Operating System Lab Report

Lab Report on Unix/Linux Topics

---

1. Objectives

- Understand the fundamentals of Unix/Linux operating systems.
- Explore and practice system administration commands, file handling, and directory structures.
- Gain proficiency in using shell scripting to automate tasks and solve real-world problems.
- Develop familiarity with user and group management, process management, network commands, and archival in Linux.
- Learn to implement and manipulate variables, arrays, conditional statements, and arithmetic operations in shell scripting.

---

2. Introduction

The Unix and Linux operating systems provide a powerful, multitasking, multiuser environment widely used for various applications, including software development, system administration, and server management. This lab report covers essential Unix/Linux commands, system administration techniques, and shell scripting concepts. By working through these topics, users gain critical skills in managing files, directories, users, processes, and scripts, which are vital for operating system administration and automation.

---

3. Theory

3.1 Unix and Linux Basics

- **Unix** is a powerful operating system developed in the 1970s, focusing on simplicity, portability, and multitasking. It provides a hierarchical file structure, shell command interface, and support for network communication.
- **Linux** is a Unix-like operating system kernel developed by Linus Torvalds. It has grown to include various distributions and is widely used for servers, desktops, and embedded systems.

3.2 Super User

- The **superuser** or **root** user in Unix/Linux has unrestricted access to all commands, files, and system processes, essential for performing critical system administration tasks.

3.3 System Directory Structure

- Unix/Linux follows a hierarchical directory structure where directories and subdirectories organize the filesystem, including `/home` for user files, `/bin` for binary files, `/etc` for configuration, and `/var` for variable data.

3.4 File Handling Commands

- **ls, cp, mv, rm, mkdir, rmdir**: Commands for listing, copying, moving, and deleting files and directories.
- **cat, touch, nano, vi**: Commands for creating, editing, and viewing text files.

3.5 System Administration Commands

- Commands such as `top`, `ps`, `kill`, and `service` are crucial for monitoring and managing processes and services on a Unix/Linux system.

3.6 Text Processing Commands

- Text processing commands like `grep`, `awk`, `sed`, and `cut` allow users to search, filter, and modify text data, making Unix/Linux ideal for data manipulation.

3.7 File Permissions

- File permissions in Unix/Linux control access to files and directories, represented by read, write, and execute permissions for the owner, group, and others.

3.8 User and Group Management

- Creating, deleting, and modifying users and groups are vital tasks in system administration, allowing control over system access and resource allocation.

3.9 Process Management

- Unix/Linux systems support multitasking, with process management commands like `ps`, `top`, `kill`, and `nice` allowing users to view, terminate, and prioritize processes.

3.10 Archival and Compression

- Archival commands such as `tar`, `gzip`, and `zip` allow files to be packaged and compressed, useful for backups and file transfer.

3.11 Network Commands

- Commands like `ping`, `ifconfig`, `netstat`, and `ssh` enable network management, troubleshooting, and secure remote connections in Linux.

3.12 Shell Scripting Basics

- Shell scripting provides automation capabilities in Unix/Linux, allowing users to create scripts that execute a series of commands, making tasks repeatable and efficient.

3.13 Data Types and Variables in Shell Scripting

- Shell scripting supports variables for storing data, including integers, strings, and arrays, used to control and manipulate information within scripts.

3.14 Input/Output and Arithmetic Operations

- Commands like `echo`, `read`, and arithmetic operators allow user interaction and mathematical operations within scripts.

3.15 Conditional Statements and Relational Operators

- **if, else, elif** statements, along with operators (`-eq`, `-ne`, `-gt`, etc.), provide logic-based decisions, controlling script flow.

3.16 Arrays

- Arrays store multiple values in a single variable, useful for handling lists and collections of data within shell scripts.

4. Software/Hardware Used

- **Software**: Unix/Linux OS (Ubuntu), terminal or command line interface, text editor (nano, vim).
- **Hardware**: Computer with at least 4 GB RAM, preferably with Unix/Linux OS pre-installed or a virtual machine with Linux.

5. Sample Input and Output

Task 1

Objective: Basic directory and file handling commands in Linux.

