FAST NATIONAL UNIVERSTIY OF COMPUTER AND EMERGING SCIENCES, PESHAWAR

DEPARTMENT OF COMPUTER SCIENCE

CL220 - OPERATING SYSTEM LAB

LAB MANUAL # 02

INSTRUCTOR: MUHAMMAD ABDULLAH

SEMESTER SPRING 2021

Contents

BASICS OF UBUNTU

1. Ubuntu's File System Structure

Ubuntu uses the Linux file system, which is based on a series of folders in the root directory. These folders contain important system files that cannot be modified unless you are running as the root user or use <mark>sudo.</mark> This restriction exists for both security and safety reasons; computer viruses will not be able to change the core system files, and ordinary users should not be able to accidentally damage anything vital.

The most root or basic directory in linux is the root directory and in this directory all files and folders are residing.
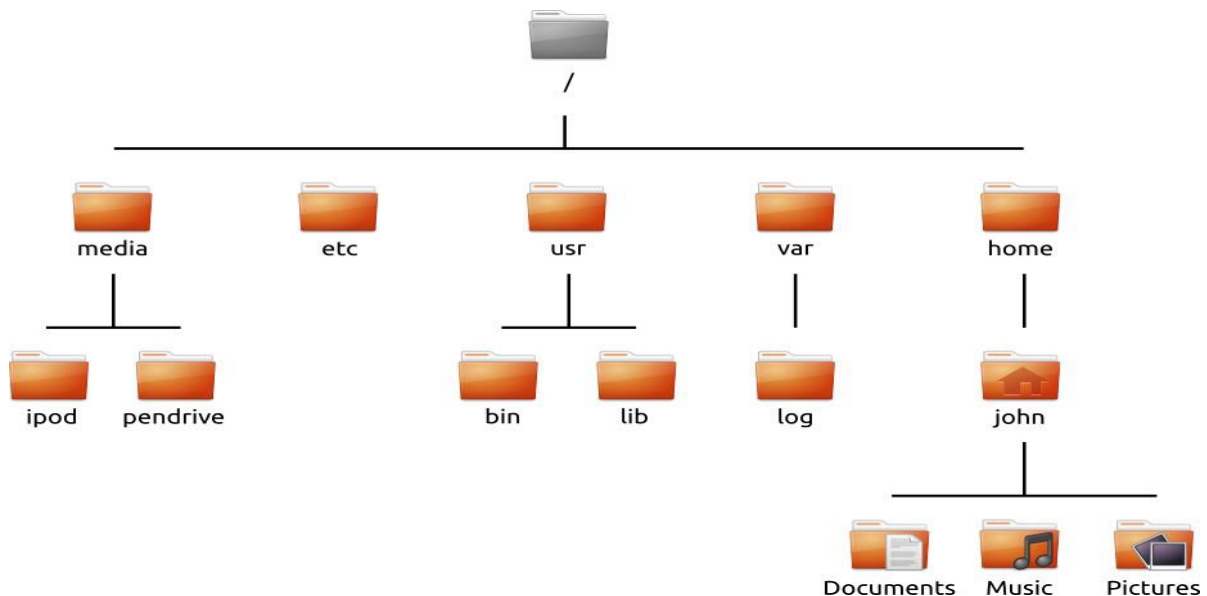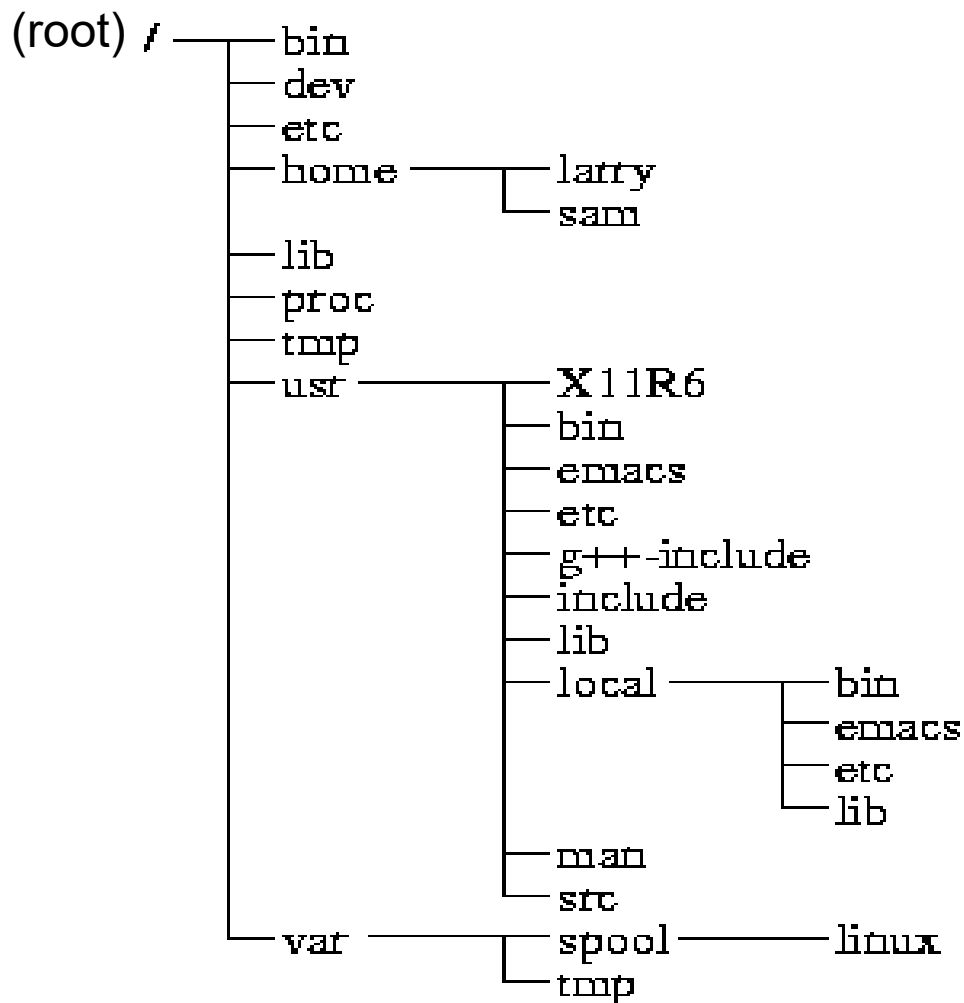

