Lab Number: 03 **Date: July 1st, 2025**

Title: OS Installation & Practice on Basic Networking Command

Theory

Linux

Linux is a family of open-source operating systems, based on the Linux kernel, first released in 1991 by Linus Torvalds. It's known for its flexibility, security, and wide range of applications, from smartphones and servers to supercomputers and embedded systems. While often referred to as an OS, Linux is technically a kernel, which is the core of an operating system. Linux distributions, like Ubuntu or Fedora, bundle the kernel with other software to create a complete OS.

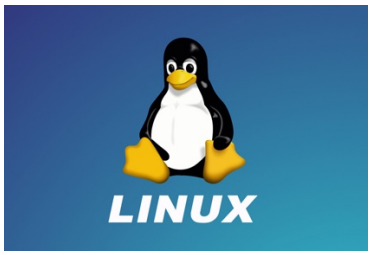


Fig.: Linux OS and its Evolution and Future

Virtual Box

VirtualBox is free and open-source virtualization software that allows users to run multiple operating systems on a single physical machine. It enables the creation and management of virtual machines (VMs), each capable of hosting a different OS alongside the host system. This capability is useful for testing software, developing applications on various platforms, and creating isolated environments for experimentation. Essentially, it provides the functionality of multiple computers within a single computer.

Fig.: Virtual Box

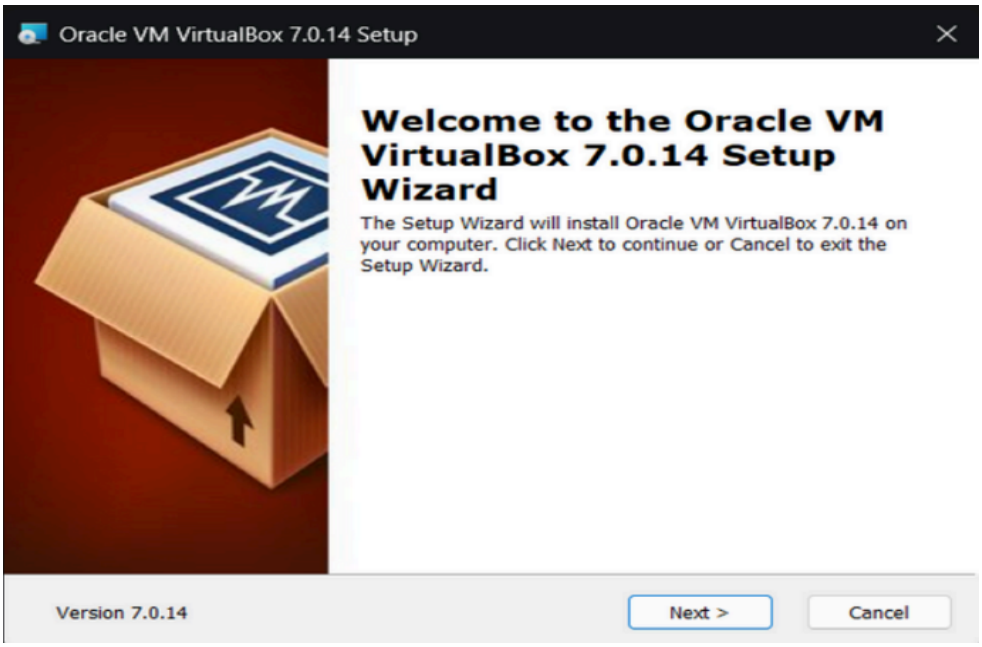
VMware

VMware is a leading provider of virtualization and cloud computing software and services. It enables the creation of virtual machines, which are software-based versions of physical computers, allowing users to run multiple operating systems and applications on a single physical machine. This virtualization technology offers benefits like increased efficiency, improved resource utilization, and simplified data center management. VMware's core product, vSphere, is a suite of virtualization tools that includes the ESXi hypervisor, which manages the allocation of hardware resources to virtual machines.

Fig.: VMware

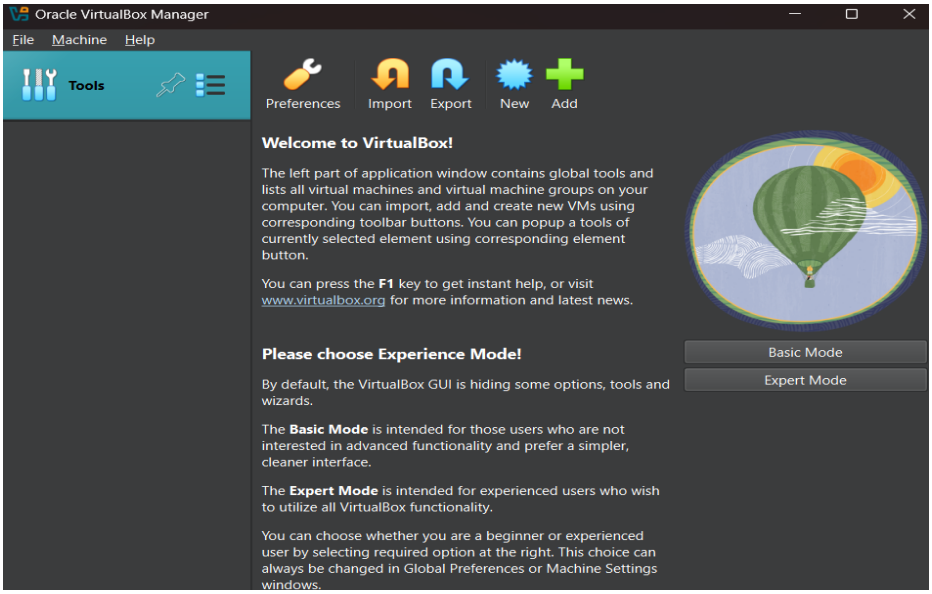
Installing Virtual Box

Installing a virtual machine (VM) enables running multiple operating systems on one physical machine parallelly, useful for testing, development in isolated environments, and efficient resource use. VMs also support backups, disaster recovery, and enhanced security compared to traditional setups.



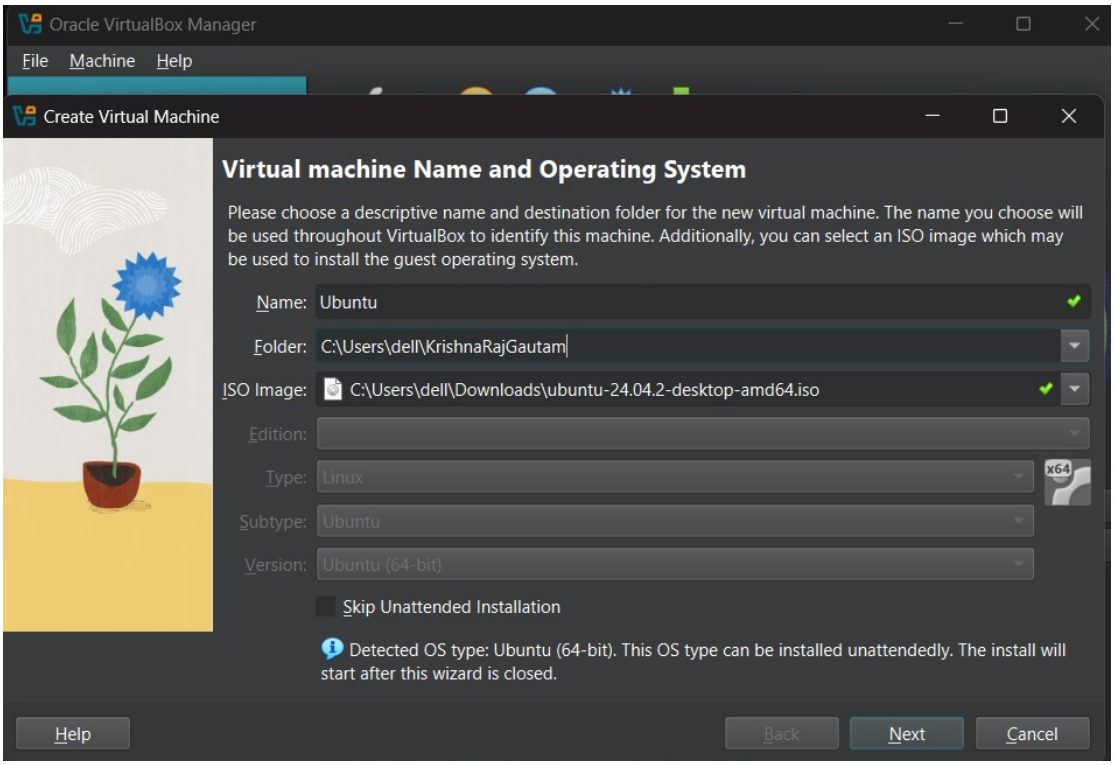
Here, click on **Next** and go to next phase.

