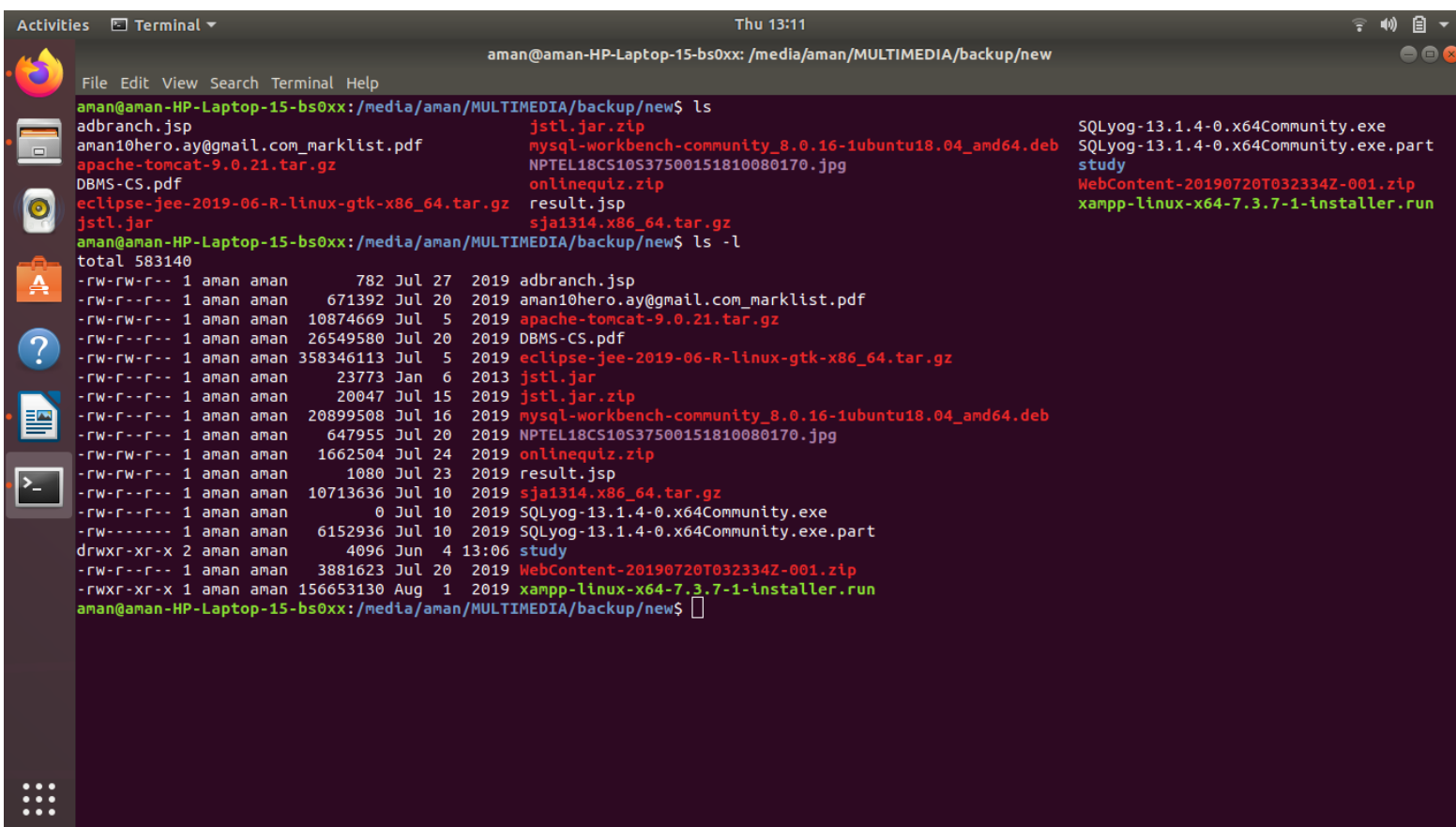# File Security in Linux

Linux is a multi-user operating system, so it has security to prevent people from accessing each other's confidential files.Linux file security is quite simplistic in design, yet quite effective in controlling access to files and directories.Directories and the files which are stored in them are arranged in a hierarchical tree structure. Access can be controlled for both the files and the directories allowing a very flexible level of access.

