Exercises about file and directory permission

STUDENT: Marta Muñoz San Román

1. List the permissions in your current directory, including hidden files.
To do this, the -a parameter will show the hidden files starting with a dot (.). Then, I use ls and -l parameter to check the permissions of an existing file or dir.

ls -la

```
marta@marta-VirtualBox:~$ ls -la
total 152
drwxr-xr-x 20 marta marta 4096 ene 13 16:07 .
drwxr-xr-x  5 root  root  4096 nov 19 02:25 ..
-rw-rw-r--  1 marta marta   79 dic  1 20:05 actors.txt
-rw-rw-r--  1 marta marta  276 dic  1 18:53 awards.txt
-rw-------  1 marta marta 2580 dic 16 18:04 .bash_history
-rw-r--r--  1 marta marta  220 sep 30 16:58 .bash_logout
-rw-r--r--  1 marta marta 3771 sep 30 16:58 .bashrc
drwxr-xr-x 15 marta marta 4096 nov 10 18:35 .cache
drwx------ 16 marta marta 4096 nov 24 17:43 .config
drwxrwxr-x  2 marta marta 4096 nov 10 18:16 config
-rw-rw-r--  1 marta marta 1024 nov 24 11:57 .critical.sh.swp
drwxr-xr-x  2 marta marta 4096 ene 12 19:18 Desktop
-rw-rw-r--  1 marta marta   80 dic  8 19:54 directors.txt
```

2. Create a file called perm1. Now, check the default permissions and user and group ownership.
To create a new empty file, I can use the touch command. Then, I will procede to check the default permissions with ls -l:

touch perm1
ls -l perm1

```
marta@marta-VirtualBox:~$ ls -l perm1
-rw-rw-r-- 1 marta marta 0 ene 13 16:11 perm1
```

It has read and write permissions.

3. Change permissions of perm1 so that everyone can read and only the owner user can write. Specify the command in all possible ways.
To change it, I have to execute chmod. Al the possible ways could be these ones:

chmod a=r,u=rw perm1
chmod a=r,u+w perm1
chmod ugo a=r,u=rw perm1
chmod ugo=r,u+w perm1

Or with the octal mode, which is the quickest way in this case (where there are more than one permission in each user, group and others):

chmod 644 perm1

```
marta@marta-VirtualBox:~$ chmod a=r,u=rw perm1
marta@marta-VirtualBox:~$ ls -l perm1
-rw-r--r-- 1 marta marta 0 ene 13 16:11 perm1
```

4. Create a file called script1.sh, including the content below. List the default permissions.

```
#!/bin/bash
clear
who
```

So, in order to create a file and write information on it I will use the nano command. To list it, I will use the ls -l script1.sh:

nano script1.sh
ls -l script1.sh

```
marta@marta-VirtualBox:~$ nano script1.sh
marta@marta-VirtualBox:~$ ls -l script1.sh
-rw-rw-r-- 1 marta marta 22 ene 13 16:25 script1.sh
```

5. Remove the read permission from the owner and try to open the file.

To modify permissions, I will use the chmod in this way (with the -operator):

chmod u-r script1.sh

Then, I will try to read it with the nano command:

nano script1.sh

```
chmod u-r script1.sh
nano script1.sh
```

BUT it will throw the following error:

```
[ Error reading script1.sh: Permission denied ]
^O Write Out    ^W Where Is    ^K Cut Text    ^J Ju
```

6. Remove the write permission from the owner on the file script1.sh. Add the line below. Is it possible? Why?

new line

To remove the write permission, I will use the hmod in this way (with the -operator):

chmod u-w script1.sh

```
marta@marta-VirtualBox:~$ chmod u-w script1.sh
marta@marta-VirtualBox:~$ ls -l script1.sh
----rw-r-- 1 marta marta 22 ene 13 16:25 script1.sh
```

While I try to write a line using nano and overwrite the file, it throws the following error:

nano script1.sh

```
[ Error writing script1.sh: Permission denied ]
^O Write Out    ^W Where Is    ^K Cut Text    ^J
```

7. Change the permissions on the file script1.sh so that the owner can read, write and execute, but you deny all the permissions from the group and others.