Once Virtual Box is installed then we can create, add or import new VMs using toolbar buttons.

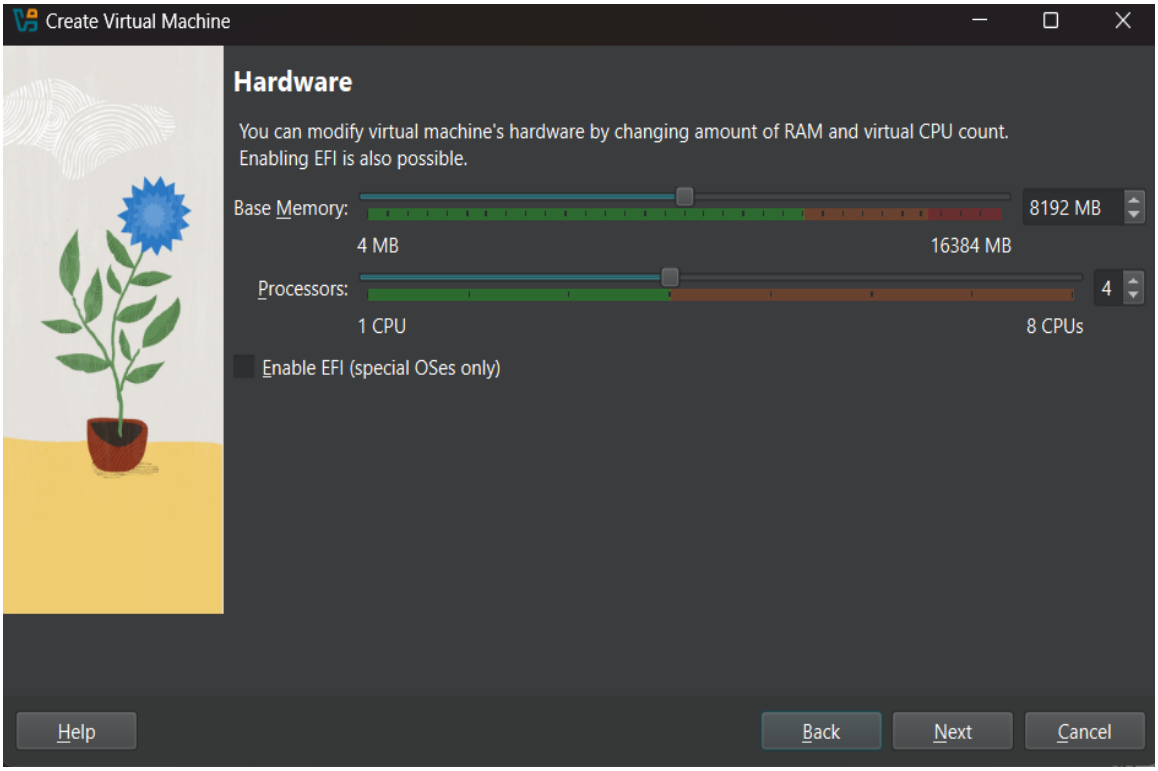


Installing Linux (Ubuntu) Desktop OS:

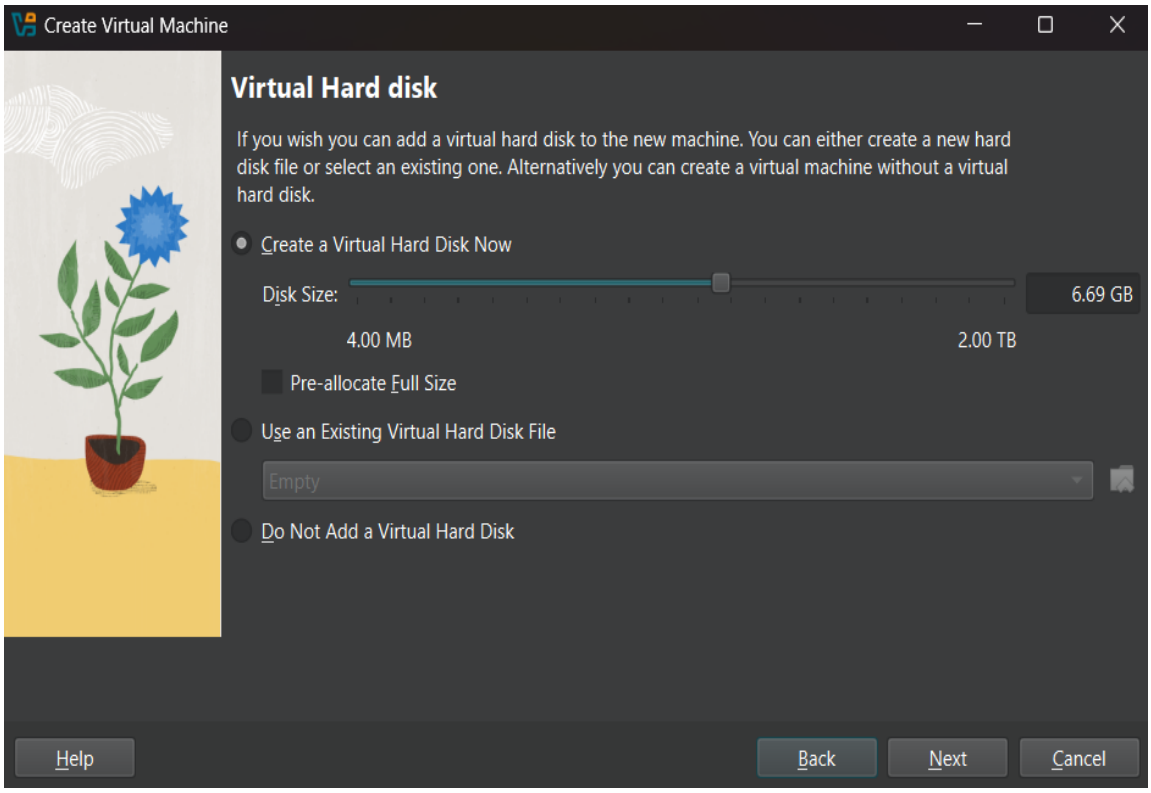
Step 1: Click New on Virtual Machine toolbar and add details about the new VM and required OS.