When we execute the "ls" command, we are not given any information about the security of the files, because by default "ls" only lists the names of files. We can get more information by using an "option" with the "ls" command. All options start with a '-'. For example, to execute "ls" with the "long listing" option, we would type ls -l, When we do so, each file will be listed on a separate line in long format.

There's a lot of information in those lines.

The first character will almost always be either a '-', which means it's a file, or a 'd', which means it's a directory.

The next nine characters (rw-r- -r- -) show the security;
The next column shows the owner of the file.
The next column shows the group owner of the file.
The next column shows the size of the file in bytes.
The next column shows the date and time the file was last modified.
And, of course, the final column gives the filename.

Understanding the security permissions :-

First, we must think of those nine characters as three sets of three characters Each of the three "rwx" characters refers to a different operation we can perform on the file.

---	---	---
rwx	rwx	rwx
user	group	other

Read, write, execute and –

The 'r' means you can "read" the file's contents. The 'w' means you can "write", or modify, the file's contents. The 'x' means you can "execute" the file. This permission is given only if the file is a program. If any of the "rwx" characters is replaced by a '-', then that permission has been revoked.

User, group and others :-

user – The user permissions apply only the owner of the file or directory, they will not impact the actions of other users.

group – The group permissions apply only to the group that has been assigned to the file or directory, they will not effect the actions of other users.

others – The others permissions apply to all other users on the system, this is the permission group that you want to watch the most.

Reading the security permissions :-

For example, consider that the user's permissions for some files is "rw-" as the first three characters. This means that the owner of the file can "read" it (look at its contents) and "write" it (modify its contents). We cannot execute it because it is not a program; it is a text file.If "r-x" is the second set of 3 characters it means that the members of the group can only read and execute the files. The final three characters show the permissions allowed to anyone who has a UserID on this Linux system. Let us say we have the permission ("r–"). This means anyone in our Linux world can read, but they cannot modify the contents of the files or execute it.

Changing security permissions

The command you use to change the security permissions on files is called "chmod", which stands for "change mode", because the nine security characters are collectively called the security "mode" of the file.

1. The first argument you give to the "chmod" command is 'u', 'g', 'o'. We use : u for user, g for group, o for others, you can also use a combination of them (u,g,o).This specifies which of the three groups you want to modify.
2. After this use a '+' for adding a '-' for removing and a "=" for assigning a permission.
3. Then specify the permission r,w or x you want to change. Here also you can use a combination of r,w,x.This specifies which of the three permissions "rwx" you want to modify
4. We can use commas to modify more permissions Finally, the name of the file whose permission you are changing

For example, if you want to give "execute" permission to the world ("other") for file "abc.txt", you would start by typing

