

# User management in Unix / Linux

Unix supports a concept of Group Account, which groups a number of accounts logically. Each account will be part of any group account. The Unix team plays an important role in executing process management and allowing access to files.

There are 3 types of accounts on a Unix system:

1. **Root account** : It is also called superuser and will have absolute control to the system. A superuser can run any command without restriction. This user can be likened to a system manager.
2. **System accounts** : **System** accounts are needed for system-specific operations such as mail accounts and sshd accounts. These accounts are usually needed for some specific functions on your system, and any modifications to them may adversely affect the system.
3. **Individual user accounts** : These accounts provide interactive access to the system with users and user groups and are often limited to access to important files and folders.

Unix supports a concept of Group Account, which groups a number of accounts logically. Each account will be part of any group account. The Unix team plays an important role in executing process management and allowing access to files.

## User management in Unix / Linux

1. Man and group management used in Unix / Linux
2. Create a group in Unix / Linux
3. Editing a group in Unix / Linux
4. Delete a group in Unix / Linux
5. Create a personal account in Unix / Linux
6. Edit an account
  1. Check account
7. Delete an account in Unix / Linux

## Man and group management used in Unix / Linux

There are 4 main files to manage users:

1. / etc / passwd: Keep user accounts and password information. This file holds important information about accounts on Unix systems.
2. / etc / shadow: Keep the password compiled into the password of the corresponding account. Not all systems support this file.

3. / etc / group: This file holds group information for each account.
4. / etc / gshadow: This file holds the security group account information.

You can check all the above files with the cat command.

The following commands are available in most Unix systems to create and manage individual and group accounts.

Command Description  
useradd Adds individual accounts to the system.  
usermod Edit the properties of personal accounts.  
userdel Delete individual accounts from the system.  
groupadd Add group accounts to the system.  
groupmod Edit the properties of the group account.  
groupdel Remove group accounts from the system.

## Create a group in Unix / Linux

You may need to create groups before creating any account, otherwise you must use existing groups on your system. You will have all the groups listed in the / etc / groups file.

All default groups will be specific account groups on the system and it is not recommended to use them for regular accounts. Therefore, below is the syntax to create a new account group.

```
groupadd [- g gid [- o ]] [- r ] [- f ] groupname
```

The following table details the parameters:

Optional	Description
-g	GID Number value of group ID.
-o	This option allows to add groups with GID not unique.
-r	This sign indicates grouping to the system account.
-f	This option causes it to only exit with a successful state if the specified group already exists. With -g, if GID already exists, then another (unique) GID is selected.
groupname	The group name is actually created.

If you do not specify any parameters, the system will use the default values.

The following example will create a group of developers with default values, which are approved by most managers.

```
$ groupadd developers
```

## Editing a group in Unix / Linux

To edit a group, use **the groupmod command** syntax:

```
$ groupmod - n new_modified_group_name old_group_name
```

To change the developers\_2 group name to developer, type the following:

```
$ groupmod - n developer developer_2
```

Here's how to change GID to 545:

```
$ groupmod - g 545 developers
```

## Delete a group in Unix / Linux

To delete an existing group, all you need to do is **groupdel** and the group name. To delete the developer group, the command is:

```
$ groupdel developer
```

This command only removes the group, not any files related to the group. The file is still accessible by its owner.

## Create a personal account in Unix / Linux

Let us see how to create a new personal account on your Unix system. Here is the syntax for creating a personal account:

```
useradd - d homedir - g groupname - m - s shell - u userid accountname
```

The following table details the parameters:

Options	Description
-d homedir	Specify the main directory for the account.
-g groupname	Specify a group account for this personal account.
-m	Create the main directory if it does not exist.
-s shell	Define the default shell for this personal account.
-u userid	You can specify a personal ID for this account.
accountname	The personal account name is actually created.

If you do not specify any parameters, the system will use the default values. The **useradd** command modifies / etc / passwd, / etc / shadow, / etc / group and creates a master directory.

Below is an example that will create a mcmohd account to set its home directory to / home / mcmohd and the group as developers. This user is Kenny Main that is assigned to it.

```
$ useradd - d / home / mcmohd - g developers - s / bin / ksh mcmohd
```

Before announcing the above command, make sure you already have developers groups created with the **groupadd** command.

When a personal account is created, you can set the password for it using the **passwd** command as follows:

```
$ passwd mcmohd20 Changing password for user mcmohd20 . New UNIX password : R
```

When you type **passwd accountname**, it gives you the option to change the password provided if you are superuser, otherwise you can only change the password using the same command but do not specify your account name. friend.

## Edit an account

Instructions for changing usernames in Linux below are carried out on Ubuntu Server, but the process will be similar on most other Linux distributions.

The **usermod** command gives you the ability to make changes to an existing personal account from the command line. It uses arguments like the **useradd** command, plus the **-l** argument, which allows changing the

account name.

A prerequisite is that you have access to user accounts with sudo permission (and the username must be changed). Let's follow the steps below!


Suppose you need to rename the testaccount account (already on the system) to haversham. To do this, enter the command:

```
sudo usermod -l haversham testaccount
```

At this point, the username has changed. However, the main directory associated with the username is still the **testaccount**. To change that, enter the command:

```
sudo usermod -d /home/haversham -m haversham
```

If **ls /home /** command is given , the user will see the main directory that reflects the new username.

