Experiment No. 2

Aim: To create and run virtual machines on Hosted **Hypervisor** like **KVM** (Kernel Based Virtual Machine) and **Virtual Box.**

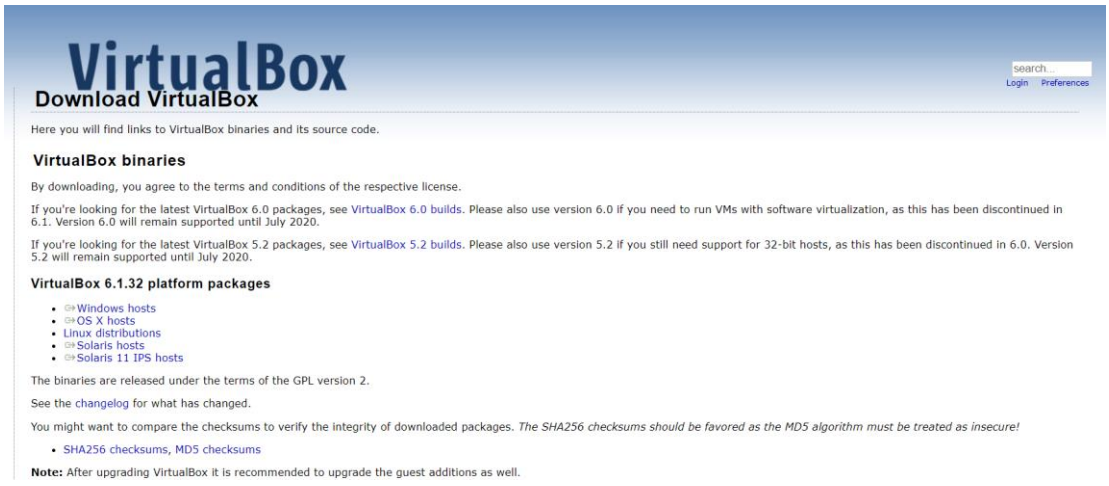
Requirements: Windows/Linux O.S for **Virtual Box** and Linux O.S for **KVM** with compatible hardware.

Theory:

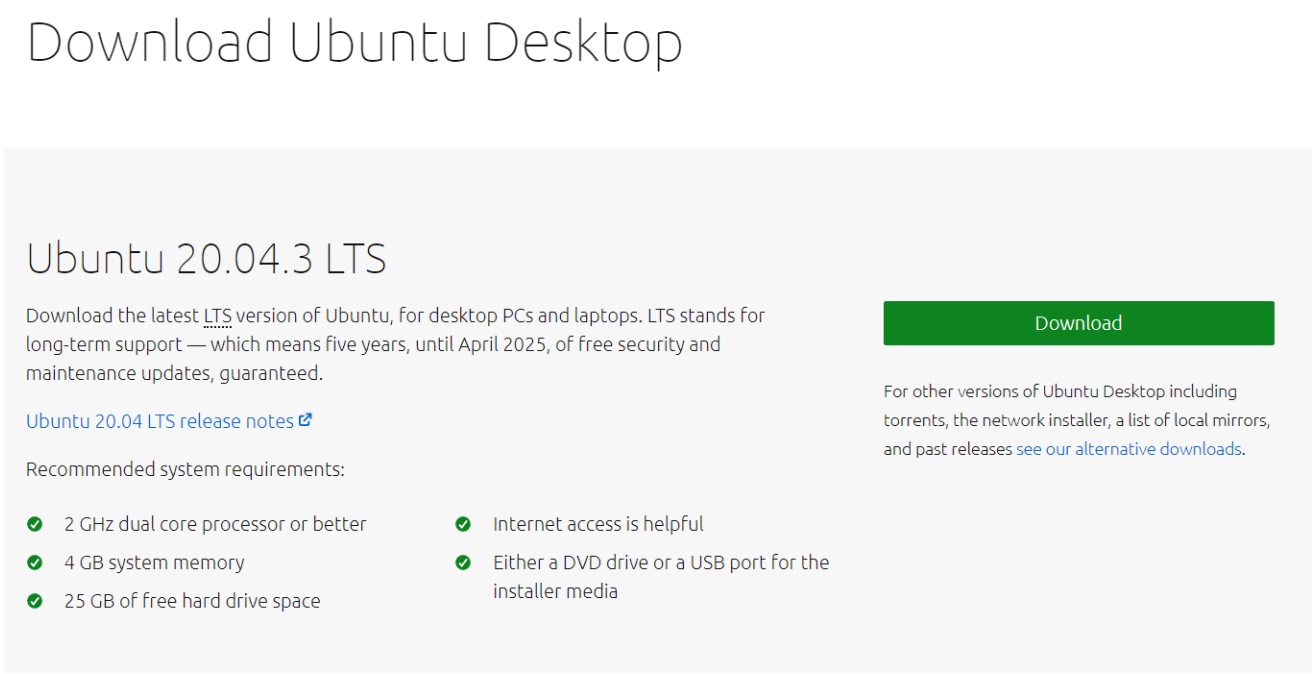
Hypervisor: A **hypervisor**, also known as a virtual machine monitor or **VMM**, is software that creates and runs virtual machines (VMs). A hypervisor allows one host computer to support multiple guest VMs by virtually sharing its resources, such as memory and processing. There are two main hypervisor types, referred to as "Type 1" (or "bare metal") and "Type 2" (or "hosted"). A **type 1 hypervisor** acts like a lightweight operating system and runs directly on the host's hardware, while a **type 2 hypervisor** runs as a software layer on an operating system, like other computer programs. **Oracle Virtual Box** is type 2 hypervisor and **KVM** is type 2 hypervisor.

