

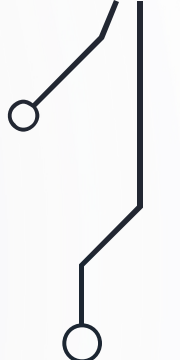
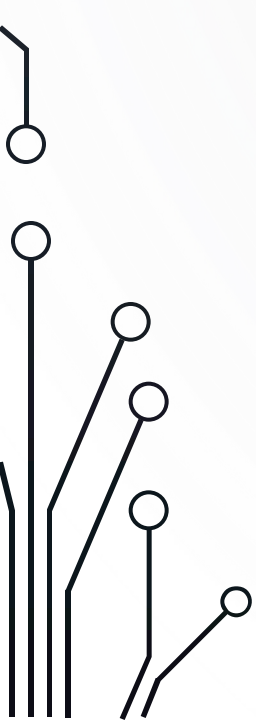
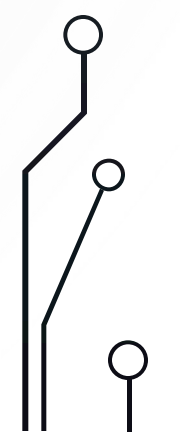


## ***Virtualization and Installation of On-Prem Video Conferencing Platform***

**Kernel-based Virtual Machine (KVM)**

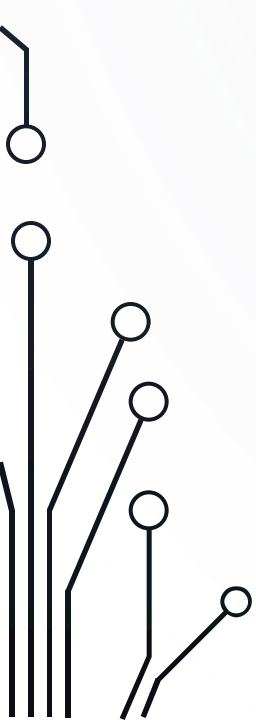


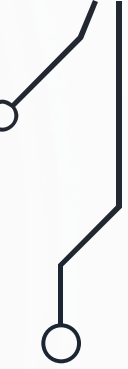
# VIRTUALIZATION

- Abstracting functionality away from the computing (processors, memory, storage and more) resources .
  - Allows to use physical computing resources by distributing its capabilities among many users or environments
  - Creates an external interface that hides the underlying implementations
  - The technology that drives cloud computing economics
  - Virtualization enables more efficient utilization of physical computer hardware and allows a greater return on an organization's hardware investment.
  - Software called Hypervisors enable virtualization
- 
- 
- 



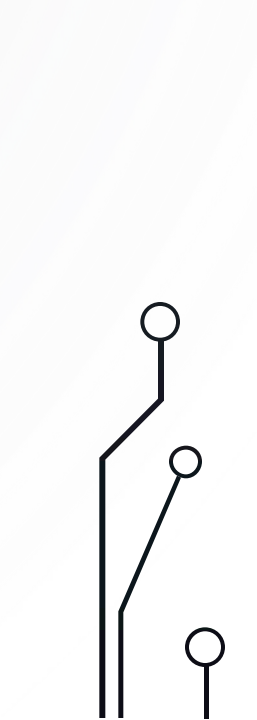
# VIRTUALIZATION BENEFITS

- Virtualization has below benefits and more
  - Improved capacity usage
  - Lower power consumption
  - Reduced system administration overhead
  - Better reliability (uptime, data loss)
  - Heterogeneous hardware platform created through virtualization
  - Not tied to a particular vendor - Migrations are easy
  - Reduced physical space
  - Reduced shipping costs
  - Reduced logistics
  - For testing and education
- 





# TYPES OF VIRTUALIZATION

- Server virtualization
  - Desktop virtualization
  - Operating system virtualization
  - Network virtualization
  - Storage virtualization
  - Data virtualization
  - Application virtualization
- 
- 
- 

# HYPERVISORS

- A hypervisor is a software that creates and runs virtual machines (VMs)
- Also called a virtual machine monitor (VMM)
- Treats the resources like CPU, memory, and storage as a pool
- Manages and schedules the resources given to VMs against the physical resources
- Host machine - A computer on which a hypervisor runs virtual machines Guest machine – Virtual machine
- Two types - Type 1 hypervisor and Type 2 hypervisor

