THADOMAL SHAHANI TSEC ENGINEERING COLLEGE

EXPERIMENT NO. 2

Aim :- To study and implement Hosted Virtualization using VirtualBox and KVM.

Theory :-

Virtualization in Cloud Computing :-

Virtualization in cloud computing refers to the creation of virtualized versions of computing resources, such as servers, storage, and networks. It enables the efficient use of hardware resources by allowing VMs to run on a single physical machine. It is basically a process of creating a virtual version of computer hardware. This flexibility and resource optimization

Benefits of Virtualization :-

1) Resource Optimization : Efficiently utilize hardware resources, reducing costs.
2) Isolation : Isolate applications and services in independent virtual environments.
3) Flexibility : Easily scale resources up or down based on demand.
4) Cost Savings : Reduce hardware costs.
5) Improved Security : Isolate Virtual Machines, enhancing overall system security.

Hypervisor :-

A hypervisor, or Virtual Machine Monitor (VMM), is a software or hardware component that creates and runs virtual machines. It manages the allocation of physical resources, to multiple VMs, allowing them to operate independently.

Types of Hypervisors :

Type 1 :-
They run directly on the hardware, without need for host OS.
Provides higher performance and efficiency.
Eg. VMware ESXi, KVM.

Type 2 :-
They run on a host operating system.
Easy to install and use but may introduce extra overhead.
Eg. VirtualBox, Oracle VM ware Workstation

Comparisons between VirtualBox and KVM:

	KVM	VirtualBox
Architecture	Type 1 hypervisor	Type 2 hypervisor
Performance	Higher performance as it runs directly on hardware.	Generally lower performance due to running on top of host OS.
Resource Allocation	Supports dynamic allocation of resources to virtual machines	Limited resource management, less suitable for resource-intensive applications
Management	Managed through command-line tools and APIs, making it suitable for system administrators.	Provides a user-friendly graphical interface that makes it easier for beginners to manage VMs.
Hardware Support	Better hardware support and features like nested virtualization allowing running VMs within VMs.	Though supporting wide range of hardware, lacks same level of hardware compatibility as KVM.

A+
24/11/24.

C23

Rishab Mandal

2103110

Aim : - To study and implement Hosted Virtualization using VirtualBox& KVM.

Theory : -

Virtualization in Cloud Computing:

Virtualization in cloud computing refers to the creation of virtualized versions of computing resources, such as servers, storage, and networks. It enables the efficient use of hardware resources by allowing multiple virtual machines (VMs) to run on a single physical machine. This flexibility and resource optimization contribute significantly to the scalability and efficiency of cloud environments.

Benefits of Virtualization:

Resource Optimization:

Efficiently utilize hardware resources, reducing costs.

Isolation:

Isolate applications and services in independent virtual environments.

Flexibility:

Easily scale resources up or down based on demand.

Cost Savings:

Reduce hardware and maintenance costs.

Snapshot and Backup:

Create snapshots for easy backup and recovery.

Improved Security:

Isolate VMs, enhancing overall system security.

Compatibility:

Run multiple operating systems on a single physical machine.

What is Hypervisor? Examples of Hypervisor:

A hypervisor, or Virtual Machine Monitor (VMM), is a software or hardware component that creates and runs virtual machines. It manages the allocation of physical resources to multiple VMs, allowing them to operate independently. Examples of hypervisors include:

Type 1 (Bare-Metal Hypervisors):

VMware ESXi

Microsoft Hyper-V

KVM (Kernel-based Virtual Machine)

Type 2 (Hosted Hypervisors):

Oracle VirtualBox

VMware Workstation

Microsoft Virtual PC

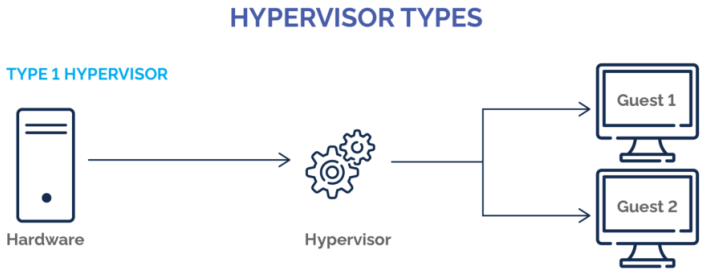
Types of Hypervisors:

Type 1 Hypervisors:

Run directly on the hardware, without the need for a host operating system.

