LAB # 01

Statement Purpose:

This lab will introduce the Linux Operating System to you. You will learn the how to create VM using Virtual-Box, Installing Ubuntu on VM and the basic syntax of Linux Commands

Activity Outcomes:

This lab teaches you the following topics:

- Introduction of Linux OS, Linux Distros and Virtual Machines
- Creating VM in Virtual-Box
- Installing Ubuntu on VM
- Writing basic commands in CLI

Instructor Note:

As pre-lab activity, read Chapter 1 to 6 from the book “The Linux Command Line”, William E. Shotts, Jr.

1) Stage J (Journey)

1. Linux

Linux is a generic term referring to Unix-like computer operating systems based on the Linux kernel. Their development is one of the most prominent examples of free and open source software collaboration; typically all the underlying source code can be used, freely modified, and redistributed by anyone. Many quantitative studies of free / open source software focus on topics including market share and reliability, with numerous studies specifically examining Linux. The Linux market is growing rapidly.

1.1 History

All modern operating systems have their roots in 1969 when Dennis Ritchie and Ken Thompson developed the C language and the Unix operating system at AT&T Bell Labs. They shared their source code (yes, there was open source back in the Seventies) with the rest of the world, including the hippies in Berkeley California. By 1975, when AT&T started selling Unix commercially, about half of the source code was written by others. The hippies were not happy that a commercial company sold software that they had written; the resulting (legal) battle ended in there being two versions of Unix: the official AT&T Unix, and the free BSD Unix.

In the Eighties many companies started developing their own Unix: IBM created AIX, Sun SunOS (later Solaris), HP HP-UX and about a dozen other companies did the same. The result was a mess of Unix dialects and a dozen different ways to do the same thing. And here is the first real root of Linux, when Richard Stallman aimed to end this era of Unix separation and everybody re-inventing the wheel by starting the GNU project (GNU is Not Unix). His goal was to make an operating system that was freely available to everyone, and where everyone could work together (like in the Seventies). Many of the command line tools that you use today on Linux are GNU tools.

The Nineties started with Linus Torvalds, a Swedish speaking Finnish student, buying a 386 computer and writing a brand new POSIX compliant kernel. He put the source code online, thinking it would never support anything but 386 hardware. Many people embraced the combination of this kernel with the GNU tools, and the rest, as they say, is history. Linux kernel version 4.0 was released in April 2015. Its source code grew by several hundred thousand lines (compared to version 3.19 from February 2015) thanks to contributions of thousands of developers paid by hundreds of commercial companies including Red Hat, Intel, Samsung, Broadcom, Texas Instruments, IBM, Novell, Qualcomm, Nokia, Oracle, Google, AMD and even Microsoft (and many more).

1.2 Popularity

Today more than 97 percent of the world's supercomputers (including the complete top 10), more than 80 percent of all smartphones, many millions of desktop computers, around 70 percent of all web servers, a large chunk of tablet computers, and several appliances (DVD players, washing machines, DSL modems, routers, self-driving cars, space station laptops...) run Linux. Linux is by far the most commonly used operating system in the world.

1.3 Linux Distribution

The Linux kernel: This is a small, but essential part of an operating system. The kernel is responsible for interfacing with a device's hardware, providing services to the rest of the system, and performing tasks such as managing the device's CPU and memory. The Linux kernel, like any kernel, can only function as part of a wider operating system. It's impossible to have an operating system that consists solely of a Linux kernel. Since Android is a complete operating system, we can immediately rule out classifying Android as a Linux kernel.

A Linux distribution or distro: This is an operating system that contains the Linux kernel and additional software such as utilities, libraries and a GUI, plus pre-installed applications such as web browsers, text editors, and music players. Even if this additional software was designed specifically to run on the Linux kernel, it's not part of the Linux kernel. When discussing operating systems that use the Linux kernel, the terms 'distribution,' 'distro' and 'operating system' are interchangeable. Since anyone can take the Linux kernel, add their own software, and create a complete operating system, there are countless Linux distros currently available.

A Linux distribution -- often shortened to "Linux distro" -- is a version of the open source Linux operating system that is packaged with other components, such as an installation programs, management tools and additional software such as the KVM hypervisor. Linux distributions, which are based on the Linux kernel, are often easier for users to deploy than the traditional open source version of Linux. This is because most distributions eliminate the need for users to manually compile a complete Linux operating system from source code, and because they are often supported by a specific vendor.
Hundreds of Linux distributions are available today, and each targets specific users or systems such as desktops, servers, mobile devices or embedded devices. Most distributions come ready to use, while others are packaged as source code that a user must compile during installation. A list of most popular Linux distros is given below

1. Ubuntu
2. Slackware
3. SuSE
4. Debain
5. RedHat
6. Fedora
7. Turbo Linux

In this course, we will use the Ubuntu distro. Ubuntu is a popular and to use graphical Linux distro. It was developed and released by Canonical Ltd. in 2004. It is freely available and can be downloaded from http://www.ubuntu.com/download/desktop.

2. Installing Ubuntu

Before discussing the options available to install Ubuntu, we discuss the basic system requirement. It is recommended to Ubuntu should be installed on a system that has a 2 GHz dual core processor with 2GB RAM and 25GB of free hard disk space.

There are many ways to use Ubuntu. It can be installed on a system as a stand-alone OS. Similarly, it can be installed as multi-boot system where it is installed on a system that already has any other OS like windows. Further, it can also be used without installing from a bootable USB. However, in this course we will run the Ubuntu on virtual machine. To create virtual machine we will use Oracle VM Virtual-box. In the following, first we give an overview of Virtual-Box and then discuss the installation process of Ubuntu on VM.

