-- *cd /* => / means root
cd /home/.....

-- *cd* .. => go backwards to parent folder

-- *pwd* => to find current working directory

-- instead of cd /home/darsh/documents/...
So instead of this big command to go to darsh(user) folder we can use
cd ~

-- *ls* => list contents in current directory

-- *ls -l* or *ll* => detailed listing (data of creation, owner, size, type of file,etc)

-- *less <filename>* => root command to get details of that file. But no use I suppose

-- *ls -lah => a* - will list all hidden folders also
h - to get human readable format of file sizes (eg. 4K)

-- *ls --help* => manual for that command

-- *ls -R =>* list subdirectories recursively. So first it will show detail listing of /darsh, then it will show detail listing of /darsh/documents, then /darsh/Documents/images ..... and so on

-- *man* => manual

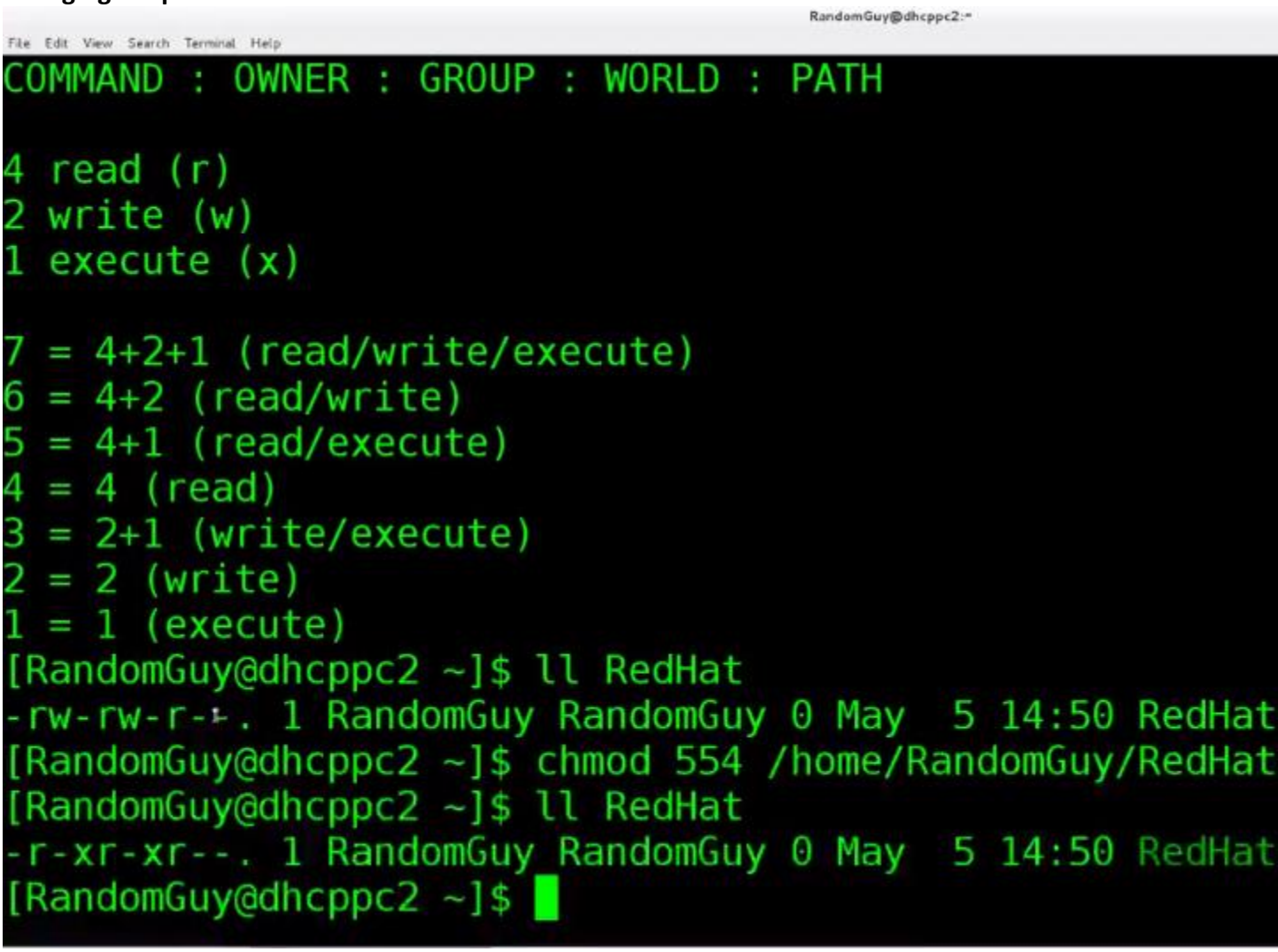
-- *touch <file> =>* Creation of a file
touch /Documents/.... <file> => giving the path of directory to create a file

