

1. Linux Introduction

Operating System is a program that mediates between the user and the computer hardware.

Linux is a family of Open source Unix like OS based ON Linux kernel

Distributions include the Linux kernel and supporting system software's and libraries

Unix variants from vendors including University of California, Berkeley (BSD), Microsoft (Xenix), Sun Microsystems (SunOS/Solaris), HP/HPE (HP-UX), and IBM (AIX).

Linux Origins

- 1984: The GNU Project and the Free Software Foundation
 - Creates open source version of UNIX utilities
 - Creates the General Public License (GPL)
 - Software license enforcing open source principles
- 1991: Linus Torvalds
 - Creates open source, UNIX-like kernel, released under the GPL
 - Ports some GNU utilities, solicits assistance online
- Today:
 - Linux kernel + GNU utilities = complete, open source, UNIX-like operating system
 - Packaged for targeted audiences as *distributions*

What is Open Source?

- Open source: software and source code available to all
- The Free Software Foundation specifies four freedoms
 - The freedom to run the program for any purpose.
 - The freedom to study and modify the source code
 - The freedom to redistribute the program
 - The freedom to create derivative programs
- Many open-source licenses exist, each with different particulars

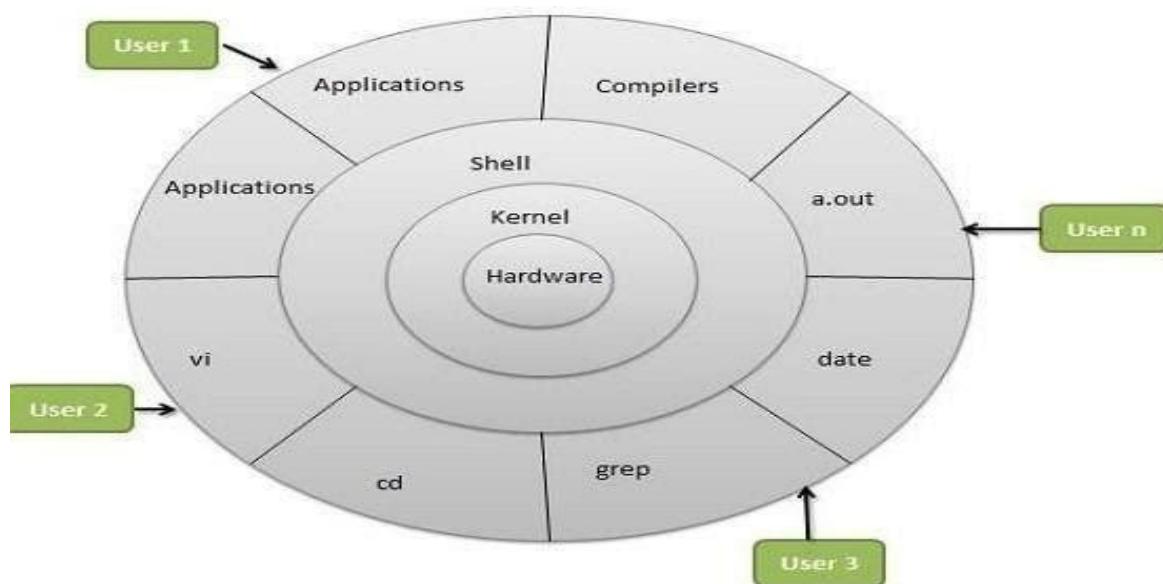
Linux principles

- Everything is a file (including hardware)
- Small, single-purpose programs
- Ability to chain programs together to perform complex tasks
- Avoid captive user interfaces
- Configuration data stored in text

Why Linux?

- OpenSource.
- Community support.
- Heavily customizable.
- Most Servers runs on Linux.
- DevOps most of the tools implements on Linux only.
- Automation
- Secure.

Architecture of Linux



Kernel (It's software residing in memory that tells the CPU what to do.)

The Linux kernel is the main component of a Linux operating system (OS) and is the core interface between a computer's hardware and its processes

What the kernel does

The kernel has 4 jobs:

Memory management:

Keep track of how much memory is used to store what, and where

Process management:

Determine which processes can use the central processing unit (CPU), when, and for how long

Device drivers: Act as mediator/interpreter between the hardware and processes

System calls and security: Receive requests for service from the processes

Some Important Directories

- Home Directories: /root, /home/username
- User Executable: /bin, /usr/bin, /usr/local/bin
- System Executables: /sbin, /usr/sbin, /usr/local/sbin
- Other Mountpoints: /media, /mnt
- Configuration: /etc
- Temporary Files: /tmp
- Kernels and Bootloader: /boot
- Server Data: /var, /srv
- System Information: /proc, /sys
- Shared Libraries: /lib, /usr/lib, /usr/local/lib

Diffrent Linux distros

→

→ Popular Desktop Linux OS

- Ubuntu Linux
- Linux Mint
- Arch Linux
- Fedora
- Debian
- OpenSuse
- Kali linux

→ Popular Server Linux OS

- Red Hat Enterprise Linux --- > RHEL
- Ubuntu Server --> debain
- Centos -->
- SUSE Enterprise Linux

Most used Linux distros currently in IT industry.

RPM based:- RHEL & Centos

Debian based :- Ubuntu Server

Difference between RPM based and Debian based.

From user's point of view, there isn't much difference in these tools. The RPM and DEB formats are both just archive files, with some metadata attached to them. They are both equally arcane, have hardcoded install paths and only differ in subtle details. DEB files are installation files for Debian based distributions. RPM files are installation files for Red Hat based distributions. Ubuntu is based on Debian's package manager based on APT and DPKG. Red Hat, CentOS and Fedora are based on the old Red Hat Linux package management system, RPM.

DEB or .deb (Debian based softwares)

DEB is the extension of the Debian software package format and the most often used name for such binary packages. DEB was developed by Bedian.

Example: Google chrome software

Package name: google-chrome-stable_current_amd64.**deb**

Installation: dpkg -i google-chrome-stable_current_amd64.**deb**

RPM or .rpm (Red Hat based softwares.)

It is a package management system. The name RPM variously refers to the .rpm file format, files in this format, software packaged in such files, and the package manager itself. RPM was intended primarily for Linux distributions; the file format is the baseline package format of the Linux Standard Base. RPM was developed by Community & **Red Hat**.

Example: Google chrome software

Package Name: google-chrome-stable-57.0.2987.133-1.x86_64.rpm

Installation: rpm -ivh google-chrome-stable-57.0.2987.133-1.x86_64.rpm

NOTE: You will also encounter different commands, packages and service names while using both kinds of distros.

2.

Basic Commands

→ Open Terminal

File commands

1. ls Directory listing
2. ls -al Formatted listing with hidden files
3. ls -lt Sorting the Formatted listing by time modification
4. cd dir Change directory to dir
5. cd Change to home directory
6. pwd Show current working directory
7. mkdir dir Creating a directory dir
8. cat >file Places the standard input into the file
9. more file Output the contents of the file
10. head file Output the first 10 lines of the file
11. tail file Output the last 10 lines of the file
12. tail -f file Output the contents of file as it grows, starting with the last 10 lines
13. touch file Create or update file
14. rm file Deleting the file
15. rm -r dir Deleting the directory
16. rm -f file Force to remove the file
17. rm -rf dir Force to remove the directory dir
18. cp file1 file2 Copy the contents of file1 to file2
19. cp -r dir1 dir2 Copy dir1 to dir2; create dir2 if not present
20. mv file1 file2 Rename or move file1 to file2, if file2 is an existing

Process management

1. ps To display the currently working processes
2. top Display all running process
3. Unix/Linux Command Reference
3. kill pid Kill the process with given pid
4. killall proc Kill all the process named proc
5. pkill pattern Will kill all processes matching the pattern
6. bg List stopped or background jobs, resume a stopped job in the background
7. fg Brings the most recent job to foreground
8. fg n Brings job n to the foreground

File permission

1. chmod octal file Change the permission of file to octal, which can be found separately for user, group, world by adding,

- 4-read(r)
- 2-write(w)
- 1-execute(x)

Searching

1. grep pattern file Search for pattern in file

2. grep -r pattern dir Search recursively for pattern in dir

3. command | grep

Pattern

Search pattern in the output of a command

4. locate file Find all instances of file

5. find . -name filename Searches in the current directory (represented by

a period) and below it, for files and directories with names starting with filename

6. pgrep pattern Searches for all the named processes , that matches with the pattern and, by default, returns their ID

File permission

1. chmod octal file Change the permission of file to octal, which can be found separately for user, group, world by adding,

- 4-read(r)
- 2-write(w)
- 1-execute(x)

Searching

1. grep pattern file Search for pattern in file

2. grep -r pattern dir Search recursively for pattern in dir

3. command | grep

pattern

Search pattern in the output of a command

4. locate file Find all instances of file

5. find . -name filename Searches in the current directory

(represented by a period) and below it, for files and directories with names starting with filename

6. pgrep pattern Searches for all the named processes , that matches with the pattern and, by default, returns their ID