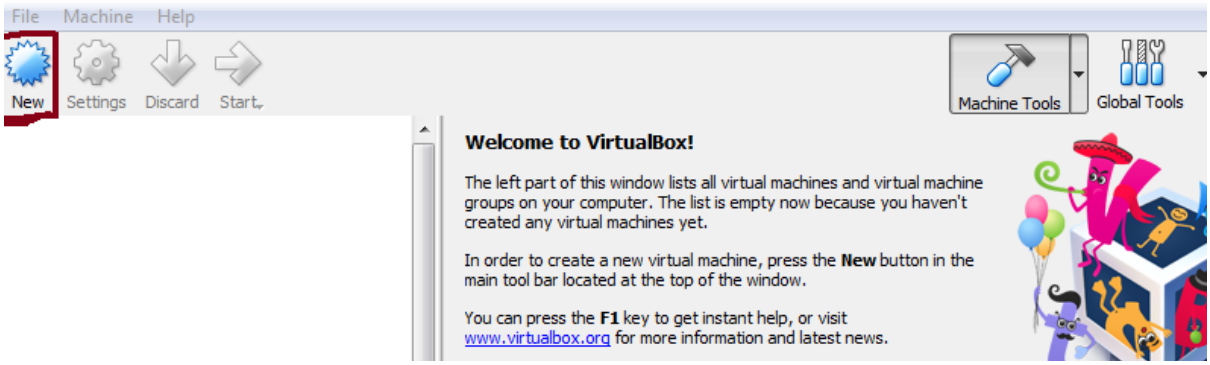
2.1 Oracle VM Virtual-Box

Virtual-Box is a cross-platform virtualization application. What does that mean? For one thing, it installs on your existing Intel or AMD-based computers, whether they are running Windows, Mac, Linux or Solaris operating systems. Secondly, it extends the capabilities of your existing computer so that it can run multiple operating systems (inside multiple virtual machines) at the same time. So, for example, you can run Windows and Linux on your Mac, run Windows Server 2008 on your Linux server, run Linux on your Windows PC, and so on, all alongside your existing applications. You can install and run as many virtual machines as you like -- the only practical limits are disk space and memory. Virtual-Box can be downloaded from https:// www.virtualbox.org/ wiki/ Downloads for free.

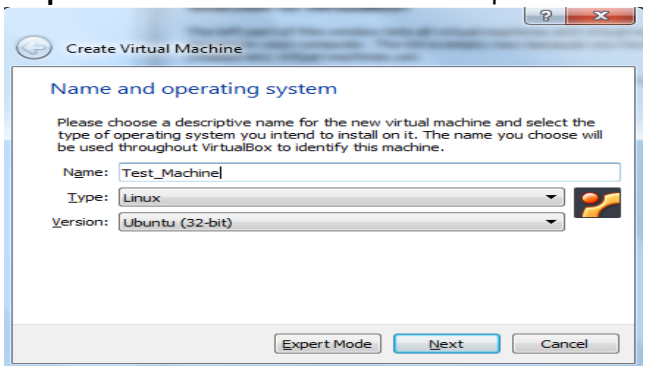
2.2 Installing Ubuntu on VM

Before installing the Ubuntu, we need to ensure that Virtual-Box has been installed on the machine and a compatible version of Ubuntu has been downloaded. To install Ubuntu on VM we need to create a virtual machine on Virtual-Box. The process to create a VM on Virtual-Box is given below.

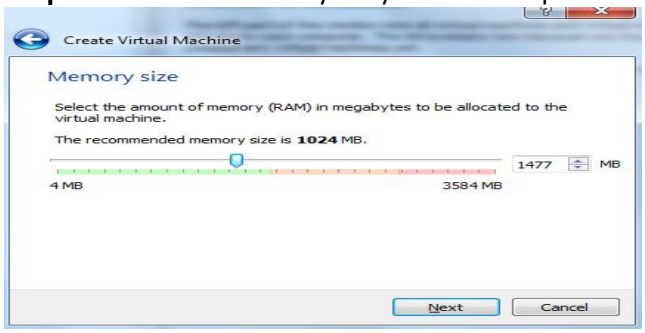
Step 1: Start the Virtual-Box and click the new button to start creating a new VM



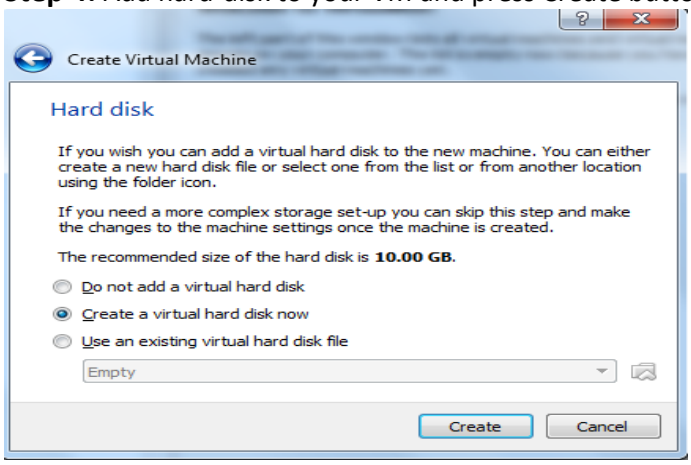
Step 2: Enter the basic information and press Next



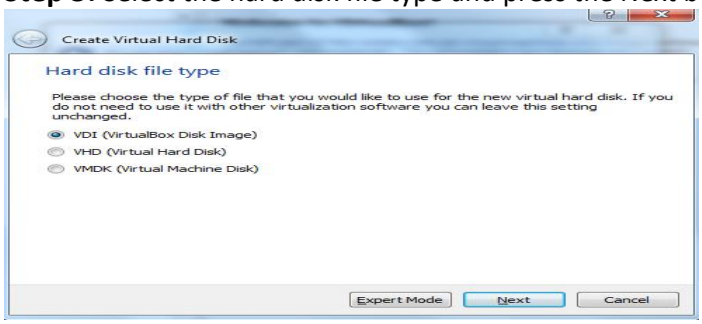
Step 3: Select the memory for your VM and press Next



