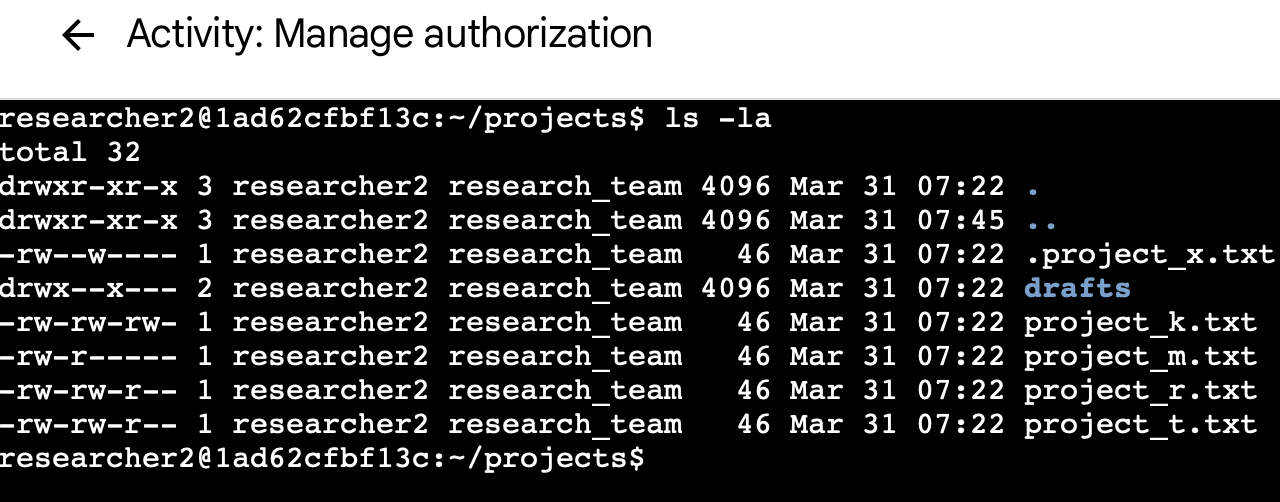
# File permissions in Linux

## Project description

The team working on research at my company needs to adjust the access rights, for files and folders in the projects folder. The current permissions don't match the required level of authorization. Verifying and modifying these permissions is essential, for maintaining the security of their system. I carried out the following actions to address this issue:

## Check file and directory details

I used linux commands to determine existing permissions set for specific directories in the file system



The first line displays the command i entered and the other lines displays 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.

**Current file permissions**

This document displays the file structureof the /home/researcher2/projects directory and the permissions of the files and subdirectory it contains.

In the /home/researcher2/projects directory, there are five files with the following names and permissions:

1. **project\_x.txt**

* User = read,write
* Group = write
* Other = none

1. **project\_k.txt**

* User = read,write
* Group = read,write
* Other = read,write

1. **project\_m.txt**

* User = read,write
* Group = read
* Other = none

1. **project\_r.txt**

* User = read,write
* Group = read,write
* Other = read

1. **project\_t.txt**

* User = read,write
* Group = read,write
* Other = read

## Describe the permissions string

The 10-character string(drwxrwxwrx) 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 (d) represents a directory or hyphen (-) which represents a regular file.
* 2nd-4th character: represents owners type which is user. It includes (r)read, (w)write, (x)execute when one of the characters is a hyphen (-) it means permission is not granted to the user.
* 5th-7th character: represents the owners type which is group. Includes (r)read, (w)write, (x)execute when one of the characters is a hyphen (-) it means permission is not granted to the group.
* 8th-10th characters: represents the owners type which is other. Includes (r)read, (w)write, (x)execute when one of the characters is a hyphen (-) it means permission is not granted to the other. This owner type consists of all other users on the system apart from the user and the group.