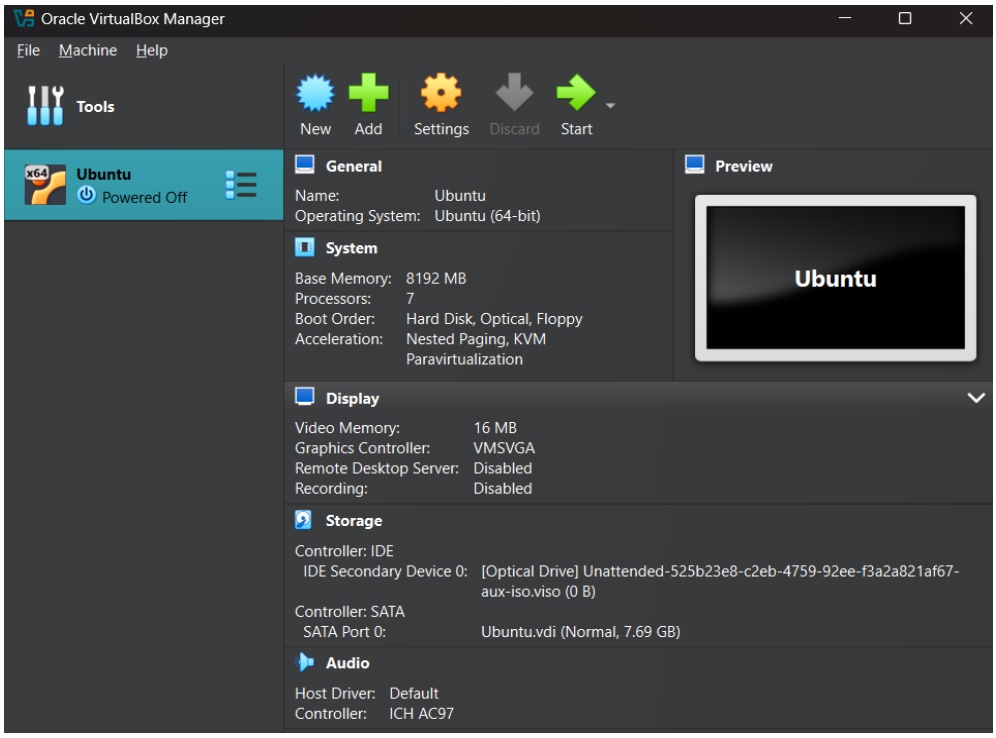
Step 2: Select the Base Memory and Processors to be allocated to the Virtual Machine as per your need.



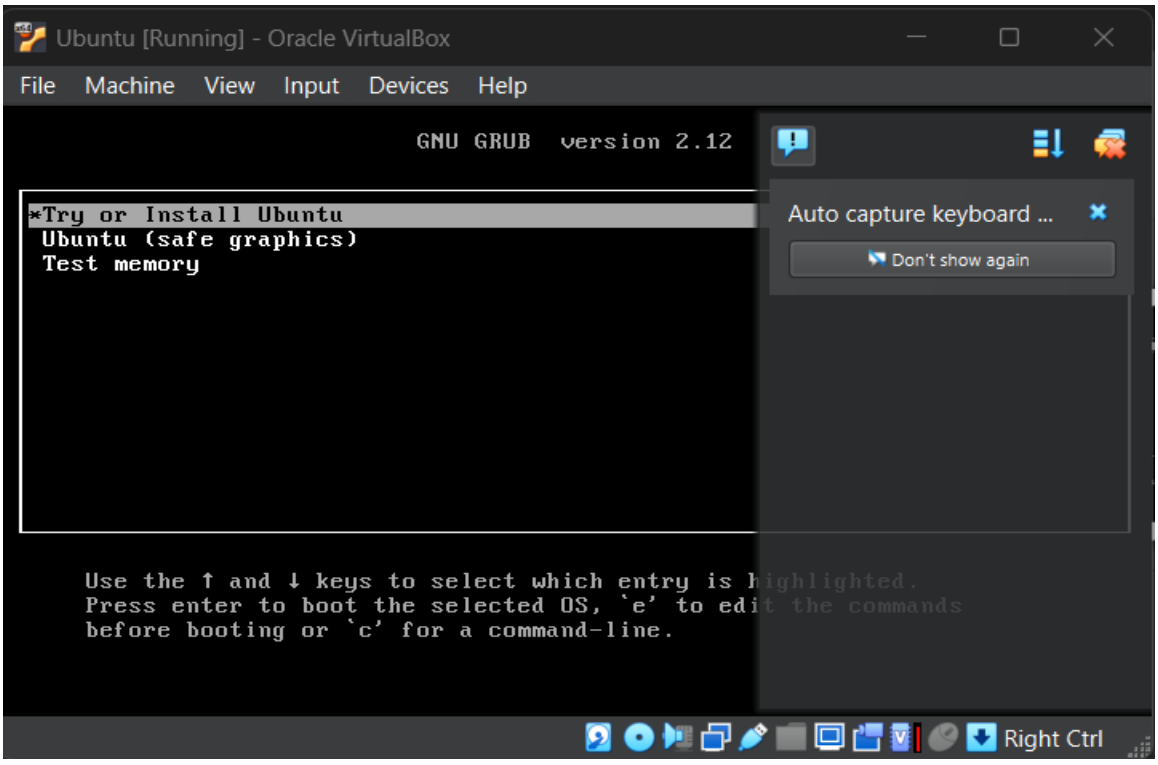
Step 3: Select the size of memory you want for Virtual Hard Disk.



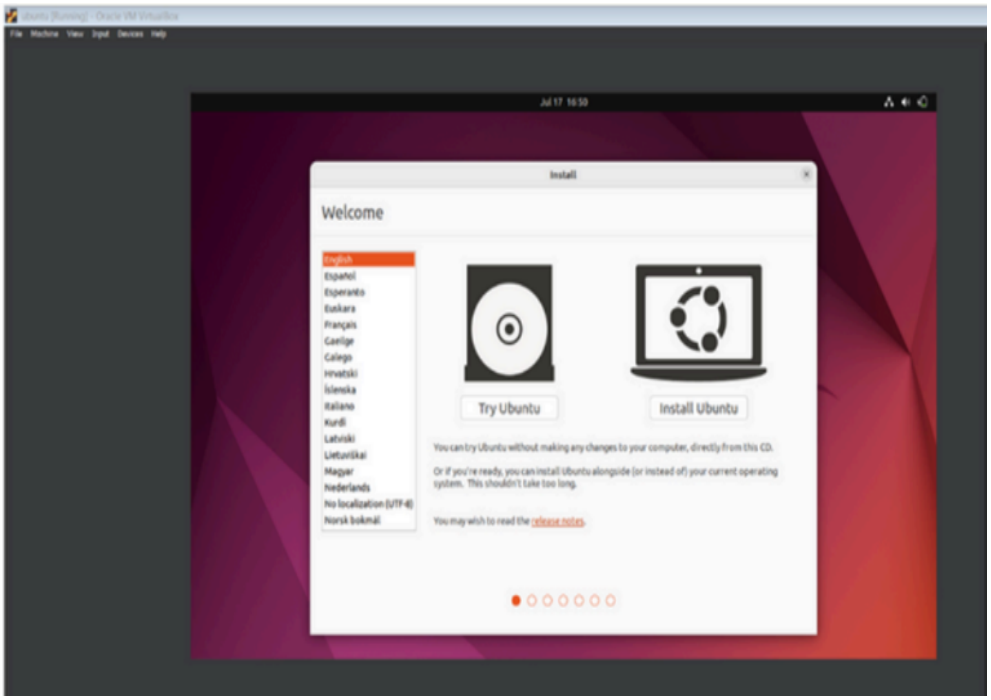
Step 4: This is the interface for the newly created Virtual Machine for Ubuntu Desktop OS. Select Server and click on Start to run the Virtual Machine