Fig 1: Ubuntu File System Structure

```
(root) /────┬── bin
            ├── dev
            ├── etc
            ├── home ──────┬── larry
            │              └── sam
            ├── lib
            ├── proc
            ├── tmp
            ├── usr ───────┬── X11R6
            │              ├── bin
            │              ├── emacs
            │              ├── etc
            │              ├── g++-include
            │              ├── include
            │              ├── lib
            │              ├── local ──────┬── bin
            │              │               ├── emacs
            │              │               ├── etc
            │              │               └── lib
            │              ├── man
            │              └── src
            └── var ───────┬── spool ──────── linux
                           └── tmp
```

Fig 2: Directory Tree

At the top of the hierarchy is the root directory which is denoted by **/**. The root directory contains all other directories and files on your system. Below the root directory are the following essential directories:

- **/bin** and **/sbin** Many essential system applications (equivalent to **C:\Windows**).

- */etc* System-wide configuration files.

- **/home** Each user will have a subdirectory to store personal files (for example, **/home/yourusername**) which is equivalent to **C:\Users or C:\Documents and Settings** in Microsoft Windows.

- **/lib** Library files, similar to .dll files on Windows.

- A **DLL** file is a library that contains a set of code and data for carrying out a particular activity in **Windows**.

  A DLL is a library that contains code and data that can be used by more than one program at the same time. **For example,** in windows operating systems, the Comdlg32 DLL performs common dialog box related functions. Each program can use the functionality that is contained in this DLL to implement an Open dialog box.
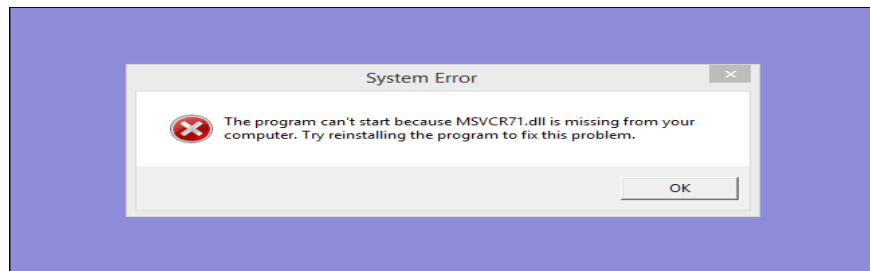

Fig 3: Window Dialog box

- **/media** Removable media (cd-roms and usb drives) will be mounted in this directory.
- **/root** This contains the root user's files (not to be confused with the root directory).
- **/usr** Pronounced "user," it contains most program files (not to be confused with each user's
- home directory). This is equivalent to C:\Program Files in Microsoft Windows.
- **/var/log** Contains log files written by many applications.

2. The Linux System

User commands	
Shell	
Kernel	File Systems
	Device Drivers
Hardware	

i.   User Commands

User commands includes executable programs and scripts.

ii.   Shell

The shell interprets user commands. It is responsible for finding the commands and starting their execution. Several different shells are available. Bash (Bourne-Again **Shell**) is popular.
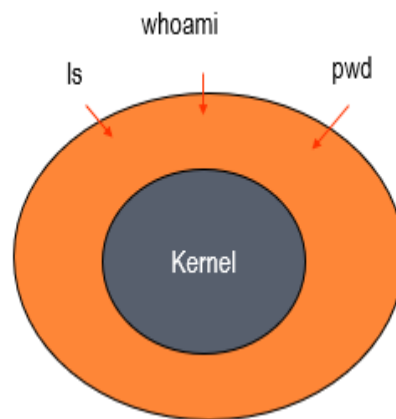
Shell is a program which processes commands and returns output, like bash in Linux. Terminal is a program that run a shell.

The shell is a program that takes commands from the keyboard and gives them to the operating system to perform.

iii.   Kernel

The kernel manages the hardware resources for the rest of the system.

Linux Shell

- Shell interprets the command and request service from kernel.

- Similar to DOS but DOS has only one set of interface while Linux can select different shell

  - Bourne Again shell (Bash), TC shell (Tcsh), Z shell (Zsh)

- Different shell has similar but different functionality.

- Bash is the default for Linux.

- Graphical user interface of Linux is in fact an application program work on the shell.

