

The main idea of this lab exercise to give hands on experience on

grep

constructs

command line arguments

1. write a shell script to get value the pattern and file name from the user and check the pattern exists or not. If the pattern exists print the relevant

message, if pattern not found print relevant message.

```
Open  lab4.sh
~/lab/linux
1 #!/usr/bin/bash
2
3
4 read -p "Enter the pattern to search: " pattern
5 read -p "Enter the file name: " file
6
7
8 if grep -q "$pattern" "$file"; then
9     echo "Pattern '$pattern' found in file '$file'"
10 else
11     echo "Pattern '$pattern' not found in file '$file'"
12 fi
```

```
(ceyona@kali)-[~/lab/linux]
$ bash lab4.sh
Enter the pattern to search: hi
Enter the file name: file1.txt
Pattern 'hi' found in file 'file1.txt'
```

2. Modify the above script to pass the arguments from command line arguments.

```
Open  2.sh
~/lab/linux/lab4
1 #!/usr/bin/bash
2 if [[ $# -ne 2 ]]; then
3     echo "Usage: $0 <pattern> <file>"
4     exit 1
5 fi
6
7 pattern=$1
8 file=$2
9
10
11 if grep -q "$pattern" "$file"; then
12     echo "Pattern '$pattern' found in file '$file'"
13 else
14     echo "Pattern '$pattern' not found in file '$file'"
15 fi
```

```
(ceyona@kali)-[~/lab/linux/lab4]
$ bash 2.sh hi file1.txt
Pattern 'hi' found in file 'file1.txt'
```

3. Modify the above script to pass the values inside the script.

```
Open 2.sh ~/lab/linux/lab4
1 #!/usr/bin/bash
2
3 pattern="hi"
4 file="file1.txt"
5
6
7 if grep -q "$pattern" "$file"; then
8     echo "Pattern '$pattern' found in file '$file'"
9 else
10    echo "Pattern '$pattern' not found in file '$file'"
11 fi
```

```
(ceyona@kali)-[~/lab/linux/lab4]
$ bash 2.sh
Pattern 'hi' found in file 'file1.txt'
```

4. validate the script (script 1, script 2)

- the file exists or not
- arguments passed or not

```
Open 2.sh ~/lab/linux/lab4
1 #!/usr/bin/bash
2
3 pattern=$1
4 file=$2
5
6 if [[ $# -ne 2 ]]; then
7     echo "Usage: $0 <pattern> <file>"
8     exit 1
9 fi
10
11 if grep -q "$pattern" "$file"; then
12     echo "Pattern found"
13 else
14     echo "Pattern not found"
15 fi
```

```
(ceyona@kali)-[~/lab/linux/lab4]
$ bash 2.sh
Usage: 2.sh <pattern> <file>

(ceyona@kali)-[~/lab/linux/lab4]
$ bash 2.sh hello file1.txt
Pattern found

(ceyona@kali)-[~/lab/linux/lab4]
$ bash 2.sh hello
Usage: 2.sh <pattern> <file>

(ceyona@kali)-[~/lab/linux/lab4]
$ bash 2.sh hello file3.txt
grep: file3.txt: No such file or directory
Pattern not found
```

5. Apply grep commands

Note: Make sure to use the options -e -c -n -q -s -f -A -B -C -i -h, -l -o -w

Frame the questions (as per your choice)

to extract user information

to extract network information

to extract login details

1. To extract network information

```
(ceyona@kali)-[~/lab/linux/lab4]
$ grep -A 2 -B 2 "text" file1.txt
hi
how are you
text
Are you happy
crypto lab
los lab
text
hi
how are you
text
are you happy
crypto lab
los lab
text
enter your password
```

2. To extract login details

```
(ceyona@kali)-[~/lab/linux/lab4]
$ grep -c "login" file1.txt
3
```

3. Using Multiple options

```
(ceyona@kali)-[~/lab/linux/lab4]
$ grep -o -n -w "login" file1.txt
5:login
11:login
18:login
```

4. To search multiple pattern using -e

```
(ceyona@kali)-[~/lab/linux/lab4]
$ grep -e "login" -e "hi" file1.txt
hi
hi
login here
login to begin
hi
login
```

