File permissions in Linux

Project description

In this lab activity, you'll use Linux commands to configure authorization.

Authorization is the concept of granting access to specific resources in a system. It's important because without authorization any user could access and modify all files belonging to other users or system files. This would certainly be a security risk.

In Linux, file and directory permissions are used to specify who has access to specific files and directories. You'll explore file and directory permissions and change the ownership of a file and a directory to limit who can access them.

As a security analyst, setting appropriate access permissions is critical to protecting sensitive information and maintaining the overall security of a system.

Check file and directory details

Checked my current directory, which was "**/home/researcher2**"
Moved into the folder that i want to check permissions by command"**cd projects**".

```
researcher2@e65ae8cc17b2:~$ pwd
/home/researcher2
researcher2@e65ae8cc17b2:~$ ls
projects
researcher2@e65ae8cc17b2:~$ cd projects
researcher2@e65ae8cc17b2:~/projects$ pwd
/home/researcher2/projects
researcher2@e65ae8cc17b2:~/projects$ ls
drafts  project_k.txt  project_m.txt  project_r.txt  project_t.txt
```

To display all files and their permissions I used command "**ls**" with option "**-l**" -> "**ls -l**".
For instance, a directory with full permissions for all owner types would be "**drwxrwxrwx**":

```
researcher2@e65ae8cc17b2:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Jan 23 13:17 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jan 23 13:17 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jan 23 13:17 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 13:17 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 13:17 project_t.txt
researcher2@e65ae8cc17b2:~/projects$
```

Then I used command "**ls**" with options "**-l**" and "-**a**" -> "**ls -la**" and I received a list of all files, including hidden, and their permissions.

```
researcher2@e65ae8cc17b2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 13:17 .
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 14:13 ..
-rw--w---- 1 researcher2 research_team   46 Jan 23 13:17 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jan 23 13:17 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jan 23 13:17 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jan 23 13:17 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 13:17 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 13:17 project_t.txt
researcher2@e65ae8cc17b2:~/projects$
```

Let's take for example a file with the name "**.project_k.txt**" and its permissions.

First letter could be (**d**) directory or (**-**) file.

Next 9 letters are grouped into 3 groups. Letters 2,3,4 is **User** ; letter 5,6,7 is **Group** and letters 8,9,10 is **Other users**.

All groups have the same permission options (**r**) - read, (**w**) - write, (**x**) - execute. When there is a (**-**), instead of a letter, it means lack of permissions.

So, in this example which is "**-rw-rw-rw-**" means that:

- this is a **file**, not directory
- **User** has permission to **read** and **write**
- **Group** have permission to **read** and **write**
- **Other users** have permission to **read** and **write**

Describe the permissions string

Let's take for example a file with the name ".project_k.txt" and its permissions.
`-rw-rw-rw-`

First letter could be (**d**) directory or (**-**) file.

Next 9 letters are grouped into 3 groups. Letters 2,3,4 is **User** ; letter 5,6,7 is **Group** and letters 8,9,10 is **Other users**.

All groups have the same permission options (**r**) - read, (**w**) - write, (**x**) - execute. When there is a (**-**), instead of a letter, it means none permissions.

So, in this example which is "**-rw-rw-rw-** means that:

- this is a **file**
- **User** has permission to **read** and **write**
- **Group** have permission to **read** and **write**
- **Other users** have permission to **read** and **write**

None of the groups have permission to **execute** this file. That's why the (**x**) letter doesn't occur there.

Change file permissions

The organization does not allow others to have write access to any files. Based on the previous part, we can verify which of the files has an incorrect permission for **Other users**. That means, that we have to take a look at letters number: 8,9,10

```
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 13:17 .
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 14:13 ..
-rw--w---- 1 researcher2 research_team   46 Jan 23 13:17 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jan 23 13:17 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jan 23 13:17 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jan 23 13:17 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 13:17 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 13:17 project_t.txt
researcher2@e65ae8cc17b2:~/projects$
```

Only “**project_k.txt**” has a (**w**) letter and we have to change it.

```
researcher2@e65ae8cc17b2:~/projects$ chmod o-w project_k.txt
researcher2@e65ae8cc17b2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 13:17 .
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 14:13 ..
-rw--w---- 1 researcher2 research_team   46 Jan 23 13:17 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jan 23 13:17 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 13:17 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jan 23 13:17 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 13:17 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 13:17 project_t.txt
researcher2@e65ae8cc17b2:~/projects$
```

To do this we will use “**chmod”** command with option “**o-w**”
O - Other users
W - write
And use “**ls -la**” to verify results.

