File permissions in Linux

Name: Brian Trujillo

Project description

I am tasked with updating the file permissions for files and directories within the projects directory. My objective is to conduct a thorough review and adjustment of these permissions. The aim is to better understand command line operations and the nuances of file permissions within a Linux environment. This will involve analyzing the current permission sets, identifying any discrepancies, and modifying them to match our security requirements. Learning these fundamentals is crucial for maintaining the integrity and confidentiality of stored data. It ensures that correct access is granted, allowing authorized personnel to perform their duties effectively without compromising the organization's security protocols.

Describe the permissions string

```
researcher2@e491d3dcbb45:~$ ls
projects
researcher2@e491d3dcbb45:~$ cd projects
researcher2@e491d3dcbb45:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Dec 14 04:28 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Dec 14 04:28 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Dec 14 04:28 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 14 04:28 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 14 04:28 project_t.txt
researcher2@e491d3dcbb45:~/projects$
```

There are access rights for three types of users: the owner, the group, and others. Each string consists of 10 characters. The first character indicates the type of item: a dash (-) represents a file, and a d represents a directory. The next three characters are for the owner's permissions, followed by three for the group's permissions, and the final three for others. An example is:

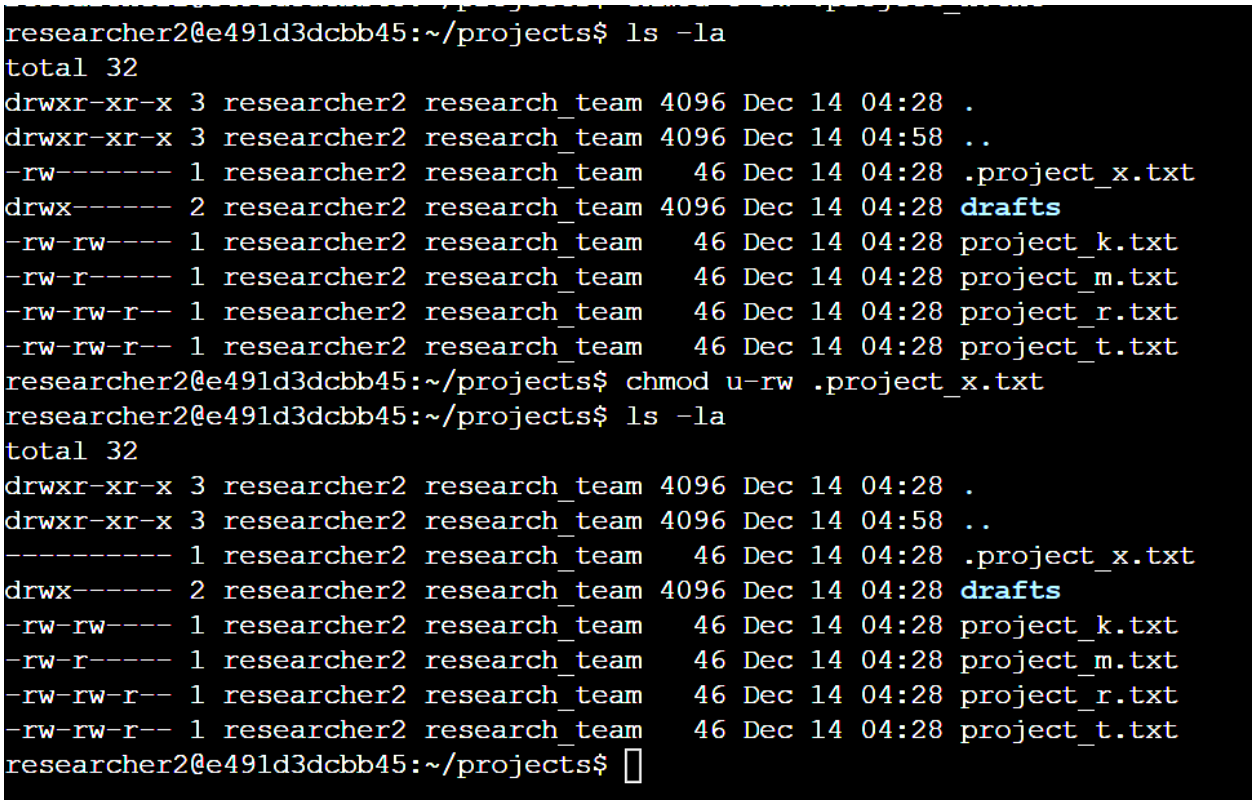
project_k.txt: -rw-rw-rw- means the owner,the group, and others have read and write permissions.

