

Vivekanand Education Society's Institute of Technology
Department of Computer Engineering



Subject: Cloud Computing Lab

Class :- (CMPN) D12

Semester: - VI

Div.:- A

Roll No: 56	Name: Meet D. Patel		
Exp No.: 2	Title: Running Virtual Machines on Hosted Hypervisor like Virtual Box and KVM.		
DOP:	29/01/2022	DOS:	04/02/2022
GRADE:		LAB OUTCOMES: LO1	SIGNATURE:

Cloud Computing Lab Experiment No. 2

Aim: Creating and running virtual machines on Hosted Hypervisor like Virtualbox and KVM

Theory:

Virtualization, in computing, refers to act of creating a virtual (rather than actual) version of ~~some~~ computer hardware platform, OS, storage device or computer network resources. In cloud computing, virtualization can be achieved using Hypervisor (a computer hardware platform virtualization software that allows several OS to share a single hardware host^{OS have}). Each host's processor, memory & resources to it). There are various hypervisor products are available in market like - ~~Linux~~ Citrix Xen, KVM, VMware ESXI, Open Stack, Eucalyptus, etc. Server virtualization is the partitioning of a physical server into smaller virtual servers to help maximize your server resources. In server virtualization the resources of the server itself are hidden (masked) from users, and software is used to divide the physical server into multiple virtual environments, called virtual or private servers.

Server virtualization is masking of server resources, including the number and identity of individual physical servers, processors, and operating systems, from server users. The server administrator uses a software application to divide one physical server into multiple isolated virtual environments or virtual private servers.

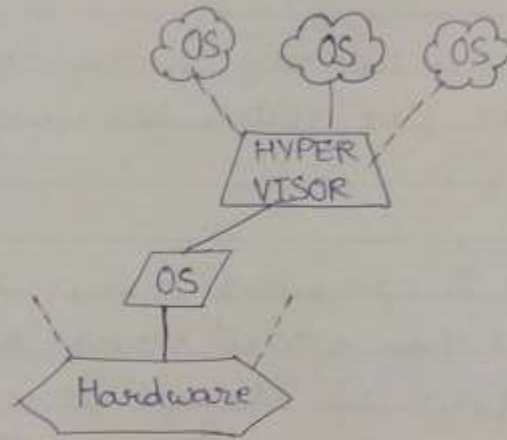


Diagram for Hosted Hypervisor

Meet D. Patel 56
D12A

Step 2: Once the installation is over, create a new VM by clicking "New" button which runs host OS. Allows you to specify resources for guest VM like VCPU, Memory, Disk Size, network configuration, etc.

Step 3: Once VM is created, start the installation by clicking "Start" button and specify path of iso file through which installation will be done.

Step 4: Once OS is installed it can be accessed inside the VMbox.

Step 5: To connect OS to the network change network Mode to Bridge Adapter.

→ Steps to create & run VM in KVM is as follows:

Step 1: Check whether CPU has hardware virtualization support.

Step 2: Install KVM and supporting packages

Step 3: Create user in libvirt group.

Step 4: Check whether everything is working correctly.

Step 5: Open VMM and create Virtual Machine.

Step 6: Create and run virtual machines.

Conclusion:

We can run multiple desktop OS on single machine. It is only feasible if base machine has high hardware configuration otherwise there is no guarantee for Guest OS of getting resources at critical time.

1) Hosted Virtualization on Oracle Virtual Box Hypervisor

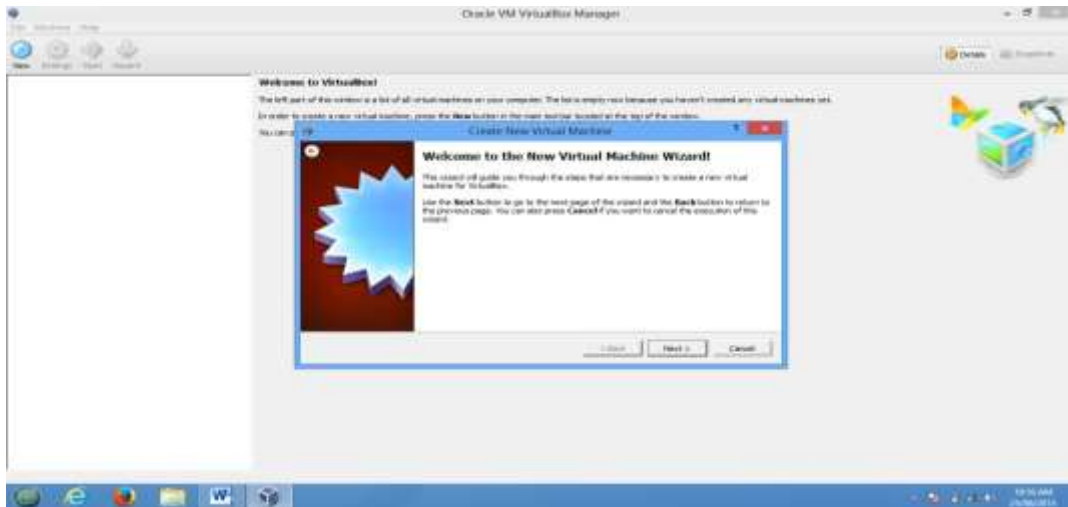
Step 1: Download Oracle Virtual box from <https://www.virtualbox.org/wiki/Downloads>



