# File permissions in Linux

## Project description

The research team at my organization must revise the file permissions for specific files and directories within the projects directory to ensure they align with the appropriate level of authorization. Checking and updating these permissions is essential for maintaining system security. To achieve this, I meticulously inspected and managed permissions for files in the /home/researcher2/projects directory assigned to the researcher2 user, who is part of the research_team group. Any discrepancies were promptly addressed by adjusting permissions as needed.

## Check file and directory details

I started  by determining the current directory, displaying its contents, and navigating into it. Next, I will utilize the `ls -l` command to examine the contents of the project directory, revealing who has access to specific files within it. This command not only displays permissions for files and directories but also provides additional details such as owner name, group, file size, and the time of last modification.

```
researcher2@55ddb716e3ec:~$ pwd
/home/researcher2
researcher2@55ddb716e3ec:~$ ls
projects
researcher2@55ddb716e3ec:~$ cd projects
researcher2@55ddb716e3ec:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Feb 28 16:04 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Feb 28 16:04 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Feb 28 16:04 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_t.txt
researcher2@55ddb716e3ec:~/projects$
```

## Describe the permissions string

In Linux, permissions are depicted using a 10-character string. These permissions encompass:

- Read: For files, this allows reading the file contents; for directories, it permits reading all contents within the directory, including both files and subdirectories.

- Write: For files, this grants the ability to modify the file contents; for directories, it enables the creation of new files within the directory.

- Execute: For files, this authorizes the execution of the file if it's a program; for directories, it allows entering the directory and accessing its files.

For example

```
-rw-rw-rw- 1 researcher2 research_team   46 Feb 28 16:04 project_k.txt
```

The first character in the string indicates the type of the file, where "d" represents a directory and "-" represents a file. In the provided example, since the first character is "-", it indicates that we are working with a file.

The next three characters in the string indicate the permissions for the user, where each character represents read, write, and execute permissions, respectively. In this example, the user has read and write permissions.

The following three characters represent the permissions for the group, where each character denotes read, write, and execute permissions, respectively. In this case, the group also has read and write permissions.

Lastly, the last three characters represent the permissions for others, with each character indicating read, write, and execute permissions, respectively. In this instance, others also have read and write permissions.

## Change file permissions

The organization prohibits others from having write access to any files. I needed to Identify the files requiring permission adjustments using a Linux command. Then, utilize another Linux command to modify these permissions accordingly.

```
researcher2@55ddb716e3ec:~/projects$ chmod o-w project_k.txt
researcher2@55ddb716e3ec:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Feb 28 16:04 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Feb 28 16:04 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_t.txt
researcher2@55ddb716e3ec:~/projects$
```

The file `project_m.txt` is a restricted file and should not be readable or writable by the group or other; only the user should have these permissions on this file.

```
-rw-r----- 1 researcher2 research_team   46 Feb 28 16:04 project_m.txt
```

The group has permission to read the file and I needed to change that so that only the user can do this by using the chmod command

```
researcher2@55ddb716e3ec:~/projects$ chmod g-r project_m.txt
researcher2@55ddb716e3ec:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Feb 28 16:04 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_k.txt
-rw------- 1 researcher2 research_team   46 Feb 28 16:04 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_t.txt
researcher2@55ddb716e3ec:~/projects$
```

## Change file permissions on a hidden file

I had inspected the directory for any hidden files. The hidden files with incorrect permissions were change to fit the company guidence them as necessary. This process enhances security by eliminating unauthorized access.

I used the following command `ls -a`, which reveals hidden files indicated by a period (.) at the beginning. Additionally, combining this command with `ls -l` results in `ls -la`, which displays permissions for files and directories, encompassing hidden files. This amalgamation provides a comprehensive overview, including hidden files, along with their respective permissions.

```
researcher2@55ddb716e3ec:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Feb 28 16:04 .
drwxr-xr-x 3 researcher2 research_team 4096 Feb 28 16:57 ..
-rw--w---- 1 researcher2 research_team   46 Feb 28 16:04 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Feb 28 16:04 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_k.txt
-rw------- 1 researcher2 research_team   46 Feb 28 16:04 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_t.txt
researcher2@55ddb716e3ec:~/projects$
```

The file .project_x.txt, which is a hidden file and has been archived, should not permit any write operations from users. However, both the user and the group should retain read access to this file. I adjusted  the permissions of the file .project_x.txt to allow both the user and the group to read the file, while disallowing any write actions.

```
researcher2@55ddb716e3ec:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@55ddb716e3ec:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Feb 28 16:04 .
drwxr-xr-x 3 researcher2 research_team 4096 Feb 28 16:57 ..
-r--r----- 1 researcher2 research_team   46 Feb 28 16:04 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Feb 28 16:04 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_k.txt
-rw------- 1 researcher2 research_team   46 Feb 28 16:04 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_t.txt
researcher2@55ddb716e3ec:~/projects$
```

## Change directory permissions

The files and directories within the "projects" directory are owned by the "researcher2" user. Specifically, only "researcher2" should have access to the "drafts" directory and its contents. To modify the permissions accordingly, a Linux command can be utilized.

```
researcher2@55ddb716e3ec:~/projects$ chmod g-x drafts
researcher2@55ddb716e3ec:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Feb 28 16:04 .
drwxr-xr-x 3 researcher2 research_team 4096 Feb 28 16:57 ..
-r--r----- 1 researcher2 research_team   46 Feb 28 16:04 .project_x.txt
drwx------ 2 researcher2 research_team 4096 Feb 28 16:04 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_k.txt
-rw------- 1 researcher2 research_team   46 Feb 28 16:04 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 28 16:04 project_t.txt
researcher2@55ddb716e3ec:~/projects$
```

## Summary

I adjusted various permissions to align with the desired authorization levels set by my organization for files and directories within the projects directory. Initially, I utilized `ls -la` to review the permissions of the directory, which guided my subsequent actions. I then employed the `chmod` command multiple times to modify permissions on both files and directories accordingly.