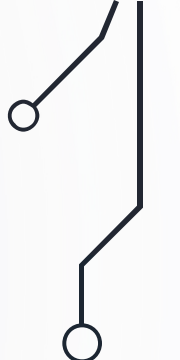


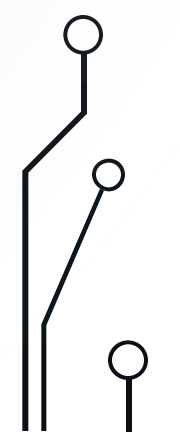
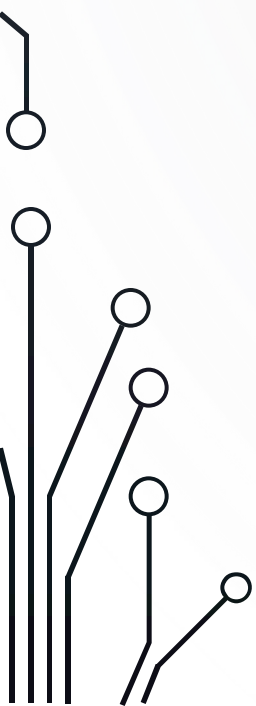
# TYPE 1 HYPERVISORS

- Runs on bare-metal ( takes the place of the operating system ) Direct access to the hardware resources
- High performance as there is no middle layer
- Better scalability – support large number of VMs
- Better security
- Most common in virtual server/machine scenarios
- Ex:
  - KVM (converts the linux to a type 1 hypervisor)
  - VMware ESXi
  - Microsoft Hyper-V
  - Citrix ZenServer

