DevOps
AIDE501 – M2 – 3 Credits
2026/2027

Course Contents

- **Chapter 1: Cloud Computing**
- **Chapter 2: Containerization**
- **Chapter 3: Shell Scripting**
- **Chapter 4: DevOps**
- **Chapter 5: Orchestration**
- **Chapter 6: GitLab CI/CD**

Chapter 1:

Cloud Computing

Chap.1: Cloud Computing

- Definitions
- Characteristics
- Deployment models: Public, Private, Hybrid
- Service models: IaaS, PaaS, SaaS
- Challenges
- Virtualization
- Storage and computational resources
- Networking & security in IaaS


Virtualization in
Cloud Computing

Virtualization Basics

A **virtual machine** is a software computer that, like a physical computer, runs an operating system and applications.
Each virtual machine contains its own virtual, or software-based, hardware, including a virtual CPU, memory, hard disk, and network interface card.

Virtualization is a technique of how to separate a service from the underlying physical delivery of that service. It is the process of creating a virtual version of something like computer hardware.

With the help of Virtualization, multiple operating systems and applications can run on same machine and its same hardware at the same time, increasing the utilization and flexibility of hardware.

One of the main cost effective, hardware reducing, and energy saving techniques used by cloud providers is virtualization

Virtualization Basics

Virtualization allows to share a single physical instance of a resource or an application among multiple customers and organizations at one time.

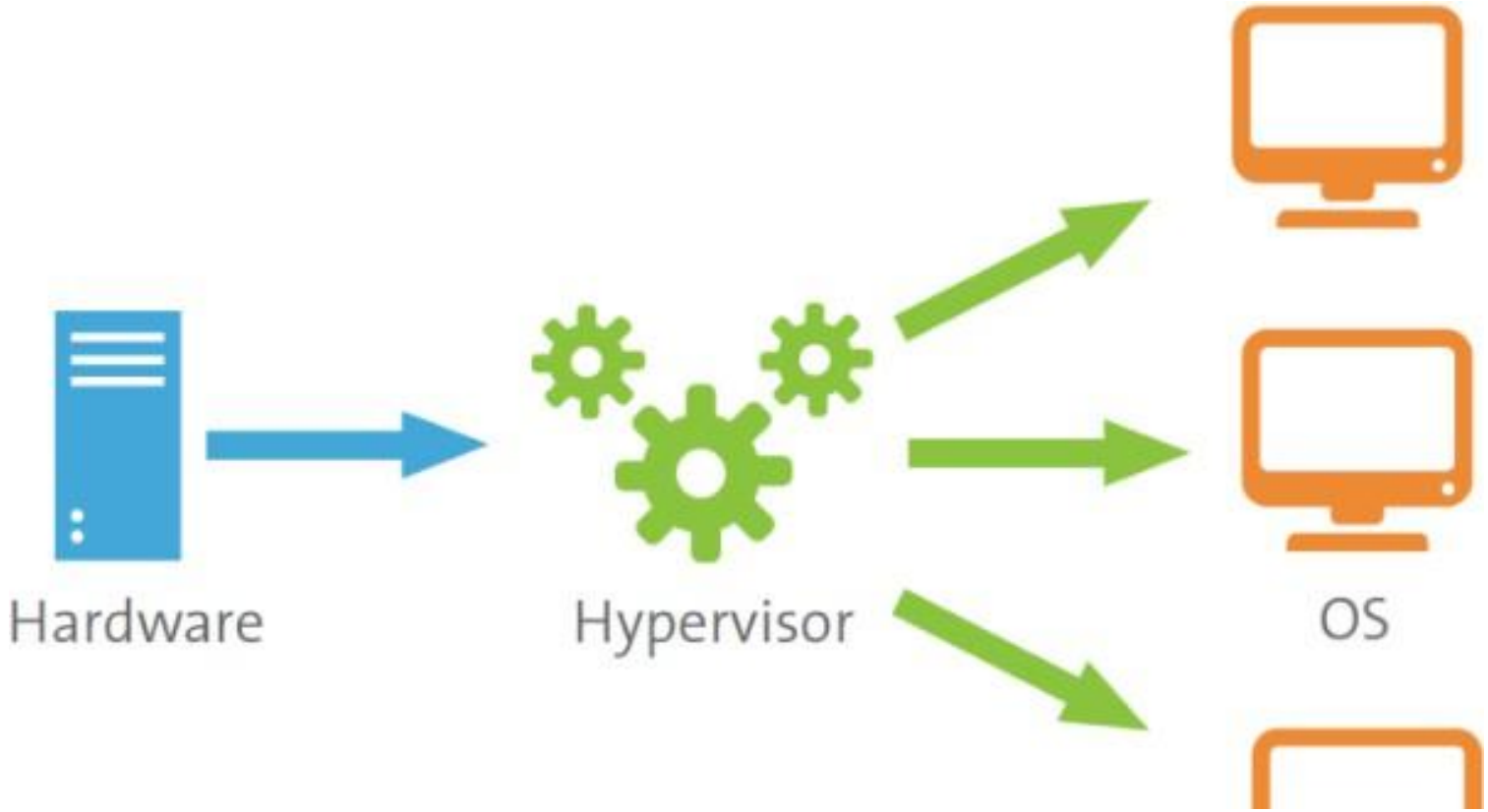
The term virtualization is often synonymous with hardware virtualization, which plays a fundamental role in efficiently delivering Infrastructure-as-a-Service (IaaS) solutions for cloud computing.