System Info

1. date Show the current date and time

2. cal Show this month's calender

3. uptime Show current uptime

4. w Display who is on line

5. whoami Who you are logged in as

Unix/Linux Command Reference

6. finger user Display information about user

7. uname -a Show kernel information

8. cat /proc/cpuinfo Cpu information

9. cat proc/meminfo Memory information

10. man command Show the manual for command

11. df Show the disk usage

12. du Show directory space usage

13. free Show memory and swap usage

14. whereis app Show possible locations of app

15. which app Show which applications will be run by default

Compression

1. tar cf file.tar file Create tar named file.tar containing file

2. tar xf file.tar Extract the files from file.tar

3. tar czf file.tar.gz files Create a tar with Gzip compression

4. tar xzf file.tar.gz Extract a tar using Gzip

5. tar cjf file.tar.bz2 Create tar with Bzip2 compression

6. tar xjf file.tar.bz2 Extract a tar using Bzip2

7. gzip file Compresses file and renames it to file.gz

8. gzip -d file.gz Decompresses file.gz back to file

Network

1. ping host Ping host and output result

3. dig domain Get DNS information for domain

4. dig -x host Reverse lookup host

5. wget file Download file

6. wget -c file Continue a stopped downloadile permission

1. chmod octal file Change the permission of file to octal,which can

be found separately for user,group,world by adding,

- 4-read(r)
- 2-write(w)
- 1-execute(x)

Searching

1. grep pattern file Search for pattern in file
2. grep -r pattern dir Search recursively for pattern in dir
3. command | grep

pattern

Search pattern in the output of a command

4. locate file Find all instances of file
5. find . -name filename Searches in the current directory (represented by a period) and below it, for files and directories with names starting with filename
6. pgrep pattern Searches for all the named processes , that matches with the pattern and, by default, returns their ID

System Info

1. date Show the current date and time
2. cal Show this month's calender
3. uptime Show current uptime
4. w Display who is on line
5. whoami Who you are logged in as
6. finger user Display information about user
7. uname -a Show kernel information
8. cat /proc/cpuinfo Cpu information
9. cat proc/meminfo Memory information
10. man command Show the manual for command
11. df Show the disk usage
12. du Show directory space usage
13. free Show memory and swap usage
14. whereis app Show possible locations of app
15. which app Show which applications will be run by default

Compression

1. tar cf file.tar file Create tar named file.tar containing file
2. tar xf file.tar Extract the files from file.tar
3. tar czf file.tar.gz files Create a tar with Gzip compression

4. tar xzf file.tar.gz Extract a tar using Gzip
5. tar cfj file.tar.bz2 Create tar with Bzip2 compression
6. tar xjf file.tar.bz2 Extract a tar using Bzip2
7. gzip file Compresses file and renames it to file.gz
8. gzip -d file.gz Decompresses file.gz back to file

Network

1. ping host Ping host and output result
 3. dig domain Get DNS information for domain
 4. dig -x host Reverse lookup host
 5. wget file Download file
 6. wget -c file Continue a stopped **download**
 1. date Show the current date and time
 2. cal Show this month's calendar
 3. uptime Show current uptime
 4. w Display who is on line
 5. whoami Who you are logged in as
- Unix/Linux Command Reference
6. finger user Display information about user
 7. uname -a Show kernel information
 8. cat /proc/cpuinfo Cpu information
 9. cat proc/meminfo Memory information
 10. man command Show the manual for command
 11. df Show the disk usage
 12. du Show directory space usage
 13. free Show memory and swap usage
 14. whereis app Show possible locations of app
 15. which app Show which applications will be run by default

Compression

1. tar cf file.tar file Create tar named file.tar containing file
2. tar xf file.tar Extract the files from file.tar
3. tar czf file.tar.gz files Create a tar with Gzip compression
4. tar xzf file.tar.gz Extract a tar using Gzip
5. tar cfj file.tar.bz2 Create tar with Bzip2 compression
6. tar xjf file.tar.bz2 Extract a tar using Bzip2
7. gzip file Compresses file and renames it to file.gz
8. gzip -d file.gz Decompresses file.gz back to file

Network

1. ping host Ping host and output result
3. dig domain Get DNS information for domain

4. dig -x host Reverse lookup host
5. wget file Download file
6. wget -c file Continue a stopped download

→ Know where you are? Present Working Directory

```
imran@DevOps:~$ pwd
/home/imran
imran@DevOps:~$ █
```

→ Create a directory/folder in your home directory.

```
imran@DevOps:~$ mkdir linux-practices
imran@DevOps:~$ █
```

→ Change your current working directory to linux-practices(Go to linux-practices folder).

```
imran@DevOps:~$ cd linux-practices/
imran@DevOps:~/linux-practices$ █
```

→ Create some more directories and list them with “ls” command.

```
imran@DevOps:~/linux-practices$ mkdir vmdir
imran@DevOps:~/linux-practices$ mkdir testdir
imran@DevOps:~/linux-practices$ mkdir devopsdir
imran@DevOps:~/linux-practices$ ls
devopsdir testdir vmdir
```

→ Create some empty files with “touch” command and list them.

```
imran@DevOps:~/linux-practices$ touch file2 file3 file4
imran@DevOps:~/linux-practices$ ls
devopsdir file1 file2 file3 file4 testdir vmdir
```

→ Reconfirm your location in your system.

```
imran@DevOps:~/linux-practices$ pwd  
/home/imran/linux-practices  
imran@DevOps:~/linux-practices$ ls  
devopsdir file1 file2 file3 file4 testdir vmdir
```

Absolute path and Relative path

What is a path?

A path is a unique location to a file or a folder in a file system of an OS. A path to a file is a combination of / and alpha-numeric characters.

What is an absolute path?

An absolute path is defined as specifying the location of a file or directory from the root directory(/). In other words we can say absolute path is a complete path from start of actual filesystem from / directory.

Some examples of absolute path:

/home/imran/linux-practices/

/var/ftp/pub

/etc/samba.smb.conf

/boot/grub/grub.conf

If you see all these paths started from / directory which is a root directory for every Linux/Unix machines.

What is the relative path?

Relative path is defined as path related to the present working directory(pwd). Suppose I am located in /home/imran and I want to change directory to /home/imran/linux-practices. I can use relative path concept to change directory to linux-practices and also devopsdir directory.

```
File Edit View Search Terminal Help  
imran@DevOps:~$ pwd  
/home/imran  
imran@DevOps:~$ cd linux-practices/  
imran@DevOps:~/linux-practices$ ls  
devopsdir file1 file2 file3 file4 testdir vmdir  
imran@DevOps:~/linux-practices$ pwd  
/home/imran/linux-practices  
imran@DevOps:~/linux-practices$ cd devopsdir/  
imran@DevOps:~/.../devopsdir$ pwd  
/home/imran/linux-practices/devopsdir  
imran@DevOps:~/.../devopsdir$ █
```

If you see all these paths did not start with / directory.

→ Creating directories in devopsdir directory with absolute and relative path.

```
File Edit View Search Terminal Help
imran@DevOps:~/linux-practices$ ls
devopsdir file1 file2 file3 file4 testdir vmdir
imran@DevOps:~/linux-practices$ mkdir devopsdir/ansible
imran@DevOps:~/linux-practices$ mkdir /home/imran/linux-practices/devopsdir/aws
imran@DevOps:~/linux-practices$ ls devopsdir/
ansible aws
imran@DevOps:~/linux-practices$ █
```

→ Copying files into directory.

```
File Edit View Search Terminal Help
imran@DevOps:~/linux-practices$ pwd
/home/imran/linux-practices
imran@DevOps:~/linux-practices$ ls
devopsdir file1 file2 file3 file4 testdir vmdir
imran@DevOps:~/linux-practices$ cp file1 testdir/
imran@DevOps:~/linux-practices$ cd testdir/
imran@DevOps:~/.../testdir$ ls
file1
imran@DevOps:~/.../testdir$ █
```

→ Copying directories from one location to another.

```
File Edit View Search Terminal Help
imran@DevOps:~/linux-practices$ cd
imran@DevOps:~$ cd -
/home/imran/linux-practices
imran@DevOps:~/linux-practices$ pwd
/home/imran/linux-practices
imran@DevOps:~/linux-practices$ ls
devopsdir file1 file2 file3 file4 testdir vmdir
imran@DevOps:~/linux-practices$ cp -rvfp testdir/ vmdir/
'testdir/' -> 'vmdir/testdir'
'testdir/file1' -> 'vmdir/testdir/file1'
imran@DevOps:~/linux-practices$ ls vmdir/
testdir
imran@DevOps:~/linux-practices$ █
```

→ Moving files from one location to another.

```
imran@DevOps:~/linux-practices$ pwd
/home/imran/linux-practices
imran@DevOps:~/linux-practices$ ls
devopsdir file1 file2 file3 file4 testdir vmdir
imran@DevOps:~/linux-practices$ mv devopsdir/ vmdir/
imran@DevOps:~/linux-practices$ ls
file1 file2 file3 file4 testdir vmdir
imran@DevOps:~/linux-practices$ ls vmdir/
devopsdir testdir
imran@DevOps:~/linux-practices$ mv file3 file4 vmdir/
imran@DevOps:~/linux-practices$ ls
file1 file2 testdir vmdir
```

→ Removing files and directories.

```
imran@DevOps:~/linux-practices$ rm file1
imran@DevOps:~/linux-practices$ ls
file2 testdir vmdir
imran@DevOps:~/linux-practices$ rm -rf testdir/
imran@DevOps:~/linux-practices$ ls
file2 vmdir
```

VIM EDITOR

→ Install vim editor.

vim

vi

nano

touch fileame

vi filename

```
imran@DevOps:~/linux-practices$ sudo apt-get install vim
[sudo] password for imran:
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

→ Open up a file in vim editor

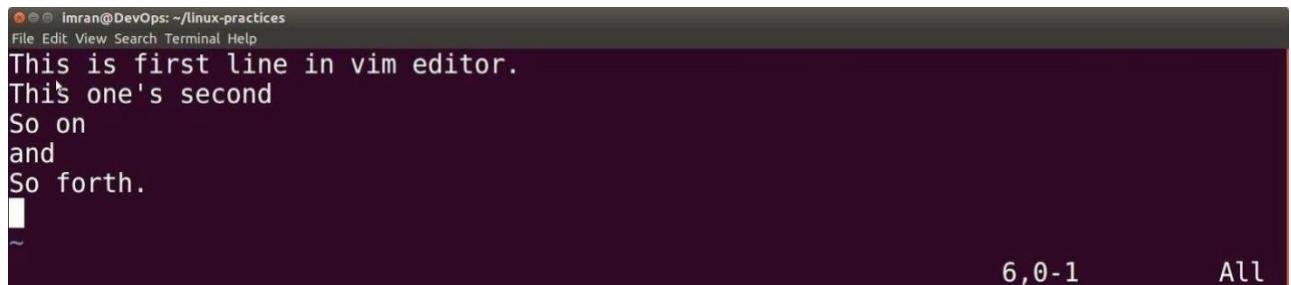
```
imran@DevOps:~/linux-practices$ vim firstfile.txt
```

→ Hit i to enter into insert mode



```
-- INSERT --
```

=> type few lines => hit Esc



```
This is first line in vim editor.
This one's second
So on
and
So forth.
```

=> type :wq

```
imran@DevOps: ~/linux-practices
File Edit View Search Terminal Help
This is first line in vim editor.
This one's second
So on
and
So forth.