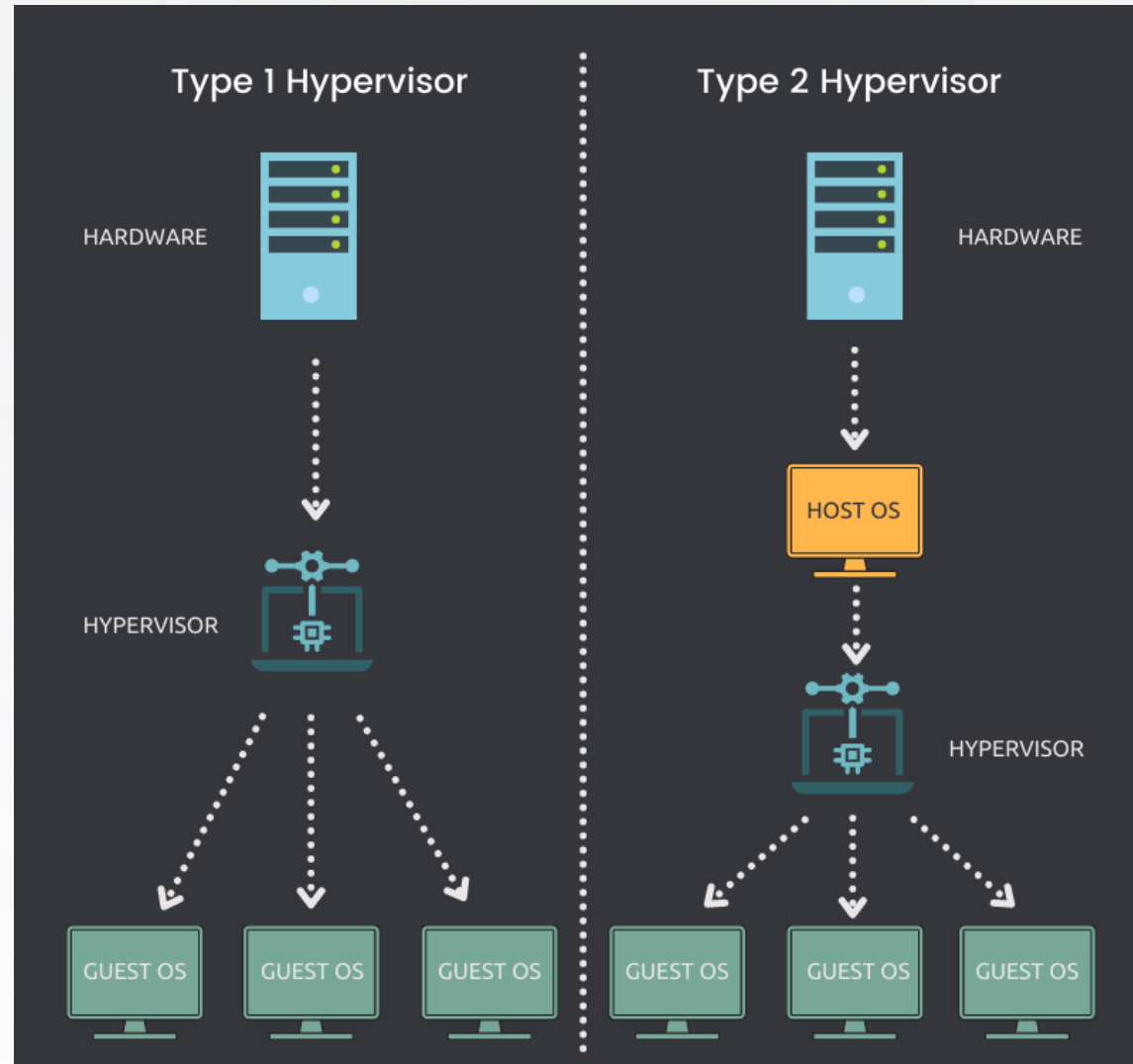


# TYPE 2 HYPERVISOR

- Run as an application on an existing operating system
  - Has to access the hardware resource through the OS
  - Reduced performance due to underlying OS
  - Less scalability than bare-metal hypervisor
  - Less secure since depends on the OS security
  - Used in Desktop/Laptops etc to run alternate OSs and for testing purposes
  - Ex:
    - VirtualBox
    - VMware Workstation
    - Microsoft Virtual PC
- 



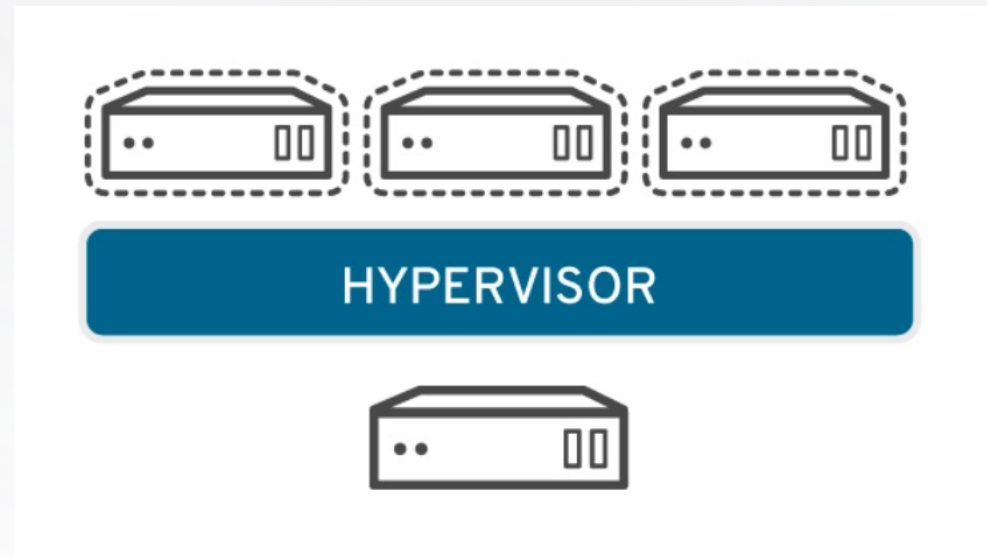
# TYPE 1 AND TYPE 2 HYPERVISORS





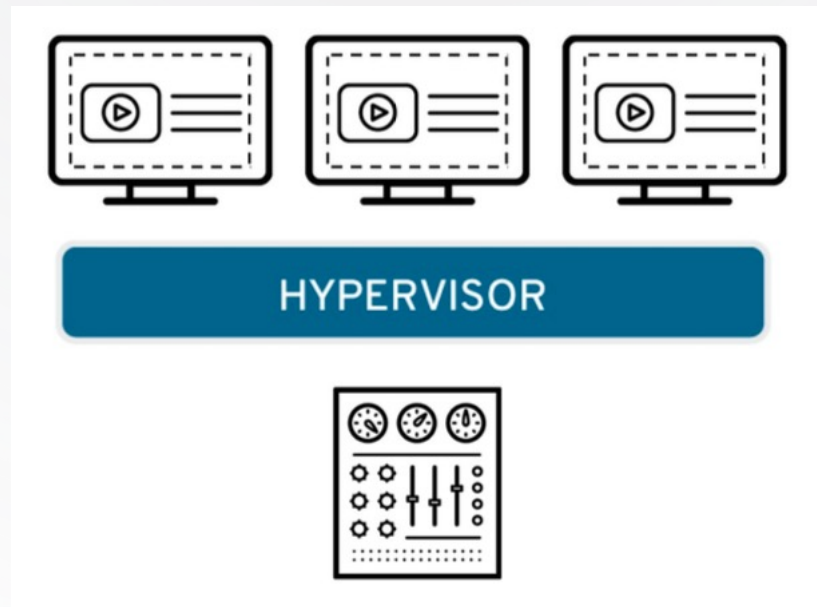
# SERVER VIRTUALIZATION

- Creates multiple virtual machines from one physical server.
- Virtual machines can be installed an its own operating system.



