**Linux**

**Operating system**

1. Operating system acts as a intermediate distance between hardware and user.
2. Operating system manages hardware in the system.
3. And also acts as interpreter

**Types of operating system**

* Desktop machine:-

It has only used for general purpose like presentation,video games,etc…

* Server machine:-

Whenever multiple user works on the website at same time.it doesn’t affect the user performance because of the system to be created like this called as server machine.

**Development of operating system**

* Single user single tasking---eg ms-doc
* Single user multitasking
* Multiple user multitasking

**What is hypervisor**

It is a software that creates space and runs virtual machine like (VMS) etc

We use two operating system at a same time with the help of hypervisor …

**Windows**

1. It is proprietary based

(Means licening is required)

1. It is paid
2. It is less secure

(Because it gives (read, write, execute) permission)

1. It is a closed source.

(We cant change source code)

1. It’s a heavy hardware

(To install OS in the hardware you need to give more space)

1. It’s a user friendly GUI
2. It is a non-portable
3. It is a 80% on desktop

**Linux**

1. It is a community based

(means licening is not required)

1. It is free of cost
2. Is is more secure

(it gives only read,write permission)

1. It is open source
2. Light weight software
3. It is portable
4. It is less user-friendly
5. It is 90% on desktop

**What is interpreter**

It is used to translate/converts high level language to low level language called as interpreter.

**What is campilar**

It is used to translate/converts high level language to low level language called as interpreter.

**What is linux**

Linux is **an open source operating system (OS)**. An operating system is the software that directly manages a system's hardware and resources, like CPU, memory, and storage.

The OS sits between applications and hardware and makes the connections between all of your software and the physical resources that do the work.

**History**

* In 1983, american software engineer **Richard Stallman** started project “GNU project” (means GNU is not unix).
* And in 1985, FSF started (free software foundation).
* In 1991 started programme like debugger,shell,editor.
* But GNU is hard to develop then in 1991 Linus torwards develop kernel.

**GNU/LINUX----IINUX Os**

**Operating systems are…**

* **Multics**

Multics, also known as Multiplexed Information and Computing Service, was a time-sharing [operating system](https://www.computerhope.com/jargon/o/os.htm) developed by [MIT](https://www.computerhope.com/jargon/m/mit.htm), General Electric, and [Bell Labs](https://www.computerhope.com/comp/att.htm), first released in [1964](https://www.computerhope.com/history/1964.htm). Notable [developers](https://www.computerhope.com/jargon/d/develope.htm) who worked on Multics include [Ken Thompson](https://www.computerhope.com/people/ken_thompson.htm) and [Dennis Ritchie](https://www.computerhope.com/people/dennis_ritchie.htm).

* **UNIX**

It is a multiuser, multitasking operating system (OS) designed for flexibility and adaptability. Originally developed in the 1970s, Unix was one of the first OSes to be written in the C programming language.

* **BSD**

Berkeley Software Distribution (BSD) is a group of related open source Unix-like operating systems (OS) with origins in early versions of Research Unix at Bell Labs. FreeBSD is the most popular member. BSD is configured for internet hosting, web hosting, and hosting many servers on one system.

* **Posix**

 It is a "Portable Operating System Interface for uni-X", POSIX is a set of standards codified by the IEEE and issued by ANSI and ISO.

* **Minix**

MINIX 3 is a free, open-source, operating system designed to be highly reliable, flexible, and secure. It is based on a tiny microkernel running in kernel mode with the rest of the operating system running as a number of isolated, protected, processes in user mode.

**Architecture of OS**

* Hardware
* Kernel
* Shell
* Application
* User

OS ----depends on kernel and shell

Kernel ----it is core part of OS

Shell ---it gives input from user

**Linux distributions**

* Redhat
* Centos
* Mentos
* Fedora
* ubuntu

**Root user:-**

* It is a super user.
* Home directory of root user is (/root).
* All permission given to a root user

**[root@localhost~]#-----prompt**

**Localhost ---- hostname**

**Root ------super user**

**# ------to login with root**

**/ ------root directory**

**~ ------home directory**

**$ ------to login with local**

**Local user:-**

* It is a local user
* Home directory is (/home/username)
* We can create different localuser in linux

**Important sites to resolve error**

* Stack overflow
* Stack exchange
* Serverfault.com

**To switch user**

* Home directory—root directory

Su – root

* Root directory----home directory

Su – (username)

**File system hierarchy**

* /home—Home directory for home user
* /root ---Root directory for root user
* /boot ---Contain bootable files for linux
* /etc ----All configuration files
* /usr ----By default software are intsall in

This directory

* /bin ----Contains command user by all

User including root user

* /sbin----Contains command use by only

Root user

* /opt -----Optional application software

Package

* /dev -----Any other device attached to

System files printers,tty,etc

* Lib----To store architecture information
* /temp--Temporary files by user,system
* /srv-----to store service information
* /var-----logs,mails,webpages,etc
* /mnt----mount point harddisk
* /media--mnt point reader,floopy disk
* /proc----process information
* /sys------system related information
* /run-----runing information