-- *find /home/ -name "File_01" =>* it gives path where File_01. If not found command returns nothing
Now if I write "file_01" it will return nothing, remember linux is case-sensitive. So what I can use is
find /home/ -iname "File_01"

-- *locate File_01 =>*
sudo -s
updatedb -- you will need to do this before locating file

-- To go to folder and search and if you use :
locate /home File_01 -> it will show the whole list where it tried to search. This is not required
During such situations use *'locate --help".* We found that this is the addition to be done inorder to restrict the output to required location where File_01 is there.

Changing file permissions and CAT command :



CAT command can be used for :

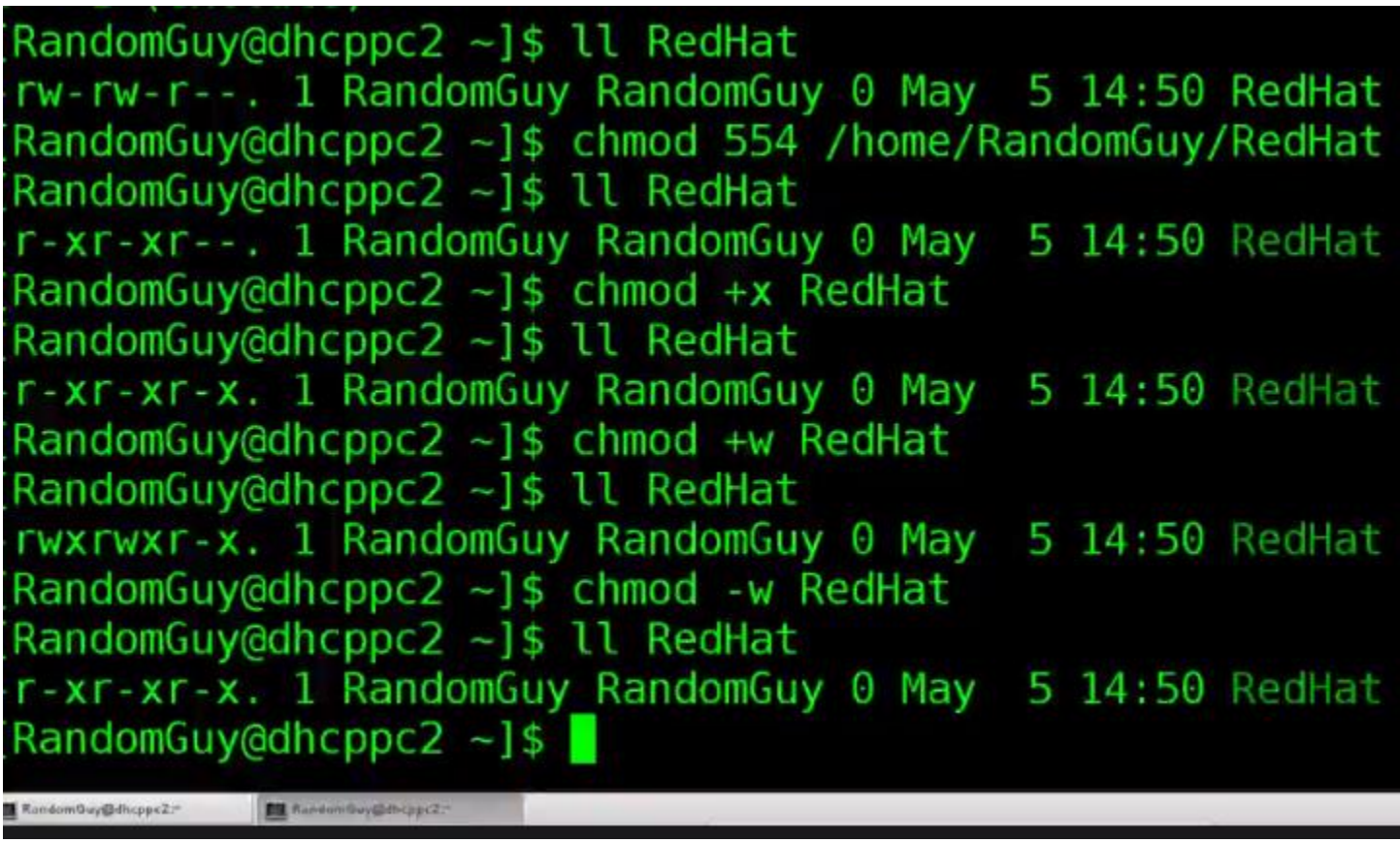
- Display text files on screen.
- Copy text files.
- Combine text files.
- Create new text files.

