# File permissions in Linux

## **Project description**

## The task is to examine existing permissions on the file system within the **project** directory. There is a need to determine if the permissions match the authorization that should be given. If they do not match, maybe there is a need to modify the permissions to authorize the appropriate users and remove any unauthorized access.

## Check file and directory details.

To check the permission, type ls -l to check the files and its respective permission. To show hidden files, type ls -a. To do both, just type **ls -la**

## 

In the /home/researcher2/projects directory, there are five files with the following

names and permissions:

● **project\_k.txt**

○ User = read, write,

○ Group = read, write

○ Other = read, write

● **project\_m.txt**

○ User = read, write

○ Group = read

○ Other = none

● **project\_r.txt**

○ User= read, write

○ Group = read, write

○ Other = read

● **project\_t.txt**

○ User = read, write

○ Group = read, write

○ Other = read

● .**project\_x.txt**

○ User = read, write

○ Group = write

○ Other = none

## **Describe the permissions string**

The 10-character string can be deconstructed to determine who is authorized to access the file and their specific permissions. The characters and what they represent are as follows:

● 1st character: This character is either a d or hyphen **(-)** and indicates the file type. If it’s a **d,** it’s a directory. If it’s a hyphen **(-)**, it’s a regular file.

● 2nd-4th characters: These characters indicate the read **(r),** write **(w),** and execute **(x)** permissions for the user. When one of these characters is a hyphen **(-)** instead, it indicates that this permission is not granted to the user.

● 5th-7th characters: These characters indicate the read **(r),** write **(w)**, and execute **(x)** permissions for the group. When one of these characters is a hyphen **(-)** instead, it indicates that this permission is not granted for the group.

● 8th-10th characters: These characters indicate the read **(r),** write **(w),** and execute **(x)**permissions for other. This owner type consists of all other users on the system apart from the user and the group. When one of these characters is a hyphen **(-)** instead, that indicates that this permission is not granted for other.

Example: **project\_k.txt**

10 strings as follows: **-rw-rw-rw-**

There is also one subdirectory inside the **projects** directory named drafts. The

permissions on drafts are:

