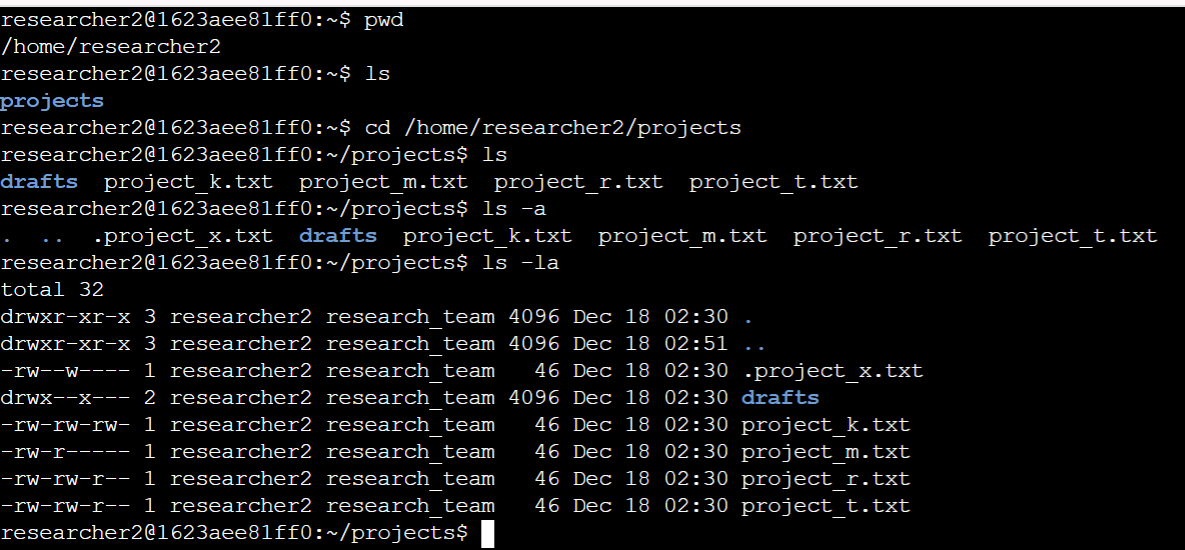
# File permissions in Linux

## Project description

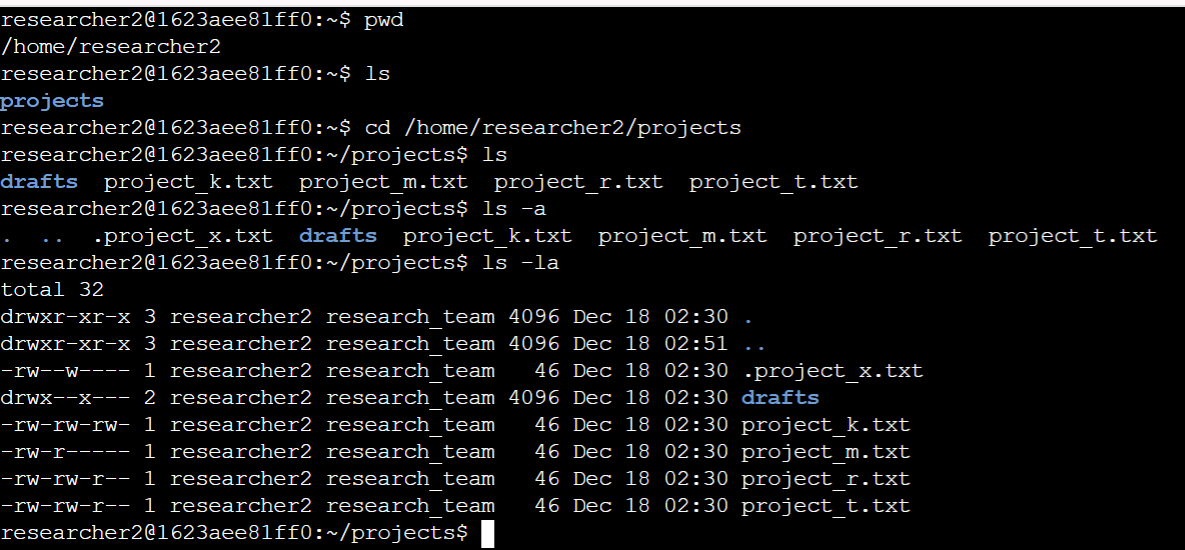
Through Linux commands, I will manage file permissions to ensure proper authorization and security within a system. This will include displaying current permissions, interpreting permission strings, and modifying access levels for files and directories. By using commands like ls -la and chmod, I will ensure that sensitive files and directories are appropriately restricted to authorized users only, aligning with organizational security policies.

