

## File Security/ Permission Command

- Linux being a multi-user system uses permissions and ownership for security.
  - There are three user types on a Linux system viz. User, Group and Other  
Linux divides the file permissions into read, write and execute denoted by r,w, and x
  - The permissions on a file can be changed by 'chmod' command which can be further divided into Absolute and Symbolic mode.
  - The 'chown' command can change the ownership of a file/directory. Use the following commands: chown user file or chown user:group file.
  - The 'chgrp' command can change the group ownership **chgrp group filename**
  - What does x - eXecuting a directory mean? A: Being allowed to "enter" a dir and gain possible access to sub-dirs.

1. ls -l

2.chmod permissions filename

The characters are pretty easy to remember.

**r** = read permission

**w** = write permission

**x** = execute permission

**-** = no permission

There are 2 ways to use the command -

1.Absolute mode

2.Symbolic mode

### **Absolute(Numeric) Mode**

In this mode, file **permissions are not represented as characters but a three-digit octal number.**

The table below gives numbers for all for permissions types.

Number	Permission Type	Symbol
0	No Permission	---
1	Execute	--x
2	Write	-w-
3	Execute + Write	-wx
4	Read	r--
5	Read + Execute	r-x
6	Read + Write	rw-
7	Read + Write + Execute	rwX

## Symbolic Mode

In the Absolute mode, you change permissions for all 3 owners. In the symbolic mode, you can modify permissions of a specific owner. It makes use of mathematical symbols to modify the file permissions.

Operator	Description
+	Adds a permission to a file or directory
-	Removes the permission
=	Sets the permission and overrides the permissions set earlier.

The various owners are represented as -

User Denotations
u user/owner
g group
o other
a all