File permissions in Linux

Project description

In this scenario I'll examine and manage the permissions on the files and on a subdirectory in the `/home/researcher2/projects` directory for the `researcher2` user. I'll use basic Linux Bash shell commands to complete various tasks.

Check file and directory details

```
researcher2@0a0f8a243615:~$ pwd
/home/researcher2
researcher2@0a0f8a243615:~$ cd projects/
researcher2@0a0f8a243615:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Aug  6 12:15 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Aug  6 12:15 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  6 12:15 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:15 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:15 project_t.txt
researcher2@0a0f8a243615:~/projects$
```

After entering the projects subfolder with the `cd projects` command, all files and subfolders within that directory can be listed with the `ls` command. By adding the `-l` option all their permissions can be listed

Describe the permissions string

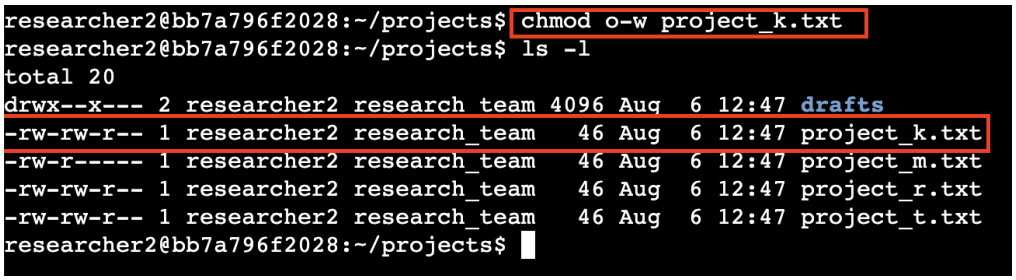
The 10 character string indicates how the permissions on the file or directory are set. The 1st character indicates the file type. The d indicates it's a directory. When this character is a hyphen (-), it's a regular file.
The remaining 3x3 characters indicate the read (r), write (w), and execute (x) permissions for the 3 owner types. When one of these characters is a hyphen (-) instead, it indicates that this permission is not granted to the given owner.
The 2nd-4th characters are for the user, the 5th-7th characters are for the group and the 8th-10th characters are for the other owner type.

In the above example the `project_t.txt` file has the following permission string:
`-rw-rw-r--`
1st character is a hyphen (-) that indicates a file. Both the user and the group have permissions to read (r) and to write (w), but not to execute the file. The other owner type has only read (r) permission.

Change file permissions

According to the scenario, the owner type other should not have permission to write any of the files in the `projects` folder. There are 2 ways to set the correct permission for the `project_k.txt` file. Both use the `chmod` (change mode) command. This command needs 2 parameters: the 1st indicates what needs to be set, the 2nd points to the file or directory that needs to be changed.

1. Removing the write permission:
   `chmod o-w project_k.txt`
   Meaning: for the other type (o) the write (w) permission to be removed (-) for the `project_k.txt` file
2. Setting the needed permission:
   `chmod o=r project_k.txt`
   Meaning: for the other type (o) the read (r) permission to be set (=) for the `project_k.txt` file. This removes all other permissions for the given owner type. To set read and write permissions, the 1st parameter should be `o=rw`



Change file permissions on a hidden file

According to the scenario, the hidden `.project_x.txt` file should not have write permissions for any owner type and only the user and the group should be allowed to read it. To list all files including hidden ones the `ls -la` command can be used (using `ls -a` list only the hidden files)
Before the changes, both user and group can write the hidden file and only the user can read it. To change this, multiple permissions can be given and removed using one command:

Option 1: `chmod u-w,g-w,g+r .project_x.txt`

- Meaning: remove write permission from user and group and add read permission for group

Option 2: `chmod u=r,g=r .project_x.txt`

- Meaning: set write permission for user and group

```
researcher2@bb7a796f2028:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug  6 12:47 .
drwxr-xr-x 3 researcher2 research_team 4096 Aug  6 13:30 ..
-rw--w---- 1 researcher2 research_team   46 Aug  6 12:47 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Aug  6 12:47 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  6 12:47 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_t.txt
researcher2@bb7a796f2028:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@bb7a796f2028:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug  6 12:47 .
drwxr-xr-x 3 researcher2 research_team 4096 Aug  6 13:30 ..
-r--r----- 1 researcher2 research_team   46 Aug  6 12:47 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Aug  6 12:47 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  6 12:47 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_t.txt
researcher2@bb7a796f2028:~/projects$
```

Change directory permissions

According to the scenario, only researcher2 can have access (execute permission) to the drafts folder.
For this the execute permission from the group owner should be removed using the following command:

```
chmod g-x drafts
```

```
researcher2@bb7a796f2028:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Aug  6 12:47 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  6 12:47 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_t.txt
researcher2@bb7a796f2028:~/projects$ chmod g-x drafts
researcher2@bb7a796f2028:~/projects$ ls -l
total 20
drwx------ 2 researcher2 research_team 4096 Aug  6 12:47 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  6 12:47 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  6 12:47 project_t.txt
researcher2@bb7a796f2028:~/projects$
```

Summary

Using the scenarios above, I wanted to demonstrate how to examine file and directory permissions, change permissions on files, on hidden files and on directories using basic Linux Bash shell commands.