Change file permissions

```
researcher2@e491d3dcbb45:~/projects$ ls -l
total 20
drwx------ 2 researcher2 research_team 4096 Dec 14 04:28 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Dec 14 04:28 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Dec 14 04:28 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 14 04:28 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 14 04:28 project_t.txt
researcher2@e491d3dcbb45:~/projects$ chmod o-r,o-w project_k.txt
researcher2@e491d3dcbb45:~/projects$ ls -l
total 20
drwx------ 2 researcher2 research_team 4096 Dec 14 04:28 drafts
-rw-rw---- 1 researcher2 research_team   46 Dec 14 04:28 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Dec 14 04:28 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 14 04:28 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 14 04:28 project_t.txt
researcher2@e491d3dcbb45:~/projects$
```

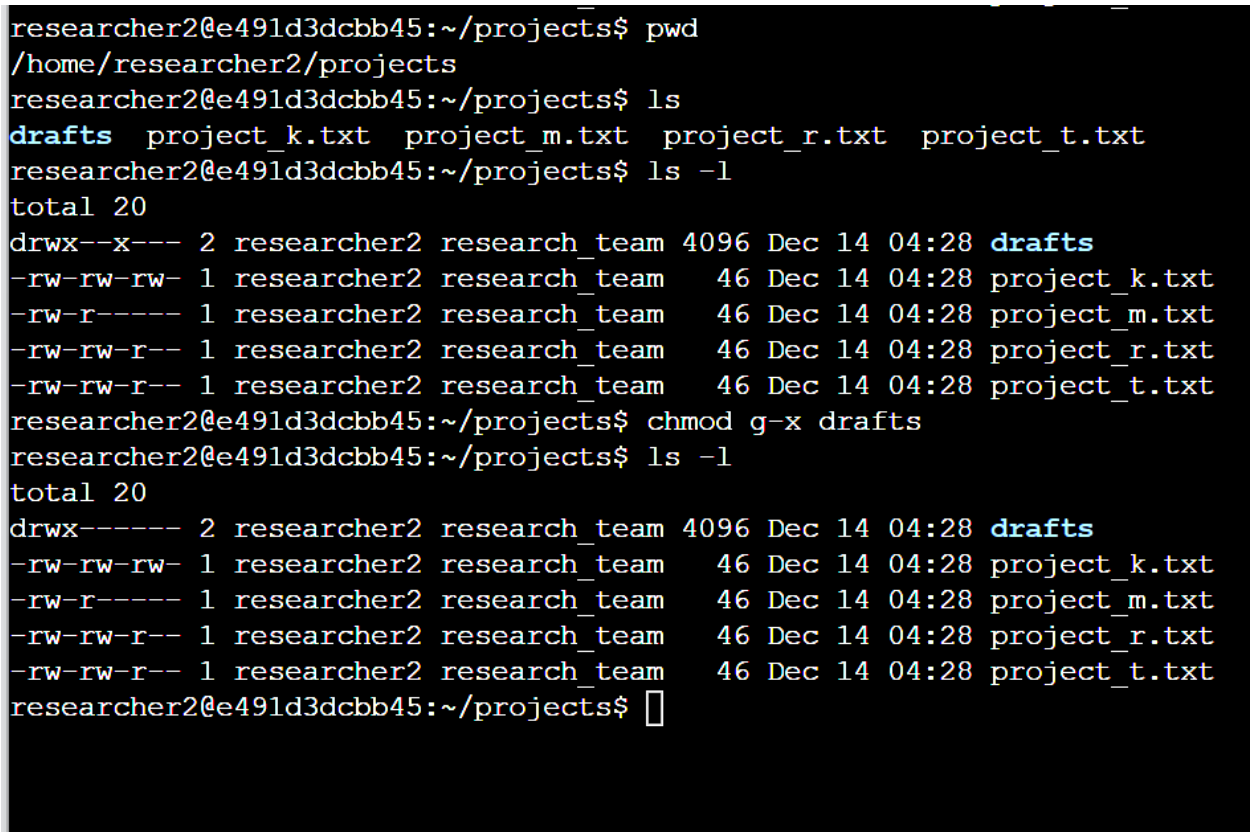
The ls -l command was executed within the projects directory to display detailed information about files, including their permissions. The file project_k.txt initially had open permissions (-rw-rw-rw-), allowing read and write access to the owner, group, and all other users. To change the file permissions, the command chmod o-r,o-w project_k.txt was used to alter these permissions, revoking both read and write access for users other than the owner and group. As a result, the file's new permissions (-rw-rw----) reflect that only the user and group in this case have read and write permissions.

Change file permissions on a hidden file



Viewing hidden files using LS -LA, thereafter adjusting permissions on a hidden file, .project_x.txt, initially had -rw------- permissions, meaning the owner had read and write access, while the group and others had none. Utilizing the chmod u-rw .project_x.txt command, the owner's read and write permissions were removed, resulting in ---------, a state where no user, including the owner, has permissions to read, write, or execute the file.

Change directory permissions



Modified the permissions for the 'drafts' directory. Initially, it was set with drwxr-x--- permissions, meaning the user had full access to read, write, and execute, group could only read and execute, and others had no access at all I ran the chmod g-x drafts command, which specifically removes the execute permission from the group category. After this change, the permission string updated to drwxr-----, effectively ensuring that group could still view files within the directory but could no longer execute any files or list the directory contents

Summary

Began by reviewing the current permissions using the ls -l command, which revealed that some files, like project_k.txt, had overly permissive settings, allowing read and write access to all users. I addressed this by using the chmod command to appropriately restrict these permissions, ensuring that only authorized personnel retained write access. When it came to hidden files, such as .project_x.txt, I discovered that the file permissions allowed the owner to read and write. I executed chmod u-rw .project_x.txt, stripping all permissions from the owner, group, and others, effectively rendering the file inaccessible to everyone. The directory's initial

permissions allowed group members to read and execute files within. I issued chmod g-x drafts, which removed the group's execute permission.