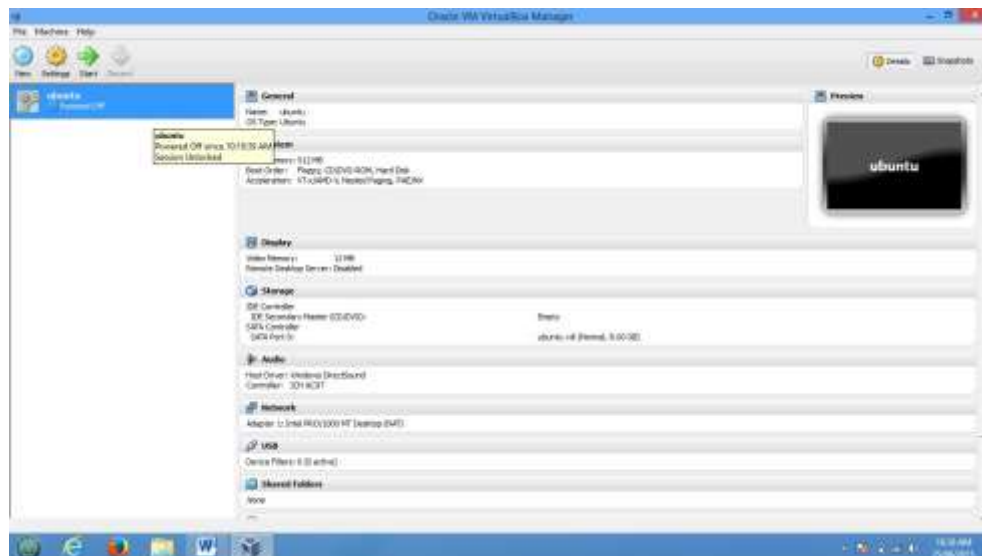
Step 2: Install it in Windows, Once the installation has done open it.



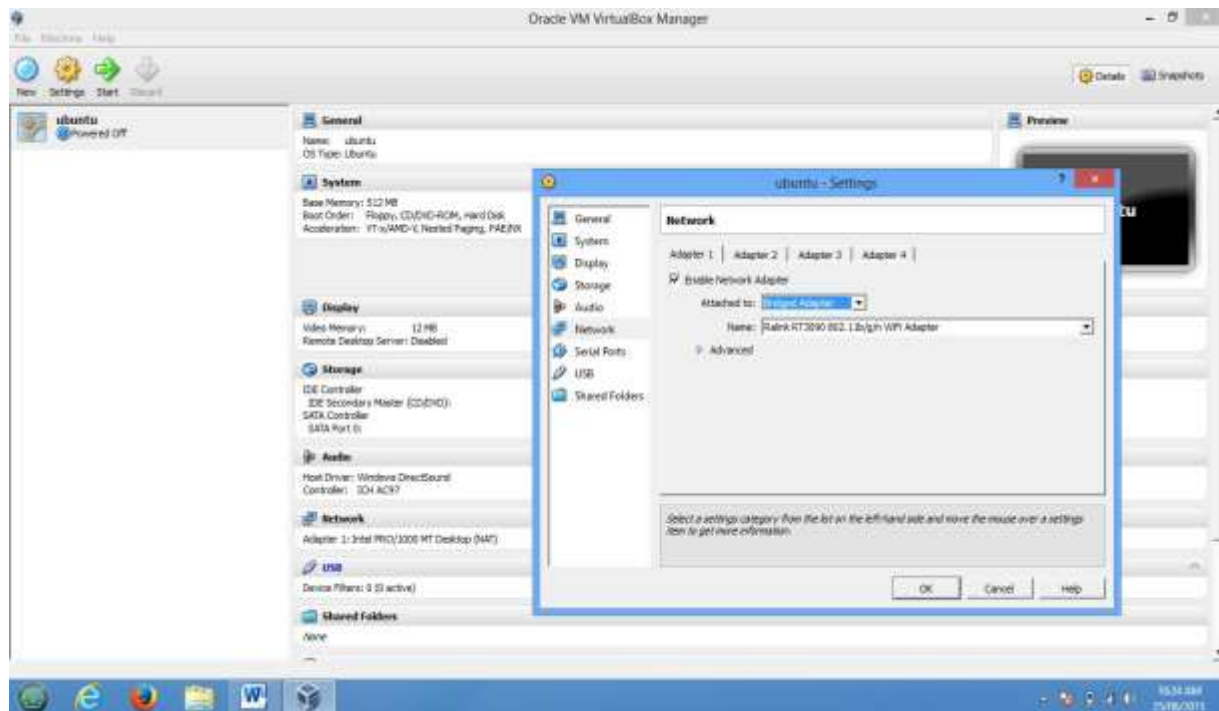
Step 3:-Create Virtual Machine by clicking on New



