ETHIOPIAN IT PARK
የኢትዮጵያ ኢንፎርሜሽን ቴክኖሎጂ ፓርክ

CITCOT

ZERGAW CLOUD

IT Infrastructure Administration

Day 7 & 8: Server Virtualization Concepts

Ephrem Teshale(PhD)
Kibrom Temesgen
Tadios Abebe

July 29/30, 2025

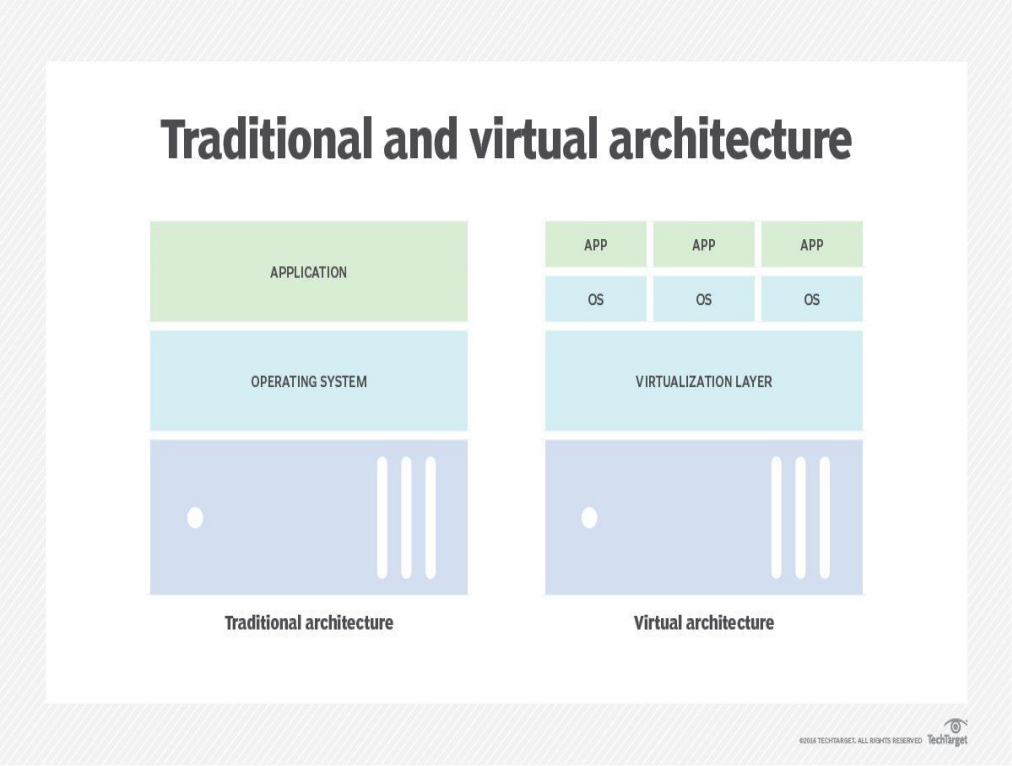
Server Virtualization Concepts

Day 7 & 8
Training Outline

What is virtualization
Why virtualization
Types of virtualization
Hypervisor
Types of hypervisor
Key players in server virtualization
KVM/QEMU virtualization
QEMU
KVM
Libvirt
Installing kvm/qemu
Virtual machine manager
Launching your first virtual machine
KVM/QEMU networking
Networking modes
KVM/QEMU storage
Virtual hard disks
Other features
Knowledge check
Exercise

What is Virtualization

- Virtualization is a concept that creates virtualized resources (CPU, Memory, Storage, Network interface, I/O devices…) and maps them to physical resources.
- Virtualization depends on 2 important concepts: virtual machines and hypervisors.
  - **Virtual machine (VM):** is a computing environment that functions as an isolated system with its own CPU, OS, memory, network interface, and storage, created from a pool of hardware resources.
  - **Hypervisors:** is software that separates a system's physical resources and divides those resources so that virtual environments can use them as needed.



Why Virtualization

- Hardware is underutilized
- Data Centers run out of space
- Energy cost go through the roof
- System administration costs becomes expensive
- Benefits
  - Resource efficiency
  - Easier management
  - Minimal downtime
  - Fast provisioning
  - Cost-effectiveness
  - Disaster recovery (DR)

Types of Virtualization

- **Server virtualization:** Server virtualization one of the most common types of virtualization, It is the process of dividing a physical server into multiple unique and isolated virtual servers by means of a software application. Each virtual server can run its own operating systems independently.
  - Allows maximum usage of existing physical servers
  - It provides:
    - Higher server ability
    - Cheaper operating costs
    - Eliminate server complexity
    - Increased application performance
    - Deploy workload quicker

- **Network Virtualization:** Network virtualization is the transformation of a network that was once hardware-dependent into a network that is software-based.
  - can combine multiple physical networks to one virtual network, or it can divide one physical network into separate, independent virtual networks.
  - allows to move virtual machines across different domains without reconfiguring the network
  - Physical network resources, such as switching, routing, firewalling, load balancing, virtual private networks (VPNs), and more, are pooled,
  - delivered in software, and require only IP packet forwarding from the underlying physical network