**Commands**

* hostname------to check hostname
* hostname -i---to check machine address
* Ifconfig -------to check IP address
* which----------to check path
* whoami--------to check who am i
* echo------------to check shell
* grep------------to find out words
* sort-------------to arrange words
* clear------------to clear screen
* echo “$SHELL”----to check shell
* lscpu--------to check cpu information
* lsusb---------to check usb information
* lspci---------peripheral component

interconnect

* free----------to check memory
* free -h ------to check memory in human

readable format

* man----------to see manual page
* info-----------to see information page
* date-----------to check date
* date +%a----to see weekdays
* date +%b----to see month
* date +%c-----to see date and time
* date +%T-----to see time in sec
* date +%t-------to see time
* date -s ---------to change date
* cal---------------to check calender
* cal -y -----------to see year calender
* cal -j ------------to see in julian form
* history----------to see history of command
* tty --------------terminal type show
* who-----to show whos teminal we use and

how many teminals we used

* w------------------load average
* uname-----------to see kernel details
* uname -a--------to see kernel version
* uname -r--------to see kernel version
* shutdown-------to shutdown
* shutdown -c----to cancel shutdown
* shutdown now---to shutdown Immediate
* sudo halt -----to turf off terminal
* sudo reboot -----to reboot the terminal
* dmidecode------to check hardware infor.

to check motherboard.

to check bios informat.

* wc -----------to check words lines characte
* diff ----------to see diff between two files
* df ------------to display diskspace
* alias---------to create custom shortcuts
* pipe---------gives output from first

and input from second

* tee----------show content on terminal as

well as save on file

* tab -------to fill terminal
* ctrl+z -----to stop command forcefully
* ctrl+c -----to stop any command
* **How to change hostname permanently**

hostnamectl set-hostname (name)

* **How to change hostname temporarly**

hostname (name)

* **How to copy file**

cp file /home/mayur/

cp -r file dir1 /home/mayur**/**

* **How to move file**

mv file1 /home/mayur/new1

* **how to rename file**

mv file1 myfiles

**Format of change directory**

**Absolute path**

* cd /home/
* cd /root/
* cd /boot/

**Relative path**

* ./linux/
* ./boot/

**How to check files**

* ls  ----to check list
* ls -l ---to check list with information
* ls -a ----to check hide files
* ls -al -----to check all files include hidden
* ls -ltr ------to check files newly created
* ls -ld ----to check directory
* ll -------to check all files
* pwd ----print working directory
* cd -----change directory

**Command line mode / insert mode**

(vim /etc/passwd )

* G------curser from top to bottom
* gg -----curser from bottom to top
* yy -----copy line where curser is placed
* nyy ---copy number of lines
* P -----paste line where curser is placed
* p -----paste line below curser is placed
* dd------to delete line where cursor is placed
* ndd -----to delete number of lines
* dw ------delete word
* u ------to undo the word
* O -------to create line above the cursor
* o -------to create line below the cursor
* I --------to beginning of the line
* A -------end of the line
* i --------to enter into insert mode
* w ------to save
* q ------to exit
* wq ----to save and quit
* wq! ----to save and quit
* q! ----------forcefully quit
* set nu ---------give numbers to a lines
* set nonu ------remove number
* /words -------to highlight words
* Nohl --------to removes highlight
* V ---------------to enter into visual mode

**Read operation**

1. Cat -----------to view file from bottom to

top

1. Less ----------to view file from top to

bottom

1. More ---------to view file from top to

bottom

1. Head ---------to view first top 10 lines
2. Head -n ------to view numbers of lines

From top

1. Tail -----------to view last 10 lines
2. Tail -n --------to view numbers of lines

bottom

**/etc/passwd**

It stores all the information about users

***mayur:X:1000:1000:hello:/home/mayur:/bin/bash***

mayur--------------username

X ---------------hash form/ encrypted format passwd

1000 --------------user ID

1000 --------------group ID

Hello--------------GECOS

/home/mayur-----home directory

/bin/bash ---------our login shell

**Changes in /etc/passwd file**

1. **To add user with a particular UID**

Useradd -u 1000 user1

1. **To add user with a particular GID**

Useradd -g 1000 user2

1. **To add personal information GECOS**

Useradd -c “hello” user3

1. **to add home directory of user**

useradd -d /mayur user4

1. **to add login shell of user**

useradd -s /sbin/nologin user5

------------------------------------------------------------**/etc/shadow**

It stores all the imformation about the passwords in hashed format /encrypted format

***Mayur:abcdefgh:18009:0:120:7:14:18759***

mayur------------- -username

abcdefgh---------- -hash format password

18009---------------last time passwd change

0 ----------minimum password age

120----------------maximum password age

7 ------------------warning period

14-----------------inactivity period

18759-------------expiration date

**1.to change last time passwd**

* echo $(($(date +%s)/86400))

18816

* date -s “10 july 2022”
* passwd mayur
* echo $(($(date +%s)/86400))