➢ chmod o

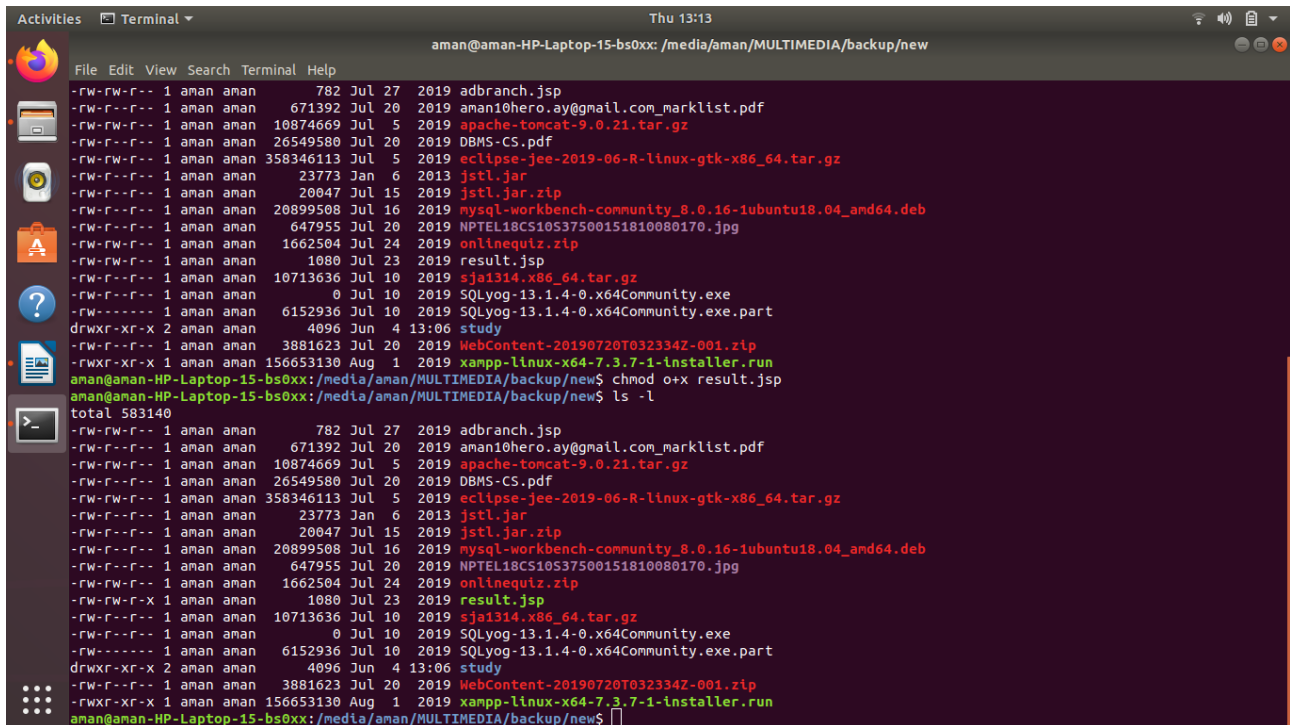Now we can type a '+' to say that we are "adding" a permission.

➢ chmod o+

Then we can type an 'x' to say that we are adding "execute" permission.

➢ chmod o+x

Finally, specify which file we are changing.

➢ chmod o+x result.jsp

we can see the change in the picture below.

You can also change multiple permissions at once. For example, if you want to take all permissions away from everyone, you would type