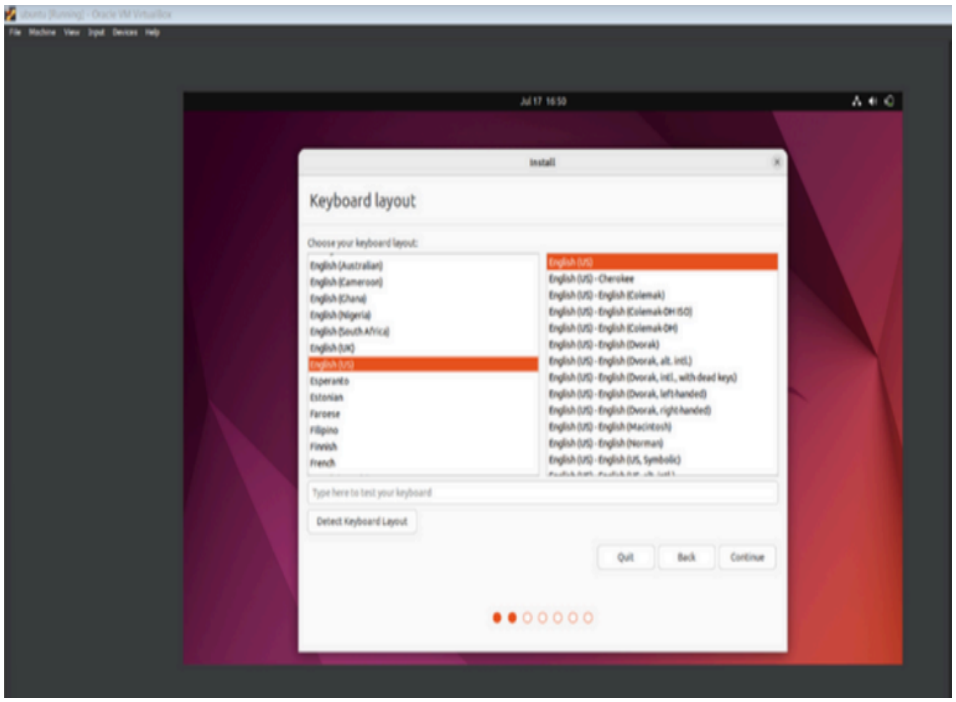
Step 5: Press Enter to install Ubuntu Server.



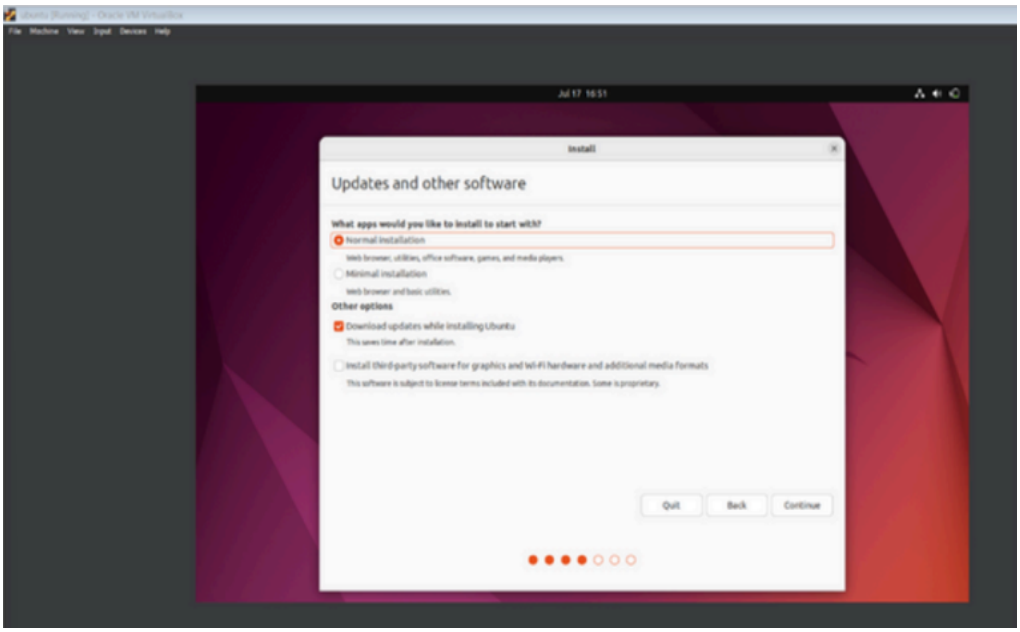
Step 6: Select the preferred language and click on Install Ubuntu.



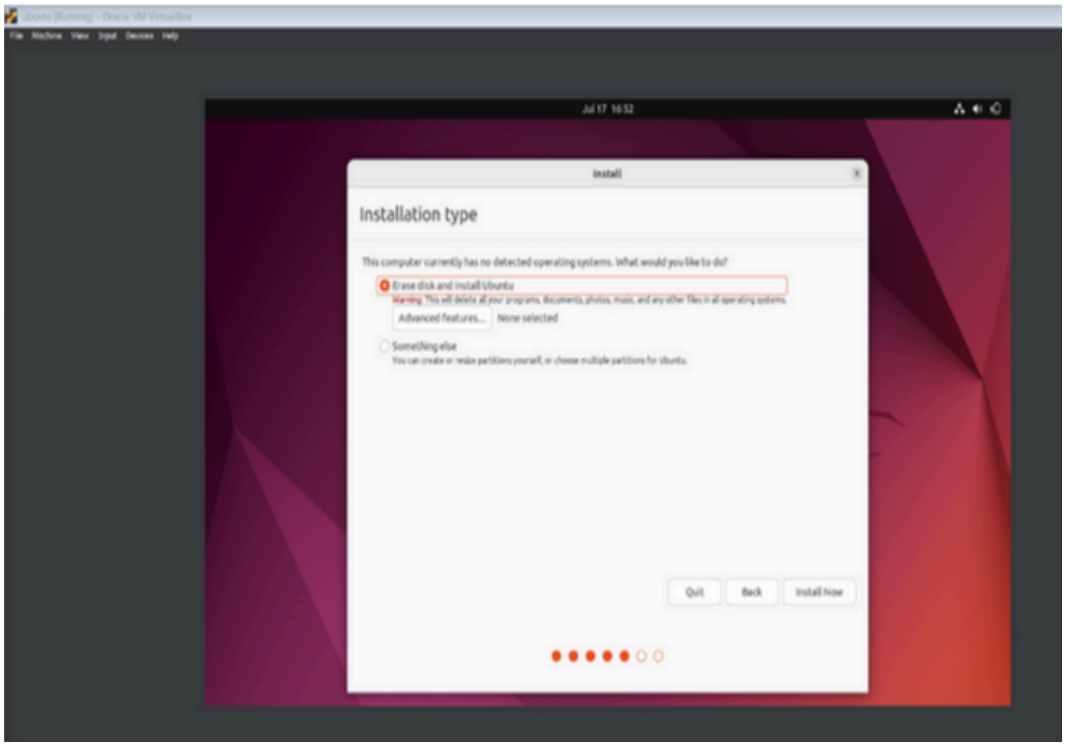
Step 7: Select the preferred keyboard layout and its variant, click on Continue once finished.