~                                         :wq █
```

=> Enter.

→ Read file with cat command.

```

imran@DevOps:~/linux-practices$ cat firstfile.txt
This is first line in vim editor.
This one's second
So on
and
So forth.

imran@DevOps:~/linux-practices$ 

```

VIM EDITOR

VI Visual display editor

VIM Visual display editor improved

This is command mode editor for files. Other editors in Linux are emacs, gedit
vi editor is most popular

It has 3 modes:

- 1 Command Mode
- 2 Insert mode (edit mode)
- 3 extended command mode

Note: When you open the vim editor, it will be in the command mode by default.

Command Mode:

gg	To go to the beginning of the page
G	To go to end of the page
w	To move the cursor forward, word by word
b	To move the cursor backward, word by word
nw	To move the cursor forward to n words (SW)
nb	To move the cursor backward to n words (SB)
u	To undo last change (word)

U	To undo the previous changes (entire line)
Ctrl+R	To redo the changes
YY	To copy a line
nyy	To copy n lines (Syy or 4yy)
p	To paste line below the cursor position
P	To paste line above the cursor position
dw	To delete the word letter by letter {like Backspace}
x	To delete the world letter by letter (like DEL Key)
dd	To delete entire line
ndd	To delete n no. of lines from cursor position{Sdd}
I	To search a word in the file

Extended Mode: (Colon Mode)

Extended Mode is used for save and quit or save without quit using "Esc" Key with ":"

Esc+w	To Save the changes	j"-' \''' .. /
Esc+q	To quit (Without saving)	...^Z ..
Esc+wq	To save and quit	r \
Esc+w!	To save forcefully	/_.. \>:/
Esc+wq!	To save and quit forcefully	f_I ^Z
Esc+x	To save and quit	
Esc+X	To give password to the file and remove password	
Esc+20(n)	To go to line no 20 or n	
Esc+: se nu	To set the line numbers to the file	/
Esc+:se nonu	To Remove the set line numbers	

ls command options

Options	Description
-l	Long listing format of files and directories, one per line
-a	List all hidden files and directories started with '.'
-F	Add a '/' classification at the end of each Directory
-g	List all files and directories with the group name
-i	Print index number of each files and directories
-m	List all file and directories separated by comma ','
-n	List numeric UID and GID of Owner and Groups
-r	List all files and directories in reverse order
-R	Short list all directories
-t	Sorted by modified time, started with the newest file

Types of files in linux.

File Type	First Character in File Listing	Description
Regular file	-	Normal files such as text, data, or executable files
Directory	d	Files that are lists of other files
Link	l	A shortcut that points to the location of the actual file
Special file	c	Mechanism used for input and output, such as files in /dev
Socket	s	A special file that provides inter-process networking protected by the file system's access control
Pipe	p	A special file that allows processes to communicate with each other without using network socket semantics

Symbolic links

Symbolic links are like desktop shortcuts we use in windows.

Create a soft link for /var/log directory in our current working directory.

```
imran@DevOps:~/linux-practices$ ls
file2 firstfile.txt vpdir
imran@DevOps:~/linux-practices$ ls /var/log/
alternatives.log auth.log.1 cups fontconfig.log kern.log.1 prime-supported.log vbox-install.log Xorg.1.log.old
alternatives.log.1 boot.log dist-upgrade fsck lastlog speech-dispatcher wtmp Xorg.2.log
apport.log boot-sav dmesg gpu-manager.log lightdm syslog wtmp.1
apport.log.1 bootstrap.log dpkg.log installer old-logs syslog.1 Xorg.0.log
apt btmp dpkg.log.1 jenkins php7.0-fpm.log unattended-upgrades Xorg.0.log.old
auth.log btmp.1 faillog kern.log php7.0-fpm.log.1 upstart Xorg.1.log
imran@DevOps:~/linux-practices$ ln -s /var/log/ logdir
imran@DevOps:~/linux-practices$ ls -l
total 8
-rw-rw-r-- 1 imran imran 0 Apr 2 18:00 file2
-rw-rw-r-- 1 imran imran 73 Apr 2 18:29 firstfile.txt
lrwxrwxrwx 1 imran imran 9 Apr 2 18:41 logdir -> /var/log/
drwxrwxr-x 4 imran imran 4096 Apr 2 18:21 vpdir
imran@DevOps:~/linux-practices$ ls logdir
alternatives.log auth.log.1 cups fontconfig.log kern.log.1 prime-supported.log vbox-install.log Xorg.1.log.old
alternatives.log.1 boot.log dist-upgrade fsck lastlog speech-dispatcher wtmp Xorg.2.log
apport.log boot-sav dmesg gpu-manager.log lightdm syslog wtmp.1
apport.log.1 bootstrap.log dpkg.log installer old-logs syslog.1 Xorg.0.log
apt btmp dpkg.log.1 jenkins php7.0-fpm.log unattended-upgrades Xorg.0.log.old
auth.log btmp.1 faillog kern.log php7.0-fpm.log.1 upstart Xorg.1.log
imran@DevOps:~/linux-practices$
```

4. Filter & IO redirection command.

Grep

grep command is used to find texts from any text input.

Passwd file: stores information about all the users in the system

```
imran@DevOps:~/linux-practices$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
imran:x:1000:1000:imran:/home/imran:/bin/bash
```

→ Finding line which contains word as “root” from /etc/passwd file.

```
imran@DevOps:~/linux-practices$ grep root /etc/passwd
root:x:0:0:root:/root:/bin/bash
imran@DevOps:~/linux-practices$
```

→ Linux is case sensitive, Root is different than root. Ignoring case in grep with -i option.

```
imran@DevOps:~/linux-practices$ grep Root /etc/passwd
imran@DevOps:~/linux-practices$ grep -i Root /etc/passwd
root:x:0:0:root:/root:/bin/bash
```

→ To display things except the given word use -v option

Filter Commands

- **less:** Displays file content page wise or line wise. Ex: less /etc/passwd

Note: -press **Enter** key to scroll down line by line (or)

Use **d** to go to next page

Use **b** to go to previous page

Use **/** to search for a word in the file

Use **v** to go vi mode where you can edit the file and once you save it you will back to less command

- **more**

more is exactly same like **less**

Ex: #more /etc/passwd

Note: -press **Enter** key to scroll down line by line (or)

Use **d** to go to next page

Use **/** to search for a word in the file

Use **v** to go vi mode where you can edit the file and once you save it you will back to more command

- **head**

It is used to display the top **10 lines** of the file.

Ex:# head /etc/passwd

```
[root@ktlinux ~]# head /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
```

- **tail**

It is used to display the **last 10 lines** of the file

#tail /etc/passwd

```
[root@ktlinux ~]# tail /etc/passwd
apache:x:48:48:Apache:/var/www:/sbin/nologin
nslcd:x:65:55:LDAP Client User:::/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
pulse:x:496:494:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72:::/sbin/nologin
visitor:x:500:500:visitor:/home/visitor:/bin/bash
ktuser:x:501:501::/home/ktuser:/bin/bash
```

- **cut**

```
# cut -d -f filename (where d stands for delimiter ex. :, " " etc and f stands for field)
```

```
[root@ktlinux ~]# cut -d: -f1 /etc/passwd
root
bin
daemon
adm
lp
sync
shutdown
halt
mail
uucp
```

To delimit spaces and print the field

```
#cut -d " " -f1 filename
```

- **sed**

sed stands for **stream editor**, which is used to search a word in the file and replace it with the word required to be in the output

Note: it will only modify the output, but there will be no change in the original file.

```
#sed 's/searchfor/replacewith/g' filename
```

```
[root@ktlinux ~]# cat ktfile
Welcome to Kernel Tech
[root@ktlinux ~]# sed 's/Tech/Technologies/g' ktfile
Welcome to Kernel Technologies
[root@ktlinux ~]# cat ktfile
Welcome to Kernel Tech
```

I/O redirection

Redirection is a process where we can copy the output of any command(s), file(s) into a new file. There are two ways of redirecting the output into a file.

Using **>** or **>> filename** after the command, and

→ Create a file named devopstools with below content.

