

File permissions in Linux

Project description

You are a security professional at a large organization. You mainly work with their research team. Part of your job is to ensure users on this team are authorized with the appropriate permissions. This helps keep the system secure.

Your task is to examine existing permissions on the file system. You'll need to determine if the permissions match the authorization that should be given. If they do not match, you'll need to modify the permissions to authorize the appropriate users and remove any unauthorized access.

Check file and directory details

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.

To check the file and directory details I had to first use the `cd` command to move into the project folder, I then used the `ls -la` command which shows all the permission and details of files, subdirectories and hidden files which in this case starts with `.` being `.project_x.txt`.

```
researcher2@af587f9844b2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug 28 01:54 .
drwxr-xr-x 3 researcher2 research_team 4096 Aug 28 02:44 ..
-rw--w---- 1 researcher2 research_team  46 Aug 28 01:54 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Aug 28 01:54 drafts
-rw-rw-rw- 1 researcher2 research_team  46 Aug 28 01:54 project_k.txt
-rw-r----- 1 researcher2 research_team  46 Aug 28 01:54 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Aug 28 01:54 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Aug 28 01:54 project_t.txt
researcher2@af587f9844b2:~/projects$
```

Describe the permissions string

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.

The 10-character string determines the authorization of accessing the file and their specific permissions. The characters and what they represent are as follows: We'll take the first row from the picture above:

`drwxr-xr-x`

1st character: This character is either a d or hyphen (-) and indicates the file type. Character d shows that it is a directory and drafts is the example. A hyphen (-) shows that it is a regular file.

2nd-4th characters: These characters indicate the read (r), write (w), and execute (x) permissions for the user. When one of these characters is a hyphen (-) instead, it indicates that this permission is not granted to the user.

5th-7th characters: These characters indicate the read (r), write (w), and execute (x) permissions for the group. When one of these characters is a hyphen (-) instead, it indicates that this permission is not granted for the group.

8th-10th characters: These characters indicate the read (r), write (w), and execute (x) permissions for others. It includes all other users on the system that are not users and the group. When one of these characters is a hyphen (-) instead, that indicates that this permission is not granted for others.

Change file permissions

The organization does not allow other 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.

I used the following command `chmod` to remove the write permissions of the one file in the other section. This accomplished the task of making the others have no access to write to any of the files or subdirectories in the directory

```
researcher2@af587f9844b2:~/projects$ chmod o-w project_k.txt
researcher2@af587f9844b2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug 28 01:54 .
drwxr-xr-x 3 researcher2 research_team 4096 Aug 28 02:44 ..
-rw--w---- 1 researcher2 research_team  46 Aug 28 01:54 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Aug 28 01:54 drafts
-rw-rw-r-- 1 researcher2 research_team  46 Aug 28 01:54 project_k.txt
-rw-r----- 1 researcher2 research_team  46 Aug 28 01:54 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Aug 28 01:54 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Aug 28 01:54 project_t.txt
researcher2@af587f9844b2:~/projects$
```

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.

I used the following `chmod` command but instead of using `+` or `-` I used the `=` signs which makes all the permission for the section equal to what I put into the command which in this case was overwrite the permission and change them to have user and group be the only ones to have read permissions and nothing for the rest.

```
researcher2@af587f9844b2:~/projects$ chmod u=r,g=r .project_x.txt
researcher2@af587f9844b2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug 28 01:54 .
drwxr-xr-x 3 researcher2 research_team 4096 Aug 28 02:44 ..
-r--r----- 1 researcher2 research_team  46 Aug 28 01:54 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Aug 28 01:54 drafts
-rw-rw-r-- 1 researcher2 research_team  46 Aug 28 01:54 project_k.txt
-rw-r----- 1 researcher2 research_team  46 Aug 28 01:54 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Aug 28 01:54 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Aug 28 01:54 project_t.txt
researcher2@af587f9844b2:~/projects$
```

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.

The first thing I did was go back into the root directory of the file system, and then changed the permission using the `chmod` command to remove all access to the contents of this directory

from group and other making only the files and subdirectories in the projects directory only accessible by the main user (researcher2).

```
researcher2@af587f9844b2:~/projects$ cd ..
researcher2@af587f9844b2:~$ ls
projects
researcher2@af587f9844b2:~$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug 28 02:44 .
drwxr-xr-x 1 root          root          4096 Aug 28 01:54 ..
-rw----- 1 researcher2 research_team  388 Aug 28 03:17 .bash_history
-rw-r--r-- 1 researcher2 research_team  220 Apr 18  2019 .bash_logout
-rw-r--r-- 1 researcher2 research_team 3574 Aug 28 01:54 .bashrc
-rw-r--r-- 1 researcher2 research_team 3574 Aug 28 01:54 .profile
drwxr-xr-x 3 researcher2 research_team 4096 Aug 28 01:54 projects
researcher2@af587f9844b2:~$ chmod g-rx,o-rx projects
researcher2@af587f9844b2:~$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug 28 02:44 .
drwxr-xr-x 1 root          root          4096 Aug 28 01:54 ..
-rw----- 1 researcher2 research_team  420 Aug 28 03:18 .bash_history
-rw-r--r-- 1 researcher2 research_team  220 Apr 18  2019 .bash_logout
-rw-r--r-- 1 researcher2 research_team 3574 Aug 28 01:54 .bashrc
-rw-r--r-- 1 researcher2 research_team 3574 Aug 28 01:54 .profile
drwx----- 3 researcher2 research_team 4096 Aug 28 01:54 projects
researcher2@af587f9844b2:~$
```

Summary

This scenario demonstrates my capability to match the level of authorization my organization set for files and directories in the project directory. The command `ls -la` displays all the files in the directory while `chmod` allows you to change permissions and directories.