



Department of Computer Science and Engineering (Data Science)

A.Y.: 2022-23

Sub: System Fundamentals

**Experiment 4
(Linux Commands)**

Name – DIVYESH KHUNT

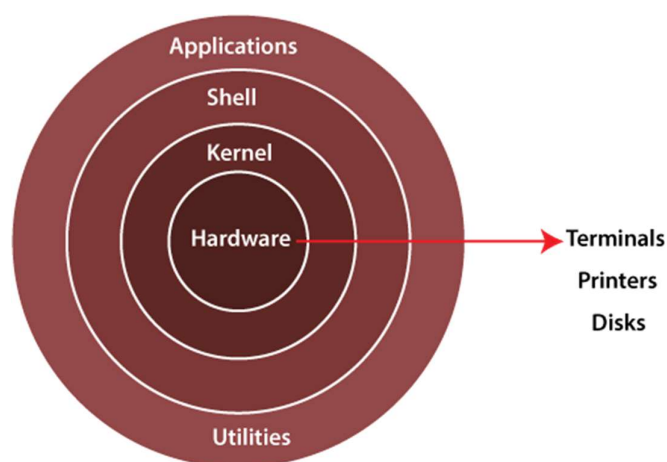
SAP – 60009210116

Aim: Explore the internal commands of Linux

Theory:

Linux operating system: An operating system can be described as an interface among the computer hardware and the user of any computer. It is a group of software that handles the resources of the computer hardware and facilitates basic services for computer programs.

Architecture of Linux system



Kernel:- The kernel is one of the core section of an operating system. It is responsible for each of the major actions of the Linux OS. The kernel facilitates required abstraction for hiding details of low-level hardware or application programs to the system. There are some of the important kernel types which are mentioned below:

- Monolithic Kernel
- Micro kernels
- Exo kernels
- Hybrid kernels

Shell:- The shell is the Linux command line interpreter. It is an interface among the kernel and user. It can take commands through the user and runs the functions of the kernel.

Linux Commands

A Linux command is a program or utility that runs on the command line. A command line is an interface that accepts lines of text and processes them into instructions for computer.

(I) File and Directory Related commands

1) pwd

This command prints the current working directory

**Department of Computer Science and Engineering (Data Science)**

A.Y.: 2022-23

Sub: System Fundamentals

```
cseeds-student@cseedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ pwd
/home/cseeds-student
```

2) ls

This command displays the list of files in the current working directory.

\$ls -l Lists the files in the long format

\$ls -t Lists in the order of last modification time

\$ls -d Lists directory instead of contents

\$ls -u Lists in order of last access time

```
cseeds-student@cseedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ ls
Anonym          hadoop-3.3.4      snap              Untitled7.ipynb
Anonym.c         hadoop_store      Templates         Untitled8.ipynb
Anonym.o         input            TURBOC3          'Untitled Document 1'
bar.parquet      'MapReduce Program'  units            Untitled.ipynb
Desktop          MapReduceTutorial  Untitled1.ipynb  Videos
desktop.ipynb    minikube-linux-amd64  Untitled2.ipynb  WordCountExample
Documents        Music              Untitled3.ipynb  zoo.orc
Downloads        Pictures            Untitled4.ipynb
foo.csv          Public              Untitled5.ipynb
grep_example     'Sachin 60009210052'  Untitled6.ipynb
```

3)**cd**

This command is used to change from the working directory to any other directory specified.

\$cd directoryname

```
cseeds-student@cseedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ cd
```

4) cd ..

This command is used to come out of the current working directory.

\$cd ..

5) mkdir

This command helps us to make a directory.

\$mkdir directoryname

```
cseeds-student@cseedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ mkdir
mkdir: missing operand
Try 'mkdir --help' for more information.
```

6) rmdir

This command is used to remove a directory specified in the command line. It requires the specified directory to be empty before removing it.