3. Terminal

In order to fully realize the power of Ubuntu, you will need to learn how to use the terminal. Most operating systems, including Ubuntu, have two types of user interfaces.
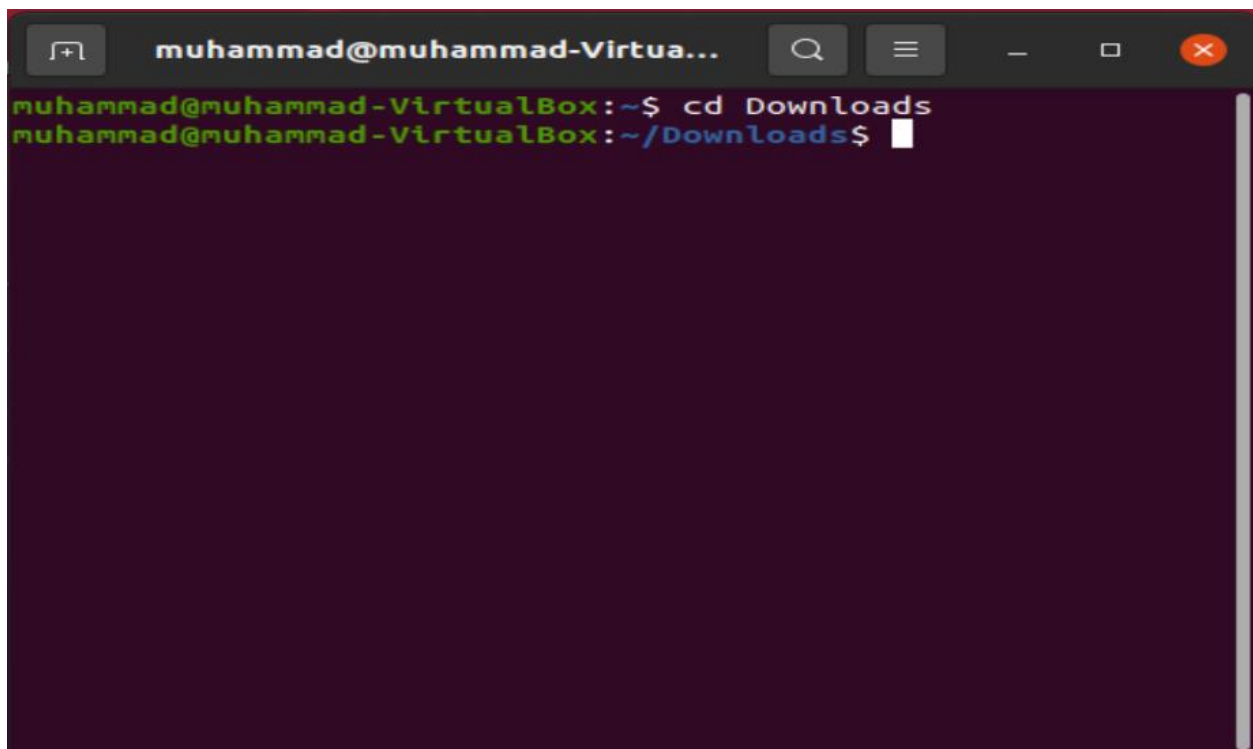
The first is a GUI. This is the desktop, windows, menus, and toolbars you click to get things done. The second, much older type of interface is the command-line interface (CLI). The terminal is Ubuntu's CLI. It is a method of controlling some aspects of Ubuntu using only commands that you type on the keyboard.

Why would you want to use the terminal?

You can perform most day-to-day activities without ever needing to open the terminal. However, the terminal is a powerful and invaluable tool that can be used to perform many useful tasks you might not be able to accomplish with a gui. **For example:**

- Troubleshooting any difficulties that may arise when using Ubuntu sometimes requires you to use the terminal.
- A command-line interface is sometimes a faster way to accomplish a task. For example, it is often easier to perform operations on many files concurrently using the terminal.