```
imran@DevOps:~/linux-practices$ cat devopstools
chef tech
ansible tech
git tech
docker tech
aws tech
```

→ Search for text “tech” replace it with “tools” and redirect output to a new file.

```
imran@DevOps:~/linux-practices$ sed 's/tech/tools/g' devopstools
chef tools
ansible tools
git tools
docker tools
aws tools
imran@DevOps:~/linux-practices$ sed 's/tech/tools/g' devopstools > newtools.txt
imran@DevOps:~/linux-practices$ cat newtools.txt
chef tools
ansible tools
git tools
docker tools
aws tools
```

Note: if the given name of the file is not available a new file will be created automatically. If the file already exists then it will overwrite contents of that file.

→ Appending another output in same file with “>>” .

```
imran@DevOps:~/linux-practices$ tail /etc/passwd >> newtools.txt
imran@DevOps:~/linux-practices$ cat newtools.txt
chef tools
ansible tools
git tools
docker tools
aws tools
pulse:x:117:124:PulseAudio daemon,,,:/var/run/pulse:/bin/false
rtkit:x:118:126:RealtimeKit,,,:/proc:/bin/false
saned:x:119:127::/var/lib/saned:/bin/false
usbmux:x:120:46:usbmux daemon,,,:/var/lib/usbmux:/bin/false
imran:x:1000:1000:Imran,,,:/home/imran:/bin/bash
jenkins:x:121:131:Jenkins,,,:/var/lib/jenkins:/bin/bash
guest-lx1wni:x:999:999:Guest:/tmp/guest-lx1wni:/bin/bash
guests:x:1001:1001,,,:/home/guests:/bin/bash
nvidia-persistenced:x:122:132:NVIDIA Persistence Daemon,,,:/sbin/nologin
guest-yjzlgk:x:998:998:Guest:/tmp/guest-yjzlgk:/bin/bash
imran@DevOps:~/linux-practices$
```

→ Redirecting only error to a file “2>>”.

```
root@localhost:~ [root@localhost ~]# uptimer 2>> /tmp/error.log [root@localhost ~]# |
```

→ Redirecting all the output to a file “&>>”.

```
root@localhost:~ [root@localhost ~]# uptimer &>> /tmp/error.log [root@localhost ~]# uptime &>> /tmp/error.log [root@localhost ~]# |
```

Piping

So far we've dealt with sending data to and from files. Now we'll take a look at a mechanism for sending data from one program to another. It's called piping and the operator we use is (|). What this operator does is feed the output from the program on the left as input to the program on the right.

```
imran@DevOps:~$ cd linux-practices/
imran@DevOps:~/linux-practices$ ls
chefdk_1.2.22-1_amd64.deb  file2      logdir      tree_1.7.0-3_amd64.deb
devopstools                 firstfile.txt newtools.txt  vmdir
imran@DevOps:~/linux-practices$ ls | head -3
chefdk_1.2.22-1_amd64.deb
devopstools
file2
imran@DevOps:~/linux-practices$ ls | grep logdir
logdir
imran@DevOps:~/linux-practices$ cat /etc/passwd | grep root
root:x:0:0:root:/root:/bin/bash
imran@DevOps:~/linux-practices$ █
```

Find

find command is used to find the files or directory's path, it is exactly like the find option in windows where you can search for a file.

```
imran@DevOps:~/linux-practices$ find /home/imran/ -name newtools.txt
/home/imran/linux-practices/newtools.txt
```

Options that can be used with find command:

Option	Usage
-name	For searching a file with its name
-inum	For searching a file with particular inode number
-type	For searching a particular type of file
-user	For files whose owner is a particular user
-group	For files belonging to particular group

5. Users & Groups.

USERS

Some Important Points related to Users:

- Users and groups are used to control access to files and resources
 - Users login to the system by supplying their username and password
 - Every file on the system is owned by a user and associated with a group
 - Every process has an owner and group affiliation, and can only access the resources its owner or group can access.
-
- Every user of the system is assigned a unique user ID number (the UID)
 - Users name and UID are stored in **/etc/passwd**
 - User's password is stored in **/etc/shadow** in encrypted form.
 - Users are assigned a **home directory** and a program that is run when they login (**Usually a shell**)
 - Users cannot read, write or execute each other's files without permission.

Types of user

TYPE	EXAMPLE	USER ID (ID)	GROUP ID (GID)	HOME DIR	SHELL
ROOT	root	0	0	/root	/bin/bash
REGULAR	imran, vagrant	1000 to 60000	1000 to 60000	/home/username	/bin/bash
SERVICE	ftp, ssh, apache	1 to 999	1 to 999	/var/ftp etc	/sbin/nologin

In Linux there are three types of users.

1. Super user or root user

Super user or the root user is the most powerful user. He is the administrator user.

2. System user

System users are the users created by the softwares or applications. For example if we install Apache it will create a user apache. These kinds of users are known as system users.

3. Normal user

Normal users are the users created by root user. They are normal users like Rahul, Musab etc. Only the root user has the permission to create or remove a user.

Whenever a user is created in Linux things created by default:-

- A home directory is created(/home/username)
- A mail box is created(/var/spool/mail)
- unique UID & GID are given to user

Passwd file

1. /etc/passwd

```
[root@ktlinux ~]# head /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
```

The above fields are

- **root** =name
- **x**= link to password file i.e. /etc/shadow
- **0 or 1**= UID (user id)
- **0 or 1**=GID (group id)
- **root or bin** = comment (brief information about the user)
- **/root or /bin** = home directory of the user
- **/bin/bash or /sbin/nologin** = shell

Group file

2. /etc/group

The file /etc/group stores group information. Each line in this file stores one group entry.

Group name, group password, GID, group members

```
[root@localhost ~]# head /etc/group
root:x:0:
bin:x:1:
daemon:x:2:
```

ADD USER, SET PASSWORD & SWITCH TO USER

```
dino@localhost:~$ [vagrant@localhost ~]$ sudo useradd dino
[vagrant@localhost ~]$ sudo passwd dino
Changing password for user dino.
New password: I
Retype new password:
Sorry, passwords do not match.
New password: I
Retype new password:
passwd: all authentication tokens updated successfully.
[vagrant@localhost ~]$ su - dino
Password:
[dino@localhost ~]$ pwd
/home/dino
[dino@localhost ~]$ id
uid=1002(dino) gid=1003(dino) groups=1003(dino) context=unconfined_u:unconfined_
r:unconfined_t:s0-s0:c0.c1023
[dino@localhost ~]$ |
```

ADD USER, GROUP & USER INTO GROUP

```
root@localhost:~# [root@localhost ~]# useradd devops
[root@localhost ~]# id devops
uid=1001(devops) gid=1001(devops) groups=1001(devops)
[root@localhost ~]# grep devops /etc/passwd
devops:x:1001:1001::/home/devops:/bin/bash
[root@localhost ~]# groupadd opsadmin
[root@localhost ~]# usermod -G opsadmin devops
[root@localhost ~]# grep opsadmin /etc/group
opsadmin:x:1002:devops
[root@localhost ~]# id devops
uid=1001(devops) gid=1001(devops) groups=1001(devops),1002(opsadmin)
[root@localhost ~]# |
```

DELETE USER & GROUP

```
vagrant@localhost:~$ [vagrant@localhost ~]$ sudo userdel -r dino
[vagrant@localhost ~]$ sudo groupdel opsadmin
[vagrant@localhost ~]$ sudo id dino
id: dino: no such user
[vagrant@localhost ~]$ |
```

3. The /etc/shadow file

This file stores users' password and password related information. Just like */etc/passwd* file, this file also uses an individual line for each entry.

1. Username
2. Encrypted password
3. Number of days when password was last changed
4. Number of days before password can be changed
5. Number of days after password must be changed
6. Number of days before password expiry date to display the warning message

7. Number of days to disable the account after the password expiry
8. Number of days since the account is disabled
9. Reserved field

```
[root@localhost ~]# cat /etc/shadow
root:$1$M.FEVNi$OYiaRNHMHZS85/wnDHccI.:0
bin:*:18353:0:99999:7:::
daemon:*:18353:0:99999:7:::
adm:*:18353:0:99999:7:::
lp:*:18353:0:99999:7:::
sync:*:18353:0:99999:7:::
shutdown:*:18353:0:99999:7:::
halt:*:18353:0:99999:7:::
mail:*:18353:0:99999:7:::
```

USER & GROUP cheatsheet

COMMANDS	DESCRIPTION
useradd	Creates user in RedHat
adduser	Creates user in ubuntu
id	Shows user info
groupadd	Creates group
usermod -G grpnam username	Adds user to group
passwd	set/reset password
userdel -r	removes user with home dir
groupdel	removes group
last	shows last login in system
who	who is logged into system
whoami	username
lsof -u user	List files opened by user

6. File permissions

Viewing Permissions from the Command-Line

- File permissions may be viewed using **ls -l**

```
$ ls -l /bin/login
-rwxr-xr-x 1 root root 19080 Apr 1 18:26 /bin/login
```

- Four symbols are used when displaying permissions:
 - r: permission to read a file or list a directory's contents
 - w: permission to write to a file or create and remove files from a directory
 - x: permission to execute a program or change into a directory and do a long listing of the directory
 - -: no permission (in place of the r, w, or x)

Changing File Ownership

- Only root can change a file's owner
- Only root or the owner can change a file's group
- Ownership is changed with **chown**:
 - **chown [-R] user_name file|directory ...**
- Group-Ownership is changed with **chgrp**:
 - **chgrp [-R] group_name file|directory ...**

