

# CSE 4502: Operating System Lab

## Lab 2: Intermediate Linux Commands

Jibon Naher, Lecturer, CSE

November 27, 2025

### Introduction

This lab follows Lab 1 and introduces **30 intermediate and advanced commands** essential for text processing, data compression, system monitoring, and process management. Successful completion requires demonstrating the use of command chaining and I/O control.

---

### Introductory Survey - 20 minutes

1. Please go to the link: Introductory Student Survey 2. Fill up and submit the survey

## 1 Part 1: Preparation, Help, and System Information (6 Commands)

### Objective

Set up the workspace, utilize built-in help features, and retrieve basic system information.

### Tasks

- Create a main directory `lab2` and a sub-directory `text_data`.
- Use the appropriate command to display the manual page for the `tar` command.
- Display the last 10 commands you executed.

Table 1: Commands for Help and Metadata

No.	Command	Description
1	<code>mkdir</code>	Create the workspace ( <code>lab2</code> and <code>text_data</code> ).
2	<code>cd</code>	Navigate into the <code>lab2</code> directory.
3	<code>man</code>	Display the manual page for a command (e.g., <code>man tar</code> ).
4	<code>history</code>	View the list of previously executed commands.
5	<code>stat</code>	Display detailed file status and metadata (Check metadata for <code>text_data</code> ).
6	<code>uname</code>	Print system information (e.g., kernel name/version).

## 2 Part 2: Text Processing and Filters (8 Commands)

### Objective

Practice filtering, counting, sorting, and transforming text data.

### Tasks

- Create two simple text files (`fileA.txt`, `fileB.txt`) with minor differences.
- Combine the content of both files, sort the lines, and count the total lines.
- Remove duplicate lines from a file after sorting it.

Table 2: Commands for Text Manipulation

No.	Command	Description
7	<code>wc</code>	Word Count (count lines, words, and bytes).
8	<code>sort</code>	Sort lines of text files.
9	<code>uniq</code>	Report or omit repeated lines (use after <code>sort</code> ).
10	<code>diff</code>	Find the differences between two files.
11	<code>cut</code>	Remove sections from each line of files (e.g., specific column data).
12	<code>tr</code>	Translate or delete characters (e.g., change all lowercase to uppercase).
13	<code>sed</code>	Stream Editor (Perform a basic substitution, e.g., change 'old' to 'new' in a file).
14	<code>more</code>	Pager utility to view file content page by page (Use for a large output).

## 3 Part 3: Archiving, Compression, and Linking (6 Commands)

### Objective

Compress data for storage and sharing, and understand file linking.

### Tasks

- Create an archive of the entire `text_data` folder and compress it using `.tar.gz` format.
- Decompress and extract the archive into a new directory `extracted_data`.
- Create a symbolic link (`-s`) to a file, and a hard link to another file.

## 4 Part 4: System Status and Process Control (6 Commands)

### Objective

Monitor system resource utilization and manage running processes.

Table 3: Commands for Archiving and Links

No.	Command	Description
15	<code>tar -czf</code>	Create a compressed archive (gzip format).
16	<code>tar -xf</code>	Extract files from an archive.
17	<code>gzip</code>	Compress a single file using the GZIP algorithm.
18	<code>gunzip</code>	Decompress a file previously compressed with gzip.
19	<code>ln -s</code>	Create a symbolic (soft) link to a file.
20	<code>ln</code>	Create a hard link to a file.

## Tasks

- Display the available and used disk space on your mounted file systems.
- View the memory usage statistics (RAM and Swap).
- Run the live process monitor and observe system performance for a few seconds.
- Use a command to display the network configuration of your interfaces.
- Use a command to find the process ID (PID) of a command you run in the background, and then terminate it.

Table 4: Commands for System Monitoring

No.	Command	Description
21	<code>df</code>	Disk Free (report file system disk space usage).
22	<code>free</code>	Display amount of free and used memory (RAM and swap).
23	<code>top</code>	Display Linux processes (Interactive system-monitor).
24	<code>ps</code>	Report a snapshot of the current processes.
25	<code>kill</code>	Send a signal (terminate) to a process using its PID.
26	<code>ip addr</code>	Display network interface addresses (Modern replacement for ifconfig).

## 5 Part 5: I/O Control, Aliases, and Utilities (4 Commands)

### Objective

Master command chaining, output redirection, and customize your environment.

## Tasks

- Use the pipe (|) to chain the output of one command to the input of another (e.g., `history | grep`).
- Use output redirection (>>) to append a message to an existing file.
- Create a temporary alias for a long command, test it, and then remove it.
- Display the system time and date, and then the calendar for the current month.

Table 5: Commands for I/O and Utilities

No.	Command	Description
27	alias / unalias	Create and remove command shortcuts.
28	>>	I/O Redirection operator (Append output to a file).
29		Pipe operator (Chain commands by passing output as input).
30	date / cal	Print the system date/time and display a calendar.

## Submission Requirements

1. **List of Commands:** A numbered list of all 30 commands used (matching the numbers in the tables).
2. **Syntax:** The exact command line syntax you used for each command (e.g., tar -czf archive.tar.gz text\_data/).
3. **Execution Proof:** A single, clear, scrolling screenshot (or multiple combined screenshots) showing the execution and output of **at least 10 commands**(randomly chosen) performed sequentially in your terminal. This shows the successful completion of the entire assignment flow.