- Learning the command-line interface is the first step towards more advanced troubleshooting, system administration, and software development skills. If you are interested in becoming a developer or an advanced Ubuntu user, knowledge of the command-line is essential.

The terminal gives you access to what is called a shell. When you type a command in the terminal, the shell interprets this command, resulting in the desired action. All commands in the terminal follow the same approach: Type a command, possibly followed by some parameters, and press Enter to perform the specified action. Parameters (also called switches) are extra segments of text, usually added at the end of a command, that change how the command itself is interpreted. These usually take the form of **-h** or **--help**, for example. In fact, **--help** can be added to most commands to display a short description of the command, as well as a list of any other parameters that can be used with that command. Often, some type of output will be displayed confirming the action was completed successfully, although this can depend on the command being executed. For example, using the **cd** command to change your current directory will change the prompt but will not display any output, as shown in the figure below.
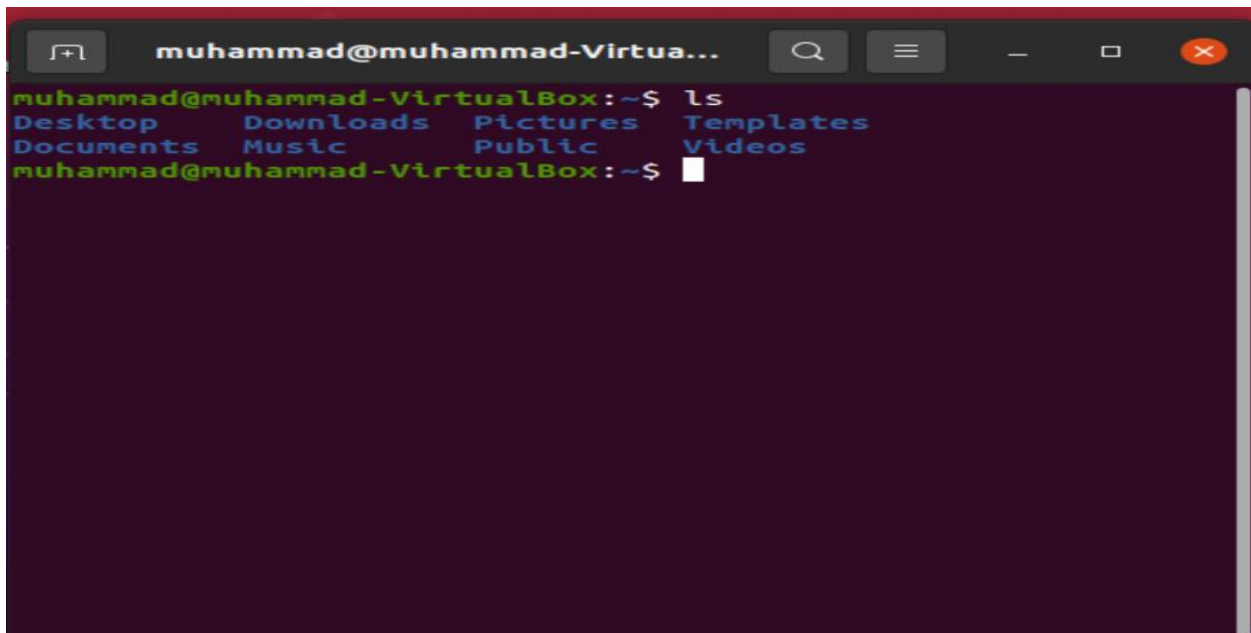
Fig 4: Changing the current directory