Provide higher performance and efficiency.

Examples: VMware ESXi, KVM.

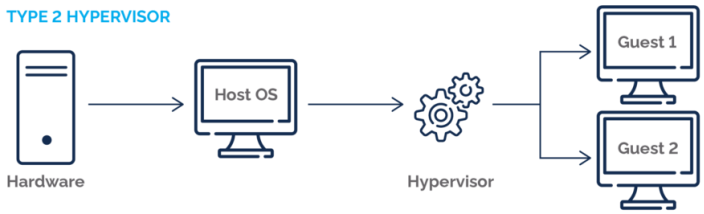


Type 2 Hypervisors:

Run on a host operating system.

Easy to install and use but may introduce overhead.

Examples: VirtualBox, VMware Workstation.



Comparisons between VirtualBox and KVM:

Architecture:

VirtualBox: Type 2 hypervisor.

KVM: Type 1 hypervisor.

Performance:

VirtualBox: Generally lower performance due to running on top of the host OS.

KVM: Higher performance as it runs directly on the hardware.

Host OS Compatibility:

VirtualBox: Compatible with various host operating systems.

KVM: Primarily used with Linux hosts.

Ease of Use:

VirtualBox: User-friendly with a graphical interface.

KVM: Typically used with command-line tools, may have a steeper learning curve.

Integration with Cloud Platforms:

VirtualBox: Primarily used for desktop virtualization.

KVM: Commonly used in cloud environments due to its performance.

Steps of Installation for the Experiment:

Installing VirtualBox:

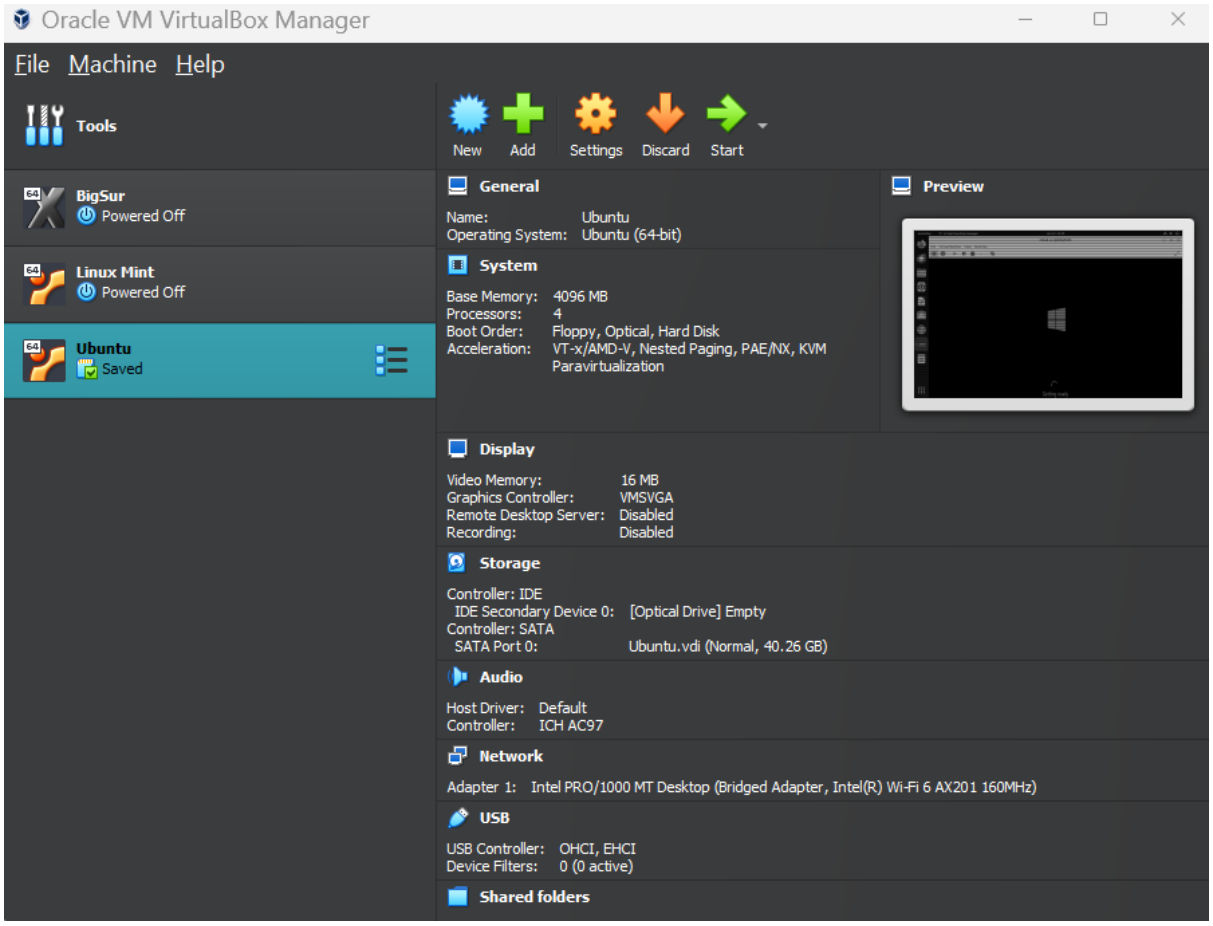
Download the VirtualBox installer from the official website.

