# File Permissions in Linux

## Project description

My goal is to show employers that I am proficient with Linux. Specific to this project, I will be using Linux commands to change permissions in a scenario where I am a security professional working with the research team. This is done on a Debian-based Linux distribution in the bash shell.

I aim to ensure users in the research team have proper authorization to access files and directories by changing the permissions in order to keep the system secure. This follows the principle of least privilege.

## Check file and directory details

```
researcher2@f2be1db09d34:~$ cd projects
```

First I went to the desired directory "projects" using the `cd projects` command. This could also be done by using the absolute path `cd /home/researcher2/projects`

```
researcher2@f2be1db09d34:~/projects$ ls -la
```

Next, I used the command `ls -la` to show all of the permissions of the files and directories in the "projects" directory. The `-la` option shows hidden files as well as normal files and directories.

```
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Oct 11 22:17 .
drwxr-xr-x 3 researcher2 research_team 4096 Oct 11 23:03 ..
-rw--w---- 1 researcher2 research_team   46 Oct 11 22:17 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Oct 11 22:17 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Oct 11 22:17 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Oct 11 22:17 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 11 22:17 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 11 22:17 project_t.txt
```

This outputted 4 files, 1 hidden file, and 1 directory, alongside their permissions at the beginning of each line. The "." represents the current directory, and the ".." represents the parent directory.

## Describe the permissions string

```
-rw-r----- 1 researcher2 research_team   46 Oct 11 22:17 project_m.txt
```

The permissions for `project_m.txt` are `-rw-r-----`, which shows that it is a file (due to the first character being a "-"). It also shows that the user has read and write privileges (`rw-`), the group has read privileges (`r--`), and the other has no privileges (`---`).
The first character is either a "-" or a "d" which shows that it is either a file or a directory.
"r" stands for read, "w" stands for write, and "x" stands for execute.
The 2-4 characters show the permissions for the "user".
The 5-7 characters show the permissions for the "group".
The 8-10 characters show the permissions for the "other".

## Change file permissions

The organization does not want the "other" to have write access to any of these files.

```
-rw-rw-rw- 1 researcher2 research_team   46 Oct 11 23:36 project_k.txt
```

The file "`project_k.txt`" shows that "other" has permission to write in this file, as seen in the output from the `ls -la` command. I will change this using the `chmod` command.

```
researcher2@037440ee5a86:~/projects$ chmod o-w project_k.txt
```

By using the command `chmod o-w project_k.txt`, I removed the write permissions from "other" for the file "`project_k.txt`".
The argument "o-w" is what altered the permissions.
The argument "`project_k.txt`" is the file in which the permissions are being altered.
The "o" stands for other
The "-" removes permission
The "w" is the permission being altered

```
-rw-rw-r-- 1 researcher2 research_team   46 Oct 11 23:36 project_k.txt
```

After checking the permissions again using the `ls -l` command, you can see in the permissions string at the beginning of the line that the write permission was removed from the "other".

## Change file permissions on a hidden file

```
-rw--w---- 1 researcher2 research_team   46 Oct 11 23:36 .project_x.txt
```

The research team has archived **`.project_x.txt`** and should not have write permissions for anyone. The "." in front of the file name shows that it is a hidden file. I will change the permissions so the research team can read but not write in the file.

```
researcher2@037440ee5a86:~/projects$ chmod u-w,g-w,g+r .project_x.txt
```

To change the permissions I used the command `chmod u-w,g-w,g+r .project_x.txt`. This can also be accomplished by using a similar command `chmod u=r,g=r .project_x.txt`.

```
-r--r----- 1 researcher2 research_team   46 Oct 11 23:36 .project_x.txt
```

After checking the permissions again using `ls -la`, it shows that now the user and group have only read permissions, and the other has none.

## Change directory permissions

```
drwx--x--- 2 researcher2 research_team 4096 Oct 11 23:36 drafts
```

The files and directories in the projects directory belong to the **`researcher2`** user. Only **`researcher2`** should be allowed to access the **`drafts`** directory and its contents.

```
researcher2@037440ee5a86:~/projects$ chmod g-x drafts
```

Using the command `chmod g-x drafts`, I removed the execute permission from the "group".

```
drwx------ 2 researcher2 research_team 4096 Oct 11 23:36 drafts
```

## Summary

In this project, I used the Linux command `chmod` in conjunction with `cd` and `ls` to change the permissions of files and directories for the research team. This was done with the intent to implement the principle of lease privilege and improve the security of the system. If you are an employer reading this, thank you for giving me a chance and looking at my work!