



K.R. MANGALAM UNIVERSITY
THE COMPLETE WORLD OF EDUCATION

LINUX BASH SCRIPTING LAB

(ENCS356)

Lab file Submitted to

K.R. MANGALAM UNIVERSITY

For

Bachelor Of Technology

In

Computer Science Engineering Specialization

(Cyber Security)

Submitted By :

Srishti Mishra (2301410005)

Faculty:

Dr. Rajesh Kumar



Experiment No. 1

Exploring Linux System Commands, Run Levels (Targets), and Help Utilities.

Problem Statement

Efficient use of the Linux operating system requires a strong understanding of basic system commands, system states, and built-in help mechanisms. Unlike graphical operating systems, Linux relies heavily on command-line utilities for system interaction, administration, and automation.

The problem is to explore and practice essential Linux system commands, understand system run levels / targets, and effectively use Linux help utilities such as `man`, `--help`, and *auto-* completion to access command documentation and usage information.

Aim

To study and execute basic Linux system commands, understand Linux run levels (systemd targets), and use Linux help utilities such as `man`, `--help`, and tab auto-completion.

Theory

Linux Command Line Interface (CLI)

The Linux Command Line Interface allows users to interact with the operating system by typing commands. It provides better control, flexibility, and efficiency compared to graphical interfaces, especially for system administration and automation.

System Run Levels / Targets

- Modern Linux systems use **systemd**, where traditional run levels are replaced by **targets**.

Common systemd targets:

- `graphical.target` – Multi-user mode with GUI
- `multi-user.target` – Multi-user mode without GUI
- `rescue.target` – Single-user rescue mode
- `poweroff.target` – Shutdown system
- `reboot.target` – Reboot system

Linux Help Utilities

Linux provides built-in help tools for learning commands:

- `man` – Displays complete manual pages
- `--help` – Shows brief command usage
- **Tab auto-completion** – Completes commands and file names automatically

Commands Used

Commands	Description
<code>pwd</code>	Display present working directory
<code>ls</code>	List files and directories
<code>cd</code>	Change directory
<code>whoami</code>	Display current user
<code>date</code>	Show system date and time
<code>uptime</code>	Show system running time
<code>man</code>	Display manual pages
<code>--help</code>	Display help for a command
<code>systemctl</code>	Manage system services and targets
<code>runlevel</code>	Display previous and current run level
<code>clear</code>	Clear terminal screen

Procedure / Steps

Step 1: Login:

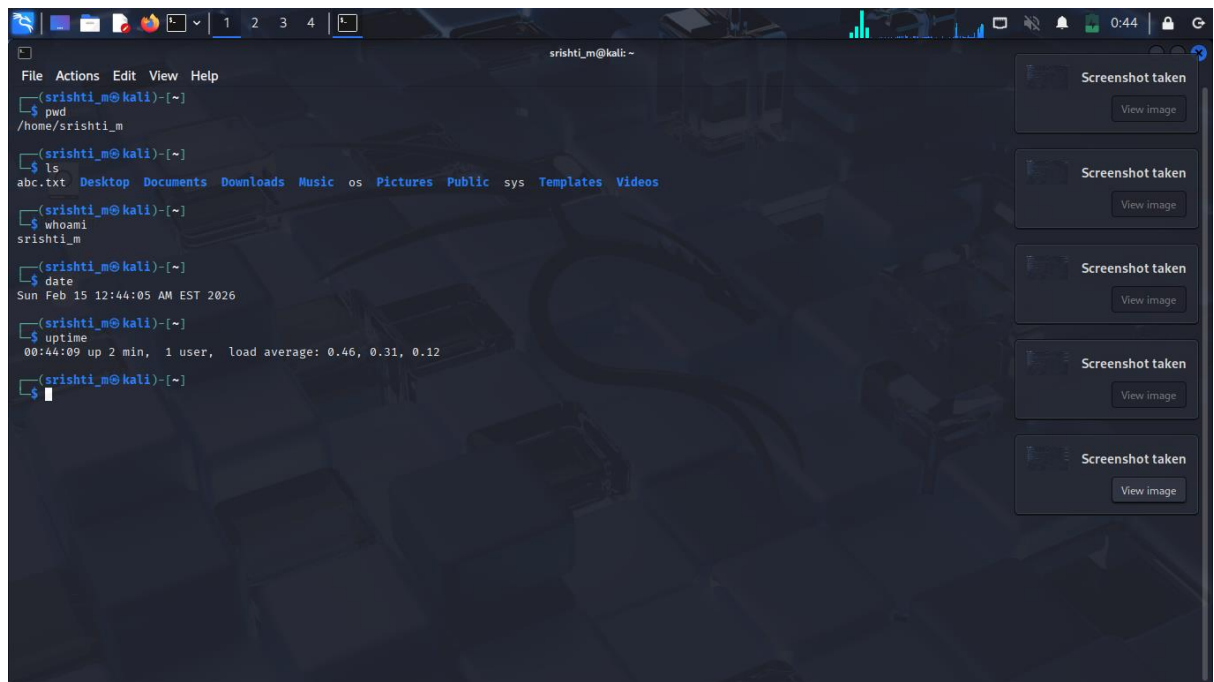
Log in to the Linux virtual machine and open the **Terminal**.

