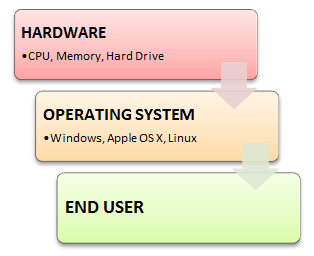
# Introduction

## What is an Operating System?

Every time you switch on your computer, you see a screen where you can perform different activities like write, browse the internet or watch a video. What is it that makes the computer hardware work like that? **How does the processor on your computer know that you are asking it to run a mp3 file?**

**Well, it is the** **operating system or the kernel** **which does this work**. A kernel is the program at the heart of any operating system that takes care of fundamental stuff, like letting hardware communicate with software

**So, to work** **on your computer you need an Operating System(OS)** .In fact,  you are using one as you read this on your computer. Now, you may have used popular OS's like Windows**, Apple OS X** but here we will learn what Linux is and what benefits it offers over other OS choices.

[](https://cdn.guru99.com/images/OperatingSystemBasics.png)

## What is Linux? Who created Linux?

Linux is an operating system or a kernel which germinated as an idea in the mind of young and bright **Linus Torvalds** when he was a computer science student. He used to work on the **UNIX OS (a proprietary software),** and thought that it needed improvements.

However, when his suggestions were rejected by the designers of UNIX, he thought of launching an OS which will be **receptive to changes, modifications suggested by its users**.

## The Lone Kernel & the early days

So **Linus devised a Kernel** named Linux in 1991. Though he would need programs like File Manager, Document Editors, Audio -Video programs to run on it. Something as you have a cone but no ice-cream on top J.

As time passed by, he  collaborated with other **programmers in places like MIT** and applications for Linux started to appear. So around 1991, a working Linux operating system with some applications was officially launched and this was the start of one of the **most loved and open-source OS options available today**.

The earlier versions of Linux were not so user friendly as they were in use by computer programmers and **Linus Torvalds never had it in mind to commercialize** his product.

This definitely curbed the Linux's popularity as other commercially oriented Operating System Windows got famous. Nonetheless, the open-source aspect of the Linux operating system made it more robust.

## Linux gets its due attention

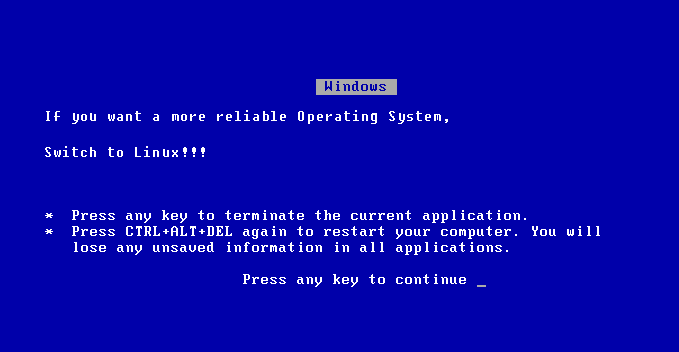
[](https://cdn.guru99.com/images/OurTimeIsNow.png)

The main advantage of Linux was that programmers were able to use the Linux Kernel in order to design their own custom operating systems. With time, a new range of user-friendly OS's stormed the computer world. Now, **Linux is one of most popular and widely used Kernel,** and it is the backbone of popular operating systems like **Debian, Knoppix, Ubuntu, and Fedora**. Nevertheless, the list does not end here as there are thousands of OS's based on Linux which offer a variety of functions to the users.

## The benefits of using Linux

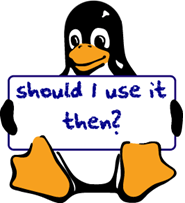
Linux now enjoys popularity at its prime and it's famous among programmers as well as regular computer users around the world. Its main benefits are -

It offers a **free operating system**. You do not have to shell hundreds of dollars to get the OS like Windows!

[](https://cdn.guru99.com/images/Linux_Windows_Error.png)

* Being open-source, anyone with programming knowledge can modify it.
* The Linux operating systems now offer **millions of programs/applications to choose from**, most of them free!
* Once you have Linux installed you no longer need an antivirus! Linux is a highly secure system. More so , there is a global development community constantly looking at ways to enhance its security. With each upgrade, the OS becomes more secure and robust
* Linux is the OS of choice for Server environments due to its stability and reliability (Mega-companies like Amazon, Facebook ,and Google use Linux for their Servers). A Linux based server could run non-stop without a reboot for years on end.

## Is it for me?

[](https://cdn.guru99.com/images/Should_I_Use_IT.png)

Users, who are new to Linux, usually shun it by falsely considering it as a difficult and technical OS to operate but, to state the truth, in the last few years Linux operating systems have become a lot more user-friendly than their counterparts like Windows**,** so trying them is the best way to know whether Linux suits you or not.

There are **thousands of Linux based operating systems;** most of them offer **state-of-the-art security and applications**, **all of it for free!**

This is what Linux is all about, and now we will move on to how to install Linux and which Distribution you should choose.

# Selecting a Linux Distribution and Types of Installation

Now that we know what Linux is, it is time that to learn how we should install it on the computer and choose which Distribution we should use. Let us start by understanding what a Linux Distribution is.

## What is a Linux Distribution?

Well, now as you know that **Linux is an open-source, free to use kernel** .It is used by programmers, organizations, profit and non-profit companies around the world in order to **create Operating systems to suit their individual requirements**.

To prevent hacking attempts, many organizations keep their Linux operating systems private.

Many others, make their variations of Linux available publicly so the whole world can benefit at large.

These versions/ types /kinds of **Linux operating system are called Distributions**.

**How many distributions are out there?**

[](https://cdn.guru99.com/images/BestLinuxDistro.png)

There are **hundreds of Linux operating systems or Distributions** available these days. Many of them are designed with a specific purpose in mind. For example, to run a **web server or to run on network switches like routers, modems** etc.

The latest example of one of the most popular smartphone based **Linux Distribution is Android!**

Many of these Distributions are built to offer**excellent personal computing**.

