# File permissions in Linux

## Project description

## Ensure users on the Research team are authorized with the appropriate permissions. This helps keep the system secure.

Examine existing permissions on the file system. Determine if the permissions match the authorization that should be given. If they do not match, modify the permissions to authorize the appropriate users and remove any unauthorized access.

## Check file and directory details

A screen shot of a computer

Description automatically generatedThe first line of the screenshot shows the command I entered, while the subsequent lines show its output. The command lists all the contents of the “projects” directory. I used the ls command with the -la option to display a detailed view of the files, including hidden ones. The output reveals one directory called "drafts," a hidden file named “.project\_x.txt”. and five additional project files. The 10-character string in the first column represents the permission settings for each file or directory.

Describe the permissions string

The 10-character string can be broken down to determine who has access to the file and their specific permissions. Here is what each character represents:

* **1st character:** This character is either a "d" or a hyphen (-), indicating the file type. "d" means it is a directory, while a hyphen (-) means it is a regular file.

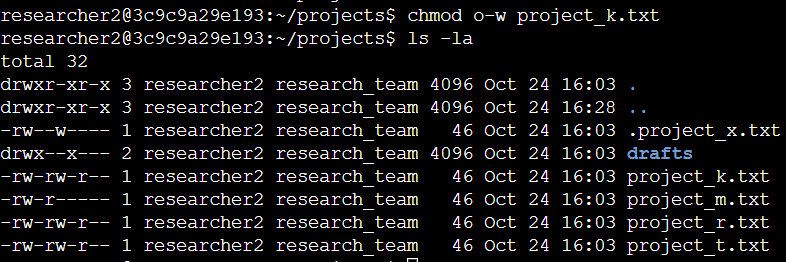
## **2nd-4th characters:** These represent the read (r), write (w), and execute (x) permissions for the user (owner). If any of these is replaced by a hyphen (-), that permission is not granted to the user.

## **5th-7th characters:** These represent the read (r), write (w), and execute (x) permissions for the group. A hyphen (-) in place of any of these indicates that permission is not granted to the group.

## **8th-10th characters:** These represent the read (r), write (w), and execute (x) permissions for others (all users apart from the user and group). A hyphen (-) here indicates that permission is not granted to others.

## Change file permissions

The organization decided that others should not have write access to any of their files. To ensure compliance, I reviewed the previously displayed file permissions and identified that write access for "other" needed to be removed from "project\_k.txt."

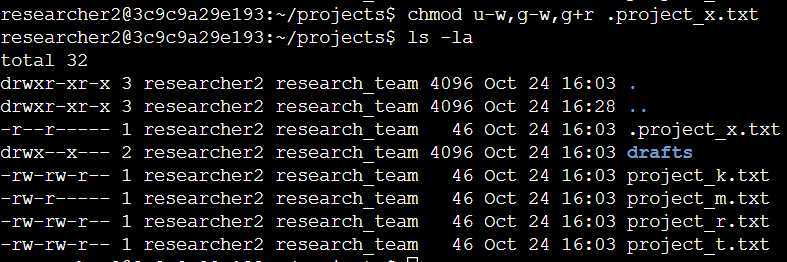
The following code shows how I used Linux commands to accomplish this:

The first two lines of the screenshot show the commands I entered, while the remaining lines display the output of the second command. The chmod command modifies the permissions for files and directories. The first argument specifies which permissions to change, and the second argument identifies the file or directory. In this case, I removed write permissions for others on the "project\_k.txt" file. Afterward, I used ls -la to verify the changes I made.

Change file permissions on a hidden file

## The research team recently archived "project\_x.txt" and requested that no one have write access to the file, while both the user and group should retain read access.

## The following code shows how I used Linux commands to modify the file permissions:

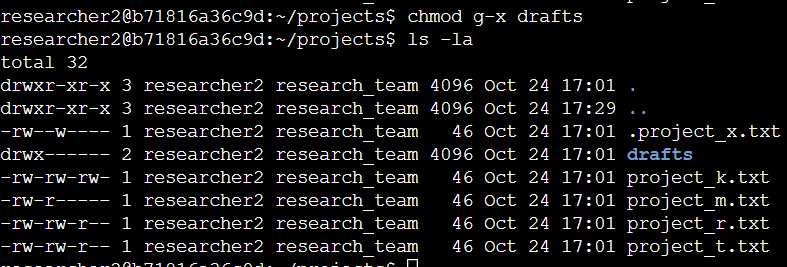


The first two lines of the screenshot show the commands I entered, while the remaining lines display the output of the second command. I can tell that “. project\_x.txt" is a hidden file because its name begins with a period (.). In this example, I removed write permissions from both the user and the group, and added read permissions for the group. I removed the user's write permissions with u-w, removed the group's write permissions with g-w, and added read permissions for the group using g+r.

## Change directory permissions

## My organization requires that only the "researcher2" user has access to the "drafts" directory and its contents. This means no one else should have execute permissions.

## The following code shows how I used Linux commands to modify the permissions:



The first two lines of the screenshot show the commands I entered, while the remaining lines display the output of the second command. I had previously identified that the group had execute permissions, so I used the chmod command to remove them. Since the "researcher2" user already had execute permissions, no changes were needed for that user.

## Summary

I modified several permissions to align with the level of access my organization required for the files and directories in the "projects" directory. First, I used ls -la to review the current permissions, which guided my next steps. I then used the chmod command multiple times to adjust the permissions for various files and directories.