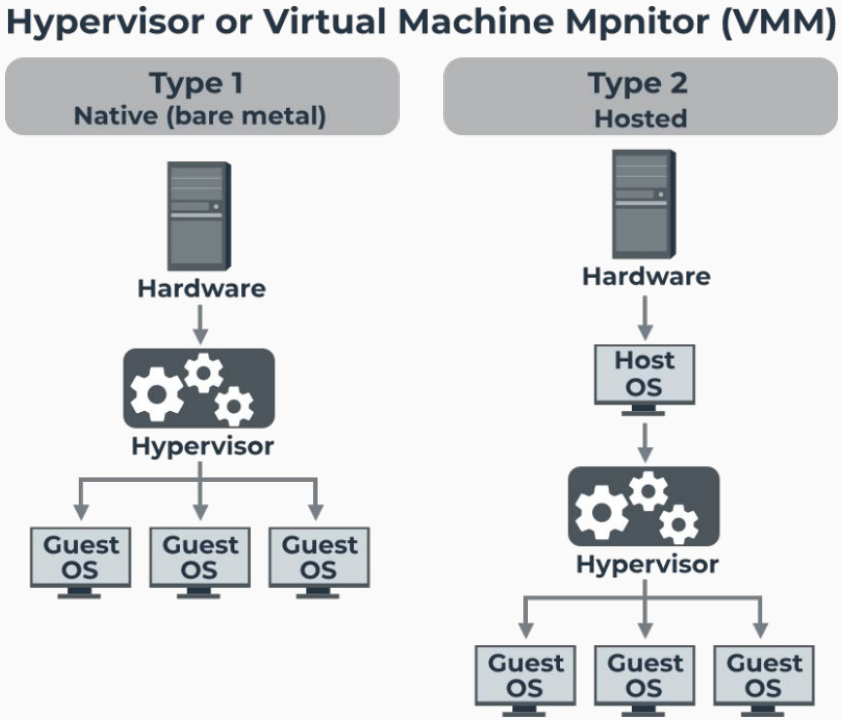
Cont…

- **Storage Virtualization:** pooling of physical storage resource from multiple network storage devices into a single virtual storage device that is managed centrally.
  - Equips these storage's with enhanced capabilities such as replication and disaster recovery
  - used to create flexible and scalable pools of storage resources.
  - It could provide on block-level, file-level and object-level storage
  - Method of storage virtualization
    - Network-based
    - Host-based
    - Array-based

- **Other virtualizations worth mentioning**
  - Network Functions Virtualization
  - Desktop Virtualization
  - Containerization
  - Data Center Virtualization
    - takes the concept of server virtualization and applies it to an entire data center
    - virtualizing not just individual servers, but also storage, networking, and other resources
    - creates a software-defined data center (SDDC) where resources can be easily pooled and allocated to different workloads on demand.

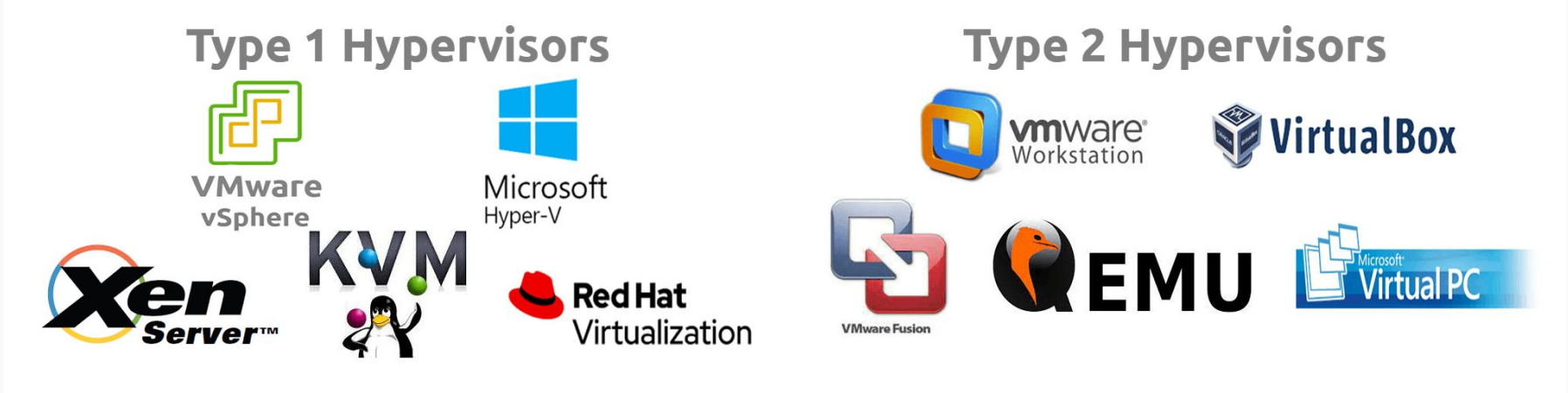
Hypervisor

- Hypervisor is a piece of software that is responsible for monitoring and controlling virtual machines or guest OSes.
- Responsible for providing virtual hardware, virtual machine lifecycle management, migrating virtual machines, allocating resources in real time, defining policies for virtual machine management, and so on
- responsible for efficiently controlling physical platform resources, such as memory translation and I/O mapping.
- Each VM operates in a separate, protected space
- Hypervisors can sit on top of an operating system (like on a laptop) or be installed directly onto hardware (like a server)
- In terms of types, we can categorize hypervisors as either type 1 or type 2 base on where they reside in the system.



Types of Hypervisors

- **Type 1 (Baremetal):** hypervisor runs directly on top of the hardware replacing the traditional operating system altogether.
  - Directly interacts with the system hardware.
  - Most commonly appear in virtual server scenarios
  - Small in size; optimized to give most of the physical resources to the virtual machines
- **Type 2 (Hosted):** run as an application on an existing OS
  - Most commonly used on endpoint devices to run alternative operating systems
  - carry a performance overhead because they must use the host OS to access and coordinate the underlying hardware resource.

