

# School of Engineering & Technology

---

*Department: Computer Science and Engineering*

## **Assignment No. 02: Basics of Linux and Open-Source Tools**

Course: Computer Science Fundamentals & Career Pathways (ETCCCP105)

Programme: B.Tech CSE

Semester: 1

### **Submitted By:**

Name: Ankur Negi

University: KR Mangalam University

Roll. No: 2501010236

Faculty: Dr. Rajesh Kumar

Session: Odd

Date: November 16, 2025

## 1. Introduction

The purpose of this assignment is to gain practical understanding of Linux, its installation using Windows Subsystem for Linux (WSL), and the use of shell commands and scripting for automation. Linux is widely used in software development, cybersecurity, and cloud computing environments due to its stability, flexibility, and open-source nature. This assignment also explores the integration of version control using GitHub for managing and hosting code repositories.

## 2. Step 1 – Linux Installation using WSL

In this step, Ubuntu was installed using Windows Subsystem for Linux (WSL) on Windows 11. This allows Linux to run natively alongside Windows without requiring a virtual machine.

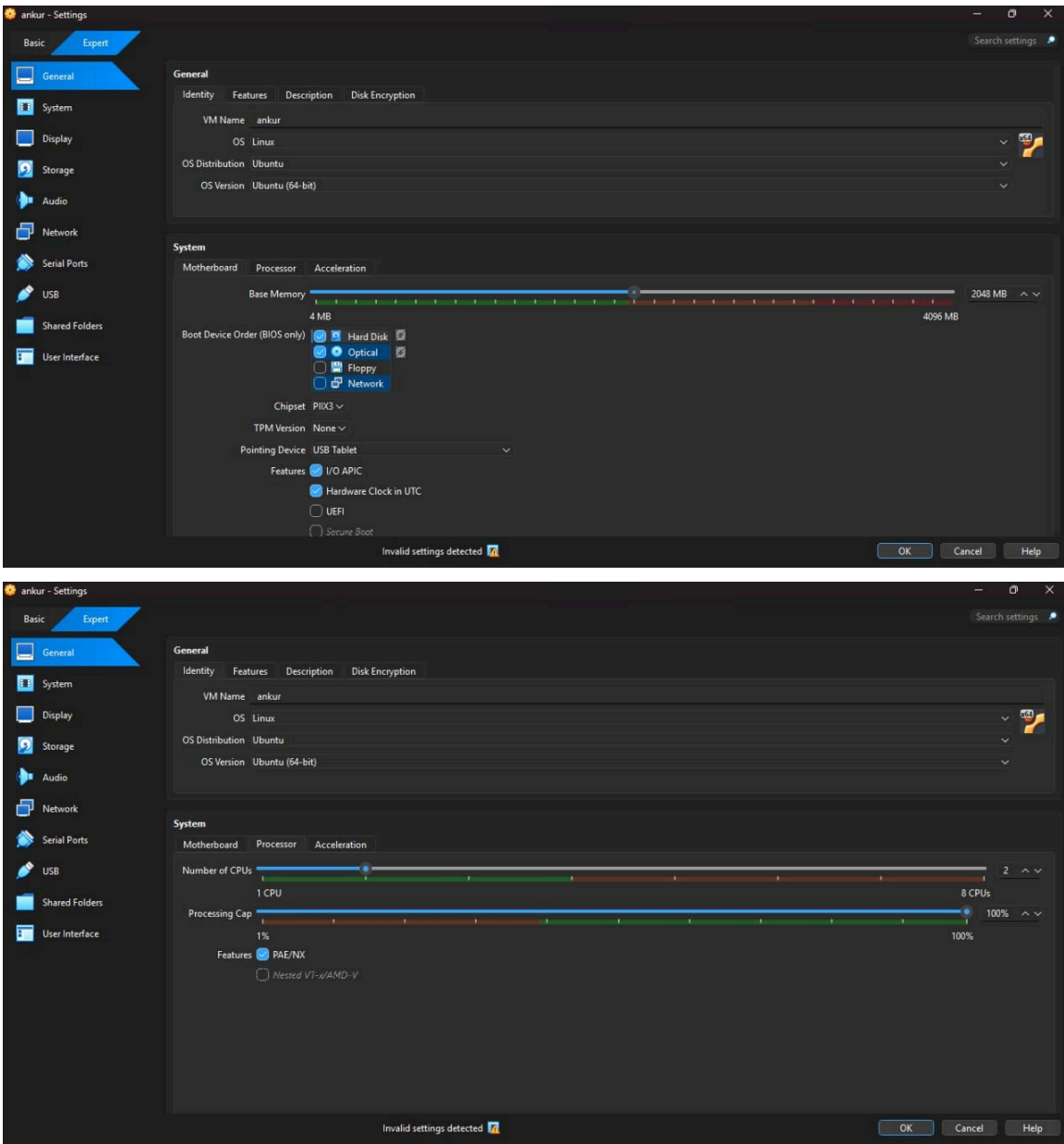
### **\*\*Hardware Configuration:\*\***

- CPU: Intel Core i5 (4 cores)
- RAM: 4 GB
- Disk Space Allocated: 25 GB

### **\*\*Installation Steps:\*\***

1. Open Microsoft Store and search for 'Ubuntu'.
2. Click 'Get' to download and install Ubuntu on Windows.
3. Once installed, launch Ubuntu from the Start menu.
4. Set up a username and password when prompted.
5. Run ``sudo apt update && sudo apt upgrade`` to update all packages.
6. Confirm installation by checking the version using ``lsb_release -a``.

[Screenshot Placeholder: Ubuntu Installation and Terminal Output]



### 3. Step 2 – Shell Command Implementation

The following table lists 20 commonly used shell commands in Linux. Each command includes its syntax, description, and example usage.

Command: ls

Description: Lists files and directories in the current path

Example: ls -l

Command: cd

Description: Changes the current directory

Example: cd /home/user

Command: pwd

Description: Displays the current working directory

Example: pwd

Command: tree

Description: Displays directory structure in tree format

Example: tree

Command: mkdir

Description: Creates a new directory

Example: mkdir new\_folder

Command: touch

Description: Creates an empty file

Example: touch file.txt

Command: cp

Description: Copies files or directories

Example: cp file1.txt backup/

Command: mv

Description: Moves or renames files or directories

Example: mv file.txt newname.txt

Command: rm

Description: Removes files or directories

Example: rm file.txt

Command: chmod

Description: Changes file permissions

Example: chmod 755 script.sh

Command: chown