Step 4: Add hard-disk to your VM and press Create button

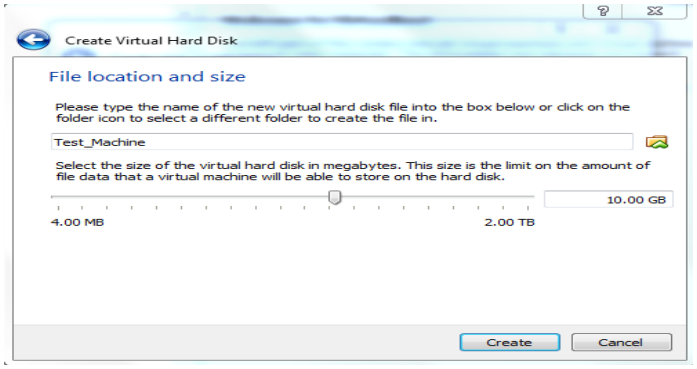


Step 5: Select the hard disk file type and press the Next button

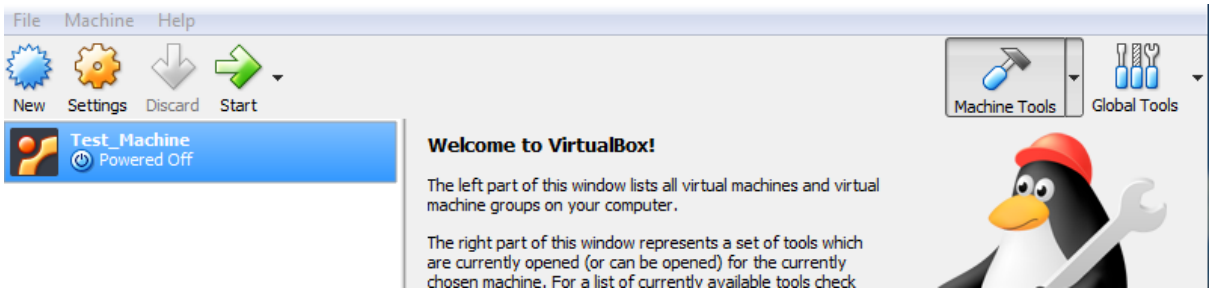


Step 6: Select the storage type on Physical hard disk and press Next

Step 7: Select the location and size on physical hard disk to store the VM and press the Create button

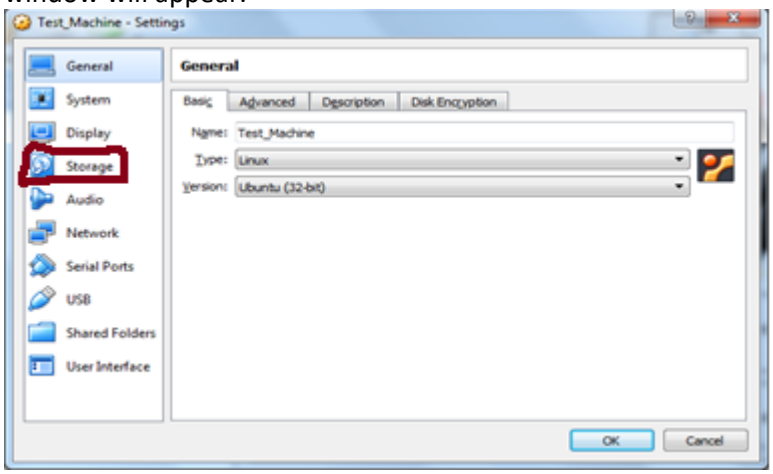


It completes the creation process of VM on Virtual-Box. The newly created VM can be seen on main screen

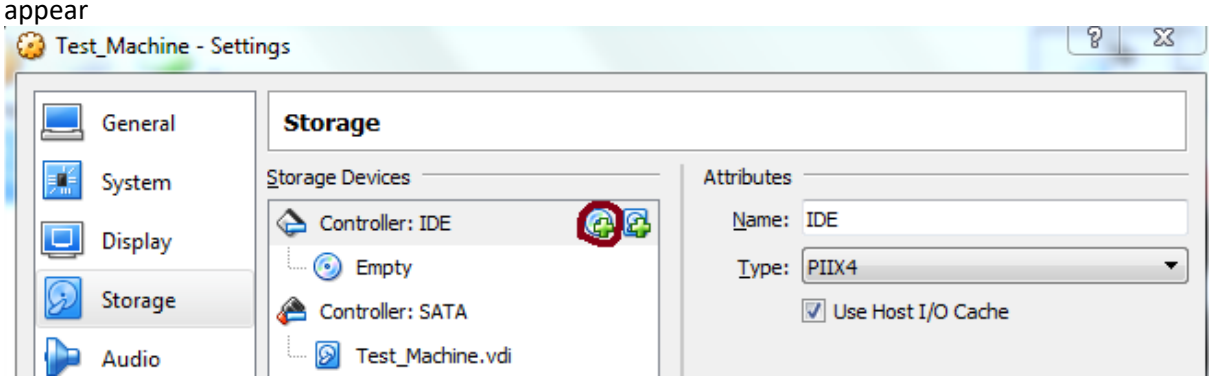


To install Ubuntu on the newly created VM, we need to attach the Ubuntu installation file with it. It can be done as given below.

Step 1: Select the VM on main screen of Virtual-Box and click the settings button. The following window will appear.