```

File Edit View Search Terminal Help
imran@DevOps:~/linux-practices$ sudo adduser sam
[sudo] password for imran:
Adding user `sam' ...
Adding new group `sam' (1002) ...
Adding new user `sam' (1002) with group `sam' ...
Creating home directory `/home/sam' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for sam
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] Y
imran@DevOps:~/linux-practices$ id sam
uid=1002(sam) gid=1002(sam) groups=1002(sam)
imran@DevOps:~/linux-practices$ ls -l devopstools
-rw-rw-r-- 1 imran imran 53 Apr  2 19:09 devopstools
imran@DevOps:~/linux-practices$ chown sam.sam devopstools
chown: changing ownership of 'devopstools': Operation not permitted
imran@DevOps:~/linux-practices$ sudo chown sam.sam devopstools
imran@DevOps:~/linux-practices$ ls -l devopstools
-rw-rw-r-- 1 sam sam 53 Apr  2 19:09 devopstools
imran@DevOps:~/linux-practices$ ls
devopstools file2 firstfile.txt logdir newtools.txt vmdir
imran@DevOps:~/linux-practices$ ls -l vmdir/
total 8
drwxrwxr-x 4 imran imran 4096 Apr  2 18:14 devopsdir
-rw-rw-r-- 1 imran imran   0 Apr  2 18:00 file3
-rw-rw-r-- 1 imran imran   0 Apr  2 18:00 file4
drwxrwxr-x 2 imran imran 4096 Apr  2 18:17 testdir
imran@DevOps:~/linux-practices$ chown sam.sam vmdir/ -R
chown: changing ownership of 'vmdir/devopsdir/ansible': Operation not permitted
chown: changing ownership of 'vmdir/devopsdir/aws': Operation not permitted
chown: changing ownership of 'vmdir/devopsdir': Operation not permitted
chown: changing ownership of 'vmdir/file4': Operation not permitted
chown: changing ownership of 'vmdir/file3': Operation not permitted

```

Changing Permissions - Symbolic Method

- To change access modes:
 - **chmod [-OPTION] ... mode[,mode] file| directory ...**
 - *mode* includes:
 - **u,g or o** for user, group and other
 - **+ - or =** for grant, deny or set
 - **r, w or x** for read, write and execute
 - Options include:
 - **-R** Recursive
 - **-v** Verbose
 - **--reference** Reference another file for its mode
 - Examples:
 - **chmod ugo+r file:** Grant read access to all for *file*
 - **chmod o-wx dir:** Deny write and execute to others for *dir*
-

Changing Permissions - Numeric Method

- Uses a three-digit mode number
 - first digit specifies owner's permissions
 - second digit specifies group permissions
 - third digit represents others' permissions
- Permissions are calculated by adding:
 - 4 (for read)
 - 2 (for write)
 - 1 (for execute)
- Example:
 - **chmod 640 myfile**
 - **.rW-r--r--**

10. 6 4 4 644]]

```
drwxr-xr-x 2 teja teja 4096 Oct 28 08:21 Pictures
777
```

```
imran@DevOps:~/linux-practices$ ls -l
total 16
-rw-rw-r-- 1 sam sam 53 Apr 2 19:09 devopstools
-rw-rw-r-- 1 imran imran 0 Apr 2 18:00 file2
-rw-rw-r-- 1 imran imran 73 Apr 2 18:29 firstfile.txt
lrwxrwxrwx 1 imran imran 9 Apr 2 18:41 logdir -> /var/log/
-rw-rw-r-- 1 imran imran 612 Apr 2 19:14 newtools.txt
drwxrwxr-x 4 sam sam 4096 Apr 2 18:21 vmdir
imran@DevOps:~/linux-practices$ chmod u+x newtools.txt
imran@DevOps:~/linux-practices$ ls -l newtools.txt
-rwxrw-r-- 1 imran imran 612 Apr 2 19:14 newtools.txt
imran@DevOps:~/linux-practices$ chmod o-r newtools.txt
imran@DevOps:~/linux-practices$ ls -l newtools.txt
-rwxrw---- 1 imran imran 612 Apr 2 19:14 newtools.txt
imran@DevOps:~/linux-practices$ chmod 700 newtools.txt
imran@DevOps:~/linux-practices$ ls -l newtools.txt
-rwx----- 1 imran imran 612 Apr 2 19:14 newtools.txt
imran@DevOps:~/linux-practices$ chmod 755 newtools.txt
imran@DevOps:~/linux-practices$ ls -l newtools.txt
-rwxr-xr-x 1 imran imran 612 Apr 2 19:14 newtools.txt
imran@DevOps:~/linux-practices$
```

7. Sudo

sudo gives power to a normal user to execute commands which is owned by root user.

Example shown below:

If a user has already full sudoers privilege, it can become a root user anytime.

→ sudo -i changes from normal user to root user

```
imran@DevOps:~/linux-practices$ id
uid=1000(imran) gid=1000(imran) groups=1000(imran),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),113(lpadmin),
imran@DevOps:~/linux-practices$ sudo -i
[sudo] password for imran:
root@DevOps:~# id
uid=0(root) gid=0(root) groups=0(root)
root@DevOps:~#
```

Note: User imran was already a sudo user with full privilege.

→ Adding user sam in sudoers list.

```
imran@DevOps:~/linux-practices$ sudo -i
root@DevOps:~# export EDITOR=vim
root@DevOps:~# visudo
```

```
File Edit View Search Terminal Help
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults      env_reset
Defaults      mail_badpass
Defaults      secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL
sam    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL
%dev_dock      ALL=(ALL:ALL) ALL
```

21,3

Top

→ Like a user a group can also be added into sudoers list.

```
# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL
```

→ Every time you enter sudo command it asks your own password. To turn that off use NOPASSWD in sudoers file.

```
# User privilege specification
root    ALL=(ALL:ALL) ALL
sam     ALL=(ALL:ALL) NOPASSWD: ALL
```

→ Changing to any other user with “su -” command.

```
imran@DevOps:~/linux-practices$ su - sam
Password:
sam@DevOps:~$ █
```

→ Become a root user from sam user login.

```
sam@DevOps:~$ sudo -i
root@DevOps:~# █
```

sudo su

8. Software Management.

ios -- appstore -- chrome - rhel
google -- playstore -- chrome - debain

→ Download package from internet.

For CentOS

To install Tree

```
# curl https://rpmfind.net/linux/centos/7.9.2009/os/x86\_64/Packages/tree-1.6.0-10.el7.x86\_64.rpm -o tree-1.6.0-10.el7.x86_64.rpm
```

```
# rpm -ivh tree-1.6.0-10.el7.x86_64.rpm
```

```
[root@Imran ~]# curl https://rpmfind.net/linux/centos/7.9.2009/os/x86_64/Packages/tree-1.6.0-10.el7.x86_64.rpm -o tree-1.6.0-10.el7.x86_64.rpm
% Total    % Received % Xferd  Average Speed   Time   Time     Current
          Dload  Upload Total Spent   Left  Speed
100 47508  100 47508    0     0  57226      0 --:--:-- --:--:-- 57238
[root@Imran ~]# ls
anaconda-ks.cfg  http://httpd-2.4.6-95.el7.centos.x86\_64.rpm  original-ks.cfg  tree-1.6.0-10.el7.x86_64.rpm
[root@Imran ~]# rpm -ivh tree-1.6.0-10.el7.x86_64.rpm
Preparing...: ################################################ [100%]
Updating...: 1:tree-1.6.0-10.el7  ################################################ [100%]
[root@Imran ~]# tree /var/log/
/var/log/
└── anaconda
    ├── anaconda.log
    ├── ifcfg.log
    ├── journal.log
    ├── ks-script-8988xq.log
    ├── ks-script-dyarry.log
    ├── ks-script-kPd16m.log
    ├── ks-script-Wnz4e2.log
    ├── packaging.log
    ├── program.log
    ├── storage.log
    └── syslog
└── audit
    └── audit.log
├── boot.log
├── btmp
├── chrony
├── cloud-init.log
├── cron
├── dmesg
├── grubby_prune_debug
├── lastlog
├── maillog
├── messages
├── qemu-ga
├── rhsm
├── secure
└── spooler
```

To install httpd

```
# curl https://rpmfind.net/linux/centos/7.9.2009/os/x86\_64/Packages/httpd-2.4.6-95.el7.centos.x86\_64.rpm -o httpd-2.4.6-95.el7.centos.x86_64.rpm
```

```
# rpm -ivh httpd-2.4.6-95.el7.centos.x86_64.rpm
```

```
[root@Imran ~]# curl https://rpmfind.net/linux/centos/7.9.2009/os/x86_64/Packages/httpd-2.4.6-95.el7.centos.x86_64.rpm -o httpd-2.4.6-95.el7.centos.x86_64.rpm
  % Total    % Received % Xferd  Average Speed   Time     Time   Current
     0   2779k   100 2779k      0      0  1943k      0  0:00:01  0:00:01 ---:--- 1945k
[root@Imran ~]# ls
anaconda-ks.cfg  httpd-2.4.6-95.el7.centos.x86_64.rpm  original-ks.cfg
[root@Imran ~]# rpm -ivh httpd-2.4.6-95.el7.centos.x86_64.rpm
warning: httpd-2.4.6-95.el7.centos.x86_64.rpm: Header V3 RSA/SHA256 Signature, key ID f4a80eb5: NOKEY
error: Failed dependencies:
        /etc/mime.types is needed by httpd-2.4.6-95.el7.centos.x86_64
        system-logos >= 7.92.1-1 is needed by httpd-2.4.6-95.el7.centos.x86_64
        httpd-tools = 2.4.6-95.el7.centos is needed by httpd-2.4.6-95.el7.centos.x86_64
        libapr-1.so.0()(64bit) is needed by httpd-2.4.6-95.el7.centos.x86_64
        libaprutil-1.so.0()(64bit) is needed by httpd-2.4.6-95.el7.centos.x86_64
[root@Imran ~]# |
```

Due to Dependencies its failing and it will be installed one we install all the dependencies. But what if we have Hundreds of dependencies, And that can be solved easily by other package managers like YUM .

repos. d/ directory. It reads each YUM Repository configuration file to get the information required to

download and install new software, resolves software dependencies and installs the required RPM package files. YUM Repository configuration files must: be located in /etc/yum.repos.d