\$rmdir directoryname

```
cseeds-student@cseedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ mkdir amitesh
cseeds-student@cseedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ rmdir amitesh
```

7) cat

This command helps us to list the contents of a file we specify.

\$cat [option][file]

cat > filename – This is used to create a new file.

cat >>filename – This is used to append the contents of the file

**Department of Computer Science and Engineering (Data Science)**

A.Y.: 2022-23

Sub: System Fundamentals

Eg:

cat file1

cat file1 file2 > all

cat file1 >> file2

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ cat>file1
This is a new file.
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ cat>file2
This is second file.
```

8)

cp

This command helps us to create duplicate copies of ordinary files.

\$cp source destination

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ cp file1 file2
```

9) **mv**

This command is used to move files.

\$mv source destination

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ mv file1 file2
```

10) **ln**

This command is to establish an additional filename for the same ordinary file.

\$ln filename secondname

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ ln file2 myfile
```

11) **rm**

This command is used to delete one or more files from the directory.

\$rm [option] filename

\$rm -i Asks the user if he wants to delete the file mentioned.

\$rm -r Recursively delete the entire contents of the directory as well as the directory itself.

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ rm myfile
```

(II) Process and status information commands1) **who**

This command gives the details of who all have logged in to the UNIX system currently.

\$ who

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ who
cseds-student tty2          2023-03-31 07:35 (tty2)
```

2) **who am i**

This command tells us as to when we had logged in and the system's name for the connection being used.

\$who am i

8

**Department of Computer Science and Engineering (Data Science)**

A.Y.: 2022-23

Sub: System Fundamentals

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ whoami
cseds-student
```

3) date

This command displays the current date in different formats.

+%D mm/dd/yy +%w Day of the week

+%H Hr-00 to 23 +%a Abbr.Weekday

+%M Min-00 to 59 +%h Abbr.Month

+%S Sec-00 to 59 +%r Time in AM/PM

+%T HH:MM:SS +%y Last two digits of the year

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ date
Friday 31 March 2023 08:09:09 AM IST
```

4) echo

This command will display the text typed from the keyboard.

\$echo

Eg: \$echo Have a nice day

O/p Have a nice day

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ echo hello world
hello world
```

(III) Text related commands**1. head**

This command displays the initial part of the file. By default it displays first ten lines of the file.

\$head [-count] [filename]

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ head file1
This is a new file1cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$
```

2. tail

This command displays the later part of the file. By default it displays last ten lines of the file.

\$tail [-count] [filename]

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ tail file1
This is a new file1This is the second and last line
```

3.**wc**

This command is used to count the number of lines, words or characters in a file.

wc -l <filename> print the line count

wc -c <filename> print the byte count

wc -m <filename> print the character count

wc -L <filename> print the length of longest line

wc -w <filename> print the word count

**Department of Computer Science and Engineering (Data Science)**

A.Y.: 2022-23

Sub: System Fundamentals

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ wc -l file1
1 file1
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ wc -c file1
52 file1
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ wc -m file1
52 file1
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ wc -L file1
51 file1
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ wc -w file1
11 file1
```

4. find

The find command is used to locate files in a directory and in a subdirectory.

The -name option

This lists out the specific files in all directories beginning from the named directory. Wild cards can be used.

The -type option

This option is used to identify whether the name of files specified are ordinary files or directory files. If the name is a directory, then use “-type d” and if it is a file then use “-type f”.

The -mtime option

This option will allow us to find that file which has been modified before or after a specified time. The various options available are -mtime n(on a particular day),-mtime +n(before a particular day),-mtime -n(after a particular day)

The -exec option

This option is used to execute some commands on the files that are found by the find command.

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ find -name file1
./file1
```

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ find . -name "file1" -type f -exec wc -c "{}" \;
52 ./file1
```

(IV) File Permission commands**1) chmod**

Changes the file/directory permission mode:

\$chmod [option] mode files

options:

-R Descend directory arguments recursively while setting modes.

-f Suppress error messages if command fails.