➢ chmod ugo-rwx result.jsp

The code above revokes all the read(r), write(w) and execute(x) permission from all user(u), group(g) and others(o) for the file result.jsp which results to this.

```
Activities    Terminal ▾                          Thu 13:14                                    ⚡ ◀) 🔋 ▾
                          aman@aman-HP-Laptop-15-bs0xx: /media/aman/MULTIMEDIA/backup/new        ⊖ ⊖ ⊗
File  Edit  View  Search  Terminal  Help
-rw-rw-r-- 1 aman aman       782 Jul 27  2019 adbranch.jsp
-rw-r--r-- 1 aman aman    671392 Jul 20  2019 aman10hero.ay@gmail.com_marklist.pdf
-rw-rw-r-- 1 aman aman  10874669 Jul  5  2019 apache-tomcat-9.0.21.tar.gz
-rw-r--r-- 1 aman aman  26549580 Jul 20  2019 DBMS-CS.pdf
-rw-rw-r-- 1 aman aman 358346113 Jul  5  2019 eclipse-jee-2019-06-R-linux-gtk-x86_64.tar.gz
-rw-r--r-- 1 aman aman     23773 Jan  6  2013 jstl.jar
-rw-r--r-- 1 aman aman     20047 Jul 15  2019 jstl.jar.zip
-rw-r--r-- 1 aman aman  20899508 Jul 16  2019 mysql-workbench-community_8.0.16-1ubuntu18.04_amd64.deb
-rw-r--r-- 1 aman aman    647955 Jul 20  2019 NPTEL18CS10S37500151810080170.jpg
-rw-rw-r-- 1 aman aman   1662504 Jul 24  2019 onlinequiz.zip
-rw-rw-r-x 1 aman aman      1080 Jul 23  2019 result.jsp
-rw-r--r-- 1 aman aman  10713636 Jul 10  2019 sja1314.x86_64.tar.gz
-rw-r--r-- 1 aman aman         0 Jul 10  2019 SQLyog-13.1.4-0.x64Community.exe
-rw------- 1 aman aman   6152936 Jul 10  2019 SQLyog-13.1.4-0.x64Community.exe.part
drwxr-xr-x 2 aman aman      4096 Jun  4 13:06 study
-rw-r--r-- 1 aman aman   3881623 Jul 20  2019 WebContent-20190720T032334Z-001.zip
-rwxr-xr-x 1 aman aman 156653130 Aug  1  2019 xampp-linux-x64-7.3.7-1-installer.run
aman@aman-HP-Laptop-15-bs0xx:/media/aman/MULTIMEDIA/backup/new$ chmod ugo-rwx result.jsp
aman@aman-HP-Laptop-15-bs0xx:/media/aman/MULTIMEDIA/backup/new$ ls -l
total 583140
-rw-rw-r-- 1 aman aman       782 Jul 27  2019 adbranch.jsp
-rw-r--r-- 1 aman aman    671392 Jul 20  2019 aman10hero.ay@gmail.com_marklist.pdf
-rw-rw-r-- 1 aman aman  10874669 Jul  5  2019 apache-tomcat-9.0.21.tar.gz
-rw-r--r-- 1 aman aman  26549580 Jul 20  2019 DBMS-CS.pdf
-rw-rw-r-- 1 aman aman 358346113 Jul  5  2019 eclipse-jee-2019-06-R-linux-gtk-x86_64.tar.gz
-rw-r--r-- 1 aman aman     23773 Jan  6  2013 jstl.jar
-rw-r--r-- 1 aman aman     20047 Jul 15  2019 jstl.jar.zip
-rw-r--r-- 1 aman aman  20899508 Jul 16  2019 mysql-workbench-community_8.0.16-1ubuntu18.04_amd64.deb
-rw-r--r-- 1 aman aman    647955 Jul 20  2019 NPTEL18CS10S37500151810080170.jpg
-rw-rw-r-- 1 aman aman   1662504 Jul 24  2019 onlinequiz.zip
--------- 1 aman aman      1080 Jul 23  2019 result.jsp
-rw-r--r-- 1 aman aman  10713636 Jul 10  2019 sja1314.x86_64.tar.gz
-rw-r--r-- 1 aman aman         0 Jul 10  2019 SQLyog-13.1.4-0.x64Community.exe
-rw------- 1 aman aman   6152936 Jul 10  2019 SQLyog-13.1.4-0.x64Community.exe.part
drwxr-xr-x 2 aman aman      4096 Jun  4 13:06 study
-rw-r--r-- 1 aman aman   3881623 Jul 20  2019 WebContent-20190720T032334Z-001.zip
-rwxr-xr-x 1 aman aman 156653130 Aug  1  2019 xampp-linux-x64-7.3.7-1-installer.run
aman@aman-HP-Laptop-15-bs0xx:/media/aman/MULTIMEDIA/backup/new$ ▯
```

The octal notations :-

We can also use octal notations as given below :-

Octal	Binary	File Mode
0	000	---
1	001	--x
2	010	-w-
3	011	-wx
4	100	r--
5	101	r-x
6	110	rw-
7	111	rwx

Using the octal notations table instead of 'r', 'w' and 'x'. Each digit octal notiation can be used of either of the group 'u','g','o'.

So, the following work the same.

chmod ugo+rwx [file_name]
chmod 777 [file_name]

Both of them provides full read write and execute permission (code=7) to all the group.

Same is the case with this..

chmod u=r,g=wx,o=rx [file_name]
chmod 435 [file_name]

Both the codes give read (code=4) permission to user, write and execute (code=3) for group and read and execute (code=5) for others.

And even this…

chmod 775 [file_name]
chmod ug+rwx,o=rx [file_name]

Both the commands give all permissions (code=7) to user and group, read and execute (code=5) for others.