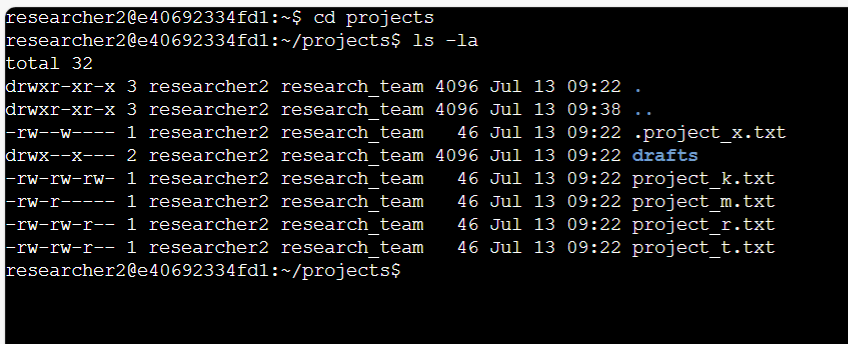
# File permissions in Linux

## Project Description

In this project, I managed file permissions in a Linux environment to ensure the security of a research team's data. My task was to examine and update file permissions to align with organizational policies, ensuring that only authorized users had the necessary access.

## Check file and directory details

I used the ls -la command to check the current file and directory permissions. This command lists all files and directories in the specified path, including hidden files, and displays their permissions.



## Describe the permissions string

The 10-character string representing file permissions consists of:

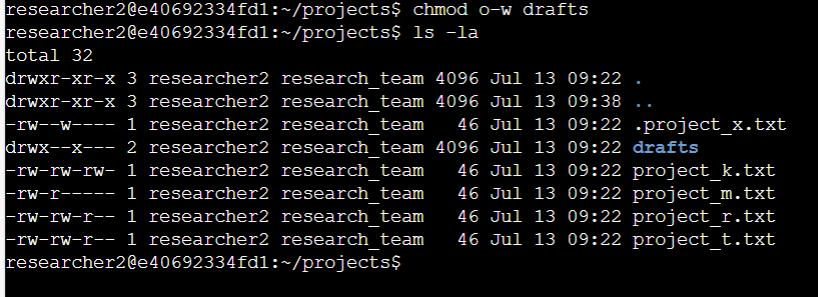
* The file type (e.g., - for a regular file, d for a directory).
* The permissions for the owner, group, and others.

For example, -rw-r--r-- means:

* -: Regular file
* rw-: Owner can read and write
* r--: Group can read
* r--: Others can read

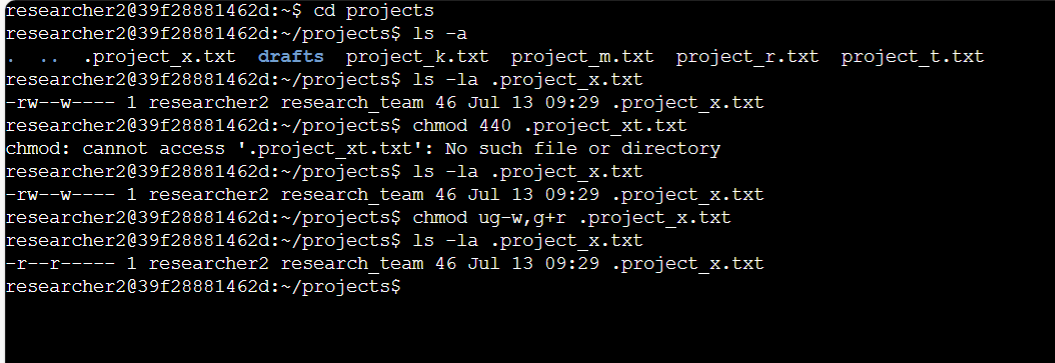
## Change file permissions

To remove write access for others I used “chmod o-w drafts” command



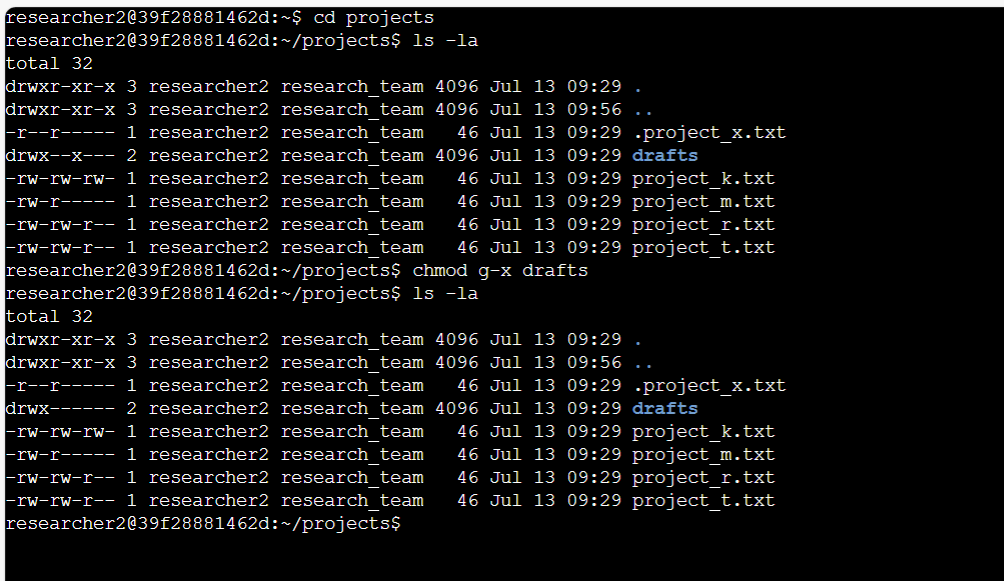
## Change file permissions on a hidden file

For the hidden file .project\_x.txt, assign read permissions to the user and group, and no write permissions for anyone:



## Change directory permissions

To ensure only researcher2 has access to the drafts directory:



## Summary

In this project, I checked the current file permissions using ls -la, identified unauthorized permissions, and modified them using chmod. I ensured that the hidden file .project\_x.txt had appropriate read permissions and restricted access to the drafts directory for the user researcher2. These steps helped secure the file system and align permissions with organizational requirements, ensuring that only authorized users had the necessary access.