

INDEX

Name SIMRAN

Std. CSE

Div. II Year

Sub. UNIX LAB

Roll No. 220180101049

Experiment-1

Date 15/2/24

Expt. No. 1

Page No. 1

Q- What are the differences between Linux and Windows?

LINUX	WINDOWS
1. Linux is an open source operating system.	1. Windows are not an open source operating system.
2. It is free of cost.	2. It is costly.
3. Its file name is case sensitive.	3. Its file name is not case sensitive.
4. Monolithic kernel is used.	4. Hybrid kernel is used.
5. It is more efficient.	5. It is less efficient.
6. Forward slash is used for separating the directories.	6. Backward slash is used for separating the directories.
7. It provides more security than windows.	7. It provides less security than linux.
8. It is widely used in hacking purpose based systems.	8. Windows does not provide much efficiency in hacking.
9. There are three types of user account - (i) Regular (ii) Root (iii) Service account.	9. There are four types of user account - (i) Administrator (ii) Standard (iii) Child (iv) Guest
10. Root user is the super user and has all administrative privileges.	10. Administrator user has all administrative privileges of the computer.
11. Linux file naming convention is case sensitive.	11. In Windows, you cannot have two files with same name in the same folder.

Teacher's Signature :

Experiment-2

Date 15/2/24

Expt. No. 2

Page No. 2.

Q- Write about the different versions of Linux.

Ans- There are on an average 600 Linux versions (distributions) providing different features. Some of the popular versions are:

- (i) Ubuntu:- came into existence in 2004 by Canonical. It is the most well known Linux distribution.
- (ii) Linux Mint:- is based on Ubuntu and uses its repository software so some packages are common in both.
- (iii) Debian:- exists since 1993 and is one of the most stable Linux distributor.
- (iv) Red Hat enterprise / CentOS:- is a commercial Linux distributor. Their products are red hat enterprise Linux (RHEL) and Fedora which are freely available.
- (v) Fedora:- focuses mainly on free software and provides latest version of software.
- (vi) Arch Linux:- is based on 5 principles: Simplicity, modernity, pragmatism, user centrality and versatility.
- (vii) Oracle Linux:- is distributed by oracle since late 2006 under the GNU GPL.
- (viii) Gentoo:- is a distribution designed to have highly optimized and frequently updated software.

Teacher's Signature :

Experiment-3

Date 2/2/24

Expt. No. 3

Page No. 3

Q- Explain the different types of editors in Linux.

(i) vi Editor

The default editor that comes with the Linux/UNIX operating system is called vi (visual editor). Using vi editor, we can edit an existing file or create a new file from scratch. We can also use this editor to just read a text file. The advanced version of the vi editor is the vim editor.

To open vi editor, we need to type the command
vi [file-name].

For ex -

vi unixlab

(ii) Nano Text Editor

Nano is a user-friendly, simple and WYSIWYG (what you see is what you get) text editor, which improves the features and user-friendliness of UN Pico text editor. Unlike vim editor or any other command-line editor, it doesn't have any mode. It has an easy GUI (Graphical User Interface) which allows users to interact directly with the text in spite of switching between the modes as in vim editor.

To create and open a new file -

\$ nano new-filename.

Teacher's Signature :

Experiment - 4

Date 29/2/24

Expt. No. 4

Page No. 4

Q- write the syntax and output of the different general purpose utility command list.

(i) cd :- changes the current working directory.

Syntax:- cd Documents.

(ii) ls :- lists files and directories in the current directory.

Syntax:- ls

(iii) pwd:- prints the current working directory.

Syntax:- pwd

(iv) mkdir :- creates a new directory.

Syntax:- mkdir new_folder

(v) rmdir:- removes an empty directory.

Syntax:- rmdir empty_folder

(vi) mv:- moves files or directories.

Syntax:- mv file1.txt Documents/

(vii) cp:- copies files or directories.

Syntax:- cp file1.txt file2.txt

(viii) man:- gives detailed information about various commands, utilities and system calls.

Syntax:- man [option] [command]

Teacher's Signature :

Expt. No. _____

(ix) who :- display information about currently logged-in user.
Syntax:- \$ who

(x) cat :- concatenates files and displays their contents.
Syntax:- cat file1.txt file2.txt

(xi) ps :- displays information about active processes, including status and IDs. Syntax:- ps aux

(xii) echo :- used to display lines of text or strings passed as arguments.
Syntax:- echo [option] [string].

(xiii) more :- displays the contents of a file page by page.
Syntax:- more file.txt

(xiv) rm :- removes/deletes files and directories from file system.
Syntax:- rm [option] ... FILE

(xv) date :- command is used to display the system date and time.
Syntax:- date [OPTION] ... [+FORMAT]

(xvi) time :- measures the time it takes to execute a command.
Syntax:- time [options] command [arguments]

(xvii) kill :- command is used to terminate processes manually.
Syntax:- kill [signal] PID

Teacher's Signature : _____

Expt. No. _____

(xix) history :- command is used to display the list of commands that have been previously executed, with the most recent commands displayed last.