Step 4:- Specify RAM Size, HDD Size, and Network Configuration and Finish the wizard



Step 5:- To Select the media for installation Click on start and browse for iso file



2) Hosted Virtualization on KVM Hypervisor

The Steps to Create and run Virtual machines in KVM are as follows

1) Check whether CPU has hardware virtualization support.

KVM only works if your CPU has hardware virtualization support – either Intel VT-x or AMD-V. To determine whether your CPU includes these features, run the following command:

```
#sudo grep -c "svm|vmx" /proc/cpuinfo
```

```
root@ubuntu:/home/tsec# sudo grep -c "svm|vmx" /proc/cpuinfo
3
root@ubuntu:/home/tsec#
```

A 0 indicates that your CPU doesn't support hardware virtualization, while a 1 or more indicates that it does.

2) Install KVM and supporting packages.

Virt-Manager is a graphical application for managing your virtual machines. you can use the kvm command directly, but libvirt and Virt-Manager simplify the process.

```
#sudo apt-get install qemu-kvm libvirt-bin bridge-utils virt-manager
```

```
root@ubuntu:/home/tsec# apt-get install qemu-kvm libvirt-bin bridge-utils virt-manager
```


3) Create User.

Only the root user and users in the libvirtd group have permission to use KVM virtual machines. Run the following command to add your user account to the libvirtd group:

```
#sudo adduser tsec
```

```
#sudo adduser tsec libvirtd
```

After running this command, **log out** and **log back in as tsec**

```
root@ubuntu:/home/tsec# adduser tsec
root@ubuntu:/home/tsec# adduser tsec libvirtd
```

4) Check whether everything is working correctly.

Run following command after logging back in as **tsec** and you should see an empty list of virtual machines.

This indicates that everything is working correctly.

```
#virsh -c qemu:///system list
```

