CS303 Operating System Lab-1

Submitted by : Parth Goyal

(2017csb1095)

**Q1. Exploring Unix commands.**

1. **pwd** : The pwd command prints the full system path of the current working directory.
   1. -L, --logical : use PWD from environment, even if it contains symlinks (To avoid symlinks pass the -P option)
   2. -P, --physical : avoid all symlinks

*SYNOPSIS : $ pwd [OPTION] ...*

Example : $ pwd

/home/2017csb1095

1. **cd** : The cd command, also known as chdir(change directory), used to change the current working directory.
   1. $ cd [directory\_name] : if directory is available in the current working directory, it makes this new directory as working directory.

Examples : $ pwd

/home/2017csb1095

$ cd Desktop

/home/2017csb1095/Desktop

1. **cp** : A command line utility for copying files and directories. It supports moving one or more files or folders with options for taking backups and preserving attributes.
   1. --backup : make a backup of each existing destination file
   2. --attributes-only : don’t copy the file data, just the attributes.
   3. --parents : use full source file name under DIRECTORY
   4. --verbose : explain what is being done

*SYNOPSIS : cp [OPTION]... [-T] SOURCE DEST*

Examples : $ ls

File1.txt File2.txt

$ cp --verbose File1.txt File2.txt

'File1.txt' -> 'File2.txt'

1. **ls** : List directory contents, lists information about the FILES (the current directory by default). Sort entries alphabetically.
   1. -a, --all : do not ignore entries starting with .
   2. --author : with -l, print the author of each file
   3. -C : list entries by columns
   4. -d, --directory : list directories themselves, not their contents
   5. -l : use a long listing format
   6. -i : print the index number of each file

*SYNOPSIS : ls [OPTION]... [FILE]...*

Example : $ ls

File1.txt File2.txt

$ ls -l

total 8

56492468 -rw-r--r-- 1 2017csb1095 phdstudent 74 Aug 16 09:45 File1.txt

56492284 -rw-r--r-- 1 2017csb1095 phdstudent 74 Aug 16 09:47 File2.txt

1. **mkdir** : The mkdir command allows user to create directories or folders as they are referred to. It can create multiple directories at once and also set permissions when creating the directory.
   1. -m, --mode=MODE : set file mode
   2. -p, --parents : no error is existing, make parent directories as needed
   3. -v, --verbose : print a message for each created directory

*SYNOPSIS : mkdir [OPTION]... DIRECTORY…*

Example : $ mkdir --verbose -p parent/daughter

mkdir: created directory 'parent'

mkdir: created directory 'parent/daughter'

1. **rm** : remove files or directories. It removes each specified file. By default, it does not remove directories.
   1. -f, --force : ignore nonexistent files and arguments
   2. -i : prompt before every removal
   3. --verbose : explain what is being done

*SYNOPSIS : rm [OPTION]... [FILE]...*

Example : $ ls

File1.txt File2.txt parent

$ rm --verbose -i File1.txt

removed 'File1.txt'

$ ls

File2.txt parent

1. **chmod** : change file mode bits
   1. -c, --changes : like verbose but report only when a change is made
   2. -f, --silent, --quiet : suppress most error messages
   3. --no-preserve-root : do not treat ‘/’ specially

*SYNOPSIS : chmod [OPTION]... MODE[.. , MODE]... FILE…*

Example : $ ls -l

-rw-r--r-- 1 2017csb1095 phdstudent 74 Aug 16 09:47 File2.txt

$ chmod u=rw File2.txt --verbose

mode of 'File2.txt' changed from 0444 (r--r--r--) to 0644 (rw-r--r--)

1. **gzip** : It is a compressing tool used to truncate the file size. Use gunzip command to decompress a file.
   1. -a, --ascii : convert end-of-lines using local conventions.
   2. -d, --decompress
   3. -f, --force : force compression or decompression
   4. -k : keep the original file
   5. *SYNOPSIS : gzip [ -acdfhklLnNrtvV19 ] [--rsyncable] [-S suffix] [ name ... ]*
   6. Example : $ ls

File2.txt parent

$ gzip -k File2.txt --verbose

File2.txt: 10.8% -- replaced with File2.txt.gz

$ ls

File2.txt File2.txt.gz parent

1. **find** : search for files in a directory hierarchy
   1. -name : specify the name of the file/folder

*SYNOPSIS : find [options] [path] [expression]*

Example : $ find ./ -name daughter

./Desktop/parent/daughter

./.local/share/Trash/files/parent/daughter

10) **less** : less command doesn’t load entire file but loads it by part-by-part

a) –help

b) --version

*SYNOPSIS : less [-[+]aABcCdeEfFgGiIJKLmMnNqQrRsSuUVwWX~]*

[-b space] [-h lines] [-j line] [-k keyfile]

[-{oO} logfile] [-p pattern] [-P prompt] [-t tag]

[-T tagsfile] [-x tab,...] [-y lines] [-[z] lines]

[-# shift] [+[+]cmd] [--] [filename]...