Create "mydir" directory: `mkdir mydir`

Create "cpdir" directory: `mkdir cpdir`

Navigate to "mydir": `cd mydir`

Create "test.txt" file: `touch test.txt`

Move "test.txt" to "cpdir": `mv test.txt ../cpdir`

Print history: `history`

```
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_1$ mkdir mydir
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_1$ mkdir cpdir
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_1$ cd mydir
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_1/mydir$ touch test.txt
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_1/mydir$ mv test.txt ../cpdir
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_1/mydir$ ls -a
.  ..
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_1/mydir$ cd ../cpdir/
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_1/cpdir$ ls -a
.  ..  test.txt
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_1/cpdir$
```

```
  336  mkdir mydir
  337  mkdir cpdir
  338  cd mydir
  339  touch test.txt
  340  mv test.txt ../cpdir
  341  ls -a
  342  cd ../cpdir/
  343  ls -a
  344  history
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_1/cpdir$
```

Task 2

Objective: Working with process listing and network configuration.

List processes: `ps`

Display network configuration: `ifconfig`

Print history: `history`

```
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_2$ ps
    PID TTY          TIME CMD
   4288 pts/1    00:00:00 bash
   4589 pts/1    00:00:00 ps
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_2$ ifconfig
enp3s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.103  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::da43:aeff:fea0:6660  prefixlen 64  scopeid 0x20<link>
        ether d8:43:ae:a0:66:60  txqueuelen 1000  (Ethernet)
        RX packets 25421  bytes 20969472 (20.9 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 19832  bytes 7389294 (7.3 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 1117  bytes 113747 (113.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1117  bytes 113747 (113.7 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_2$
```

```
  350  ps
  351  ifconfig
  352  history
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_2$
```

Task 3

Objective: Basic permissions management and file creation with `chmod`.

Create "install.sh": `nano install.sh`

```
echo "hello world"
```

Create "update.sh": `nano update.sh`

```
echo "updating system..."
```

Add write and execute permissions to "install.sh" for user: `chmod u+rwx install.sh`

Add read and execute permissions to "install.sh" for group using numeric notation: `chmod 750 install.sh`

```
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_3$ nano install.sh
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_3$ nano update.sh
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_3$ chmod u+rwx install.sh
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_3$ chmod 750 install.sh
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_3$ ls -l install.sh
-rwxr-x--- 1 asif-ahammed asif-ahammed 19 Oct 27 22:18 install.sh
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_3$ ls -l update.sh
-rw-rw-r-- 1 asif-ahammed asif-ahammed 27 Oct 27 22:18 update.sh
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_3$
```

```
  359  nano install.sh
  360  nano update.sh
  361  chmod u+rwx install.sh
  362  chmod 750 install.sh
  363  ls -l install.sh
  364  ls -l update.sh
  365  history
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_3$
```

Task 4

Objective: User management commands.

Create a user with your name: `sudo useradd nishat`

Give sudo permissions: `sudo usermod -aG sudo nishat`

Create "user1": `sudo useradd user1`

Delete "user1": `sudo userdel user1`

```
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_4$ sudo useradd nishat
[sudo] password for asif-ahammed:
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_4$ sudo usermod -aG sudo nishat
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_4$ cat /etc/passwd
```

```
asif-ahammed:x:1000:1000:Asif-ahammed:/home/asif-ahammed:/bin/bash
nishat:x:1001:1001::/home/nishat:/bin/sh
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_4$
```

```
  374  sudo useradd nishat
  375  sudo usermod -aG sudo nishat
  376  cat /etc/passwd
  377  history
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_4$
```

Task 5

Objective: Group management commands.

Create a group "bubtcse": `sudo groupadd bubtcse`

Add your user to "bubtcse": `sudo usermod -aG bubtcse nishat`

Rename "bubtcse" to "cse": `sudo groupmod -n cse bubtcse`

Print group names of your user: `groups nishat`

```
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_5$ sudo groupadd bubtcse
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_5$ sudo usermod -aG bubtcse nishat
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_5$ sudo groupmod -n cse bubtcse
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_5$ groups nishat
nishat : nishat sudo cse
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_5$ cat /etc/group
```

```
asif-ahammed:x:1000:
nishat:x:1001:
cse:x:1002:nishat
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_5$
```

```
  393  sudo groupadd bubtcse
  394  sudo usermod -aG bubtcse nishat
  395  sudo groupmod -n cse bubtcse
  396  groups nishat
  397  cat /etc/group
  398  history
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_5$
```

Task 6

Objective: Process management and priority setting.

List processes with `top`: `top`

List all current processes with `ps`: `ps -ef`

Set custom priority to a process: `nice -n 10 ./my_script`

```
  422  top
  423  ps -ef
  424  nice -n 10 ./my_script.sh
  425  history
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_6$
```

Task 7

Objective: File handling, zipping, and moving files.

