To modify an existing user, like adding that user to a new group, use :

DATE: PAGE:

Users, Groups and Permissions

Cmd : sudo usermod -a -G <groupName(s)> <userName>

-a : append to the list of groups user belongs to

-G : allows multiple group names to be given separated by ,

To add a user :

Cmd : sudo adduser <username> <optional groupname>

To see which group new user has been added to use: groups <user name>

o/p : [sudo] password for kunal: 1

o/p : Enter new UNIX password :

i/p : 1

o/p : Retype new UNIX password :

i/p : 1

o/p : passwd : password updated successfully.

Changing the user information for <username> Appears on terminal

FULL NAME []:

i/p : Parag Jain → Appears when you log in

o/p : Is this information correct ?

i/p : y

To remove a user :

Cmd : sudo userdel <username>

To remove a user from a group

sudo deluser <username> <groupname>

To see everything about a user do :

Cmd : sudo gedit /etc/passwd

To change the passwd for user :

cmd : sudo passwd <username>

o/p : Enter new UNIX password :

i/p :

o/p : Retype new UNIX password :

i/p :

To add a group :

Cmd : sudo agroupadd <groupname>

To delete a group :

Cmd : Sudo groupdel <groupname>

DATE: PAGE:

To see everything about a group:

Cmd: sudo gedit /etc/group

To change the permission of a file

Cmd: sudo chmod 777 <filename>

To change the permission of a folder

Cmd: sudo chmod 777 <foldername> -R

How do you change the ownership of a file or folder?

We know that each file/folder has a user owner and a group owner associated with it.

To change the user owner of a file.

Cmd: sudo chown <username> <filename>

To change the user owner of a folder:

Cmd: sudo chown -R <username> <foldername>

To change the group owner of a file

Cmd: sudo chgrp <groupname> <filename>

To change the group owner of a folder

Cmd: sudo chgrp -R <groupname> <foldername>

DATE: PAGE:

Execute Permission on the directory :

Without " " " " " , you can't

i) stat ii) open iii) rename iv) delete

v) Descend into subdirectories inside that directory.

The only thing you can do is see the list of which filenames exist, and then only if you have read permission (and read but not execute is a strange set of permissions to have for a directory).

Consider you have [rw-] on a directory. You know that filename [foo] exists inside this " ". In order to delete it you need to look it up, and you even need access to the inode (to decrement its link count. For that matter, you need access to the inode in order to tell if its a directory or not (because if its a directory, [unlink] should fail and [rmdir] should succeed, and the reverse if its not a directory).

With the execute bit set on the directory :

i) You can [cd] into the directory.

ii) Also for long listing [ls -l] i.e., to view the metadata of the files inside the directory (Provided that [read] permission is there for the directory).

For a directory,

i) The write bit allows the affected user to create, rename or delete files within the directory, and modify the directory's attributes.

ii) The read bit allows the affected user to list the files within the directory.

iii) The execute bit allows the " " to enter the directory, and access files and directories inside.

iv) The sticky bit states that files " " within that directory may only be deleted or renamed by their owner.