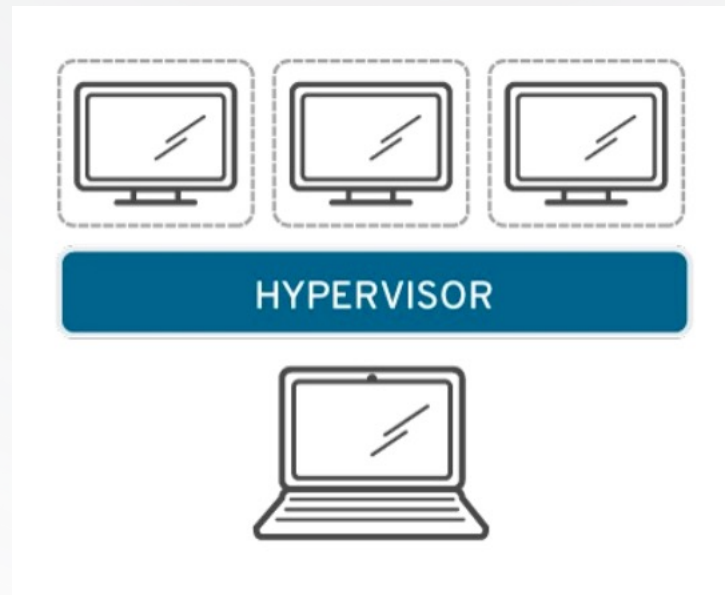
# DESKTOP VIRTUALIZATION

- Simulated desktop environments on a virtualization platform so that it can be remotely accessed.
- Desktop virtualization allows admins to perform mass configurations, updates, and security checks on all virtual desktops



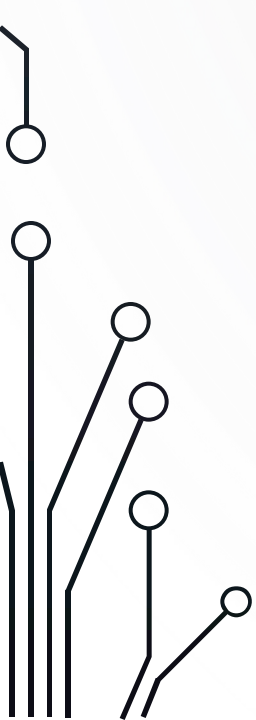
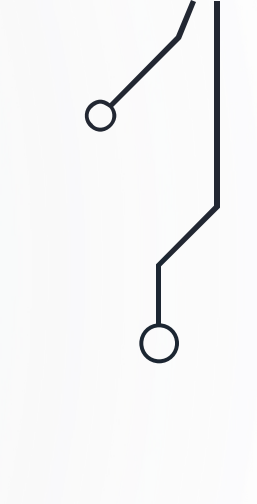
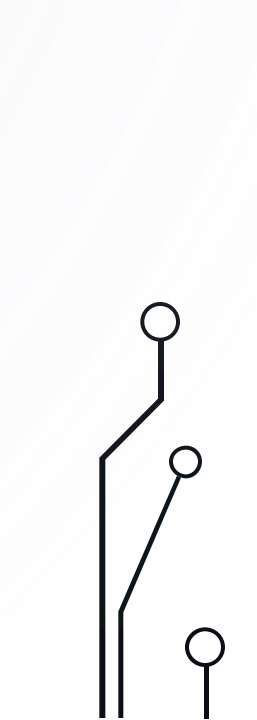
# OPERATING SYSTEM VIRTUALIZATION

- Implemented at the Operating system Kernel.
- Kernel of an OS allows more than one isolated user-space instances
- Ex: Docker/LXC Containers, Jails etc

