chown command in Linux

Different users in the operating system have ownership and permission to ensure that the files are secure and put restrictions on who can modify the contents of the files

In Linux, there are different users who use the system:

Root User: It is a *superuser* who has access to all the directories and files in our system and it can perform any operation. An important thing to note is that only the root user can perform changing of permissions or ownerships of the files that are not owned by them.

Regular User: These users have limited access to files and directories and can only modify the file that they own.

Each *user* has some properties associated with them, such as a user ID and a home directory. We can add users to a group to make the process of managing users easier. A *group* can have zero or more users. A specified user can be associated with a "default group". It can also be a member of other groups on the system as well.

Ownership and Permissions: To protect and secure files and directories in Linux we use permissions to control what a user can do with a file or directory. Linux uses three types of permissions:

- **Read:** This permission allows the user to read files in directories, it lets the user read directories and subdirectories stored in it.
- **Write:** This permission allows a user to modify and delete a file. Also, it allows a user to modify its contents (create, delete, and rename files in it) for the directories. Unless the execution permission is given to directories changes do affect them.

- **Execute:** This permission on a file allows it to get executed. For example, if we have a file named *php.sh* so unless we don't give it execute permission it won't run.

  User: This type of file permission affects the owner of the file.

  Group: This type of file permission affects the group which owns the file. Instead of the group permissions, the user permissions will apply if the owner user is in this group.

  Other: These types of file permission affect all other users on the system.

Note: To view the permissions we use:

ls –l

chown command is used to change the file Owner or group. Whenever you want to change ownership, you can use chown command.

Syntax:

chown [OPTION]… [OWNER][:GROUP] FILE…
chown [OPTION]… –reference=RFILE FILE…

To change the Owner of the file:

Syntax:

chown <owner_name> <file_name>

[root@ip-10-7-2-205 devops]# ls -l file1
-rw-r--r-- 1 root root 319 May 16 00:04 file1
[root@ip-10-7-2-205 devops]# chown kumar file1
[root@ip-10-7-2-205 devops]# ls -l file1
-rw-r--r-- 1 kumar root 319 May 16 00:04 file1

To change the group of the file:

the group1 group is assigned as the group of file1.txt

Syntax:

```
chown  :<group_name> <filename>
```

```
[root@ip-10-7-2-205 devops]# chown :kumar file1
[root@ip-10-7-2-205 devops]# ls -l
total 4
-rw-r--r-- 1 kumar kumar 319 May 16 00:04 file1
```

To change both owner and group of the file:

the user <kumar> is assigned as the owner and the group <kumar> is assigned as the group of file.

Syntax:

```
chown <groupname>:<ownername> <filename>
```

```
[root@ip-10-7-2-205 devops]# chown kumar:kumar file2
[root@ip-10-7-2-205 devops]# ls -l file2
-rw-r--r-- 1 kumar kumar 165 May 16 00:58 file2
```