Here, are a few popular Linux Distributions (also called Linux Distro) -

|  |  |  |
| --- | --- | --- |
| **Linux Distribution** | **Name** | **Description** |
| [Selecting a Linux Distribution and Types of Installation](https://cdn.guru99.com/images/ArchLinux.png) | **Arch** | This Linux Distro is popular amongst Developers. It is an independently developed system .It is designed for users who go for a do-it-yourself approach. |
| [Selecting a Linux Distribution and Types of Installation](https://cdn.guru99.com/images/centos.png) | **CentOS** | It is one of the most used Linux Distribution for enterprise and web servers. It is a free enterprise class Operating system and is based heavily on Red Hat enterprise Distro. |
| [Selecting a Linux Distribution and Types of Installation](https://cdn.guru99.com/images/debian.png) | **Debian** | Debian is a stable and popular non-commercial Linux distribution. It is widely used as a desktop Linux Distro and is user-oriented. It strictly acts within the Linux protocols. |
| [Selecting a Linux Distribution and Types of Installation](https://cdn.guru99.com/images/fedora.png) | **Fedora** | Another Linux kernel based Distro, Fedora is supported by the Fedora project, an endeavor by Red Hat. It is popular among desktop users. Its versions are known for their short life cycle. |
| [Selecting a Linux Distribution and Types of Installation](https://cdn.guru99.com/images/gentoo.png) | **Gentoo** | It is a source based Distribution which means that you need to configure the code on your system before you can install it. It is not for Linux beginners, but it is sure fun for experienced users. |
| [Selecting a Linux Distribution and Types of Installation](https://cdn.guru99.com/images/LinuxMint.png) | **LinuxMint** | It is one of the most popular Desktop Distributions available out there. It launched in 2006 and is now considered to be the fourth most used Operating system in the computing world. |
| [Selecting a Linux Distribution and Types of Installation](https://cdn.guru99.com/images/OpenSuse.png) | **OpenSUSE** | It is an easy to use and a good alternative to MS windows. It can be easily set up and can also run on small computers with obsolete configurations. |
| [Selecting a Linux Distribution and Types of Installation](https://cdn.guru99.com/images/redhat%282%29.png) | **RedHat enterprise** | Another popular enterprise based Linux Distribution is Red Hat Enterprise.It has evolved from Red Hat Linux which was discontinued in 2004. It is a commercial Distro and very popular among its clientele. |
| [Selecting a Linux Distribution and Types of Installation](https://cdn.guru99.com/images/slackware.png) | **Slackware** | Slackware is one of the oldest Linux kernel based OS's. It is another easy desktop Distribution. It aims at being a 'Unix like' OS with minimal changes to its kernel. |
| [Selecting a Linux Distribution and Types of Installation](https://cdn.guru99.com/images/ubuntu_logo.png) | **Ubuntu** | This is the third most popular desktop operating system after Microsoft Windows and Apple Mac OS. It is based on the Debian Linux Distribution and it is known for its desktop environment. |

**The Best Linux Distribution !**

The term best is **relative. Each Linux distribution is built for a specific purpose built to meet the demands of its target users.**

The desktop Distributions are **available for free** at their respective websites. You might want to try them one by one till you get to know which Distribution you like the most. Each one of them offers it's own unique design, **applications** and **security**.

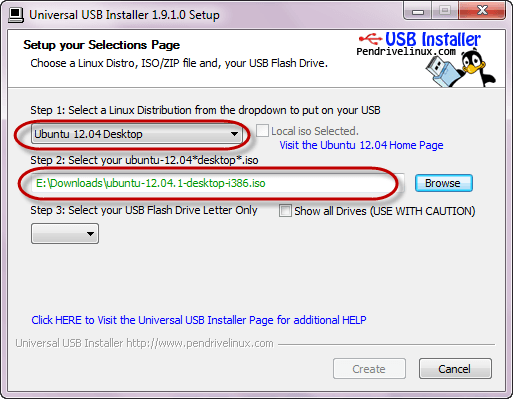
We will be using Ubuntu for our learning purpose as it's easy for a beginner to understand.

**Installing Linux**

Let's look the various methods we can use to install Ubuntu.

**USB stick**

This is one of the easiest methods of installing Ubuntu or any distribution on your computer. Follow the steps.

[](https://cdn.guru99.com/images/Linux.png)

* Download the .iso or the OS files on your computer from this link .
* Download free software like 'Universal USB installer'to make a bootable USB stick.
* Boot your computer through it and follow the instructions as they come.

**Live CD**

Those who like the way a CD runs should try using this method.

[](https://cdn.guru99.com/images/LiveCD.png)

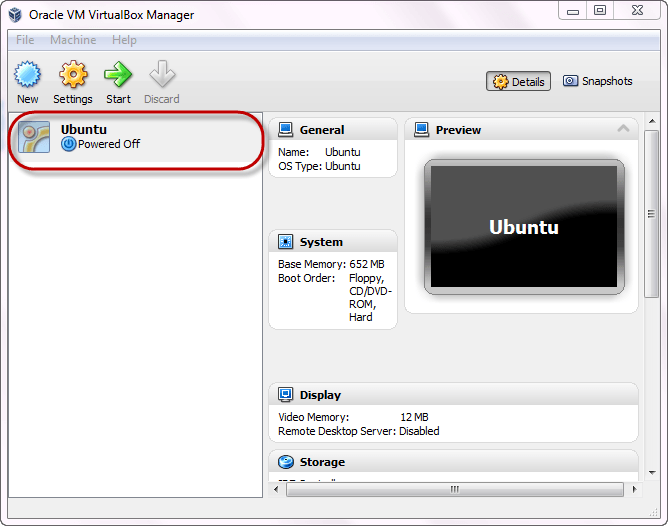
* Download the .iso or the OS files onto  your computer from this link http://www.ubuntu.com/download/desktop.
* Burn the files on a CD .
* Boot your computer through the optical drive and follow the instructions as they come.

**Virtual Installation**

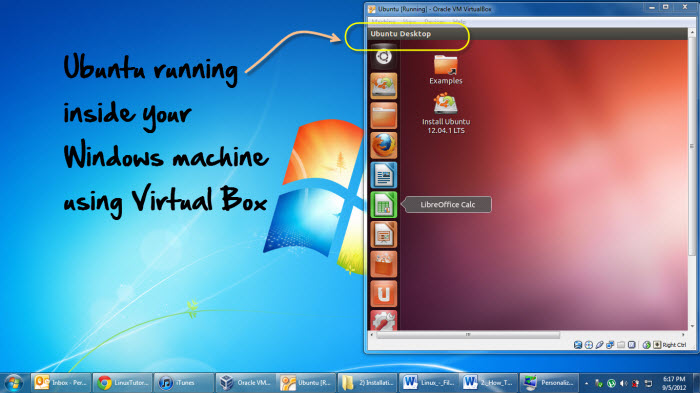
This is a popular method to install a Linux operating system. Virtual installation offers you the freedom of running Linux on an existing OS already installed on your computer. This means if you have Windows running, then you can just run Linux with a click of a button.

Virtual machine software like Oracle VM can install Ubuntu in easy steps. Let us look at them.

