File permissions in Linux

Project description

I was given the security task of ensuring the right people had the appropriate permissions to help keep the system secure. I was able to accomplish this through linux commands and being able to understand permission strings.

Check file and directory details

The command to check permissions is *Ls -l*

```
drwx--x--- 2 researcher2 research_team 4096 Jan  1 15:58 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jan  1 15:58 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jan  1 15:58 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan  1 15:58 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jan  1 15:58 project_t.txt
```

Describe the permissions string

“Drwx- -x - - -”
This permission String is to identify if it is a normal file or a directory and what permissions different users have. Firstly we know it is a directory due to the permission string starting with a “d” if it was a regular file it would have a “-” to represent it. The first 3 characters after “d” represent the permissions that the user has which include r for read, w for write , and x for execute permissions. The next two “-” show us that the “group” user only has permission to execute. The “-” in a permission string represents when users don't have that permission which is why the “other” users don't have any permissions with that directory.

Change file permissions

```
-rw-rw-rw- 1 researcher2 research team   46 Jan  1 15:58 project k.txt
```

The first permission string shows that the “other”users have read and write permissions in the *project_k.txt* file.

```
researcher2@1e4588559a60:~/projects$ chmod o-w project k.txt
```

The command to remove write permission is *chmod o-w project_k.txt*
Chmod- is a command to change file or directory permissions. In the command we use “o” to represent others and the “-” to remove the “write” permission and we continue with the name of the file to ensure the permissions are being taken away in the right place.

```
-rw-rw-r-- 1 researcher2 research_team   46 Jan  1 15:58 project_k.txt
```

After the command we can see how the permissions change in the permission string with now “other” users only having read permissions.

Change file permissions on a hidden file

```
-rw--w---- 1 researcher2 research_team   46 Jan  1 15:58 .project_x.txt
```

Currently the hidden file has “write” permissions for both user and group users. They should only have “read” permissions so we will have to change that. In order to see a hidden file the command “*ls -la*” must be used to show the hidden file and its permissions

```
researcher2@1e4588559a60:~/projects$ chmod u-w,g-w,g+r .project_x.txt
```

With this command we will remove the users “write” permission leaving him with only “read” permissions and we also removed the write permission for the group but assigned them the read permission.

Change directory permissions

```
drwx--x--- 2 researcher2 research_team 4096 Jan  1 15:58 drafts
```

In the Drafts directory only the user should have access for it. Meaning the group users need their executable permissions removed.

```
researcher2@1e4588559a60:~/projects$ chmod g-x drafts
```

In order to remove permission from the group user we used the following command which leaves only the user with permissions to the drafts directory

Summary

With the instructions I was given to manage and examine existing permissions I was able to change a few for security strengthening. I was able to change file permissions on multiple files and directories including removing and adding permissions to certain groups. This includes hidden files as well. For example I was able to remove writing permissions from the “user” and add “read” permissions to the “group” users. Directories were also modified by removing the “group” users executable permissions from the “drafts” directory. This was all done in order to match the permissions with the authorization.