1. User Administration

User and Group management are one of the key role of System Administration. In this section I will discuss about basic commands to administer users/Group.

1.1 Creating user in Linux

1.1.1 Adding user using console based command

Basic Syntax:

useradd [options] <username>

Example 1 Adding user shiba

useradd shiba

Example 2 Adding user dipika with nologin as shell. We add this type of user usually for ftp or any other services where user login is not required.

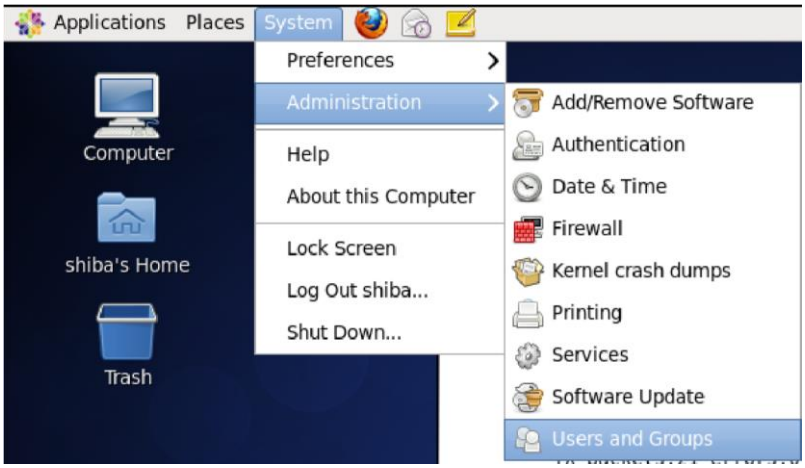
useradd -s /sbin/nologin dipika

Similarly, you can use other options to customize different parameters during user creation. Additional Options to customizing different parameters can be found in following table.

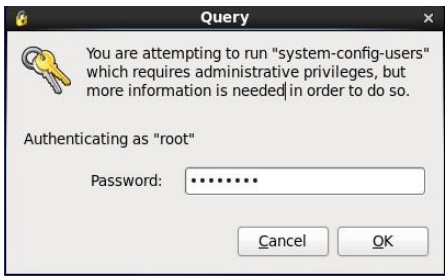
-c <comment>	Change the comment field. This field is often used for Full Name.
-d <home dir>	Change the home directory
-e <expire date>	Set date on which the account will expire and be disabled.
-g <group>	Change the initial login group (initial group have same name as of user.
-G <group,[…]>	A comma separated list of supplementary groups for the user.
-l <login name>	Change the login name
-s <shell>	Change the shell.
-u <uid>	Change the UID.
-L	Lock the password
-U	unlock the password.

1.1.2 Adding user using GUI based User Manager

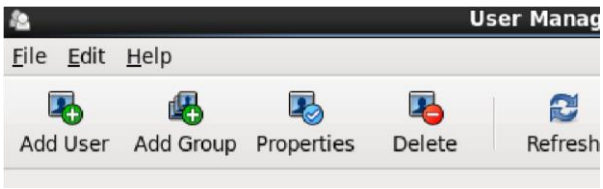
Step 1: Go to System→Administration→Users and Groups



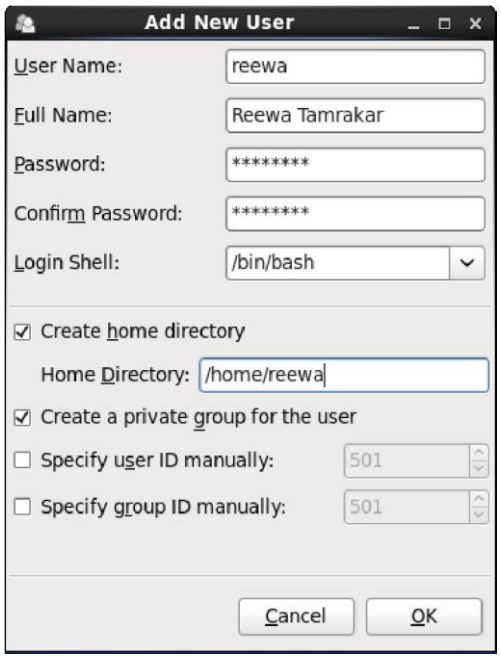
Step 2: If you are logged in to GUI with normal user, you will be asked for root password. Just provide root credential.



Step 3: Click to Add user Button.



Step 4: Fill all the details of the used to be added and click to ok.



Note: In above dialog box you will be able to choose different shell like /bin/bash, /bin/csh, /sbin/nologin and more. Similarly, you can modify home and group settings.

1.2 Modifying user's setting

1.2.1 Modify user's parameters using console based command

Basic Syntax:

```
usermod [options] <username>
```

Example 3 Changing home directory of user shiba. Default home directory of user shiba is /home/shiba. With following command you will be able to change home directory to /guests/shiba, where /guests/shiba directory should exist.

```
usermod -d /guests/shiba shiba
```

Example 4 Adding full name of use dipika and changing User ID (UID) to 5000. Note till Redhat Enterprise Linux 6, uid for normal users start from 500, however, in Redhat Enterprise Linux 7 uid for normal users start from 1000.

```
usermod -c "Dipika Parajuli" –u 5000 dipika
```

Example 5 Locking user. Locking existing user disables the particular user to login the system.

```
usermod -L dipika
```

Note: For all options see above table

1.2.2 Modify user's parameters using GUI User Manager

Select the user in list and click to properties button. In following figure, you can see that reewa user is selected and I clicked to Properties button to modify settings.