1. Implementing virtual machine on Oracle Virtual Box:

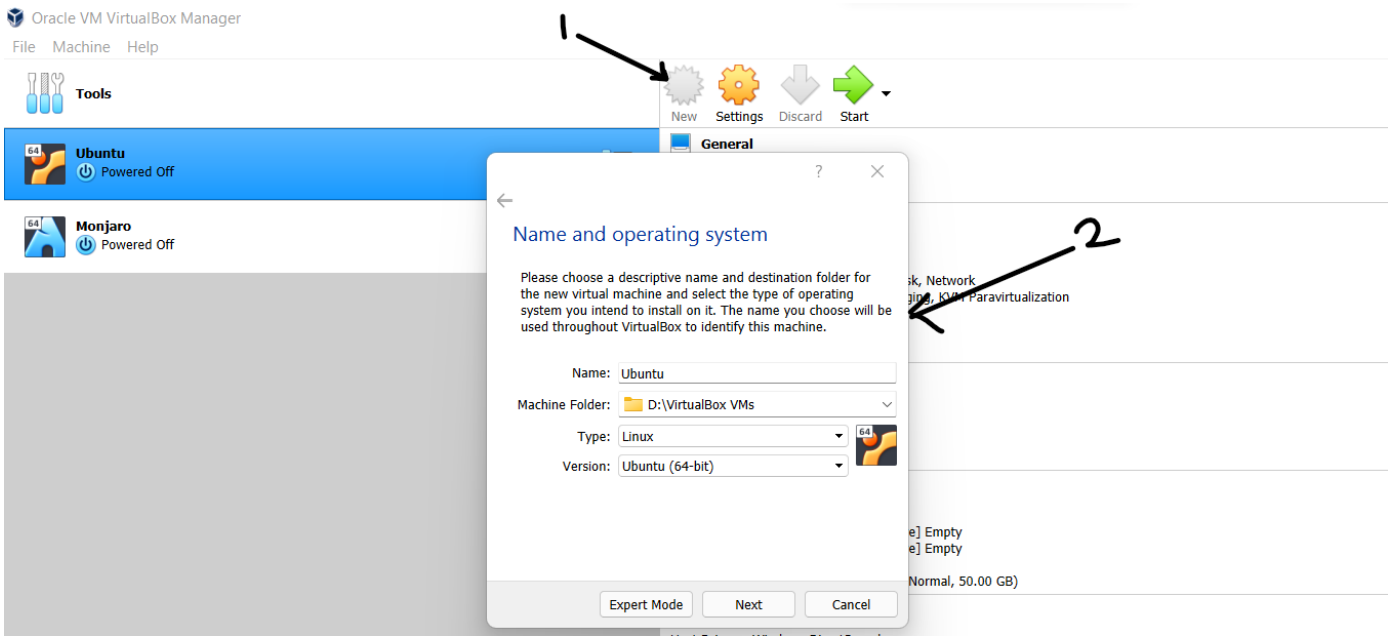
Step 1: Download Oracle Virtual Box compatible for your O.S from https://www.virtualbox.org/wiki/Downloads