Run the installer and follow on-screen instructions.

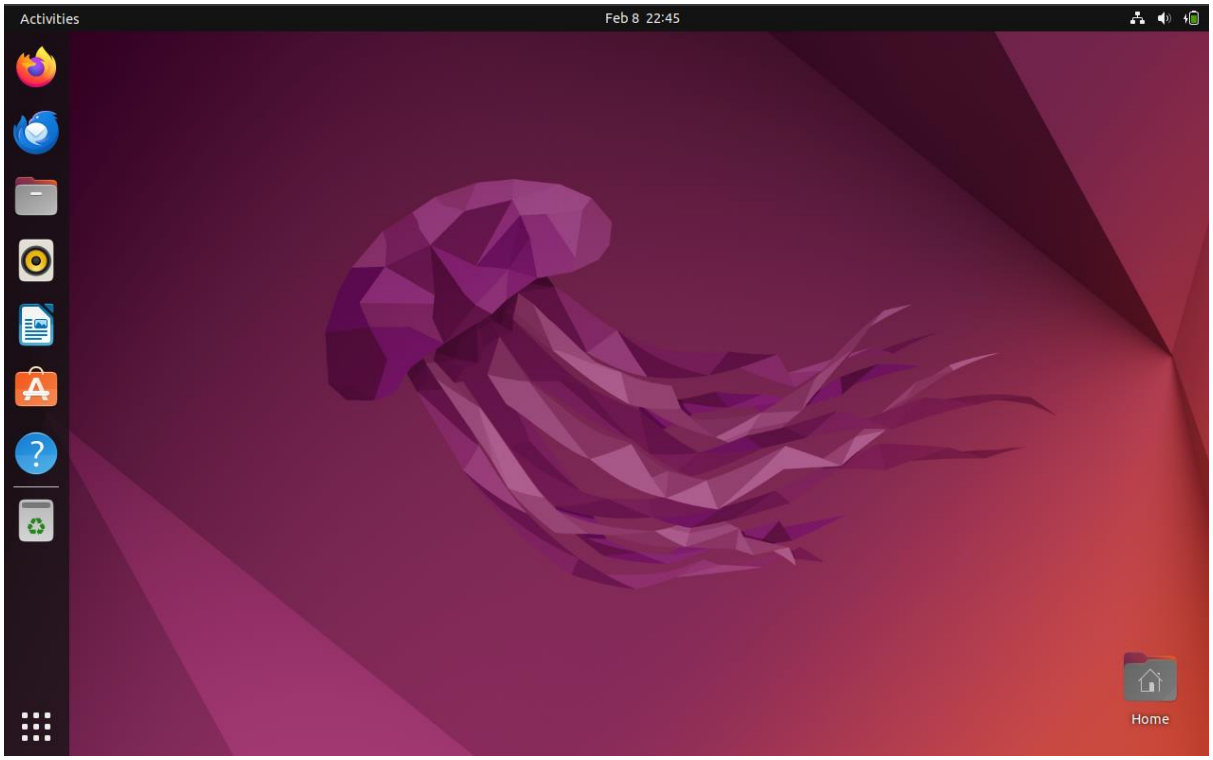
Launch VirtualBox after installation.

