Using Linux Commands To Manage File Permissions

Overview

This activity is completed as part of a training exercise for Coursera's Google Cybersecurity Certificate. It is conducted at a fictitious organization.

Scenario

The exercise sets out a scenario in which I, as security professional at an organization, must examine and manage the permissions on files and subdirectories, including hidden files in the ***/home/researcher2/projects*** directory for a user called ***researcher2*** that is part of the ***research2_team*** group.

My task is to ensure the permissions are aligned with the authorization that should be given for these files and directories and modify them if they don't comply. The goal is to remove unauthorized access and guarantee authorization for appropriate users.

Next, I will show all the steps followed using Linux command lines.

Check file and directory details

First , the ***cd*** command is used to go to the *project* directory for the user *research2*. Then the command ***ls -la*** is executed to display a detailed content of the directory including hidden files. The following screenshot shows the command and the output obtained after it was executed.

```
researcher2@1a13504e4350:~$ ls
projects
researcher2@1a13504e4350:~$ cd projects
researcher2@1a13504e4350:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jun  5 23:02 .
drwxr-xr-x 3 researcher2 research_team 4096 Jun  5 23:11 ..
-rw--w---- 1 researcher2 research_team   46 Jun  5 23:02 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jun  5 23:02 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jun  5 23:02 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jun  5 23:02 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  5 23:02 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  5 23:02 project_t.txt
researcher2@1a13504e4350:~/projects$
```

The output shows a first column with a 10 character string that represents the permission set for each file or directory. There is a subdirectory called ***drafts***, a hidden file that starts with a point(**.**) called ***.project_x.txt*** and four additional ***.txt*** project files. The third column is the **user** and the fourth is the **group**.

Describe the permissions string

The string has 10 characters, the meaning of each one is as follow:

Examples: drwxrwxrwx , -rwxrwxrwx, d-w--w–w-, -r-x-w-x--x.

- The **1s**t character represents the file type, **d** for directory and hyphen**(-**) for a regular file.
- The **2nd-4th** characters represent the permissions granted to the **user**, **r** stands for read, **w** for write, **x** for execute and hyphen**(-)** means that permission is not granted.The **user** is the owner of the file.
- The **5th-7th** characters represent the permissions granted to the **group**, **r** stands for read, **w** for write, **x** for execute and hyphen(**-**) means that permission is not granted. The **group** is to which the user owner of the file belongs.
- The **8h-10th** characters represent the permissions granted to **others (represent all other users)**, **r** stands for read, **w** for write, **x** for execute and hyphen(**-**) means that permission is not granted.

Change file permissions

The organization determined that **other** can not have write permission to any file.
Based on the current permissions , **project_k.txt** is identified as the only file the write access must be removed for **other**. To accomplish this task the command **chmod o-w project_k.txt** is used, chmod changes the permissions, the first argument **o-w** indicates to remove the write access for **other** and the second argument specifies the file or directory.

```
researcher2@1a13504e4350:~/projects$ chmod o-w project_k.txt
researcher2@1a13504e4350:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jun  5 23:02 .
drwxr-xr-x 3 researcher2 research_team 4096 Jun  5 23:11 ..
-rw--w---- 1 researcher2 research_team   46 Jun  5 23:02 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jun  5 23:02 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Jun  5 23:02 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jun  5 23:02 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  5 23:02 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  5 23:02 project_t.txt
researcher2@1a13504e4350:~/projects$
```

The screenshot shows the command executed for the change and the additional **ls-la** command is to verify it. As it can be seen in the output ,the **project_k.txt** file allows read access only for **other**.

Change file permissions on a hidden file

The **.project.txt** file has been archived by the research team and they need that nobody has write permission on it, but the **user** and **group** should read the file.

Based on the current access permissions , it is identified that the **write** permission must be removed for **user** and **group** and the **read** access must be granted for **group.**
As it can be seen on the screenshot , the chmod command was used to modify the permissions of this hidden file. In this case the first argument is **u-w,g-w+r** where minus sign(-) is to remove access permission and plus(+) for granting it.

```
researcher2@1a13504e4350:~/projects$ chmod u-w,g-w+r .project_x.txt
researcher2@1a13504e4350:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jun  5 23:02 .
drwxr-xr-x 3 researcher2 research_team 4096 Jun  5 23:11 ..
-r--r----- 1 researcher2 research_team   46 Jun  5 23:02 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jun  5 23:02 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Jun  5 23:02 project_k.txt
-rw------- 1 researcher2 research_team   46 Jun  5 23:02 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  5 23:02 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  5 23:02 project_t.txt
researcher2@1a13504e4350:~/projects$
```

Change directory permissions

The company has determined that only the user **researcher2** should have access to the **draft** directory and its contents.
From the current permissions status obtained in the first step of this activity, the **group** has **execute(x)** access to this directory, so it is necessary to remove it with a linux command.
The following screenshot shows the command used to update directory permissions **chmod g-x drafts**, the first argument **g-x** indicates that for **group** the execute **(-x)** access is removed **(-).**
The output of the command **ls-la** confirms that only **user** has **read(r), write(w) and execute(x**) access to draft directory.

```
researcher2@1a13504e4350:~/projects$ chmod g-x drafts
researcher2@1a13504e4350:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jun  5 23:02 .
drwxr-xr-x 3 researcher2 research_team 4096 Jun  5 23:11 ..
-r--r----- 1 researcher2 research_team   46 Jun  5 23:02 .project_x.txt
drwx------ 2 researcher2 research_team 4096 Jun  5 23:02 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Jun  5 23:02 project_k.txt
-rw------- 1 researcher2 research_team   46 Jun  5 23:02 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  5 23:02 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  5 23:02 project_t.txt
researcher2@1a13504e4350:~/projects$
```

Summary

In this activity it was used the command **ls -la** to determine the current permissions status for files and directories in the **/home/researcher2/project** directory, then the command **chmod** was used with different arguments (**o-w , u-w, g-w+r, g-x)** and files to update the permissions as needed to match them with the authorizations required by the organization.