11) **tail** : output the last part of files

a) -n, --lines : output the last NUM files, instead of last 10

b) -q, --quiet : never output headers giving file names

c) -c, --bytes : output the last NUM bytes

*SYNOPSIS : tail [OPTION]... [FILE]...*

Example : $ tail File2.txt

Hello, I’m Parth Goyal.

$ tail -c 18 File2.txt

I’m Parth Goyal

12) **top** : Display Linux Processes

a) -u : to specify user

*SYNOPSIS : top -hv|-bcHiOSs -d secs -n max -u|U user -p pid -o fld -w [cols]*

Example : $ top

1048 parthgo+ 20 0 635740 71780 56912 S 2.0 0.9 0:40.91 Xorg

1289 parthgo+ 20 0 3749292 292356 105252 S 2.0 3.7 0:41.91 gnome-shell

1822 parthgo+ 20 0 1531080 204012 132508 S 1.0 2.6 0:42.69 soffice.bin

1988 parthgo+ 20 0 763236 38584 27412 S 1.0 0.5 0:07.79 gnome-term+

13) **wc** : print newline, word, and byte counts for each file

a) -c : print the byte counts

b) -m : character count

c) -l : lines count

*SYNOPSIS : wc [OPTION]... [FILE]...*

Example : $ wc File2.txt

1 4 24 File2.txt

14) **diff** : compare files line by line

a) –normal : output a normal diff

b) -q : report a brief detail

*SYNOPSIS : diff [OPTION]... FILES*

Example : $ diff -q File2.txt File1.txt

Files File2.txt and File1.txt differ

15) **kill** : send a signal to a process

a) -l : list signal names

b) -L : list signal names in a nice table

*SYNOPSIS : kill [options] <pid> [...]*

Example : $ kill 1048

16) **ifconfig** : interface configuration

*SYNOPSIS : $ ifconfig*

17) **shutdown** : Halt, power-off or reboot the machine

a) -r : reboot

b) -k : do not halt

c) -c : cancel pending shutdowns

*SYNOPSIS : shutdown [OPTIONS...] [TIME] [WALL...]*

Example : $ shutdown -r

18) **which** : locate a command

a) -a : print all matching pathnames of each argument

*SYNOPSIS : which [-a] filename ...*

Example : $ which python

/home/2017csb1095/anaconda3/envs/envn/bin/python

19) **cat** : concatenate files and print on the standard output

a) -A : show all

b) -n : number all output lines

*SYNOPSIS : cat [OPTION]... [FILE]...*

Example : $ cat File1.txt File2.txt

Hello, I'm Parth.

Hello, I'm Parth Goyal.

20) **mv** : move (rename) files

a) -b : make a backup

b) -n : do not overwrite an existing file

*SYNOPSIS : mv [OPTION]... SOURCE... DIRECTORY*

Example : $ mv File1.txt File3.txt --verbose

renamed 'File1.txt' -> 'File3.txt'

21) **grep** : global regex print

a) -f : obtain patterns from FILE, one per line

b) -c : count of matching lines for each input file

*SYNOPSIS : grep [OPTIONS] PATTERN [FILE...]*

Example : $ grep Parth File2.txt

Hello, I'm **Parth** Goyal.

22) **cut** : print selected parts of lines from each file to standard output

*SYNOPSIS : cut [OPTION]... SOURCE... DIRECTORY*

Example : $ cut File2.txt -c 1

23) **history** : GNU history library. It is able to keep track of lines read as input

a) -w : write history to a file

b) -c : clear the history

*SYNOPSIS : history*

Example : $ history

1993 cut File2.txt -c l

1994 cut File2.txt -c -l

1995 cut File2.txt -c

1996 cut --help

1997 cut File2.txt -c 1

1998 cut File2.txt -c 10

1999 cut File2.txt -c 10 --verbose

2000 cut File2.txt -c 10 -v

2001 kill -l

2002 man history

2003 history

24) **echo** : Echo is used to output a string of characters to the stdout

a) -n : do not print the trailing new line

b) -e : enable interpretation of backslash escapes.

*SYNOPSIS : echo [SHORT-OPTION]... [STRING]...*

Example : $ echo OS LAB1

OS LAB1

**Q2. Explore vi Unix Editor.**