



Bahir Dar University

Software engineering department

System Call Implementation

Name: Dagmawit Amsalu

ID No.1601218

Course: Operating system and system programming

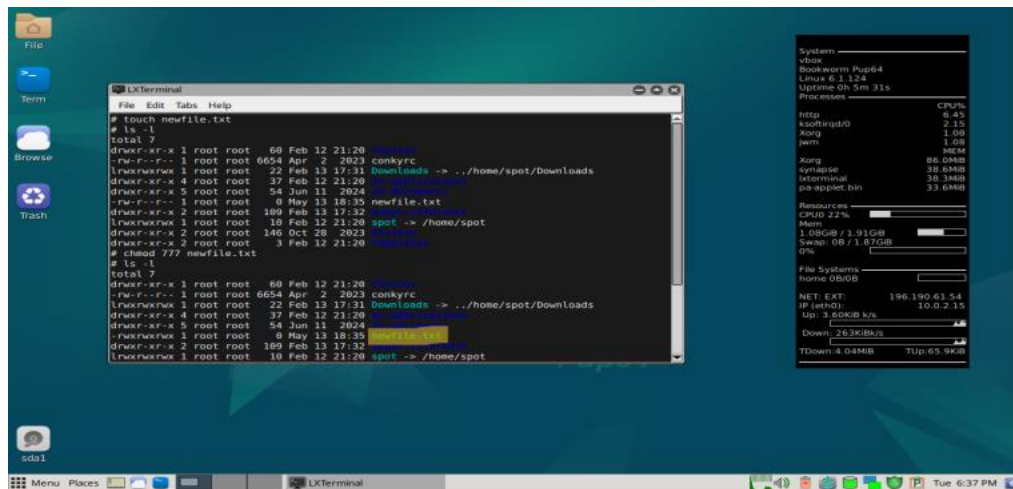
Submitted to: Lecturer Wendimu B.

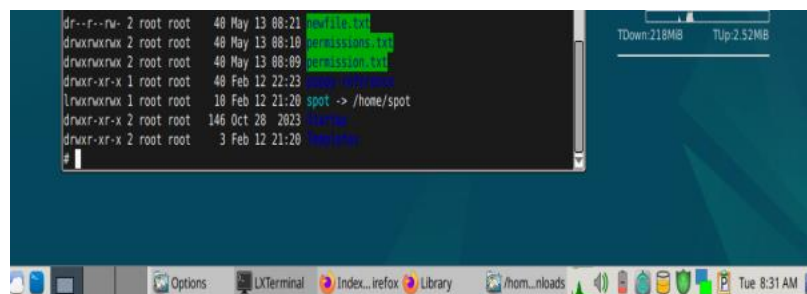
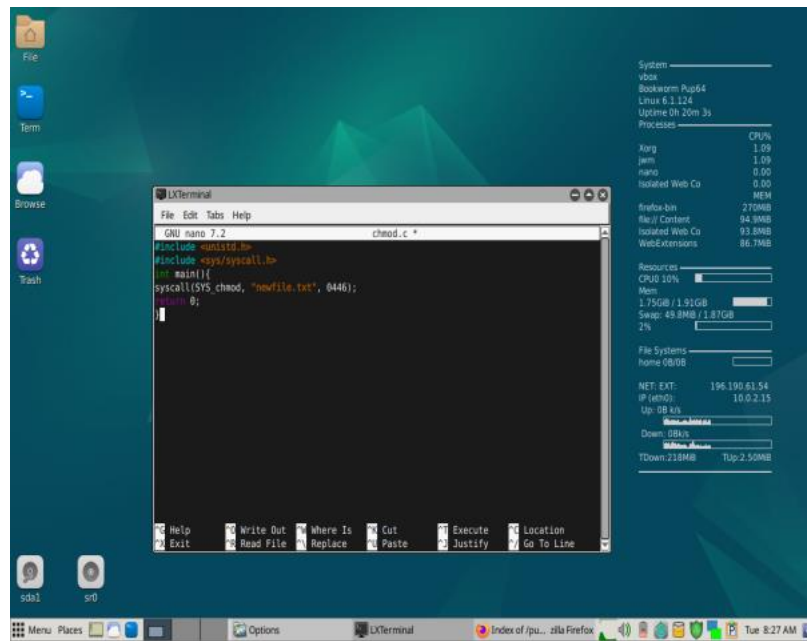
implementing a system call

A system call is how a program talks to the operating system. Programs can't do important things like opening files, reading from the keyboard, or using the internet directly. They have to ask the operating system (like Linux) to do it for them. That asking is done using system calls. Common System Calls:

- `open()` – open a file
- `read()` – read from a file
- `write()` – write to a file
- `close()` – close a file
- `fork()` – create a new process
- `exit()` – end a program

The system call i will be demonstrating is `chmod()` changing the permission of a file.





This C program uses the `syscall` function to directly invoke the `SYS_chmod` system call, which changes the permissions of a file. The program attempts to modify the permissions of a file named `newfile.txt`, setting the permissions to `0446`, which means the owner has read and write permissions, the group has read permissions, and others also have read permissions. The system call is part of the low-level interface provided by the operating system, bypassing higher-level C library functions like `chmod()`. This demonstrates a more manual and low-level way of manipulating file permissions in Unix-like systems.