Working with Basic Commands

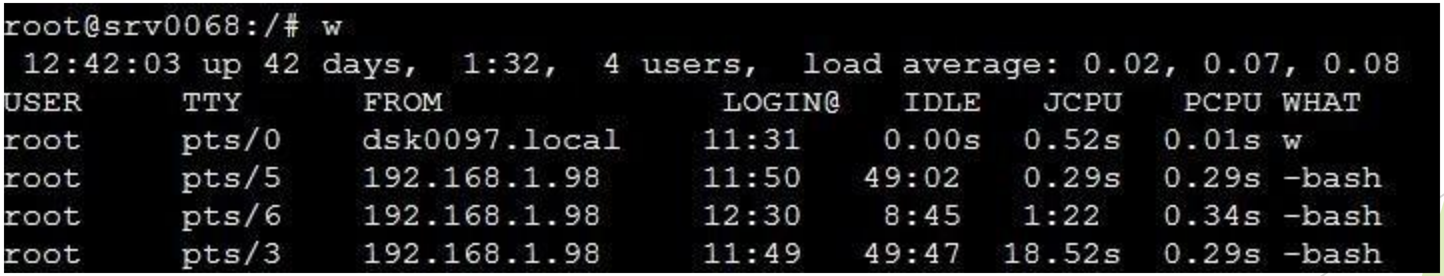
history- prints all the previously typed commands

who – prints the name of all users who have currently logged

in who am i- prints the name and details of the current user

whoami- prints the name of the current user

w- prints full information of all the users who have currently logged in



```
root@srv0068:/# w
 12:42:03 up 42 days,  1:32,  4 users,  load average: 0.02, 0.07, 0.08
USER     TTY      FROM              LOGIN@   IDLE   JCPU   PCPU WHAT
root     pts/0    dsk0097.local    11:31    0.00s  0.52s  0.01s w
root     pts/5    192.168.1.98     11:50   49:02   0.29s  0.29s -bash
root     pts/6    192.168.1.98     12:30    8:45   1:22   0.34s -bash
root     pts/3    192.168.1.98     11:49   49:47  18.52s  0.29s -bash
```

Working with Basic Commands

free-prints the size and usage of the ram and virtual memory

du- Summarize the disk usage of each file, recursively for directories Options:

-s –displays only the total sum for the each specified file/directory

-h –displays in human readable format

df- displays free disk space available for each mount we have Options:

-h –displays in human readable format

-i –list of inode information

-a –includes dummy file system

Working with Basic Commands

stat –displays file system/file status
tree — lists the contents of directories in a tree like
format Options:

-a – all files including hidden files are printed.

-d – list of directories only

-f – prints the full path prefix for each file

-u – print the username along with the tree

-s – size of the file

exit – allows to exit from program, shell or come out of a linux
network logout – just to come out of a linux network

Getting Help on Commands

help – it provides information about the command including options but not elaborately

syn: <command_name> --help

whatis – it provides the information about the command with out the sub commands(options)

syn: whatis <commad_name>

info –it provides the information of all the commands

man --manual, providing all information of the command

syn: man <command_name>

Getting Help on Commands

Contd…

whereis –locates a binary,source,and manual page for a command

syn: whereis <comman_name>

which – locate a command

syn: which <command_name>

File Permissions

Permissions – In Linux, each and every file/directory will have permissions. we have three different kind of permissions for each file/dir.

- read	-r
- write	-w
-execute	-x

- no permission - -

These permissions are having the numerical values assigned like:

read – 4 write – 2 execute – 1 no permission - 0

We have three different kinds of users for each file/dir and each user will have three different permissions

- owner	-u
- group	-g
-others	-o

owner –means who created the file.

group – means no. of users belongs to that group

FilePermissions

To change the file permissions or changing the ownership and groups of file/dir.

We use the following commands:

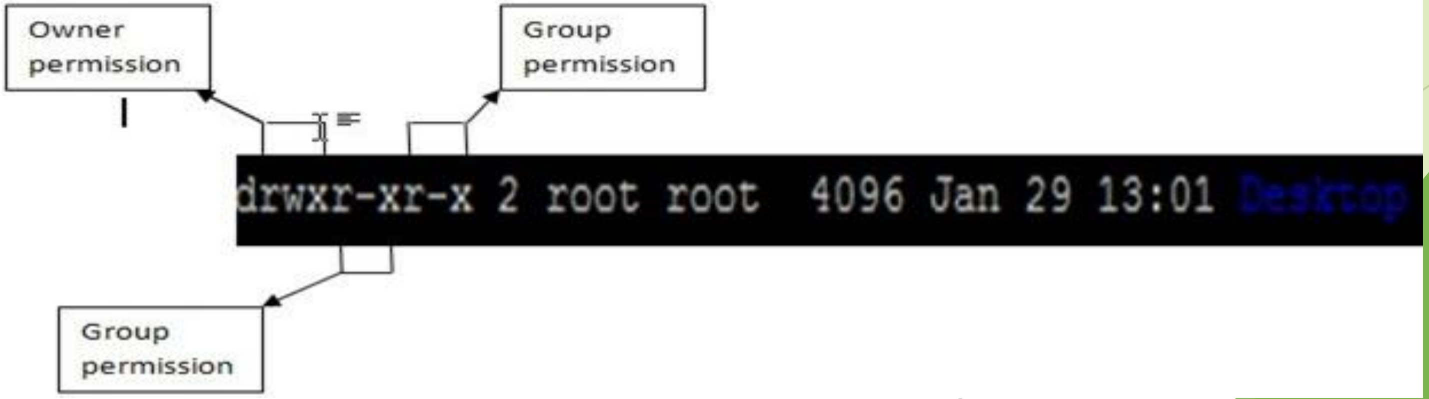
chmod – To change the file/dir permissions

