# File permissions in Linux

## Project description

In this project, I executed a series of Linux commands to manage file permissions within the /home/researcher2/projects directory, focusing on enhancing data security and access control. Initially, I assessed the current directory structure and file permissions using ls -la, identifying areas where security improvements were necessary. The table below summarizes the initial permissions and the security policies implemented for each file and directory:

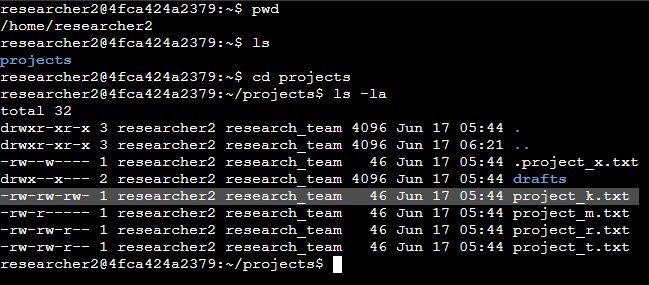
Check file and directory details:

|  |  |  |
| --- | --- | --- |
| File Name | Current Permissions | Security Policy Implemented |
| project\_k.txt | User = read, write  Group = read, write  Other = read, write | Adjusted to restrict write permissions for ‘Other’ users only |
| project\_m.txt | User = read, write  Group = read  Other = none | Revoked read permissions for ‘Group’ and ‘Other’ users, ensuring exclusive access for ‘User’ (reasercher2) |
| .project\_x.txt | User = read, write  Group = write  Other = none | Adjusted to remove write permissions for ‘Users’ and ‘Group’, while retaining read access for both |
| drafts | User = read, write, execute  Group = execute  Other = none | Restructured to remove execute permissions for ‘Group’, ensuring exclusive execute privileges for ‘User’ (researcher2) |

## First, I ran pwd to check my current directory. If I am in the right path (/home/researcher2), I then display the directory contents using ls to check if the projects directory exists. After confirming its existence, I navigate into the projects directory using cd projects, and finally run ls -la to display the permissions of all files, including hidden ones.

## In this task, I must determine whether any files have incorrect permissions and then change the permissions as needed. This action will remove unauthorized access and strengthen security on the system. None of the files should allow other users to write to files. The file that had write permissions for others was project\_k.txt.

## Commands:



## Describe the permissions string

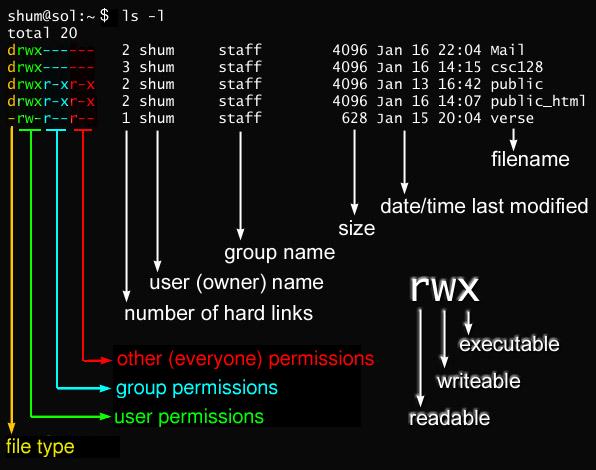
## Let's take the permissions string for project\_k.txt as an example: -rw-rw-rw-.

## File type: The first character indicates the type of the file. A - means it is a regular file. If it was a directory, it would be a d.

## User permissions: The next three characters (rw-) indicate the permissions for the user (owner) of the file. r means readable, w means writable, and - means not executable.

