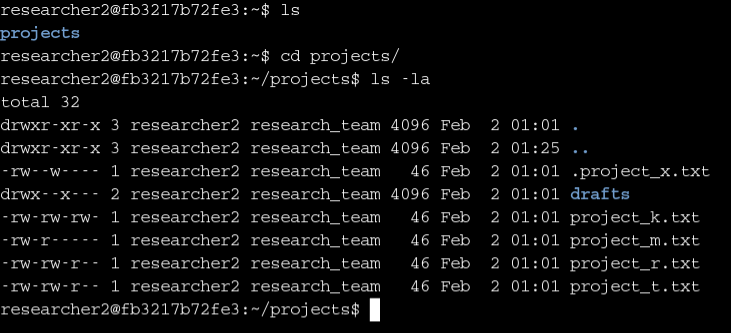
# File permissions in Linux

## Project description

A security analyst is working with the security research team. One of the tasks is to ensure that files and directories have proper permissions set so that people who need access can access them and any unauthorized access can be prevented.

## Check file and directory details

We can use, ls -l to find the files and directory permissions. To also show the hidden files we can add the flag ‘a’. So the final looks like ls -la. We can see this command in action in the figure below.



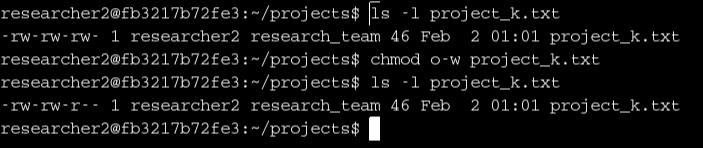
## Describe the permissions string

From the figure above, let’s look at the permissions of the file project\_k.txt. We find, -rw-rw-rw-. There are 10 characters here. The first field tells us if we are dealing with a file or a directory. If it is a directory it is denoted by the string ‘d’. For files, we see a hyphen. Following this character, we have 9 permissions which can be grouped into 3. The first 3 represent the permissions for the user (owner) of the file/directory, the second is for the group and the last 3 are for others. In each 3 we have ‘read’, ‘write’, and ‘execute’ permissions which are represented by strings r, w, and x respectively. For a directory, the write permissions mean that one can create, modify, or delete files. An execute permission for a directory would mean that one can cd into that directory.

With these ideas in mind from its permissions, we can infer for the project\_k.txt that it is a file given the first hyphen. For the user, we have rw-, which means that the user can read and write but cannot execute the file. The group and others have the same permission strings, rw- which means both of them can read and write and cannot execute.

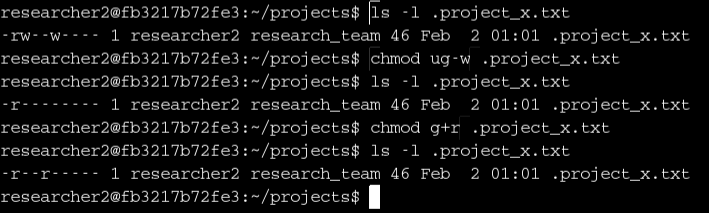
## Change file permissions

Now we are going to look at the file permissions for project\_k.txt to make sure the permissions are properly set. The permissions read -rw-rw-rw-. Here we find others can write into the file which cannot be allowed. To modify the permission we issue chmod o-w project\_k.txt. Here o represents others and -w shows that were ‘minusing’ or taking out write permission from others. The process can be seen in the figure below.



## Change file permissions on a hidden file

The hidden file .project\_x.txt is an archived file. When we look at the file permission we see that both user and group can still write into it, which cannot be allowed. We only want read permissions for both of them. First, we take out the write permission from both the user and the group using the command chmod ug-w .project\_x.txt. Then we add the read permission for the group using chmod g+r .project\_x.txt. Now the permissions are properly set as -r—r—---.



## Change directory permissions

There is a directory called drafts in the projects directory. It belongs only to the researcher2 and neither group or others cannot have any sort of permissions on it. At the moment others have execute permission on the folder. We modify the permission by taking out the execute permission for others by chmod g-x drafts. The final permissions string looks like drwx—--- which is what we want.

## 

## Summary

One of the important responsibilities of a security analyst is to make sure that directory and file permissions are properly set so as to prevent any unwanted or unauthorized access. In this project file/directory permissions are studied using linux command line tools. Any improper permission on regular or hidden files are checked. It is made sure the proper permissions are set. Any unwanted or insecure permissions are modified to ensure security.