Step 2: Execute Basic Linux Commands in the Terminal.

Commands:

- `pwd` shows current directory
- `ls` lists files
- `whoami` displays logged-in user
- `date` shows system date/time

- uptime shows how long system is running

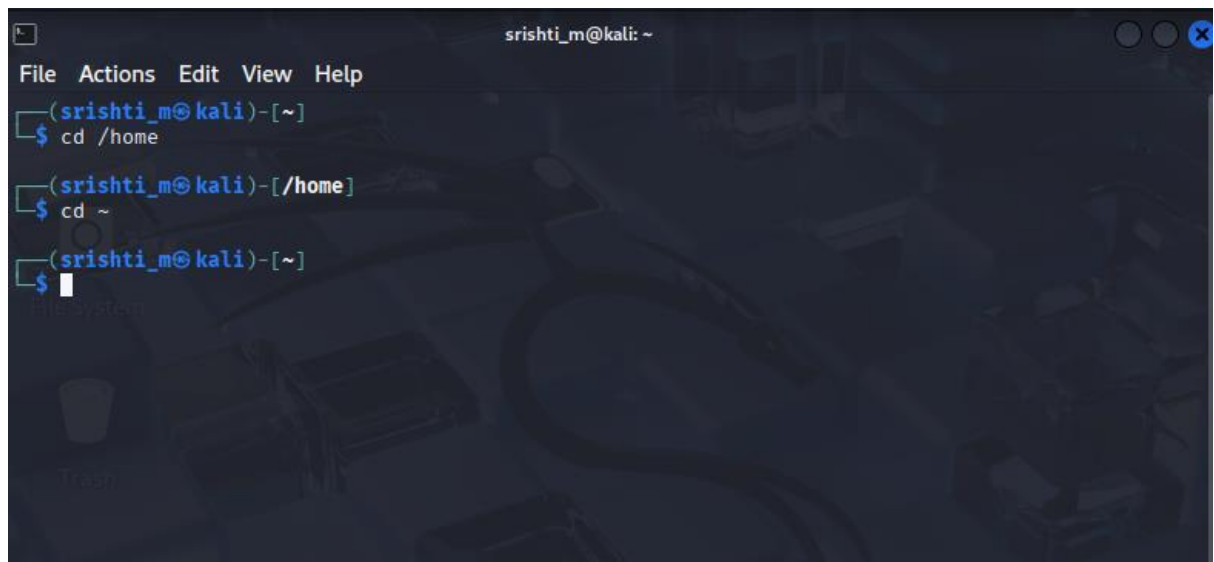


```

srishti_m@kali: ~
File Actions Edit View Help
(srishti_m@kali)-[~]
$ pwd
/home/srishti_m
(srishti_m@kali)-[~]
$ ls
abc.txt Desktop Documents Downloads Music os Pictures Public sys Templates Videos
(srishti_m@kali)-[~]
$ whoami
srishti_m
(srishti_m@kali)-[~]
$ date
Sun Feb 15 12:44:05 AM EST 2026
(srishti_m@kali)-[~]
$ uptime
00:44:09 up 2 min, 1 user, load average: 0.46, 0.31, 0.12
(srishti_m@kali)-[~]
$
  
```

Step 3: Directory Navigation using:

- `cd /home` moves to home directory.
- `cd ~` returns to user's home directory.



```

srishti_m@kali: ~
File Actions Edit View Help
(srishti_m@kali)-[~]
$ cd /home
(srishti_m@kali)-[/home]
$ cd ~
(srishti_m@kali)-[~]
$
  
```

Step 4: Using Manual Pages

- `man ls`
- `man cd`

It Read command details.