## Check file and directory details



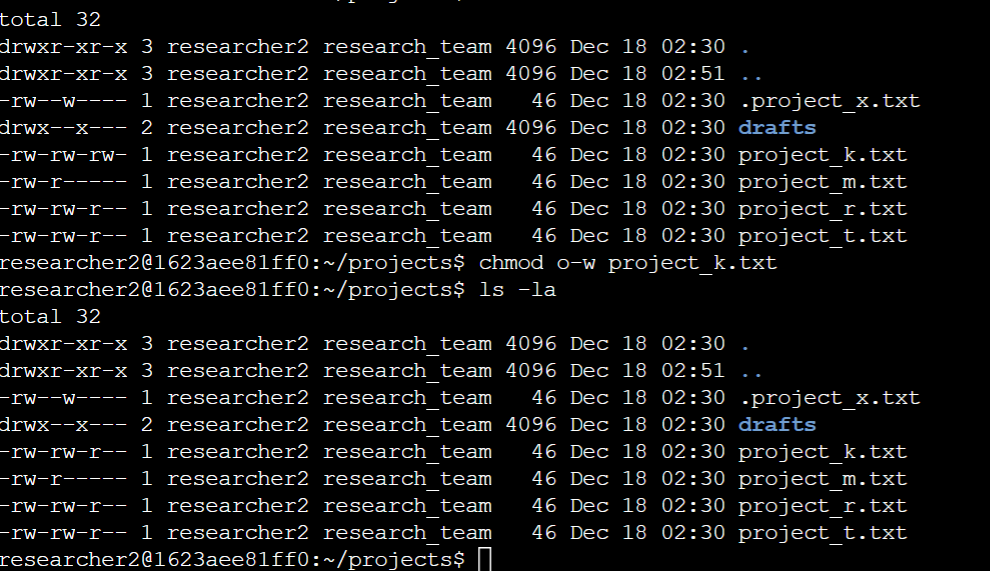
* The first thing I did was to ensure I was in the correct parent directory using the “pwd” command
* Then I navigated to the focus directory itself, “projects”, using the “cd” command and displayed the content of the subdirectory using “ls”
* The directory also contained some hidden files, using the “ls -a” command, I ensured those were also displayed
* With me being able to see all the subdirectories and files in the “projects” directory, I could efficiently check the directory details to see the permissions of every file present. I did that using the “ls -la” command, which displays the permission for every content in the “projects” directory including the hidden files or subdirectories.

## Describe the permissions string



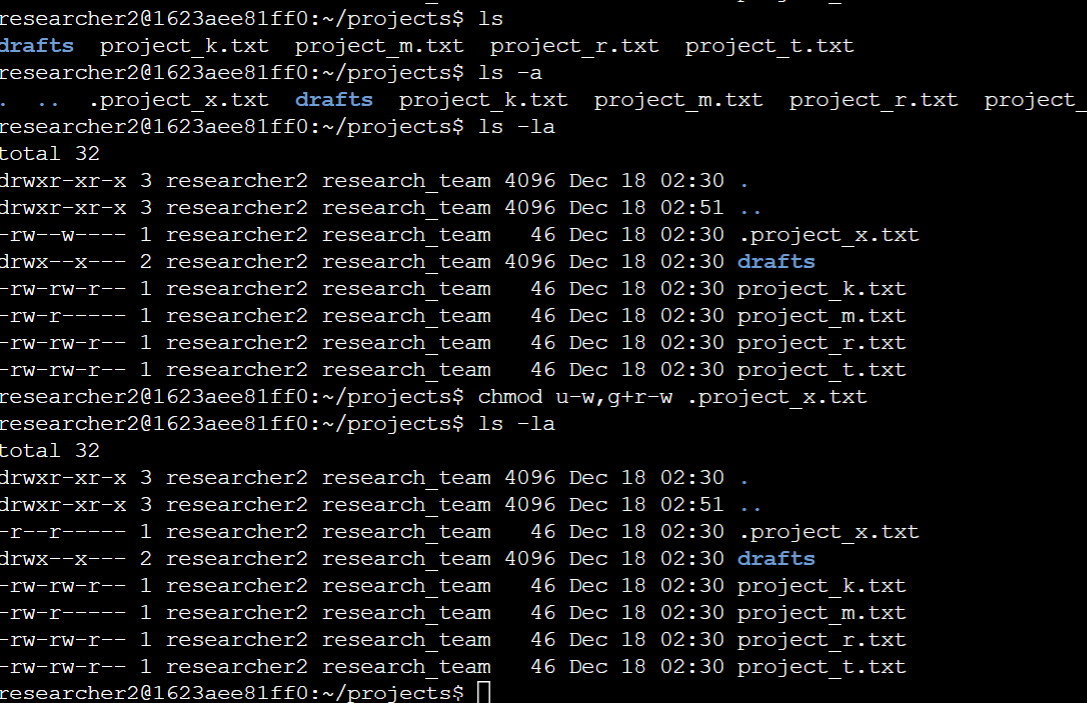
* Each subdirectory or file present after displaying the permissions tell a story, the story is written in 10-character string
* Using the “draft” subdirectory as an example, we see the permission been given as “drwx--x---"
* Here, the “d” in the string indicated that that is a directory, the next three characters “rwx” represent the permission given to the user. Here rwx stands for that the user can read, write and execute contents in this subdirectory. The next three characters “--x” represent the permission given to the user’s group, that is members of the “user’s organizational group”. Here the “-“, represent no permission in that section has been given to the user’s group, for this we see that the group cannot read or write on contents in this directory, but they have “x”, which mean they can execute content in this directory, what this means is that they can enter or access the directory. Finally, the last three characters “---” represent permission that would have been given “others”, which are people who are not the user or are not in the user’s group. As said earlier “-“ represent no permission, so from this we see that other people are not allowed to read, write or execute contents in this directory.