To do this, I can create the permissions by using the octal mode or the symbolic one. The symbolic one will be done in this way:

chmod u=rwx,go-rwx script1.sh

```
marta@marta-VirtualBox:~$ chmod u=rwx,go-rwx script1.sh
marta@marta-VirtualBox:~$ ls -l script1.sh
-rwx------ 1 marta marta 22 ene 13 16:25 script1.sh
```

With the octal mode, the command will look like this:

chmod 700 script1.sh

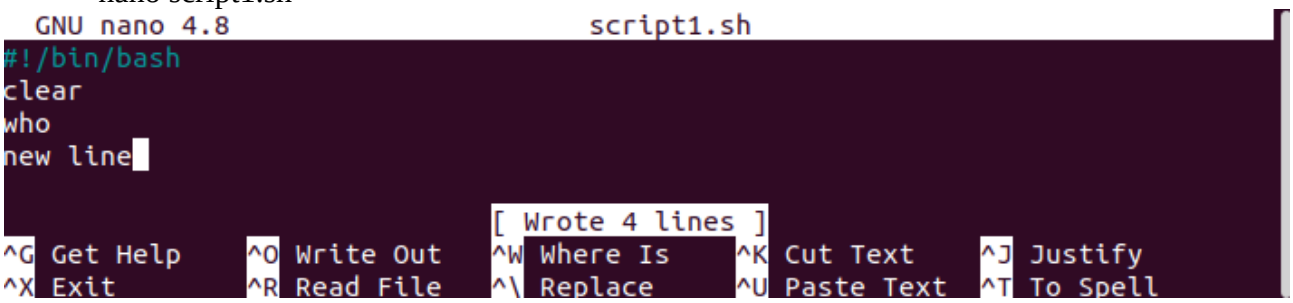
```
marta@marta-VirtualBox:~$ chmod 700 script1.sh
marta@marta-VirtualBox:~$ ls -l script1.sh
-rwx------ 1 marta marta 22 ene 13 16:25 script1.sh
```

8. Add the line indicated in exercise 6, in case it was not possible. Try to run the file like a command.

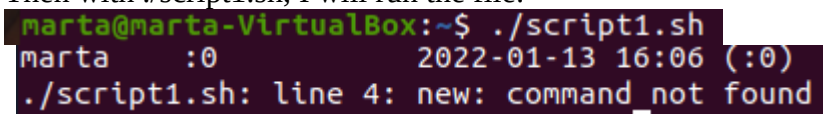
new line

In exercise 6, writing the file it was not possible. But now it's possible by using nano:

nano script1.sh



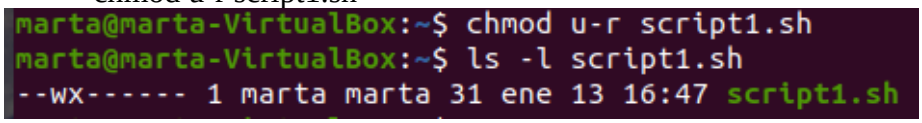
Then with ./script1.sh, I will run the file:



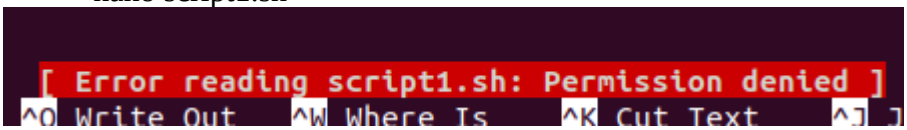
9. Remove the read permission from the owner on the file script1.sh. Try to run the file. Is it possible?

I will use the chmod to modify the required permission:

chmod u-r script1.sh



nano script1.sh



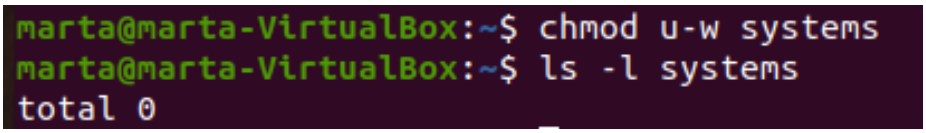
10. Create a directory called "systems". Remove the write permission from it and try to copy script1.sh inside.