* Download and install Oracle VM on your computer.
* Click on new on the VM tool bar. Enter the VN Name and OS Type.
* Follow the instructions. You will see Ubuntu configured in your VM.

[](https://cdn.guru99.com/images/Ubuntu.png)

Double click on it . Ubuntu will load as a separate machine while you are working on Windows

[](https://cdn.guru99.com/images/Ubuntu_VM.jpg)

**Testing a Distribution**

Those who want to test a distribution before installing it on a computer and replacing the existing OS would be surprised to know that you can easily run it from the USB stick/CD . Then you can easily check out the interface. This way you can learn whether you like the distribution or not and can either install or move on to another one.

You can even install Linux side by side with Windows or any other OS (dual booting). So you can also go for a system which has two of your favorite operating systems installed on it.

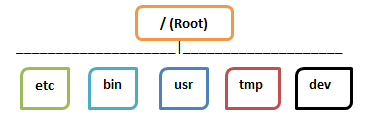
**Summary**

* An operating system based on the Linux kernel is called a Distribution or Distro
* There are hundreds of Distributions available, some of which are designed to accomplish a sole purpose like running servers, act as network switches etc.
* Naming the best Linux Distribution is difficult as they are made for different.
* Linux can be installed in your system via the below mentioned methods:
* USB stick
* Live CD
* Virtual Installation

**Root directory**

This root directory can be considered as the start of the file system and it further branches out various other subdirectories. The root is denoted with a forward slash '/'.

A general tree file system on your UNIX may look like this.

[](https://cdn.guru99.com/images/FolderStructure.png)

## Types of Files

In Linux and UNIX, everything is a file. Directories are files, files are files, and devices like Printer, mouse, keyboard etc.are files.

Let's look into the File types in more detail '

## ****General Files****

General Files also called as Ordinary files. They can contain image, video, program or simply text. They can be in ASCII or a Binary format. These are the most commonly  used files by Linux Users

## ****Directory Files****

These files are a warehouse for other file types. You can have a directory file within a directory (sub-directory).You can take them as   'Folders' found in Windows operating system.

## ****Device Files:****

In MS Windows, devices like Printers, CD-ROM and hard drives are represented as drive letters like G: H:. In Linux, there are represented as files.   For example, if the first SATA hard drive had three primary partitions, they would be named and numbered as /dev/sda1, /dev/sda2 and /dev/sda3.

**Note**: All device files reside in the directory /dev/

All the above file types (including devices) have permissions, which allow a user to read, edit or execute (run) them. This is a powerful Linux/Unix feature. Access restrictions can be applied for different kinds of users, by changing permissions.

## Users in Linux

There are 3 types of users in Linux

1. Regular
2. Administrative(root)
3. Service

## Regular User

A regular user account is created for you when you install Ubuntu on your system. All your files and folders are stored in /home/ which is your home directory. As a regular user, you do not have access to directories of other users.

## Root User

Other than your regular account another user account called root is created at the time of installation.  The root account is a **super user** who can access restricted files, install software and has administrative privileges. Whenever you want to install software, make changes to system files or perform any administrative task on Linux; you need to log in as a root user. Otherwise, for general tasks like playing music and browsing the internet you can use your regular account.

## Service user

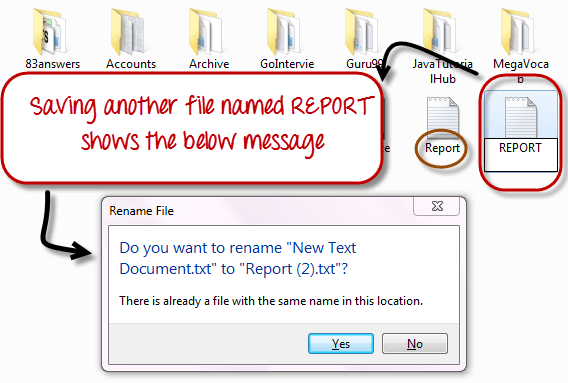
Linux is widely used as a Server Operating System. Services such as Apache, Squid, mail, etc. have their own individual service accounts.  Having service accounts increases security of your computer. Linux can allow or deny access to various resources depending on the service.

Note:

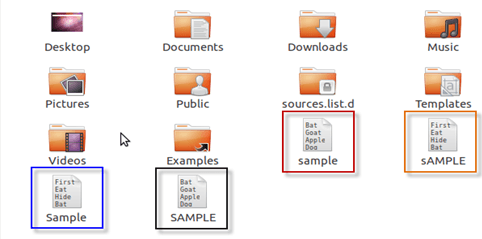
1. You will not see service accounts in Ubuntu Desktop version.
2. Regular accounts are called standard accounts in Ubuntu Desktop

## File Name Convention

 In Windows, you cannot have 2 files with the same name in the same folder. See below -

[](https://cdn.guru99.com/images/WindowsError.png)

While in Linux, you can have 2 files with the same name in the same directory, provided they use different cases.

[](https://cdn.guru99.com/images/LinuxFileNaming.png)

## The HOME Directory

For every user in Linux, a directory is created as **/home/**

Consider, a regular user account "Tom". He can store his personal files and directories in the directory "/home/tom". He can't save files outside his user directory and does not have access to directories of other users. For instance, he cannot access directory "/home/jerry" of another user account"Jerry".

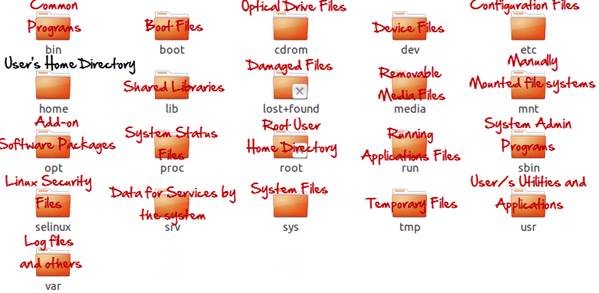
The concept is the similar to C:\Documents and Settings in Windows.

When you boot the Linux operating system, your user directory (from the above example /home/tom) is the **default working directory**. Hence the directory "/home/tom is also called the **Home directory** which is a misnomer.

The working directory can be changed using some commands which we will learn later.

## ****Other Directories****

In Windows, System and Program files are usually saved in C: drive. But, in Linux you would find the system and program files in different directories. For example, the boot files are stored in the /boot directory and program and software files can be found under /bin , device files in /dev. Below are important Linux Directories and short description of what they contain.

[](https://cdn.guru99.com/images/LinuxDirectories.png)

These are most striking differences between Linux and other Operating Systems.  There are more variations you will observe when switching to Linux and we will discuss them as we move along in our tutorials.