With this dialog box you will be able to change different parameters like password, home directory, Account info (expiry date, locking user), Password Info (password expiry details) and modify groups from Groups tab.

1.3 Deleting existing user

1.3.1 Deleting user using console based command

Basic Syntax:

```
userdel [options] <username>
```

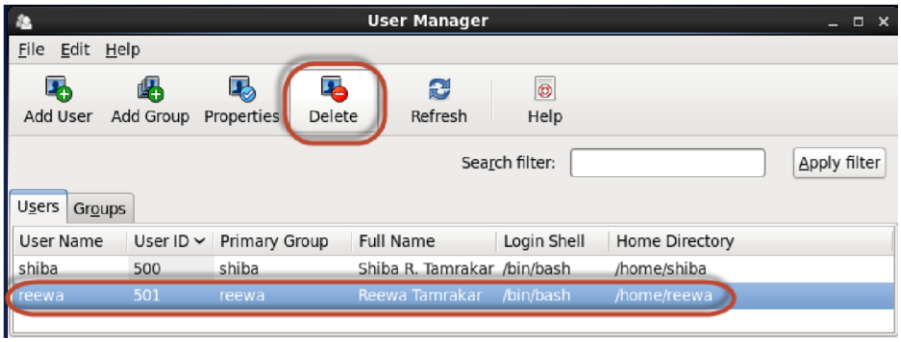
Example 6 Deleting user

```
userdel dipika
```

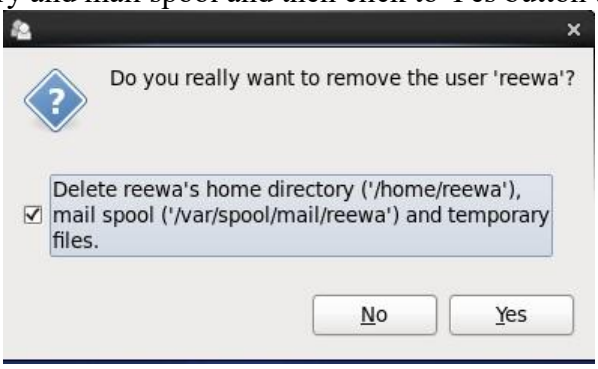
Note: use -r option to delete home directory of the user when user is deleted. Normally home directory of the user does not deleted when you delete the user. It is sometime important to save data of particular user even though the user is not required.

1.3.2 Deleting user using GUI User Manager

Step 1: Select the user you want to delete from list and click to delete button.



Step 2: Following Dialog box appears for conformation. You may select or deselect “delete reewa’s home directory and mail spool and then click to Yes button to conform.



1.4 Advanced User settings

1.4.1 Changing age of user (expire date)

Basic Syntax:

```
chage [options] <username>
```

Available Options

-m	minimum days between password changes
-M	maximum days between password changes
-I	number of days inactive since password expired before locking account
-E <date>	expires the account on this date (YYYY-MM-DD format)
-W	number of days before a required change to start warnings

Example 7 Setting user password age to 90 days

```
chage -M 90 shiba
```

Example 8 To set expiry date to 2015-02-12

```
chage -E 2015-02-12 shiba
```

Example 9 List details about password expiry date

```
chage -l shiba
```

Example 9 Changing warning message to 3 days

```
chage -W 3 shiba
```

1.4.2 Login Shell Scripts

/etc/profile

It is executed every time a user logs into the system containing environmental variables settings.

/etc/profile.d/*.sh

It contains initialization scripts specific to software packages installed by RPMs called by /etc/profile.

~/.bash_profile
It is a script which runs next which typically calls ~/.bashrc and /etc/bashrc. It contains system wide environment variable settings.

~/.bashrc
It allows users to customize their own aliases and functions without the intervention of the administrator. It runs whenever a user starts up a non-login interactive shell, and the default user.

~/.bash_profile
It is executed once at login time. It is usually used to set environment variables and to start programs at login.

1.4.3 Switching Accounts

su
To allow user to be another temporary. root is the default user. – option will reset environment setting for switching user based on his/her configuration.

Basic Syntax:

```
su [-] [user] -c command
```

Example 10 Switching to user dipika. Without resetting user environment.

```
su dipika
```

Example 11 Switching to user dipika. resetting user environment, all the environment setting for dipika will be configure on switching.

```
su - dipika
```

Example 12 Switching to user root. You don't have to provide username for root

```
su
```

1.4.4 Working with sudo

sudo command can be used to access special command available for particular user with his/her access privilege. For example, if you want to run command which is only available for root user, than you can use sudo to run the command with root privilege.

