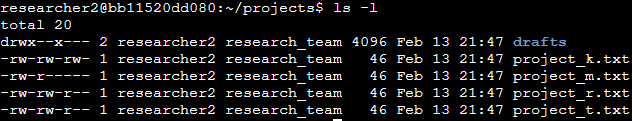
# File permissions in Linux

## Project description

Fictional Scenario: The research team at my organization needs to update the file permissions for certain files and directories within the “projects” directory. The permissions do not currently reflect the level of authorization that should be given. Checking and updating these permissions will help keep their system secure. For this project, I will be showing how to view the permissions for files and folders in Linux. I will also be explaining what exactly the permissions string is meant to convey as well as how to change the permissions for files if they need to be changed for different users. In addition, I will show how these commands can also be applied to hidden files as well as entire directories. To complete the tasks mentioned at the beginning, I performed the following tasks:

## Check file and directory details



Using the command ls with -l added on, we can list the contents of this directory along with the permissions. If we add on the “-a” argument as well, we can view the hidden files but that will be touched on in a later step.

## Describe the permissions string



Using this one string as an example, we can see what permissions this file has. The first character, which is a “-“, indicates that it is a file. If the first character was a “d” instead, that would indicate that it is a directory. There are 3 permissions, which are read, write and execute. The next 3 characters, which are “rw-“ are the user’s permissions. The “’r” stands for read and the “w” stands for write meaning that the user is able to read the contents of the file as well as edit it. The next 3 characters, which are “rw-“ are the permissions for the group meaning that the group is able to read and write to the file. The last 3 characters which have same permissions of the previous 2 are for other users. The execute permission is not enabled for any of the users but is typically indicated with a “x” in the 3rd space rather than the “-“. When any of the characters in the permission string besides the first one is a “-“, that means that the permission is not granted to the user, group, or other depending on where it is positioned.

## Change file permissions



The company decided that the “other users” group should not have write access to any of these files. To change the permissions, we can use the “chmod” command followed by the “o” for others and which permission we would like to add or remove. In this case, I removed the write permission from the “other users” group for the project\_k.txt file. This means that other users are still able to read the file but are no longer able to change the contents of it.

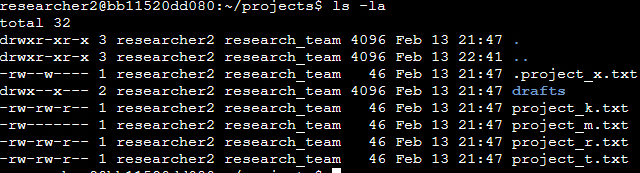


After running the command “chmod o-w project\_k.txt”, this is the new permission string for this file. As indicated in the screenshot, the third set of permissions, indicating the other users group, have been changed and the “W” is no longer there for other users meaning that they no longer have write permissions which is exactly what the organization needed and can be repeated for other files.

## Change file permissions on a hidden file

In a previous section, I demonstrated how to view folders and files in Linux using ls paired with the -l argument. However, this does not show hidden files. To view hidden files/folders we would need to use the -a argument with ls. We can actually pair this with the -l to show the permissions for the hidden content using ls -la such as in the following example:





Once we run this, we can now see new results that are different from when we only use “ls – l”. The hidden files/folders show up with a “.” before the file name. After adding the “-a” argument, we can see a file that the research team at the company archived titled “project\_x.txt”. Similar to how we can change permissions for a regular file, we can do the same for a hidden file. For this example, we can use the project\_x.txt file shown below:



For this example, we will pretend that the organization determined that the project\_x.txt file should not be written to by anyone but the user and the group should be able to read it. For us to make these changes, we can run the following commands:

’

From top to bottom in the commands above, we removed the write permission from the user (u-w), removed the write permission from the group (g-w) and granted the read permission to the group (g+r). Now, if we go back and check the permissions for this file. We can see that now the user and group can ONLY read the file but not write to it as shown below:



## Change directory permissions

As shown in the previous section in the folder that we are working with, there is a directory titled “drafts”. I will be using that directory to also show that these permissions can also be changed for entire directories. For reference, these are the current permissions on the “drafts” directory:



For this example, we can see that the group is able to access the drafts directory as well as its contents. The organization determined that only the user should have access to any of the contents in the “drafts” directory. We will be using the “chmod” command shown in the previous examples to edit these permissions. The command used to remove these permissions would be “chmod g-x drafts” indicating that we want to remove the execute permission from the group:



Now if we use “ls -l” to view the permissions, we can see that the “drafts” directory no longer has the execute permission for the group as shown below meeting the organizations requirements:



## Summary

Over the course of this project, I changed multiple permissions to match the level of authorization that the organization wanted for files and directories in the “projects” directory. The first step in this was using “ls” command with the -l argument to view the permissions for the directory. This assisted me with planning the steps I needed to take for the next few tasks. I then used then utilized the “chmod” command multiple times and in different ways to change the permissions on files and directories to match the needs.