## Summary

* Unix/Linux uses a tree like hierarchical file system.
* There are no drives in Linux , unlike windows
* Peripherals like  hard drives , cd rom , printers are also considered files in Linux/Unix
* There are 3 types of user account types 1) Regular , 2) Root and 3) Service Account
* Root user is the super user and has all administrative privileges
* Linux file naming convention is case sensitive. Thus , sample and SAMPLE are 2 different files in Linux/Unix operating system
* For every user /home/username directory is created which is called his home directory.

# Terminal V/s File Manager & The CD command

The most frequent tasks that you perform on your PC is creating, moving or deleting Files. Let's look at various options for File Management.

To manage your files , you  can either use

1. Terminal (Command Line Interface - CLI)
2. File manager (Graphical User Interface -GUI)

In the course, we will focus on the CLI , which brings us to out next question

**Why learn Command Line Interface ?**

Even though the world is moving to GUI based systems, CLI has its specific uses and is widely used in scripting and server administration. Let's look at it some compelling uses -

* Comparatively , Commands offer more options & are  flexible .Piping and stdin/stdout are immensely powerful are not available in GUI
* Some configurations in GUI are up to 5 screens deep while in a CLI  it's just a single command
* Moving, renaming 1000's of file in GUI will be time consuming (Using Control /Shift to select multiple files) ,while in CLI  , using regular expressions so can do the same task in a single command.
* CLI load fast and do not consume RAM compared to GUI. In crunch scenarios this matters.

**Both GUI and CLI have their specific uses.** For example, in **GUI , performance monitoring graphs** give **instant visual feedback** on system health , while seeing hundreds of lines of logs in CLI is an eyesore.

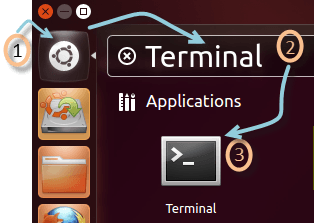
You must learn to use both GUI(File Manager) and CLI (Terminal)

GUI of a Linux based OS is similar to any other OS. Hence, we will focus on CLI and learn some useful commands.

## Launching the CLI on Ubuntu

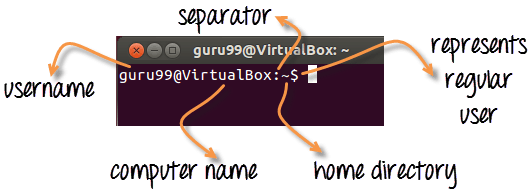
There are 2 ways to launch the terminal

1) Go to the Dash and type terminal

[](https://cdn.guru99.com/images/Terminal.png)

2) Or you can press **CTRL + Alt + T** to launch the Terminal

Once you launch the CLI (Terminal), you would find something as guru99@VirtualBox(see image) written on it.

[](https://cdn.guru99.com/images/Terminal_Description.png)

1) The first part of this line is the name of the **user** (bob, tom, ubuntu, home...)

2) The second part is the computer name or the host name. The hostname helps identify a computer over the network. In a server environment host-name becomes important.

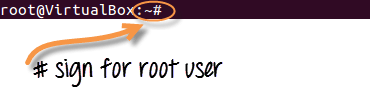
3) The **':'** is a simple separator

4) The tilde '~' sign shows that the user in working in the **home directory**. If you change the directory this sign will vanish

[Terminal V/s File Manager &  The CD command](https://cdn.guru99.com/images/Home-Directory.png)

In the above illustration we have moved from the /home directory to /bin using the **'cd'command**. The ~ sign does not display while working in /bin directory .It appears while moving back to the home directory.

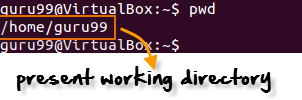
5) The '$' sign suggests that you are working as a regular user in Linux.  While working as a root user ,  '#' is displayed

[](https://cdn.guru99.com/images/root_user.png)

## Present working Directory

The directory that you are currently browsing is called the Present working directory. You log on to the home directory when you boot your PC .  If you want to determine the directory you are presently working on , use the command -

***pwd***

[](https://cdn.guru99.com/images/pwd.png)

pwd command stands for **p**rint **w**orking **d**irectory

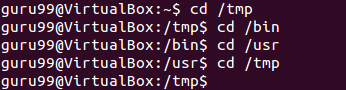
Above figure  shows that /home/guru99 is the directory we are currently working on.

## Changing Directories

If you want to change your current directory use the '**cd**' command .

**cd**

Consider the following example

[](https://cdn.guru99.com/images/cd.png)

Here, we moved from directory /tmp to /bin to /usr and then back to /tmp.

## Navigating to home directory

If you want to navigate to the home directory then type **cd**.

[Terminal V/s File Manager &  The CD command](https://cdn.guru99.com/images/cd%281%29.png)

You can also use the **cd ~** command

[Terminal V/s File Manager &  The CD command](https://cdn.guru99.com/images/cd_.png)

## Moving to root directory

The root of file system in Linux is denoted by '/'. Similar to  'c:\' in Windows.

Note: In Windows you use backward slash "\" while in UNIX/Linux , forward slash is use "/"

Type 'cd /' to move to the root directory.

[Terminal V/s File Manager &  The CD command](https://cdn.guru99.com/images/cd_root.png)

**TIP**: Do not forget space between **cd** and **/**. Otherwise you will get an error.

## Navigating through multiple directories

You can navigate through multiple directories at the same time by specifying its complete path.

Example:  If you want  to move the /cpu directory under /dev, we do not need to break this operation in two parts.

Instead, we can type '/dev/cpu' to reach the directory directly.

[Terminal V/s File Manager &  The CD command](https://cdn.guru99.com/images/navigating_multiple.png)

## Moving up one directory level

For navigating up one directory level, try **'cd ..'**.

[Terminal V/s File Manager &  The CD command](https://cdn.guru99.com/images/navigation_up_one_directory.png)

Here by using the 'cd ..' command, we have moved up one directory from '/dev/cpu' to '/dev'.

Then by again using the same command, we have jumped from '/dev' to '/' root directory.

## Relative and Absolute Paths

A path in computing is the address of a file or folder.

Example -

**C:\documentsandsettings\user\downloads**   in Windows or

**/home/user/downloads** in Linux

There are two kinds of paths:

## Absolute Path:

Let's say you have to browse the images stored in the Pictures directory of the home folder 'guru99'.

The absolute file path of Pictures directory **/home/guru99/Pictures**

To navigate to this directory, you can use the command**cd /home/guru99/Pictures'**

[Terminal V/s File Manager &  The CD command](https://cdn.guru99.com/images/absolute_path.png)

This is called absolute path as you are specifying the full path to reach the file.

## Relative Path:

Relative path comes in handy when you have to browse another subdirectory within a given directory.