**Department of Computer Science and Engineering (Data Science)**

A.Y.: 2022-23

Sub: System Fundamentals**mode:**

Who u=user, g=group, o=other, a=all (default)

Opcode

+ means add permission

— means remove permission

= means assign permission and remove the permission of unspecified fields

Permission r=Read, w=write, x=Execute

Eg. \$ chmod 777 file1

Gives full permission to owner, group and others

\$ chmod o-w file1

Removes write permission for others.

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-In-One-PC: $ chmod 777 file1
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-In-One-PC: $ chmod o-w file1
```

2) chgrp

chgrp user file Makes file belong to the group user.

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-In-One-PC: $ chgrp cseds-student file1
```

3) chown

chown cliff file Makes cliff the owner of file.

chown -R cliff dir Makes cliff the owner of dir and everything in its directory tree.

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-In-One-PC: $ chown cseds-student file1
```

(V) Other Useful Commands:1) **exit** - Ends your work on the UNIX system.

9

2) Ctrl-l or clear

Clears the screen.

3) Ctrl-c

Stops the program currently running.

4) Ctrl-z

Pauses the currently running program.

5) man COMMAND

Looks up the UNIX command COMMAND in the online manual pages.

6) history

List all commands typed so far.

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-In-One-PC: $ history
1 sudo apt-get purge penjdk-\\*
2 sudo apt-get purge openjdk-\\*
3 clear
4 cd /home/cseds-student/Downloads
5 sudo tar -xvf jdk-19_linux-x64_bin.tar.gz
```

7) more FILE

Display the contents of FILE, pausing after each screenful.

**Department of Computer Science and Engineering (Data Science)**

A.Y.: 2022-23

Sub: System Fundamentals

There are several keys which control the output once a screenful has been printed.

<enter> Will advance the output one line at a time.

<space bar> Will advance the output by another full screenful.

"q" Will quit and return you to the UNIX prompt.

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ more file1
This is a new file
```

8) less FILE

"less" is a program similar to "more", but which allows backward movement in the file as well as forward movement.

9) lpr FILE

To print a UNIX text or PostScript file, type the following command at the system prompt:

(VI) Meta characters

Some special characters, called metacharacters may be used to specify multiple filenames. These characters substitute filenames or parts of filenames.

1) The "*"

This character is used to indicate any character(s)

2) \$ cat ap*

This displays the contents of all files having a name starting with ap followed by any number of characters.

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ cat file1*
This is a new file
```

3) The "?" This character replaces any one character in the filename.

\$ ls ?st

list all files starting with any character followed by st.

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ ls file1
file1
```

4) The [] These are used to specify range of characters.

\$ ls [a-z]pple

Lists all files having names starting with any character from a to z.

(VII) Absolute path and relative path

Generally, if a command is given it will affect only the current working directory. For example, the following command will create a directory named curr in the current working directory.

1) \$ mkdir curr

**Department of Computer Science and Engineering (Data Science)**

A.Y.: 2022-23

Sub: System Fundamentals

The directory can also be created elsewhere in the file system using the absolute and relative path. If the path is given with respect to the root directory, then it is called **full path or absolute path**

```
$ mkdir /home/it2006/it2k601/curr
```

The full path always starts with the /, which represents the root directory. If the path is given with respect to the current working directory or parent directory then it is called **relative path**.

```
$ mkdir ../curr
```

The above command will create a directory named curr in the parent directory.

```
$ mkdir ./first/curr
```

The above command will create a directory named curr inside first directory, where the directory first is located in the current working directory.

Note “.” Represents current directory and “..” represents parent directory.

(VIII) PIPES AND FILTERS

In UNIX commands were created to perform single tasks only. If we want to perform multiple tasks we can go for pipes and filters.

PIPES

A pipe is a mechanism which takes the output of a command as its input for the next command. For example, list the numbers of users currently login in the system and then sort it.

