# File permissions in Linux

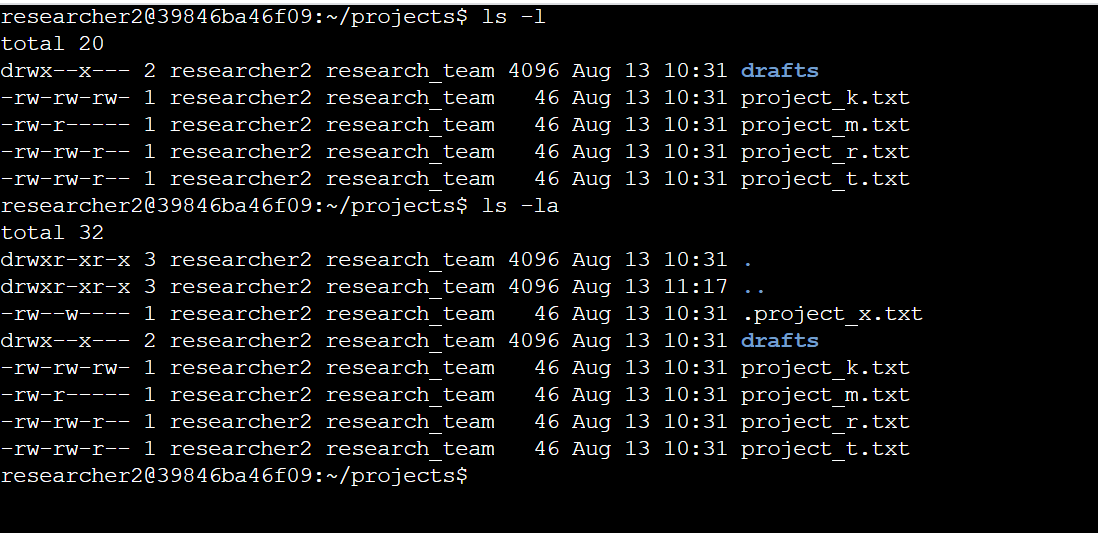
## Project description

Using Linux commands, the objective of this is to look at current permissions on the file system. Permissions should be checked to see if the proper authorization is given and to modify the permissions if not. Any additional permissions should be modified to remove unauthorized access. This will ensure that the system remains secure and follows the principle of “least privilege.” The organization would like:

* To remove write permissions for other users for “project\_k.txt”.
* Read access for the user and group but no write access for anyone for the hidden file “.project\_x.txt”.
* Ensure that only the user has access to the “drafts” directory (thus no one else should have execute permissions).

## Check file and directory details

In order to view the existing permission, we can use specific Linux commands. This is done once you are in the specific directory in the file system. As seen below the first command allows us to see the file contents while the second command allows us to see the file contents including the hidden files. We see that there is a directory, a hidden file, and five other files. On the left hand side, we see the permission settings for these files.



## Describe the permissions string

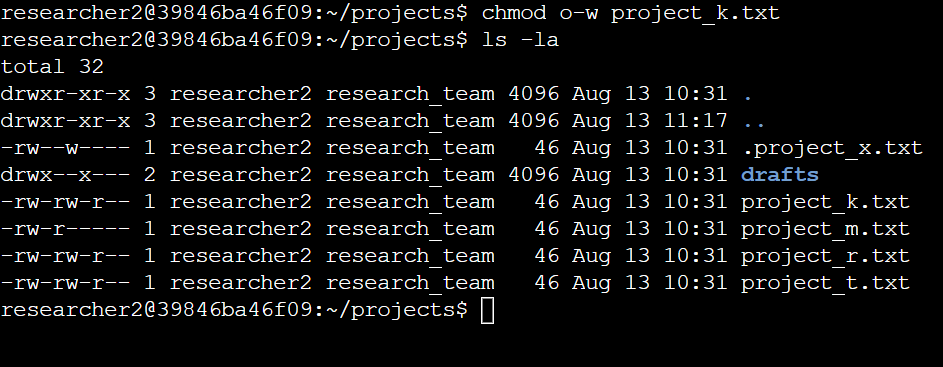
The permissions string as viewed on the left hand side shows us the existing permission settings. This 10-character string shows us authorizations for the user, group type and other.