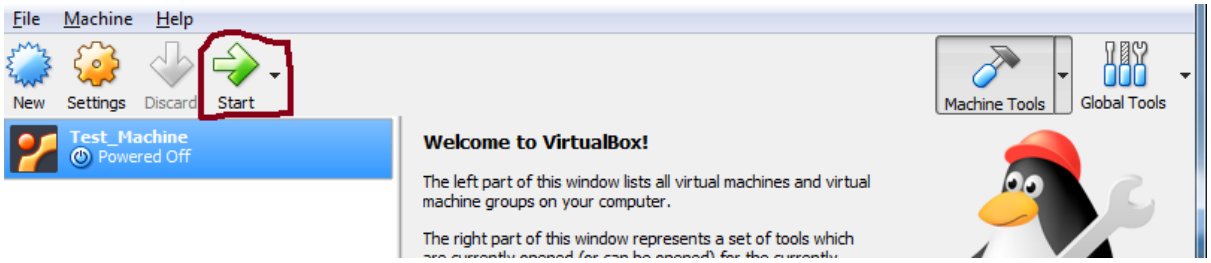


Step 2: Select the storage button as highlighted in the above figure. The following window will appear

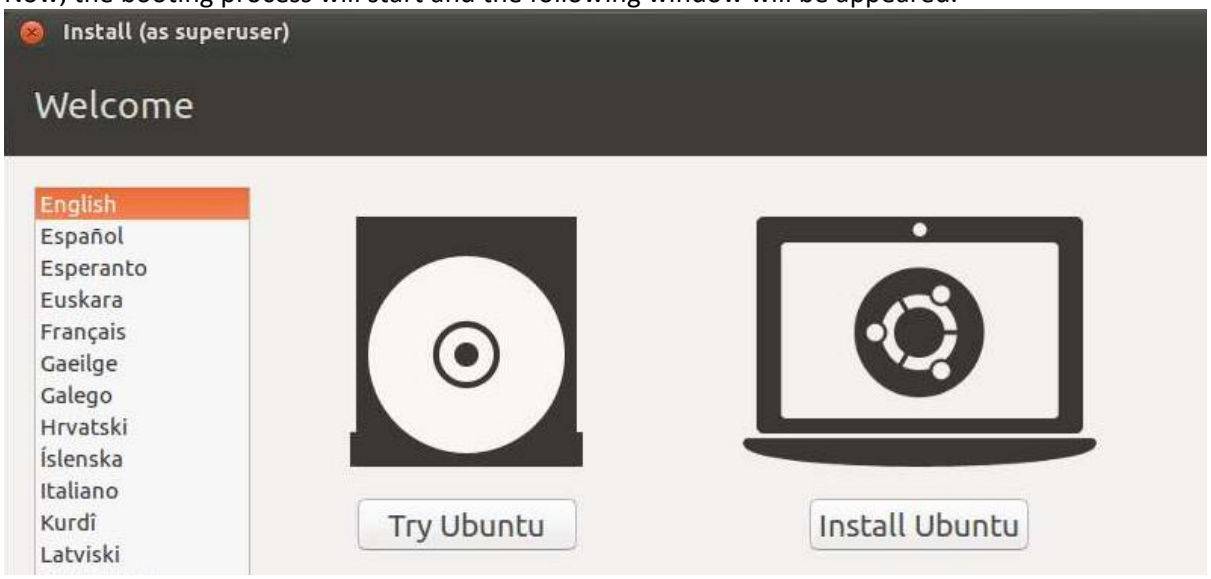


Step 3: Select + button (as highlighted in above figure) and browse and select the Ubuntu installation file downloaded earlier. Now, press the OK button to complete the process of attaching Ubuntu with VM.

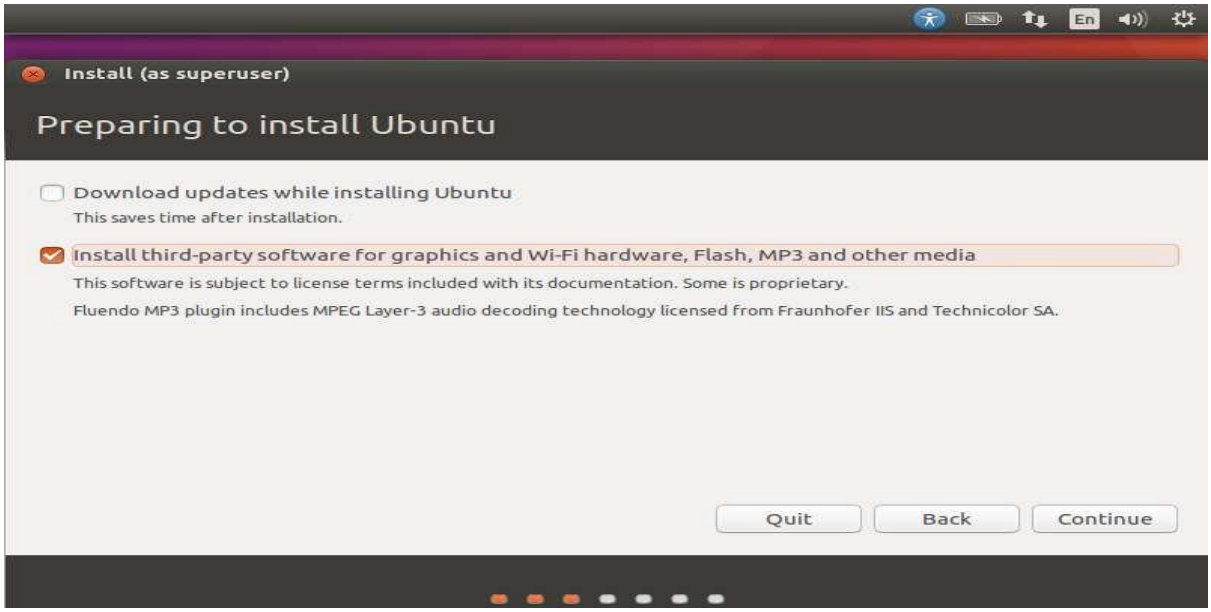
Once the VM has been created and Ubuntu Installation file is attached with it, we can start the installation process. To do this, select the VM on main window of Virtual-Box and click the start button as given below.



Now, the booting process will start and the following window will be appeared.



Here, we can try the Ubuntu without installing. To install Ubuntu, click Install Ubuntu button. The next screen gives you 2 options. One is to download updates in the background while installing and the other is to install 3rd party software. Check the option to install 3 party software. Then click the Continue button.