4. Basic Commands

a) **ls:** list directory contents

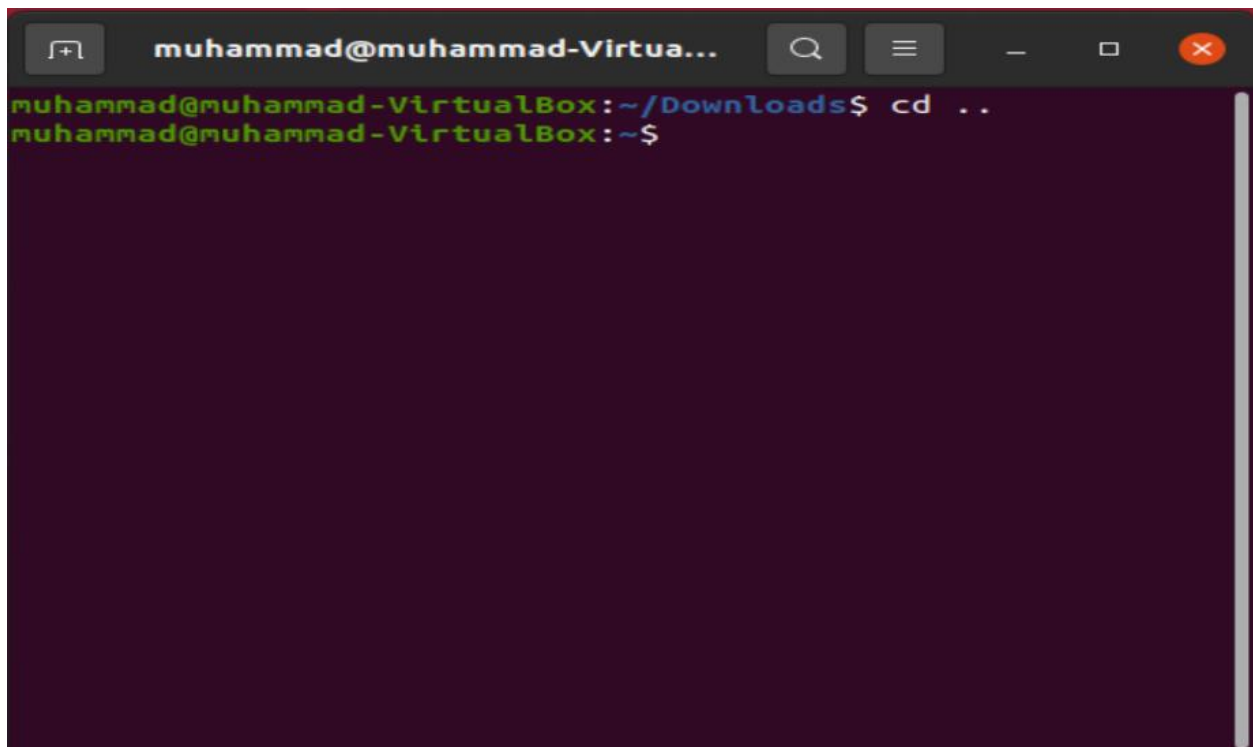(The **ls** command will show you the list of files in your current directory).