18817

**2.to change minumum passwd days**

* chage -l mayur

minimum no of days between passwd change:0

* chage -m 2 mayur
* chage -l mayur

minimum no of days between passwd change:2

* su – mayur
* passwd

you must wait longer to change your passwd

**3.to change maximum passwd days**

* chage -l mayur

maximum no of days between password change:99999

* chage -M 2 mayur
* chage -l mayur

maximum no of days between password change:2

* su – linux
* su – mayur
* warning:-your passwd will expire in 2 days

**3.to change warning period**

* chage -W 10 mayur

**4.to change inactivity days**

* chage -I 20 mayur

**5.to change expiration date(account expire)**

* chage -E “13 oct 2022” mayur

-------------------------------------------------------------

**/etc/group**

***Tech:X:1021:user1,user2***

Tech------------groupname

X ---------------passwd hash format

1021------------groupID

User1,user2----list of members in group

**1.to add group**

groupadd tech

**2.to add group with GID**

groupadd -g 1021 tech

**3.add no of users in a group**

gpasswd -M user1,user2 tech------add multiple

-------------------------------------------------------------

* **to add one user in a group**

gpasswd -a user1 tech

* **to add multiple user in a group**

gpasswd -M user1,user2 tech

* **to delete user in a group**

gpasswd -d user1 tech

* **to remove passwd of a user/group**

gpasswd -r tech

* **to admin a user**

gpasswd -A user1 tech

* **to remove admin user**

gpasswd -A ‘’ tech

* **to add user with usermod**

usermod -G tech user1

usermod -aG tech user1

**/etc/gshadow**

It stores all information about group in a hashed format/encrypted format

***Tech:X:new:user1,user2***

Tech------------groupname

X ---------------passwd hash format

New------------group administrator

User1,user2----list of members in group

**1.to add group**

groupadd tech

**2.to admin a user**

gpasswd -A new tech

**3.add no of users in a group**

gpasswd -M user1,user2 tech------add multiple

------------------------------------------------------

**How to remove skeleton files and add also**

* cd /home/mayur

rm -rf .bashrc .bash\_logout .bash\_profile

* su – mayur

**[-bash\_4.2]$**

* cd /etc/skel

cp .bashrc .bash\_logout .bash\_profile

/home/mayur

* ls -al /home/mayur
* Logout
* su – mayur

**[mayur@localhost~]$**

**Skeleton files are**

[ .bashrc .bash\_logout .bash\_profile ]

**Usermod/user modify command**

* **To modify login shell of existing user**

usermod -s /sbin/nologin user1

* **To lock the user**

usermod -L mayur

* **to unlock the user**

usermod -U user

* **to change userID of existing user**

usermod -u 1000 user1

* **to change groupID of existing user**

usermod -g 1000 user1

* **to change groupID of existing group**

usermod -g 1000 tech

* **to modify name of existing group**

groupmod -n tech techy

**How to delete users and groups**

* **For user**

rm -rf user1

userdel -r user1

* **for group**

rm -rf tech

groupdel -f tech

----------------------------------------------------

* **how to hide /etc/shadow**

pwunconv

* **how to unhide /etc/shadow**

pwconv

* **how to hide /etc/gshadow**

grpunconv

* **how to unhide /etc/gshadow**

grpconv

**linux file system security**

***drwxrwxrwx 3 root root 8118 jul 1 8:31 file1***

d---------------------directory

rwx------------------owner

rwx-------------------group

rwx-------------------other

3----------------------link count

Root------------------owner

Root------------------group

8118 -----------------file size in byte

Jul 1 8:31------------time and date

File1------------------file name

**1.file type**

1. Normal file--------------(-)
2. Directory----------------(d)
3. Link file-----------------(l)
4. Block device file-------(b)
5. Socket file---------------(s)
6. Character device file---(c)
7. Pipe file------------------(p)

**2.Link count**

File ----------------------1

Directory----------------2

Parent directory--------3

**Hard-link**

* To create a backup file use hard-link
* Its file type is normal file
* Link count increases by 1
* Inode number is same (ls -i)
* Cannot create hard-link of directory

**Soft-link**

* To create a shortcut file use soft-link
* Its file type is link file
* Link count does not increases
* Inode number is different (ls -i)
* Can create directory as well as file

**Metadata**

The data that provide information about other data but not the content of data

**3.Ownership**

* **To change group**

chgrp root new

* **to change owner**

Chown root new

* **Change both owner and group**

Chown root**:**root new

**4.Permissions**

Owner--------rwx

Group--------rwx

Other --------rwx

r----------read----------4

w---------write---------2

x----------execute------1

**In binary form**

rwx ----------111

r-x ----------101

rw- -----------110

r- - -----------100

-w- ------------010

--x ------------001

**Binary**  **In octal form**

000 0\*2^2+0\*2^1+0\*2^0

001 0\*2^2+0\*2^1+1\*2^0

010 0\*2^2+1\*2^1+0\*2^0

111 1\*2^2+1\*2^1+1\*2^0

**5.Default permission**

**For root**

File----------644---{rw- r—r--}

Directory—755—{rwx r-x r-x}

**For local-user**

File---------664-----{rw- rw- r--}

Directory-----775-----{rwx rwx r-x}

Read--------------ls (long listing)

Write-------------create,delete

Execute----------change directory

**6.Full permission**

File------------666----{rw- rw- rw-}

Directory-----777----{rwx rwx rwx}

**7.umask**

umask gives making value of particular default user/permission.

Root---022------🡪default umask

Local—002-----🡪default umask

* **For root user /umask**

Umask=full permission-default permission

=666-644--------------file

=022

=777-755--------------directory

=022

* **For local-user/umask**

Umask=full permission-default permission

=666-664--------------file

=002

=777-775-------------directory

=002

* **To change umask temporary**

Umask (change value)

* **To change umask permanently**

Vim /etc/profile---------------permanent

Source /etc/profile------------update bash

**8.To change permission**

u-------user

g-------group

o-------other

(+) -----------to add permission

(-) ------------to remove permission

(=) -----------to replace permission

**1st method**

1. chmod u+x user1

rwx --- ---

2) chmod u=x user1

--x --- ---

3) Chmod ugo=rwx user1

rwx rwx rwx

**2nd method**

1. chmod 777 user1

rwx rwx rwx

1. chmod 536 user1

r-x -wx rw-

**special permissions**

**1.suid (super user identification permission):-**

it is defined as giving permission to a user to run a file with the permission of file owner.