Firstly, I will create the directory using the mkdir command:

mkdir systems

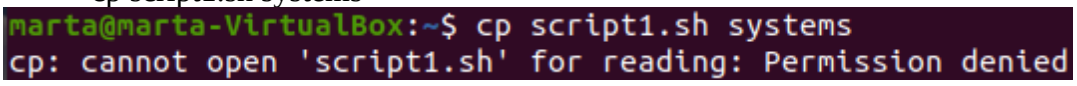
Then, I will take away the write permission ,from the owner for example:

chmod u-w systems



If I try to copy that, it seems to be impossible:

cp script1.sh systems



11. If you were not able to copy the file, add the write permission again and copy the file inside.

Unhopefully, as I wasn't able to copy the file, I will need again to change permissions with chmod with the + operator:

chmod u+w systems

```
marta@marta-VirtualBox:~$ chmod u+w script1.sh
marta@marta-VirtualBox:~$ ls -l script1.sh
--wx------ 1 marta marta 31 ene 13 16:47 script1.sh
```

Then, I will try to copy the file, adding read permissions to script1.sh, because you cannot copy a file which cannot be read:

chmod u+r script1.sh

cp script1.sh systems

```
marta@marta-VirtualBox:~$ cp script1.sh systems
```

12. Remove the read permission from the user on the directory "systems" and try to list its contents.

Using the chmod command:

chmod u-r systems

And then the ls command:

ls systems

```
marta@marta-VirtualBox:~$ chmod u-r systems
marta@marta-VirtualBox:~$ ls systems
ls: cannot open directory 'systems': Permission denied
```

13. Change the permissions from "sytems" so that the owner can read, write and execute, but the group and others can only read.

To do this, I will use chmod in this way:

chmod u=rwx,go=r systems

```
marta@marta-VirtualBox:~$ chmod u=rwx,go=r systems
marta@marta-VirtualBox:~$ ls -l systems
total 0
```

If I change it with the octal mode, it will give the following result:

chmod 744 systems

14. Remove the execute permission from "systems". Can you execute systems/script1.sh? Is it possible to acces the directory to execute the file?

To do this, I will use again the chmod command:

chmod u-x systems

If I try to execute the file systems/script1.sh with nano, it will throw the following error:

nano systems/script1.sh

```
marta@marta-VirtualBox:~$ chmod u-x systems
marta@marta-VirtualBox:~$ nano systems/script1.sh
```

```
[ Path 'systems' is not accessible ]
ite Out   ^W Where Is   ^K Cut Text
```

to the directory again

To assign it, I will use the chmod and the + operator:

chmod u+wx systems

```
marta@marta-VirtualBox:~$ chmod u+wx systems
marta@marta-VirtualBox:~$ ls -l systems
total 4
-rw-rw--wx 1 marta marta  0 ene 13 17:19 charles
-r-xrw--w- 1 marta marta  0 ene 13 17:19 lucy
-rwx------ 1 marta marta 31 ene 13 16:47 script1.sh
```

16. Create two files called "lucy" and "charles" into "systems". Change permissions of "charles", so that others can write and execute.

I will use this commands and parameters to create those files, inside the systems dir:

cd systems
touch lucy charles

```
marta@marta-VirtualBox:~$ cd systems
marta@marta-VirtualBox:~/systems$ touch lucy charles
marta@marta-VirtualBox:~/systems$ ls
charles  lucy  script1.sh
```

To change the permissions for “charles”, I will use chmod:

chmod o=wx charles

```
marta@marta-VirtualBox:~/systems$ chmod o=wx charles
marta@marta-VirtualBox:~/systems$ ls -l charles
-rw-rw--wx 1 marta marta 0 ene 13 17:19 charles
```

17. Change permissions of “lucy” so that the owner can read and execute, the group can read and write and others can only write. Specify the command in all possible ways.

To change the permissions for “charles”, I will use chmod:

chmod u=rx,g=r,o=w lucy
chmod u=rx,go=rw,o-r lucy
chmod 562 lucy

```
marta@marta-VirtualBox:~/systems$ chmod 562 lucy
marta@marta-VirtualBox:~/systems$ ls -l lucy
-r-xrw--w- 1 marta marta 0 ene 13 17:19 lucy
```

18. Log in as root. Change the ownership of “charles” to “root”. Exit the root session. Now, try to change the permission so that others cannot read and execute. Is it possible? Why?