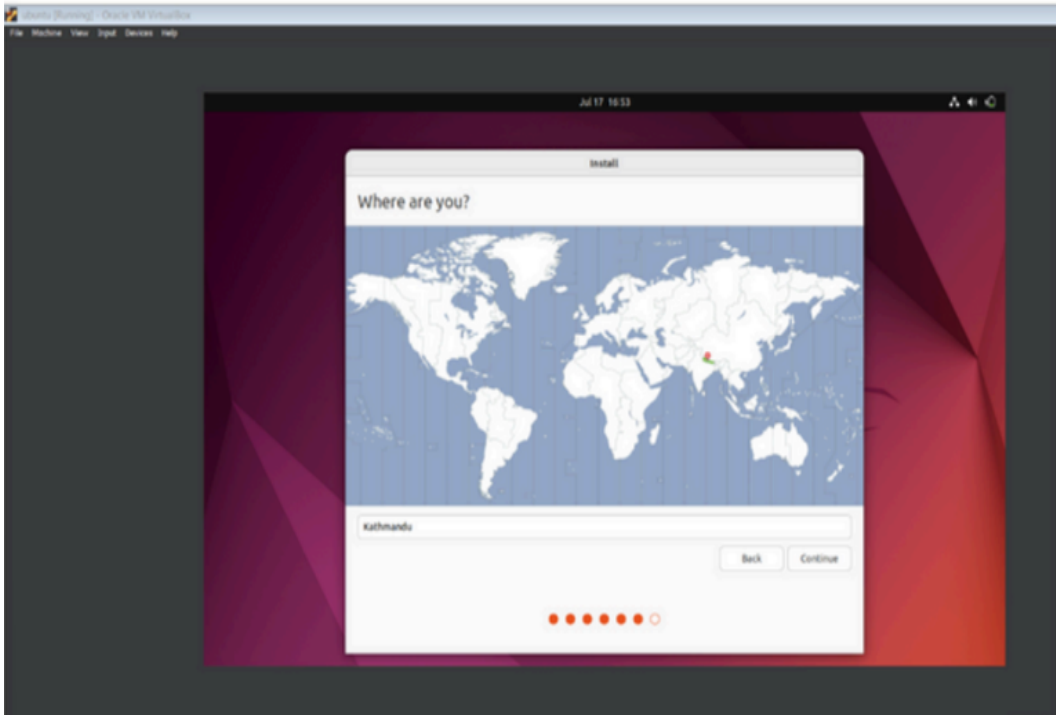
Step 8: It will prompt us to choose updates and other software to install alongside Ubuntu. We choose the options and click on Continue.



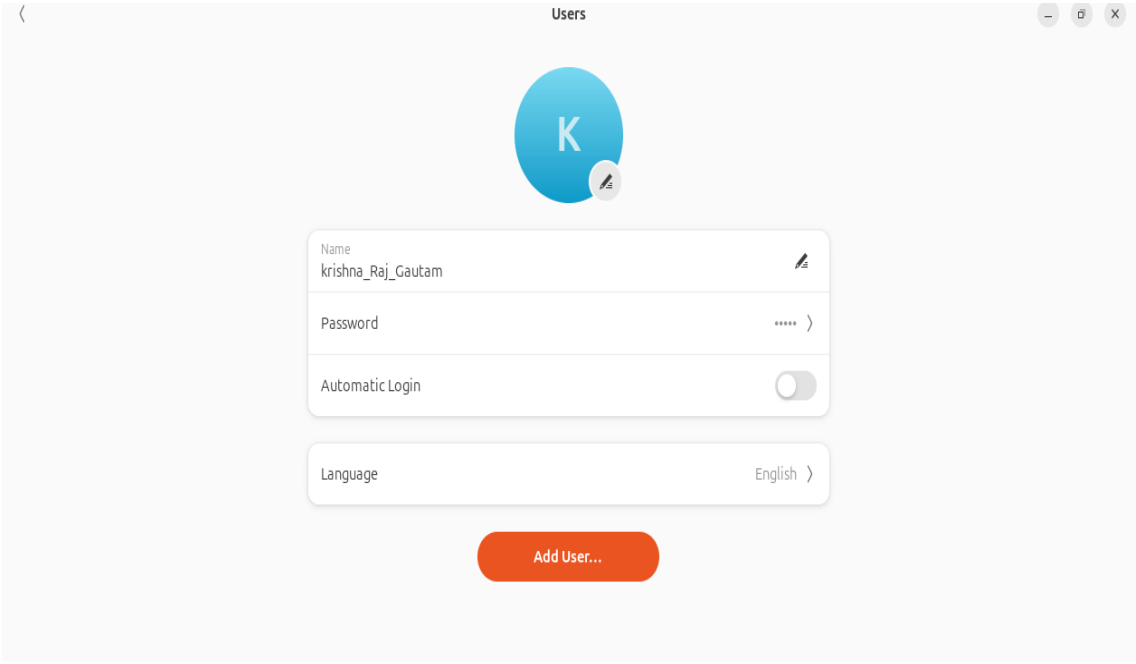
Step 9: We will be asked to choose the Installation type. Once selected we can press Enter. We will be informed about the partitions being formatted. If it's preferred, we press on Continue and proceed.



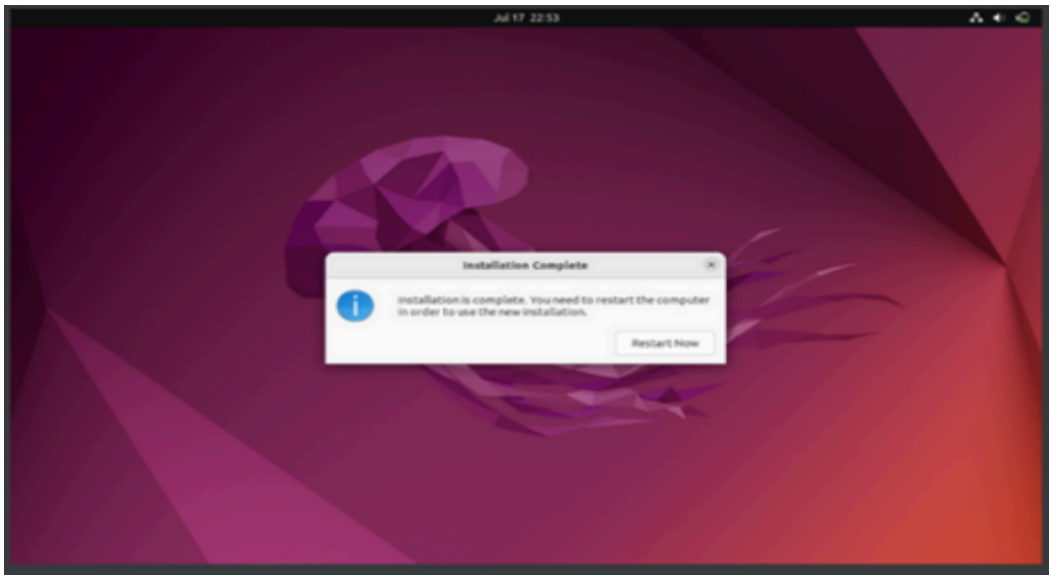
Step 10: Select the location and press Continue.



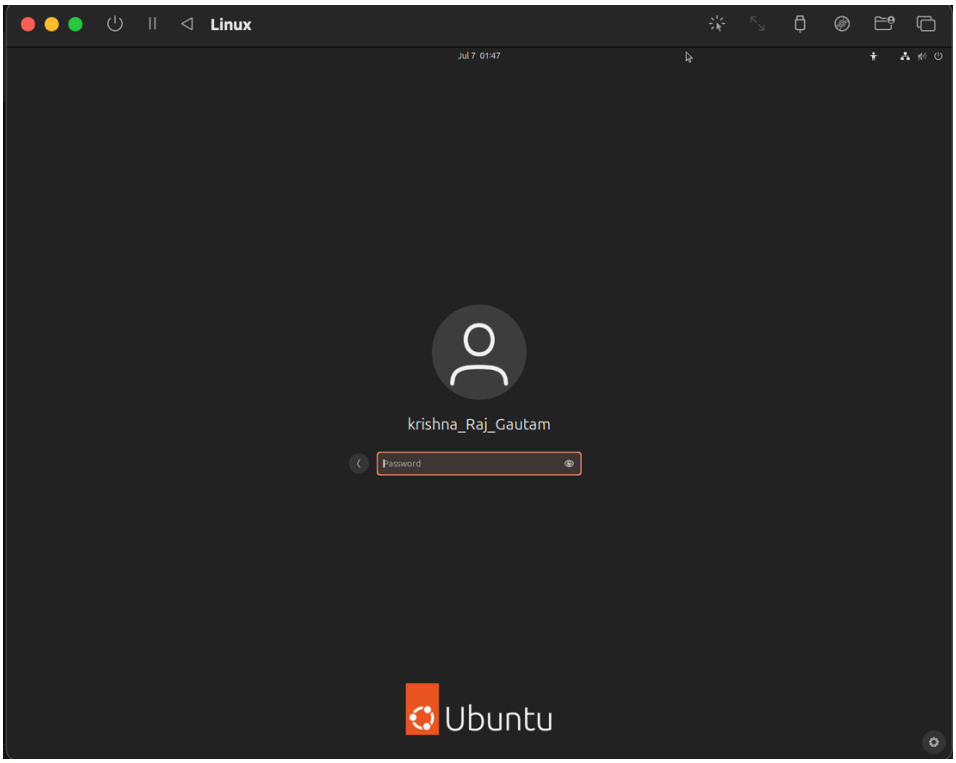
Step 11: Setup the profile for the user that includes username and password, once finished press Enter