When there is minus (**-**) , it means that we **remove** privilege “(**w**) write” from group **Other users**. On the other hand, when we use (**+**), then we add privileges.

And then use again command “**ls -la**” to check permissions on that directory.
As we can see on projects “**project_k.txt**” 8,9,10 letters, there is “**r--**”

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.

Command “ls -la” to see hidden files with their permissions.

```
researcher2@ccd402d89fd2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 15:22 .
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 15:53 ..
-rw--w---- 1 researcher2 research_team   46 Jan 23 15:22 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jan 23 15:22 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jan 23 15:22 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jan 23 15:22 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 15:22 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 15:22 project_t.txt
researcher2@ccd402d89fd2:~/projects$
```

On the screenshot we can check that, user and group have incorrect permissions.
User has (**r**) - read and (**w**) - write
Group have (**w**) - write

This file should not have write permissions for anyone, but the user and group should be able to read the file.

To execute this I will use the command “**chmod**” with options “**u=r**” and “**g=r**” which means that user and group will have only read permission.

```
researcher2@ccd402d89fd2:~/projects$ chmod u=r,g=r .project_x.txt
```

Now we check again the permissions by command “**ls -la**” to verify results.

```
researcher2@ccd402d89fd2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 15:22 .
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 15:53 ..
-r--r----- 1 researcher2 research_team   46 Jan 23 15:22 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jan 23 15:22 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jan 23 15:22 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jan 23 15:22 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 15:22 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 15:22 project_t.txt
researcher2@ccd402d89fd2:~/projects$
```

Now the 10 letter code for hidden project “**.project_x.txt**” is “**-r--r----”**” which means that user and group has only permission to read that file. Success.

Change directory permissions

Check my current directory.
Move to the correct directory, where is the directory with incorrect permissions.

```
researcher2@ccd402d89fd2:~$ pwd
/home/researcher2
researcher2@ccd402d89fd2:~$ cd /home/researcher2/projects
researcher2@ccd402d89fd2:~/projects$
```

Check permissions by “**ls -la**”

```
researcher2@ccd402d89fd2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 15:22 .
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 15:53 ..
-r--r----- 1 researcher2 research_team   46 Jan 23 15:22 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jan 23 15:22 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jan 23 15:22 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jan 23 15:22 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 15:22 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 15:22 project_t.txt
researcher2@ccd402d89fd2:~/projects$
```

As we know, only the `researcher2` user should be allowed to access the `drafts` directory and its contents.
Now the 10 letter code is "**drwx-- x---** " which means that permission to execute has a user and group. So we have to remove group's permission by command "**chmod g-x drafts**"
And then we check our results by command "**ls -la**"

```
researcher2@ccd402d89fd2:~/projects$ chmod g-x drafts
researcher2@ccd402d89fd2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 15:22 .
drwxr-xr-x 3 researcher2 research_team 4096 Jan 23 15:53 ..
-r--r----- 1 researcher2 research_team   46 Jan 23 15:22 .project_x.txt
drwx------ 2 researcher2 research_team 4096 Jan 23 15:22 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jan 23 15:22 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jan 23 15:22 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 15:22 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan 23 15:22 project_t.txt
researcher2@ccd402d89fd2:~/projects$
```

Success again.

Summary

[Add content here.]

In this activity we showed how to check and modify permissions in Linux bash.
We learned a few new commands and options like: **chmod, ls -l, ls -a, ls -la**. We got to know the 10 letter code by which we can read the permissions and how to read it.

Permissions are very important for safety.