Linux Short notes

FOR DEVOPS ENGINEERS

TRAIN WITH SHUBHAM

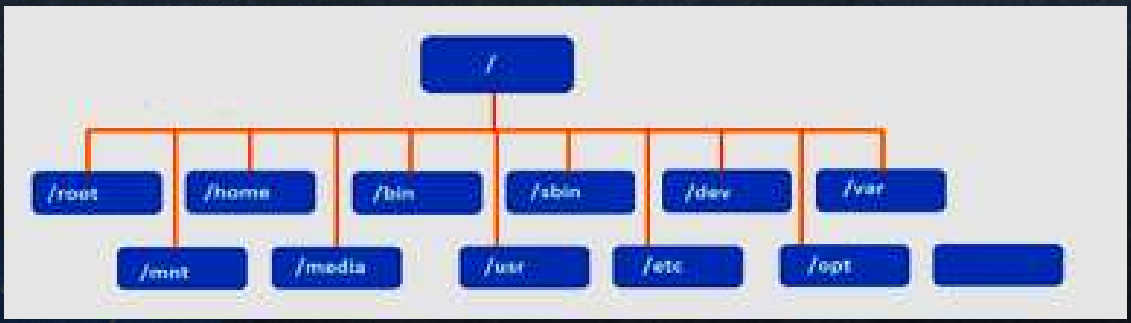
Linux Short Notes

History of LINUX

- Linux came from a Unix family, Linux is a free and open-source software operating system, which Linus Torvalds developed in September 1991.
- In 1991, Linus Torvalds was a student at the University of Helsinki, Finland, USA.
- He developed the first code of Linux 0.01 and post it on the Minix newsgroup on 17 Sep 1991, his code become so popular people encouraged him to develop new code and he was led to develop new code and release the first "official" version of Linux, version 0.02 on October 5, 1991.
- Today many year pass and Linux become one of the most popular operating systems. Today 90% fastest Supercomputers out of 500 run on Linux variants including the top 10.

Linux File System Hierarchy

- In Linux everything is represented as a file including a hardware program, the files are stored in a directory, and every directory contains a file with a tree structure. That is called File System Hierarchy.
- Linux uses single rooted, inverted tree-like structure.
- Root Directory represents with / (forward slash) It is a top-level directory in Linux.



📎 /

- The base of the Linux directory is the root. This is the starting point of FSH. Every directory arises from the root directory. It is represented by a forward slash (/).
- If someone says to look into the slash directory, they refer to the root directory.

📎 /root

- It is the home directory for the root user (superuser).

📎 /bin → User Binaries

- Contains binary executable.
- Common Linux commands you need to use in single-user modes are located under this directory.
- Commands used by all the users of the system are located here.

📎 /sbin → SystemBinaries

- Just like /bin, /sbin also contains binary executables.
- But, the linux commands located under this directory are used typically by system aministrator, for system maintenance purpose.
- For example: iptables, reboot, fdisk, ifconfig, swapon

📎 /dev → Device Files

- it contains hardware device files,
- Contains device files.
- These include terminal devices, usb, or any device attached to the system.
- For example: /dev/tty1, /dev/usbmon0

/var→ Variable Files

- The variable data files such as log files are located in the /var directory.
- File contents that tend to grow are located in this directory.
  ⟩ This includes

- /var/log: System log files generated by OS and other applications.
- /var/lib: Contains database and package files.
- /var/mail: Contains Emails.
- /var/tmp: Contains Temporary files needed for reboot.

/mnt→ Mount Directory

- This directory is used to mount a file system temporarily.

/media→ Removable Media Devices

- The /media directory contains subdirectories where removable media devices inserted into the computer are mounted.

/usr→ User Binaries

- The /usr directory contains applications and files used by users, as opposed to applications and files used by the system.

/etc → Configuration files

- It contains all configuration files of server
- The core configuration files are stored in the /etc directory. It controls the behavior of an operating system or application. This directory also contains startup and shutdown program scripts that are used to start or stop individual programs.

/boot → Boot Loader Files

- The /boot directory contains the files needed to boot the system — for example,
- the GRUB boot loader's files and your Linux kernels are stored here.