➤ Press `q` to exit the manual.

```
(srishti_m@kali)-[~]  
$ man ls  
  
(srishti_m@kali)-[~]  
$ man cd  
No manual entry for cd  
  
(srishti_m@kali)-[~]  
$
```

```
(srishti_m@kali)-[~]  
$ man ls  
  
(srishti_m@kali)-[~]  
$ man cd  
No manual entry for cd  
  
(srishti_m@kali)-[~]  
$
```

```
srishti_m@kali: ~  
File Actions Edit View Help  
LS(1) User Commands LS(1)  
  
NAME  
    ls - list directory contents  
  
SYNOPSIS  
    ls [OPTION]... [FILE]...  
  
DESCRIPTION  
    List information about the FILES (the current directory by default). Sort entries al-  
    phabetically if none of -cftuvSUX nor --sort is specified.  
  
    Mandatory arguments to long options are mandatory for short options too.  
  
    -a, --all  
        do not ignore entries starting with .  
  
    -A, --almost-all  
        do not list implied . and ..  
  
    --author  
        with -l, print the author of each file  
  
    -b, --escape  
        print C-style escapes for nongraphic characters  
  
    --block-size=SIZE  
        with -l, scale sizes by SIZE when printing them; e.g., '--block-size=M'; see SIZE  
        format below  
  
Manual page ls(1) line 1 (press h for help or q to quit)
```


Step 5: Using Help Option:

- `ls -help`
- `systemctl -help`

This displays short usage instructions and optio

```
(srishti_m@kali)-[~]
$ man cd
No manual entry for cd

(srishti_m@kali)-[~]
$ ls --help
Usage: ls [OPTION]... [FILE]...
List information about the FILES (the current directory by default).
Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.

Mandatory arguments to long options are mandatory for short options too.
  -a, --all                do not ignore entries starting with .
  -A, --almost-all        do not list implied . and ..
                        with -l, print the author of each file
  -b, --escape             print C-style escapes for nongraphic characters
                        with -l, scale sizes by SIZE when printing them;
                        e.g., '--block-size=M'; see SIZE format below
  -B, --ignore-backups    do not list implied entries ending with ~
  -c                      with -lt: sort by, and show, ctime (time of last
                        change of file status information);
                        with -l: show ctime and sort by name;
                        otherwise: sort by ctime, newest first

  -C                      list entries by columns
  --color[=WHEN]          color the output WHEN; more info below
  -d, --directory         list directories themselves, not their contents
  -D, --dired             generate output designed for Emacs' dired mode
  -f                      same as -a -U
  -F, --classify[=WHEN]  append indicator (one of */=>@|) to entries WHEN
                        --file-type          likewise, except do not append '*'

File Actions Edit View Help

(srishti_m@kali)-[~]
$ systemctl --help
systemctl [OPTIONS ...] COMMAND ...

Query or send control commands to the system manager.

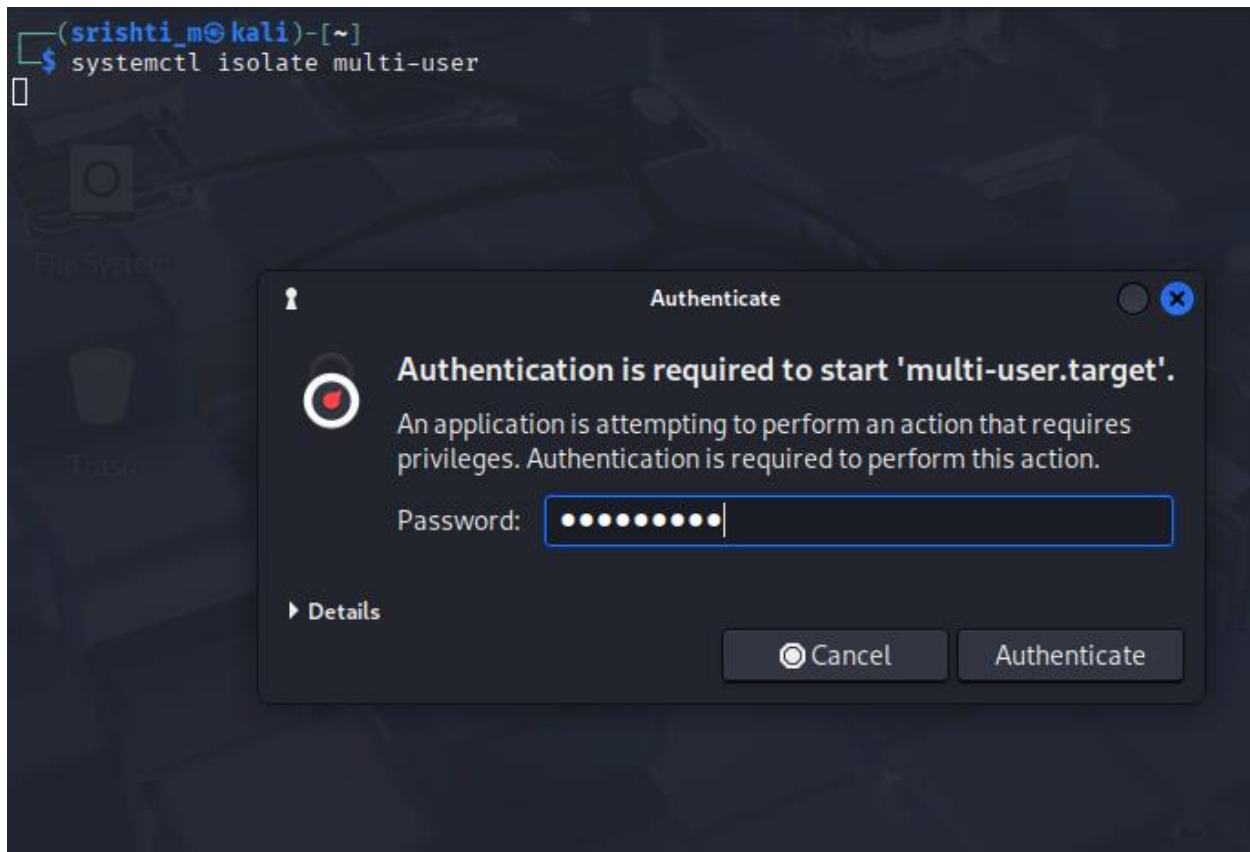
Unit Commands:
list-units [PATTERN ...]      List units currently in memory
list-automounts [PATTERN ...] List automount units currently in memory,
                                ordered by path
list-paths [PATTERN ...]      List path units currently in memory,
                                ordered by path
list-sockets [PATTERN ...]     List socket units currently in memory,
                                ordered by address
list-timers [PATTERN ...]      List timer units currently in memory,
                                ordered by next elapse
is-active PATTERN ...          Check whether units are active
is-failed [PATTERN ...]        Check whether units are failed or
                                system is in degraded state
status [PATTERN ... |PID ...]  Show runtime status of one or more units
show [PATTERN ... |JOB ...]    Show properties of one or more
                                units/jobs or the manager
cat PATTERN ...                Show files and drop-ins of specified units
help PATTERN ... |PID ...      Show manual for one or more units
list-dependencies [UNIT ...]   Recursively show units which are required
                                or wanted by the units or by which those
                                units are required or wanted
start UNIT ...                 Start (activate) one or more units
stop UNIT ...                  Stop (deactivate) one or more units
reload UNIT ...                Reload one or more units
```