```
# ls /etc/yum.repos.d/
```

```
[root@Imran ~]#  
[root@Imran ~]# ls /etc/yum.repos.d/  
CentOS-Base.repo  CentOS-CR.repo  CentOS-Debuginfo.repo  CentOS-fasttrack.repo  CentOS-Media.repo  CentOS-Sources.repo  CentOS-Vault.repo
```

Shows the usage of YUM Command with options

```
# yum --help
```

```
[root@Imran ~]#  
[root@Imran ~]# yum --help  
Loaded plugins: fastestmirror  
Usage: yum [options] COMMAND  
  
List of Commands:  
check      Check for problems in the rpmdb  
check-update  Check for available package updates  
clean       Remove cached data  
depplist    List a package's dependencies  
distribution-synchronization Synchronize installed packages to the latest available versions  
downgrade   downgrade a package  
erase       Remove a package or packages from your system  
fs          Acts on the filesystem data of the host, mainly for removing docs/languages for minimal hosts.  
fssnapshot  Creates filesystem snapshots, or lists/deletes current snapshots.  
groups      Display, or use, the groups information  
help        Display a helpful usage message  
history     Display, or use, the transaction history  
info        Display details about a package or group of packages  
install     Install a package or packages on your system  
list        List a package or groups of packages  
load-transaction load a saved transaction from filename  
makecache   Generate the metadata cache  
provides    Find what package provides the given value  
reinstall   reinstall a package  
repo-pkgs   Treat a repo. as a group of packages, so we can install/remove all of them  
repolist    Display the configured software repositories  
search      Search package details for the given string  
shell       Run an interactive yum shell  
swap        Simple way to swap packages, instead of using shell  
update      Update a package or packages on your system  
update-minimal Works like upgrade, but goes to the 'newest' package match which fixes a problem that affects your system  
updateinfo   Acts on repository update information  
upgrade     Update packages taking obsoletes into account  
version     Display a version for the machine and/or available repos.  
  
Options:  
 -h, --help           show this help message and exit  
 -t, --tolerant      be tolerant of errors
```

To Update all your packages

```
# yum update
```

```
[root@Imran ~]# yum update  
Loaded plugins: fastestmirror  
Loading mirror speeds from cached hostfile  
 * base: download.cf.centos.org  
 * extras: download.cf.centos.org  
 * updates: download.cf.centos.org  
Resolving Dependencies  
--> Running transaction check  
--> Package acl.x86_64 0:2.2.51-14.el7 will be updated  
--> Package acl.x86_64 0:2.2.51-15.el7 will be an update  
--> Package bash.x86_64 0:4.2.46-33.el7 will be updated  
--> Package bash.x86_64 0:4.2.46-34.el7 will be an update  
--> Package bind-export-libs.x86_64 32:9.11.4-9.P2.el7 will be updated  
--> Package bind-export-libs.x86_64 32:9.11.4-26.P2.el7_9_7 will be an update  
--> Package binutils.x86_64 0:2.27-44.base.el7_7_2 will be updated  
--> Package binutils.x86_64 0:2.27-44.base.el7 will be an update  
--> Package ca-certificates.noarch 0:2019.2.32-76.el7_7 will be updated  
--> Package ca-certificates.noarch 0:2021.2.50-72.el7_9 will be an update  
--> Package centos-release.x86_64 0:7-7.1908.0.el7.centos will be updated  
--> Package centos-release.x86_64 0:7-9.2009.1.el7.centos will be an update  
--> Package chkconfig.x86_64 0:1.7.4-1.el7 will be updated  
--> Package chkconfig.x86_64 0:1.7.6-1.el7 will be an update  
--> Package cloud-init.x86_64 0:18.5-3.el7.centos will be updated
```

To install httpd

```
# yum install httpd -y
```

```
[root@Imran ~]#
[root@Imran ~]# yum install httpd -y
Loaded plugins: fastestmirror
Determining fastest mirrors
 * base: download.cf.centos.org
 * extras: download.cf.centos.org
 * updates: download.cf.centos.org
base
extras
updates
(1/4): base/7/x86_64/group_gz | 3.6 kB 00:00:00
(2/4): extras/7/x86_64/primary_db | 2.9 kB 00:00:00
(3/4): base/7/x86_64/primary_db | 2.9 kB 00:00:00
(4/4): updates/7/x86_64/primary_db | 153 kB 00:00:00
Resolving Dependencies
--> Running transaction check
Total                                         22 MB/s | 24 MB 00:00:01
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
Importing GPG key 0xF4A80EB5:
Userid : "CentOS-7 Key (CentOS 7 Official Signing Key) <security@centos.org>"
Fingerprint: 6341 ab27 53d7 8a78 a7c2 7bb1 24c6 a8a7 f4a8 0eb5
Package  : centos-release-7-1908.0.el7.centos.x86_64 (installed)
From    : /etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : apr-1.4.8-7.el7.x86_64          1/6
  Installing : apr-util-1.5.2-6.el7.x86_64      2/6
  Installing : httpd-tools-2.4.6-97.el7.centos.x86_64 3/6
  Installing : centos-logos-70.0.6-3.el7.centos.noarch 4/6
  Installing : mailcap-2.1.41-2.el7.noarch       5/6
  Installing : httpd-2.4.6-97.el7.centos.x86_64   6/6
  Verifying  : mailcap-2.1.41-2.el7.noarch       1/6
  Verifying  : apr-1.4.8-7.el7.x86_64           2/6
  Verifying  : apr-util-1.5.2-6.el7.x86_64       3/6
  Verifying  : httpd-2.4.6-97.el7.centos.x86_64 4/6
  Verifying  : httpd-tools-2.4.6-97.el7.centos.x86_64 5/6
  Verifying  : centos-logos-70.0.6-3.el7.centos.noarch 6/6
Installed:
  httpd.x86_64 0:2.4.6-97.el7.centos

Dependency Installed:
  apr.x86_64 0:1.4.8-7.el7    apr-util.x86_64 0:1.5.2-6.el7  centos-logos.noarch 0:70.0.6-3.el7.centos  httpd-tools.x86_64 0:2.4.6-97.el7.centos
  mailcap.noarch 0:2.1.41-2.el7

Complete!
```

To remove httpd

```
# yum remove httpd -y
```

```
[root@Imran ~]# yum remove httpd -y
Loaded plugins: fastestmirror
Resolving Dependencies
--> Running transaction check
--> Package httpd.x86_64 0:2.4.6-97.el7.centos will be erased
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package          Arch      Version           Repository      Size
=====
Removing:       httpd      x86_64    2.4.6-97.el7.centos @updates      9.4 M

Transaction Summary
=====
Remove 1 Package

Installed size: 9.4 M
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Erasing   : httpd-2.4.6-97.el7.centos.x86_64          1/1
  Verifying : httpd-2.4.6-97.el7.centos.x86_64          1/1

Removed:
  httpd.x86_64 0:2.4.6-97.el7.centos

Complete!
```

For Ubuntu

```
# wget http://archive.ubuntu.com/ubuntu/pool/universe/t/tree/tree_1.7.0-3_amd64.deb -o tree_1.7.0-3_amd64.deb
```

```
# dpkg -i tree_1.7.0-3_amd64.deb
```

```

root@Imran:~#
root@Imran:~# wget http://archive.ubuntu.com/ubuntu/pool/universe/t/tree/tree_1.7.0-3_amd64.deb
--2021-09-27 04:30:17-- http://archive.ubuntu.com/ubuntu/pool/universe/t/tree/tree_1.7.0-3_amd64.deb
Resolving archive.ubuntu.com (archive.ubuntu.com)... 91.189.88.142, 91.189.88.152, 2001:67c:1360:8001::24, ...
Connecting to archive.ubuntu.com (archive.ubuntu.com)|91.189.88.142|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 40572 (40K) [application/x-debian-package]
Saving to: 'tree_1.7.0-3_amd64.deb'

tree_1.7.0-3_amd64.deb      100%[=====] 39.62K --.-KB/s   in 0.07s

2021-09-27 04:30:17 (551 KB/s) - 'tree_1.7.0-3_amd64.deb' saved [40572/40572]

root@Imran:~# dpkg -i tree_1.7.0-3_amd64.deb
Selecting previously unselected package tree.
(Reading database ... 63739 files and directories currently installed.)
Preparing to unpack tree_1.7.0-3_amd64.deb ...
Unpacking tree (1.7.0-3) ...
Setting up tree (1.7.0-3) ...
Processing triggers for man-db (2.9.1-1) ...
root@Imran:~# tree /var/log/
/var/log/
└── amazon
    └── SSM
        ├── amazon-ssm-agent.log
        │   ├── audits
        │   │   └── amazon-ssm-agent-audit-2021-09-27
        │   └── hibernate.log
    └── apt
        ├── eipp.log.xz
        ├── history.log
        └── term.log
    auth.log
    btmp
    cloud-init-output.log
    cloud-init.log
    dist-upgrade
    dmesg

```

We have seen YUM Like that for Ubuntu we have a package manager ‘apt’.

The **sources.list** file is a key factor in adding or upgrading applications to your Ubuntu installation. This is also used by your system for system updates. The file is basically the roadmap for your system to know where it may download programs for installation or upgrade.