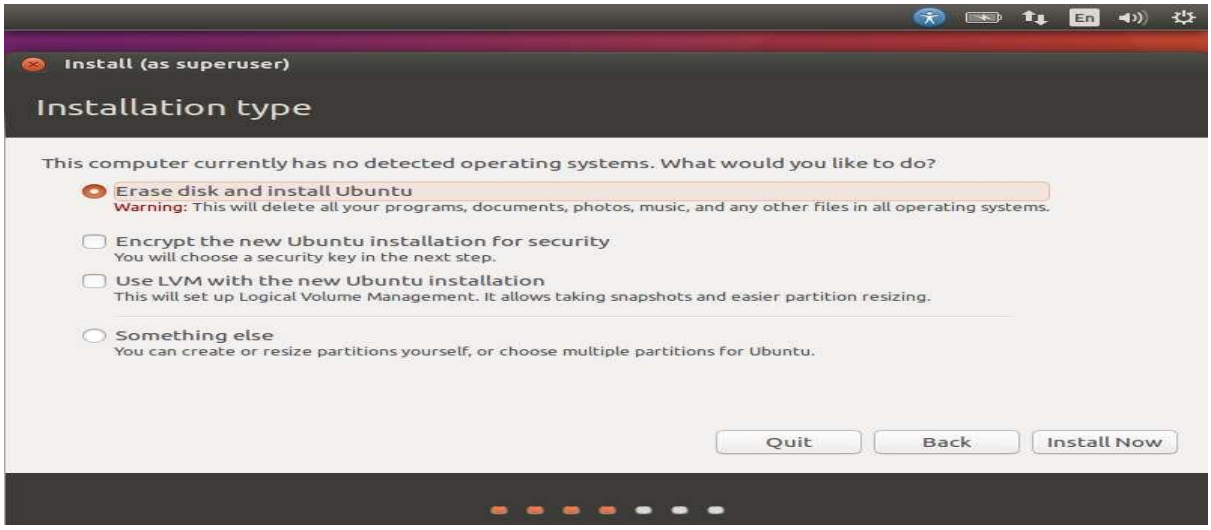


In the next screen, the following options are presented -

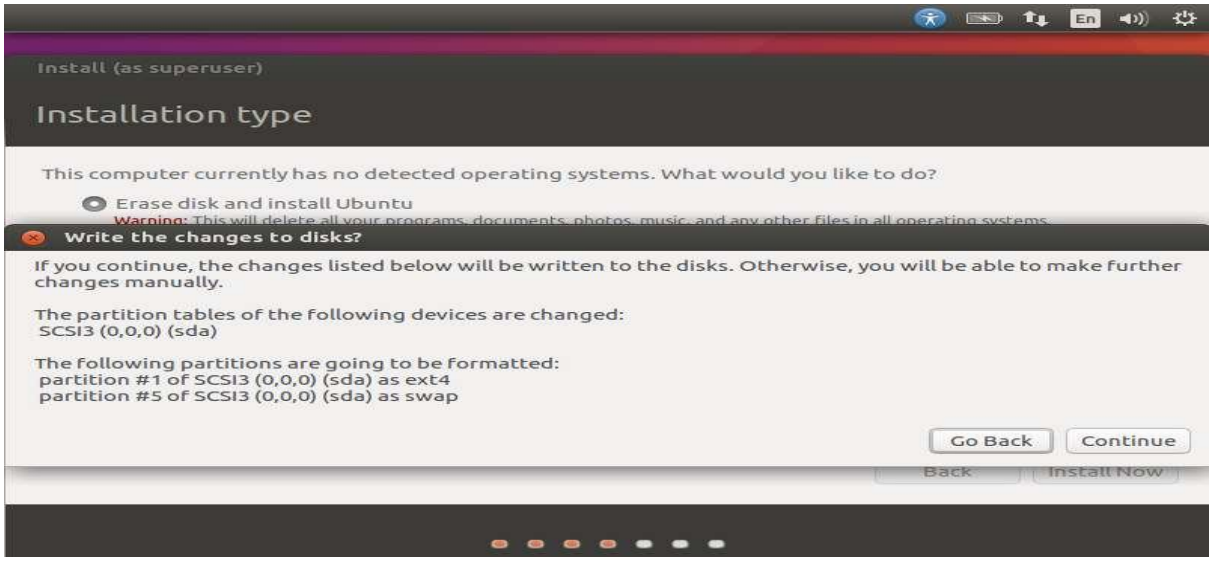
- The disk is erased and the installation is carried out. If there was another operating system already on the disk, then Ubuntu would detect it and give the user the option to install the operating system side by side.

- There is an option to encrypt the installation. This is so that if anybody else were to steal the data, they would not be able to decrypt the data.
- Finally, Linux offers a facility called LVM, which can be used for taking snapshots of the disk.

For the moment, to make the installation simple, let's keep the options unchecked and proceed with the installation by clicking the Install Now button.

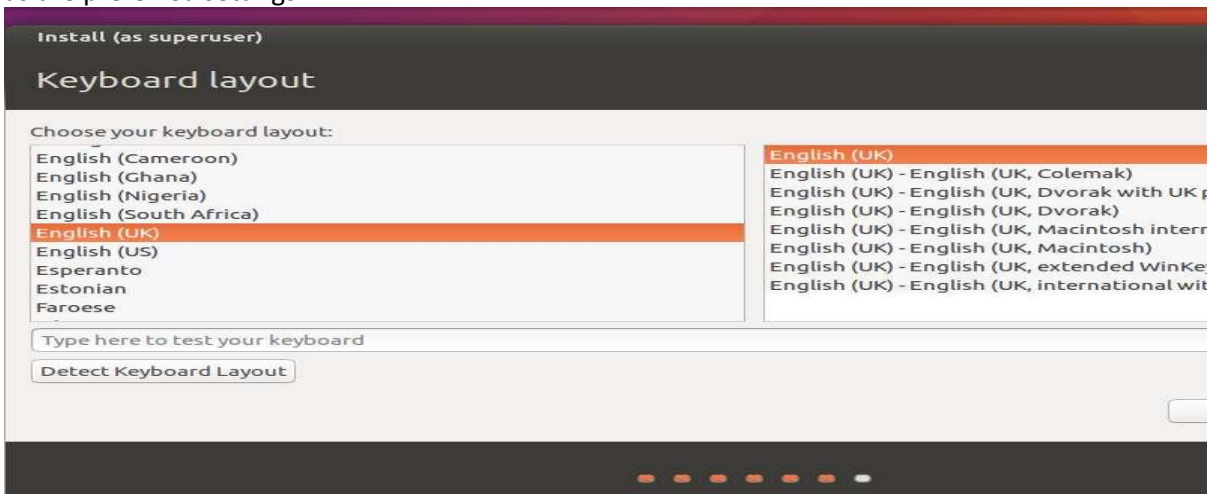


In the following screen, we will be prompted if we want to erase the disk. Click the Continue button to proceed.