All users gets access of executable files.

**How to apply SUID**

Chmod u+s “command”

**How to remove SUID**

Chmod u-s “command”

**Applying SUID,,,**

* Chmod u+s /sbin/dmidecode
* ls -l /sbin/dmidecode

rw**s** r-x r-x

**2.Sgid (super user group permission):-**

This permission can inherient the group property

Group of parent directory inherit with the child directory and files.

* **How to apply Sgid**

Chmod g+s “directory/file name”

* **How to remove Sgid**

Chmod g-s “directory/file name”

**1st step:-create a group**

groupadd cloudblitz

passwd cloudblitz

**2nd step:-add member in a group**

gpasswd -M user1,user2 cloudblitz

**3rd step:-to check the group**

tail /etc/group

**4th step:-to create directory**

* mkdir /project
* ls -ld /project

rwx r-x r-x -----default permission

**5th step:-to remove permissions**

chmod 770 /project

**6th step:-to add group cloublitz in**

**/project**

* Chgrp cloudblitz /project
* ls -l /project

rwx rwx rwx 2 root cloudblitz

**7th step:-to add sgid for access the**

**Cloudblitz**

* Chmod g+s /project
* ls -ld /project

drwx rws --- root cloudblitz

**8th step:-to create file in /project**

* su – user1
* cd /project
* touch sample
* ls -l

rw- rw- r-- user1 cloudblitz new

**9th step:-also create file in /project**

* su -user2
* cd /project
* touch sample2
* ls -l

**3.Sticky bit permission:-**

It is basically used for control all other users to resist the delete or modify operation.

Only owners and root user have to access to delete or rename a file.

**How to apply sticky bit permission**

* chmod o+t “directory name”

**How to remove sticky bit**

* chmod o-t “directory name”

**applying sticky bit permission**

* chmod o+t /project
* ls -ld /project

drwx rwx –T root cloudblitz

**T-----execution permission removed**

**t------there is a execution permission**

**4.Acl permission (access**

**Control list)**

It is special type of permission which is act on a directory and add a extra permission to a directory.

Also denies all other users to use these directory/file called ass acl

**How to apply acl for a user**

setfacl -m u:(username):rwx /test

**How to apply acl for a group**

setfacl -m g:(groupname):rwx /test

**How to remove acl**

setfacl -x u:(username):--- /test

setfacl -x g:(groupname):--- /test

**How to remove all acl**

setfacl -b /test

**How to check acl**

getfacl /test

* **applying acl…**

**1st step:-create a user**

useradd user1

passwd user1

**2nd step:-create a directory**

mkdir /test

**3rd step:-check the directory**

ls -ld /test

drwx r-x r-x ----default permission

**4th step:-apply acl for user1**

setfacl -m u:user1:rwx /test

**5th step:-to check acl**

getfacl /test

user:user1:rwx

user :rwx

group:r-x

**6th step:-create a file in /test**

su – user1

cd /test

touch sample1

ls

sample

**To denies all others users permission…**

**1st step:-to denies other user**

Setfacl -m u:user2:--- /test

**2nd step:-to see user2 access or not**

su – user2

cd /test

ls

Permission denied:cannot see any

Directory

**Removing the acl of users from directory**

Setfacl -x u:user1:rwx /test

**Removing all acl from directory**

setfacl -b /test

* **apply acl on group**

**1st step:-create a directory**

groupadd tech

passwd tech

**2nd step:-apply on group**

setfacl -m g:tech:rwx /test

**3rd step:-to check acl**

getfacl /test

user:rwx

group:tech:rwx

other:r-x

**To remove acl on group**

setfacl -x g:tech:rwx /test

**4.sudo (super user do)**

**permission**

it allows you to temporarily elevate your current user account to a root user/previlage.

means we can use root command as a local user

**configuration files**

1.vim /etc/sudoers

2.visudo

**1st technique:-to do permission with password**

**1st step:-vim /etc/sudoers**

* To search /root in the file and add

Username below that with specific

Path

* **Linux ALL=(ALL) /sbin/dmidecode**
* wq!

**To check permission of /etc/sudoers**

ls -ltr /etc/sudoers

r-- --- --- /etc/sudoers

**2nd step:-**

* su -linux
* sudo dmidecode
* Warning generated

(do not use sudo command) only for root

* Type the password
* And run the command in linux user

**3rd step:-**

* su – user1
* sudo dmidecode
* user1 is not in sudoers file .this incident will be recorded

**4th step:-**

* If another user access the sudo file
* these is an error and send a mail to the root user
* to access the sudo file

**5th step:-**

* **Y**ou have a mail in /var/spool/mail/root/

**6th step:-**

* Mail generated in /var/spool/mail/root/
* Vim /var/spool/mail/root/

**7th step:-To check direct mail**

* Mail

**2nd technique:-to do permission without password**

**1st step:-vim /etc/sudoers**

**2nd step:-**

* Type a path below (same thing without

Password)

* **Linux ALL=(ALL) /sbin/dmidecode**
* wq!