Create four text files: `touch file1.txt file2.txt file3.txt file4.txt`

Create a tar file: `tar cvf file.tar file1.txt file2.txt file3.txt file4.txt`

Create "myfolder": `mkdir myfolder`

Move text files to "myfolder": `mv file1.txt file2.txt file3.txt file4.txt myfolder`

Zip "myfolder": `zip -r myfile.zip myfolder`

```
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$ touch file1.txt file2.txt
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$ ls
file1.txt  file2.txt
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$ tar cvf file.tar file1.txt file2.txt
file1.txt
file2.txt
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$ ls
file1.txt  file2.txt  file.tar
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$ mkdir myfolder
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$ ls
file1.txt  file2.txt  file.tar  myfolder
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$ mv file1.txt file2.txt myfolder
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$ ls
file.tar  myfolder
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$ zip -r myfile.zip myfolder
  adding: myfolder/ (stored 0%)
  adding: myfolder/file1.txt (stored 0%)
  adding: myfolder/file2.txt (stored 0%)
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$ ls
file.tar  myfile.zip  myfolder
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$
```

```
  431  touch file1.txt file2.txt
  432  ls
  433  tar cvf file.tar file1.txt file2.txt
  434  ls
  435  mkdir myfolder
  436  ls
  437  mv file1.txt file2.txt myfolder
  438  ls
  439  zip -r myfile.zip myfolder
  440  ls
  441  history
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_7$
```

Task 8

Objective: Download files using `wget`.

Download a file: `wget https://docs.google.com/document/d/1dFs2qCFJ8bew06wGzgeQTKHfTsD2xp3LGVussbPg99I/export?format=pdf`

```
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_8$ wget "https://docs.google.com/document/d/1
dFs2qCFJ8bew06wGzgeQTKHfTsD2xp3LGVussbPg99I/export?format=pdf"
--2024-10-27 23:26:00--  https://docs.google.com/document/d/1dFs2qCFJ8bew06wGzgeQTKHfTsD2xp3LGVussbPg99I/export?format=pdf
Resolving docs.google.com (docs.google.com)... 142.250.183.238, 2404:6800:4007:81e::200e
Connecting to docs.google.com (docs.google.com)|142.250.183.238|:443... connected.
HTTP request sent, awaiting response... 307 Temporary Redirect
Location: https://doc-0g-8g-docstext.googleusercontent.com/export/e6hpso97lrhpva19l3uocm525o/mko3uruthnkko8bs5oqe7vcb4g/173004996
0000/11253961004303746 4085/*/1dFs2qCFJ8bew06wGzgeQTKHfTsD2xp3LGVussbPg99I?format=pdf [following]
Warning: wildcards not supported in HTTP.
--2024-10-27 23:26:00--  https://doc-0g-8g-docstext.googleusercontent.com/export/e6hpso97lrhpva19l3uocm525o/mko3uruthnkko8bs5oqe7
vcb4g/1730049960000/11253961004303746 4085/*/1dFs2qCFJ8bew06wGzgeQTKHfTsD2xp3LGVussbPg99I?format=pdf
Resolving doc-0g-8g-docstext.googleusercontent.com (doc-0g-8g-docstext.googleusercontent.com)... 172.217.160.129, 2404:6800:4007:
80a::2001
Connecting to doc-0g-8g-docstext.googleusercontent.com (doc-0g-8g-docstext.googleusercontent.com)|172.217.160.129|:443... connect
ed.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/pdf]
Saving to: ‘export?format=pdf’

export?format=pdf                   [     <=>                                          ] 625.31K   522KB/s    in 1.2s

2024-10-27 23:26:02 (522 KB/s) - ‘export?format=pdf’ saved [640322]

asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_8$ ls
'export?format=pdf'
asif-ahammed@asif-ahammed-MS-7E02:~/Documents/operating system/Lab_report_code/Task_8$
```

Task 9

Objective: Basic shell scripting for input and output prompts.

Create "task1.sh": `nano task1.sh`

Content:

```
#! /bin/bash

echo "Enter username:"

read username

echo "Enter password:"

read password

echo "Username: $username, Password: $password"


echo "Enter three course names:"

read course1 course2 course3

echo "The courses are: $course1 $course2 $course3"
```

Task 10

Objective: Arithmetic operations in shell script.