chown – To change the file/dir ownership

chgrp – To change the file/dir group.

- To know the permissions of a file/dir use the command : ls –l

In the o/p of ls –l command first column will have the permissions information.



```
drwxr-xr-x 2 root root  4096 Jan 29 13:01 Desktop
```

File Permissions

chmod – To Change the file/dir permissions using the abbreviations..

```
syn: chmod [options] <perms> <file/dir>
```

- To add write permissions to the group

```
chmod g+w file1    ( + is to add the permissions )
```

- To add read and execute permissions to the others

```
chmod o+rx file1
```

- To remove the write permissions for group and others

```
chmod go-w file1   ( - is to remove the permissions )
```

- To add execute permissions to all the users and remove the write permissions to others

```
chmod ugo+x,o-w file1
```

Options –

-R - This option will be used to when you are changing permissions to directories to recursively apply permissions all the files/subdir in the dir.

```
chmod –R g+w dir1
```

FilePermissions

- To change the file/dir permissions based on the numerical values.

- Each user will have all the three kinds of permissions read, write, execute order, like rwx. For example, if anybody don't have specific permission it contains – in that place… like r-x, means no write permissions.

- To calculate the numerical value of the each kind user's take the some of their permissions numerical values….

  For example, file1 contains the permissions like : rw-r-xr—
  For owner – 4+2+0 – 6

  For Group - 4+0+1 – 5

  For Others – 4+0+0 - 4

  So existing permissions for file 1 is 654

- To Change the file permissions to the like : rwxrw-r-x ; numerical value is 765

  chmod 765 file1.

- Using –R options same thing can be applicable for even directories also.

File Permissions

- if you want to have the same file permissions what other file is having. chmod --reference file1 file2

  - whatever the file1 permissions you will get same as the file2.

- This chmod command can be executed to change the file/directory permissions only by administrator or owner of the file.. No other users will have the permissions to change even if they have the write permissions to do that.

- Maximum permission what we can set is 777 and minimum permissions can set is 000

Permission	Value
---	0
--x	1
-w-	2
-wx	3
r--	4
r-x	5
rw-	6
rwx	7

FilePermissions

chown - To Change the ownership of file/dir permissions

chown user2 file1

-To change ownership of file1 as user2

chown –R user2 dir1 (For directories)

chgrp – To Change the group of file/dir permissions

chgrp grp2 file1

-To change group of file1 as grp2 chown –

R grp2 dir1 (For directories)

-To Change the ownership and group of a file or directory at a time

chown <new username>:<new grpname> file/dir

chown user2:grp2 file1

- Only admin can execute these two commands, no other user cannot execute these commands.

Run Levels

- Running the system in different states.

We have totally 7 run levels

1 - Shutdown

2 - Single User Mode (no networking only root can login, if it is physical)

3 - Mutiluser mode + few networking services

4 - Mutiluser mode + all networking services

5 - Not Used (But operating as 3)

6 - Runlevel 3 + GUI (GUI will be available only on runlevel)

7 - Reboot

- How to switch each run level

```
init <run level>
```

```
init 0
init 1
```

Run Levels

-Each run level will have an associated directory with list of services

/etc/rc0.d

/etc/rc1.d

/etc/rc2.d

/etc/rc3.d

/etc/rc4.d

/etc/rc5.d

/etc/rc6.d

-If you go into the directories some of the files(services) will be starting with "S" and some will be starting with "K"

-if any file/service started with the S -- that service will run in that run level

if any file/service started with the K -- that service will stop in that run level

Run Levels

-All these run level files/services are linked files to the /etc/init.d files.

-/etc/init.d directory contains all the services which are available/provided by the redhat by default.

-Run level services will start/stop the respective services automatically.

-if you want to stop/start/status of any service there are two ways: service <servicename> start/stop/status/restart

-How to change default init level: syn:
id:<init_level>:initdefault:

Ex: id:3:initdefault:

How to shutdown and reboot the machine

--For Shut downing halt

shutdown

poweroff

init 0

-- For restarting/rebooting reboot

shutdown -r

init 6

--shutdown -y now (Shut down will start immediately)

--shutdown -y 1 (Shut down will start in 1 min)

--shutdown -y 5 (Shut down will start in 5 mins)