● User = read, write, execute

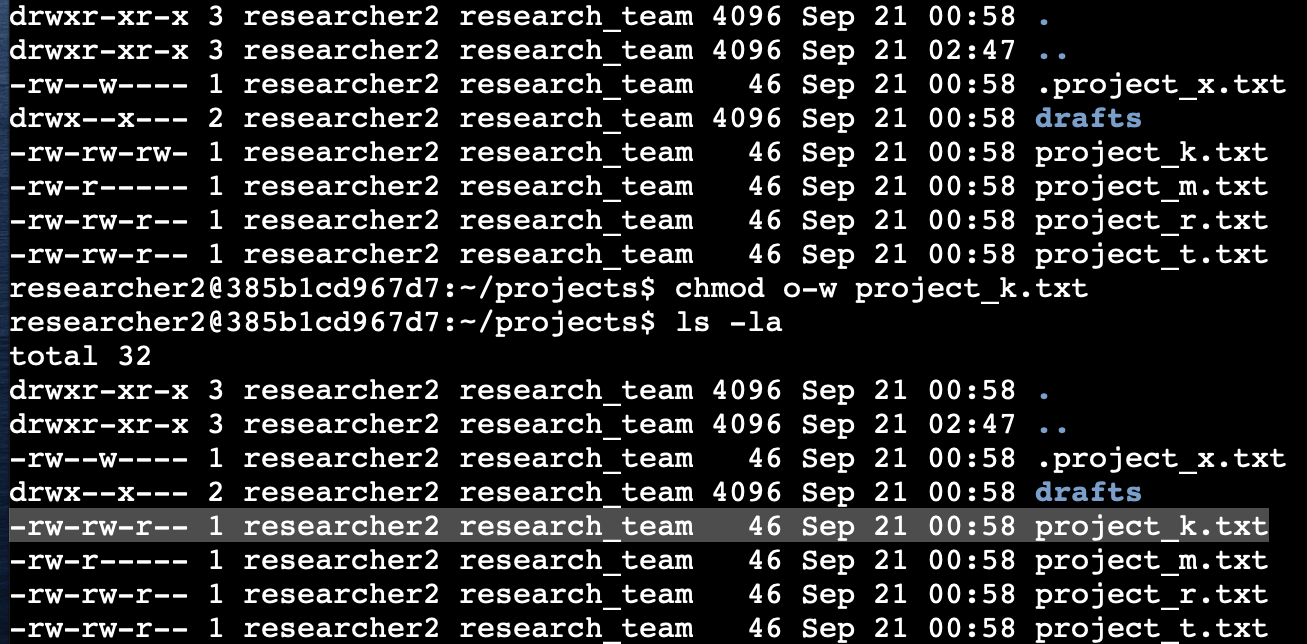
● Group = execute

● Other = none

10-character string is to show what are the user, group, and others current permission is. As for the example mentioned, **project\_k.txt** shows **-rw-rw-rw-** which means that the user, group, and other has the same privileges which is read and write only without execute permissions.

## **Change file permissions**

## The organization does not allow other to have write access to any files.

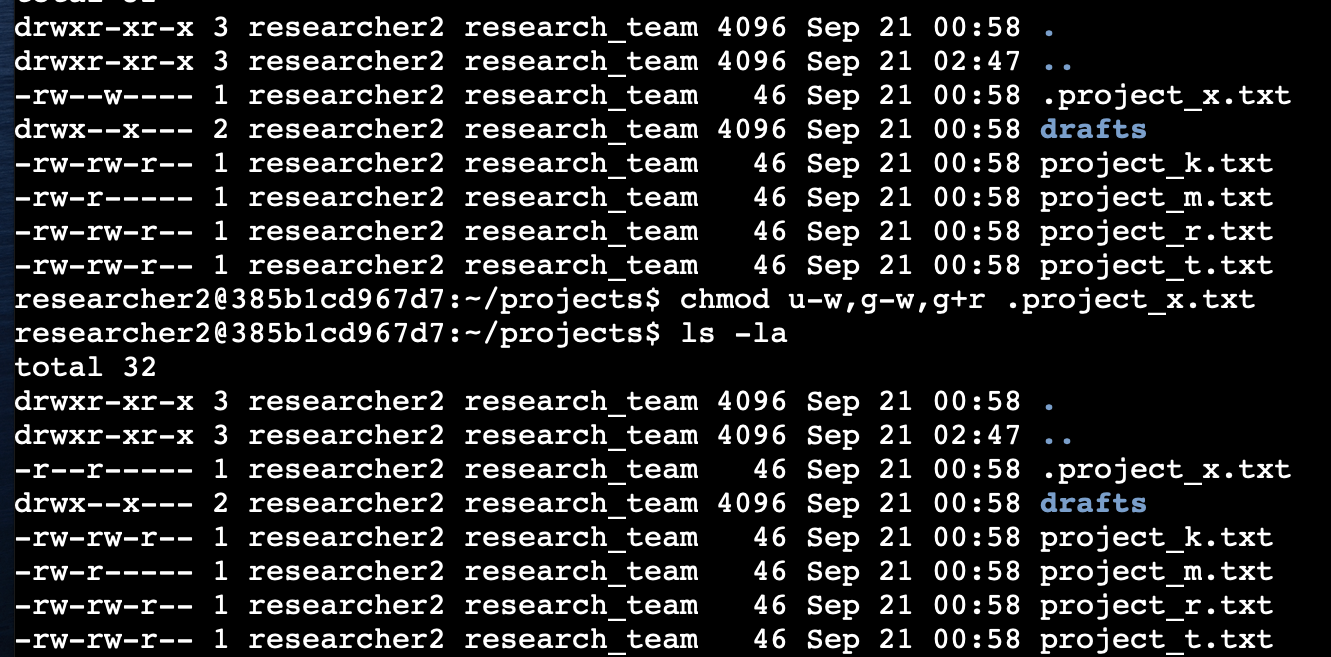


As per screenshot above, project\_x.txt still has a write permission. To remove or change permission, type the command **chmod o-w project\_k.txt** (**chmod** is to initiate change permissions then first argument of “o-w” represents “other minus/remove “w” or “write” permission, then finally the second argument is which file you want to make changes which is **project\_x.txt**).