Step 6: Check System Run Level / Target

- `runlevel`- “Kali Linux uses systemd, so traditional runlevel commands are deprecated and `systemctl` targets are used instead.”
- `systemctl get-default` shows default system target.

```
(srishti_m@kali)-[~]  
$ runlevel  
unknown  
  
(srishti_m@kali)-[~]  
$ systemctl get-default  
graphical.target
```

```
srishti_m@kali: ~  
File Actions Edit View Help  
(srishti_m@kali)-[~]  
$ systemctl isolate multi-user
```



File Actions Edit View Help

<code>status [PATTERN... PID ...]</code>	Show runtime status of one or more units
<code>show [PATTERN... JOB ...]</code>	Show properties of one or more units/jobs or the manager
<code>cat PATTERN...</code>	Show files and drop-ins of specified units
<code>help PATTERN... PID ...</code>	Show manual for one or more units
<code>list-dependencies [UNIT...]</code>	Recursively show units which are required or wanted by the units or by which those units are required or wanted
<code>start UNIT...</code>	Start (activate) one or more units
<code>stop UNIT...</code>	Stop (deactivate) one or more units
<code>reload UNIT...</code>	Reload one or more units
<code>restart UNIT...</code>	Start or restart one or more units
<code>try-restart UNIT...</code>	Restart one or more units if active
<code>reload-or-restart UNIT...</code>	Reload one or more units if possible,

```
—(srishti_m@kali)-[~]
$ runlevel
unknown

—(srishti_m@kali)-[~]
$ systemctl get-default
graphical.target

—(srishti_m@kali)-[~]
$ systemctl isolate graphical.target

—(srishti_m@kali)-[~]
$ systemctl isolate graphical.target

—(srishti_m@kali)-[~]
$
```


Step 7:- Practice auto-completion by typing partial command names and pressing Tab.

Step 8 :- Clear the terminal screen: Clear

Outcomes

- Successful execution of basic Linux commands
- Display of manual pages and help options
- Identification of current run level/target
- Improved command-line efficiency

Result

Thus, basic Linux system commands, system run levels, and help utilities were successfully explored. The student gained practical understanding of Linux command-line interaction and Linux documentation tools.