Description: Changes file owner

Example: sudo chown user:user file.txt

Command: ps

Description: Displays active processes

Example: ps aux

Command: top

Description: Displays real-time system processes

Example: top

Command: kill

Description: Terminates a process by ID

Example: kill 1234

Command: ping

Description: Tests network connectivity

Example: ping google.com

Command: ifconfig

Description: Displays network interface information

Example: ifconfig

Command: ip

Description: Alternative to ifconfig for modern systems

Example: ip a

Command: netstat

Description: Displays network connections

Example: netstat -tulnp

Command: echo

Description: Prints text or variables to terminal

Example: echo 'Hello Linux'

Command: cat

Description: Displays contents of a file

Example: cat file.txt

## 4. Step 3 – Shell Script Development

**\*\*1. Backup a Directory\*\***

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

```
# Author: Ankur Negi
```

```
# Date: $(date)
```

```
# Purpose: Backup a directory with timestamp
```

```
src_dir="/home/user/documents"
```

```
backup_dir="/home/user/backup"
```

```
timestamp=$(date +%Y%m%d_%H%M%S)
```

```
mkdir -p $backup_dir
cp -r $src_dir $backup_dir/backup_$(date +%Y%m%d_%H%M%S)
```

```
echo "Backup completed at $(date +%Y%m%d_%H%M%S)"
```

[Screenshot Placeholder: Backup Script Execution]

**\*\*2. CPU/Memory Monitoring\*\***

```
#!/bin/bash
# Author: Ankur Negi
# Purpose: Logs CPU and memory usage

log_file="/home/user/system_log.txt"
while true; do
    echo "$(date) - CPU and Memory Usage:" >> $log_file
    top -bn1 | head -n 5 >> $log_file
    sleep 60
done
```

**\*\*3. Automated Download Task\*\***

```
#!/bin/bash
# Author: Ankur Negi
# Purpose: Download file using wget

url="https://example.com/sample.txt"
destination="/home/user/downloads"
mkdir -p $destination
wget -P $destination $url
echo "File downloaded successfully to $destination"
```

## 5. Step 4 – GitHub Repository Submission

A GitHub repository named **\*\*linux-shell-assignment\*\*** was created to store all shell scripts and documentation files. The repository includes a README.md file containing project overview, usage instructions, and screenshots.

Repository Link: <https://github.com/negiji892-tech/linux-shell-assignment.git>

## 6. Reflection

During this assignment, I learned the fundamentals of Linux installation, command usage, and shell scripting for automation. I faced challenges in understanding permissions and process management initially but overcame them through practice. This experience

strengthened my understanding of Linux as a developer environment and improved my problem-solving skills. The ability to automate repetitive tasks using scripts is particularly valuable in real-world system administration and DevOps applications.