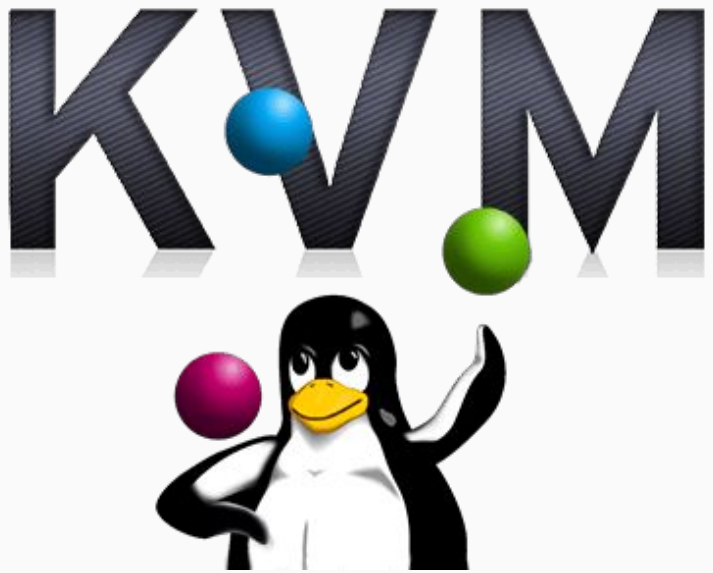


Key Players in Server Virtualization

- **VMware:** Industry-leading enterprise hypervisor with vCenter for centralized management. Very mature and feature-rich. provides both Type 1 (VMware-ESXI) and Type 2 hypervisors (VMware-Workstation), commercial for type 1 hypervisor orchestration.
- **Microsoft Hyper-V:** A Type 1 hypervisor Built into Windows Server and Pro editions. Integrates with Windows environments and Azure. commercial for production use.
- **Linux KVM:** Kernel-based Virtual Machine built into Linux. Scalable, widely used (even in clouds). Paired with libvirt/QEMU for management. open source platform.
  - **Redhat:** enterprise virtualization platform based on KVM + libvirt. oVirt is the upstream open-source version.
  - **Proxmox:** Open-source virtualization management platform using KVM for VMs and LXC for containers. Includes web UI, clustering, backup tools. open source platform
- **Citrix/Xen:** Xen an open-source hypervisor technology. Citrix acquired Xen in 2008 and offered a commercial product (Citrix Hypervisor)
  - Oracle: A type 1 hypervisor Built on Xen hypervisor. Oracle-focused workloads (e.g., Oracle Linux, DB). commercial platform. also provides a type 2 hypervisor with the name Oracle VirtualBox.

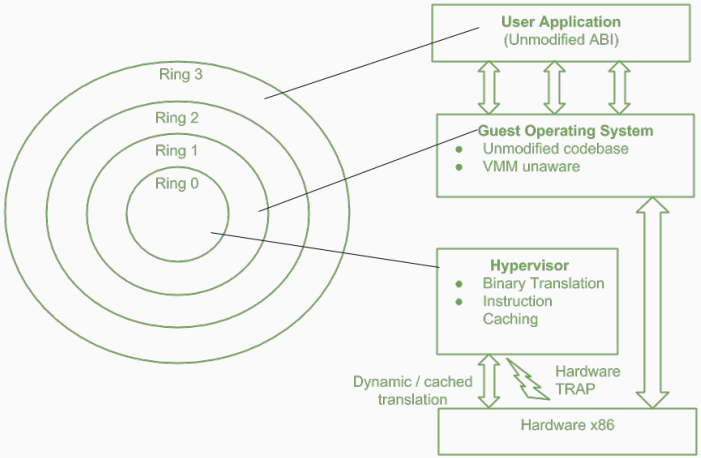
KVM/QEMU Virtualization

- Kernel-based Virtual Machine (KVM) is an open source virtualization technology for Linux operating systems. With KVM, Linux can function as a hypervisor that runs multiple, isolated virtual machines (VMs).
- KVM was announced in 2006 and merged into the Linux kernel a year later. Many open source virtualization technologies
  - Redhat virtualization
  - Proxmox
- It consists of KVM, QEMU and libvirt which their interaction gives the full virtualization with hardware assisted capabilities.



QEMU

- It's a free piece of software and mainly licensed under GNU's General Public License (GPL).
- QEMU is a generic and open source machine emulator and virtualizer.
- **QEMU as an emulator**
  - It is capable of running OSes/programs made for one machine type on a different machine type
  - It uses binary translation methods. It transforms the binary code written for a given processor into another form of binary code (such as ARM in X86)
  - QEMU emulates CPUs through dynamic binary translation techniques, peripherals such as disks, networks, VGA, PCI, serial and parallel ports, USB…
  - Enabled to run different unmodified guest OSes with different architectures with Sacrifice a bit of execution speed



