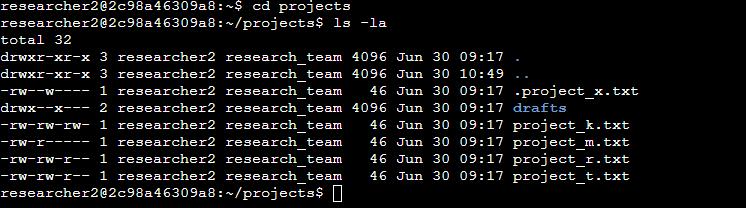
# File permissions in Linux

## Project description

As a Security Analyst at a Technology organization, I’m tasked with updating file permissions for some files and directories within the projects directory. The permissions do not currently reflect the level of authorization that should be given. Checking and updating these permissions will help keep the organization’s system secure. I performed the following tasks to ensure these needs were met:

## Check file and directory details

The following code demonstrates how I used Linux commands to determine the existing permissions set for a specific directory in the file system.



The first line of the screenshot displays the command I entered, and the other lines display the output. The code lists all contents of the projects directory. I used the ls command with the -la option to display a detailed listing of the file contents that also returned hidden files. The output of my command indicates that there is one directory named drafts, one hidden file named.project\_x.txt, and five other project files. The 10-character string in the first column represents the permissions set on each file or directory.

## 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.

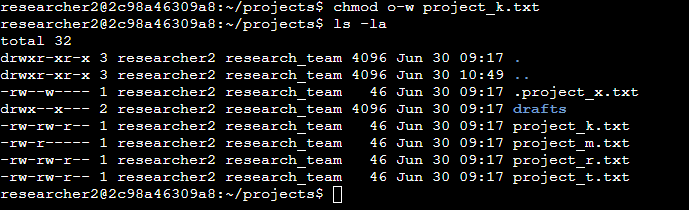
For example, the file permissions for project\_t.txt are -rw-rw-r--. Since the first character is a hyphen (-), this indicates that project\_t.txt is a file, not a directory. The second, fifth, and eighth characters are all r, which indicates that user, group, and other all have read permissions. The third and sixth characters are w, which indicates that only the user and group have write permissions. No one has execute permissions for project\_t.txt.

## Change file permissions

The organization determined that other shouldn't have write access to any of their files. To comply with this, I referred to the file permissions that I previously returned. I determined project\_k.txt must have the write access removed for other.

Our organization determined that other shouldn’t have any write access to any of their files. Using the results I returned earlier, I determined that project\_k.txt must have the write access removed for other.

The following below depicts how I used Linux commands to do this:

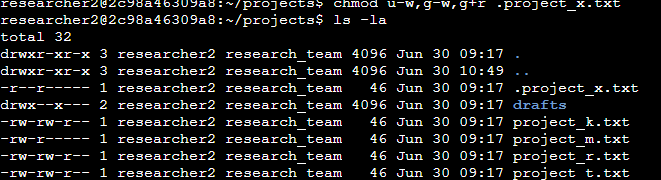


I used the chmod command to change the permissions on files and directories. The first argument indicates what permissions should be changed, and the second argument specifies the file or directory. In this example, I removed write permissions from other for the project\_k.txt file. After this, I used ls -la to review the updates I made.

## Change file permissions on a hidden file

The research team at my organization recently archived project\_x.txt. They do not want anyone to have write access to this project, but the user and group should have read access.

The below depicts how I used Linux commands to change the permissions:

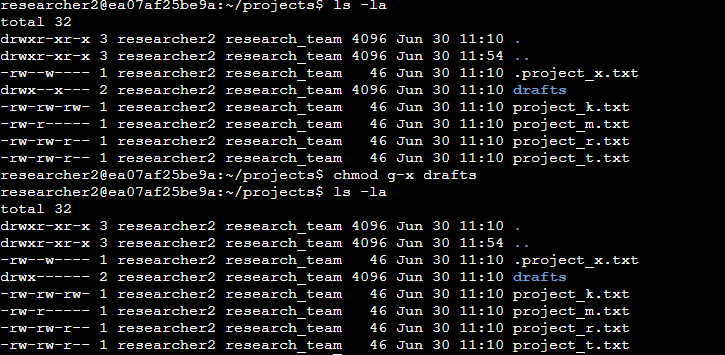


Our file .project\_x.txt is a hidden file and this is because it starts with a period (.). I used the command u-w to remove permissions from the user, then g-w, to remove permissions from the group and g+r to add permissions to the group.

## 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. I

To achieve the above, I used the below Linux commands to change the permissions:



From my first Linux command, I had determined that the group (research\_team) had execute permissions. I then used the second Linux command using the chmod g-x drafts.

## Summary

I used multiple Linux commands to change permissions and match the level of authorization my organization wanted for files and directories in the projects directory. The first step in this was using ls -la to check the permissions for the directory. This informed my decisions in the following steps. I then used the chmod command multiple times to change the permissions on files and directories.