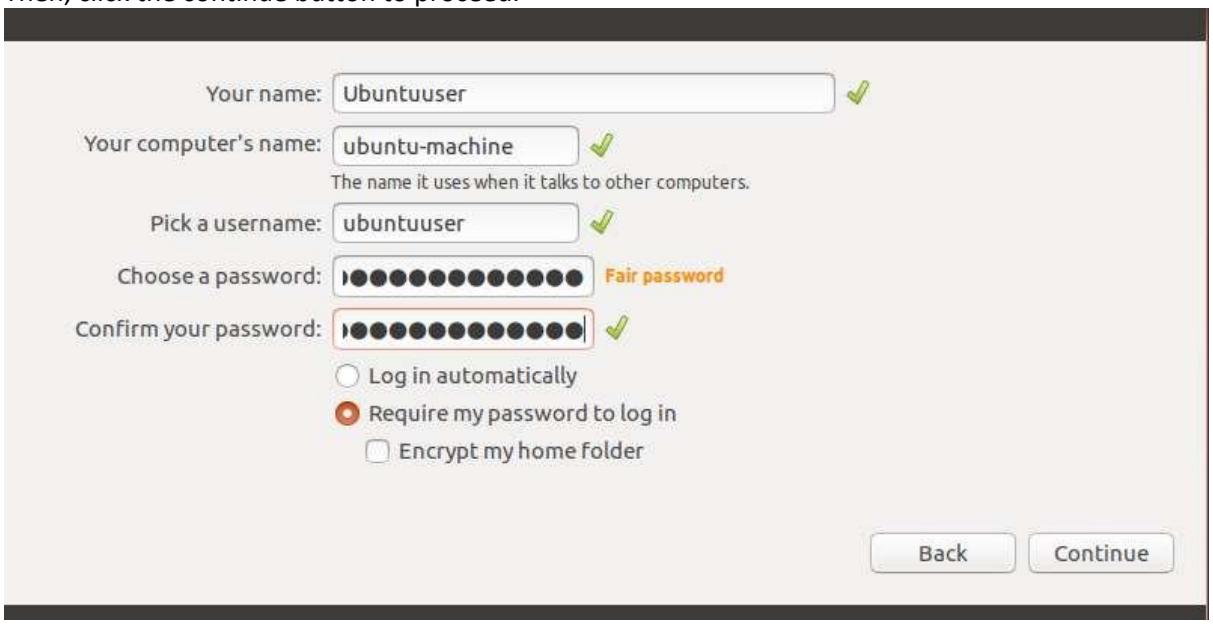


In next screen, we will be asked to confirm our location. Click the Continue button to proceed.

Now, we will be asked to confirm the language and the keyboard settings. Let us select English (UK) as the preferred settings.

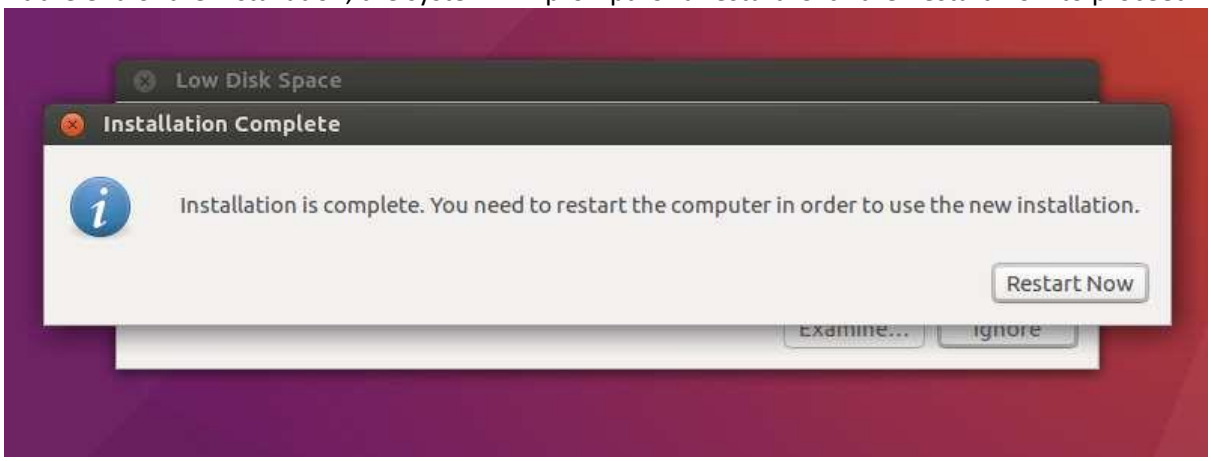


In the following screen, we will need to enter the user name, computer name and password which will be used to log into the system. Fill the necessary details as shown in the following screenshot. Then, click the continue button to proceed.