Fig 5: ls command

b) cd: Change Directory
(The **cd** command will allow you to change directories)

Fig 6: cd command

c. pwd: print the current/working directory.

(The **pwd** command will allow you to know in which directory you are currently working)

Fig 7: pwd command

d) adduser : Addition of new user

(This command will create a new user in /home directory).
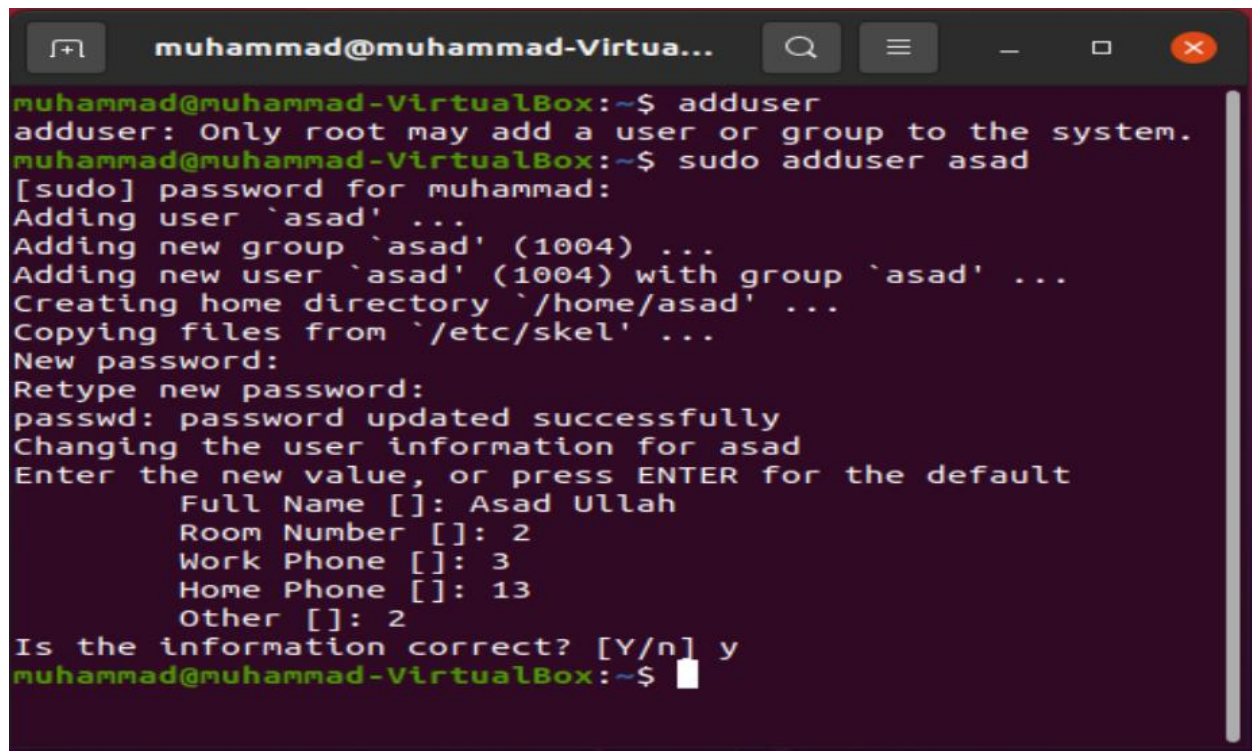
Fig 8: adduser command

e) passwd : Change password for user.
(This command changes the password of a specific user).