Cont…

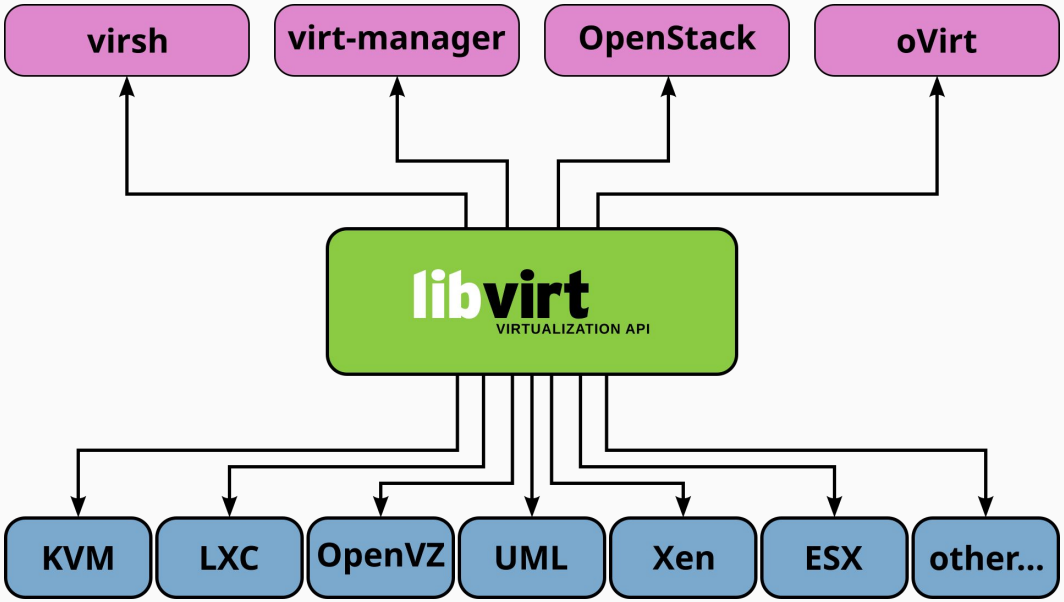
- **QEMU as a virtualizer:** In this mode where QEMU executes the guest code directly on the host CPU
  - QEMU can operate in this mode when working under Xen/KVM hypervisors,
  - when working with KVM, QEMU-KVM creates and initializes virtual machines
  - it also initializes different POSIX threads for each virtual CPU (vCPU) of a guest
  - the user-space part of the KVM hypervisor is provided by QEMU. QEMU runs the guest code via the KVM kernel module.
  - It also provides a framework that's used to emulate the virtual machine's physical address space within the user-mode address space of QEMU-KVM:
- KVM makes use of QEMU to become a complete hypervisor.
- KVM is an accelerator or enabler of the hardware virtualization extensions (VMX or SVM) provided by the processor so that they're tightly coupled with the CPU architecture.

KVM

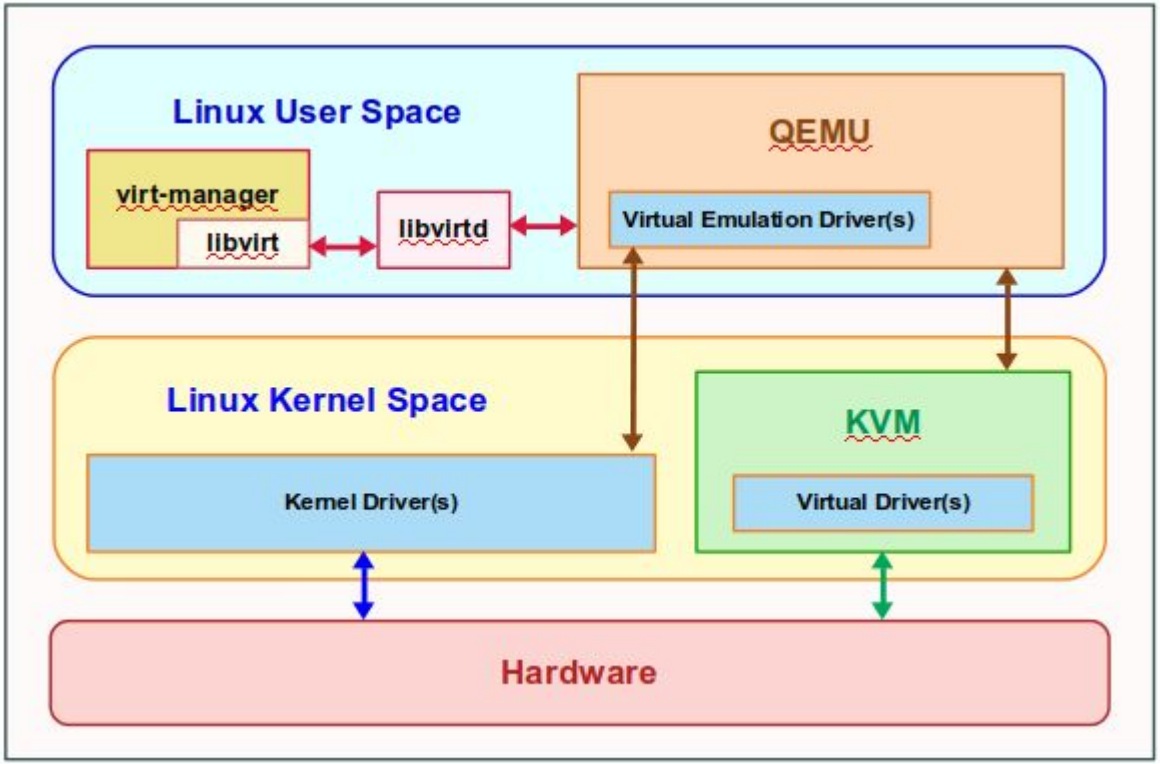
- KVM is not a full hypervisor; however, with the help of QEMU and emulators (a slightly modified QEMU for I/O device emulation and BIOS), it can become one
- KVM needs hardware virtualization-capable processors to operate. Using these capabilities, KVM turns the standard Linux kernel into a hypervisor
- When KVM runs virtual machines, every virtual machine is a normal Linux process, which can obviously be scheduled to run on a CPU by the host kernel, as with any other process present in the host kernel.