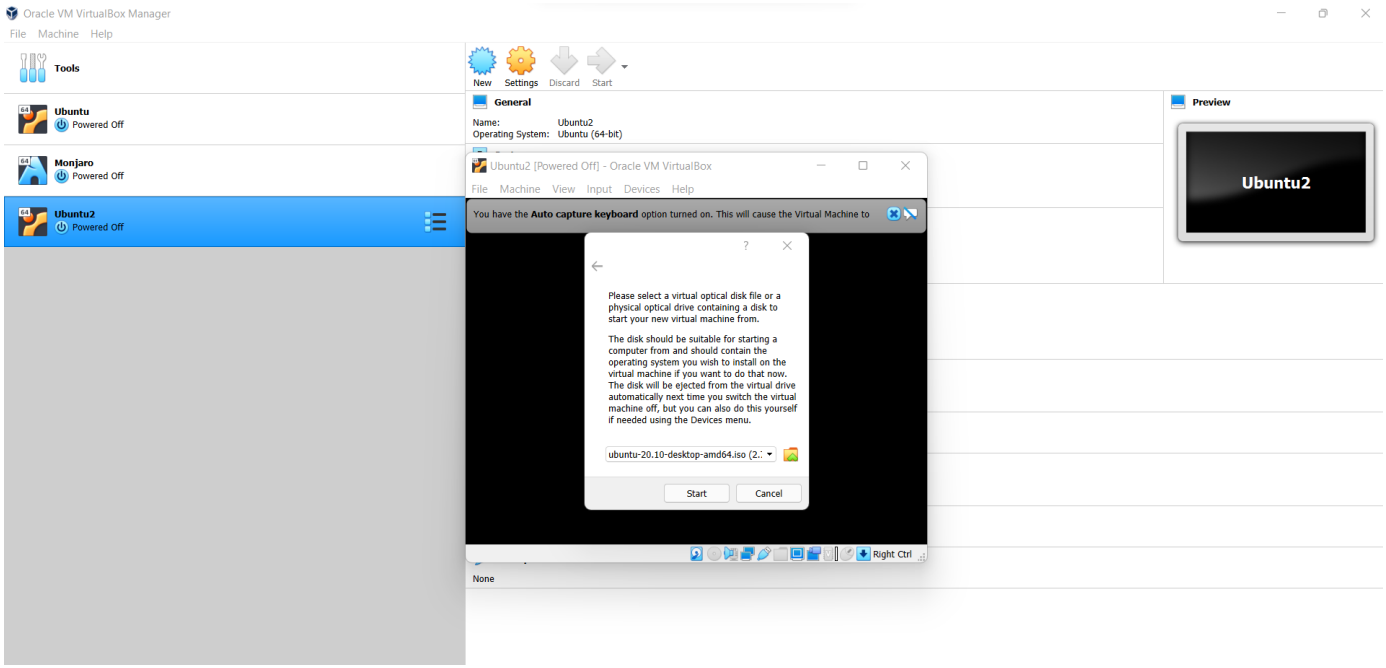
Step 2: Download Disc Image of Ubuntu (Linux) from https://ubuntu.com/download/desktop



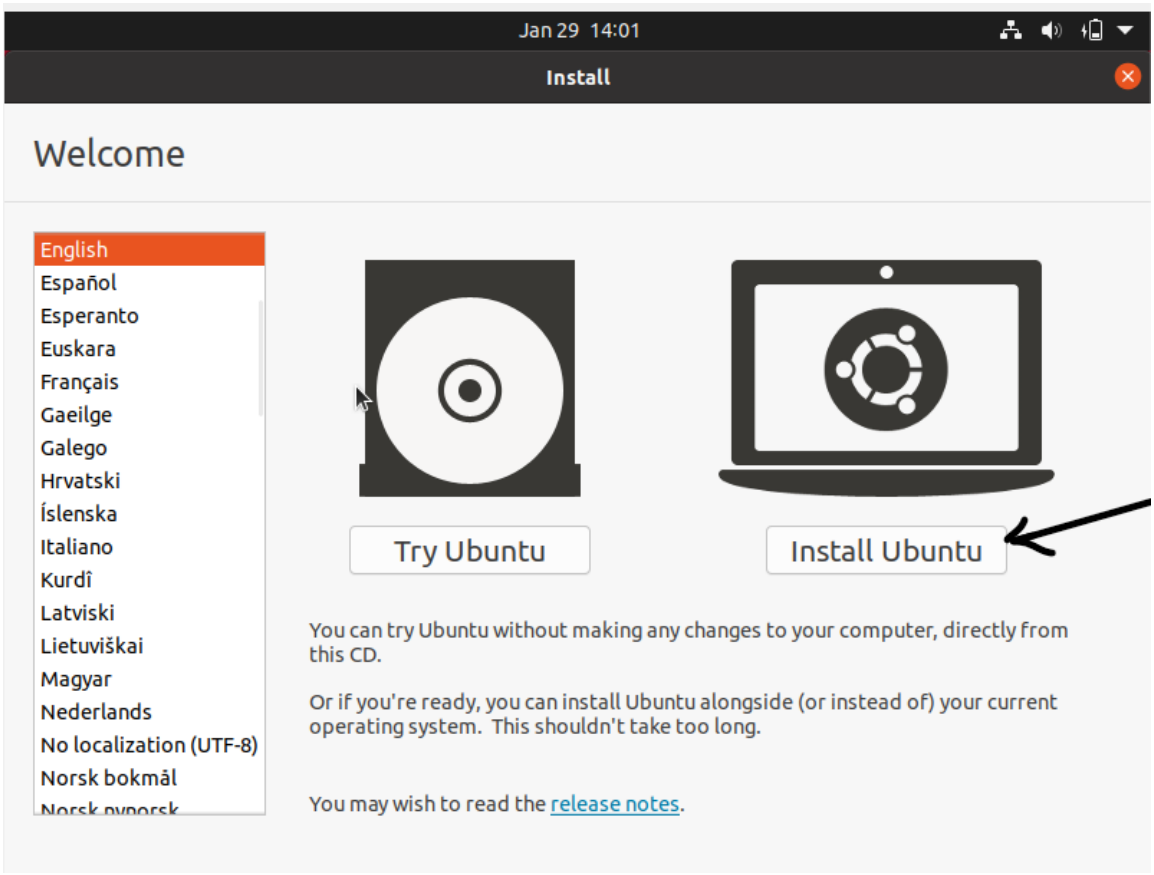
Step 3: Open Virtual Box and click on New and give a desired name and select type Linux and version Ubuntu and Click on next setup hardware specifications.