**3rd step:-then go to the linux user**

* Su -linux
* Sudo dmidecode
* File will be accessable

**3rd technique:-to do permission with wheel group**

**1st step:-to add user into wheel group**

* gpasswd -a linux wheel

**2nd step:-**

* su – linux
* sudo dmidecode, sudo shutdown,
* all commands get access by only linux user

**4th technique:-permission with wheel group (add command into wheel)**

**1st step:-vim /etc/sudoers**

**2nd step:-**

* Type a path below (wheel group)
* Edit below wheel (all)
* All-----/sbin/dmidecode

**3rd step:-**

* Su -linux
* Sudo dmidecode
* Access the permission

**Filter and search utility**

Filter:-it is a program that take plain text as standard output,transform it into a meaningful format and then return as a standard output.

**Example:-**

1. **To show no of that line**

grep -n root /etc/passwd

1. **To show how many lines**

grep -c root /etc/passwd

1. **For extended words**

grep -E root /etc/passwd

1. **To show either capital or small**

grep -i root /etc/passwd

1. **To find out words only**

grep root /etc/passwd

search:-to search a particular file/directory

**1.locate:-**

can find a file in database system

**updatedb:-**

can update file in database system

--------------------------------------------------------

**2.find:-**

To find out file and related information

**Examples:-**

* Find / -name mayur
* Find / -user mayur -type d

/home/mayur

**If we have to copy all related data of a user**

Find / -user mayur -exec cp-rvf ‘{}’ /backup\:

But the owenership be change to (root/root)

**If we have not change the ownership**

Find / -user mayur -exec cp-aprvf ‘{}’ /backup\:

**Archieving**

**Tar**:-it can be used to create,compressed archieves files and also maintain and modify them

**Examples:-**

* tar -cvf /mnt/etc.tar /etc-------------to create
* ls /mnt-----------------------------------to see file

etc.tar

* du -sh /mnt/etc.tar-------------------to check size

38mb

* tar -xvf /mnt/etc.tar -C /mnt----------to extract
* ls /mnt------------------------------------to check

etc etc.tar

* du -sh /mnt/etc-----------------------to check size

43mb

**There are three methods to compress**

1. gzip (z) ---------------------gz
2. bzip2 (j)---------------------bz2
3. xzip (J)----------------------xz

**1st method :-gzip (z)**

* Tar -czvf /test/etc.tar.gz /etc**---------compress**
* ls /test

etc.tar.gz

* du -sh /test/etc.tar.gz--------------**to check size**

12mb

* Tar -xzvf /test/etc.tar.gz -C /test--------**extract**
* ls /test

etc etc.tar.gz

* du -sh /test/etc

43mb

**2nd method:-bzip2 (j)**

* Tar -cjvf /test/etc.tar.bz2 /etc--------**compress**
* ls /test

etc.tar.bz2

* du -sh /test/etc.tar.bz2---------**to check size**

11mb

* Tar -xjvf /test/etc.tar.bz2 -C /test------**extract**
* Ls /test

etc etc.tar.bz2

* du -sh /test/etc

43mb

**3rd method:-xzip (J)**

* Tar -cJvf /test/etc.tar.xz /etc------**compress**
* Ls /test

etc.tar.xz

* du -sh /test/etc.tar.xz---------------**to check size**

8mb

* Tar -xJvf /test/etc.tar.xz -C /test------**extract**
* ls /test

etc etc.tar.Jz

* du -sh /test/etc

43mb

**------------------------------------------------------------Scheduling**

There are three tool for scheduling

1. At------------------(non periodic task)
2. Cron---------------(periodic task)
3. Anacron-----------(desktop task)

**At command:-**

It is linux-command line utility used to schedule a job for later execution

**Steps:-**

* At “07:23” -----------take particular time

>touch /task1 --------giving task

>mkdir /dir1 ---------giving task

* Ctrl+D ---------------for save and quit
* Atq --------------------to check task
* ls / -------------------to check files

Task1

dir1

**Cron command:-**

It is a linux command line utility used to schedule a job to run in the background automatically.

**Criteria:-**

1. Minute
2. Hour
3. Day of month
4. Month (1-12)
5. Day of week (0-6) (Sunday 7)

Steps:-

* Crontab -e ----------------------command

\* \* \* \* \* /bin/touch file1---create

* wq! -----------------------------save and quit
* systemctl restart crond------- to update
* crontab -l --------------------to running task
* ls / -----------------------------to see files

file1

**tasks:-**

1. 10 sept 20:45
2. At 7.00am on Monday to Friday
3. Perform task only on Saturday
4. Every minute
5. Every 5 minute
6. Every 180 minute
7. Every 90 minute

8.For every 4th Saturday

9.Every 2nd Saturday

10.Every hours

11.For 10min and every 2 hour

--------------------------------------------------------

**Package management**

**Redhat family:-**

1. Low value tool----------rpm
2. High value tool---------yum

**Debian family**

1. Low value tool---------dpkg
2. High value tool---------apt

**Package downloader**

1. wget
2. curl -O

**To check package**

1. rpm –q (package name)
2. yum list installed

**To check all packages**

1. rpm -qa

**To check information of package**

1. rpm -qi (package name)

**To check information of downloaded package**

1. rpm -qip

**How to download package**

1.wget (package name)