/opt → Optional Applications

- The opt directory is used for installing the application software from third-party vendors that are not available in the Linux distribution. Usually, the software code is stored in the opt directory and the binary code is linked to the bin directory so that all users can run that software.

/home → Home Directory

- It contains secondary users home directory.

/tmp → Temporary Files

- Directory that contains temporary files created by system and users.
- Files under this directory are deleted when system is rebooted.

Basic Commands

- #pwd → it shows the present working directory
- #ls → it shows available files and directory list in the present working directory.
- #uname → it shows the name of the kernel (OS)
- #uname -r → it shows version of the kernel
- #cd → it use for change directory
- #clear → it use for clear screen
- #whoami → it show currently login user name
- #history → it show list of previously used commands
- #date → it show time and date

Create file or directory

1. For create single directory

```
#mkdir /shubham
```

2. For create multiple directory

```
#mkdir dev qa test
```

3. For create directory path (directory inside directory)

```
#mkdir -p /dev/qa/test/devops
```

4. For create number of directory

```
#mkdir /student{1..10}
```

TRAIN WITH SHUBHAM

Create file

Touch:

- Touch command is use for create empty file, we can' t write data in a file, can' t edit or save file.

1. Create single file with touch command

```
#touch notes
```

2. Create multilpe file

```
#touch python java react
```

3.Create number of files

```
#touch books{1..10}
```

For copy and paste

CP:

- cp :
- cp command is used for copy and paste file or directory
- Syntax:
- #cp <option> <source> <destination>
- Options
- -r for recursive
- -v for verbose
- -f for forcefully

1. For copy file

```
#cp -rvf /root/anaconda-ks.cfg /home
```

2. For copy all data which start form D alphabet

```
#cp -rvf /root/D* /home
```

For remove file & directory

1. For delete file or directory

```
#rm -rvf /india/pune
```

For move or rename file & directory

1. For move file or directory

```
#mv /home/shub/root/Desktop
```

2. For rename file or directory

```
#mv dev devops
```

User Management

For create user account

```
#useradd shub
```

For check user account properties

```
#grep shub/etc/passwd

shub:1001:1001: :/home/shub:/bin/bash
```

For create user account password

```
#passwd shub
```

For check user password properties

```
#grep shub /etc/shadow

shub:@s$!1bc25f%:18002:0:99999:7: : :
```

For switch user account

```
#su shub
```

For switch user account

```
#su shub
```

For logout from user account

```
#exit
```

Or

```
Press “ctrl+d” Key
```

For Delete user account

```
#userdel shub
```

For change user Login name

```
#usermod -l devops shub
```

Group Management

A group is a collection of user accounts that is very use full to administrators for managing and applying permission to a number of users.

For add Group account

```
#groupadd ibmgrp
```

For check group account property

```
#grep ibmgrp /etc/group
Ibmgrp:x:1001:ajay,vijay,sachin
```

For check group admin property

```
#grep ibmgrp /etc/gshadow
Ibmgrp:!::
```

For Delete group Account

```
#groupdel ibmgrp
```

For add single member in group

```
#gpasswd -a ajay ibmgrp
```

For add multiple member in group

```
#gpasswd -M rahul,virat,rohit ibmgrp
```

For remove group member

```
#gpasswd -d virat ibmgrp
```

For make group admin

```
#gpasswd -A sachin ibmgrp
```

Linux File System Permission

Type of File Permission

- Basic Permission
- Special Permission
- Access Control List (ACL) Permission

For check file permission

```
#ls -l /notes.txt

-rw-r--r--. 1 root root 0 Jan 4 14:59 /notes.txt
```

- Permission
- Link
- Owner
- Group owner
- Size of file
- Date & time of file creation
- Name of file

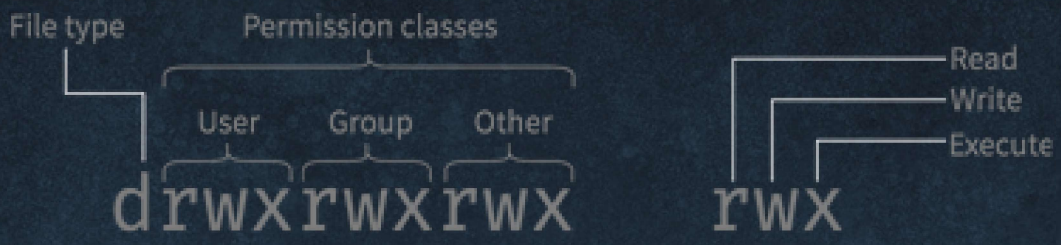
For check directory permission

```
#ls -ld /dev
```