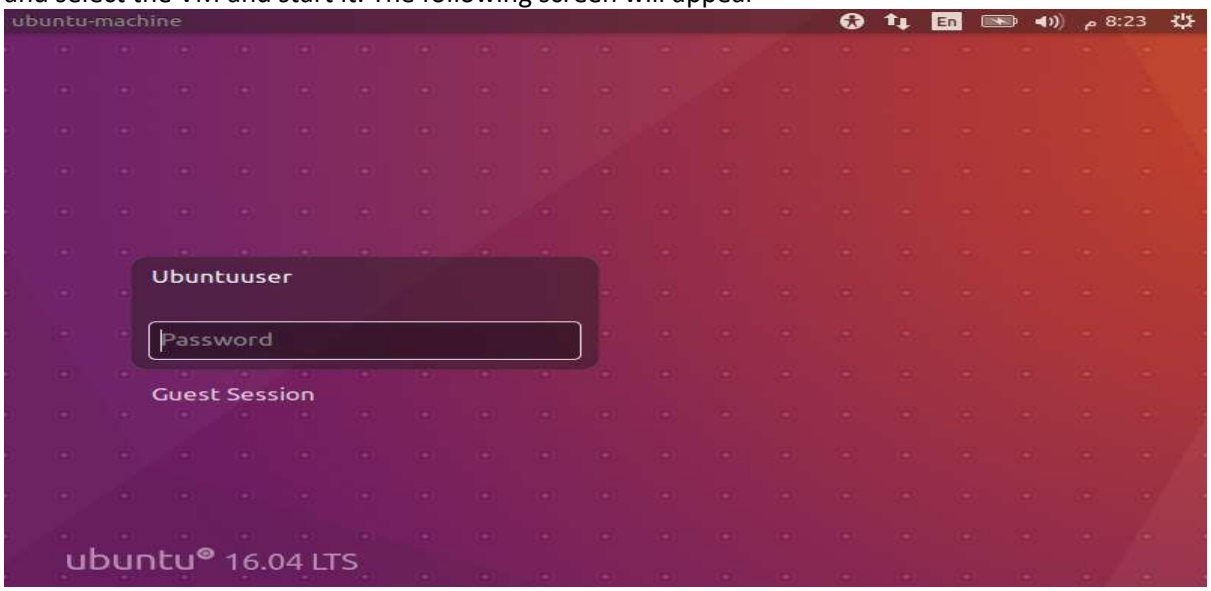


The system will now proceed with the installation and we will see the progress of the installation as shown in the following screenshot.

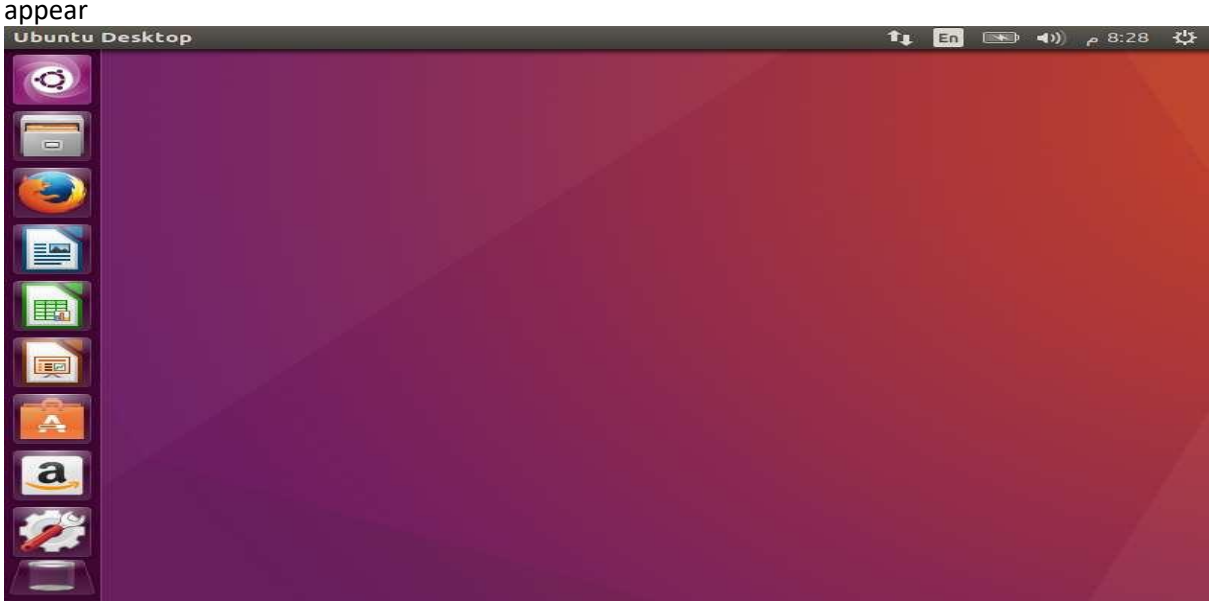
At the end of the installation, the system will prompt for a restart. Click the Restart Now to proceed.



When the installation is completed, we can login into the system. To do this, start the Virtual-Box and select the VM and start it. The following screen will appear



By entering username and password, we can login to the system and the following window will appear