```
ls who | sort
```

```
$ who | wc -l
```

```
$ cat text.c | head -3
```

FILTERS

Filters are used to extract the lines, which contain a specific pattern, to arrange the contents of a file in a sorted order, to replace existing characters with some other characters, etc.

1. Sort filter

The sort filter arranges the input taken from the standard input in alphabetical order. The

sort command when used with “-r” option will display the input taken from the keyboard in the reverse alphabetical order. When used with “-n” option arranges the numbers, alphabets and special characters according to their ASCII value. If we want to sort on any one field, then sort provides us with an option called “+pos1 -pos2” option.

2. Grep filter

This command is used to search for a particular pattern from a file or from standard input and display those lines on the standard output. Grep stands for “Global search for regular expression”.

There are diverse options available with grep command.

-v displays only those lines which do not match the pattern specified.

-c displays only the count of those lines which match the pattern specified

-n displays matched lines with line numbers

**Department of Computer Science and Engineering (Data Science)**

A.Y.: 2022-23

Sub: System Fundamentals

-i displays matched pattern ignoring case distinction

Eg: grep -c tech file1

3.Uniq filter

The uniq filter compares adjacent lines in the sorted input file and when used with different options displays single and multiple occurrences.

-d displays only the lines which are duplicated in the input file.

-u displays only the lines with single occurrences.

4.Pg and more filter

These commands display the output of the command on the screen page by page. The difference between pg and more filter is that the viewing screen of the latter can be done by pressing space bar while that of the former is done by pressing enter.

5.Cut command

One particular field from any file or from output of any command can be extracted and displayed using this cut command. One particular character can also be extracted using the -c option of this command.

6.Tr command

This command is used to translate characters taken from the standard input. This command when used with "-s" option is used to squeeze multiple spaces into a single space.

Additional Commands**rmdir**

This is the opposite of 'mkdir'- which is used to delete the directories. It should be pointed out that in order to use it, the directory must be empty.

'rm' command for removing files in the directory.

who

This is used to find out who's working on this system. Linux is a multiuser system.

For example, if we logged in as 'root' but are working as 'nitin'.

root tty1 May 20 09:48 'root' started working on terminal 1 on May 20 at 9:48

nitin tty2 May 20 10:05 *nitin* started working on terminal 2 at 10:05

whoami

It is a little program that tells us user names.

\$whoami

ps

It gives us a list of the processes running on the system.

\$ps

editing files:

vi filename Edit a file using the vi editor. All UNIX systems will have vi in some form.

emacs filename Edit a file using the emacs editor. Not all systems will have emacs.

man command

Shows manual (description) for given command.

\$man ls



Department of Computer Science and Engineering (Data Science)

A.Y.: 2022-23

Sub: System Fundamentals

Lab Assignments to complete in this session

1. Display top 10 processes in descending order

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ head file1
This is a new file1cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$
```

2. Display processes with highest memory usage.

```
$ top -o %MEM
export "PS1=$ "
top -o %MEM

bash: /usr/bin/top: Permission denied
export "PS1=$ "
```

3. Display current logged in user and logname

```
$ whoami
export "PS1=$ "
whoami

webmaster
export "PS1=$ "

$ logname
logname: no login name
```

4. Display current shell, home directory, operating system type, current path setting, current working directory.

```
$ echo $SHELL
export "PS1=$ "
echo $SHELL

/bin/bash
export "PS1=$ "

$ echo $HOME
export "PS1=$ "
echo $HOME

/home/webmaster
export "PS1=$ "

$ uname -s
Linux
$ echo $PATH
export "PS1=$ "
echo $PATH

/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:./usr/local/icon-v951/bin
export "PS1=$ "

$ pwd
/home/cg/root/6427ef4a707e5
```



Department of Computer Science and Engineering (Data Science)

