# File permissions in Linux

## Project description

In this task, we will review the files and directories to adjust their permissions based on specific requirements. Our goal is to implement authorization following the principle of least privilege, ensuring that access is granted strictly on a need-to-know basis.

## Check file and directory details

So here we are in the project directory. The command ***ls -l*** helps us to review all the files and directory in the project directory.

A screen shot of a computer program

Description automatically generated

To look for all the files including hidden files permission we can use the command ***ls -la*** where option “**a**” can be used to list all files including hidden files. So combining it with option “**l**” we can review the permissions for all files including the hidden files.

A screen shot of a computer

Description automatically generated

## Describe the permissions string

Now to break it down, the permission is of 10 characters, The permissions are of 3 types Read, Write and Execute which can be represented as r, w and x respectively. The first character is either d or – where d stand for directory permission and –(hyphen) stands for file permissions. The next 3 character which is 2,3 and 4 character determines the permission for user, Similarly 5,6 and 7 character determines the permission for group and 8, 9 and 10 character determines the permission for others. The full flagged permission looks like ***drwxrwxrwx*** for directory and ***-rwxrwxrwx*** for files. And also if any group doesn’t have any particular permission it would change that permission letter to -.

After that the researcher2 is the name of the user and research\_team is the name of the group. And last we see the timestamp of any permission changed.

So here we will look into the current permissions for files and directory.

Here in the project directory we have 5 files and 1 directory.

* .project\_x.txt : Hidden file
  + **User = read, write**
  + **Group = write**
  + **Other = none**
* project\_k.txt : file
  + **User = read, write**
  + **Group = read, write**
  + **Other = read, write**
* project\_m.txt : file
  + **User = read, write**
  + **Group = read**
  + **Other = none**
* project\_r.txt : file
  + **User = read, write**
  + **Group = read, write**
  + **Other = read**
* project\_t.txt : file
  + **User = read, write**
  + **Group = read, write**
  + **Other = read**
* Drafts : Sub directory under porjects directory
  + **User = read, write, execute**
  + **Group = execute**
  + **Other = none**

## Change file permissions

Now we gonna change the file permissions for any files where the **other** users have the write permission to the file.

After closely looking on all permissions we see that project\_k.txt file has write permission for other users.



So we will remove the **other** user permission to write the file. We can do this with the **chmod** command with symbolic representation which allow us to make changes to the permissions of the files. Here we will use 3 symbols +, - and = where

+ -> Adds the permission

- -> Removes the permission

= -> Overwrites any permission the user have to the given permission

Also here we represents all 3 types of user with

u -> the primary user

g -> the group user

o -> other user

So to remove the write permission with chmod it takes 2 arguments

Chmod Argument1(Changes in permission that needs to be done, Multiple permission changes can be separated with the comma) Argument2(name of the file).   
***chmod o-w project\_k.txt***

A screen shot of a computer

Description automatically generated

Here we can see that after running the command it has removed the write access from the **other** user.

Similarly we can do for project\_m.txt where this file could only be readable or writeable by the user and not by the group so we can use chmod to remove the write permission of group user

***Chmod g-r project\_m.txt***



## Change file permissions on a hidden file



Here we are going to disable write permission from the user and the group as we gonna pretend that this files has been archieved and none should have access to write permission for the file but still both should be able to read the file.

***chmod u-w,g-w,g+r .project\_x.txt***

After running the following command we have successfully revoked all the permissions from the user and the group user.



## Change directory permissions



Here we are going to make the permission changes, Only the researcher2 user should be allowed to access the drafts directory and its contents. (This means that only researcher2 should have execute privileges.). Currently we see that even the group user have access to the execute permission which we have to revoke it. So we will run the command

***chmod g-x drafts***

After running the command we have successfully revoked the permission from the user.



## Summary

In conclusion, after thoroughly reviewing the files and directories, we successfully adjusted the permissions to align with the principle of least privilege. Access was granted only to those who require it, ensuring a more secure and efficient system. This approach minimizes potential security risks by limiting unnecessary access and maintaining a need-to-know basis for authorization.