Create a new virtual machine by clicking "New."

Follow the wizard to configure VM settings and install the desired guest OS.



Start the VM and install the guest OS using an ISO image.

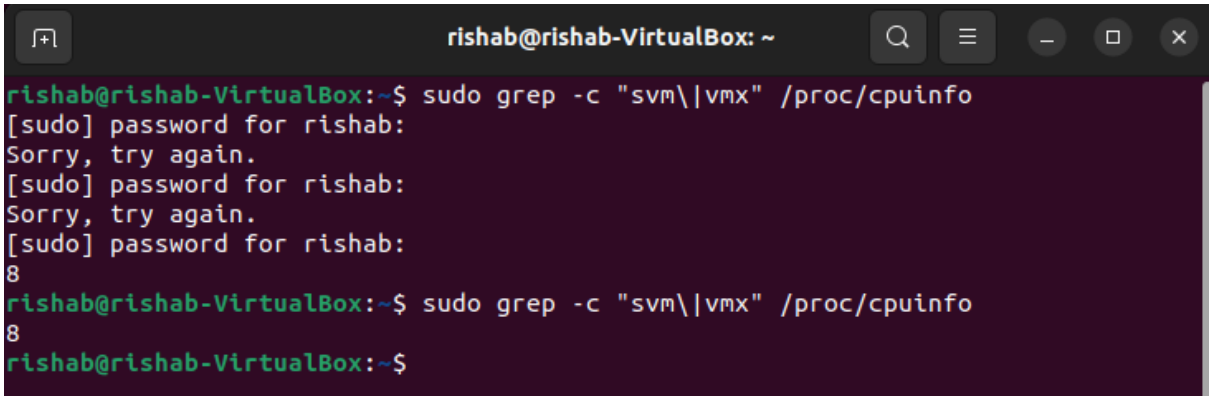


Installing KVM:

Ensure hardware virtualization support is enabled in the BIOS.

Open a terminal on a Linux host.

Install KVM packages using the package manager (e.g., sudo apt install qemu-kvm libvirt-clients libvirt-daemon-system on Ubuntu).



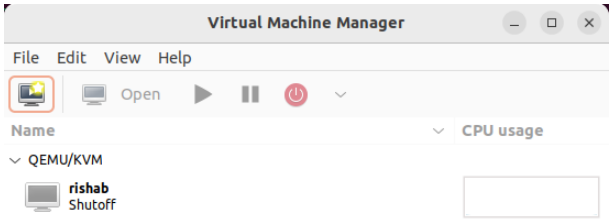
Add the user to the "libvirt" and "kvm" groups: sudo usermod -aG libvirt, kvm <username>.

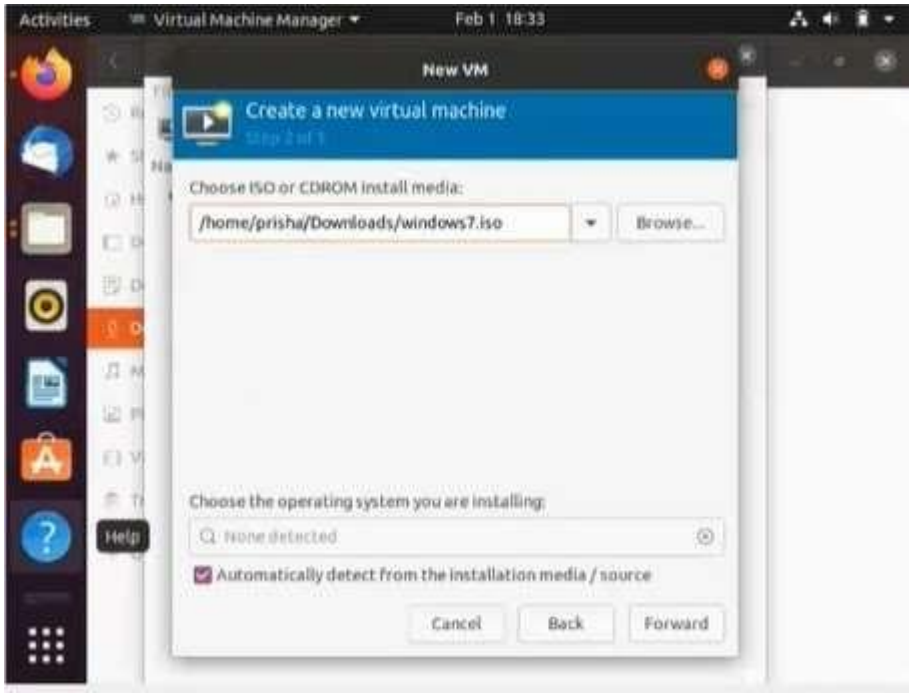
```
rishab@rishab-VirtualBox:~$ kvm-ok
INFO: /dev/kvm exists
KVM acceleration can be used
rishab@rishab-VirtualBox:~$
```