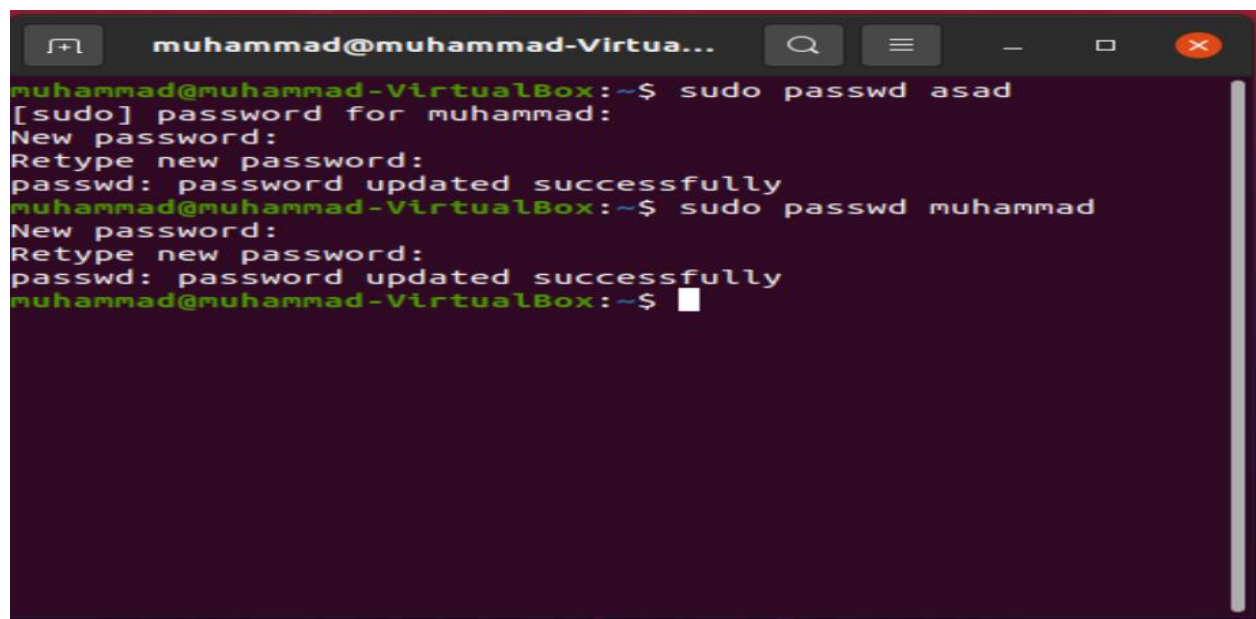*Fig 9: passwd command*

f) sudo command
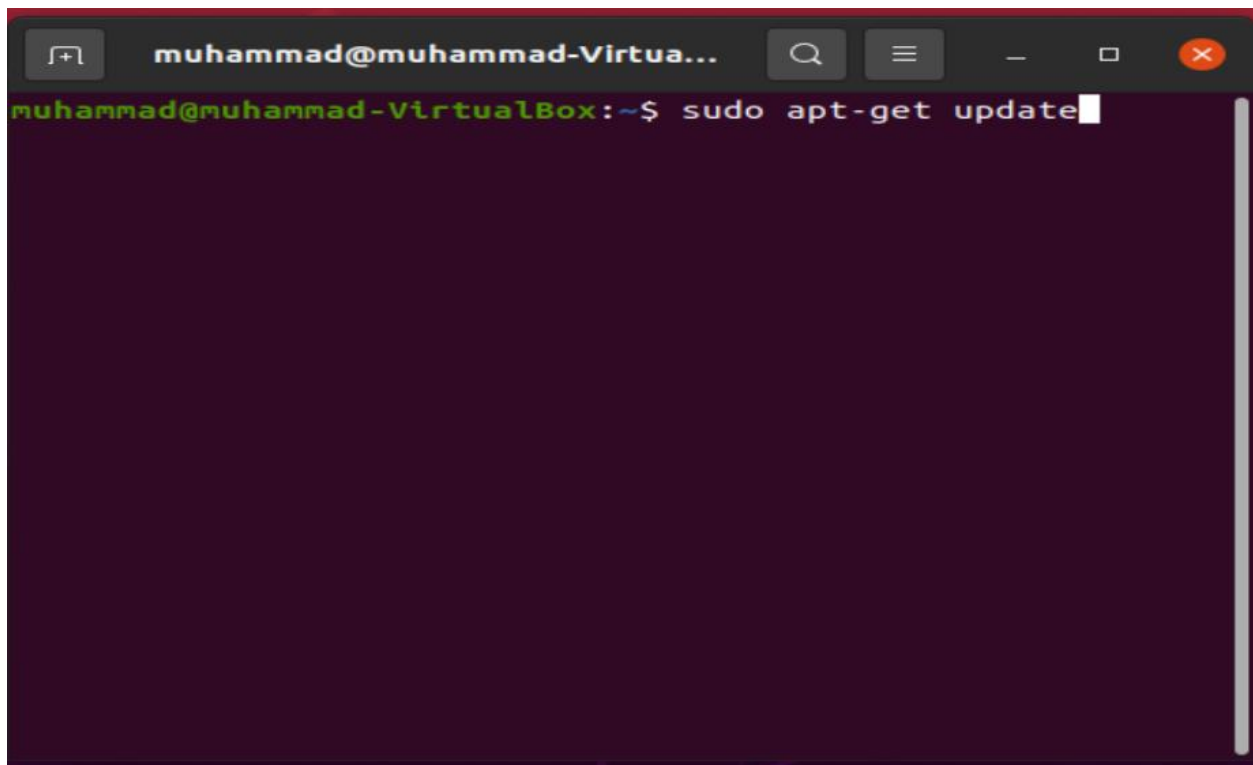(run command as a super user (root))

Fig 10: running a command as a super user

Note: apt-get update updates the list of available packages and their versions, but it does not install or upgrade any packages. **sudo apt-get upgrade** actually installs newer versions of the packages you have. After updating the lists, the package manager knows about available updates for the software you have installed.

apt: Advanced Package Tool, or APT, is a free-software user interface that works with core libraries to handle the installation and removal of software on Debian, Ubuntu, and related Linux distributions.

g) ifconfig:

show network information

h) iwconfig:

show wireless information

More about ls command

ls: The **ls** command will show you the list of files in your current directory. **By default we are in home directory.**

ls Documents /: will show list of files or directories in Documents directory.

ls /: will show us the contents of root directory.

ls ..: go one folder back.

ls ../.. : go two folder back.