It saves you from the effort to type complete paths all the time.

Suppose you are currently in your Home directory. You want to navigate to the Downloads directory.

You do no need to type the absolute path c**d /home/guru99/Downloads'.**

[Terminal V/s File Manager &  The CD command](https://cdn.guru99.com/images/relative_path.png)

Instead, you can simply type **'cd Downloads'** and you would navigate to the Downloads directory as you are already present within the **'/home/guru99'** directory.

[Terminal V/s File Manager &  The CD command](https://cdn.guru99.com/images/relative_path_1.png)

This way you do not have to specify the complete path to reach a specific location within the same directory in the file system.

**Summary:**

* To manage your files, you can use either the GUI(File manager) or the CLI(Terminal) in Linux. Both have its relative advantages. In the tutorial series we will focus on the CLI aka the Terminal
* You can launch the terminal from the dashboard or using the shortcut key **Cntrl + Alt + T**
* The pwd command gives the present working directory.
* You can use the cd command to change directories
* Absolute path is complete address of a file or directory
* Relative path is relative location of a file of directory with respect to current directory
* Relative path help avoid typing complete paths all the time.

|  |  |
| --- | --- |
| * Command | Description |
| cd or cd ~ | Navigate to HOME directory |
| cd .. | Move one level up |
| cd | To change to a particular directory |
| cd / | Move to the root directory |

# Must Know Linux/Unix Commands

File Management becomes easy if you know the right commands.

Sometimes, commands are also referred as "programs" since whenever you run a command, actually, it's the corresponding program code, written for the command, which is being executed.

## Listing files (ls)

If you want to see the list of files on your UNIX or Linux system, use the '**ls'** command.

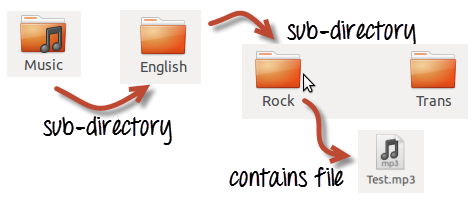
It shows the files /directories in your current directory.

[](https://cdn.guru99.com/images/ls%281%29.png)

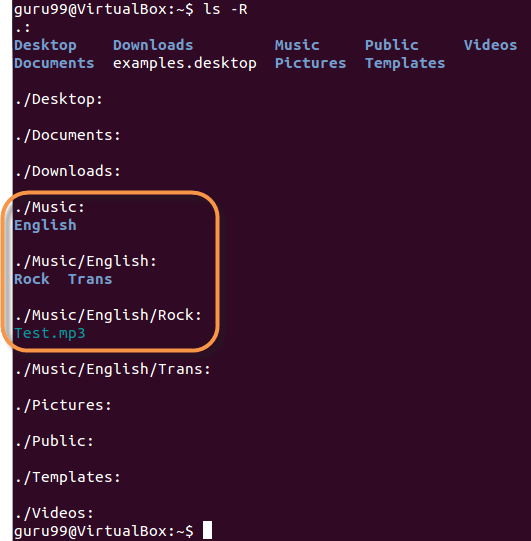
Note:

* Directories are denoted in blue color.
* Files are denoted in white.
* You will find similar color schemes in different flavors of Linux.

Suppose, your "Music" folder has following sub-directories and files.

[](https://cdn.guru99.com/images/sub-directory%281%29.png)

You can use **'ls-R' to shows all the files not only in directories but also subdirectories**

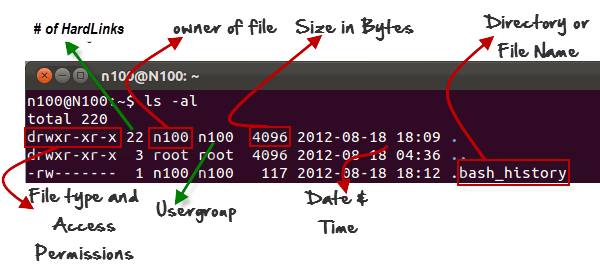
[](https://cdn.guru99.com/images/ls-R%281%29.png)

NOTE: The command is case-sensitive. If you enter, "**ls - r**" you will get an error

**'ls -al'** gives detailed information of the files. The command provides information in a columnar format. The columns provide the following information:

|  |  |
| --- | --- |
| **1st Column** | **File type and access permissions** |
| **2nd Column** | # of HardLinks to the File |
| **3rd Column** | Owner and the creator of the file |
| **4th Column** | Group of the owner |
| **5th Column** | File size in Bytes |
| **6th Column** | Date and Time |
| **7th Column** | Directory or File name |

Let's see an example -

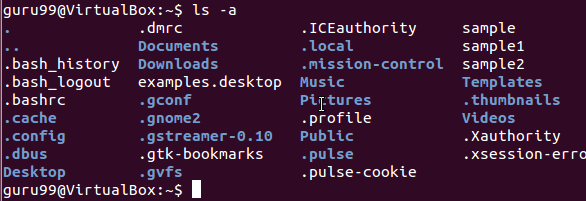
[](https://cdn.guru99.com/images/ls-al%282%29.png)

## Listing Hidden Files

Hidden items in UNIX/Linux begin with -[Must Know Linux/Unix Commands](https://cdn.guru99.com/images/period_symbol%282%29.png)at the start, of the file or directory.

Any Directory/file starting with a '.' will not be seen unless you request for it.  To view hidden files, use the command

***ls  - a***

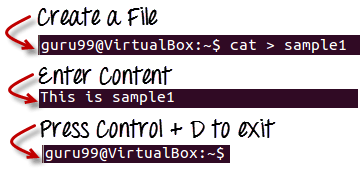
[](https://cdn.guru99.com/images/ls-a%282%29.png)

## Creating & Viewing Files

The 'cat' command is used to display text files. It can also be used for copying, combining and creating new text files.  Let's see how it works

To create a new file, use the command

1. cat > filename
2. Add content
3. Press 'ctrl + d' to return to command prompt.

[](https://cdn.guru99.com/images/cat_filename%281%29.png)

To view a file, use the command -

cat

Let's see the file we just created -

[Must Know Linux/Unix Commands](https://cdn.guru99.com/images/cat_view_a_file%281%29.png)

Let's see another file sample2

[Must Know Linux/Unix Commands](https://cdn.guru99.com/images/cat_sample2.png)

The syntax to combine 2 files is -

***cat file1 file2 > newfilename***

Let's combine sample 1 and sample 2.

[Must Know Linux/Unix Commands](https://cdn.guru99.com/images/cat_combine.png)

As soon as you insert this command and hit enter, the files are concatenated, but you do not see a result. This is because **Bash Shell (Terminal) is silent type**.  It will never give you a confirmation message like "OK" or "Command Successfully Executed". It will only show a message when something goes wrong or when an error has occurred.