cat /etc/password => The above command will display the contents of a file named /etc/passwd.

cat /etc/passwd > /tmp/test/txt => In the above example, the output from cat command is written to /tmp/test.txt file instead of being displayed on the monitor screen.

cat /etc/hosts /etc/resolve.conf /etc/dp => It concatenates three mentioned files
or *cat /etc/hosts /etc/resolve.conf /etc/dp > /tmp/output.txt*

cat > foo.txt => creates a file named foo.txt. If file already exists, it will be overwritten. To append what you write, we can use :
cat >> foo.txt



If you are in one folder and want to change permissions in other folder, then you can use the following command :
chmod -R +x /home/Documents
or *chmod -R 555 /home/Documents*

Difference between *chmod and chown :*
In simple term `chown` is used to change the ownership of a file while `chmod` is for changing the file mode bits.

- `chown` defines who owns the file.
- `chmod` defines who can do what.

chown root <file> => we changed the owner of file
or *chown root:root <file>* => we changed owner as well as group that owns this file
check using : ll <filename>

mv,rm,cp :

- rm <directory> -- will not remove directory

use rm --help -- and we can see that -r or -R will help us do that. -f will force something without prompting. So we use

- *rm -rf <directory name>*

- *cp [FROM] [TO]*

use cp --help and we see --> -R , -f, are useful

- We can use mv command to **rename** something like

- *mv File_01 IamHere*
usual movement is done using
- *mv File_01 <path>*

grep, pipe, echo, cat :

- *echo "Hey guys!"* --> we echo'd it out to stdout

- here stdout is the output console or screen where we are able to see. We can echo this to our stdout using
echo "Hey Guys!" > DP
cat DP
Hey Guys

- *echo -e " Hey there\nhow are you\nI feel good" > DP*
-e to allow for escape characters like \n. Look "man echo"

- Imagine there is a file "dp" with 400 lines. We want to find certain thing we use grep : like

- *cat dp | grep feel* --> it will print lines with word feel in it and that word will be highlighted. To ignore case sensitivity we can use the command like this

-cat dp | grep -i Feel

YUM packet manager :

This is the default packet manager of redhat.

Packet manager on any linux distribution is a piece of software which enables you to pull from remote location/repositores certain piece of code that is within those repo. Almost everything is there on the repo. This restricts user to go to insecured internet and download. Repo data is safe and secure. That is one of the main reason why linux is so useful than other operating systems.

yup .deb => ubuntu
yup .rpm => CentOS/ RHEL

useful arguments:
-v verbose operation
-y answer yes for all questions
-q quiet for all operations
--skip-broken skip packages with depsolving problems

useful commands with yum :
clean - remove cached data
deplist - list a package dependencies
downgrade - downgrade a package
install - install packages
info - get information about package
update - update a package

- yum -C repolist --> gives reponame and repo id, etc

yum search gedit => searched packages with that name closely

yum install gedit

yum reinstall gedit

yum install gedit -y =>install gedit without prompting y/N : not recommendable

yum info gedit => information about the package

yum remove gedit => uninstall package

yum search sql | grep -i maria => filter some search results

RPM Packet Manager :

rpm -iv nmap.....rpm=> install this package

rpm -q nmap....rpm=> query the current status

rpm -e nmap.....rpm => erase

rpm --help | grep "\-e" => search results in help with -e in it. Backslash char '\' to include '-' in search