ls-l: long list which list the files in the current working directory in long format.

drwxr: these are rights of logged in owner.

d means directory here, **r** means read, **w** means write and **x** means execute and **r** means again read. So, these are the rights of the owner. Owner means whoever is logged in.

so muhammad is the owner in my case which has the to read, write and execute the desktop directory.

xr: is the group rights. x means execute and r means read right of the group.

x: is for the others which is the execute right of directory.

ls -a: this will show you the hidden files also.

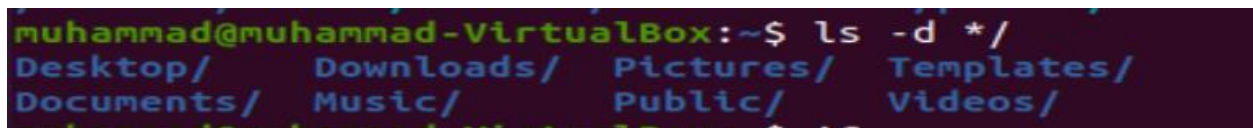**ls -al:** will give you both hidden files and long list.

ls -al: sort directory by size.

ls Documents/*.html: will show only html files.

ls Documents/*.*: will show all the files.

ls -lS >output.txt: will create file of type txt namely output and will put all contents displayed by ls -lS command in it.

ls -d */: will show all the directories only.

```
muhammad@muhammad-VirtualBox:~$ ls -d */
Desktop/      Downloads/   Pictures/   Templates/
Documents/    Music/       Public/     Videos/
```

man ls: This command is used to know more about ls command.