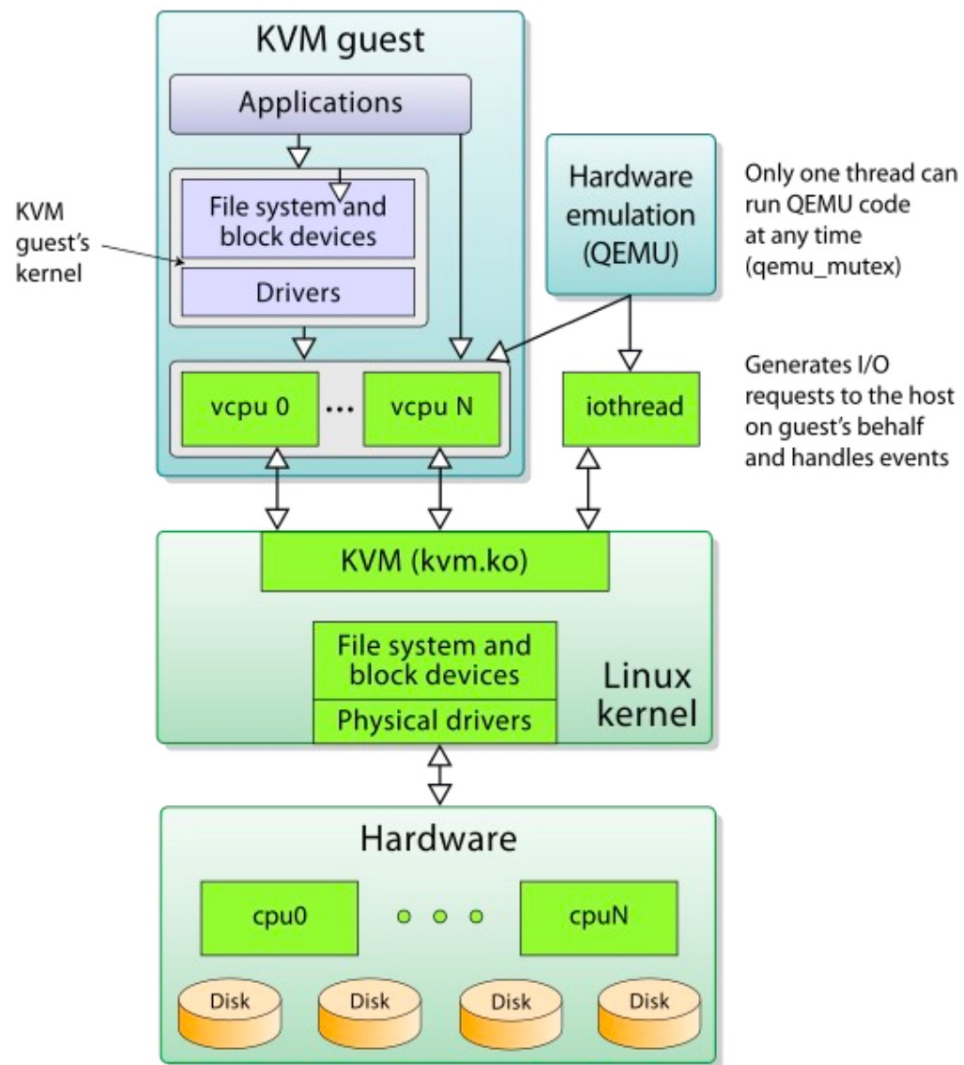




# KVM

- A Virtualization module in the Linux Kernel.
  - Full virtualization solution for Linux on x86 hardware
  - KVM converts Linux into a type-1 (bare-metal) hypervisor
  - Can run multiple virtual machines running Linux or Windows
  - Inherits performance of Linux
  - Supports 32 and 64 bit (on 64 bit hosts) guests
  - Support Snapshots, Live migration (move between two hosts while running)
  - Open source and default in Linux
- 
- 
- 

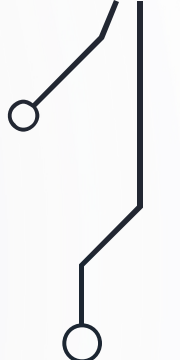
# KVM ARCHITECTURE

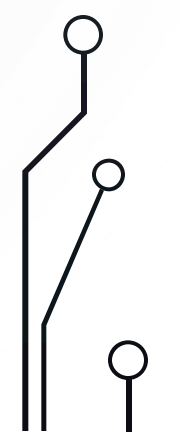






# QEMU

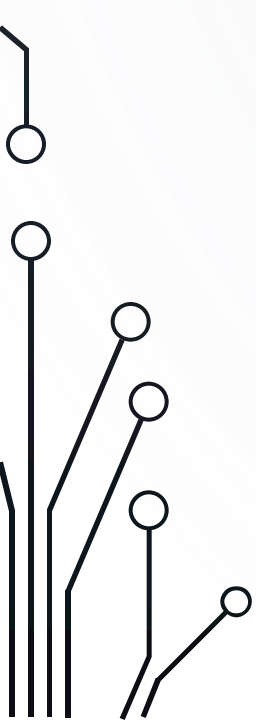
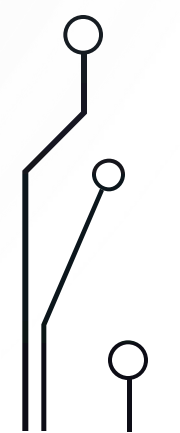
- User space software which emulates disk, network, VGA, PCI, USB, serial/parallel ports, etc.
  - Type 2 hypervisor that runs on operating system
  - QEMU uses KVM as accelerator to access the physical CPU virtualization extensions
  - Supports the emulation of various architectures including x86, MIPS64, ARM, SPARC, RISC-V etc
- 





# SUPPORTED HARDWARE



- An Intel processor with the Intel VT-x and Intel 64 virtualization extensions for x86-based systems
  - An AMD processor with the AMD-V and the AMD64 virtualization extensions.
- 
- 