Permission in details

Type of File Permission

- Basic Permission
- Special Permission
- Access Control List (ACL) Permission



Permission Group

Permission Description

- Owner (u) → Permissions used for the owner of the file
- Group (g) → Permissions used by members of the group
- Other (o) → Permissions used by all other users

Permission Set

Permission	Access for a file	Access for a directory
Read (r)	display file contents and copy the file.	View contents of directory
Write (w)	modify the file contents.	modify the contents of a directory.
Execute (x)	execute the file if it is an executable.	Allow use of cd command to access the directory

Permission with numeric & symbol

Number	Permission Type	Symbol
0	No Permission	---
1	Execute	--x
2	Write	-w-
3	Execute + Write	-wx
4	Read	r--
5	Read + Execute	r-x
6	Read + Write	rw-
7	Read + Write + Execute	rwx

For change permission

For add read permission to owner

```
#chmod u+r /notes.txt
```

For add read write permission to group

```
#chmod g+rw /notes.txt
```

For remove read permission to others

```
#chmod o-r /notes.txt
```

For change ownership

Syntax:

```
#chown <user name> <file/directory name>

eg.
#chown ajay /notes.txt
```

For change group ownership

Syntax:

```
#chgrp <group name> <file/directory name>
eg.

#chgrp ibmgrp /notes.txt
```

Set permission with a numeric value

r (read) = 4

w (write) = 2

x (execute) = 1

For set permission with a numeric value

```
#chmod 751 /shub
```

Access Control List (ACL)

- Access control list (ACL) provides an additional, more flexible permission mechanism for file systems.

- Access control list is a service which is used for providing special permission to specific users and group to particular directories and file

Use of ACL

Think of a scenario in which a particular user is not a member of group created by you but still you want to give some read or write access, how can you do it without making user a member of group, here comes in picture Access Control Lists, ACL helps us to do this trick.

For check ACL Permission

Syntax:

```
#getfacl <name of file or directory>

eg. #getfacl /devops
```

TRAIN WITH SHUBHAM

For set ACL permission to user

```
#setfacl -m u:shub:rwx /de
```

For remove ACL permission of user

```
#setfacl -x u:shub: /devops
```

For set ACL permission to Group

```
#setfacl -m g:shubgrp:rwx /devops
```

For remove ACL permission of group

```
#setfacl -x g:shubgrp: /devops
```

For remove all ACL permissions

```
#setfacl -b /devops
```

Access Control List (ACL)

Regular Expressions

What are Regular Expressions?

- Regular expressions are special characters which help search data, matching complex patterns.

GREP (Global Regular Expression Print)

- The grep filter searches a file for a particular pattern of characters, and displays all lines that contain that pattern.

Search a word (string in a file)

```
#grep root /etc/passwd
```

Search a string in multiple files

```
#grep root /etc/passwd /etc/group
```

Search a string insensitive in file

```
#grep -i Root /etc/passwd
```

Search a string in all files recursively

```
#grep -r root /
```

Inverting the string match

```
#grep -v root /etc/passwd
```

Displaying the string match total line no

```
#grep -c root /etc/passwd
```

Display the file names that match the string

```
#grep -l root /etc/passwd/etc/shadow
```

Display the file names that do not contain the string

```
#grep -L root /etc/passwd /etc/shadow
```

Displaying the string match line with number

```
#grep -n root /etc/passwd
```

Display the lines that start with a string

```
#grep ^root /etc/passwd
```

Display the lines that end with a string

```
#grep /bin/bash$ /etc/passwd
```

Search and redirect output in a new file

```
#grep root /etc/passwd > /mnt/find.txt
```

Find

The Linux Find Command is one of the most important and much used command in Linux systems.

Find command used to search and locate the list of files and directories based on conditions you specify for files that match the arguments.

