

Becoming the superuser for a short while

It is often useful to become the superuser to perform important system administration tasks, but as you have been warned (and not just by me!), you should not stay logged on as the superuser. In most distributions, there is a program that can give you temporary access to the superuser's privileges. This program is called su(short for substitute user) and can be used in those cases when you need to be the superuser for a small number of tasks. To become the superuser, simply type the su command. You will be prompted for the superuser's password:

```
[me@linuxbox me]
```

```
$ su Password:
```

```
[root@linuxbox me]#
```

After executing the sucommand, you have a new shell session as the superuser. To exit the superuser session, type exit and you will return to your previous session.

In some distributions, most notably Ubuntu, an alternate method is used. Rather than using su, these systems employ the sudo command instead. With sudo, one or more users are granted superuser privileges on an as needed basis. To execute a command as the superuser, the desired command is simply preceded with thesudo command. After the command is entered, the user is prompted for the user's password rather than the superuser's:

```
[me@linuxbox me]
```

```
$ sudo some_command Password:
```

```
[me@linuxbox me]$
```

Changing file ownership

You can change the owner of a file by using thechown command. Here's an example: Suppose I wanted to change the owner ofsome_file from "me" to "you". I could:

```
[me@linuxbox me]$ su
```

```
Password: [root@linuxbox me]#chownyou some_file
```

```
[root@linuxbox me]# exit [me@linuxbox me]$
```

Notice that in order to change the owner of a file, you must be the superuser. To do this, our example employed the sucommand, then we executed chown, and finally we typed exit to return to our previous session.

chown works the same way on directories as it does on files.

Changing group ownership

The group ownership of a file or directory may be changed with chgrp. This command is used like this: [me@linuxbox me]\$ chgrp new_group some_file In the example above, we changed the group ownership of some_file from its previous group to "new_group". You must be the owner of the file or directory to perform a chgrp.

chown Command:

chown command is used to change the owner / user of the file or directory. This is an admin command, root user only can change the owner of a file or directory.

SYNTAX:

The Syntax is chown [options] newowner filename/directoryname OPTIONS:

- ❖ -R Change the permission on files that are in the subdirectories of the directory that you are currently in.
- ❖ -c Change the permission for each file.
- ❖ -f Prevents chown from displaying error messages when it is unable to change the ownership of a file.

EXAMPLE:

1. chown hiox test.txt The owner of the 'test.txt' file is root, Change to new user hiox.
2. chown -R hiox test The owner of the 'test' directory is root, With -R option the files and subdirectories user also gets changed.
3. chown -c hiox calc.txt Here Change The Owner For The Specific 'Calc.Txt' File Only.
chmod Command: chmod command allows you to alter / Change access rights to files and directories.

File Permission is given for users,group and others as,

	Read	Write	Execute
User	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Group	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Others	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Permission	<input type="text" value="000"/>		
Symbolic Mode	<input type="text" value="___"/>		

SYNTAX:

The Syntax is chmod [options] [MODE] FileName.

File Permission

- ❖ # File Permission
- ❖ 0 none

- ❖ 1 execute only
- ❖ 2 write only
- ❖ 3 write and execute
- ❖ 4 read only
- ❖ 5 read and execute
- ❖ 6 read and write
- ❖ 7 set all permissions

OPTIONS:

- ❖ -c Displays names of only those files whose permissions are being changed
- ❖ -f Suppress most error messages
- ❖ -R Change files and directories recursively
- ❖ -v Output version information and exit.

EXAMPLE:

1. To view your files with what permission they are: `ls -alt` This command is used to view your files with what permission they are.
2. To make a file readable and writable by the group and others. `chmod 066 file1.txt`
3. To allow everyone to read, write, and execute the file `chmod 777 file1.txt` with friend ship | with friend ship.

chgrp Command:

chgrp command is used to change the group of the file or directory. This is an admin command. Root user only can change the group of the file or directory.

SYNTAX:

The Syntax is `chgrp [options] newgroup filename/directoryname`

OPTIONS:

- ❖ -R Change the permission on files that are in the subdirectories of the directory that you are currently in.
- ❖ -c Change the permission for each file.
- ❖ -f Force. Do not report errors.