Libvirt

- When working with KVM, you're most likely to first interface with its main Application Programming Interface (API), called libvirt
- **libvirt** is an open-source APi, daemon, management library and toolkit for platform virtualization.
- The goal of the libvirt library is to provide a common and stable layer for managing virtual machines running on a hypervisor, including: KVM / QEMU, Xen, LXC…
- It has both CLI utilities called "*virsh*" and a Gnome-based graphical utility called "*virt-manager*"
- The libvirt process is also daemonized, and it is called *libvirtd.*



Libvirt, QEMU, KVM



Installing kvm/qemu

Prerequisite

- ❏ Check to sure that the desktop CPU has the virtualization extensions enabled by executing the following command:
  - ❏ egrep '(vmx|svm)' /proc/cpuinfo

The following would be a typical output on an Intel based platform:

flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc cpuid aperfmperf pni pclmulqdq dtes64 monitor ds_cpl **vmx** est tm2 ssse3 sdbg fma cx16 xtpr pdcm pcid sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx f16c rdrand lahf_lm abm 3dnowprefetch cpuid_fault epb invpcid_single pti tpr_shadow vnmi flexpriority ept vpid fsgsbase tsc_adjust bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap intel_pt xsaveopt dtherm ida arat pln pts

flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc cpuid aperfmperf pni pclmulqdq dtes64 monitor ds_cpl **vmx** est tm2 ssse3 sdbg fma cx16 xtpr pdcm pcid sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx f16c rdrand lahf_lm abm 3dnowprefetch cpuid_fault epb invpcid_single pti tpr_shadow vnmi flexpriority ept vpid fsgsbase tsc_adjust bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap intel_pt xsaveopt dtherm ida arat pln pts

flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc cpuid aperfmperf pni pclmulqdq dtes64 monitor ds_cpl **vmx** est tm2 ssse3 sdbg fma cx16 xtpr pdcm pcid sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx f16c rdrand lahf_lm abm 3dnowprefetch cpuid_fault epb invpcid_single pti tpr_shadow vnmi flexpriority ept vpid fsgsbase tsc_adjust bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap intel_pt xsaveopt dtherm ida arat pln pts

flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc cpuid aperfmperf pni pclmulqdq dtes64 monitor ds_cpl **vmx** est tm2 ssse3 sdbg fma cx16 xtpr pdcm pcid sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx f16c rdrand lahf_lm abm 3dnowprefetch cpuid_fault epb invpcid_single pti tpr_shadow vnmi flexpriority ept vpid fsgsbase tsc_adjust bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap intel_pt xsaveopt dtherm ida arat pln pts

Next, check the KVM kernel module is loaded by executing the following command:

lsmod | grep kvm

```
kvm_intel        204800  0
kvm              593920  1 kvm_intel
irqbypass         16384  1 kvm
```

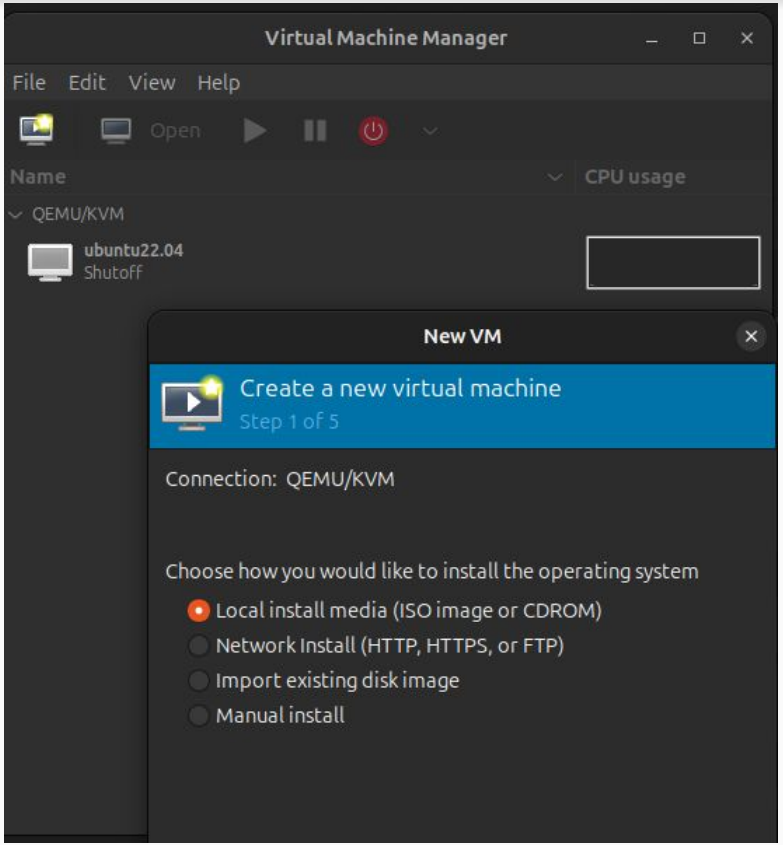
Installing kvm/qemu

- In order to create and manage virtual machine(s), we need to install QEMU along with some additional software tools.
- To install QEMU and the additional tools, execute the following command:

  sudo apt-get install qemu-kvm virt-manager bridge-utils
- The **qemu-kvm** package installs all the necessary system binaries related to QEMU.
- The **virt-manager** package installs a graphical interface tool to create and manage virtual machines on KVM. It is a Python based GUI tool that interacts with libvirt.
- The **bridge-utils** package installs a utility that will be used to create and manage bridge network devices.
- After that to check whether our host is compatible with all the necessary KVM requirements issue the following command
  - *virt-host-validate*
- To do some pre-flight checks to see whether everything that we installed was deployed correctly and works like it should.
  - *virsh net-list* - - - it should list the default network
  - virsh list - - - to list for vm's running, we will get no list since we didn't deploy a VM
  - systemctl status libvirtd