Syntax :- `history [options] [number]`

(xx) chmod :- allows an administrator to set or modify a file's permissions.

Syntax :- `chmod [permission-level] [file]`

(xxi) chown :- changes the file ownership.

Syntax :- `chown [new-user] [filename]`

(xxii) finger :- command provides details about users on a system.

Syntax :- `finger [username]`

(xxiii) cal :- command is used to see the calendar of a specific month or a whole year.

Syntax :- `cal [month] [year]`

(xxiv) logout :- command logs out the logged-in user from the system in that session.

Syntax :- `$ logout`

(xxv) shutdown :- command is used to shutdown the system in a safe way.

Syntax :- `shutdown [OPTIONS] [TIME] [MESSAGE]`

Teacher's Signature : _____

Expt. No. 5

Q- Discuss pipes and filters in UNIX.

Ans: Piping is used to give the output of one command (written on LHS) as input to another command (written on RHS).

commands are piped together using vertical bar " | " symbol.

Syntax:-

command 1 | command 2

Filters are the set of commands that take Input from standard input stream i.e. stdin, perform some operations and write output to standard output stream i.e. stdout. The stdin and stdout can be managed as per preferences using redirection and pipes.

common filter commands are: grep, more, sort.

(i) grep command searches for a pattern or regular expression that matches in files or directories and then prints found matches.

Syntax:- \$ grep [options] "pattern to be matched" filename

(ii) sort command sorts lines of text alphabetically or numerically.

Syntax:- \$ sort [options] filename

(iii) more command is used to customize the displaying contents of file.

Syntax:- \$ more [options] filename

Teacher's Signature :

Q- Write about the sed command in UNIX.

Ans:- SED command in UNIX stands for Stream editor and it can perform lots of functions on file like searching, find and replace, insertion or deletion.

- SED is a powerful text stream editor. It can do insertion, deletion, search and replace (substitution).
- SED command in unix supports regular expression which allows it to perform complex pattern matching.
- Syntax :- sed OPTIONS... [SCRIPT] [INPUTFILE...]
- It can be used for :-
 - (i) Replacing or substituting string.
 - (ii) Replacing n^{th} occurrence of a pattern in a line.
 - (iii) Replacing all the occurrence of the pattern in a line.
 - (iv) Parenthesize first character of each word.
 - (v) Replacing string on a specific line number.
 - (vi) Printing only the replaced lines.
 - (vii) Deleting lines from a particular file.

Teacher's Signature : _____

Expt. No. 7.

Q- Write about the AWK command in UNIX.

Ans: AWK is a scripting language used for manipulating data and generating reports. The awk command programming language requires no compiling and allows the user to use variables, numeric functions, string functions, and logical operators.

- It is a utility that enables a programmer to write tiny but effective programs in the form of statements that define text patterns that are to be searched. Awk is mostly used for pattern scanning and processing.
- AWK is abbreviated from the names of the developers Aho, Weinberger and Kernighan.

• AWK Operations:

- (a) Scans a file line by line
- (b) Splits each input line into fields.
- (c) Compares input line fields to pattern.
- (d) Performs action(s) on matched lines.

• Useful for:

- (a) Transform data files.
- (b) Produce formatted reports.

• Programming constructs:

- (a) Format output lines
- (b) Arithmetic and string operations.
- (c) conditionals and loops.

Teacher's Signature : _____

INDEX

Name SIMRAN

Sub. UNIX LAB

Std. CSE

Div. II Year

Roll No. 220180161049

S. No.	Topic	Page No.	Date	Grade/ Marks	Signature
1.	Difference b/w Linux & Windows	1	15/2/24		
2.	Different versions of Linux	2	15/2/24		
3.	Different types of editor vi and nano editor	3	22/2/24		
4.	Different general purpose utility command list.	4 - 6	29/2/24		
5.	Pipes and filters in LINUX	7	29/2/24		
6.	Sed Command	8	6/3/24		
7.	AWK Command	9.	6/3/24		
8.	Shell script program to: (a) display list of user currently logged in (b) to copy contents of one file to another	10.	13/3/24		
9.	Write a program using sed command to print duplicated lines of input.	11	13/3/24		
10.	Write a grep / egrep script to find the number of words and lines in a file.	12	13/3/24		

S. No.	Topic	Page No.	Date	Grade/ Marks	Signature
11.	Write an awk script to develop ----- number.	13-14	4/4/24		
12.	Write a shell script program to ----- zombie process.	15-17	4/4/24		
13.	Write a program to ----- 'child' on the screen.	18	18/4/24		
14.	Write a makefile to compile a C program.	19	18/4/24		
15.	Study to execute ----- for debugging	20	25/4/24		
16.	study to use ssh ----- network tools.	21-22	25/4/24		