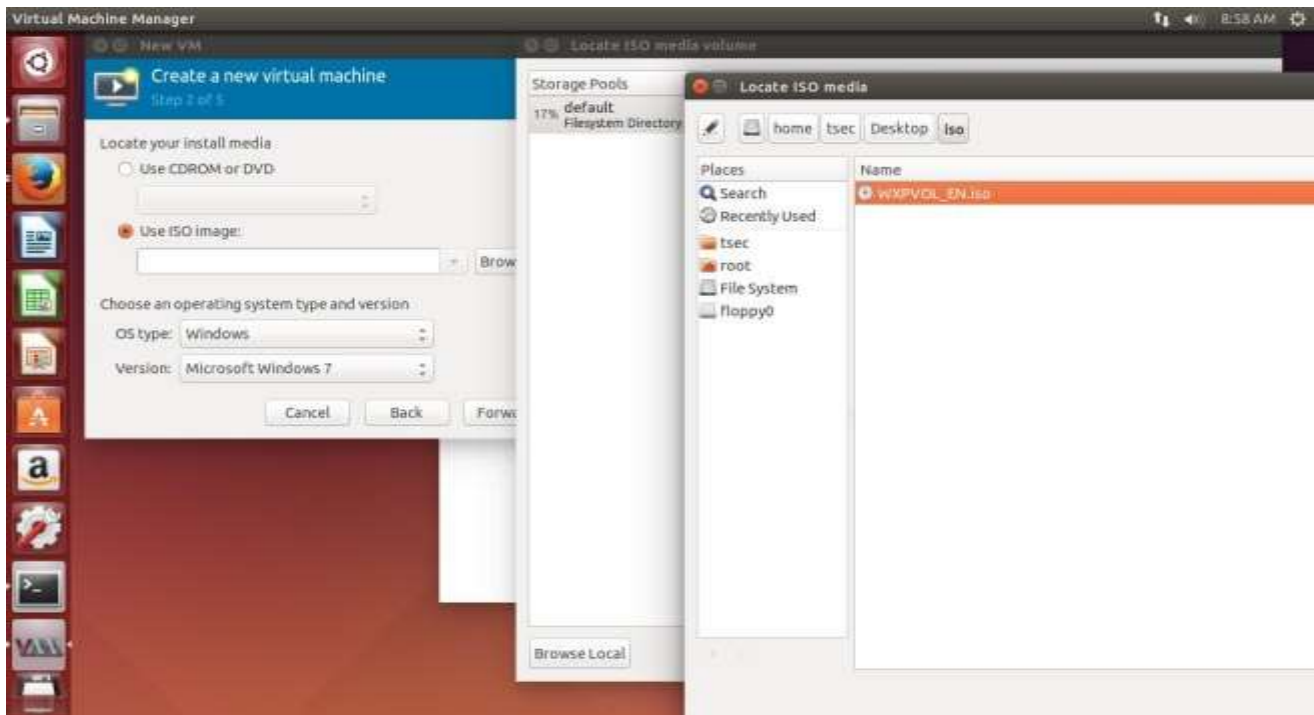
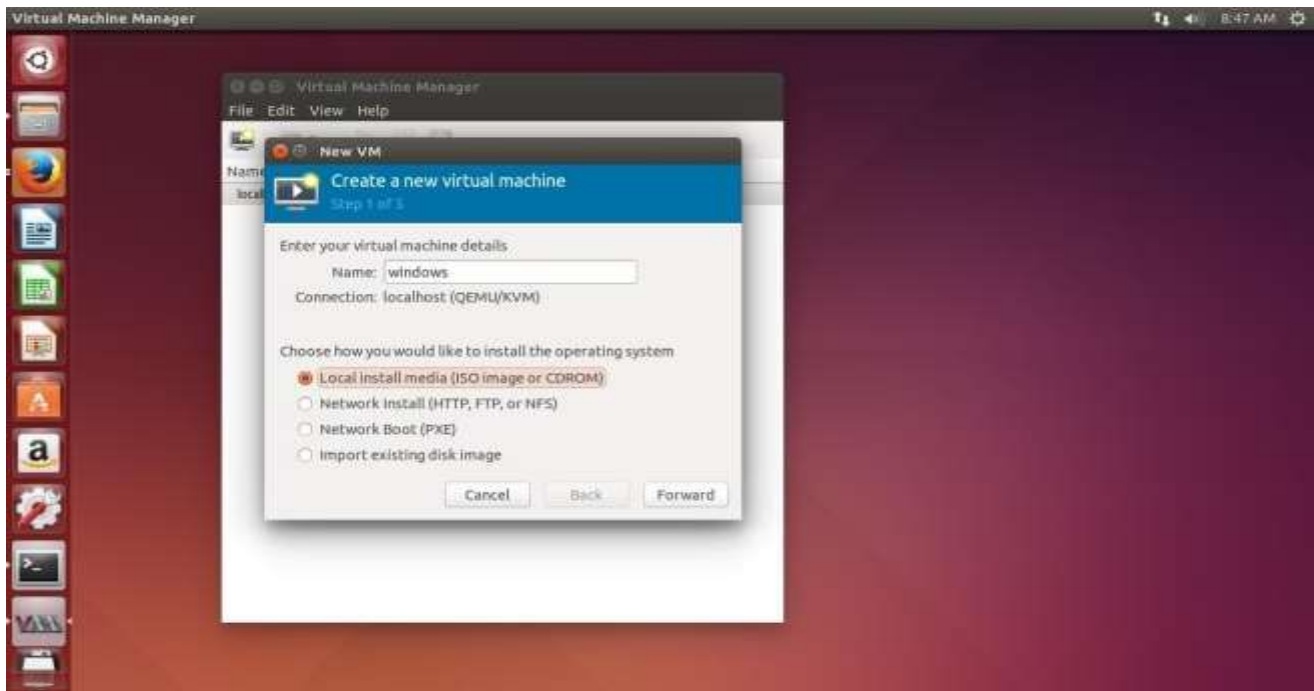
```
root@ubuntu:/home/tsec# virsh -c qemu:///system list
  Id      Name                                     State
-----
root@ubuntu:/home/tsec#
```