Virtual-Machine-Manager

- The virt-manager application is a desktop user interface for managing virtual machines through libvirt
- It primarily targets KVM VMs, but also manages Xen and LXC (linux containers).
- An embedded VNC and SPICE client viewer for a full graphical console
- we can install, delete, power off, edit network, create a new network, clone a virtual machine, …
- It also presents a summary view of running domains, their live performance & resource utilization statistics.

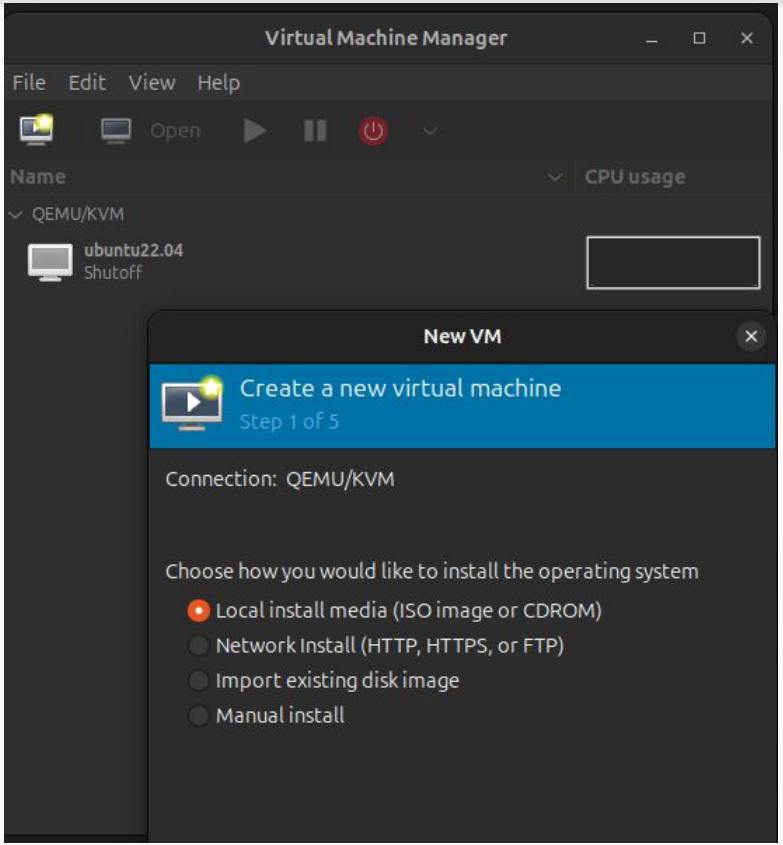


Virtual-Machine-Manager

- virt manager can be installed with the following command or OS specific binary packages

  apt install virt-manager

- To add a remote kvm host to be managed by virt manager you can go to File > Add Connection
- On the add connection window provide the ssh username and remote kvm host
- If you are using custom ssh port pass the custom port after a colon.



Launching your first virtual machine

Download cirros image

wget
https://download.cirros-cloud.net/0.6.3/cirros-0.6.3-x86_64-disk.img
sudo mv cirros-0.6.3-x86_64-disk.img /var/lib/libvirt/images/cirros.img
*sudo virt-install *
*--name cirros *
*--memory 256 *
*--vcpus 1 *
*--disk path=/var/lib/libvirt/images/cirros.img,format=qcow2 *
*--import *
*--os-variant generic *
*--network network=default *
*--graphics none *
--noautoconsole

Use the following virsh commands to interact with the newly launched VM

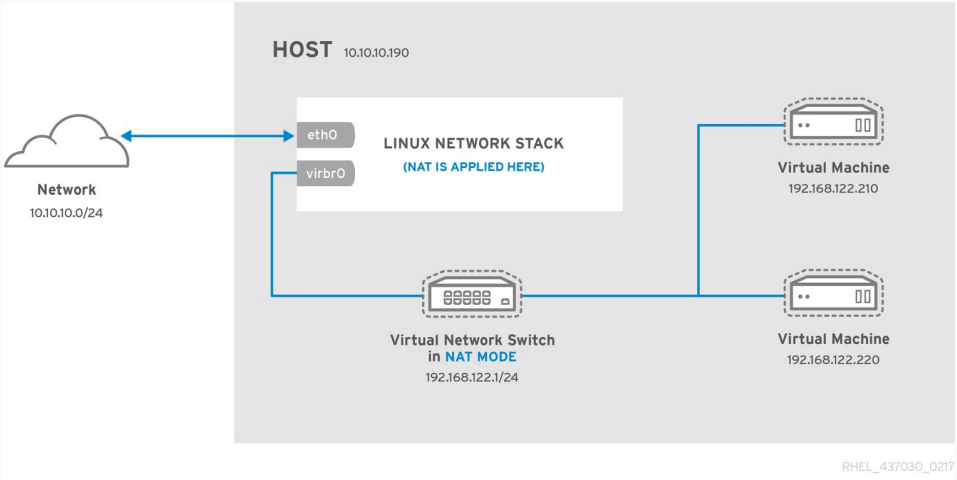
virsh list
virsh console cirros
virsh shutdown cirros
virsh start cirros
virsh destroy cirros