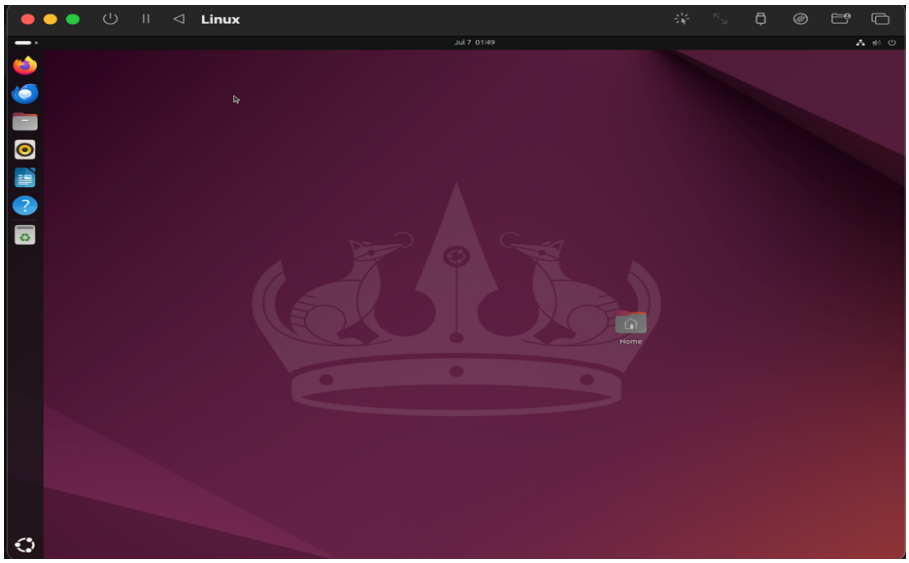
Step 12: Upon the completion of the installation process, the window on the right will prompt, Restart Now.



Step 13: Now enter the Login details on the screen and press Enter to login into the system.



Step 14: Now our system is ready and Ubuntu is installed completely.



Basic networking commands

PING

Ping (Packet Internet Groper) is a command-line utility used to test network connectivity between two hosts by sending ICMP echo request packets and waiting for ICMP echo reply to packets.

Syntax: ping <hostname/IP>

Uses: Used for checking whether any network is present and if a host is attainable.

```
Last login: Wed Jul  2 14:53:33 on ttys000
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % ping -c 4 localhost

PING localhost (127.0.0.1): 56 data bytes
64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.115 ms
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.161 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.164 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.149 ms

--- localhost ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.115/0.147/0.164/0.019 ms
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

IFCONFIG | GREP ETHER

This command is used to display the **MAC addresses** (also known as physical addresses) of all network interfaces on Unix-like systems, including macOS. The ifconfig command shows network configuration details, and grep ether filters out only the lines containing MAC addresses.

Syntax: ifconfig | grep ether

Use: To retrieve the **MAC addresses** of all available network interfaces on the system.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % ifconfig | grep ether

        ether aa:85:ba:63:26:33
        ether aa:85:ba:63:26:32
        ether aa:85:ba:63:26:12
        ether aa:85:ba:63:26:13
        ether 36:c6:3a:8f:d7:80
        ether 36:c6:3a:8f:d7:84
        ether 36:c6:3a:8f:d7:80
        ether 8e:a9:fc:b7:9b:75
        ether 2e:75:be:90:19:42
        ether 46:62:99:be:99:d4
        ether 46:62:99:be:99:d4
        ether 16:3e:f9:05:22:b7
        ether 62:3e:5f:92:d8:65
        ether 36:e0:93:62:6e:cf
        ether 62:3e:5f:92:d8:66
        ether ee:20:98:b7:f9:87
        ether 62:3e:5f:92:d8:67
        ether e2:01:46:1c:12:18
        ether 62:3e:5f:92:d8:64
```

IPCONFIG

This command is used on **macOS** to display the **IP address** assigned to the **Wi-Fi interface** (usually en0). It provides a quick and clean output of just the current IP address without extra details.

Syntax: ipconfig / ipconfig all

Uses: They are used to view and manage network configuration settings, renew DHCP leases, flush DNS cache, and configure network interfaces.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % ipconfig getifaddr en0
10.5.14.91
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % ipconfig ifcount
4
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

HOSTNAME

Hostname is a command-line utility used to display or set the name of the computer or device within a network. It can be accessed without using a particular IP address.

Syntax: hostname

Uses: Used to display the system name.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % hostname
MacBooks-MacBook-Air-Krishna-Raj-Gautam.local
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

NSLOOKUP

Nslookup is a command-line tool used to query DNS servers to obtain domain name or IP address mapping, DNS records (such as MX records), and other DNS-related information.

Syntax: nslookup <domain>

Uses: It is used for troubleshooting DNS issues, verifying DNS records, testing DNS configurations, and performing reverse DNS lookups.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % nslookup google.com
Server:         1.1.1.1
Address:        1.1.1.1#53

Non-authoritative answer:
Name:   google.com
Address: 172.217.27.174

macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

TRACERT

Traceroute (Unix/Linux) or tracert (Windows) is a command-line utility used to trace the route that packets take across an IP network to a specified destination, showing each hop and round-trip time (RTT).

Syntax: traceroute<hostname>

Uses: It is used to diagnose network connectivity issues, identify network latency or routing problems, and analyze the path packets take through the network.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % traceroute google.com

traceroute to google.com (142.250.206.110), 64 hops max, 40 byte packets
 1  192.168.18.1 (192.168.18.1)  4.509 ms  3.407 ms  3.359 ms
 2  10.17.0.1 (10.17.0.1)  7.153 ms  6.059 ms  6.069 ms
 3  10.250.0.116 (10.250.0.116)  6.048 ms  5.479 ms  4.637 ms
 4  36.253.0.57 (36.253.0.57)  9.364 ms  5.850 ms  5.764 ms
 5  116.68.209.163 (116.68.209.163)  22.933 ms  23.410 ms  25.052 ms
 6  * * *
 7  142.251.76.172 (142.251.76.172)  122.842 ms
    172.253.67.88 (172.253.67.88)  23.503 ms
    142.251.49.120 (142.251.49.120)  25.291 ms
 8  142.251.76.195 (142.251.76.195)  23.168 ms
    216.239.62.218 (216.239.62.218)  73.434 ms
    216.239.50.22 (216.239.50.22)  23.468 ms
 9  142.250.63.53 (142.250.63.53)  24.129 ms
    del11s20-in-f14.1e100.net (142.250.206.110)  23.211 ms  22.924 ms
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

NETSTAT

The netstat command displays active network connections, listening ports, and network statistics on the local computer.

