

# User Management

User management is one of the main tasks of Linux administrators . It involves creating, modifying and deleting the user account, as well as managing user permissions and access. Here the some basic commands related to user management:-

## 1. Add user:

This command is used to create a new user account.

here's the syntax:-

**Sudo adduser <username>**

```
(suman@suman)~$ sudo adduser testuser
[sudo] password for suman:
info: Adding user `testuser' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `testuser' (1001) ...
info: Adding new user `testuser' (1001) with group `testuser (1001)' ...
info: Creating home directory `/home/testuser' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for testuser
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] n
Changing the user information for testuser
Enter the new value, or press ENTER for the default
  Full Name []: test user
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
info: Adding new user `testuser' to supplemental / extra groups `users' ...
info: Adding user `testuser' to group `users' ...

(suman@suman)~$ seinfo
$

(suman@suman)~$ su testuser
Password:
(testuser@suman)~/home/suman$ whoami
testuser
```

**Note:**

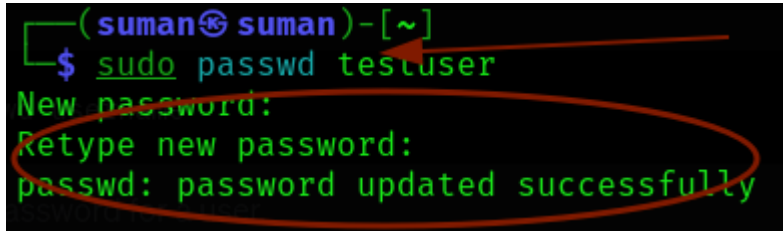
**whoami** is used to display the current username.

**su** command is used to switch user.

## 2. Changing user Password

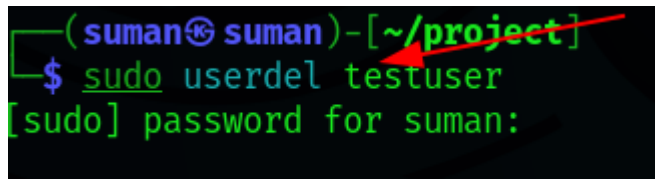
Here the syntax to change the password of user:-

**sudo passwd <username>**

A terminal window showing a user named 'suman' at a prompt. The user enters the command 'sudo passwd testuser'. The terminal displays 'New password:', followed by 'Retype new password:', and finally 'passwd: password updated successfully'. A red oval highlights the password prompts and the success message. A red arrow points to the command 'sudo passwd testuser'.

## 3. Deleting the user:-

To delete a user in linux we use 'userdel' command. Here's the example:-

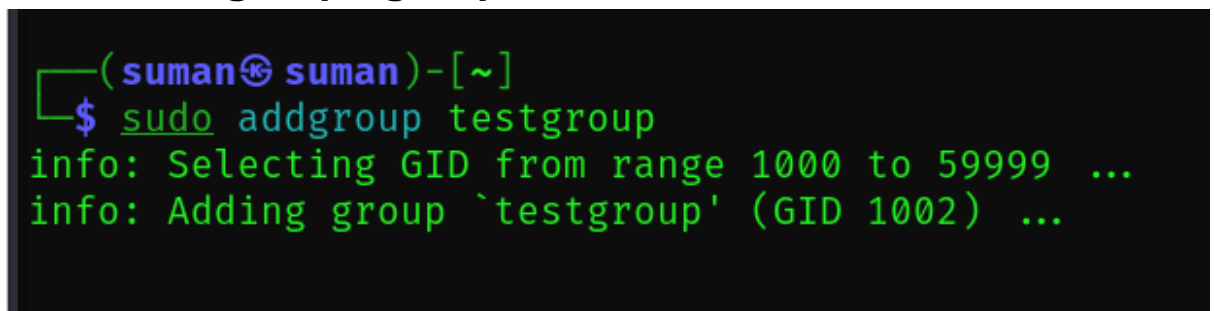
A terminal window showing a user named 'suman' at a prompt. The user enters the command 'sudo userdel testuser'. The terminal displays '[sudo] password for suman:'. A red arrow points to the command 'sudo userdel testuser'.

# Group management

## 1. Adding a Group

Here the syntax to add a new group in linux:

**Sudo addgroup <groupname>**

A terminal window showing a user named 'suman' at a prompt. The user enters the command 'sudo addgroup testgroup'. The terminal displays 'info: Selecting GID from range 1000 to 59999 ...' and 'info: Adding group `testgroup' (GID 1002) ...'.