KVM/QEMU Networking

- KVM (Kernel-based Virtual Machine) uses virtual networking to allow virtual machines (VMs) to communicate with each other and the host system, as well as with external networks.
- It essentially emulates physical networking components like switches and network interface cards (NICs) within the virtualized environment.
- **Virtual Network Interface Cards (vNICs):** Each KVM guest VM is equipped with one or more vNICs, which are software representations of physical NICs. These vNICs allow the VM to send and receive network traffic.
- **Virtual Switches (Bridges):** A core component is the virtual network switch, often referred to as a "bridge" in Linux (e.g., virbr0). This software-based switch runs on the KVM host and acts like a physical switch, allowing VMs connected to it to communicate with each other.

Networking modes

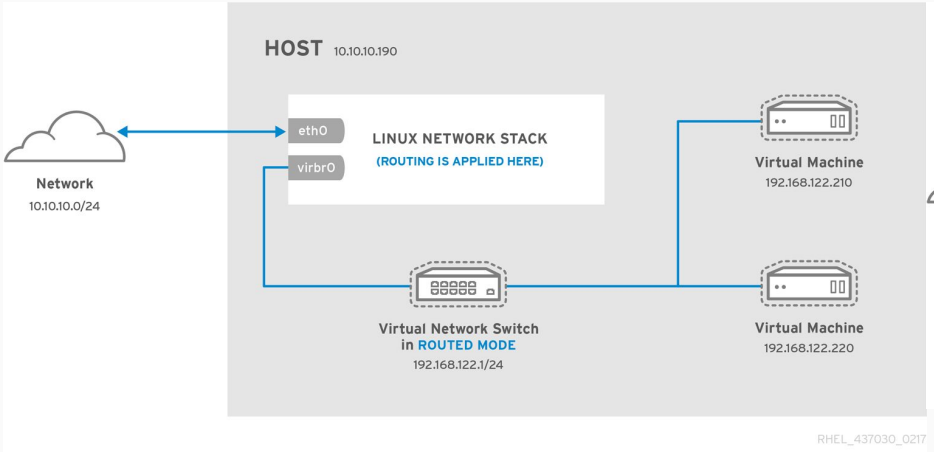
- **NAT (Network Address Translation):** In a NAT libvirt network our virtual machine is behind a libvirt switch in NAT mode.
- It is the default networking in kvm
- uses a Linux bridge in combination with Network Address Translation (NAT) to enable a guest OS to get outbound connectivity
- allows KVM guests sharing the same bridge to communicate together even if the KVM host has no physical networking installed or enabled.
- Interfaces associated to the NAT are not, by default, visible outside of the KVM host running the NAT

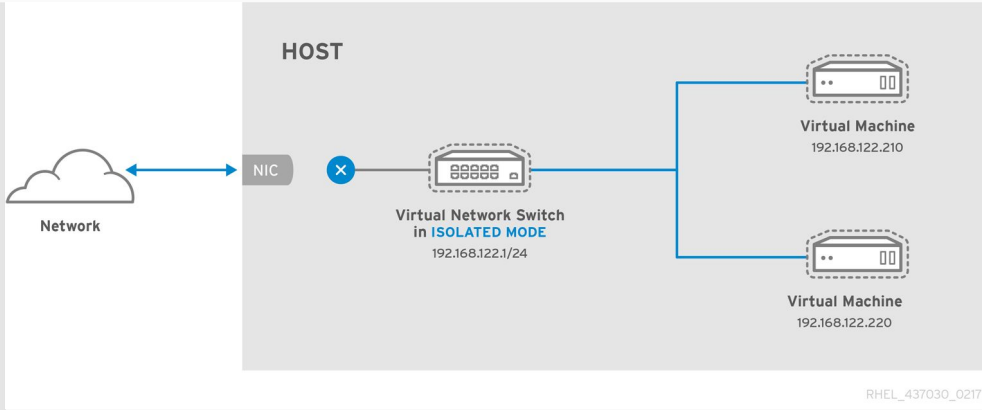


Cont…

- **Routed/Bridged networking:** our virtual machine is directly connected to the physical network via a virtual switch
- our virtual machine is in the same Layer ⅔ network as the physical host

- **Isolated network:** virtual machines attached to the same isolated switch can communicate with each other, but they cannot communicate with anything outside the host that they're running on.





KVM/QEMU Storage

- In KVM storage or A *storage pool* is a file, directory, or storage device managed by libvirt for the purpose of providing storage to guest virtual machines.
- The storage pool can be local or it can be shared over a network.
- Storage pools are divided into storage volumes
- Storage volumes are an abstraction of physical partitions, LVM logical volumes, file-based disk images and other storage types handled by libvirt.
- libvirt uses a directory-based storage pool, the `/var/lib/libvirt/images/` directory, as the default storage pool.
- Storage volumes are presented to guest virtual machines as local storage devices regardless of the underlying hardware.

