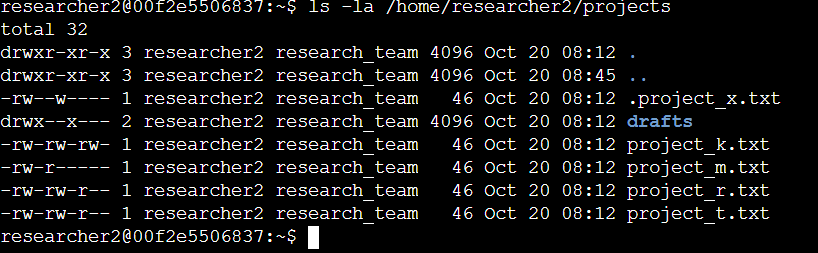
# File permissions in Linux

## Project description

I work as a security professional for a major organization, where I make sure the users of the research team are acquired with accurate permissions to keep the system safe. My duty is to examine existing permissions on the file system and decide if the permissions align with the authorization that is given. It is important for me to change the permissions to ensure the users are accurately authorized and remove any unauthorized access.

## Check file and directory details



The command I used to display this information is ls -la, where I am able to display the permissions of the files and directories as well as the hidden files. Based on the output, we can see there is one directory named drafts, one hidden file called .project\_x.txt and four other project files. The 10-characters string on the first column tells us the permission set on each file or directory.

## Describe the permissions string

The 10-character string explains who is authorized the file and the exact permissions they are granted access to, which consists of read, write and execute.

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\_k.txt are -rw-rw-rw-. Since the first character is a hyphen, this indicates that project\_k.txt is a file, not a directory. The 2nd, 5th, and 8th character represents the read command, where the user, group and other have read permissions meaning they can read the contents of the file. The 3rd, 6th and 9th characters demonstrate the write command, where the user, group and other have write permissions and can make modifications to the file. However, the 4th, 7th and 10th command is a – meaning nobody has execute permissions for project\_k.txt.

## Change file permissions

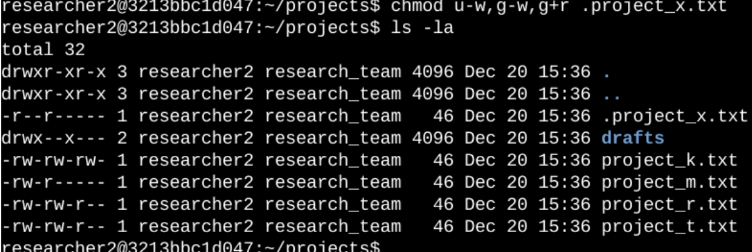
The project\_k.txt file will need to have it’s write permission modified as the others in the 4 remaining files are not granted permission to write.

## 

I used the chmod command, which enables you to make permissions to the file and directories. In this scenario, I used chmod to remove the write permissions for other in project\_k.txt file. After these modifications, I used the ls -la command to display the new updated project\_k.txt file permissions.

## Change file permissions on a hidden file

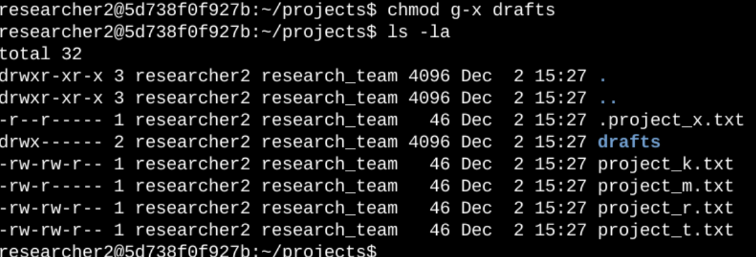
The organization decides to remove the write permissions for user and group but add the read permission for user and group using the chmod command for the hidden file .project\_x.txt.



The .project\_x.txt file is hidden as it starts with a (.). From this image, we used chmod to make permissions for the hidden file, we used u-w, which means we removed write permissions for user, g-w which removes the write permissions for group and g+r which gives read permissions for group.

## Change directory permissions

The organization only wants researcher2 user to have execute permissions for the draft directory, where we use chmod to remove execute permissions for group.



We use chmod command to remove the execute command from group, where we ensure only the user, researcher2, is permitted to execute. We used ls -la to view the updates.

## Summary

I modified many permissions for the organization to ensure the files and directories received the proper authorization throughout the project’s directory. The first command to help us establish the changes is ls -la to view the current permissions for the directory before implementing the modifications. Next, we used the chmod command to make various changes to the permissions of files and organizations to guarantee the most up-to-date authorization.