2.yumdownloader (package name)

**How to install package**

1.rpm -ivh (package name)

where i-----(to install)

v-----(verbos)

h-----(hashing)

2.yum install (package name)

**How to reinstall packages**

Yum reinstall (package name)

**How to update package**

Yum update (package name)

**How to upgrade package**

Yum upgrade (package name)

**How to unistall packages**

rpm -evh (package name)

Yum remove (package name)

Yum autoremove (package name)

**How to check provide package**

Yum provides (command name)

**How to check (search ) package**

Yum search (package)

**Dependencies:-**

One package depend on the other package

Rpm-----it does not download dependencies

Yum-----it download dependencies

**How to recover unistall command**

Suppose head command unistall and we have to recover these command

1. yum provides head

Coreutils-------package of head command

2. yum reinstall coreutils

3. the package has been reinstalled

4. then run the command

5. head anaconda

**How to create repository**

**1.create a directory**

mkdir /local

**2.then change directory /local**

cd /local

**3.the download the package**

yumdownloader tree

yumdownloader httpd

**4.ls**

httpd 2.4.6.97 el7 x86.64rpm

tree 1.6.0.10 el7 x86.64rpm

**5.to create repository of local**

createrepo /local

**6.to create configuration file**

vim /etc/yum.repos.d/local.repo

1.[abc 123]

1. name=local\_repo
2. baseurl=file:///local
3. enabled=1
4. gpgcheck=0

**7.then save and quit the file**

wq!

**8.Then yum clean all**

**9.Yum repolist all**

**10.yum install httpd**

**How to diasabled repository**

Vim /etc/yum.repos.d/local.repo

1. [abc 123]
2. Name=local\_repo
3. Baseurl=file:///local
4. Enabled=0
5. Gpgcheck=0

**2.Then save and quit the file**

wq!

**3.Yum install httpd**

The package install form update not from

[abc 123]

--------------------------------------------------------

**Networking:-**

It is a interconnected computing device that can exchange data and share resources

**OSI model (open system interconnection)**

It is a seven layer that computer system use to communicate over a network

**Sender side receiver side**

1.Application layer 7.Application layer

2.Presentation layer 6.Presentation layer

3.Session layer 5.Session layer

4.Transport layer 4.Transport layer

5.Network layer 3.Network layer

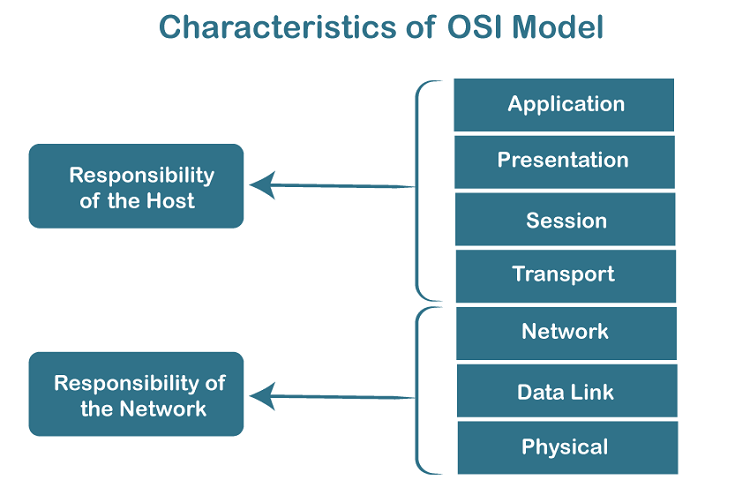
6.Data-link layer 2.Data-link layer

7.Physical layer 1.Physical layer

**1st ------three layers are software layes**

**4th ------is heart layer**

**Next ---three layers are hardware layers**



“All people seems to need data processing”

This is a shortcut sentence to learn seven layers

1.Application layer ------end user layer(ssh)

2.Presentation layer ------syntax layer(ssh,ftp)

3.Session layer ------maintain connection

4.Transport layer -----end to end connectin

5.Network layer -----packets ip/igmp/icmp

6.Data-link layer ------frames switch/ppp

7.Physical layer -------physical structure

**TCP and IP model**

(Tcp—transmission control protocol)

It is a standard that defines how to establish and maintain a network conversation by which application can exchange data. 

**IP (internet protocol)**

It is responsible for delivering packets from the source host to destination host by looking at the IP address in packets header

**UDP (user datagram protocol)**

It is connectionless protocol and used for real-time applications.such as video games, video calls.etc

Example:-video games,video calls

**Mac address**

* MAC address is the physical address, which uniquely identifies each device on a given network. To make communication between two networked devices, we need two addresses: IP address and MAC address. It is assigned to the NIC (Network Interface card) of each device that can be connected to the internet.
* It stands for Media Access Control, and also known as Physical address, hardware address, or BIA (Burned In Address).
* It is globally unique; it means two devices cannot have the same MAC address. It is represented in a hexadecimal format on each device, such as 00:0a:95:9d:67:16.
* It works on the data link layer of the OSI model.

**Why we need mac address**

every mac address is assigned to the [NIC](https://www.javatpoint.com/nic-full-form) of a hardware device that helps to identify a device over a network.