A.Y.: 2022-23

Sub: System Fundamentals

5. Display OS version, release number, kernel version

```
$ cat /etc/os-release | grep "PRETTY_NAME"
export "PS1=$ "
cat /etc/os-release | grep "PRETTY_NAME"

PRETTY_NAME="Ubuntu 22.04.1 LTS"
export "PS1=$ "

$ cat /etc/os-release | grep "VERSION_ID"
export "PS1=$ "
cat /etc/os-release | grep "VERSION_ID"

VERSION_ID="22.04"
export "PS1=$ "

$ uname -r
5.15.0-60-generic
```



Department of Computer Science and Engineering (Data Science)

A.Y.: 2022-23

Sub: System Fundamentals

6. Write a command to display the first 15 columns from each line in the file

```
$ cut -c 1-15 file1
export "PS1=$ "
cut -c 1-15 file1

This is a new f
export "PS1=$ "
```

7. Cut specified columns from a file and display them

```
$ cut -f 1,3 file1
export "PS1=$ "
cut -f 1,3 file1

This is a new file
export "PS1=$ "
```

8. Sort given file ignoring upper and lower case

```
$ sort -f file1
export "PS1=$ "
sort -f file1

This is a new file
export "PS1=$ "
```

9. Displays only directories in current working directory.

```
$ ls -dF */
export "PS1=$ "
ls -dF */
```

10. Copying files from one place to another,

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ cp file1 file2
```

11. Moving files from one place to another.



Department of Computer Science and Engineering (Data Science)

A.Y.: 2022-23

Sub: System Fundamentals

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ mv file1 file2
```

12. Removing specific directory with various options

```
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ mkdir amitesh
cseds-student@csedsstudent-HP-ProOne-600-G6-22-All-in-One-PC:~$ rmdir amitesh
```

13. List the numbers of users currently login in the system and then sort it.

```
$ who | awk '{print $1}' | sort | uniq -c
export "PS1=$ "
who | awk '{print $1}' | sort | uniq -c
export "PS1=$ "
```

14. Merge two files into one file

```
root@tryit-assured:~# cat file1 file2>merged.txt
root@tryit-assured:~# head merged.txt
This is a new file
This is the second file
```

15. Change the access mode of a given file

```
root@tryit-assured:~# chmod u+rw file1
```

16. Display the last ten lines of the file.

```
root@tryit-assured:~# tail file1
This is a new file
```

17. Locate files in a directory and in a subdirectory.

```
find directory -name "filename"
```

18. This displays the contents of all files having a name starting with ap followed by any number of characters.



Department of Computer Science and Engineering (Data Science)

A.Y.: 2022-23

Sub: System Fundamentals

```
$ for file in ap*; do cat $file; done
export "PS1=$ "
for file in ap*; do cat $file; done

export "PS1=$ "

$ cat ap*
export "PS1=$ "
cat ap*

export "PS1=$ "
```

19. Rename any file aaa to aaa.aa1, where aa1 is the user login name.

```
$ mv aaa aaa.$(whoami)
```

20. Write a command to search the word 'picture' in the file and if found, the lines containing it would be displayed on the screen.

```
grep "picture" file.txt
```

21. Write a command to search for all occurrences of 'Rebecca' as well as 'rebecca' in file and display the lines which contain one of these words.

```
grep -i "rebecca" file.txt
```

22. Write a command to search all four-letter words whose first letter is a 'b' and last letter, a 'k'.

```
grep '\bb[a-z][a-z]k\b' file.txt
```

23. Write a command to see only those lines which do not contain the search patterns

```
grep -v "pattern" file.txt
```

24. Sort a given file in alphabetical order using sort command.

```
sort file.txt
```



Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

(Autonomous College Affiliated to the University of Mumbai)

NAAC Accredited with "A" Grade (CGPA : 3.18)



Department of Computer Science and Engineering (Data Science)

A.Y.: 2022-23

Sub: System Fundamentals

25. From a given xml file search all records associated with “DataScience” branch using grep command.

```
xmlstarlet sel -t -c "//record[branch='DataScience']" file.xml
```