- **Local storage pools**: Local storage pools are directly attached to the host physical machine server.
  - local directories, directly attached disks, physical partitions, and LVM volume groups.
- **Networked (shared) storage pools**: Networked storage pools include storage devices shared over a network using standard protocols
  - Supported protocols for networked storage pools include:
    - Fibre Channel-based LUNs
    - iSCSI
    - NFS
    - GFS2

Virtual Hard Disks

- In KVM virtual hard disks are needed to store VM data. They simulate a physical hard drive for a VM. Unlike physical drives, they are just files in datastore
- the popular format are RAW and Qcow2.
- **Raw Disk Image:** Raw image format represents a plain, unstructured disk image with no compression or optimization applied mirroring the physical structure of a hard disk drive.
  - It is simple to convert Raw image to other kinds of image
  - Raw images themselves don't natively support snapshots within the format
  - They occupy the entire allocated storage space, even for unused portions
  - the performance of Raw image is high
- **Qcow2 Disk Image**: Qcow2 (QEMU Copy-On-Write version 2) is a versatile disk image format that supports features like compression, snapshots, and thin provisioning.
  - Copy-On-Write mechanism allows for efficient storage utilization by only writing changes to disk
  - Qcow2 image occupies less storage because the file system doesn't support holes
  - Qcow2 image supports snapshot and one image can include multiple snapshots.
  - the extra features in qcow2 and the COW mechanism can cause performance overhead.

Other Features

Snapshot:

A snapshot captures the entire VM state at a moment in time:

- ❏ Disk state (copy-on-write or full)
- ❏ Memory (if specified)
- ❏ CPU state (if memory is saved)

This allows you to:

- ❏ Roll back after changes or failures
- ❏ Save temporary working states
- ❏ Test risky configurations safely

Cloning:

A clone is a new VM copied from an existing one, optionally with:

- ❏ A new MAC address
- ❏ A new UUID
- ❏ Optionally a full or linked disk copy

Useful for:

- ❏ Rapid deployments
- ❏ Template-based provisioning
- ❏ Testing scenarios

Knowledge Check

Which of the following is not a valid reason to implement virtualization?

A. Simplified backups

B. Faster hardware degradation

C. Easier resource scaling

D. Legacy application support

What type of virtualization allows multiple operating systems to run on a single physical server?

A. Application virtualization

B. Network virtualization

C. Server virtualization

D. Storage virtualization

Which type of virtualization isolates applications from the underlying OS?

A. Server virtualization

B. Desktop virtualization

C. Application virtualization

D. Hardware virtualization

Knowledge Check

Which is not a form of virtualization?

A. Network

B. BIOS

C. Storage

D. Desktop

Desktop virtualization typically provides:

A. Dedicated graphics performance

B. Remote OS instances for user access

C. Layer 2 routing between VMs

D. Software RAID for data redundancy

A key characteristic of a Type 1 hypervisor is:

A. It runs inside a host operating system

B. It requires a browser interface

C. It runs directly on hardware without a host OS

D. It cannot be managed remotely

Knowledge Check

Which of the following is considered a Type 2 hypervisor?

A. Microsoft Hyper-V

B. VMware ESXi

C. Oracle VirtualBox

D. Xen

Which hypervisor is native to Microsoft?

A. KVM

B. Hyper-V

C. Xen

D. VirtualBox

Which of the following is open-source?

A. VMware Workstation

B. Microsoft Hyper-V

C. KVM

D. Parallels

Knowledge Check

What is the role of QEMU in a KVM virtualization stack?

A. Kernel-level hypervisor

B. Userspace VM emulator

C. Storage backend

D. Network packet filter

What does KVM require from the CPU?

A. ECC memory

B. Hyperthreading

C. Virtualization extensions (VT-x or AMD-V)

D. GPU passthrough

What is Libvirt's primary role?

A. Building VM templates

B. Acting as a kernel module

C. Providing a management interface for virtualization platforms

D. Compiling QEMU binaries

Knowledge Check

Which package installs KVM on Ubuntu?

A. virt-kvm

B. qemu-hypervisor

C. qemu-kvm

D. kvm-libvirt

Which tool provides a graphical interface to manage KVM virtual machines?

A. Libvirt

B. virt-manager

C. iptables

D. bridge-utils

After installing qemu-kvm, which service is used to define and start VMs from the CLI?

A. systemctl

B. kvmctl

C. virsh

D. qemuctl

Knowledge Check

Which KVM network mode allows a VM to appear as a physical machine on the LAN?

A. NAT

B. Bridge

C. Host-only

D. Virtual

What is a drawback of using NAT mode in KVM?

A. Lack of outbound internet access

B. VM has no IP address

C. Inbound connections to the VM require port forwarding

D. VMs are visible on the LAN

Which KVM networking tool is used to configure bridge interfaces on Ubuntu?

A. brctl

B. nmcli

C. virsh

D. nmtui

Knowledge Check

What is the default virtual disk format used by QEMU/KVM for efficient disk space usage?

A. VHD

B. VMDK

C. QCOW2

D. RAW

Which statement about QCOW2 is true?

A. It uses more space than RAW

B. It supports snapshots and compression

C. It can only be used with Hyper-V

D. It is read-only

Which command creates a new QCOW2 disk image?

A. mkdisk

B. virt-install

C. qemu-img create

D. qemu-make-disk

Knowledge Check

Which command lists the currently running virtual machines?

A. kvm-list

B. virsh list

C. qemu -vms

D. virt-show

What does the virt-clone tool do?

A. Creates multiple VMs at once

B. Clones a virtual machine's configuration and disk

C. Upgrades QEMU

D. Converts raw disk to qcow2

Which is not a benefit of virtualization?

A. Efficient hardware utilization

B. Isolated environments

C. Automatic source code generation

D. Easier backup and recovery

Thank You

- +2519770355111
- info@citcot.com
- citcot.com
  nunaethiopia.com