# **File permissions in Linux**

## &2 Project description

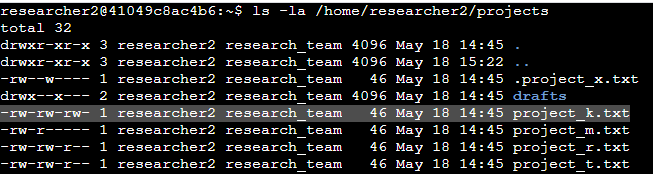
As a security professional, my task is to examine and modify file permissions on the file system to ensure that users on the research team have appropriate access and unauthorized access is removed. I will analyze the existing permissions and make necessary modifications to match the required authorization.

## Check file and directory details

To view the permissions of files and directories in Linux, I can use the ls -l command.

*ls -l /home/researcher2/projects*

This command will list the files and subdirectories in the */home/researcher2/projects* directory along with their permissions.



## Describe the permissions string

The permissions string consists of ten characters and provides information about the access levels for the owner, group, and other users. The characters in the permissions string represent read (r), write (w), and execute (x) permissions.

For example, the permissions string ***-rw-rw-rw-*** of the file *project\_k.txt* means that the owner, group, and other users have read and write permissions, while the execute permission is not granted.

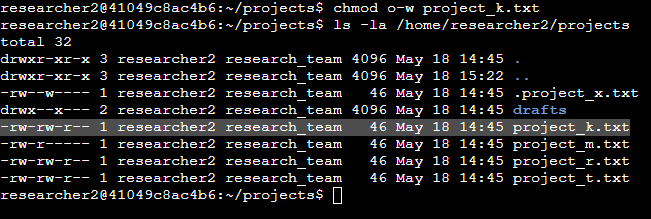
## Change file permissions

To modify file permissions, I can use the chmod command followed by the permissions string and the file name.

*chmod <permissions> <file>*

For example, to remove write permission to other on project\_r.txt, I can use the following command:

*chmod o-w project\_k.txt*

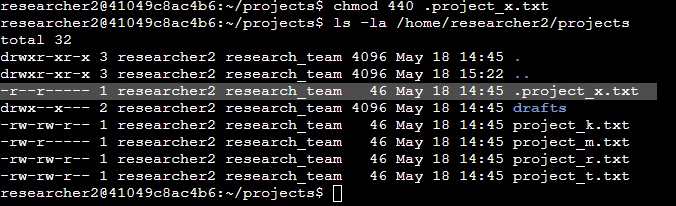
**

## Change file permissions on a hidden file

To modify the permissions of .project\_x.txt so that nobody has write permissions and the user and group can read the file, I can use the chmod command. Here's the command I use:

*chmod 440 .project\_x.txt*

This command sets the permissions to **440** for .project\_x.txt. In the 440 permissions notation, the first digit 4 represents read permission for the owner, the second digit **4** represents read permission for the group, and the third digit **0** represents no permissions for others. This ensures that the user and group can read the file, while others have no permissions at all.

**

## Change directory permissions

To modify directory permissions, I can also use the chmod command, but include the -R option to apply the changes recursively to all files and subdirectories within the directory.

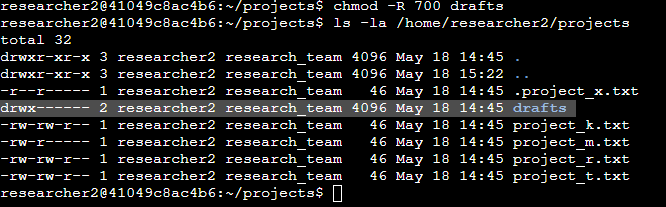
*chmod -R <permissions> <directory>*

This command sets the permissions of the drafts directory and its contents to **700**. Here's a breakdown of the permission string:

**7** indicates read, write, and execute permissions for the owner (researcher2).

**0** indicates no permissions for the group and other users.

By setting the permissions to 700, only the owner (researcher2) will have full access to the drafts directory and its contents, while the group and other users will have no access.

**

## Summary

As a security professional, I examined the file permissions in the /*home/researcher2/projects* directory and made necessary modifications to ensure appropriate access and remove unauthorized access. I have used the *ls -l* command to check the permissions, *chmod* command to change file and directory permissions, and applied modifications based on the required authorization. By doing so, I have helped maintain system security for the research team.