The first character shows us the file type where “d” indicates a directory. If a hyphen appears, it is a regular file. For the following characters, a “r” indicates read permissions, “w” means write permissions, “x” indicates execute permissions, and “-” indicates that the permission is not granted to the user.

* The 2nd-4th characters indicate the file permissions for the users.
* 5th-7th characters indicate file permissions for the group.
* 8th-10th characters indicate file permissions for all other users.

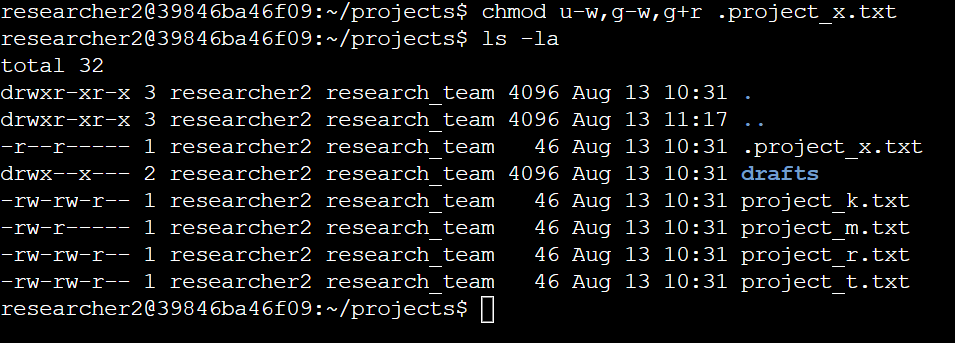
## Change file permissions

In order to change the file permissions to ensure other users do not have the write permission, we can use the “chmod” command. From the previous command, we can see that the file “project\_k.txt” includes write permission for the other users. Therefore, we know that that is the file permission we have to change. Using chmod, we can indicate the permission we want changed as the first argument and the file we want to update the permission for as the second argument. Using the “ls -la” command, we see that the command was able to update the permissions properly for that file.



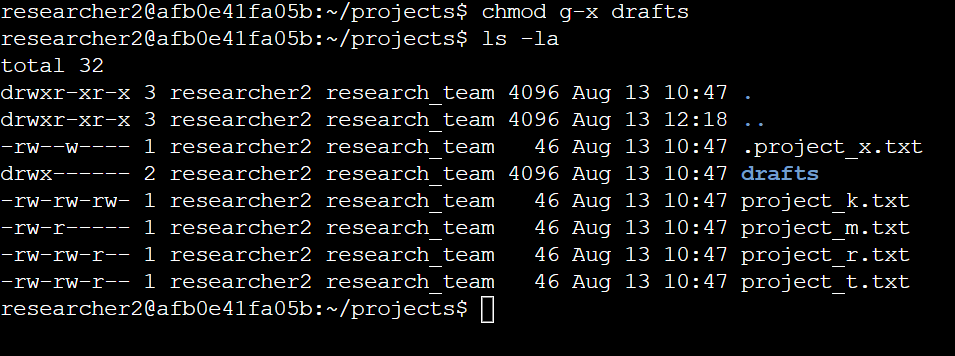
## Change file permissions on a hidden file

In order to change file permissions on the hidden file, “.project\_x.txt”, we need to update the permissions to allow user and group to have read access but no one to have write access. We again can use “chmod” to take away write permissions for user and group and then add read permission for group as see below. We execute the “ls -la” command once again to see the updated proper permissions.



## Change directory permissions

We want only the user to have access to the “drafts” directory and its contents which means that no one else should have execute permissions. As seen below we can do so by using the “chmod” command and taking away the execute permission for the group. We confirm this command by viewing the updated permissions with “ls -la”.



## Summary

Multiple permissions have been updated to the desired authorization the organization wanted in files, hidden files, and directories. This was completed using the “chmod” and “ls” commands to view and verify permissions.