## 2. Adding existing user to a group:-

Here is the command add existing user in group :-

**sudo usermod -aG <groupname> <username>**

In this command -aG stand for "append to the group list"

Here the example to add a user named "testuser" to group named "testgroup" :-

```
(suman@suman)~$  
$ sudo usermod -aG testgroup testuser  
$ groups testuser  
testuser : testuser users testgroup
```

### 3. Deleting a Group:

Here is the command to delete group:-

**sudo delgroup <groupname>**

Here the example of deleting group i have created:-

```
(suman@suman)~$  
$ sudo delgroup testgroup  
info: Removing group `testgroup' ...
```

## File and Directory Permissions:

### 1. Change the ownership of a file or directory:-

To change the ownership of a file or directory in linux, we use the 'chown' command . Here's the basic syntax and example:

**sudo chown <desired owner:group or group ID> <file or directory>**

```
(suman@suman)-[~/project]
$ ls -l
total 8
drwxr-xr-x 2 suman suman 4096 Jan 23 17:39 test_dir
-rw-r--r-- 1 suman suman 29 Jan 21 13:05 testfile

(suman@suman)-[~/project]
$ sudo chown testuser:testgroup test_dir

(suman@suman)-[~/project]
$ sudo chown testuser:testgroup testfile

(suman@suman)-[~/project]
$ ls -l
total 8
drwxr-xr-x 2 testuser testgroup 4096 Jan 23 17:39 test_dir
-rw-r--r-- 1 testuser testgroup 29 Jan 21 13:05 testfile
```

In this figure I have a directory named “test\_dir” and a file named “testfile” . Initially, both were owned by the user ‘suman’ and the group ‘suman’.

After I use the “sudo chown” command to change the ownership of both ‘test\_dir’ and ‘testfile’ to a new user named ‘testuser’ and a new group named ‘testgroup’ .

## Chmod

Files and directories have associated permissions that define who can do what with them . Permission are represented by a series of letters and symbols, often displayed using ‘ls -l’ command . Here’s a brief explanation of file and directory permissions in linux:-

```
(suman@suman)-[~/project]
$ ls -l
total 8
drwxr-xr-x 2 testuser testgroup 4096 Jan 23 17:39 test_dir
-rw-r--r-- 1 testuser testgroup 29 Jan 21 13:05 testfile

(suman@suman)-[~/project]
```

a). In directories :

‘drwxr-xr-x’ :- this represent the type and permissions of the directory :-

'd' = directory  
'rwx' = permissions of the owner (testuser)  
'r-x' = permissions for the group (testgroup)  
'r-x' = permissions for others

## b). In files:

'-rw-r--r-- 1 :-this represent the type and permissions of the directory :-

'-' = file  
'rw' = permissions of the owner (testuser)  
'r' = permissions for the group (testgroup)  
'r' = permissions for others

Here's the detailed overview of symbolic and numeric representation in permission:-

### a). Symbolic Representation:-

- r: Read
- w: Write
- x: Execute
- -: No permission

eg: if we need to change the file permission to read, write and execute to all (user, group and others) in symbolic we use this command:-

**sudo chmod u+rwx,g+rwx,o+rwx <filename>**

```
(suman@suman)-[~/project]
$ ls -l
total 0
-rw-r--r-- 1 suman suman 0 Jan 23 19:27 testfile.txt

(suman@suman)-[~/project]
$ chmod u+rwx,g+rwx,o+rwx testfile.txt

(suman@suman)-[~/project]
$ ls -l
total 0
-rwxrwxrwx 1 suman suman 0 Jan 23 19:27 testfile.txt
```

### b). Numeric Representation:-

- 4: Read
- 2: Write
- 1: Execute

eg: if we need to change the file permission to read, write and execute to all (user, group and others) in numeric we use this command:-

**sudo chmod 777 <filename>**

```
(suman@suman)-[~/project]
$ sudo chmod 777 testfile.txt

(suman@suman)-[~/project]
$ ls -l
total 0
-rwxrwxrwx 1 suman suman 0 Jan 23 19:27 testfile.txt
```

## Chown

The 'chown' command is use to changes the owner or group of file or directory .  
Here is the basic syntax and example of chown:-

**sudo chown <owner:group> <file\_or\_dir\_>**

```
(suman@suman)-[~/project]
$ ls -l
total 0
-rwxrwxrwx 1 suman suman 0 Jan 23 19:27 testfile.txt

(suman@suman)-[~/project]
$ sudo chown testuser:testgroup testfile.txt

(suman@suman)-[~/project]
$ ls -l
total 0
-rwxrwxrwx 1 testuser testgroup 0 Jan 23 19:27 testfile.t
xt
```

# Chgrp

The 'chgrp' command is specially used to change the group ownership of a file or directory . Here is the basic syntax and example:-

**sudo chgrp <group name > <file\_or\_dir>**

This command changes the ownership of 'testfile.txt' to the group 'prod\_group'

```
(suman@suman)-[~/project]
$ sudo groupadd prod_group

(suman@suman)-[~/project]
$ sudo chgrp prod_group testfile.txt

(suman@suman)-[~/project]
$ ls -l
total 0
-rwxrwxrwx 1 testuser prod_group 0 Jan 23 19:27 testfile.txt
```