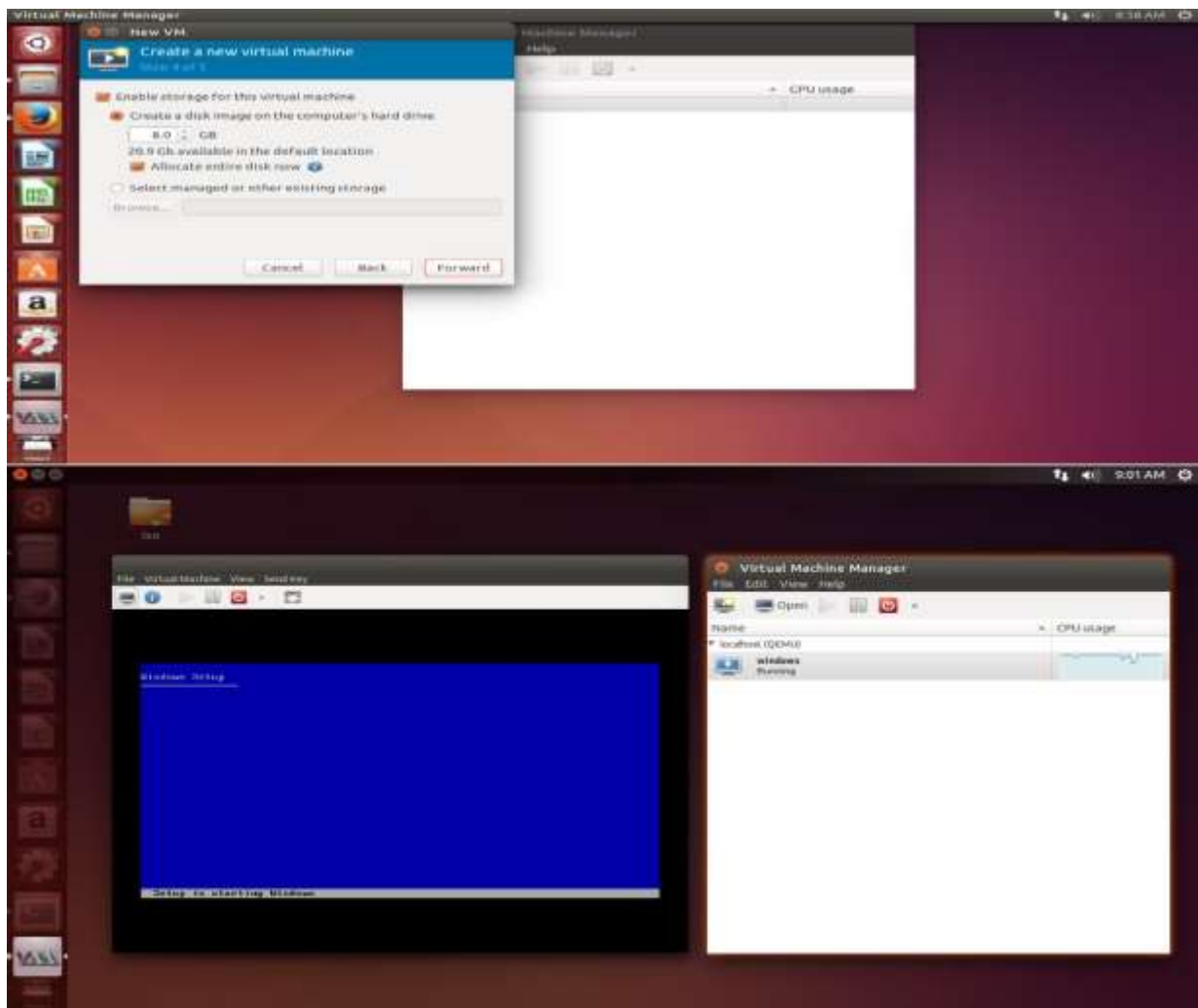
5) Open Virtual Machine Manager application and Create Virtual Machine

```
#virt-manager
```

```
root@ubuntu:/home/tsec# virt-manager
```

6) Create and run Virtual Machines





```
root$ sudo virsh list --all
```

Id	Name	State
1	ubuntu-vm	running

```
root$ virsh
```

```
Welcome to virsh, the virtualization interactive terminal.
```

```
Type: 'help' for help with commands
      'quit' to quit
```

```
virsh #
```

```
virsh #
```

```
virsh # list --all
```

Id	Name	State
1	Windows	running