# SUPPORTED GUEST OS

## Linux


- CentOS, Fedora, RedHat, Debian/Ubuntu, SUSE, Slackware, Android

## Unix

- OpenBSD, FreeBSD, Solaris, Minix

## Windows family

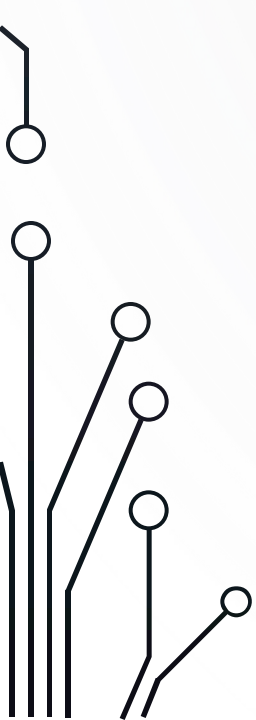
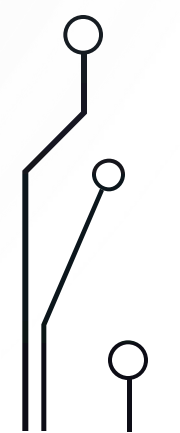
Above is very few from a long list available at [https://www.linux-kvm.org/page/Guest\\_Support\\_Status](https://www.linux-kvm.org/page/Guest_Support_Status)





# SUPPORTED GUEST LIMITS



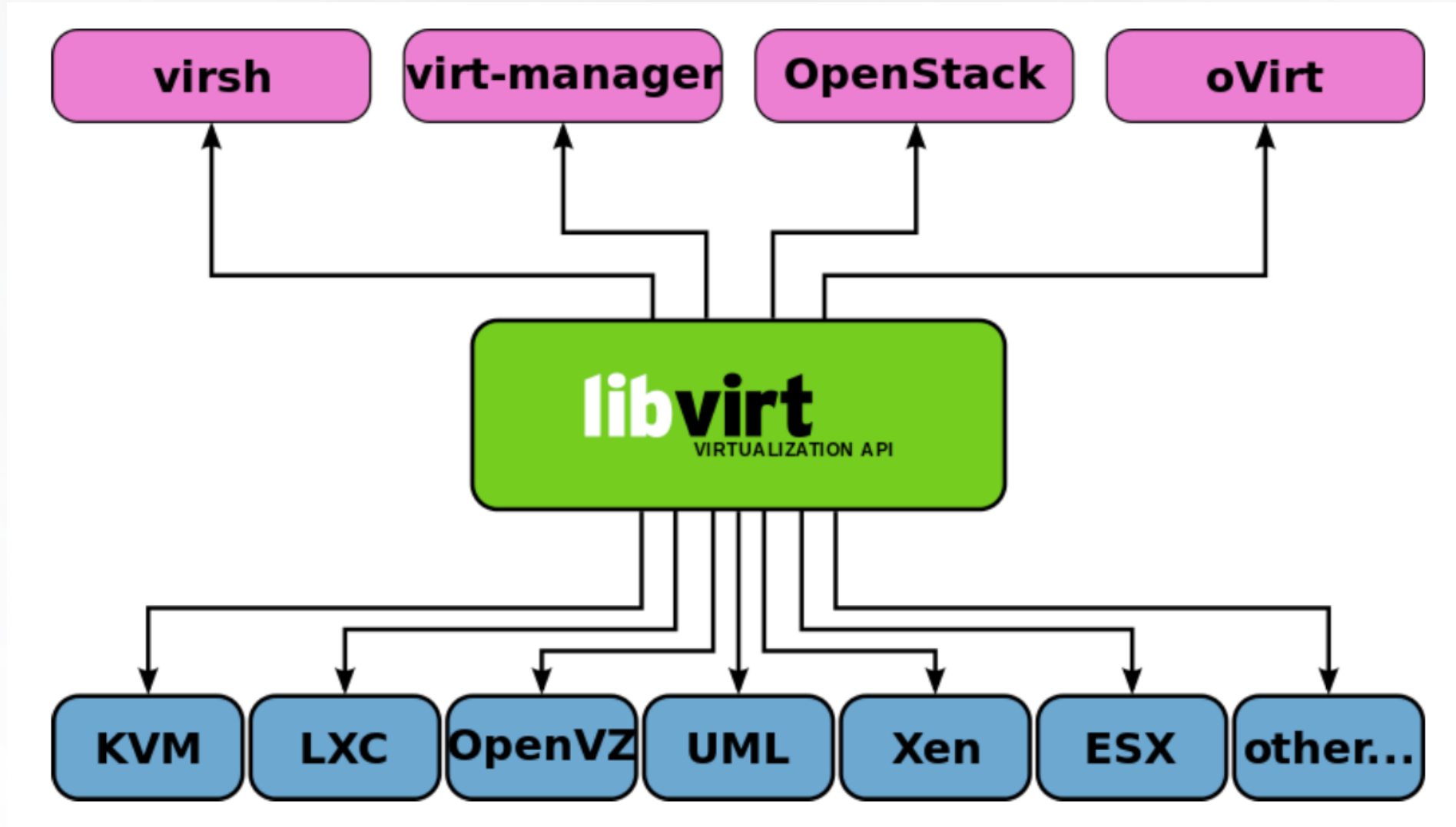
- Max. Guest RAM size – 512 GB
  - Max. Virtual CPUs per guest – 16
  - Max. NICs per guest – 8
  - Max. Block devices per guest – 4 emulated, 20 para-virtual (using virtio- blk)
  - Max. Number of Guest - no more than 8 times the number of CPU cores in the VM Host Server
- 
- 