To enter into root mode, and change the ownership:

Sudo su
chown root charles

```
marta@marta-VirtualBox:~/systems$ sudo su
[sudo] password for marta:
root@marta-VirtualBox:/home/marta/systems# chown root charles
root@marta-VirtualBox:/home/marta/systems#
```

If I try to change the permissions, with “chmod 562 charles” for example:

```
marta@marta-VirtualBox:~/systems$ chmod 562 charles
chmod: changing permissions of 'charles': Operation not permitted
marta@marta-VirtualBox:~/systems$
```

It’s not possible to change the permissions, because the owner is root.

19. Change the permissions of “charles” so that everybody can do everything

To do this, I will enter in root mode again and execute the following command, changing root to “marta”:

sudo su
chown marta charles

Now, If I try to change the permissions of “charles” with “chmod 562 charles”, it’s possible:

```
marta@marta-VirtualBox:~/systems$ sudo su
root@marta-VirtualBox:/home/marta/systems# chown marta charles
root@marta-VirtualBox:/home/marta/systems# exit
exit
marta@marta-VirtualBox:~/systems$ chmod 562 charles
```

ç

20. Change the permissions of "lucy" so that the group can read and write, but the owner and others cannot do anything. Can you open the file?

For this exercise, it's better to use the octal mode because I need to change all permissions, and I use "ls" to check it:

chmod 060 lucy
ls -l lucy

```
marta@marta-VirtualBox:~/systems$ chmod 060 lucy
marta@marta-VirtualBox:~/systems$ ls -l lucy
----rw---- 1 marta marta 0 ene 13 17:19 lucy
```

21. Create a group called "newgroup". Set the group as the owner of the file "lucy" and "root" as the owner user.

To create a group, I will use the "groupadd" command, and then change its permissions with chown:

cd ..
sudo groupadd newgroup

```
marta@marta-VirtualBox:~$ sudo groupadd newgroup
[sudo] password for marta:
marta@marta-VirtualBox:~$ cd systems
```

Then, with this group added, I will enter in root session again to change the ownership of lucy:

cd systems
sudo su
chown root:newgroup lucy

```
marta@marta-VirtualBox:~/systems$ sudo su
root@marta-VirtualBox:/home/marta/systems# chown root:newgroup lucy
root@marta-VirtualBox:/home/marta/systems#
```

22. Add your user to the secondary group "newgroup". Try to open the file "lucy" now. Is it possible?

To add my own user to the newgroup group, I use the following commands:

sudo usermod -a -G newgroup marta

```
root@marta-VirtualBox:/home/marta/systems# sudo usermod -a -G newgroup marta
```

When I exit the root session, and try to open the file with "nano":

nano lucy

It throws the following error:

```
[ Error reading lucy: Permission denied ]
^W Where Is     ^K Cut Text
```

23. Change permissions of "lucy" so that everybody can read.

To do that, in root session, I enter as root and change the ownership with chown:

sudo su
chown marta lucy

```
root@marta-VirtualBox:/home/marta/systems# chown marta lucy
root@marta-VirtualBox:/home/marta/systems#
```

24. Do exercise 13 again, but this time granting permissions to the folder "systems" including files and subfolders.

The exercise 13 was:

Change the permissions from "sytems" so that the owner can read, write and execute, but the group and others can only read.

With the octal mode, I first recreate the permissions that the exercise 13 says, adding the -R parameter:

chmod -R 744 systems

```
marta@marta-VirtualBox:~$ chmod -R 744 systems
marta@marta-VirtualBox:~$ ls -l systems
total 4
-rwxr--r-- 1 marta newgroup  0 ene 13 17:19 charles
-rwxr--r-- 1 marta newgroup  0 ene 13 17:19 lucy
-rwxr--r-- 1 marta newgroup 31 ene 13 16:47 script1.sh
```

Comparison with the exercise 20:

```
marta@marta-VirtualBox:~/systems$ ls -l lucy
----rw---- 1 marta marta 0 ene 13 17:19 lucy
```

25. Change the group owner of "systems" to "root" including files and subfolders.

To change it, in root session, I execute the command:

chgrp -R root systems

```
root@marta-VirtualBox:/home/marta# chgrp -R root systems
root@marta-VirtualBox:/home/marta# ls -l systems
total 4
-rwxr--r-- 1 marta root  0 ene 13 17:19 charles
-rwxr--r-- 1 marta root  0 ene 13 17:19 lucy
-rwxr--r-- 1 marta root 31 ene 13 16:47 script1.sh
```