Syntax: netstat -tulnp

Uses: To monitor current network connections and open ports.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % netstat -tulnp
netstat: option requires an argument -- p
Usage:  netstat [-AaLlnW] [-f address_family | -p protocol]
        netstat [-gilns] [-f address_family]
        netstat -i | -I interface [-w wait] [-abdgRtS]
        netstat -s [-s] [-f address_family | -p protocol] [-w wait]
        netstat -i | -I interface -s [-f address_family | -p protocol]
        netstat -m [-m]
        netstat -r [-Aaln] [-f address_family]
        netstat -rs [-s]

macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

NSLOOKUP

The nslookup command queries DNS servers to find the IP address associated with a domain name or vice versa.

Syntax: nslookup <domainname>

Uses**:** To resolve domain names to IP addresses.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % nslookup google.com
Server:         192.168.18.1
Address:        192.168.18.1#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.206.110

macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

ROUTE PRINT

The route print command displays the IP routing table on the local computer, showing how network traffic is directed through different gateways.

Syntax: route print

Uses**:** To view the current routing table and understand network paths.

```
krishnarajgautam@krishnarajgautam:~$ route print
Usage: route [-nNvee] [-FC] [<AF>]          List kernel routing tables
       route [-v] [-FC] {add|del|flush} ...  Modify routing table for AF.

       route {-h|--help} [<AF>]              Detailed usage syntax for specified AF.
       route {-V|--version}                  Display version/author and exit.

        -v, --verbose            be verbose
        -n, --numeric            don't resolve names
        -e, --extend             display other/more information
        -F, --fib                display Forwarding Information Base (default)
        -C, --cache              display routing cache instead of FIB

  <AF>=Use -4, -6, '-A <af>' or '--<af>'; default: inet
  List of possible address families (which support routing):
    inet (DARPA Internet) inet6 (IPv6) ax25 (AMPR AX.25)
    netrom (AMPR NET/ROM) rose (AMPR ROSE) ipx (Novell IPX)
    ddp (Appletalk DDP) x25 (CCITT X.25)
krishnarajgautam@krishnarajgautam:~$
```

NMAP

Nmap (Network Mapper) is a powerful tool used to discover hosts and services on a network by sending packets and analyzing the responses.

Syntax: nmap <target>

Uses: It helps identify open ports on a network device.

```
krishnarajgautam@krishnarajgautam:~$ nmap facebook.com
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-07 16:05 UTC
Nmap scan report for facebook.com (163.70.145.35)
Host is up (0.024s latency).
Other addresses for facebook.com (not scanned): 2a03:2880:f18a:188:face:b00c:0:25de
rDNS record for 163.70.145.35: edge-star-mini-shv-02-del2.facebook.com
Not shown: 998 filtered tcp ports (no-response)
PORT    STATE SERVICE
80/tcp  open  http
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 4.77 seconds
krishnarajgautam@krishnarajgautam:~$
```

SYSTEMINFO

Systeminfo is a command-line utility in Windows that displays detailed information about the computer's hardware and software configuration.

Syntax: uname -a

Uses: Shows detailed hardware and software information.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % uname -a
Darwin MacBooks-MacBook-Air-Krishna-Raj-Gautam.local 24.5.0 Darwin Kernel Versio
n 24.5.0: Tue Apr 22 19:48:46 PDT 2025; root:xnu-11417.121.6~2/RELEASE_ARM64_T81
03 arm64
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

CURL

CURL (Client URL) is a command-line tool used to transfer data to or from a server using various protocols such as HTTP, HTTPS, FTP, and more.

Syntax: curl <URL>

Uses: It is used for downloading or uploading files, testing APIs, automating web requests, and checking server responses.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % curl google.com
<HTML><HEAD><meta http-equiv="content-type" content="text/html;charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.com/">here</A>.
</BODY></HTML>
```

Conclusion

In this lab, I learned how to install Ubuntu using VirtualBox, which introduced me to the process of setting up and configuring virtual machines. I understood how virtualization tools like VirtualBox and VMWare help in running multiple operating systems on a single host. Along with the OS installation, I practiced several basic networking commands used in both Linux and Windows environments. Commands such as ping, ifconfig, hostname, traceroute, netstat, and curl were explored in detail. I learned their purpose, syntax, and how to use them effectively for network testing and troubleshooting. The comparison of Linux and Windows command syntax helped me recognize platform-specific differences. Each command was tested and analyzed to see how it works in real-time scenarios. Overall, this lab helped me gain essential knowledge in operating system installation and basic networking, which is fundamental for any computer network professional.

Lab Number: 04 **Date: July 4st, 2025**

Title: Static IP Setting on Windows/Linux Machine using GUI and Command Prompt

Theory

1. IP Address

An IP (Internet Protocol) address is a unique identifier assigned to every device connected to a network. It enables communication between devices by ensuring that data is sent and received at the correct destination. IP addresses are essential for identifying both the sender and receiver in a network communication process. These addresses can be assigned either manually or automatically, depending on the type of network setup. There are two main types of IP addresses: static and dynamic.

i. Static IP Address

A static IP address is a fixed address that does not change over time. It is manually configured on the device and remains the same even after rebooting or reconnecting to the network. Static IP addresses are often used for devices like servers, routers, and printers that require consistent access or need to be easily located on a network. While they offer reliability and easier access for remote connections, static IPs require manual management and can cause conflicts if duplicated.

ii. Dynamic IP Address

A dynamic IP address is automatically assigned to a device by a DHCP (Dynamic Host Configuration Protocol) server. Unlike static IPs, dynamic addresses can change each time a device connects to the network. This method is more efficient for managing large networks because it reduces manual configuration and prevents IP conflicts. Most home routers and internet service providers use dynamic IP assignment, making it a flexible and scalable solution for general-purpose networking.

2. Subnet Mask

A subnet mask is a numerical label used in IP networking to divide an IP address into network and host portions. It helps a system determine whether another IP address belongs to the same local network or a different one. Subnet masks are crucial in organizing and managing IP address allocation within larger networks. For example, the subnet mask 255.255.255.0 is commonly used in small networks to support up to 254 host addresses. By segmenting a network, subnetting improves routing efficiency and enhances security.

3. Default Gateway

The default gateway is a device, typically a router, that connects a local network to external networks such as the internet. When a device on the network wants to communicate with a system outside its own subnet, it sends the data to the default gateway. The gateway then forwards the data to the appropriate destination. Without a default gateway, devices would be limited to communicating only within the local network. It is essential for accessing websites, online applications, and any external services.

4. DNS Server

A DNS (Domain Name System) server is responsible for translating human-readable domain names into machine-readable IP addresses. When a user enters a website address such as www.google.com, the DNS server resolves this name into an IP address like 142.250.195.78, which the browser can use to connect to the site. DNS servers make it easier for users to access internet resources without needing to remember complex numerical IP addresses. This system simplifies browsing and is a core function of the internet's operation.

Static IP Configuration Steps

a. Static IP Configuration using GUI

i. First, Test the Device’s Connectivity & IP

Before changing the IP, verify that the device is connected to the network and receiving an IP from DHCP.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % system_profiler SPNetworkDataTyp
e | grep -A 5 "en0"

      BSD Device Name: en0
      IPv4 Addresses: 10.5.14.91
      IPv4:
          Additional Routes:
              Destination Address: 10.5.14.91
              Subnet Mask: 255.255.255.255
--
          Confirmed Interface Name: en0
          Interface Name: en0
          Network Signature: IPv4.Router=10.5.50.1;IPv4.RouterHardwareAddress=74:4d:
28:4a:d5:44
          Router: 10.5.50.1
          Subnet Masks: 255.255.0.0
      IPv6:
          Configuration Method: Automatic
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

Fig: Default network information

To ensure proper connectivity, the ping command is used to test communication between the device and external servers like facebook.com.

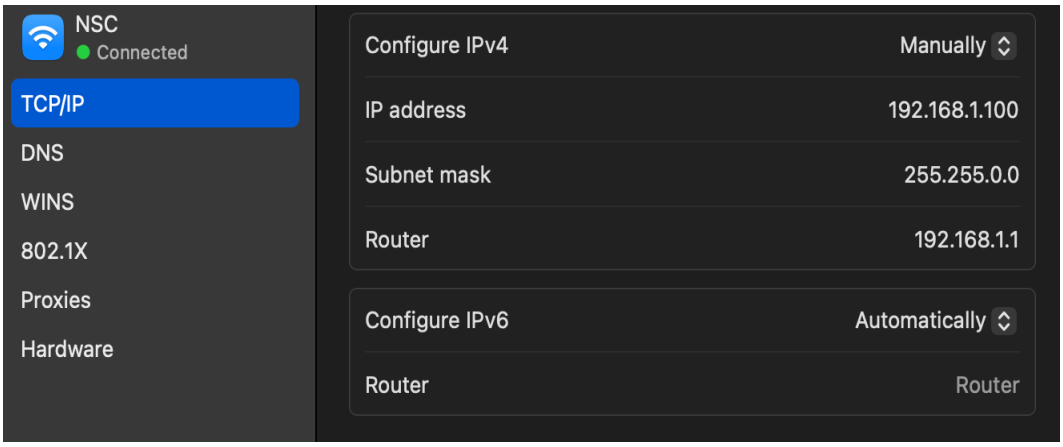
```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % ping facebook.com
PING facebook.com (57.144.146.1): 56 data bytes
64 bytes from 57.144.146.1: icmp_seq=0 ttl=53 time=19.235 ms
64 bytes from 57.144.146.1: icmp_seq=1 ttl=53 time=25.245 ms
64 bytes from 57.144.146.1: icmp_seq=2 ttl=53 time=27.879 ms
64 bytes from 57.144.146.1: icmp_seq=3 ttl=53 time=20.754 ms
64 bytes from 57.144.146.1: icmp_seq=4 ttl=53 time=20.118 ms
64 bytes from 57.144.146.1: icmp_seq=5 ttl=53 time=20.245 ms
64 bytes from 57.144.146.1: icmp_seq=6 ttl=53 time=19.318 ms
^C
--- facebook.com ping statistics ---
7 packets transmitted, 7 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 19.235/21.828/27.879/3.114 ms
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