Assigning sudo permission to users

Run:
'visudo' or 'vi /etc/sudoers' in terminal and add following lines

```
%users  ALL=/sbin/mount /mnt/cdrom, /sbin/umount
```

In above example, all the members in Group users will be able to use /sbin/mount, /mnt/cdrom and /sbin/umount command.

1.4.5 SUID and SGID

SUID and/or SGID bits set on an executable file cause it to run under the user and/or group even though the file is run be another users. If Setgid (SGID) mod activate for a directory the files created in the directory will belong to same group of parent directory.

#chmod u+s <filename> (SUID)
#chmod g+s <filename> (GUID)

1.4.6 Sticky bit

By setting sticky bit only the owner of the file can erase the file but not the other group member even though the read/write/execute permission is provide to group member. #chomd o+t <directory>

2. File Permission

1. Available Options

Option:	referred to
u	owner of file or directory (user)
g	group of the file/directory
o	other
a	all

2. Permission Types:

permission	octal value	Meaning
Read (r)	4	The file is read only, for directory it's contain can be listed
Write (w)	2	File can be modified, for directory we can create, remove files/directory
Execute (e)	1	Files can be executed if it is a program file, For directory, change into directory (cd)

3. Permission operator

+	Add permission
-	remove permission
=	To assign (absolutely) permission

4. Illustration with example

```
ls -l
```

In the left part of output file/directory permission details will be listed as: _rwx r w _ r _ _

1st position indicates the type o file

Character	Description
-	Normal File

d	Directory
l	Link file
b	Block device file
c	Character device file

2nd, 3rd and 4th positon indicates the permission for user

5th, 6th and 7th position indicates the permission for group 8th,

9th and 10th position indicates the permission for other

Example 13: File permission when new file is created.

1.4.6.1 -rw-r--r--: meaning that, the file has read/write permission for user, and read permission for group and others.

5. Default file permission

umask: It is used to set default permission on file/directory on its creation. Maximum allowed permission on file (666) and Maximum allowed permission on directory is (777)

Example 14: Set the value of umask such that perm siion on a file during its creation give read/write access to the owner, read permission to group and no permission to other.

```
umask 026
```

Note: Formula to calculate umask:
For directory: subtract the file permission value from 777.
For file: subtract the file permission value from 666.

In above example: file permission for owner is read(4)/write(2), sum value is (4+2=6), file permission for group is read (4), and permission for other is 0

666
-026
640

The default value is set in */etc/bashrc* file.

6. Changing permission of created files/directories *Basic Syntax:*

```
chmod [option] [mode/permission] <file/directory>
```

Example 15: Command to assign file permission as under: for owner: full, (read/write/execute), for group: (read/write) and for other: (read only)

```
chmod u=rwx,g=rw,o=r myfile.txt
```

or

```
chmod 764 myfile.txt
```

Example 16: Command to assign file permission as under: for owner: full, for group: read/execute, for other: none.

```
chmod u=rwx,g=rx,o= myfile.txt
```

or

```
chmod 750 myfile.txt
```

Example 17: Write a command to assign file permission as under: for owner: read, for group : none, for other:none

```
chmod u=r,g-rwx,o-rwx,o-rwx myfile.txt
```

or

```
chmod 400 myfile.txt
```

Again, Add execution permission for all:

```
chmod a+x myfile.txt
```

Example 18: Write a command to assign d irectory permission (also to it's content) as under: for owner: full, for group: none, for other: none.

```
chmod -R 700 mydirectory
```

3. Group Administration

1. To add group

Basic Syntax:

```
gourpadd [option] <groupname>
```

Example 19 You are required to create sales, hr and web groups for Sales HR and IT departments. Each department for which you create a group also needs a shared directory. This will allow users in each department to share files, but will prevent users in other departments from altering, or even seeing those files. Similarly the file which a particular user creates in a shared directory should be deleted by the owner of the file.

Adding groups

```
groupadd sales
groupadd hr
groupadd web
```

Adding users in each group

```
#Adding user ram with additional group membership in sales
useradd –G sales ram
#Setting password for user ram passed
ram

#Adding user sita with additional group membership in sales
useradd –G sales sita passed sita

#Adding user hari with additional group membership in hr
useradd –G hr hari passed hari

#Adding user gita with additional group membership in sales
useradd –G sales gita passed gita

useradd –G sales waza passed
waza

useradd –G sales rani
passed rani

#Adding user gita with additional group membership in sales, web and hr
useradd –G hr,web,sales manager
passwd manager
```

Create depts. Directory and its sub directories salesdir, hrdir, webdir

```
mkdir –p /depts/{salesdir,hrdir,webdir}
```

Change the permission to 775, all to user, read and execute to group/other

```
chgrp sales /depts./sales
chmod 755 /depts.
```

Change the file permission to all access to user/group, and no access to other

```
chmod 770 /depts./salesdir
chmod 770 /depts./hrdir
chmod 770 /depts./webdir
```

Set GID bit on in each departmental directories so that the files group is same as that of parent directory.

```
chmod g+s /depts/*
```

Set Sticky bit, so that only owner can delete the file.

```
chmod o+t /depts/*
```