## Group permissions: The next three characters (rw-) indicate the permissions for the group.

## Other permissions: The final three characters (rw-) indicate the permissions for others (everyone else).



Source: <https://remy.parkland.edu/~smauney/csc128/fig_permissions.jpg>

## Here is a breakdown of the full ls -la output for one line:

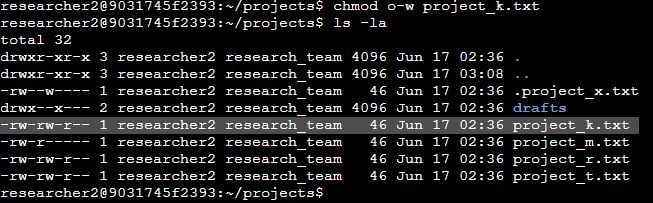
## 

## File type and permissions -rw-rw-rw-

* Number of hard link 1
* User (owner) name: researcher2
* Group name: research\_team
* Size: 46 (in bytes)
* Date/time last modified: Jun 17 05:44
* File name: project\_k.txt

The file project\_k.txt is owned by researcher2 and belongs to the research\_team group. It has a size of 46 bytes and was last modified on June 17th at 05:44. The file permissions -rw-rw-rw- indicate that researcher2 and members of the research\_team group have both read and write access to the file, while all other users also have read and write access. To adhere to security policies, the permissions of project\_k.txt should be adjusted to ensure that other users do not have write access, thereby maintaining data integrity and security within the system.

## Change file permissions for project\_k.txt

After identifying that project\_k.txt granted write access to all users, I promptly ran chmod o-w project\_k.txt to restrict write permissions to only the file's owner (researcher2) and members of the group (research\_team). This adjustment ensures compliance with security standards, safeguarding data integrity. I immediately verified the changes with ls -la. 

## Change file permissions for project\_m.txt

The file project\_m.txt is a restricted file that should not be readable or writable by the group or others; only the user (researcher2) should have these permissions. After confirming the current directory contents and permissions with ls -la, I executed chmod g-r project\_m.txt to revoke read permissions for the group. This adjustment ensures that only the owner can read or modify project\_m.txt, enhancing data security and access control.

## 

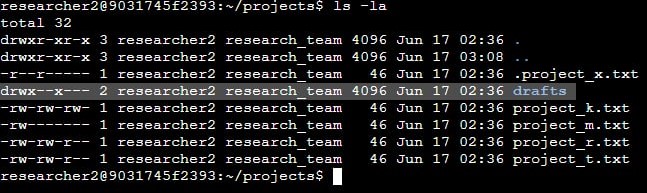
## Change file permissions for .project\_x.txt

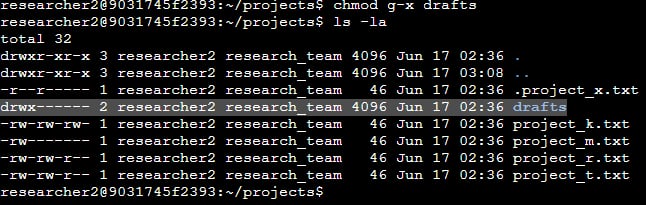
The file .project\_x.txt is a hidden file that has been archived and should not be writable by anyone, while allowing both the user and group to read it. To achieve this, I utilized chmod u-w,g-w,g+r .project\_x.txt command. This command removes write permissions for the owner and group (u-w,g-w) while adding read permissions for the group (g+r). This configuration ensures that the file remains accessible for reading while preventing accidental modifications, thus maintaining data integrity.

## 

## Change directory permissions

## The drafts directory should only be accessible to the researcher2 user, meaning only researcher2 should have execute privileges. To enforce this restriction, I executed chmod g-x drafts. This command removes execute permissions for the group on the drafts directory, ensuring that only the owner (researcher2) can access its contents. This measure enhances security by limiting access to sensitive data and project materials.





## Summary

In this project, I executed a series of essential tasks using Linux commands to manage file and directory permissions within the /home/researcher2/projects directory. Beginning with a thorough examination of current permissions using pwd, ls, and ls -la commands, I ensured that all files, including hidden ones, were correctly configured to align with security policies. Each file's permissions were meticulously reviewed, and adjustments were made where necessary to eliminate unauthorized access. Specifically, I modified permissions for project\_k.txt to restrict write access to only the file's owner and group members, ensuring data integrity and security. Additionally, I secured project\_m.txt by removing group read permissions and ensured .project\_x.txt remained readable but not writable by anyone except the owner and group. Finally, I restricted access to the drafts directory to only the researcher2 user by revoking execute permissions for the group. These actions collectively bolstered system security by strictly controlling access to sensitive files and directories, adhering to best practices in authorization management.