**Computer network types:-**

**1.lan**

**2.wan**

**3.man**

**3.pan**

**1.LAN (Local Area Network)**

* Local Area Network is a group of computers connected to each other in a small area such as building, office.
* LAN is used for connecting two or more personal computers through a communication medium such as twisted pair, coaxial cable, etc.
* It is less costly as it is built with inexpensive hardware such as hubs, network adapters, and ethernet cables.
* The data is transferred at an extremely faster rate in Local Area Network.
* Local Area Network provides higher security.



**2.PAN (Personal Area Network)**

* Personal Area Network is a network arranged within an individual person, typically within a range of 10 meters.
* Personal Area Network is used for connecting the computer devices of personal use is known as Personal Area Network.
* **Thomas Zimmerman** was the first research scientist to bring the idea of the Personal Area Network.
* Personal Area Network covers an area of **30 feet**.
* Personal computer devices that are used to develop the personal area network are the laptop, mobile phones, media player and play stations.



## **3.MAN (Metropolitan Area Network)**

* A metropolitan area network is a network that covers a larger geographic area by interconnecting a different LAN to form a larger network.
* Government agencies use MAN to connect to the citizens and private industries.
* In MAN, various LANs are connected to each other through a telephone exchange line.
* The most widely used protocols in MAN are RS-232, Frame Relay, ATM, ISDN, OC-3, ADSL, etc.
* It has a higher range than Local Area Network(LAN).



## **4.WAN (Wide Area Network)**

* A Wide Area Network is a network that extends over a large geographical area such as states or countries.
* A Wide Area Network is quite bigger network than the LAN.
* A Wide Area Network is not limited to a single location, but it spans over a large geographical area through a telephone line, fibre optic cable or satellite links.
* The internet is one of the biggest WAN in the world.
* A Wide Area Network is widely used in the field of Business, government, and education.



**Ip address:-**

Physical card------—---mac address--------48 bit

Ip4-----version 4----------------ipv6----version 6

Ipv4----------32 bit

Ipv6----------128 bit

8 bit---8 bit---8 bit---8 bit

**Classes:-**

0—----126---(n/8)------N H H H =256\*256\*256

128----191---(N/16)—-N N H H =256\*256

192----223---(N/24)----N N N H =256

224----239-------for special task and multitasking

240----255-------reaserches

**127----------------Is the system IP or loopback IP**

**Class A –**subnet mask------------N H H H /8

Total no of host =2^n

=2^24

=167777216 host

Subnet mask =255 0 0 0

**Class B –**subnet mask-----------N N H H /16

Total no of host =2^n

=2^16

=65,536

Subnet mask =255 255 0 0

-------------------------------------------------------------

**Class C** –subnet mask ----------N N N H /24

Total no of host =2^n

=2^8

=256 host

Subnet mask =255 255 255 0

-----------------------------------------------------------

**Networking to connect IP:-**

1. GUI—------graphical user interface
2. TUI---------terminal user interface
3. CLI---------command line interface

**How to give netmask , host and network**

1. **192.168.0.44/19**

**111111111/11111111/1110000/00000000**

1.Netmask-----255 255 (2^7+2^6+2^5) 0

-----255 255 224 0

2.Network-----2^3=8

3.Host ------2^12-2=4096

**2.172.25.0.5/17**

**11111111/11111111/10000000/00000000**

1.Netmask----255 255 (2^7)

----255 255 128 0

2.Network----2^1=2

3.Host---------2^15-2=3,27,666

**3.10.0.0.5/26**

**11111111/11111111/11111111/11000000**

Netmask--------255 255 255 (2^7+2^6)

1.Netmask ---------255 255 255 192

2.Network----------2^2=4

3.Host---------------2^6-2=62

------------------------------------------------------------

**4.112.25.0.5/17**

**11111111/11111111/10000000/00000000**

1.Netmask------255 255 (2^7)

-------255 255 128 0

2.Network------2^1=1

3.Host-----------2^15-2=32,766

-------------------------------------------------------------

**Private IP range:-**

Class A ------10.0.0.0-----10.255.255.255

Class B-------172.16.0.0----172.31.255.255

Class C-------192.168.0.0---192.168.255.255

**Public IP private IP**

1.internet service decide 1.system admin

IP Decides IP

2.global access charge 2.free of cost charge

3.globally unique 3.locally unique

4.routable on internet 4.no routable on internt

**Add adapter / assign IP with GUI**

1.we have to add adapter

Setting—network—adapter2---internt network---ok

2.start machine

3.ifconfig

4.enp0s8----assign IP for enp0s8

5.go to applications

6.system tools---setting

7.click on adapter enp0s8

8.click on IPV4

9.IPV4------choose manual

10. address ----192.168.0.4

Netmask—255 255 255 0

Getway-----192.1680.1

11.then apply

12.ifconfig

13.ping 192.168.0.4

**Add adapter / assign IP with nmtui**

1.we have to add adapter

Setting—network—adapter3---

Internet network---ok

2.start machine

3.ifconfig

4.enp0s9----assign IP for enp0s9

5.type nmtui

6.edit a connection

7.ethernet---add—ethernet

9.profile name----first

Device name----enp0s9

10.IPV4 confi----automatic to manual show

11.address---192.168.0.6/24

Getway---192.168.0.1

12.okk

13.to activate diactivate connection go to the nmtui again

14.activate connection

15.click on first

16.ping 192.168.0.6

**Add adapter / assign IP with CLI**

1.we have to add adapter

Setting—network—adapter3---internt network---ok

2.start machine

3.ifconfig

4.enp0s10----assign IP for enp0s10

5.on the terminal

6.nmcli connection add con-name demo ifname enp0s10 type ethernet ipv4.addresses “172.25.0.4/24” gw4