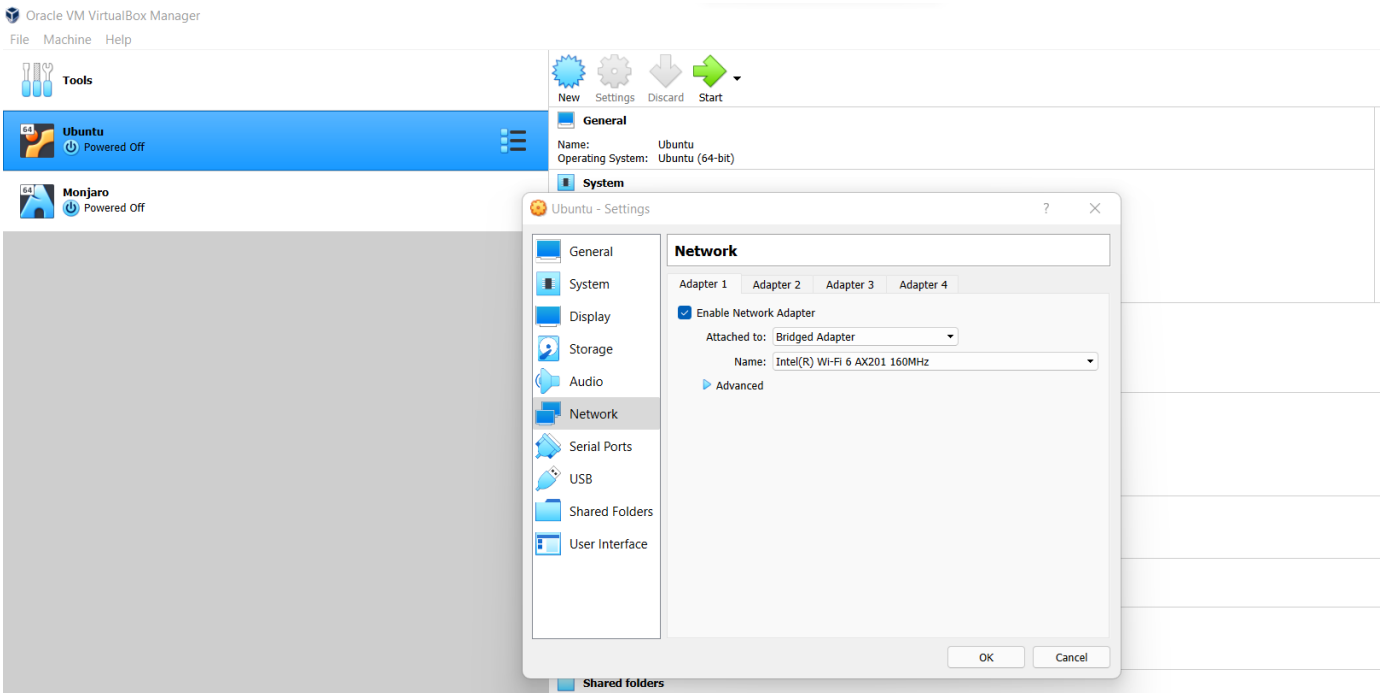
Step 4: After completion of **Step 3** click on Start button, after clicking on Start it will ask for disc image (.iso) select the Ubuntu disc image downloaded in **Step 2** and click on start.



Step 5: Select Safe Graphical Install if you graphical installation then it will show you option Try Ubuntu and Install Ubuntu click on Install Ubuntu and wait for Installation process to complete.

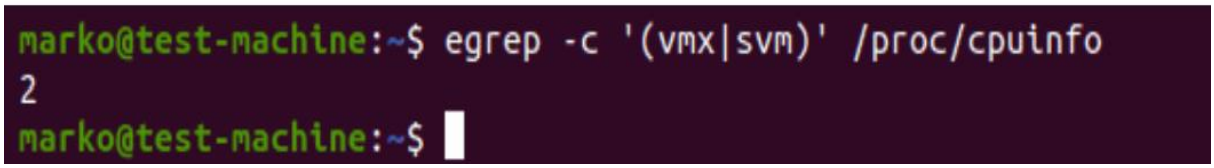


Step 6: After completion of installation enable change network mode to Bridge Adapter for Ubuntu virtual machine which is by default set to NAT by **Oracle Virtual Box**.



2. <u>Implementing virtual machine on KVM</u>:

