# File permissions in Linux

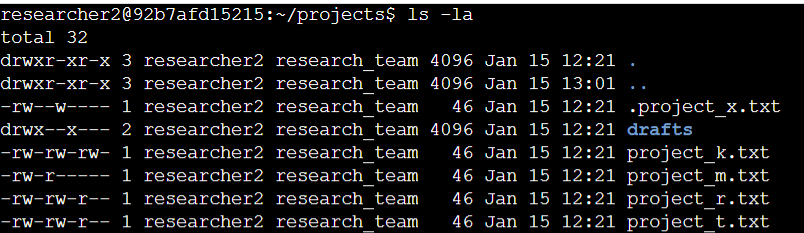
## Project description

This is a demonstration of a permissions review to comply with the company security policies. The policies are as follows:

The Other user group should not have writing permissions to any file, so we will proceed to remove them from project\_k.txt, and also remove from Group the permission to write on project\_m.txt. Furthermore, .project\_x.txt is an archived and hidden file, so no one should have writing permissions, but User and Group should have read permissions, so we will also update those.

Finally, we will ensure that only Researcher2 (the User) has access to the draft folder, as this is a personal folder.

## Check file and directory details



We are using the ls command, followed by -la:

* ls shows us the list of folders and files in the projects folder, where we are standing.
* -l shows the list of permissions on top of the files and folders.
* -a adds the hidden files to the list displayed by ls.
* By combining all of them, we can see the list of files and folders, complete with the hidden ones, plus all of their permissions.

## Describe the permissions string

The permissions string is composed of 10 characters which indicate whether it’s a file or a folder (d for folder, - for file), and whether the 3 types of users have permissions. r stands for read, w for write, and X for execute. Then, the characters 2 to 4 denote these permissions for the User, characters 5 to 7 denote these permissions for the User Group, and characters 8 to 10 denote it for the Other group. If there’s a - symbol, it means that group doesn’t have the permission.

In here, we can see the current permissions that every file (including hidden files) has on the system. Attached below you can see screenshots of the commands used, and finally the final result.

## Change file permissions



As you can see, we are using chmod here in order to remove permissions from the Other and Group user types. This is one way of using chmod to modify rights to a user group by adding or removing one permission.

## Change file permissions on a hidden file



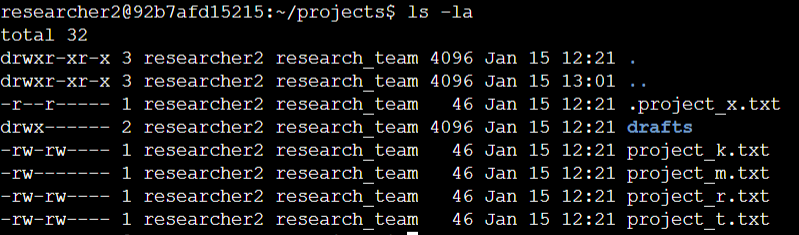
Here, for the hidden file (denoted by the dot at the beginning), we are instead using chmod to entirely overwrite the permissions of the user groups by using the = sign.

## Change directory permissions



We can also use chmod to modify the permissions relating to a folder. The logic is the same as before.

## Summary



As you can see, the final permissions follow the policies denoted in the beginning:

Drafts is only accessible for the User, while the Other group has no permissions to any files (but it does to the hidden folders, as these were not excluded from them in the original policies). Hidden files have also been accounted for.