## Change file permissions



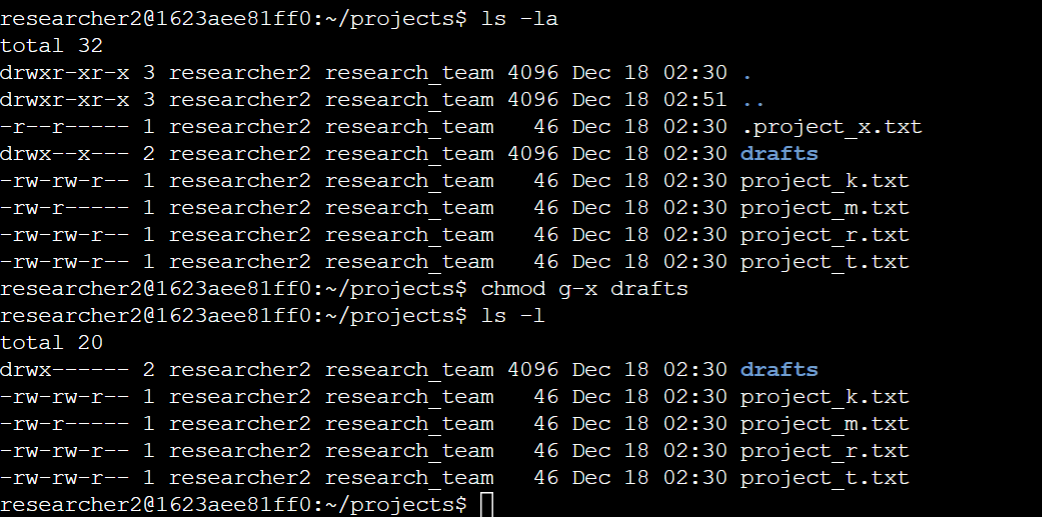
* Just as we are able to display the permissions in a directory or file, we can modify them as well to give or take away authorization.
* To do that we use the “chmod” command, which represent “Change Mode”.
* Using the example of the file “project\_k.txt”, if we want to take away the write permission from those in the others section, can achieve that using chmod and some arithmetic symbols.
* We use the “+” to give permission, “- “ to take away permission, and “=” to overwrite old permissions and give a new one.
* We can say “chmod u-w project\_k.txt”. What this simply means is that we are taking away the write permission from others in project\_k.txt.

## Change file permissions on a hidden file



* We can also do the same for hidden files, as can be seen from the screenshot, “.project\_x.txt” is a hidden file, and after displaying it and its permission using “ls -la”, we can also change permissions in it.
* For example, if we want to take away the write permission from everybody, and give only reading permission to the user and the user’s group, we can achieve that using the chmod command.
* “chmod u-w,g+r-w .project\_x.txt” or “chmod u-w,g=r .project\_x.txt” can be used.
* We use a comma to separate the user and the group.

## Change directory permissions



* As explained earlier we can also change the permission of a directory
* For example, we can choose to take away the “x”, execute permission for the user’s group in the draft directory
* Again, we use chmod, “chmod g-x drafts”
* With this we see that only the user has the permission to execute files here, in fact, he has all the permission meanwhile no other person has permission to read, write or execute content of this directory. This would be done for directories that contain sensitive information, which you don’t want anyone else to be able to see or make changes in them.

## Summary

This project demonstrates my ability to manage file permissions in a Linux environment, ensuring proper authorization and security for directories and files. I began by navigating to the correct directory and displaying its contents, including hidden files, using ls -la. I reviewed the 10-character permission strings to interpret and verify access levels for users, groups, and others. Using the chmod command, I modified permissions as required, taking away or assigning access based on security needs.

Specifically, I examined and adjusted permissions for regular files like project\_k.txt to remove write access for others and for hidden files such as .project\_x.txt to allow only reading for users and groups. I also restricted directory access by removing execute permissions for groups in sensitive directories like drafts. This activity showcases my ability to effectively interpret and manage permissions, ensuring secure file access aligned with organizational policies.