```
# cat /etc/apt/sources.list
```

```

root@Imran:~#
root@Imran:~# cd /etc/apt/
root@Imran:/etc/apt# ls
apt.conf.d auth.conf.d preferences.d sources.list sources.list.d trusted.gpg.d
root@Imran:/etc/apt# cat /etc/apt/sources.list
## Note, this file is written by cloud-init on first boot of an instance
## modifications made here will not survive a re-bundle.
## if you wish to make changes you can:
## a.) add 'apt_preserve_sources_list: true' to /etc/cloud/cloud.cfg
##     or do the same in user-data
## b.) add sources in /etc/apt/sources.list.d
## c.) make changes to template file /etc/cloud/templates/sources.list tmpl

# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
# newer versions of the distribution.
deb http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal main restricted
# deb-src http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal main restricted

## Major bug fix updates produced after the final release of the
## distribution.
deb http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal-updates main restricted
# deb-src http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal-updates main restricted

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team. Also, please note that software in universe WILL NOT receive any
## review or updates from the Ubuntu security team.
deb http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal universe
# deb-src http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal universe
deb http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal-updates universe
# deb-src http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal-updates universe

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team, and may not be under a free licence. Please satisfy yourself as to
## your rights to use the software. Also, please note that software in
## multiverse WILL NOT receive any review or updates from the Ubuntu
## security team.
deb http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal multiverse
# deb-src http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal multiverse
deb http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal-updates multiverse
# deb-src http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ focal-updates multiverse

```

Shows the usage of apt Command with options

```
# apt --help
```

```
root@Imran:~# apt --help
apt 2.0.6 (amd64)
Usage: apt [options] command

apt is a commandline package manager and provides commands for
searching and managing as well as querying information about packages.
It provides the same functionality as the specialized APT tools,
like apt-get and apt-cache, but enables options more suitable for
interactive use by default.

Most used commands:
list - list packages based on package names
search - search in package descriptions
show - show package details
install - install packages
reinstall - reinstall packages
remove - remove packages
autoremove - Remove automatically all unused packages
update - update list of available packages
upgrade - upgrade the system by installing/upgrading packages
full-upgrade - upgrade the system by removing/installing/upgrading packages
edit-sources - edit the source information file
satisfy - satisfy dependency strings

See apt(8) for more information about the available commands.
Configuration options and syntax is detailed in apt.conf(5).
Information about how to configure sources can be found in sources.list(5).
Package and version choices can be expressed via apt_preferences(5).
Security details are available in apt-secure(8).

This APT has Super Cow Powers.
```

To update all your package lists

```
#apt update
```

```
root@Imran:~# apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu focal-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
27 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

TO search for a <package> apache2

```
# apt search apache2
```

```
root@Imran:~# apt search apache2
Sorting... Done
Full Text Search... Done
apache2/focal-updates,now 2.4.41-4ubuntu3.4 amd64 [installed]
 Apache HTTP Server

apache2-bin/focal-updates,now 2.4.41-4ubuntu3.4 amd64 [installed,automatic]
 Apache HTTP Server (modules and other binary files)

apache2-data/focal-updates,now 2.4.41-4ubuntu3.4 all [installed,automatic]
 Apache HTTP Server (common files)

apache2-dev/focal-updates 2.4.41-4ubuntu3.4 amd64
 Apache HTTP Server (development headers)

apache2-doc/focal-updates 2.4.41-4ubuntu3.4 all
 Apache HTTP Server (on-site documentation)

apache2-ss1-dev/focal-updates 2.4.41-4ubuntu3.4 amd64
 Apache HTTP Server (mod_ssl development headers)

apache2-suexec-custom/focal-updates 2.4.41-4ubuntu3.4 amd64
 Apache HTTP Server configurable suexec program for mod_suexec

apache2-suexec-pristine/focal-updates 2.4.41-4ubuntu3.4 amd64
 Apache HTTP Server standard suexec program for mod_suexec

apache2-utils/focal-updates,now 2.4.41-4ubuntu3.4 amd64 [installed,automatic]
 Apache HTTP Server (utility programs for web servers)
```

To install apache2

```
# apt install apache2 -y
```

```

root@Imran:~# apt install apache2 -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
The following NEW packages will be installed:
  apache2
0 upgraded, 1 newly installed, 0 to remove and 27 not upgraded.
Need to get 95.5 kB of archives.
After this operation, 542 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 apache2 amd64 apache2 2.4.41-4ubuntu3.4 [95.5 kB]
Fetched 95.5 kB in 0s (5456 kB/s)
Selecting previously unselected package apache2.
(Reading database ... 64421 files and directories currently installed.)
Preparing to unpack .../apache2_2.4.41-4ubuntu3.4_amd64.deb ...
Unpacking apache2 (2.4.41-4ubuntu3.4) ...
Setting up apache2 (2.4.41-4ubuntu3.4) ...
Processing triggers for systemd (245.4-4ubuntu3.11) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for ufw (0.36-6) ...
root@Imran:~#

```

To remove apache2

```
# apt remove apache2 -y
```

```

root@Imran:~# apt remove apache2 -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libjansson4 liblua5.2-0 ssl-cert
Use 'apt autoremove' to remove them.
The following packages will be REMOVED:
  apache2
0 upgraded, 0 newly installed, 1 to remove and 27 not upgraded.
After this operation, 542 kB disk space will be freed.
(Reading database ... 64471 files and directories currently installed.)
Removing apache2 (2.4.41-4ubuntu3.4) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for ufw (0.36-6) ...
root@Imran:~#

```

Commands	Description	
wget link	to download file from link	
curl link	access file from link	
curl link -o outputfile	access file and store o/p to a file	
REDHAT RPM commands		
rpm -ivh {rpm-file}	Install the package	rpm -ivh mozilla-mail-1.7.5-17 .i586.rpm rpm -ivh --test mozilla-mail-1.7.5-17 .i586.rpm
rpm -Uvh {rpm-file}	Upgrade package	rpm -Uvh mozilla-mail-1.7.6-12 .i586.rpm rpm -Uvh --test mozilla-mail-1.7.6-12 .i586.rpm

<code>rpm -ev {package}</code>	Erase/remove/ an installed package	<code>rpm -ev mozilla-mail</code>
<code>rpm -ev --nodeps {package}</code>	Erase/remove/ an installed package without checking for dependencies	<code>rpm -ev --nodeps mozilla-mail</code>

	dependencies	
<code>rpm -qa</code>	Display list all installed packages	<code>rpm -qa</code> <code>rpm -qa less</code>
<code>rpm -qi {package}</code>	Display installed information along with package version and short description	<code>rpm -qi mozilla-mail</code>
<code>rpm -qf {/path/to/file}</code>	Find out what package a file belongs to i.e. find what package owns the file	<code>rpm -qf /etc/passwd</code> <code>rpm -qf /bin/bash</code>
<code>rpm -qc {pacakge-name}</code>	Display list of configuration file(s) for a package	<code>rpm -qc httpd</code>
<code>rpm -qcf {/path/to/file}</code>	Display list of configuration files for a command	<code>rpm -qcf /usr/X11R6/bin/xeyes</code>
<code>rpm -qa --last</code>	Display list of all recently installed RPMs	<code>rpm -qa --last</code> <code>rpm -qa --last less</code>
<code>rpm -qpR {.rpm-file}</code> <code>rpm -qR {package}</code>	Find out what dependencies a rpm file has	<code>rpm -qpR mediawiki-1.4rc1-4.i586.rpm</code> <code>rpm -qR bash</code>
CentOS_8 Commands		
DNF commands cheatsheet	https://www.linuxtechi.com/dnf-command-examples-rpm-management-fedora-linux/	
<code>dnf --help</code>	Show the help	

<code>dnf search PACKAGE</code>	search from available repositories	
<code>dnf install PACKAGE -y</code>	To install the package	
<code>dnf install httpd -y</code>	To Install httpd package	
<code>dnf install vim -y</code>	Installing VIM Editor	
<code>dnf reinstall PACKAGE</code>	To reinstall PACKAGE	
<code>dnf remove PACKAGE</code>	To remove PACKAGE	
<code>dnf update</code>	update all packages	
<code>dnf update PACKAGE</code>	update only a package	

<code>dnf grouplist</code>	List all available Group Packages	
<code>dnf groupinstall "GROUPNAME"</code>	Installs all the packages in a group	
<code>dnf repolist</code>	List Enabled dnf Repositories	
<code>dnf clean all</code>	Clean dnf Cache	
<code>dnf install epel-release</code>	Additional package repository that provides easy access to install packages for commonly used software.	

dnf history	View History of dnf	
dnf info package name	Shows the information of package like version, size, source, repository etc	
YUM Commands Cheatsheet	https://access.redhat.com/sites/default/files/attachments/rh_yum_cheatsheet_1214_jcs_prient-1.pdf	
yum -help	Shows the help	
yum search PACKAGE	Search from available repositories	
yum install PACKAGE -y	To install the package	
yum install httpd -y	To install httpd package	
yum install vim -y	To install VIM Editor	
yum reinstall PACKAGE	To reinstall the PACKAGE	
Yum remove PACKAGE	To Remove PACKAGE	
yum update	Update all packages	
yum update PACKAGE	To Update specific package	
yum grouplist	List all available Group packages	
yum groupinstall "Group Name"	Install all the packages in a group	