Experiment - 8

Expt. No. 8

Date 13/3/24

Page No. 10

(a) Write a shell script program to:

(a) display list of user currently logged in.
echo "Currently logged on users:"
who

(b) to copy contents of one file to another.

bash

if [! -f "\$1"]; then
echo "Source file does not exist"
exit 1

fi

cat "\$1" > "\$2"

echo "contents copied successfully from \$1 to \$2"

Teacher's Signature :

Experiment - 9

Expt. No. 9

Date 13/3/24

Page No. 11

Q- Write a program using sed command to print duplicate lines of Input.

Ans- Input = "This is a test

This is a test

This is another test

And this is yet another test

And this is yet another test"

echo "\$input" | sed -n '/^.*\n\1\$/p'

Teacher's Signature : _____

Experiment - 10

Date 13/3/24

Expt. No. 10

Page No. 12

Q- Write a grep/egrep script to find the number of words, characters and lines in a file.

if [\$# -eq 0]; then

echo "Usage: \$0 <filename>"

exit 1

fi

filename = \$1

if [! -F "\$filename"]; then

echo "File '\$filename' does not exist."

exit 1

fi

line_count = \$(wc -l < "\$filename")

word_count = \$(wc -w < "\$filename")

char_count = \$(wc -m < "\$filename")

echo "File: \$filename"

echo "Lines: \$line_count"

echo "Words: \$word_count"

echo "Characters: \$char_count"

Teacher's Signature :

Experiment - II

Date 4/4/24

Expt. No. 11

Page No. 13

(a) Write an awk script to do :-

(a) develop a Fibonacci series.

BEGIN {

num_terms = 10

a = 0

b = 1

print "Fibonacci Series:"

if (num_terms >= 1) {

 print a

}

if (num_terms >= 2) {

 print b

}

for (i = 3; i <= num_terms; i++) {

 next = a + b

 print next

 a = b

 b = next

}

}

Output:-

• 1 1 2 3 5 8 13

21 34

(b) display the pattern of given string or number.

BEGIN {

number_of_rows = 5

input = "X"

}

Teacher's Signature : _____

{

```
if (number_of_rows > 0 && $0 != " ") {
```

```
    input = $0
```

```
    for (i=1; i <= number_of_rows; i++) {
```

```
        for (j=1; j <= i; j++) {
```

```
            printf ("%c", input)
```

```
}
```

```
        printf ("\n")
```

```
}
```

```
}
```

Output:-

X

XX

XXX

XXXX

XXXXX

Experiment - 12

Date 4/4/24

Expt. No. 12.

Page No. 15

a. Write a shell script program to :-

(a) display the process attributes.

```
echo "listing all running processes with detailed attributes"  
ps aux
```

(b) change priority of processes

```
if [ $# -ne 2 ]; then
```

```
echo "Usage: $0 <PID> <new-niceness>"
```

```
exit 1
```

b^o

```
PID=$1
```

```
NEW_NICENESS=$2
```

```
if ! [[ "$NEW_NICENESS" =~ ^-[0-9]+$ ]]; then
```

```
echo "Error: Niceness must be an integer."
```

```
exit 1
```

bⁱ

```
if ! ps -p $PID >/dev/null; then
```

```
echo "Error: Process with PID $PID does not exist."
```

```
exit 1
```

b^o

```
echo "Changing the niceness of process with PID $PID to  
$NEW_NICENESS"
```

```
sudo renice $NEW_NICENESS -p $PID
```

```
if [ $? -eq 0 ]; then
```

```
echo "Successfully changed the niceness of process with PID  
$PID to $NEW_NICENESS."
```

Teacher's Signature: _____

```
else
```

```
    echo "Failed to change the niceess of process with PID $PID"
```

```
fi
```

(c) to change the ownership of processes.

```
if [ $# -lt 2 ]; then
```

```
    echo "Usage : $0 <username><command>[args...]"
```

```
    exit 1
```

```
fi
```

```
USERNAME = $1
```

```
shift
```

```
COMMAND = $@
```

```
echo "Running '$COMMAND' as user '$USERNAME' ..."
```

```
sudo -u $USERNAME $COMMAND
```

(d) to send back a process from foreground.

```
if [-z "$1"]; then
```

```
    echo "Usage : $0 <PID>"
```

```
    exit 1
```

```
fi
```

```
if ! ps -p "$1" >/dev/null; then
```

```
    echo "Process $1 not found."
```

```
exit 1
```

```
fi
```

```
if kill -STOP "$1"; then
```

```
    echo "Process $1 sent to background."
```

```
else
```

```
    echo "Failed to send processes $1 to background."
```

```
fi
```