Find can be used in a variety of conditions like you can find files by permissions, users, groups, file type, date, size, and other possible criteria

Find files under Home directory

```
#find /home -name shub.txt
```

Find files with suid permission

```
#find / -perm 4755
```

Find files with guid permission

```
#find / -perm 2644
```

Find files with sticky bit permission

```
#find / -perm 1755
```

Using Find command based on users

```
#find / -user root
```

Using Find command based on groups

```
#find / -group shubgrp
```

Search the file with less than 10MB in a folder

```
#find /tmp -size -10M
```

Search the file with more than 10MB in a folder

```
#find /tmp -size +10M
```

WC (Word Count)

The wc command is use for the count word and line numbers.

Count number of lines

```
#wc -l /etc/passwd
```

Count number of words

```
#wc -w /etc/passwd
```

head

Head command is used for to display top line of the file.

Display top 10 line of the file

```
#head /etc/passwd
```

Display top specific no line of the file

```
#head -n 15 /etc/passwd
```

tail

Tail command is used for to display the bottom line of the file.

Display bottom 10 line of the file

```
#tail /etc/passwd
```

Display bottom specific line of the file

```
#tail -n 5 /etc/passwd
```

Archive File in Linux

Archiving is the process of combining multiple files and directories (same or different sizes) into one file. Archive process is very useful for backup and compression size of data in Linux.

What is Tar

- The Linux "tar" stands for tape archive, which is used by large number of Linux/Unix system administrators to compress size or drives backup. For create archive tar used some compression algorithms Such as gzip,bz2 and xz

Tar command syntax

#tar <options> <files>

Commonly used options

c -for create

x -for extract

v -for verbose

f -for forcefully

t -for test

z -for gzip

j -for bz2

J -for xz

C -for specific destination

To create a tar archive file

```
# tar -cvf /mnt/backup.tar /var
```

To show file size in human-readable format

```
#du -sh /var
#du -sh /mnt/backup.tar
```

To extract a tar archive file on the default location

```
#tar -xvf /mnt/backup.tar
```

To extract a tar archive file on the specific location

```
#tar -xvf /mnt/backup.tar -C /root/Desktop/
```

To create a tar archive file with compress in size (gzip)

```
# tar -cvzf /mnt/backup.tar.gz /var
```

To extract a tar archive file with compress in size (gzip)

```
#tar -xvzf /mnt/backup.tar.gz
```

To create a tar archive file with compress in size (bzip2/bz2)

```
# tar -cvjf /mnt/backup.tar.bz2 /var
```

To extract a tar archive file with compress in size (bzip2/bz2)

```
#tar -xvjf /mnt/backup.tar.bz2
```

To create a tar archive file with compress in size (xz)

```
#tar -cvJf /mnt/backup.tar.xz /var
```

To extract a tar archive file with compress in size (xz)

```
#tar -xvJf /mnt/backup.tar.xz
```

Job Automation

- Job automation allows us to perform tasks automatically in OS by using tools.
- This feature is very useful for the administrator to assign tasks to OS whenever he is not present or performing daily basis work.

Two type of job automation

1. at—command is used to execute a job only one time.
2. crontab—Crontab command is used to execute a job multiple times.

To a set job with at command

```
#date
#at 8:10 AM
at>useradd shub
at>
Ctrl+d (write & quit)
```

To show pending a job

```
#atq
```

To remove a job

```
#atrm 2
```

To restrict user from accessing at

```
#vim /etc/at.deny
Shub (add here user name)
:wq (write&quit)
```