Fig.: Network connectivity of facebook's server

ii. Change IP Statically

Open: Settings > Network > Wired/Wi-Fi > IPv4 Settings

- Change method from **Automatic (DHCP)** to **Manual**
- Enter: IP Address, Subnet Mask (Netmask), Gateway



- Click **Apply** and reconnect to the network

iii. Again, Test the Connectivity & IP

Verify if the static IP was applied and test the connection again.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % system_profiler SPNetworkDataTyp
e | grep -A 5 "en0"

      BSD Device Name: en0
      IPv4 Addresses: 192.168.1.100
      IPv4:
          Additional Routes:
              Destination Address: 192.168.1.100
              Subnet Mask: 255.255.255.255
--
          Confirmed Interface Name: en0
          Interface Name: en0
          Network Signature: IPv4.Router=192.168.1.1;IPv4.RouterHardwareAddress=74:4
d:28:4a:d5:44
          Router: 192.168.1.1
          Subnet Masks: 255.255.0.0
      IPv6:
          Configuration Method: Automatic
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % ping facebook.com
PING facebook.com (57.144.146.1): 56 data bytes
64 bytes from 57.144.146.1: icmp_seq=0 ttl=53 time=20.736 ms
64 bytes from 57.144.146.1: icmp_seq=1 ttl=53 time=28.909 ms
64 bytes from 57.144.146.1: icmp_seq=2 ttl=53 time=24.256 ms
64 bytes from 57.144.146.1: icmp_seq=3 ttl=53 time=28.305 ms
64 bytes from 57.144.146.1: icmp_seq=4 ttl=53 time=30.064 ms
64 bytes from 57.144.146.1: icmp_seq=5 ttl=53 time=20.166 ms
^C
--- facebook.com ping statistics ---
6 packets transmitted, 6 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 20.166/25.406/30.064/3.936 ms
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

Fig: Default network information after static IP configuration and Network connectivity of facebook's server

b. Static IP Configuration using Command Prompt

i. First, Test the Device's Connectivity & IP

Begin by verifying network connectivity and IP assignment. Use ipconfig getifaddr en0 to check the current IP address, and ping 8.8.8.8 to confirm internet access. This ensures the device is online and receiving an IP address.

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % system_profiler SPNetworkDataTyp
e | grep -A 5 "en0"
      BSD Device Name: en0
      IPv4 Addresses: 10.5.14.91
      IPv4:
          Additional Routes:
              Destination Address: 10.5.14.91
              Subnet Mask: 255.255.255.255
--
          Confirmed Interface Name: en0
          Interface Name: en0
          Network Signature: IPv4.Router=10.5.50.1;IPv4.RouterHardwareAddress=74:4d:
28:4a:d5:44
          Router: 10.5.50.1
          Subnet Masks: 255.255.0.0
      IPv6:
          Configuration Method: Automatic
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

Fig: Default network information

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: icmp_seq=0 ttl=115 time=26.848 ms
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=27.148 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=28.346 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=27.462 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=115 time=19.485 ms
^C
--- 8.8.8.8 ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 19.485/25.858/28.346/3.226 ms
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

Fig.: Network connectivity of google's server

ii. Assign a Static IP Address

Next, configure a static IP address using the networksetup command. Identify your network service name (e.g., "Wi-Fi") using networksetup -listallnetworkservices. Then assign the desired IP address, subnet mask, and gateway using -setmanual.

Command :

sudo networksetup -setmanual "Wi-Fi" 192.168.1.100 255.255.255.0 192.168.1.1

To configure DNS servers:

sudo networksetup -setdnsservers "Wi-Fi" 8.8.8.8 1.1.1.1

```
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ % system_profiler SPNetworkDataTyp
e | grep -A 5 "en0"
ping 8.8.8.8
networksetup -getinfo "Wi-Fi"
      BSD Device Name: en0
      IPv4 Addresses: 192.168.1.100
      IPv4:
          Additional Routes:
              Destination Address: 192.168.1.100
              Subnet Mask: 255.255.255.255
--
          Confirmed Interface Name: en0
          Interface Name: en0
          Network Signature: IPv4.Router=192.168.1.1;IPv4.RouterHardwareAddress=74:4
d:28:4a:d5:44
          Router: 192.168.1.1
          Subnet Masks: 255.255.255.0
      IPv6:
          Configuration Method: Automatic
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: icmp_seq=0 ttl=115 time=21.876 ms
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=22.095 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=21.473 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=24.570 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=115 time=29.287 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=115 time=20.546 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=115 time=21.597 ms
^C
--- 8.8.8.8 ping statistics ---
7 packets transmitted, 7 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 20.546/23.063/29.287/2.787 ms
Manual Configuration
IP address: 192.168.1.100
Subnet mask: 255.255.255.0
Router: 192.168.1.1
IPv6: Automatic
IPv6 IP address: none
IPv6 Router: none
Wi-Fi ID: 60:3e:5f:29:95:81
macbook@MacBooks-MacBook-Air-Krishna-Raj-Gautam ~ %
```

Conclusion

In this lab, I practiced configuring a static IP address using both the GUI and the terminal. I first tested the device's default network connectivity and IP address. Then, I manually set a static IP through the system's GUI and verified the changes using terminal commands. I also used the terminal to configure the IP settings manually by editing the appropriate configuration files and applying the changes. After completing the configuration, I tested connectivity to ensure it was set up correctly. Including syntax commands helped me understand how IP settings were applied at the system level. This lab enhanced my practical understanding of IP management and the importance of stable network addressing.