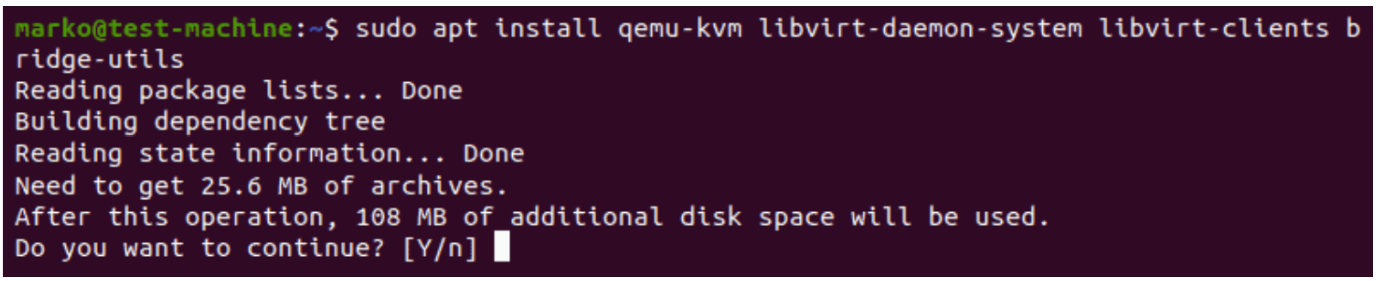
Step 1: Before you begin with installing KVM, check if your CPU supports hardware virtualization. Command: **egrep -c '(vmx|svm)' /proc/cpuinfo**



If the command returns a value of **0**, your processor is not capable of running KVM. On the other hand, any other number means you can proceed with the installation.

Step 2: Install essential KVM packages with the following command:

sudo apt install qemu-kvm libvirt-daemon-system libvirt-clients bridge-utils



Step 3: Only members of the libvirt and kvm user groups can run virtual machines. Add a user to the libvirt and kvm group by typing commands:

1. **sudo adduser 'username' libvirt**

2. **sudo adduser 'username' kvm**

```
marko@test-machine:~$ sudo adduser 'marko' libvirt
Adding user `marko' to group `libvirt' ...
Adding user marko to group libvirt
Done.
marko@test-machine:~$
```

```
marko@test-machine:~$ sudo adduser 'marko' kvm
Adding user `marko' to group `kvm' ...
Adding user marko to group kvm
Done.
marko@test-machine:~$
```

Note: If user is not created you can create user using following command: **sudo adduser 'username'**

Step 4: Confirm the installation was successful by using the **virsh** command: **virsh list –all**

```
marko@test-machine:~$ virsh list --all
 Id    Name    State
----------------------

marko@test-machine:~$
```

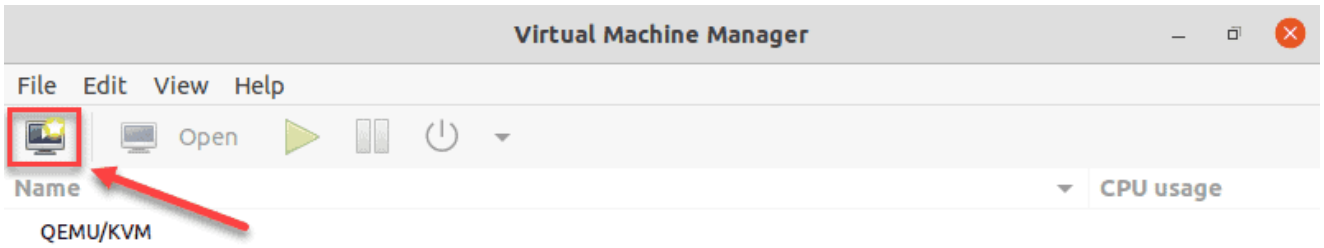
Step 5: use the systemctl command to check the status of libvirtd: **sudo systemctl status libvirtd**

```
marko@test-machine:~$ sudo systemctl status libvirtd
● libvirtd.service - Virtualization daemon
     Loaded: loaded (/lib/systemd/system/libvirtd.service; enabled; vendor preset: ena>
     Active: active (running) since Fri 2020-11-27 09:36:04 EST; 44min ago
TriggeredBy: ● libvirtd-ro.socket
             ● libvirtd-admin.socket
             ● libvirtd.socket
       Docs: man:libvirtd(8)
             https://libvirt.org
   Main PID: 18166 (libvirtd)
      Tasks: 19 (limit: 32768)
     Memory: 14.2M
     CGroup: /system.slice/libvirtd.service
             ├─18166 /usr/sbin/libvirtd
             ├─18301 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.co>
             └─18302 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.co>
```