# LIBVIRT

- A Toolkit to manage hypervisors like KVM, QEMU, Xen, VMWare ESX, LXC and more
- Provides an API to access the virtualization platforms
- API can be accessed with languages C/C++, Python, Perl, PHP, Ruby, Java, Javascript, OCaml (Written in C/C++)
- Used by many applications
  - Command line – virsh, virt-install
  - Continuous integration – Jenkins, BuildBot
  - Desktop – virt-manager, virt-viewer, GNOME Boxes, OpenStack
  - Monitoring – collectd, Nagios-virt

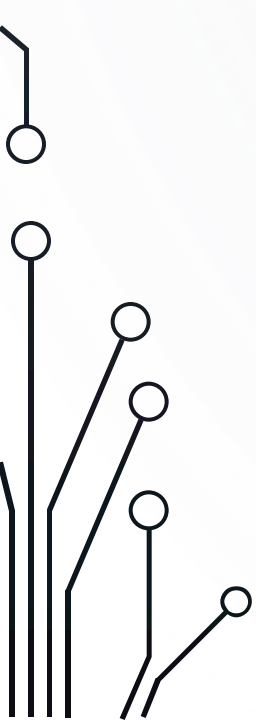
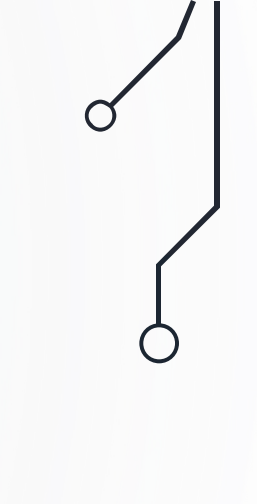
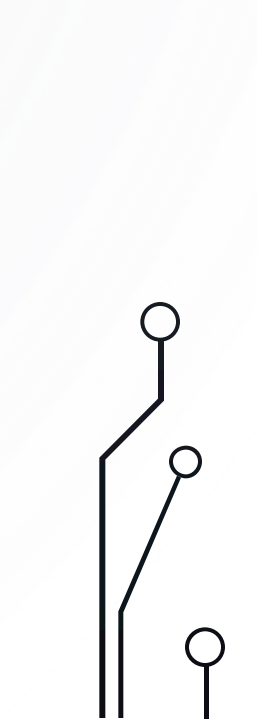