Create "task2.sh": `nano task2.sh`

Content:

```
#! /bin/bash

echo "Enter two numbers:"

read num1 num2

echo "Addition: $((num1 + num2))"

echo "Subtraction: $((num1 - num2))"

echo "Multiplication: $((num1 * num2))"

echo "Division: $((num1 / num2))"

echo "Modulus: $((num1 % num2))"

echo "Exponent: $((num1 ** num2))"
```

Task 11

Objective: Averaging and printing specific data using shell script.

Create "task3.sh": `nano task3.sh`

Content:

```
#! /bin/bash
```

```
echo "Enter course names:"

read course1 course2 course3

echo "Enter marks for $course1:"

read s1 s2 s3 s4

echo "Enter marks for $course2:"

read s5 s6 s7 s8

echo "Enter marks for $course3:"

read s9 s10 s11 s12


echo "Average for $course1: $(( (s1 + s2 + s3 + s4) / 4 ))"

echo "Average for $course2: $(( (s5 + s6 + s7 + s8) / 4 ))"

echo "Average for $course3: $(( (s9 + s10 + s11 + s12) / 4 ))"


echo "Marks for 2nd to 4th students in $course3: $s10, $s11, $s12"
```

1.

Task 12

Objective: Conditional directory and file creation.

Create "dir_file_checker.sh": `nano dir_file_checker.sh`

Content:

```
#! /bin/bash

[ -d "$1" ] || mkdir "$1"

[ -f "$1/$2" ] || touch "$1/$2"
```

Task 13

Objective: Age validation using conditional statements.

Create "age_validator.sh": `nano age_validator.sh`

Content:

```
#! /bin/bash

echo "Enter your name:"

read name

echo "Enter your age:"

read age

if [ "$age" -lt 18 ]; then

   echo "You are a minor, $name."

elif [ "$age" -lt 65 ]; then

   echo "You are an adult, $name."

else

   echo "You are a senior, $name."

fi
```

Task 14

Objective: Number comparison without complex conditionals.

Create "number_comparison.sh": `nano number_comparison.sh`

Content:

```
#! /bin/bash

echo "Enter first number:"

read num1

echo "Enter second number:"

read num2

echo "$num1 == $num2 : $((num1 == num2))"

echo "$num1 != $num2 : $((num1 != num2))"

echo "$num1 > $num2 : $((num1 > num2))"

echo "$num1 < $num2 : $((num1 < num2))"

echo "$num1 >= $num2 : $((num1 >= num2))"

echo "$num1 <= $num2 : $((num1 <= num2))"
```

Task 15

Objective: summation 1 to 10.

Create "sum_1_to_10.sh": `nano sum_1_to_10.sh`

Content:

```
#! /bin/bash

sum=0

for i in {1..10}

Do

      Echo “Enter the number $i : ”
```

```
      read n1
      sum=$((sum+n1))
done
echo “Total sum : $sum”
```

Task 16

Objective: Guess the Number.

Create "Guess_the_Number.sh": `nano Guess_the_Number.sh`

Content:

```
#!/bin/bash
sum=0
for i in {1..10}
Do
      Echo “Enter the number $i : ”
      read n1
      sum=$((sum+n1))
done
echo “Total sum : $sum”
```

Task 17

Objective:Simple calculator.

Create "simple_calculator.sh": `nano simple_calculator.sh`

Content:

```
#! /bin/bash
echo "Enter the first number:"
read num1
echo "Enter an operator (+, -, *, /):"
read operator
echo "Enter the second number:"
read num2

case $operator in
    "+")
        result=$(echo "$num1 + $num2" | bc)
        ;;
    "-")
        result=$(echo "$num1 - $num2" | bc)
        ;;
    "*")
        result=$(echo "$num1 * $num2" | bc)
        ;;
    "/")
        if [ "$num2" -eq 0 ]; then
            echo "Error: Division by zero is not allowed."
```

```
            exit 1
        fi
        result=$(echo "scale=2; $num1 / $num2" | bc)
        ;;
    *)
        echo "Invalid operator. Please use +, -, *, or /."
        exit 1
        ;;
esac
echo "Result: $result"
```

6. Conclusion

This lab provided hands-on experience with essential Unix/Linux commands and shell scripting techniques. Understanding the Unix/Linux directory structure, file permissions, user and process management commands, and network utilities equips system administrators with tools for efficient OS management. The introduction to shell scripting lays the groundwork for automating routine tasks and problem-solving in a Unix/Linux environment. These skills are foundational for advanced topics in operating systems and system administration.