File permissions in Linux

Project description

In this project, the scenario was that I was a security professional at a large organization that works with the research team. Part of my job is to ensure that users on this team are authorized with the appropriate permissions in order to help keep the system secure. The task in this project was to examine existing permissions and determine if they match the authorization that should be given, and if not I needed to modify the permissions to authorize appropriate users and remove any unauthorized access.

Check file and directory details

Here, I use the command `ls -la` to list the contents of the `projects` directory and display the permissions using `-l`, and all hidden files or directories using `-a`, which can be combined to `-la`. Hidden files and directories cannot be normally seen when only using the `ls` command and therefore, are unlikely to be seen or used by everyday users, but can be an important consideration for tech professionals, especially when dealing with security.

```
researcher2@5081790a4976:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 May 28 23:32 .
drwxr-xr-x 3 researcher2 research_team 4096 May 28 23:55 ..
-rw--w---- 1 researcher2 research_team   46 May 28 23:32 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 May 28 23:32 drafts
-rw-rw-rw- 1 researcher2 research_team   46 May 28 23:32 project_k.txt
-rw-r----- 1 researcher2 research_team   46 May 28 23:32 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 May 28 23:32 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 May 28 23:32 project_t.txt
researcher2@5081790a4976:~/projects$
```

Describe the permissions string

If we take a look at the 10-character string associated with the `drafts` directory, we can see it as `drwx--x---`. The first character denotes whether the file is a directory or not, showing a `d` if yes or `-` if no. The next three characters, 2-4, are the file owner's read, write, and execute permissions, with `r` for read, `w` for write, and `x` for execute. Here, the owner, which is `researcher2`, has all three permissions. The next three characters, 5-7, are the group permissions, which is `research_team`. They cannot read or write to this file but may execute it, which for a directory means that they may enter into it and view its contents. The last three characters, 8-10, are the other users' permissions, which in this case is none, as they are all hyphens indicating the lack of permissions.

Change file permissions

The organization does not allow other users to write to any file; as such, the `project_k.txt` write permissions for other needs to be removed, which is done using `chmod o-w project_k.txt`. Additionally, `project_m.txt` is a restricted file that should only be readable and writable by the owner, so the group also needs its read privilege removed, done by `chmod g-r project_m.txt`. The `chmod` command allows you to specify which category of user you wish to modify permissions for, what permissions you wish to add, remove, or set, and upon which file you are modifying these permissions. The `u` represents the user/owner, `g` is the group, and `o` is the other users. We already discussed `rwx` for permissions, so now we can simply use `+` or `-` to add or remove one or any of these permissions from a category, as well as use `=` to instead set the permission to whatever we wish and overwrite the current permissions.

```
researcher2@5081790a4976:~/projects$ chmod o-w project_k.txt
researcher2@5081790a4976:~/projects$ chmod g-r project_m.txt
researcher2@5081790a4976:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 May 28 23:32 .
drwxr-xr-x 3 researcher2 research_team 4096 May 28 23:55 ..
-rw--w---- 1 researcher2 research_team   46 May 28 23:32 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 May 28 23:32 drafts
-rw-rw-r-- 1 researcher2 research_team   46 May 28 23:32 project_k.txt
-rw------- 1 researcher2 research_team   46 May 28 23:32 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 May 28 23:32 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 May 28 23:32 project_t.txt
researcher2@5081790a4976:~/projects$
```

Change file permissions on a hidden file

The research team has archived `.project_x.txt`, which is why it's a hidden file and should not have any write permissions on it. However, the user and group should still be able to read the file, so we use `chmod u-w,g=r .project_x.txt` to change the permissions to read-only for them.

```
researcher2@5081790a4976:~/projects$ chmod u-w,g=r .project_x.txt
researcher2@5081790a4976:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 May 28 23:32 .
drwxr-xr-x 3 researcher2 research_team 4096 May 28 23:55 ..
-r--r----- 1 researcher2 research_team   46 May 28 23:32 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 May 28 23:32 drafts
-rw-rw-r-- 1 researcher2 research_team   46 May 28 23:32 project_k.txt
-rw------- 1 researcher2 research_team   46 May 28 23:32 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 May 28 23:32 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 May 28 23:32 project_t.txt
researcher2@5081790a4976:~/projects$
```

Change directory permissions

The files and directories in the `projects` directory belong to the `researcher2` user, and so only `researcher2` should be allowed to access the drafts directory and its contents. Currently, the group can access it due to their execute permission, and so we will remove it using `chmod g-x drafts`. After doing so, we checked the permissions again and now see that only the user has permissions associated with the `drafts` directory.

```
researcher2@5081790a4976:~/projects$ chmod g-x drafts
researcher2@5081790a4976:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 May 28 23:32 .
drwxr-xr-x 3 researcher2 research_team 4096 May 28 23:55 ..
-r--r----- 1 researcher2 research_team   46 May 28 23:32 .project_x.txt
drwx------ 2 researcher2 research_team 4096 May 28 23:32 drafts
-rw-rw-r-- 1 researcher2 research_team   46 May 28 23:32 project_k.txt
-rw------- 1 researcher2 research_team   46 May 28 23:32 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 May 28 23:32 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 May 28 23:32 project_t.txt
researcher2@5081790a4976:~/projects$
```

Summary

In conclusion, there were a few files that needed permissions to be removed from them or slightly modified, including a hidden one, as well as one directory that needed to be adjusted to be in compliance with the authorization scheme the organization had laid out. Therefore, I used the `chmod` and `ls -la` commands to view the permissions and modify them to be in accordance with the desired permissions.