ```

6. Verify that ssh service running:

   ```
   $ sudo systemctl status ssh
   ```

7. Configure firewall and open port 22:

   ```
   $ sudo ufw allow ssh

   $ sudo ufw enable

   $ sudo ufw status
   ```

8. Test it by login into the system using:

   ```
   $ ssh userName@Your-server-name-IP
   $ ssh ec2-user@ec2-aws-ip-here
   ```

Installing LAMP on Ubuntu-

Install apache: -

```
sudo apt-get install apache2
```

Install MySql: -

```
sudo apt-get install mysql-server
```

Install PHP: -

```
sudo apt install php libapache2-mod-php
```

```
sudo apt-get install php5 libapache2-mod-php5
```

Restart system: -

```
sudo systemctl restart apache2
```

Check PHP installation: -

```
php -r 'echo "\n\nYour PHP installation is working fine.\n\n\n";'
```

Reset root password: -

Booting to Recovery-

open boot menu for Ubuntu, press the "e" key to open up the grub parameters that are to be edited.

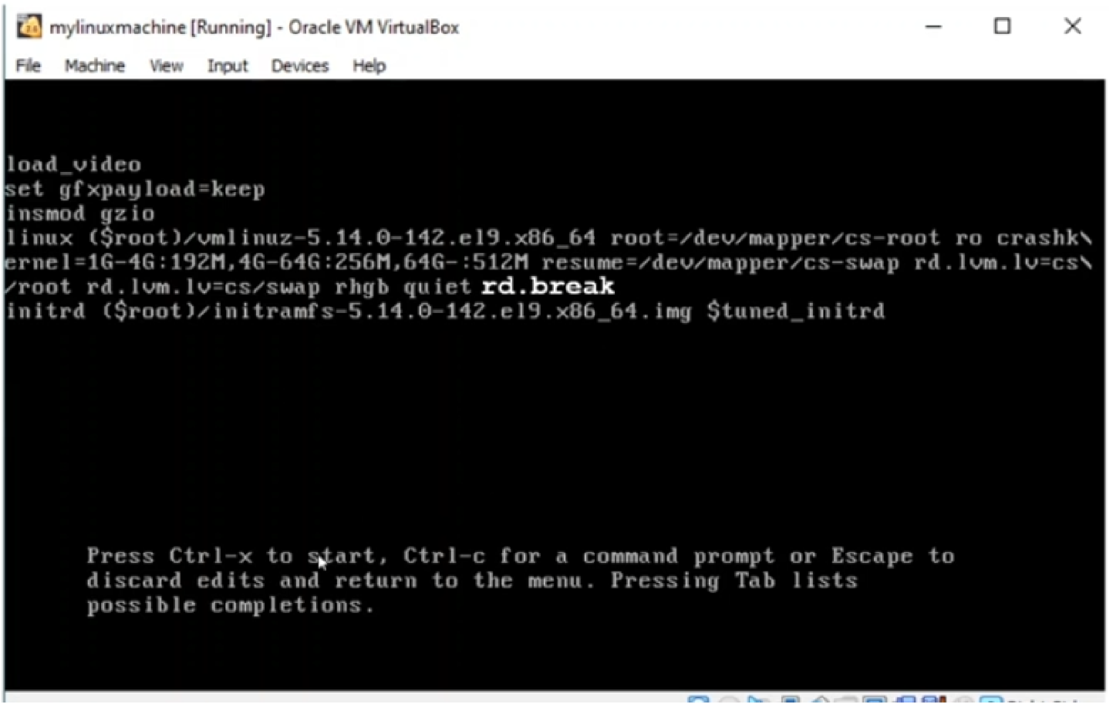
Changing the configurations-

After that use arrow keys again and then scroll down further to the bottom line that begins with "linux /boot/vmlinuz" keyword.

For Linux/Ubuntu/other versions

From the above-highlighted line, you need to replace **ro quiet splash $vt_handoff** word with **rw init=/bin/bash**

CentOS / Red Hat 9



```
load_video
set gfxpayload=keep
insmod gzio
linux ($root)/vmlinuz-5.14.0-142.el9.x86_64 root=/dev/mapper/cs-root ro crashk\
ernel=1G-4G:192M,4G-64G:256M,64G-:512M resume=/dev/mapper/cs-swap rd.lvm.lv=cs\
/root rd.lvm.lv=cs/swap rhgb quiet rd.break
initrd ($root)/initramfs-5.14.0-142.el9.x86_64.img $tuned_initrd




      Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to
      discard edits and return to the menu. Pressing Tab lists
      possible completions.
```

You just have to go toward the end of the line and

simply type rd.break

and then simply do control

now press F10 to save, the system will restart and then you will be landed to your root shell of the system.

Changing the root password-

passwd or passwd root

now type password and confirm password.

reboot the system-

Mount Disk/Directory:-

Commands for disk partition-

df and fdisk

```
df –h                                   // it will show all disk & size.

fdisk –l                                // it shows/list all disk in details.

fdisk –l
```

Now type-

```
 fdisk /dev/sdb   OR   fdisk /dev/sdc
```

# sda/sdb/sdc will be the disk name which you want to mount, choose carefully.

Then⟹ m⟹ n ⟹ p ⟹ 1 ⟹ enter ⟹ enter ⟹ w

Then format the new partition