In order to view the new combo file "sample" use the command

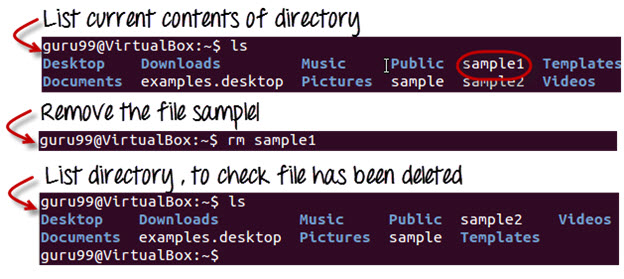
***cat sample***

[Must Know Linux/Unix Commands](https://cdn.guru99.com/images/cat_combo.png)

## Deleting Files

The 'rm' command removes files from the system without confirmation.  To delete a file use syntax -

***rm***

[](https://cdn.guru99.com/images/linux_rm_command.jpg)

## Moving and Re-naming files

In order to move a file, use the command

***mv***

Suppose we want to move the file "sample2" to location /home/guru99/Documents. Executing the command

***mv sample2  /home/guru99/Documents***

[Must Know Linux/Unix Commands](https://cdn.guru99.com/images/mv_error.png)

mv command needs super user permission. Currently, we are executing the command as a standard user. Hence we get the above error. To overcome the error use command

***sudo***

Sudo program allows regular users to run programs with the security privileges of the superuser or root.

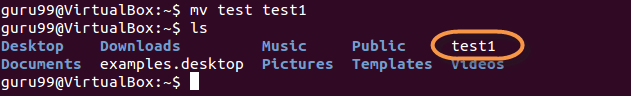
Sudo command will ask for password authentication. Though, you do not need to know the root password. You can supply your own password. After authentication, the system will invoke the requested command.

Sudo maintains a log of each command run. System administrators can trackback the person responsible for undesirable changes in the system

[Must Know Linux/Unix Commands](https://cdn.guru99.com/images/sudo_mv.png)

For renaming file:

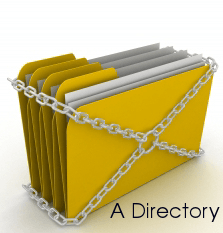
***mv filename newfilename***

[](https://cdn.guru99.com/images/mv%281%29.png)

**NOTE**: By default, the password you entered for sudo is retained for 15 minutes per terminal. This eliminates the need of entering the password time and again.

You only need root/sudo privileges, only if the command involves files or directories not owned by the user or group running the comman

## Directory Manipulations

[](https://cdn.guru99.com/images/Direct.png)

Enough with File manipulations!  Let's learn some directory commands

Creating Directories

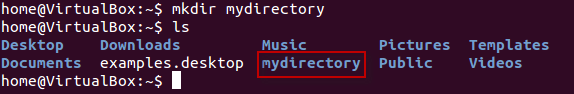
Directories can be created on a Linux operating system using the following command

***mkdir***

This command will create a subdirectory in your present working directory, which is usually your "Home Directory".

For example,

***mkdir mydirectory***

[](https://cdn.guru99.com/images/MKdir-1.png)

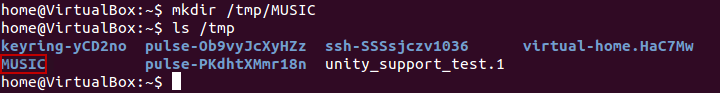
If you want to create a directory in a different location other than 'Home directory', you could use the following command -

***mkdir***

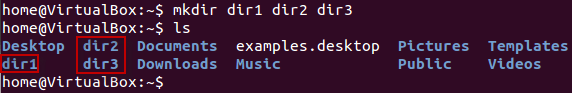
For example:

***mkdir /tmp/MUSIC***

will create a directory 'Music' under '/tmp' directory

[](https://cdn.guru99.com/images/8-2016/linux-5-1.png)

You can also create more than one directory at a time.

[](https://cdn.guru99.com/images/8-2016/linux-5-2.png)

## Removing Directories

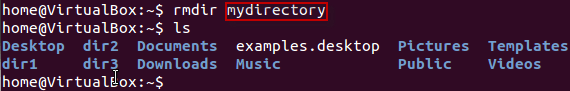
In order to remove a directory, use the command -

***rmdir***

Example

***rmdir mydirectory***

will delete the directory mydirectory

[](https://cdn.guru99.com/images/rmdir.png)

**Tip**: Ensure that there is no file / sub-directory under the directory that you want to delete. Delete the files/sub-directory first before deleting the parent directory.

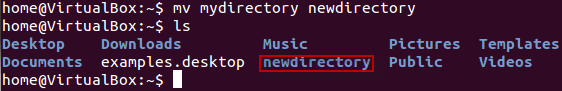
[Must Know Linux/Unix Commands](https://cdn.guru99.com/images/rmdir1.png)

## Renaming Directory

The 'mv' (move) command (covered earlier) can also be used for renaming directories. Use the below given format:

***mv directoryname newdirectoryname***

Let us try it:

[](https://cdn.guru99.com/images/8-2016/linux-5-3.png)

## ****Other Important Commands****

## The 'Man' command

Man stands for manual which is a reference book of a Linux operating system. It is similar to HELP file found in popular softwares.

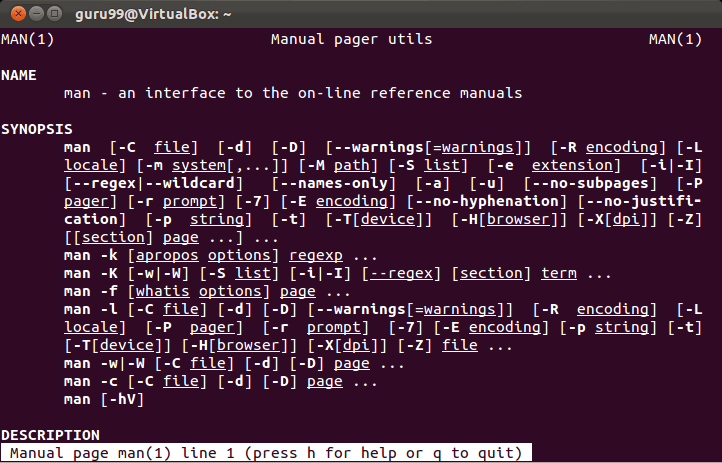
To get help on any command that you do not understand, you can type

***man***

The terminal would open the manual page for that command.