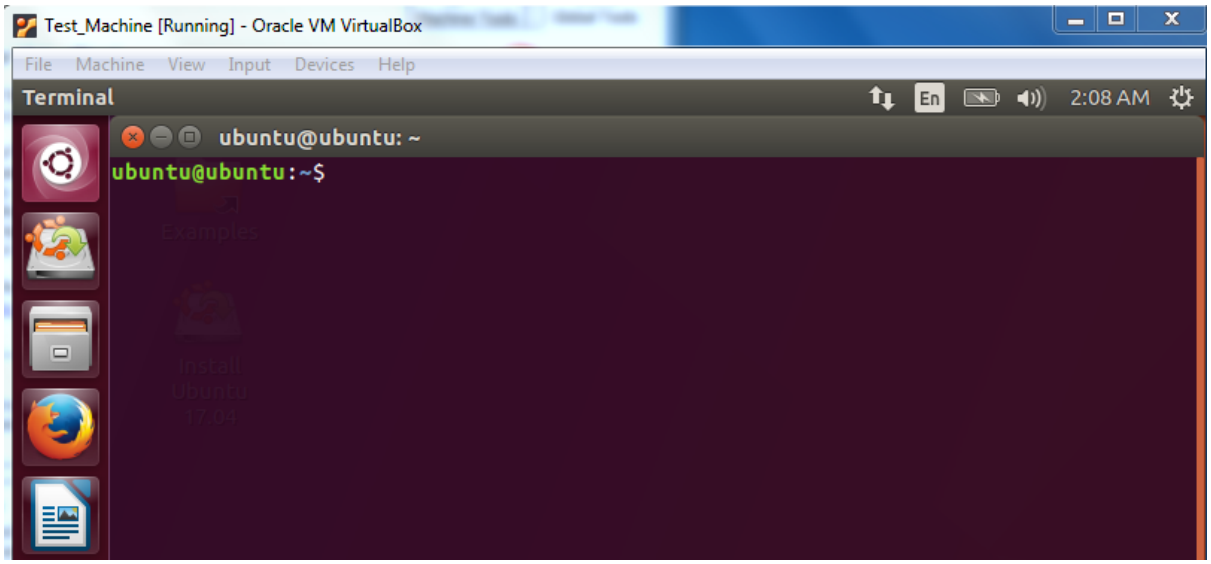


3. Writing Linux Commands

3.1 Command Line Interface

The Command Line Interface (CLI), is a non-graphical, text-based interface to the computer system, where the user types in a command and the computer then successfully executes it. The Terminal is the platform or the IDE that provides the command line interface (CLI) environment to the user. The CLI terminal accepts the commands that the user types and passes to a shell. The shell then receives and interprets what the user has typed into the instructions that can be executed by the OS (Operating System). If the output is produced by the specific command, then this text is displayed in the terminal. If any of the problems with the commands are found, then some error message is displayed.

We can open the terminal by typing Ctrl + Alt + T short-key or by right-clicking the mouse and selecting the Open New Terminal option. The terminal window looks like given below.



3.2 Basic syntax of Linux Commands

A command is an instruction given by a user telling a computer to do something, such a run a single program or a group of linked programs. Commands are generally issued by typing them in at the command line (i.e., the all-text display mode) and then pressing the ENTER key, which passes them to the shell.

A shell is a program that reads commands that are typed on a keyboard and then executes (i.e., runs) them. Shells are the most basic method for a user to interact with the system.

Options and Arguments

This brings us to a very important point about how most commands work. Commands are often followed by one or more *options* that modify their behavior, and further, by one or more *arguments*, the items upon which the command acts. So most commands look kind of like this:

```
command -options arguments
```

Most commands use options consisting of a single character preceded by a dash, for example, "-l", but many commands, including those from the GNU Project, also support long options, consisting of a word preceded by two dashes. Also, many commands allow multiple short options to be strung together.

Command History

Most Linux distributions remember the last 500 commands by default. Press the down-arrow key and the previous command disappears.

3.3 Some Basic Linux Commands

1. **Date** Command: This command is used to display the current data and time.

   Syntax:

   $date

   $date +%ch

   Options:

   a = Abbreviated weekday.
   A = Full weekday.
   b = Abbreviated month.
   B = Full month.
   c = Current day and time.
   C = Display the century as a decimal number.
   d = Day of the month.
   D = Day in „mm/dd/yy" format
   h = Abbrevated month day.
   H = Display the hour.
   L = Day of the year.
   m = Month of the year.
   M = Minute.
   P = Display AM or PM
   S = Seconds
   T = HH:MM:SS format
   u = Week of the year.
   y = Display the year in 2 digit.
   Y = Display the full year.
   Z = Time zone

   To change the format:

   Syntax:

   $date „+%H-%M-%S"

2. Calendar Command: This command is used to display the calendar of the year or the particular month of calendar year.

   Syntax :

   a.$cal <year>
   b.$cal <month> <year>

   Here the first syntax gives the entire calendar for given year & the second Syntax gives the calendar of reserved month of that year.

3. To see the current amount of free space on your disk drives, enter **df**:

4. Likewise, to display the amount of free memory, enter the **free** command.

5. We can end a terminal session by either closing the terminal emulator window, or by entering the **exit** command at the shell prompt

6. **'who'** Command: It is used to display who are the users connected to our computer currently.
   Syntax:
   $who – option‟s
   Options : -
   H–Display the output with headers
   b–Display the last booting date or time or when the system was lastly rebooted

7. **'who am i'** Command: Display the details of the current working directory.
   Syntax:
   $who am i

8. CLEAR' Command**:** It is used to clear the screen.
   Syntax:
   $clear

9. MAN' Command: It help us to know about the particular command and its options & working. It is like„help‟ command in windows .
   Syntax:
   $man <command name>

2) Stage a1 (apply)

Lab Activities:

Activity 1:

In this activity, you are required to perform tasks given below:

1. Display the current date
2. Display the calendar for the current year
3. Display the calendar of 2012
4. Display the calendar of Feb 2015

Solution:

1. date
2. cal 2019
3. cal 2012
4. cal 2 2012

Activity 2:

Perform the following tasks using Linux CLI commands

1. Display the amount of free storage on your machine
2. Display the amount of free memory on your machine
3. Display the user name of the current user
4. Open the man of date free command

Solution:

1. df
2. free
3. who
4. man date

3) Stage V (verify)

Home Activities:

1. In GUI open the Libre Office writer tool create a document that contains information about your favorite place. Try the following short- keys while formatting the document

Keyboard Shortcuts	Functions
Ctrl + C	Copy the Selected text or Object
Ctrl + X	Cut the selected text or object
Ctrl + V	Paste the Copied text or Object
Ctrl + A	Select all text or All files and folder in a Parent folder
Ctrl + B	Make the Selected text as **BOLD**
Ctrl + I	Mark the selected text as *italic*
Ctrl + U	Mark the Selected text Underline
Ctrl + N	Open a New document or Window
Ctrl + S	Save the Current Document
Ctrl + O	Open another Document
Ctrl + P	Print the Document (Print option)
Ctrl + Z	Undo the Last Change you made
Ctrl + Shift + Z	Redo a change that you just undeed

4) Stage a_2 (assess)

Lab Assignment and Viva voce