```
 mkfs –t ext2 /dev/sdb1   OR

mkfs.xfs /dev/sdb1   OR

mkfs.ext3 /dev/sdb1 OR

mkfs -t ext4 /dev/sdb1
```

Create directories that will be mounted-

```
e.g:      mkdir /akkc

          mkdir /Ironman

           mkdir /akkc/jarvis
```

Mount as directory --

```
e.g:      mount /dev/sdb1 /Ironman

          mount /dev/sdb2 /akkc/jarvis

          mount /dev/sdc1 /akkc
```

Mount as Drive --

```
          mount /dev/sdb1  /media/disk

          mount /dev/sdb1  /media/akkc       // akkc will be your drive name
```

Add these entries to /etc/fstab file so the system can mount on boot up

```
 cp /etc/fstab /etc/fstab.bak
```

vi /etc/fstab and add the following lines-

```
/dev/sdb1     /Ironman      ext4    defaults     1     1
/dev/sdb2     /akkc/Jarvis  ext3    defaults     1     1
/dev/sdc1     /akkc         xfs     defaults     0     0
```

For Unmount the Disk-

```
umount /akkc
umount /akkc/jarvis
```

How to install tar.gz file on Linux-

To install tar extraction and compiling tools on Ubuntu, Debian, and Linux Mint:

```
sudo apt update
sudo apt install tar gzip build-essential
```

To install tar extraction and compiling tools on Fedora, CentOS, AlmaLinux, and Red Hat:

```
sudo dnf groups mark install "Development Tools"
sudo dnf groupinstall "Development Tools"
sudo dnf install tar gzip
```

To install tar extraction and compiling tools on Arch Linux and Manjaro:

```
sudo pacman -Sy base-devel tar gzip
```

Start by extracting the contents of your archive-

```
tar xf software-name.tar.gz
. /configure
Make                                    // make command to build and compile the software
```

```
 sudo make install
```

How to Create LVM Partition Step-by-Step in Linux-

Prerequisites

- Raw disk attached to Linux system
- Local User with Sudo rights
- Pre-Installed lvm2 package

Identify new attached raw disk

```
sudo dmesg | grep -i sd      OR
```

```
sudo fdisk -l | grep -i /dev/sd
```

Create PV (Physical Volume)-

Before start creating pv on disk /dev/sdb, make sure lvm2 package is installed. In case it is not installed, then run following command,

```
 sudo apt install lvm2                              // On Ubuntu / Debian

sudo dnf install lvm2                               // on RHEL / CentOS
```

Run following pvcreate command to create pv on disk /dev/sdb,

```
sudo pvcreate /dev/sdb

sudo pvs /dev/sdb                              // To verify pv status

sudo pvdisplay /dev/sdb                        // To verify pv status
```

Create VG (Volume Group)-

```
sudo vgcreate <vg_name>  <pv>

sudo vgcreate volgrp01 /dev/sdb                //result will be"volgrp01" successfully created

 sudo vgs volgrp01                             // verify the status of vg (volgrp01)

sudo vgdisplay volgrp01                        // verify the status of vg (volgrp01)
```

Create LV (Logical Volume)-

```
sudo lvcreate -L <Size-of-LV> -n <LV-Name>   <VG-Name>

sudo lvcreate -L 14G -n lv01 volgrp01              // used to create lv of size 14 GB

sudo lvs /dev/volgrp01/lv01                        // Validate the status of lv

sudo lvdisplay /dev/volgrp01/lv01                  // Validate the status of lv
```

Format LVM Partition-

Run following command to format LVM partition as ext4 file system.

```
sudo mkfs.ext4 /dev/volgrp01/lv01
```

Run following command to format the lvm partition with xfs file system.

```
sudo mkfs.xfs /dev/volgrp01/lv01
```

To use above formatted partition, we must mount it on some folder. So, let's create a folder /mnt/data

```
sudo mkdir /mnt/data
```

Now run mount command to mount it on /mnt/data folder,

```
sudo mount /dev/volgrp01/lv01 /mnt/data/

 df -Th /mnt/data/
```

To mount above lvm partition permanently, add its entries in fstab file using following echo command,

```
echo '/dev/volgrp01/lv01  /mnt/data  ext4  defaults 0 0' | sudo  tee -a /etc/fstab

sudo mount -a
```

Extend the size of volume-

```
vgextend centos /dev/sdb1          //Volume group‘centos’
lvextend -L +2G /dev/centos/root   //Here is how we extend the / partition
                                   by 2GB and /home partition by 6GB
lvextend -L +6G /dev/centos/home


xfs_growfs /dev/centos/root      //increase the partition size of / and /home
                                 using the xfs_growfs command
xfs_growfs /dev/centos/home
df -h
```

Samba Installation-

```
dnf -y install samba   OR

dnf install samba samba-common samba-client

# yum install samba-client

mkdir /samba_share

chmod 777 /samba_share

chown -R  nobody:nobody /samba_share

 chcon -R -t samba_share_t /samba_share

 systemctl enable smb

 systemctl start smb

systemctl enable nmb

systemctl start nmb

useradd samba

passwd samba

smbpasswd -a samba
```

If Firewalld is running, allow Samba service.

```
firewall-cmd --add-service=samba

firewall-cmd --runtime-to-permanent
```

vi /etc/samba/smb.conf

Add this line--

```
[global]
        # line 11 : add (set charset)
        unix charset = UTF-8
        workgroup = SAMBA
        security = user
        # add IP addresses you allow to access
        hosts allow = 127. 10.0.0.
        # add (no authentication)
        map to guest = Bad User


# add to the end
# any Share name you like
[Share]
        # specify shared directory
        path = / samba_share
        # allow writing
        writable = yes
        # allow guest user (nobody)
        guest ok = yes
        # looks all as guest user
        guest only = yes
```

```
    systemctl restart smb
    ifconfig
```