To start crond service

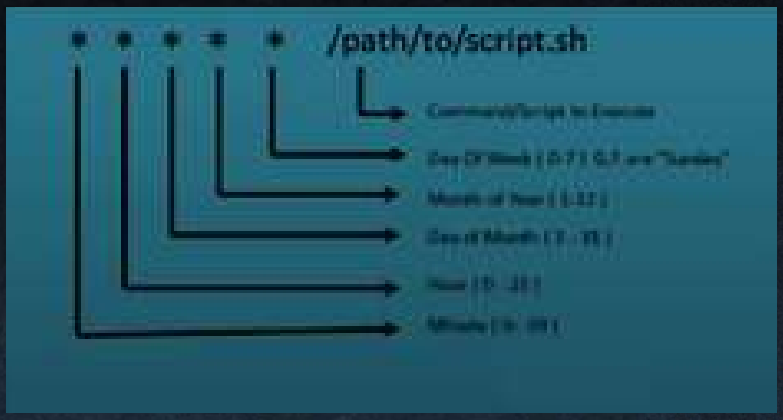
```
#systemctl start crond
```

To enable crond service (Permanent on)

```
#systemctl enable crond
```

For set cron jobs

```
#crontab -e
```



To show cron jobs of the current user

```
#crontab -l
```

To remove cron jobs

```
#crontab -r
```

Or,

Go to the crontab file and remove the job line

```
#crontab -e
```

To the set cron job to other users

```
#crontab -u shub -e
```

To show cron job, other user

```
#crontab -u shub -l
```

To restrict user from crond service

```
#vim /etc/cron.deny
```

Sudo Command

What is sudo?

- sudo (“superuser do” , or “switch user do”) allows a user with proper permissions to execute a command as another user, such as the superuser.

- sudo allows a permitted user to execute a command as another user, according to specifications in the /etc/sudoers file.

Provide sudo privilege to the user

- **For edit configuration file:**

 # vim /etc/sudoers

root ALL=(ALL) ALL

amir ALL=(ALL) ALL (add this line appro. 101 lines)

:wq

Provide sudo privilege to the group

For edit configuration file:

vim /etc/sudoers

%punegrp ALL=(ALL) ALL (line number 108)

:wq

By default, all members of punegrp group got sudo privileges

Wheel group

Wheel is a system group that by default has sudo privileges,
if we add any member to that group then that user got sudo privilege

#grep wheel /etc/group

#useradd shub

#passwd shub

#gpasswd -a shub wheel

By default, all members of the wheel group got sudo privileges

Sudo without password

For edit configuration file:

```
# vim /etc/sudoers

amir ALL=(ALL) NOPASSWD: ALL
%punegrp ALL=(ALL) NOPASSWD: ALL
:wq
```

Managing networking based on Red Hat Enterprise Linux

To show ip address

```
#ifconfig
```

Or,

```
#ip addr
```

Configure networking with nmcli

Network Manager is a daemon that monitors and manages network settings.

nmcli command used to manage networking

Manage IP configuration

For Show all list of connection

```
#nmcli con show
```

Manage IP configuration

To Show all list of connection

```
#nmcli con show
```

To show a active connection

```
#nmcli con show --active
```

To show a specific connection

```
#nmcli con show “citynet”
```

To show the device status

```
#nmcli dev status
```

To create a new connection with nmcli

```
#nmcli con add con-name “citynet” ifname ens33 type ethernet
ipv4.add 192.168.0.2/24 gw4 192.168.0.1 ip4.dns 192.168.0.1
connection.autoconnect yes ipv4.method manual
```

To show the device status

```
#nmcli con mod “citynet” ipv4.add 192.168.0.100
#nmcli con mod “citynet” gw4 192.168.0.254
#nmcli con mod “citynet” ipv4.dns 192.168.0.254
#nmcli con down citynet
#nmcli con up citynet
#nmcli con show—active
#nmcli con show citynet
```

To activate new connection

```
#nmcli con up "citynet"
#nmcli con show–active
```

To deactivate connection

```
#nmcli con down citynet
```

To start a new connection and stop the old connection

```
#nmcli connection modify “citynet” connection.autoconnect no
#nmcli connection modify “pune” connection.autoconnect yes
```

To remove existing connection

```
#nmcli con delete citynet
```

To remove the existing connection

```
#nmcli con delete citynet
```

To set hostname

```
#hostnamectl set-hostname it.citynet.com
```

To show hostname

```
#hostname
```

Configuring networking with nmtui

```
#nmtui
```

IP address configuration files

All created connections with nmcli and nmtui by default are stored in the following file

```
#cd /etc/NetworkManager/system-connections/
#ls
```

Note

We can also modify the connection using the above connection file, but it not recommended.

If update the file, restart NetworkManager service to update IP-configuration

```
#systemctl restart NetworkManager
```

Thank You Dosto

TRAIN WITH SHUBHAM