A terminal window screenshot showing the process of renaming a user account. The terminal title is 'jack@hive4:~'. The output shows the creation of a user 'testaccount' with group 'testaccount'. The user's home directory is set to '/home/testaccount'. The user is then renamed to 'haversham' using 'sudo usermod -l haversham testaccount'. The user is killed with 'sudo pkill -u testaccount'. The user 'testaccount' is not found when 'ps -u testaccount' is run. The user's home directory is changed to '/home/haversham' with 'sudo usermod -d /home/haversham -m haversham'. Finally, 'ls /home' shows the new directory structure: 'haversham jack jacob jlwallen nathan olivia stierney'. A watermark for 'Quantrimang' is visible in the terminal.

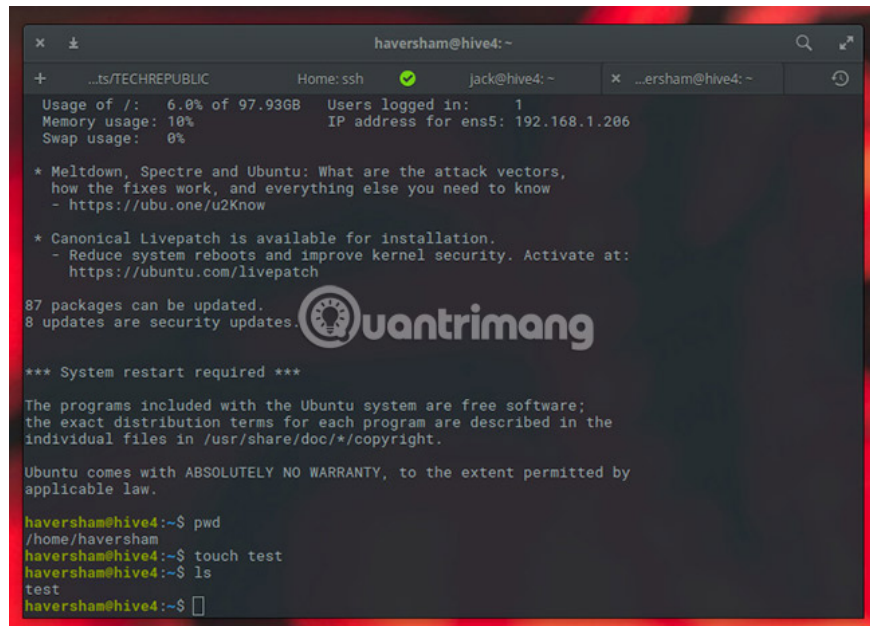
```
jack@hive4:~  
...jects/TECHREPUBLIC Home: ssh x jack@hive4:~  
Adding new user 'testaccount' (1007) with group 'testaccount' ...  
The home directory '/home/testaccount' already exists. Not copying from '/etc/skel'.  
Enter new UNIX password:  
Retype new UNIX password:  
passwd: password updated successfully  
Changing the user information for testaccount  
Enter the new value, or press ENTER for the default  
Full Name []: test account  
Room Number []:  
Work Phone []:  
Home Phone []:  
Other []:  
Is the information correct? [Y/n] y  
jack@hive4:~$ sudo usermod -l haversham testaccount  
jack@hive4:~$ sudo pkill -u testaccount  
pkill: invalid user name: testaccount  
jack@hive4:~$ ps -u testaccount  
error: user name does not exist  
  
Usage:  
ps [options]  
  
Try 'ps --help <simple|list|output|threads|misc|all>'  
or 'ps --help <s|l|o|t|m|a>'  
for additional help text.  
  
For more details see ps(1).  
jack@hive4:~$ sudo usermod -d /home/haversham -m haversham  
jack@hive4:~$ ls /home  
haversham jack jacob jlwallen nathan olivia stierney  
jack@hive4:~$
```

Finally, the user account group name must be changed from the **testaccount** to **haversham**. To do this, enter the command:

```
sudo groupmod -n haversham testaccount
```

## Check account

Before logging out, check your account by using SSH to log in to the server with a new username. Upon successful login, you will find yourself in **/home / haversham**. You can check this further by creating a test file (to ensure users have write privileges in their home directory). Try giving an order. If the error is not received (as shown below), all is fine and the administrator can transfer the renamed account to the user.

A terminal window titled 'haversham@hive4: ~' showing system status. It displays disk usage (6.0% of 97.93GB), memory usage (10%), and swap usage (0%). It lists users logged in (1) and IP address (192.168.1.206). There are system messages about Meltdown, Spectre, and Canonical Livepatch. A notification for 87 packages to be updated is shown with a 'Quantrimang' logo. The terminal shows the user running 'pwd' (output: /home/haversham), 'touch test', and 'ls' (output: test).

## Delete an account in Unix / Linux

The **userdel** command can be used to delete an existing personal account. This order is very dangerous if not used with caution.

Only one argument or option is available for the command: **-r** , to remove the account's home directory and mail.

For example, to remove the mcmohd20 account, you need to notify the following command:

```
$ userdel -r mcmohd20
```

If you want to keep the main directory for the following items, you do not use the **-r** option. You can remove the home directory the next time.

Alternatively, you can use the deluser command to delete an account in Linux, here is haversham:

```
sudo deluser -r haversham  
sudo delgroup haversham
```

The difference between userdel and deluser is: deluser and delgroup delete users, group from the system according to the command line options and configuration information in /etc/deluser.conf and /etc/adduser.conf. The front-end userdel and groupdel delete the main directory, and even all files that the user is deleted from the system

According to Tutorialspoint

Previous article: Basic file system in Unix / Linux

Next lesson: System performance in Unix / Linux

You finished reading the article "**User management in Unix / Linux**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.