# LIBVIRT

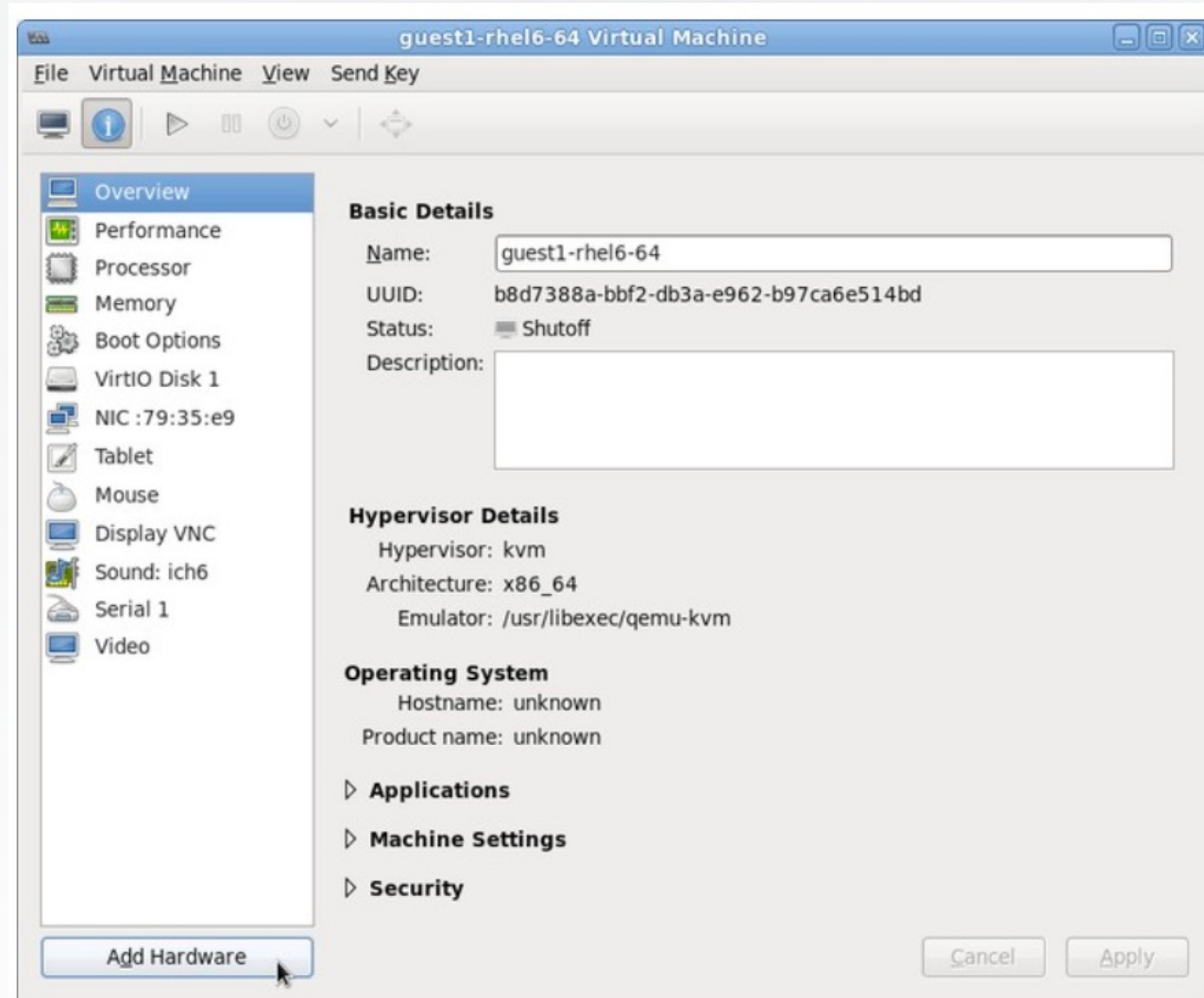


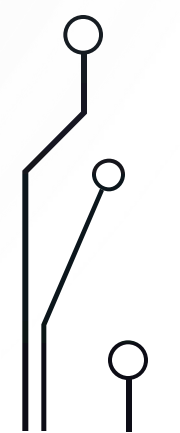
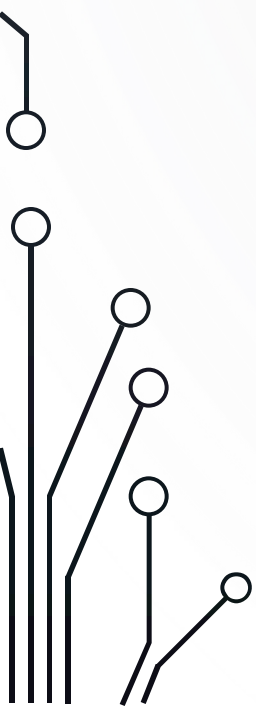
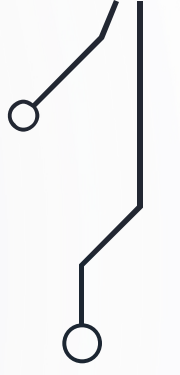
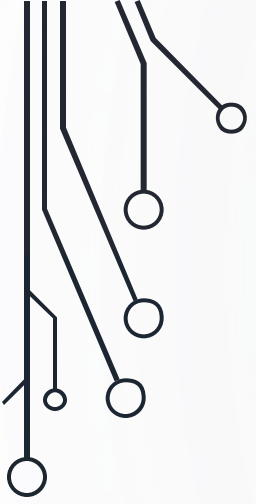


# VIRT-MANAGER

- Desktop user interface for managing virtual machines through libvirt
  - Create, edit, start and stop VMs
  - Adjust the hardware resource allocation and virtual hardware
  - It primarily targets KVM VMs
    - But also manages Xen and LXC
  - It gives summary view of running domains (OS instances), their live performance & resource utilization statistics
  - View and control each VM's console through embedded VNC, SPICE clients.
- 
- 
- 

# VIRT-MANAGER





THANK YOU