For an example, if we type man man and hit enter; terminal would give us information on man command

[Must Know Linux/Unix Commands](https://cdn.guru99.com/images/man_man.png)

[](https://cdn.guru99.com/images/man_man_1.png)

## The History Command

History command shows all the commands that you have used in the past for the current terminal session. This can help you refer to  the old commands you have entered and re-use them in your operations again.

[](https://cdn.guru99.com/images/history.png)

## ****The clear command****

This command clears all the clutter on the terminal and gives you a clean window to work on, just like when you launch the terminal.

[](https://cdn.guru99.com/images/clear.png)

## Pasting commands into the terminal

Many a times you would have to type in long commands on the Terminal. Well, it can be annoying at times, and if you want to avoid such a situation then copy, pasting the commands can come to rescue.

For copying, the text from a source, you would use **Ctrl + c,** but for pasting it on the Terminal you need to use **Ctrl + Shift + p**. You can also try **Shift + Insert or select Edit>Paste on the menu**

NOTE: With Linux upgrades, these shortcuts keep changing. You can set your preferred shortcuts via Terminal> Edit> Keyboard Shortcuts

Below is a summary of commands we have learned in this tutorial

|  |  |
| --- | --- |
| Command | Description |
| ls | Lists all files and directories in the present working directory |
| ls - R | Lists files in sub-directories as well |
| ls - a | Lists hidden files as well |
| ls - al | Lists files and directories with detailed information like permissions , size , owner etc. |
| cat > filename | Creates a new file |
| cat filename | Displays the file content |
| cat file file2 > file3 | Joins two files (file1, file2) and stores the output in a new file (file3) |
| mv  file "new file path" | Moves the files to the new location |
| mv filename new\_file\_name | Renames the file to a new filename |
| sudo | Allows regular users to run programs with the security privileges of the superuser or root |
| rm | Deletes a file |
| man | Gives help information on a command |
| history | Gives a  list of all past commands typed in the current terminal session |
| clear | Clears the terminal |
| mkdir | Creates a new directory in the present working directory |
| mkdir | Create a new directory at the specified path |
| rmdir | Deletes a directory |
| mv | Renames a directory |

# File Permissions in Linux/Unix

Linux is a clone of UNIX, the **multi-user operating system** which can be accessed by many users simultaneously. Linux can also be used in mainframes and servers without any modifications. But this raises security concerns as an unsolicited or **malign user** can **corrupt, change or remove crucial data**. For effective security , Linux divides authorization into 2 levels

* Ownership
* Permission

 The concept of **permissions** and **ownership** is crucial in Linux . Here, we will discuss both of them. Let us start with the **Ownership**.

## Ownership in Linux files

Every file and directory in your Unix/Linux system is assigned 3 types of owner , given below

## ****User****

A user is the owner of the file. By default, the person who created a file becomes its owner. Hence , a user is also sometimes called an owner.

## Group

A user- group can contain multiple users. All users belonging to a group will have the same access permissions to the file. Suppose you have a project where a number of people require access to a file. Instead of manually assigning permissions to each user, you could add all users to a group, and assign group permission to file such that only this group members and no one else can read or modify the files.

## Other

 Any other user who has access to a file. This person has neither created the file nor he belongs to a usergroup who could own the file. Practically, it means everybody else. Hence, when you set the permission for others, it is also referred as set permissions for world.

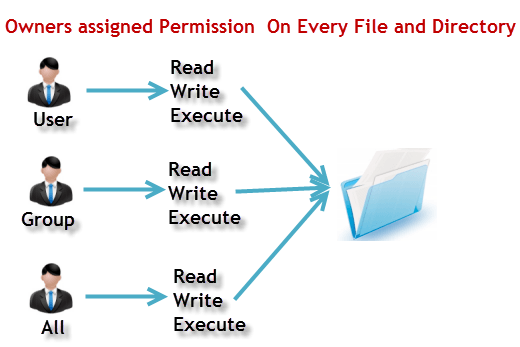
Now, the big question arises how does **Linux distinguishes** between these three user types so that a user 'A' cannot affect a file which contains some other user 'B's' vital information/data. It is like you do not want your colleague, who works on your Linux computer, to view your personal images. This is where **Permissions** set in and they define **user behavior**.

Let us understand the **Permission system** on Linux.

## Permissions

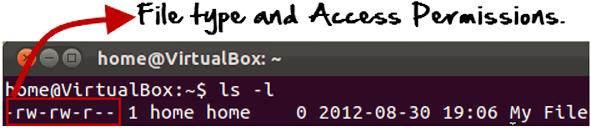
Every file and directory in your UNIX/Linux system has following 3 permissions defined for all the 3 owners discussed above

* **Read** : This permission give you the authority to open and read a file. Read permission on a directory gives you the ability to lists it's content.
* **Write:** The right permission gives you the authority to modify the contents of a file. The write permission on a directory gives you the authority to add , remove and rename files stored in the directory.  Consider a scenario where you have write permission on the file but do not have write permission on the directory where the file is stored. You will be able to modify the file contents. But you will not be able rename, move or remove the file from the directory.
* **Execute :** In windows an executable program usually has an extension ".exe" and which you can easily run. In Unix/Linux , you cannot run a program unless the execute permission is set. If the execute permission is not set, you might still be able to see/modify the program code(provided read & write permissions are set), but not actually run it.

[](https://cdn.guru99.com/images/PermissionsConcept.png)

Lets see this in action

**ls - l** on terminal gives

[](https://cdn.guru99.com/images/Permis_system.png)

Here, we have highlighted **'-rw-rw-r--'**and this weird looking code is the one that tells us about the permissions given to the owner, user group and the world.

[](https://cdn.guru99.com/images/its_a_file.png)

Here, the first **'-'**implies that we have selected a file.

Else, if it were a directory , **d**would have been shown.

[File Permissions in Linux/Unix](https://cdn.guru99.com/images/Directory.png)

The characters are pretty easy to remember.

**r** = read permission  
**w** = write permission  
**x** = execute permission  
**-** = no permission

Let us look at it this way.

The first part of the code is **'rw-'**. This suggests that the owner 'Home' can:

|  |  |
| --- | --- |
| [File Permissions in Linux/Unix](https://cdn.guru99.com/images/no_execute.png) | * Read the file * Write or edit the file * He cannot execute the file since , the execute bit is set to '-'. |

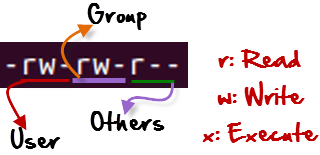
By design many Linux distributions like Fedora , CentOS , Ubuntu etc , will add users to a group of the same group name as the user name. Thus, a user 'tom' is added to a group named 'tom'.