## **Change file permissions on a hidden file**

This file should not have write permissions for anyone, but the user and group should be able to read the file. As per screenshot below, the top third row shows the file **.project\_x.txt user** and group still have the write permission and read permission is not showing in the group. For this to be sorted, I type the following command: **chmod u-w,g-w,g+r .project\_x.txt**

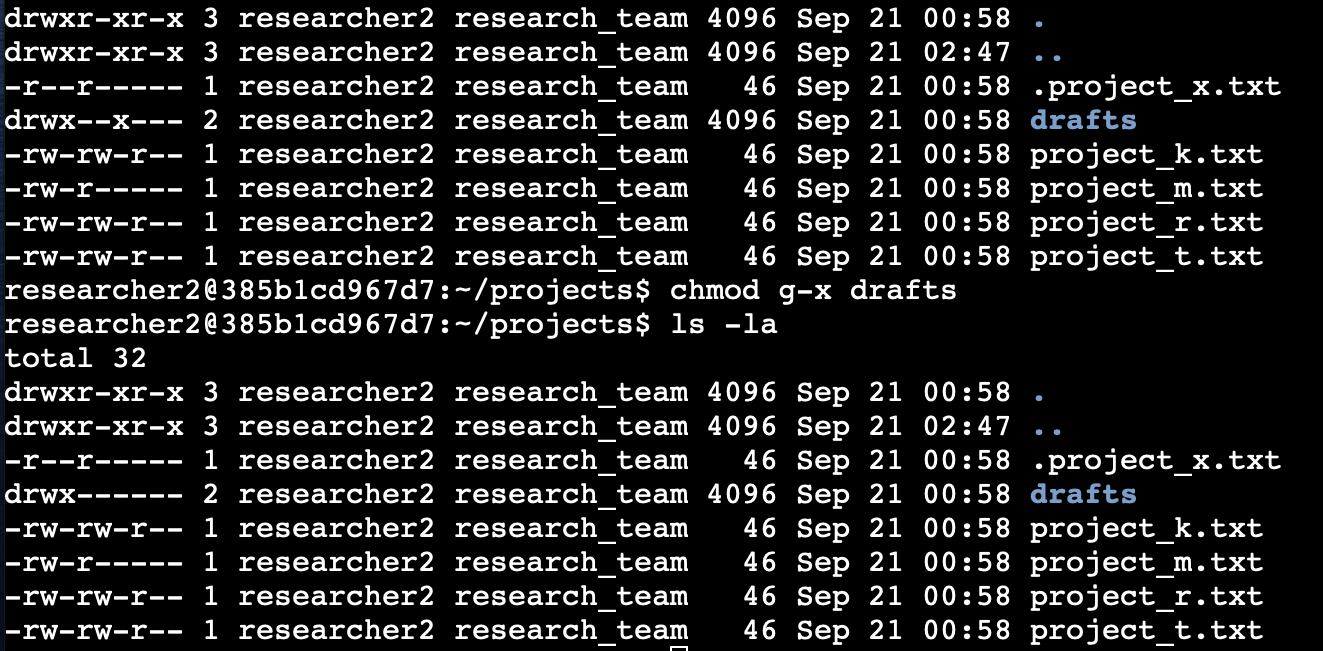
To check if the changes of the file has gone through successfully, I typed in the command **ls -la** and it shows that both user and group can read the file and no write permission has been granted to anyone. Please see screenshot below:



## **Change directory permissions**

## 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.

As per screenshot below, group has execute permissions and should be removed. I typed in the command **chmod g-x drafts .** To double check if it has taken effect, I typed in **ls -la** to view all files including hidden files. It is now showing that **researcher2** only has access to any contents of the file.



## **Summary**

To sum it up, the goal is to make sure that everyone has the right permission within the **project** directory and amend if necessary. **cmod** command has been used to change the necessary file that includes adding or removing privileges. Checking file permission which is the **ls -la** has been properly utilised ensuring the accuracy and the desired changes has been applied. The objective of making sure that the system is secured has been accomplished.