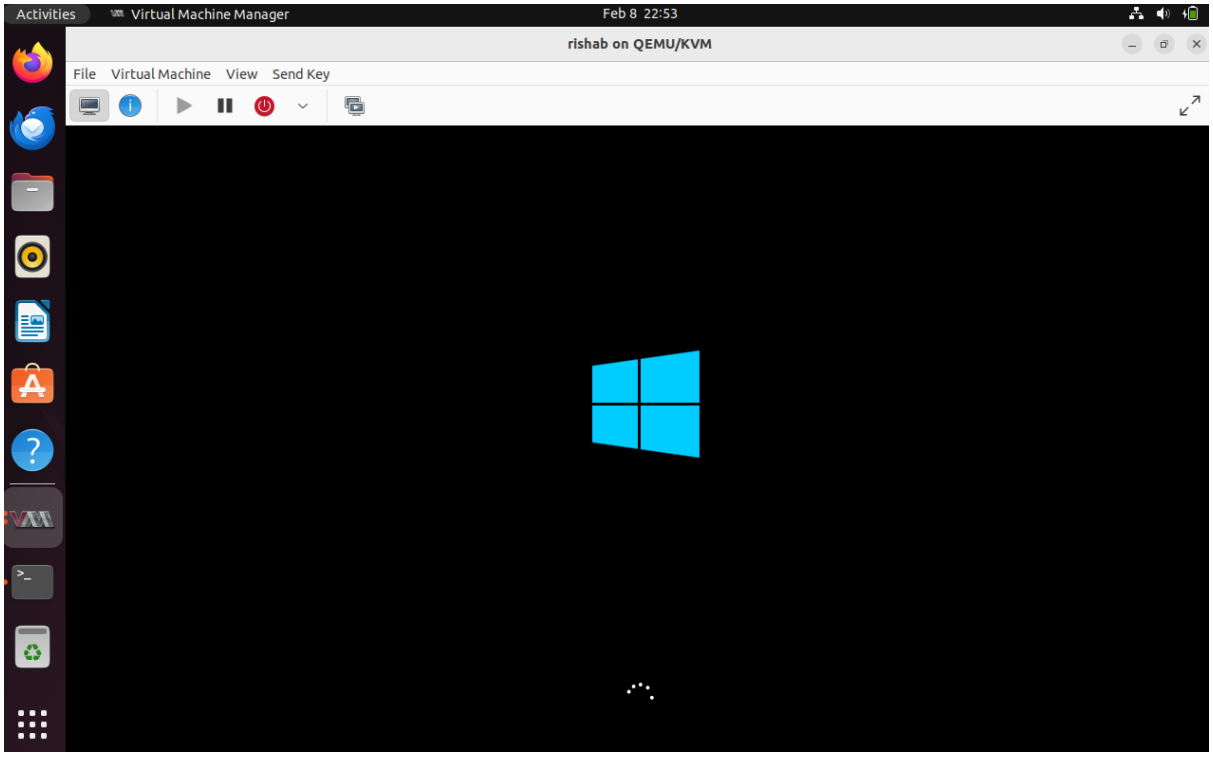
Log out and log back in to apply group changes.

Verify the installation: virsh list --all.

Create a new VM using a tool like "virt-manager" or "virsh."








```
rishab@rishab-VirtualBox:~$ sudo virsh -c qemu:///system list
 Id   Name   State
--------------------

rishab@rishab-VirtualBox:~$ sudo virsh -c qemu:///system list
 Id   Name     State
------------------------
 1    rishab   running
```