File permissions in Linux

Project description

This project is asking us to view, explain, and adjust permissions on files, hidden files, and directories. Please view screenshots and command explanations on how we accomplished these requests.

Check file and directory details

In the Manage authorization lab, check the permissions set for files and subdirectories in the projects directory. Make sure you display all permissions, including hidden files.

Describe the command you can use to check permissions in the Check file and directory details section of the File permissions in Linux template. From the lab, take a screenshot of the Linux command you used. Or, type this command directly into the template.

Then, use either the output of this command in the lab or the content or the Current file permissions document to indicate the current permissions. If using the Current file permissions document, write these in the 10-character string that would be part of the command's output.

```
researcher2@b6b2d5253a54:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug 31 16:01 .
drwxr-xr-x 3 researcher2 research_team 4096 Aug 31 17:16 ..
-rw--w---- 1 researcher2 research_team   46 Aug 31 16:01 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Aug 31 16:01 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Aug 31 16:01 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug 31 16:01 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug 31 16:01 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug 31 16:01 project_t.txt
```

We ran the **ls -la** command to view all directories, hidden files, and files in the projects directory. Based on the output, we can see there is one directory called drafts, one hidden file named .project_x.txt, and 4 regular files (project k, m, r, and t).

Describe the permissions string

Choose one example from the output in the previous step. In the Describe the permissions string section of the File permissions in Linux template, write a short description that explains the 10-character string in the example. You should describe what the 10-character string is for and what each character represents.

```
researcher2@b6b2d5253a54:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug 31 16:01 .
drwxr-xr-x 3 researcher2 research_team 4096 Aug 31 17:16 ..
-rw--w---- 1 researcher2 research_team   46 Aug 31 16:01 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Aug 31 16:01 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Aug 31 16:01 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug 31 16:01 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug 31 16:01 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug 31 16:01 project_t.txt
```

Character 1: First character is either a **d** or **hyphen** (**-**). If the first character is a d this states it's a directory. If it's a hyphen (-), it's a regular file.

Characters 2-4: These characters show permissions for the user owner and show read (r), write (w), and execute (x). If a character is a hyphen (-) that permission isn't granted to the user.

Characters 5-7: These characters show permissions for the group owner and show read (r), write (w), and execute (x). If a character is a hyphen (-) that permission isn't granted to the group.

Characters 8-10: These characters show permissions for the other owner and show read (r), write (w), and execute (x). If a character is a hyphen (-) that permissions isn't granted to other.

drafts – we can tell this is a directory because the first character in the permission string is 'd'. if it was a file it would show as '-'. The permissions for this directory are as follows:
- user – user has read, write and execute access
- group – group has execute access
- other – other doesn't have any access

.project_x.txt – this is a file because the first character is a '-'. This file is **hidden** which we found using the command **ls -la**. The 'a' portion shows the hidden files in the directory. Permissions for this file are as follows:
- user –read and write access
- group – write access
- other – no access

Change file permissions

The organization does not allow the other owner to have write access to any files. Based on the permissions established in Step 3, identify which file needs to have its permissions modified. Use a Linux command to modify these permissions.

Describe the command you used and its output in the Change file permissions section of the File permissions in Linux template. In the Manage authorization lab, take a screenshot of the Linux command you used. Or, type this command directly into the template.

Screenshot 1

```
researcher2@b6b2d5253a54:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug 31 16:01 .
drwxr-xr-x 3 researcher2 research_team 4096 Aug 31 17:16 ..
-rw--w---- 1 researcher2 research_team   46 Aug 31 16:01 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Aug 31 16:01 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Aug 31 16:01 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug 31 16:01 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug 31 16:01 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug 31 16:01 project_t.txt
```

Screenshot 2

```
researcher2@b6b2d5253a54:~/projects$ chmod o-w project_k.txt
researcher2@b6b2d5253a54:~/projects$ ls -la
total 32
drwxr-xr-- 3 researcher2 research_team 4096 Aug 31 16:01 .
drwxr-xr-- 3 researcher2 research_team 4096 Aug 31 17:16 ..
-rw--w---- 1 researcher2 research_team   46 Aug 31 16:01 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Aug 31 16:01 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Aug 31 16:01 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug 31 16:01 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug 31 16:01 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug 31 16:01 project_t.txt
```

Directions for this step indicate the **other** owner can't have any **write** access over any **files**. There are 5 files in this directory and the **other** owner only has write access over the **project_k.txt** file (screenshot 1). By using the **chmod o-w project_k.txt** command, we can remove the write access to this file for the other owner. We then run the command **ls -la** to confirm our update was successful (screenshot 2). The '**w**' that was showing as character 9 for the **project_k.txt** file in screenshot 1 is no longer showing.

Change file permissions on a hidden file

The research team has archived .project_x.txt, which is why it's a hidden file. This file should not have write permissions for anyone, but the user and group should be able to read the file. Use a Linux command to assign .project_x.txt the appropriate authorization.

Describe the command you used and its output in the Change file permissions on a hidden file section of the File permissions in Linux template. In the Manage authorization lab, take a screenshot of the Linux command you used. Or, type this command directly into the template.

Screenshot 1

```
researcher2@38d8e2636713:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Sep  2 15:23 .
drwxr-xr-x 3 researcher2 research_team 4096 Sep  2 16:19 ..
-rw--w---- 1 researcher2 research_team   46 Sep  2 15:23 .project_x.txt
```

Running the **ls -la** command allows us to see the owner permissions for the hidden file
.project_x.txt. We can see that the **user** owner currently has **read and write** access, and the
group owner has **write** access (screenshot 1). In order to align these permissions with the
request, we can run the following command: **chmod u-w,g-w,g+r .project_x.txt**
We then run the **ls -la** command once more to confirm only the user and group owners have
read access, as shown above. (screenshot 2)

Screenshot 2

```
researcher2@38d8e2636713:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@38d8e2636713:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Sep  2 15:23 .
drwxr-xr-x 3 researcher2 research_team 4096 Sep  2 16:19 ..
-r--r----- 1 researcher2 research_team   46 Sep  2 15:23 .project_x.txt
```

Change directory permissions

The files and directories in the projects directory belong to the researcher2 user. Only
researcher2 should be allowed to access the drafts directory and its contents. Use a Linux
command to modify the permissions accordingly.

Describe the command you used and its output in the Change directory permissions section
of the File permissions in Linux template. In the Manage authorization lab, take a screenshot of
the Linux command you used. Or, type this command directly into the template.

First we will run the **ls -l** command to get the current permissions for the **drafts** directory.
Since this is a directory, we don't need the 'a' after as drafts isn't hidden.

```
researcher2@38d8e2636713:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Sep  2 15:23 drafts
```

Output shows that **researcher2 has read, write and execute** access, but the **group** owner
also has **execute** access. To remove group execute access, we can run the command

chmod g-x. We then run **ls -l** to confirm our changes were successful and no one else other than the user **researcher2** has access to this directory.

```
researcher2@38d8e2636713:~/projects$ chmod g-x drafts
researcher2@38d8e2636713:~/projects$ ls -l
total 20
drwx------ 2 researcher2 research team 4096 Sep  2 15:23 drafts
```

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. I then used the **chmod** command multiple times to change the permissions on files and directories as my organization requested.