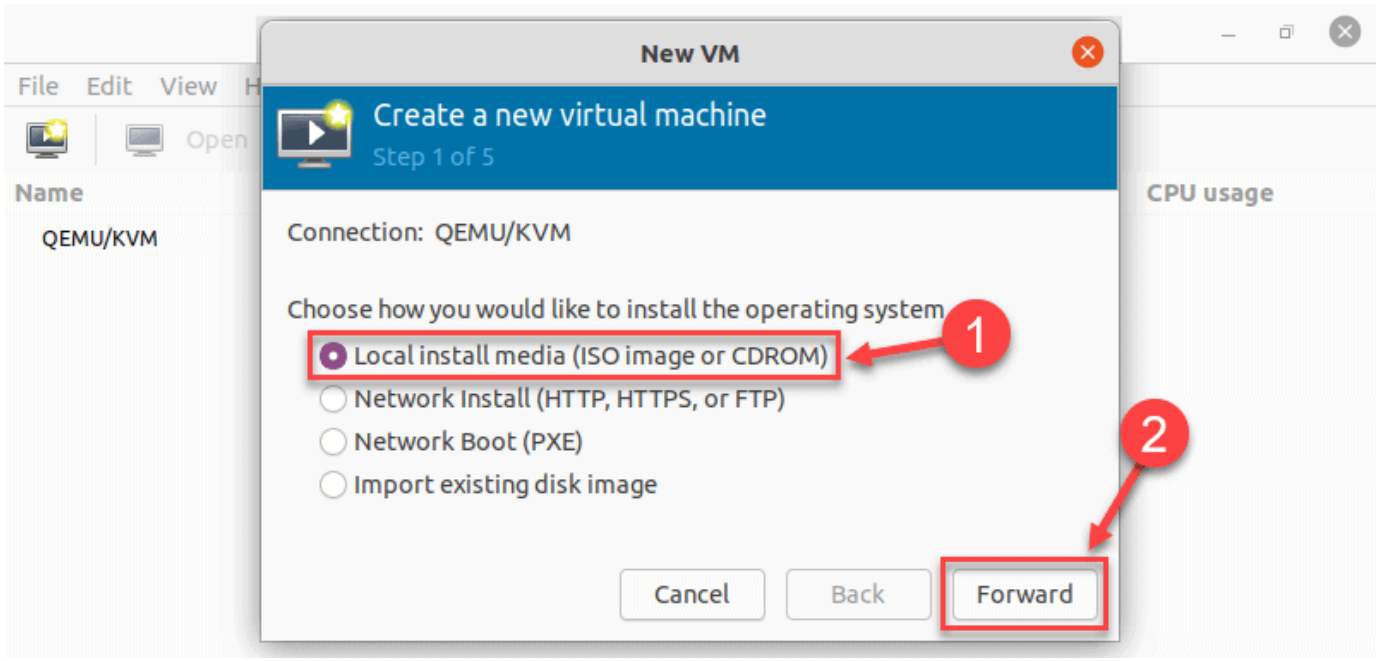
Step 6: Install virt-manager, a tool for creating and managing VMs: **sudo apt install virt-manager**

```
marko@test-machine:~$ sudo apt install virt-manager
[sudo] password for marko:
Reading package lists... Done
Building dependency tree
Reading state information... Done
0 upgraded, 33 newly installed, 0 to remove and 74 not upgraded.
Need to get 7,987 kB of archives.
After this operation, 62.5 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