The second part is **'rw-'.** It for the user group 'Home' and group-members can:

* Read the file
* Write or edit the file

The third part is for the world which means any user . It says **'r--'.** This means the user can only:

* Read the file

[](https://cdn.guru99.com/images/permission%281%29.png)

## Changing file/directory permissions with 'chmod' command

Say you do not want your colleague to see your personal images.  This can be achieved by changing file permissions.

We can use the '**chmod'** command which stands for 'change mode'. Using the command, we can set permissions (read, write, execute) on a file/directory for the owner, group and the world.

Syntax:

***chmod  permissions filename***

There are 2 ways to use the command -

1. **Absolute mode**
2. **Symbolic mode**

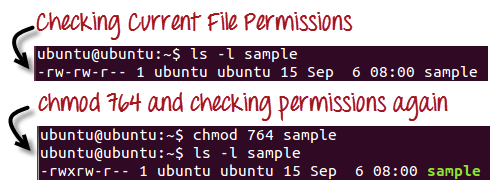
## Absolute(Numeric) Mode

In this mode file **permissions are not represented as characters but a three digit octal number**.

The table below, gives numbers for all for permissions types.

|  |  |  |
| --- | --- | --- |
| **Number** | **Permission Type** | **Symbol** |
| 0 | No Permission | --- |
| 1 | Execute | --x |
| 2 | Write | -w- |
| 3 | Execute + Write | -wx |
| 4 | Read | r-- |
| 5 | Read + Execute | r-x |
| 6 | Read +Write | rw- |
| 7 | Read + Write +Execute | rwx |

Let's see the chmod command in action.

[](https://cdn.guru99.com/images/chmod_new%281%29.png)

In the above given terminal window, we have changed the permissions of the file 'sample to '764'.

|  |  |
| --- | --- |
| [File Permissions in Linux/Unix](https://cdn.guru99.com/images/FilePermissions%281%29.png) | '764' absolute code says the following:   * Owner can read, write and execute * Usergroup can read and write * World can only read   **This is shown as '-rwxrw-r-** |

This is how you can change the permissions on a file by assigning an absolute number.

## Symbolic Mode

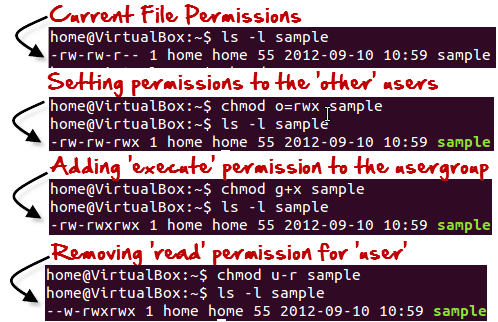
In the Absolute mode you change permissions for all  3 owners. In the symbolic mode you can modify permissions of a specific owner. It makes use of mathematical symbols to modify the file permissions

|  |  |
| --- | --- |
| **Operator** | **Description** |
| **+** | Adds a permission to a file or directory |
| **-** | Removes the permission |
| **=** | Sets the permission and overrides the permissions set earlier. |

The various owners are represented as -

|  |  |
| --- | --- |
| **User Denotations** | |
| u | user/owner |
| g | group |
| o | other |
| a | all |

We will not be using permissions in numbers like 755 but characters like rwx. Let's look into an example

[](https://cdn.guru99.com/images/Symbolic_Mode%281%29.png)

**Changing Ownership and Group**

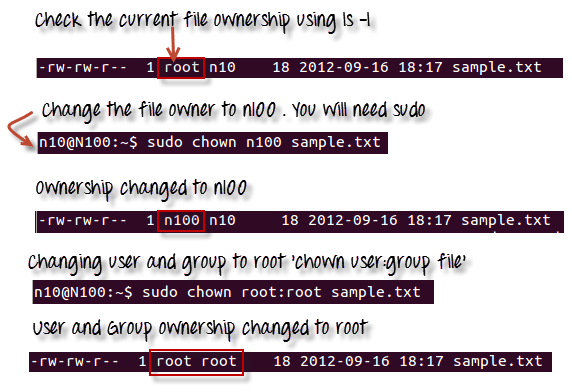
For changing the ownership of a file/directory , you can use the following command:

***chown user***

In case you want to change the user as well as group for a file or directory use the command

**chown user:group**

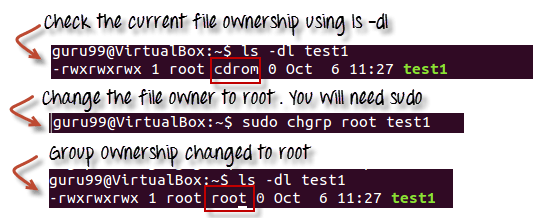
Let's see this in action

[](https://cdn.guru99.com/images/chown_comm%281%29.png)

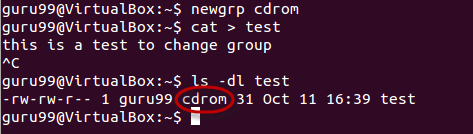
In case , you want to change  group-owner only , use the command

**chgrp**

'**chgrp'** stands for change group.

[](https://cdn.guru99.com/images/chgrp.png)

**Tip**

* The file /etc/group contains all the groups defined in the system
* You can use the command "groups" to find all the groups you are a member of
* [File Permissions in Linux/Unix](https://cdn.guru99.com/images/groups.png)
* You can use the command newgrp to work as a member a group other than your default group
* .[](https://cdn.guru99.com/images/newgrp.png)
* You cannot have 2 groups owning the same file.
* You do not have nested groups in Linux. One group can not be sub-group of other
* x- eXecuting a directory means  Being allowed to "enter" a dir and gain possible access to sub-dirs
* There are other permissions that you can set on Files and Directories which will be covered in a later advanced tutorial

Summary:

* Linux being a multi-user system uses permissions and ownerships for security\
* There are three user types on a Linux system viz. User, Group and Other
* Linux divides the file permissions into read, write and execute denoted by r,w and x
* The permissions on a file can be changed by 'chmod' command which can be further divided into Absolute and Symbolic mode
* The 'chown' command can change the ownership of a file/directory. Use the following commands: chown user file or chown user:group file
* The 'chgrp' command can change the group ownership  **chrgrp group filename**
* What does x - eXecuting a directory mean? A: Being allowed to "enter" a dir and gain possible access to sub-dirs.

**Basic Configuration**

### Set the Hostname

Please follow our instructions for setting your hostname. You can use the following commands to make sure it is set properly:

|  |  |
| --- | --- |
| 1  2 | hostname  hostname -f |

The first command should show your short hostname, and the second should show your fully qualified domain name (FQDN).