Teacher's Signature : _____

(e) to retrieve a process from background.

```
if [-z "$1"]; then
```

```
echo "Usage: $0 <PID>"
```

```
exit 1
```

```
fi
```

```
if !ps -p "$1">>/dev/null; then
```

```
echo "Process $1 not found."
```

```
exit 1
```

```
fi
```

```
if kill -CONT "$1"; then
```

```
echo "Process $1 brought to foreground."
```

```
else
```

```
echo "Failed to bring process $1 to foreground".
```

```
fi
```

(f) to create a zombie process.

```
create-zombie-process() {
```

```
echo "Forking child process..."
```

```
sleep 10 &
```

```
sleep 2
```

```
}
```

```
echo "Creating a zombie process..."
```

```
create-zombie-process
```

```
echo "Parent process exiting."
```

```
exit 0.
```

Experiment - 13

Date 18/4/24

Expt. No. 13

Page No. 18

- Q- Write a program to create a child process and allow the parent to display 'parent' and the child to display 'child' on the screen.

child-process () {

 echo "Child"

}

child-process &

 echo "Parent"

Experiment -14

Date 18/4/24

Expt. No. 14

Page No. 19

- Q. Write a makefile to compile a c program.

CC = gcc

CFLAGS = -Wall -Wextra -std=c99

TARGET = program

SRC = program.c

all : \$(TARGET)

\$(TARGET) : \$(SRC)

\$(CC) \$(CFLAGS) -o \$(TARGET) \$(SRC)

clean :

rm -f \$(TARGET)

Experiment -15

Date 25/4/24

Expt. No. 15

Page No. 20

Q. Study to execute programs using gdb to utilize its various features like breakpoints, conditional breakpoints. Also write a shell script program to include verbose and xtrace debug option for debugging.

- Ans.
- Breakpoints:- You can set breakpoints at specific lines of code or function names to pause program execution at those points.
 - conditional Breakpoints:- Breakpoints that only trigger if certain conditions are met.
 - Inspecting variables
 - Stepping through code
 - Backtrace
 - Watchpoints
 - core dumps

Program :-

set -v

set -x

echo "Starting script"

ls /nonexistent_directory

echo "Script completed"

Teacher's Signature :

Experiment -16

Date 25/4/24

Expt. No. 16

Page No. 21

Q- Study to use:

(a) SSH (Secure shell):-

- It provides secure encrypted communication between two hosts over an insecure network.
- Commonly used for remote login to a server or transferring files securely.
- Syntax: 'ssh username @ hostname'

(b) Telnet :-

- It is a network protocol used on the Internet or local area networks to provide a bidirectional interactive text-oriented communication facility using a virtual terminal connection.
- It is not secure as it transmits data, including passwords, in plain text.
- Syntax: 'telnet hostname'

(c) PUTTY :

- It is a free and open-source terminal emulator, serial console, and network file transfer application.
- It supports several network protocols, including SSH, Telnet, rlogin, and raw socket connection.
- It is widely used on Windows for SSH connection to Unix/Linux servers.
- Syntax: Run PUTTY and provide the hostname or IP address of the server.

Teacher's Signature :

(d) FTP (File Transfer Protocol): -

- It is a standard network protocol used to transfer files between a client and a server over a network.
- It operates on the client-server model and uses separate control and data connections between the client and the server.
- Syntax: 'ftp hostname'

(e) ncftp:

- It is an FTP client program with an interface similar to traditional FTP clients but with additional features like bookmarking, command-line editing, and more.
- It provides more advanced functionality compared to the basic 'ftp' command.
- Syntax: 'ncftp hostname' to connect to an FTP server

(f) Other network tools:

- These include 'ping' for testing network connectivity, 'traceroute' for tracing the route packets take to reach a destination, 'netcat' for reading from and writing to network connections using TCP or UDP, 'nmap' for network exploration and security auditing, and 'tcpdump' for network packet analysis.

Teacher's Signature: _____

(d) FTP (File Transfer Protocol): -

- It is a standard network protocol used to transfer files between a client and a server over a network.
- It operates on the client-server model and uses separate control and data connections between the client and the server.
- Syntax: 'ftp hostname'

(e) ncFTP:

- It is an FTP client program with an interface similar to traditional FTP clients but with additional features like bookmarking, command-line editing, and more.
- It provides more advanced functionality compared to the basic 'ftp' command.
- Syntax: 'ncFTP hostname' to connect to an FTP server

(f) Other network tools:

- These include 'ping' for testing network connectivity, 'traceroute' for tracing the route packets take to reach a destination, 'netcat' for reading from and writing to network connections using TCP or UDP, 'nmap' for network exploration and security auditing, and 'tcpdump' for network packet analysis.