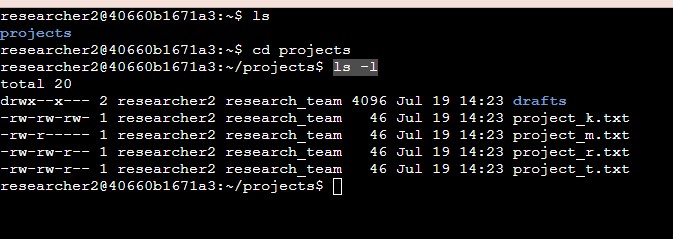
# File permissions in Linux

## Project description

In this project I am responsible for ensuring that users on the team are authorized with the proper permissions in order to help keep the system secure.

My task is to examine the existing permissions on the file system, and update them accordingly to ensure that the only permissions that are granted are the ones that are needed.

## Check file and directory details



Using the ls -l command in the projects directory allowed me to view the permissions for all of the files and directories inside of the projects directory.

## Describe the permissions string

Using project\_k.txt in the output as an example, the 10-character string displays the permissions for the file as follows:

-rw-rw-rw-

Each character in the string represents (from left to right)

- this is a file.

r the user has read permissions.

w the user has write permissions

- the user does not have execute permissions

r the group has read permissions.

w the group has write permissions.

- the group does not have execute permissions.

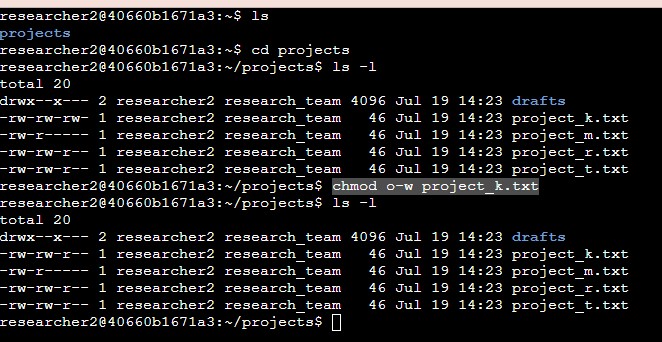
r the others have read permissions.

w the others have write permissions.

- the others do not have execute permissions.

Note: The other group in this example is anyone else who has access to this directory.

## Change file permissions



The organization does not allow other to have write access to any files.

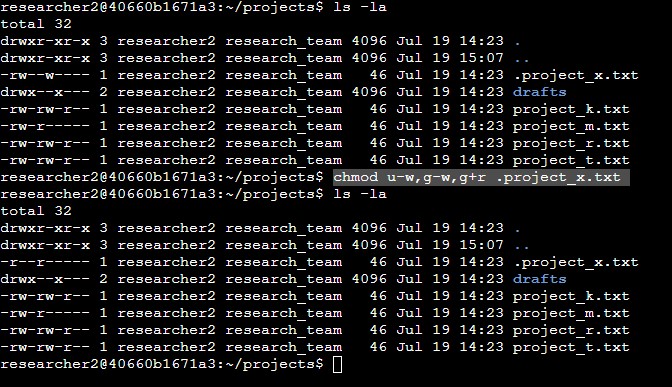
After checking the permissions in the previous step, it is evident that the other group has write permission for the file project\_k.txt.

By using the command chmod o-w project\_k.txt I am:

1. Giving the chmod command to change a permission.
2. o-w because I want to change the permission on the other group (o), I want to remove a permission (-), and the permission I want to take away is the write permission (w).
3. project\_k.txt because that is the only file that the other group has write permission for.

I used the ls -l command to verify that the permissions were updated.

## Change file permissions on a hidden file



The research team has archived .project\_x.txt, which is why it is a hidden file. No one should have write permissions for this file, but the user and group should have read permissions for the file.

When checking the permissions for the hidden file by using the ls -la command, I can see that user has read and write permissions for the file, and the group only has write permission for the file.

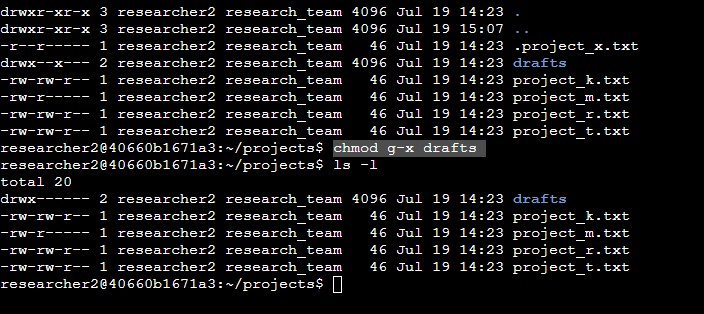
To change this so the user and group only have read permissions, I use the command:

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

I use this command because I want to remove write permissions for the user (u-w), remove the write permission for the group (g-w), and add read permission for group (g+w) to the hidden file .project\_x.txt.

I use the command ls -la again to verify that the changes were successful.

## Change directory permissions



Only researcher2 should have access to the drafts directory. I can see from the previous ls -l commands that the group has execute permissions on the drafts directory.

By entering the command chmod g-x drafts I am removing the execute permissions for the group.

I enter the ls -l command again to ensure that the changes were successful.

## Summary

This project demonstrates my ability to add and/or remove permissions in the Linux bash shell. Ensuring that only authorized users have the permissions they need to do their jobs helps to keep the organization safe and data secure.