“172.25.0.1”

7.nmcli connection show

8.to activate or diactivate the connection

9.ifup enp0s10-----for activate

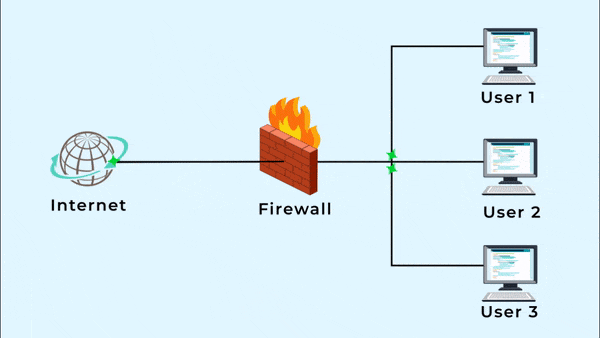
10.ifdown enp0s10---for diactivate

11.ping 172.25.0.4

**Firewall**

A Firewall is a network security device that monitors and filters incoming and outgoing network traffic based on an organization's previously established security policies. At its most basic, a firewall is essentially the barrier that sits between a private internal network and the public Internet.

(It is a additional security layer to maintained inbound traffic.)



Systemctl status firewalld

Systemctl start firewalld

Systemctl enable firewalld

**If your internet doesn’t work,try to change DNS…**

**Configuration file…**

Vim /etc/resolv.conf

Nameserver 8.8.8.8

wq!

Ping 8.8.8.8-----it works

---------------------------------------------------

**Simple way to host webpage**

**In server machine (172.25.0.4)**

**1.you have to install httpd**

Yum install httpd

**2.then configure /create a webpage**

Vim /var/www/html/index.html

<h1>

Hello world

</h1>

Wq!--------------------to save and quit

**3.to start services**

systemctl start httpd

**4.also if you have to enable the service**

systemctl enable httpd

**5.add http service**

* Firewall-cmd --add-service=http

**Or**

* Firewall-cmd --add-service=http --

Permanent

* Firewall-cmd –reload

**6.to check the services**

Firewall-cmd --list-all

**7.To check the webpage**

Curl localhost

Hello world

**8.also check in firefox**

<http://localhost>

Hello world

**In client machine (172.25.0.6)**

9. ping 172.25.0.4

10.curl 172.25.0.4

Hello world

**11.also check in firefox**

http://172.25.0.4

-----------------------------------------------------

**Port forwarding**

In computer networking, port forwarding or port mapping is an application of network address translation that redirects a communication request from one address and port number combination to another while the packets are traversing a network gateway, such as a router or firewall.

Port forwarding is to reduce the chances of hacking also enhance the security that’s why we use port forwarding

**Suppose we have to change the port**

**1.configuration file**

Vim /etc/httpd/conf/httpd.conf

**2.Then change listen**

Listen80------listen8080

**3.then restart the httpd**

systemctl restart httpd

**4.In client machine**

Curl 172.25.0.4

Does not access because we have to change ---the port 80 to 8080

**5.In server machine**

**6.add 8080 Port in firewall**

firewall-cmd --add-port=8080/tcp

**7.to check the services /port**

firewall-cmd --list-all

**8.then in client machine**

Curl 172.25.0.4:8080

Hello world

**9.also check in firefox**

<http://172.25.0.4:8080>

Hello world

**10.To change port without knowing**

**Anyones…**

Firewall-cmd --add-forward- port=port=80:proto=tcp:toport=8080

**11.To check services/port**

Firewall-cmd --list-all

**12.go to client machine**

Curl 172.15.0.4

Hello world

**13.also check in firefox**

<http://172.25.0.4>

--------------------------------------------------------

**IP address and masquerade**

Masquerade NAT allows you to translate multiple IP addresses to another single IP address. You can use masquerade NAT to hide one or more IP addresses on your internal network behind an IP address that you want to make public.

**In centos1 machine (172.25.0.4)**

**1.You have to install httpd**

Yum install httpd

**2.then configure /create a webpage**

Vim /var/www/html/index.html

<h1>

Hello world

</h1>

Wq!

**3.to start services**

systemctl start httpd

**5.add http service**

Firewall-cmd --add-service=http

**6.add masquerade**

firewall-cmd --add-masquerade

**7.then IP forward**

Firewall-cmd –add-forward-port=port=80:proto=tcp:toaddr=172.25.0.6

**Then centos 2 machine (172.25.0.6)**

**1.You have to install httpd**

Yum install httpd

**2.then configure /create a webpage**

Vim /var/www/html/index.html

<h1>

This is new page

</h1>

Wq!

**3.to start the services**

systemctl start httpd

**5.add http service**

Firewall-cmd --add-service=http

**6.add masquerade**

firewall-cmd --add-masquerade

**then in centos 3 machine (172.25.0.8)**

**1.ping the machines first**

Ping 172.25.0.4

Ping 172.25.0.6

**2.curl 172.25.0.4**

This is a new page

**But if we remove the IP forwarding**

Firewall-cmd –remove-forward-port=port=80:proto=tcp:toaddr=172.25.0.6

**In centos 3 machine**

Curl 172.25.0.4

Hello world