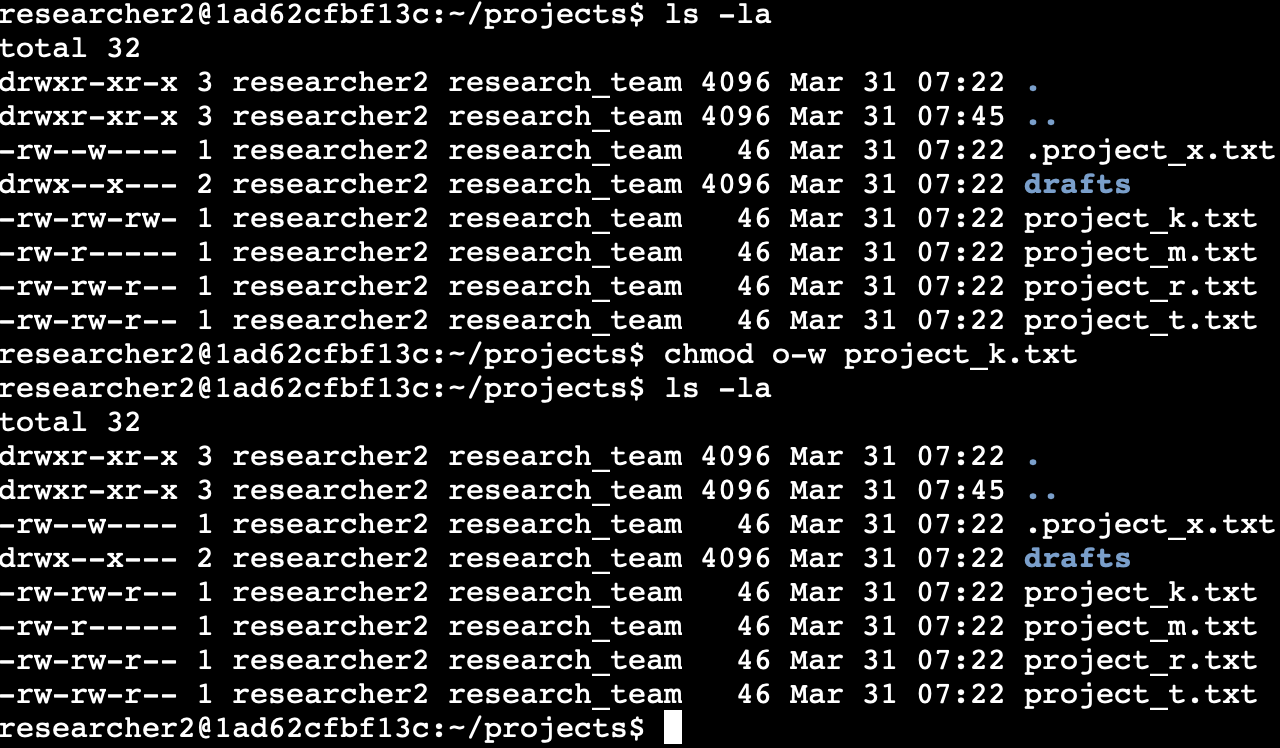
## Change file permissions

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.

The following code demonstrates how I used Linux commands to do this:



The first two lines of the screenshot display the commands I entered, and the other lines

display the output of the second command. The chmod command changes 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

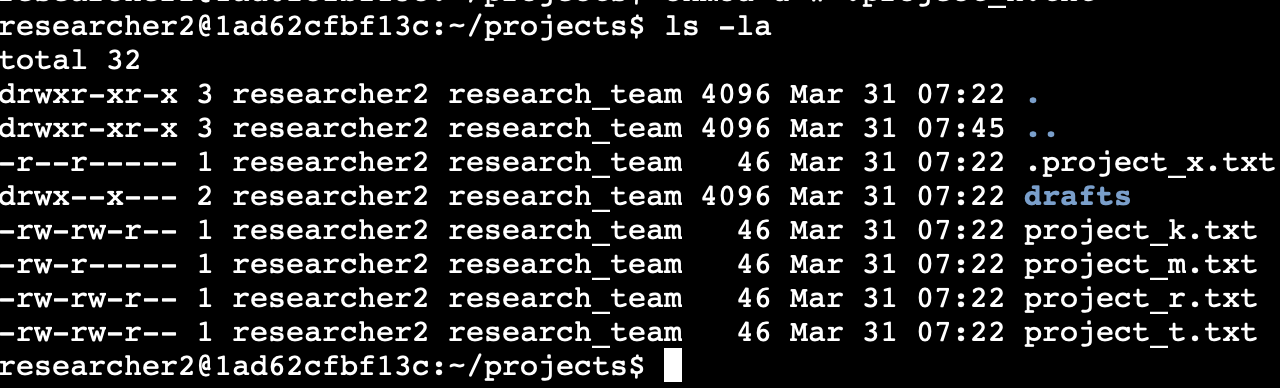
The research team at my organization recently archived project\_x.txt. They do not want

anyone to have write access to this project, but the user and group should have read access.

.project\_x.txt

The following code demonstrates how I used Linux commands to change the permissions:

Chmod u-w, g-w, g+r .project\_x.txt



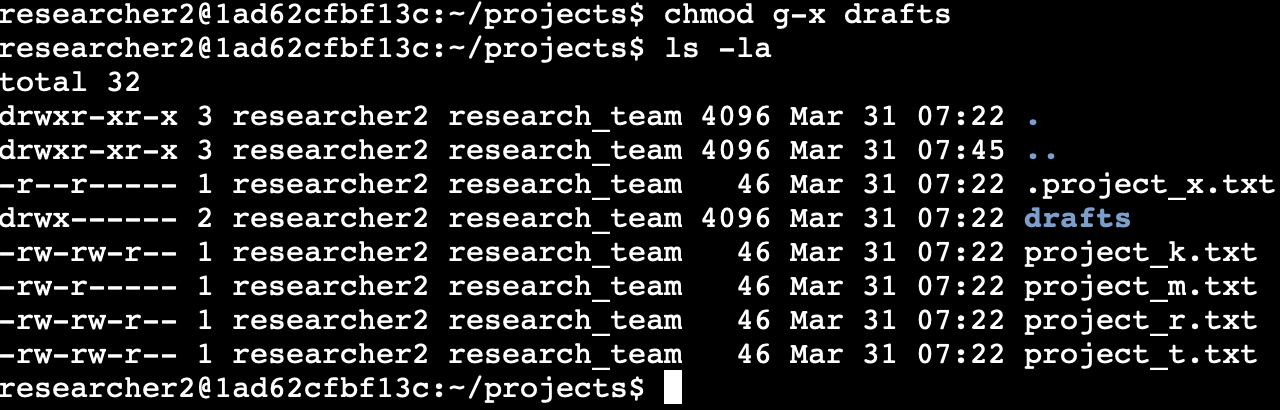
## Change directory permissions

My organization only wants the researcher2 user to have access to the drafts directory

and its contents. This means that no one other than researcher2 should have execute

permissions.

The following code demonstrates how I used Linux commands to change the permissions:



The output here displays the permission listing for several files and directories. Line 1 indicates

the current directory (projects), and line 2 indicates the parent directory (home). Line 3

indicates a regular file titled .project\_x.txt. Line 4 is the directory (drafts) with restricted

permissions. Here you can see that only researcher2 has execute permissions. It was

previously determined that the group had execute permissions, so I used the chmod command

to remove them. The researcher2 user already had execute permissions, so they did not

need to be added.

## Summary

I changed multiple permissions to 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.