Yum repolist	List Enabled YUM repositories	
yum install epel-release	Additional package repository that provides easy access to install packages for commonly used software.	
yum clean all	Clean yum cache	
yum history	View history of yum	
Yum info PACKAGE NAME	Shows the information of package like version, size, source, repository etc.	
Ubuntu20 Commands		
apt commands cheatsheet	https://itsfoss.com/apt-command-guide/	
apt search PACKAGE	search from available repositories	
apt install PACKAGE -y	To Install Packages	
apt install apache2 -y	To Install apache2	
apt reinstall PACKAGE	To reinstall PACKAGE	
apt remove PACKAGE	To remove PACKAGE	
apt update	update all packages	
apt update PACKAGE	update only a package	

apt grouplist	List all available Group Packages	
apt groupinstall "GROUPNAME"	Installs all the packages in a group.	
apt repolist	List Enabled apt Repositories	
apt clean all	Clean apt Cache	
apt history	View History of apt	
apt show package name	Shows the information of package like version, size, source, repository etc	

9. SEARCH

```
$ grep pattern files  
(you will this command  
often)  
$ grep-r pattern dir  
pattern in dir  
  
$ locate  
file  
$ find /home/tom -name  
'index*' with "index"(10 find  
examples)  
$ find /home -size  
+10000k 10000k in /home  
  
# Search for pattern in  
files  
  
# Search recursively  
for  
  
# Find all instances of  
file  
# Find files names that  
start  
  
# Find files larger  
than
```

10. LOGIN (SSH AND TELNET)

```
$ ssh user@host  
(secure data communication  
command)  
$ ssh -p port  
user@host specific  
port  
$ telnet host  
using telnet  
port  
  
# Connect to host as  
user  
  
# Connect to host  
using  
  
# Connect to the  
system
```

11. FILE TRANSFER

```
scp  
  
$ scp file.txt  
file.txt to remote host /tmp folder  
  
# Secure  
  
$ scp nixsavy@server2:/www/*.html /www/tmp # Copy *.html  
files from remote host to current system /www/tmp folder  
  
$ scp -r nixsavy@server2:/www /www/tmp  
and folders recursively from remote server to the current  
system  
/www/tmp folder  
  
rsync  
$ rsync -a /home/apps  
/backup/ source to  
destination  
$ rsync -avz /home/apps  
Synchronize files/directories between the local and remote  
system with compression enabled  
  
#  
# Synchronize  
#
```

12. DISK USAGE

```
$ df  
filesystems(commonly used  
command)  
$ df -i  
filesystems  
$ fdisk -l  
types(fdisk command  
output)  
$ du  
readable form (command  
variations)  
$ du -sh  
current  
directory  
$ # Display target mount point  
all filesystem (refer type,list,evaluate output)  
  
$ mount device-path mount-point # Mount a  
device
```

13. DIRECTORY TRAVERSE

```
$ cd .. # To go up one level of the directory tree(simple &  
most needed)  
  
$ # Go to $HOME  
c directory  
$ cd # Change to /test  
/test directory
```

14. SERVICES

Centos8

\$ sudo systemctl start httpd	# Starts httpd on centos
\$ sudo systemctl stop httpd	# Stops httpd on centos
\$ sudo systemctl restart httpd	# Restart services
\$ sudo systemctl status httpd	# shows the current status
\$ sudo systemctl reload httpd	# Reload conf
\$ sudo systemctl enable httpd	# starts httpd at boot time
\$ sudo systemctl disable httpd	# stops httpd at boot time

<code>\$ sudo systemctl is-active httpd</code>	<i># shows whether the service is active or not</i>
<code>\$ sudo systemctl is-enabled httpd</code>	<i># shows whether the service is enabled or not</i>

Ubuntu20

<code>\$ sudo systemctl start apache2</code>	<i># Starts apache2 on ubuntu</i>
<code>\$ sudo systemctl stop apache2</code>	<i># Stops apache2 on ubuntu</i>
<code>\$ sudo systemctl restart apache2</code>	<i># Restart service</i>
<code>\$ sudo systemctl reload apache2</code>	<i># Reload conf</i>
<code>\$ sudo systemctl enable apache2</code>	<i># starts apache2 at boot time</i>
<code>\$ sudo systemctl disable apache2</code>	<i># stops apache2 at boot time</i>
<code>\$ sudo systemctl is-active apache2</code>	<i># Shows whether the service is active or not</i>
<code>\$ sudo systemctl is-enabled apache2</code>	<i># Shows whether the service is enabled or not</i>

15. COMPRESSION / ARCHIVES

```
$ tar cf home.tar home # Create tar named home.tar containing
home/ (11 tar examples)

$ tar xf
file.tar
file.tar
$ tar czf
file.tar.gz
compression
$ gzip file
to file.gz (untar gzip
file)          fi      le      s      # Extract the files
from           # Create a tar with
gzip           # Compress file and renames
it
```

Compression

1. tar cf file.tar file Create tar named file.tar containing file
2. tar xf file.tar Extract the files from file.tar
3. tar czf file.tar.gz files Create a tar with Gzip compression
4. tar xzf file.tar.gz Extract a tar using Gzip
5. tar cjf file.tar.bz2 Create tar with Bzip2 compression
6. tar xjf file.tar.bz2 Extract a tar using Bzip2
7. gzip file Compresses file and renames it to file.gz
8. gzip -d file.gz Decompresses file.gz back to file

16. PROCESS RELATED

```
$ processes (many parameters to learn)
$ ps aux | grep 'telnet' telnet
process
$ pmap (kernel,user memory etc)
$ top (30 examples)
$ kill pid pid id (types of signals)
$ killall proc
$ pkill processname its name
$ # Resumes suspended jobs bringing them to foreground (bg and fg command)
$ fg foregrou nd
$ fg n
```

Display your currently active

Find all process id related to

Memory map of process

Display all running processes

Kill process with mentioned

Kill all processes named proc

Send signal to a process with

Resumes suspended jobs bringing them to foreground (bg and fg command)

Brings the most recent job to

Brings job n to the foreground

17. SYSTEM

\$ uname -a	=> Display linux system
informati	
\$ uname	=> Display kernel
information (refer uname command in detail)	
\$ cat /etc/redhat_release	=> Show which version of redhat
installed	
\$ uptime	=> Show how long system running +
load (learn uptime command)	
\$ hostname	=> Show system host name
\$ hostname -i	=> Display the IP address of the
host (all options hostname)	
\$ last reboot	=> Show system reboot history
(more examples last command)	
\$ date	=> Show the current date and time
(options of date command)	
\$ cal	=> Show this month calendar (what
more in cal)	
\$ w	=> Display who is online (learn
more about w command)	
\$ whoami	=> Who you are logged in as
(example + screenshots)	
\$ finger user	=> Display information about user (many options of finger command)

18. HARDWARE

\$ messages (dmesg many more options)	=> Detected hardware and boot
\$ cat /proc/cpuinfo	=> CPU model
\$ cat /proc/meminfo	=> Hardware memory
\$ cat /proc/interrupts per CPU per I/O device	=> Lists the number of interrupts
\$	=> Displays information on
hardware configuration of the system	
\$	=> Displays block device
information in Linux (sudo yum installed util-linux-nq)	
\$ free -m	=> Used and free memory (-m for
MB) (free command in detail)	

```
$ lspci -tv  
to find vendor  
ids)  
$ lsusb -tv  
lsusb  
options)  
$ lshal  
their  
properties  
$ dmidecode  
BIOS (vendor  
details)  
$ hdparm -i  
/dev/sda  
# Show info about disk  
# sda  
$ hdparm -Tt  
/dev/sda  
# Do a read speed test on disk  
# sda  
$ badblocks -s  
/dev/sda  
# Test for unreadable blocks  
on  
  
disk  
sda
```

19. STATISTICS

```
$ top => Display and update the top cpu processes (30 example  
options)  
  
$ mpstat 1 => Display processors related statistics (learn  
mpstat command)  
  
$ vmstat => Display virtual  
statistics (very useful performance  
tool)  
$ iostat 2 => Display I/O statistics  
Intervals) (more  
examples)  
$ tail -n 500 => Last 10  
messages,(everyday use tail  
options)  
$ tcndump -i interface eth1 (useful to sort network  
issue)  
$ tcpdump -i eth0 'port  
80'  
80 ( HTTP )  
$ => Capture all packets flows  
to all active processes.(sysadmin favorite...  
command)  
$ lsof -u  
testuser user => List files opened by  
specific  
  
$ free -m => Show amount of RAM  
usage  
command)  
  
$ watch df -h => Watch changeable data  
linux continuously(interesting  
command)
```

1. ping host Ping host and output result
 3. dig domain Get DNS information for domain
 4. dig -x host Reverse lookup host
 5. wget file Download file
 6. wget -c file Continue a stopped **download**
 1. date Show the current date and time
 2. cal Show this month's calendar
 3. uptime Show current uptime
 4. w Display who is on line
 5. whoami Who you are logged in as
- Unix/Linux Command Reference
6. finger user Display information about user
 7. uname -a Show kernel information
 8. cat /proc/cpuinfo Cpu information
 9. cat proc/meminfo Memory information
 10. man command Show the manual for command
 11. df Show the disk usage
 12. du Show directory space usage
 13. free Show memory and swap usage
 14. whereis app Show possible locations of app
 15. which app Show which applications will be run by default