My Bash Cookbook

Nikhil Cherian Kurian, December 2018

1 Basic commands

1.1 cd

To change the directory

Command	Function
cd	go to home directory
cd ..	go to one directory down
cd *MainDirA*	go to directory named *MainDirA*
cd *MainDirA/SubDirA*	go to directory named *SubDirA* in the directory *MainDirA*
cd *../../MainDirB/SubDirB*	go to directory named *SubDirB* in the directory *MainDirB* from *MainDirA/SubDirA*

1.2 rm

To remove or delete a file/directory

Command	Function
rm *fIle*	deletes the file named *File*
rm -i *fIle*	(interative mode) ask for confirmation before deleting the file named *File*
rm -r *Dir*	deletes the directory named *Dir*
rm -ri *Dir*	(interative mode) deletes the directory named *Dir*
rm -f *File*	(force mode) deletes the file *File if exists, otherwse no error is shown*
rm -v *File*	(verbose mode) deletes the file *File and prints what the system did.*

1.3 lsList Files

Command	Function
ls	list file in the current directory
ls -l	list file with all fields
ls -o	list file with some fields

1.4 date

Date gives the current date and time as follows.

date
Mon Dec 9 12:58:02 IST 2019

To get in dd/mm/yyyy format

Date +\%d/\%m/\%y
09/12/2019

To get in dd/mm/yy format

date +\%d/\%m/\%y
09/12/19

To get in dd/mm/yy format (Alternate)

date+\%D
09/12/19

1.5 sort

sort command sorts a file, say file.tex.

sort file.tex

to sort a numeric file

sort -n file.tex

to sort a file in reverse order

sort -r file.tex

to sort a numeric file in everse order

sort -nr file.tex

to sort list of directories in the order of the group (i.e. sorting different fields) The fourth field is the group +3 indicate the fourth field starting with zero. -b indiacte ignore leading blanks

ls-l | sort+3-b

To sort by the size of files

ls -l | sort+4-b -n

1.6 tail

show the tail portion of a file, file.tex

tail file.tex

Show that last three lines of file.tex

tail +3 file.tex

1.7 cmp

cmp command compares two files and says there is difference, but not what is the difference/

cmp file1.tex file2.tex

1.8 diff

diff command compares two files and says what are the differences

diff file1.tex file2.tex

1.9 touch

touch creates an empty file.

touch file.tex

1.10 wc

wc command counts characters, words and lines of a file

wc file.tex

1.11 wildcharacters

Suppose you have files as below:

file1.tex
file2.tex
file3.tex
file.tex
file.c
prog.cpp
file.doc
testfile

Then * indiacte all and ? single character, for example, the following command display all files

ls *

This command will display all file, except the test file

ls *.*

To display all tex file

ls *.tex

To display all .tex files with number 1,2,3 at the end,

ls file?.tex

To display all c files,

ls *.c*

to display all files with file names start with small case alphabets

ls [a-z]*.*

to display all files with file names start with alphabets

ls [A-z]*.*

to display all files with file names start 1,2,3,4 or 5.

ls [1-5]*.*

to display all files with file names start with alphabet or digit

ls [0-9,A-z]*.*

to display all files with files that does not start with digit

ls [!0-9]*.*

to display all files with files that does not start with digit, but second character is digit

ls [!0-9][0-9]*.*

1.12 grep

To check whether is word or character is in a file. Eg. to see whaere all the word 'display' finds in file 'linux.tex'

grep display linux.tex

Eg. to see whaere all the word 'display all fil' finds in file 'linux.tex'

grep 'display all fil' linux.tex

1.13 File redirection

The output of many commands can be rdirected to files. To direct the out put of ls -l to a file called file.dat. If file.dat is not existing, it will create, otherwise it will overwrite.

ls -l >file.dat

To direct the out put of ls -l to a file called file.dat by appending the file.

ls -l >> file.dat

1.14 Piping

The output of a command can be fed to another command. To find how many files are there in this direcotriy.

ls | wc -l

Another example. To see whether user 'sahyadri' is logged in

who | grep sahyadri

The piping can be nested. If you want to see the latest 10 files and sort them

ls -t | tail +10| sort

2. Security Commands:

Command	Function
d	file status d = direcotry, l = link, - = normal file
r	read permission for owner - = no permission
w	write permission for owner
x	execute permission for owner
r	read permission for group
w	write permission for group
x	execute permission for group
r	read permission for public
w	write permission for public
x	execute permission for public

chmode 777 *File*	*read write execute permission for owner, group and public*

There is a alternate way of doing this.

chmode u+x file.tex	permits user to have execute permission
chmode g+x file.tex	permits group to have execute permission
chmode o+x file.tex	permits others to have execute permission
chmode a+x file.tex	permits all to have execute permission
chmode u+r file.tex	permits user to have read permission
chmode g+r file.tex	permits group to have read permission
chmode o+r file.tex	permits others to have read permission
chmode a+r file.tex	permits all to have read permission
chmode u+w file.tex	permits user to have write permission
chmode g+w file.tex	permits group to have write permission
chmode o+w file.tex	permits others to have write permission
chmode a+w file.tex	permits all to have write permission
chmode ugo+rwx file.tex	permits all user to have read-execute and write permissions
chmode a+rwx file.tex	same as above (permits all user to have read, execute and write permissions)
chmode a-rwx file.tex	deny all user to read, execute and write permissions

chmode a+rwx direcotory	permits all user to have read, execute and write permissions of the directory
chmode -R a+rwx direcotory	permits all user to have read, execute and write permissions of the directory as well as all the files inside that directory