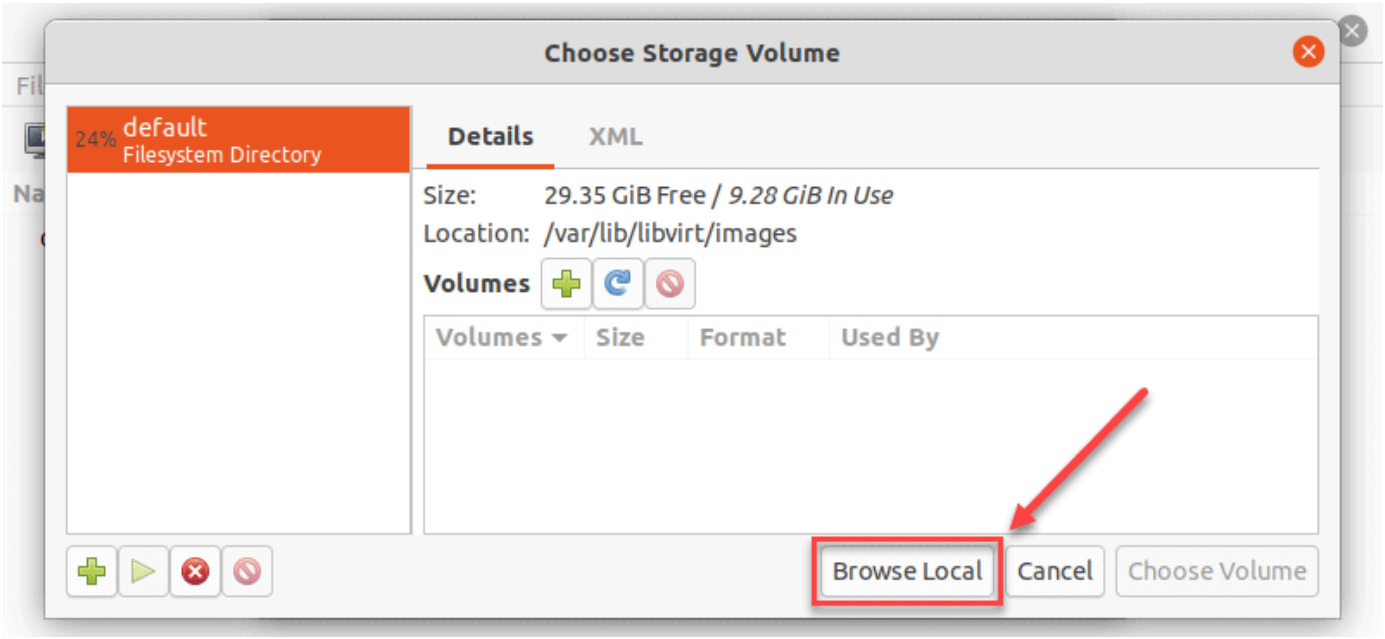
Step 7: Start virt-manager with: **sudo virt-manager** and then click the computer icon in the upper-left corner.



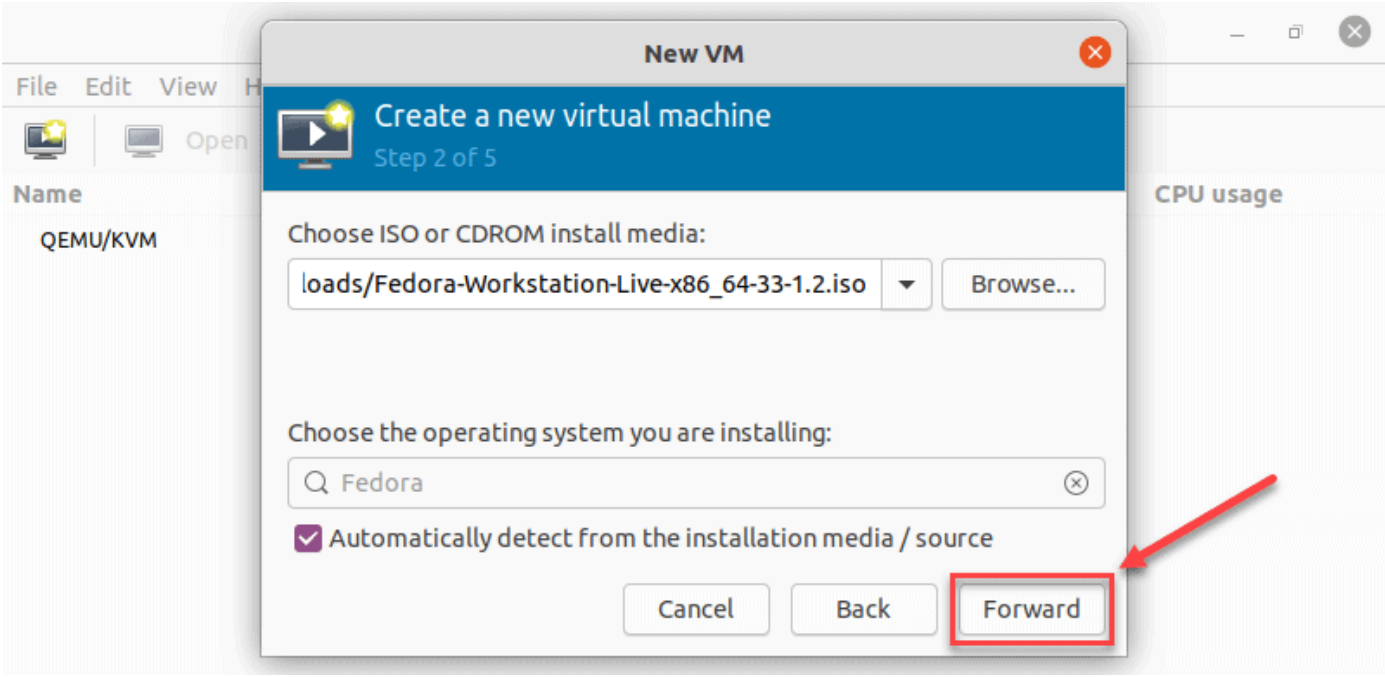
Step 8: In the dialogue box that opens, select the option to install the VM using an ISO image. Then click Forward.



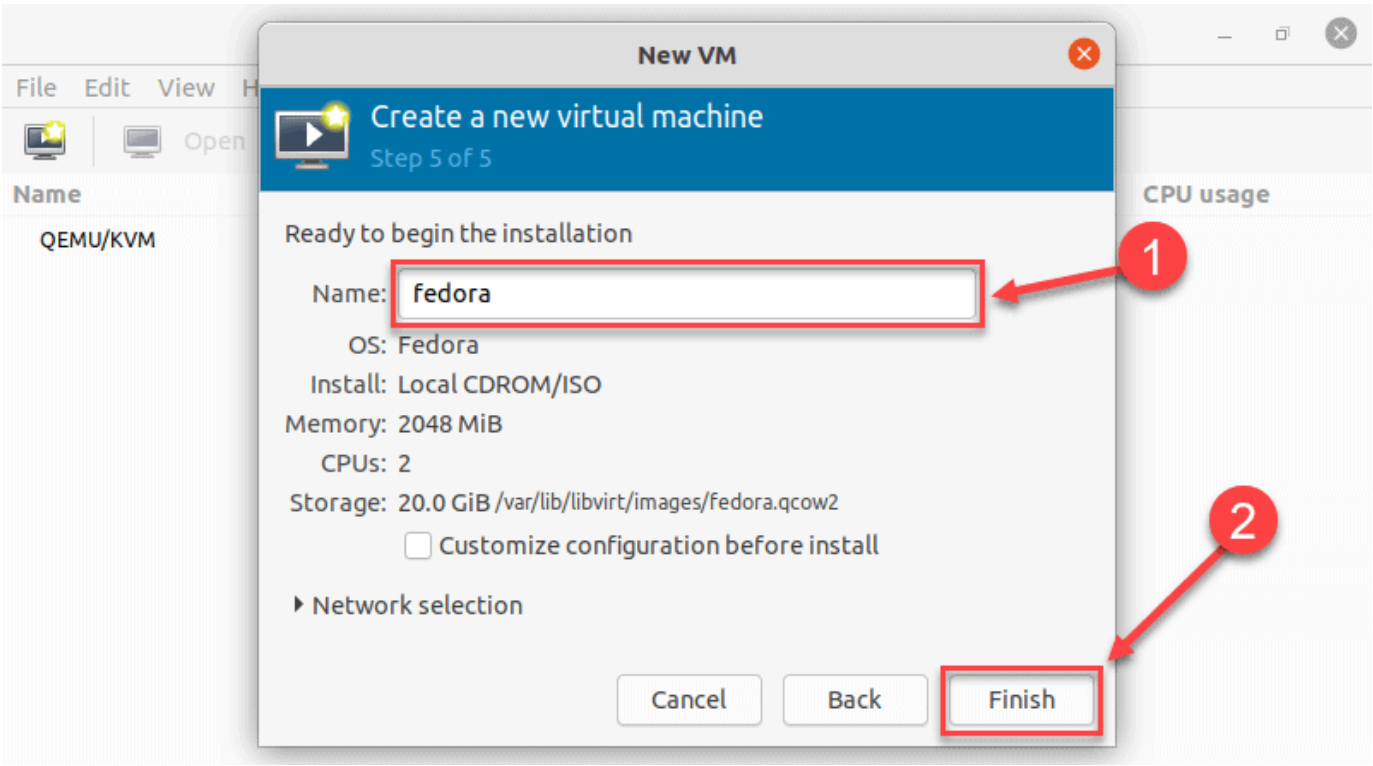
Step 8: Click Browse Local and navigate to the path where you stored the ISO you wish to install.



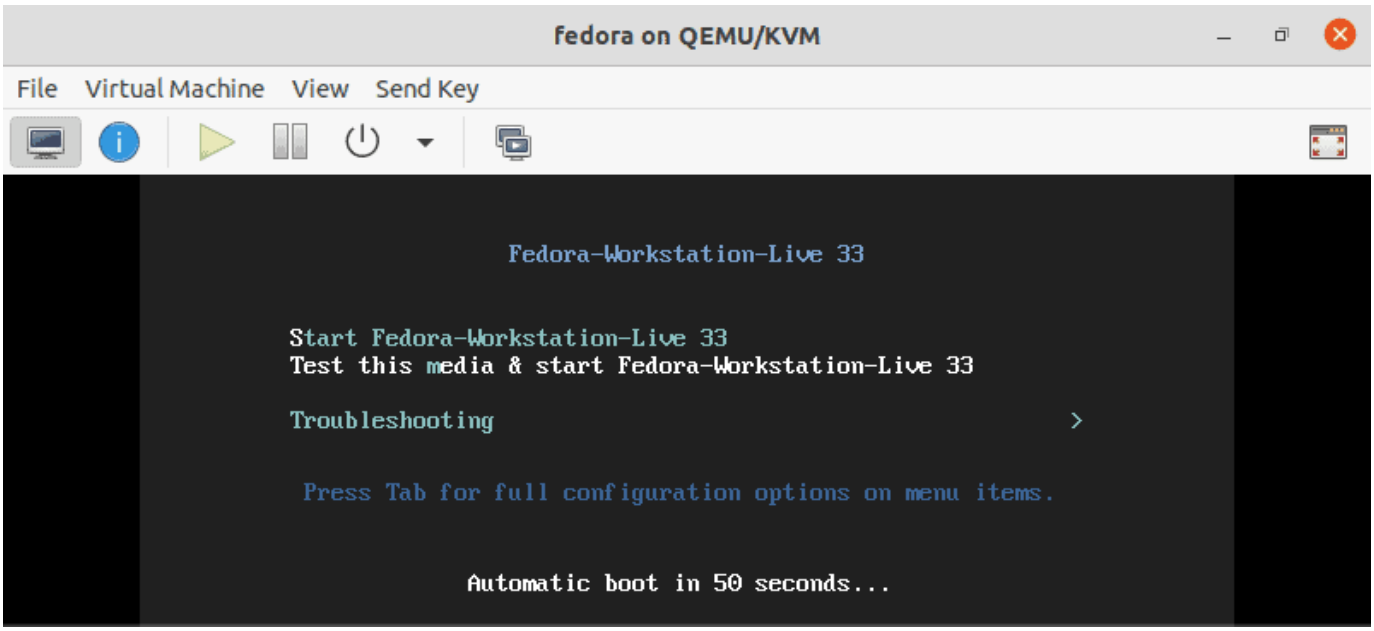
Step 9: Choose ISO file of virtual system (Here, I used Fedora because it is light weighted) and click on forward and then select hardware specification.



Step 10: Specify the name for your VM and click Finish to complete the setup.



Step 11: The VM starts automatically, prompting you to start installing the OS that's on the ISO file, now you only need to follow installation process of OS.



<u>Conclusion</u>: In this practical we successfully created and run **VM** on Hosted **Hypervisor** type 1 **Virtual Box** and type 2 **KVM.**