

# File permissions in Linux

## Project description

This project is aimed to show the linux commands I learnt throughout the lessons. The content of this file consists of checking file and directory details, describing permissions based on the 10-letter string and changing file and directory permissions using chmod. Images are included to show a clearer context of what I had done to do the tasks.

### Scenario:

I am a member of a research team in an organization tasked to update file permissions for certain files and directories, where permissions do not currently reflect the level of authorization that should be given. Checking and modifying the permissions will help keep the system secure.

## Check file and directory details

```
researcher2@fbb39e9e41de:~$ pwd
/home/researcher2
researcher2@fbb39e9e41de:~$ cd projects
researcher2@fbb39e9e41de:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 May 22 04:35 .
drwxr-xr-x 3 researcher2 research_team 4096 May 22 05:25 ..
-rw--w---- 1 researcher2 research_team  46 May 22 04:35 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 May 22 04:35 drafts
-rw-rw-rw- 1 researcher2 research_team  46 May 22 04:35 project_k.txt
-rw-r----- 1 researcher2 research_team  46 May 22 04:35 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_t.txt
researcher2@fbb39e9e41de:~/projects$
```

To check the permission details of the files, I ran the following commands on the terminal. I intend to see the permissions of the files found in the projects directory using “ls -la”. This lists all the permissions of both hidden and non -hidden files in the directory.

```
researcher2@fbb39e9e41de:~/projects$ cd ..
researcher2@fbb39e9e41de:~$ ls -l
total 4
drwxr-xr-x 3 researcher2 research_team 4096 May 22 04:35 projects
researcher2@fbb39e9e41de:~$
```

To check the permissions of the “projects” directory, I had to exit the projects directory and run “ls -l”. It shows that “projects” directory allows full access to the user, and read and

execute for both group and other users. The command “ls -l” simply means that it only lists all the files or directories that are not hidden.

## Describe the permissions string

```
drwxr-xr-x 3 researcher2 research_team 4096 May 22 04:35 projects
```

We analyse the 10-letter string on the most left part of each row of the text to be able to see the permissions of each file or directories. For example, the “projects” directory allows users (2nd to 4th letter from the left) to read(r), write(w) and execute(x). While both the group and other users are only allowed to read and execute. The group ownership’s permissions are on the 5th - 7th letter from the left, while the last 3 three letters are the permissions that other users have. The first letter “d” indicates it is a directory., a hyphen(-) indicates it is a file.

Hyphen(-) on one of the permissions means that the specified owner type does not have permission for that particular action.

## Change file permissions

```
researcher2@fbb39e9e41de:~$ cd projects
researcher2@fbb39e9e41de:~/projects$ chmod o-w project_k.txt
researcher2@fbb39e9e41de:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 May 22 04:35 .
drwxr-xr-x 3 researcher2 research_team 4096 May 22 05:25 ..
-rw--w---- 1 researcher2 research_team  46 May 22 04:35 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 May 22 04:35 drafts
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_k.txt
-rw-r----- 1 researcher2 research_team  46 May 22 04:35 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_t.txt
researcher2@fbb39e9e41de:~/projects$
```

File permissions can also be changed for each directory or file. This is done with “chmod” command which stands for “change mode”. The implementation is shown from the image above. Previously, other users were able to write on the “project\_k.txt” file, but it no longer has that permission after we modified it.

```
-rw--w---- 1 researcher2 research_team  46 May 22 04:35 .project_x.txt
```

Let’s say that we only want user and group owner types to be able to read the hidden file “project\_x.txt” (Hidden files are denoted with a full stop “.” before the file name). We run the following command:

```
researcher2@fbb39e9e41de:~/projects$ chmod u=r,g=r .project_x.txt
```

The equal sign(=) means overwriting the permission of an owner type to contain only those specific permissions. Another alternative way is the following command:

```

researcher2@fbb39e9e41de:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@fbb39e9e41de:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 May 22 04:35 .
drwxr-xr-x 3 researcher2 research_team 4096 May 22 05:25 ..
-r--r----- 1 researcher2 research_team  46 May 22 04:35 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 May 22 04:35 drafts
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_k.txt
-rw-r----- 1 researcher2 research_team  46 May 22 04:35 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_t.txt
researcher2@fbb39e9e41de:~/projects$

researcher2@fbb39e9e41de:~$ cd projects
researcher2@fbb39e9e41de:~/projects$ chmod u=r,g=r .project_x.txt
researcher2@fbb39e9e41de:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 May 22 04:35 .
drwxr-xr-x 3 researcher2 research_team 4096 May 22 05:25 ..
-r--r----- 1 researcher2 research_team  46 May 22 04:35 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 May 22 04:35 drafts
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_k.txt
-rw-r----- 1 researcher2 research_team  46 May 22 04:35 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_t.txt
researcher2@fbb39e9e41de:~/projects$

```

Notice that both commands work the same. The minus sign(-) means the permissions typed after the sign will be removed for that particular owner type, the plus sign(+) on the other hand will add a new permission.

## Change directory permissions

To change directory permissions is similar to changing file permissions. For example, we want to remove the group's permissions to execute the "drafts directory". The following commands can be done to both show the initial and final permissions and the command to modify the permissions:

```
researcher2@fbb39e9e41de:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 May 22 04:35 drafts
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_k.txt
-rw-r----- 1 researcher2 research_team  46 May 22 04:35 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_t.txt
researcher2@fbb39e9e41de:~/projects$ chmod g-x drafts
researcher2@fbb39e9e41de:~/projects$ ls -l
total 20
drwx----- 2 researcher2 research_team 4096 May 22 04:35 drafts
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_k.txt
-rw-r----- 1 researcher2 research_team  46 May 22 04:35 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 May 22 04:35 project_t.txt
researcher2@fbb39e9e41de:~/projects$
```

## Summary

The importance of knowing linux commands to change owner permissions and looking for files is very crucial. This allows us to easily prevent further cybersecurity attacks due to malicious actors modifying file contents because of the lack of management towards the details and permissions of files and directories. This is very important for organizations to ensure safety in integrity and accessibility or authorization among the workforce, only giving specific access and permissions for respective employees based on their roles.