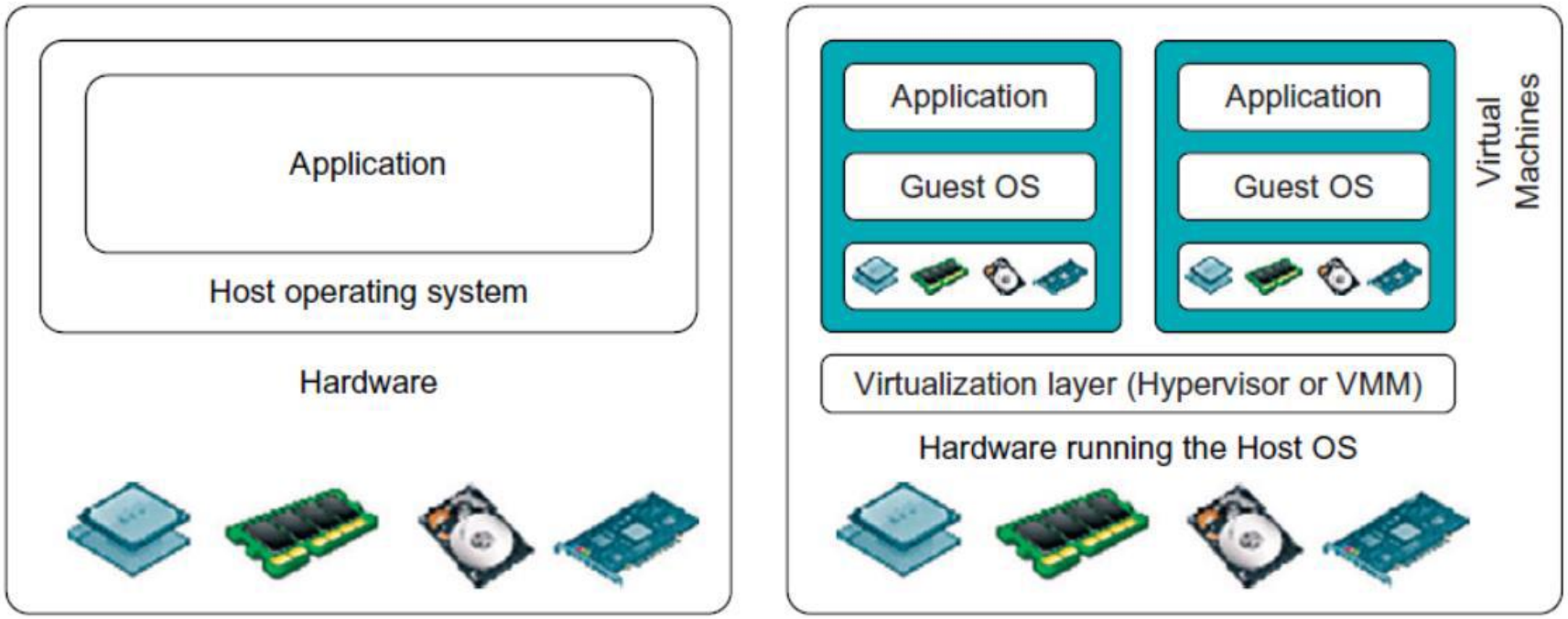
Moreover, virtualization technologies provide a virtual environment for not only executing applications but also for storage, memory, and networking.

The machine on which the virtual machine is going to be built is known as Host Machine and that virtual machine is referred as a Guest Machine.

What is a hypervisor?

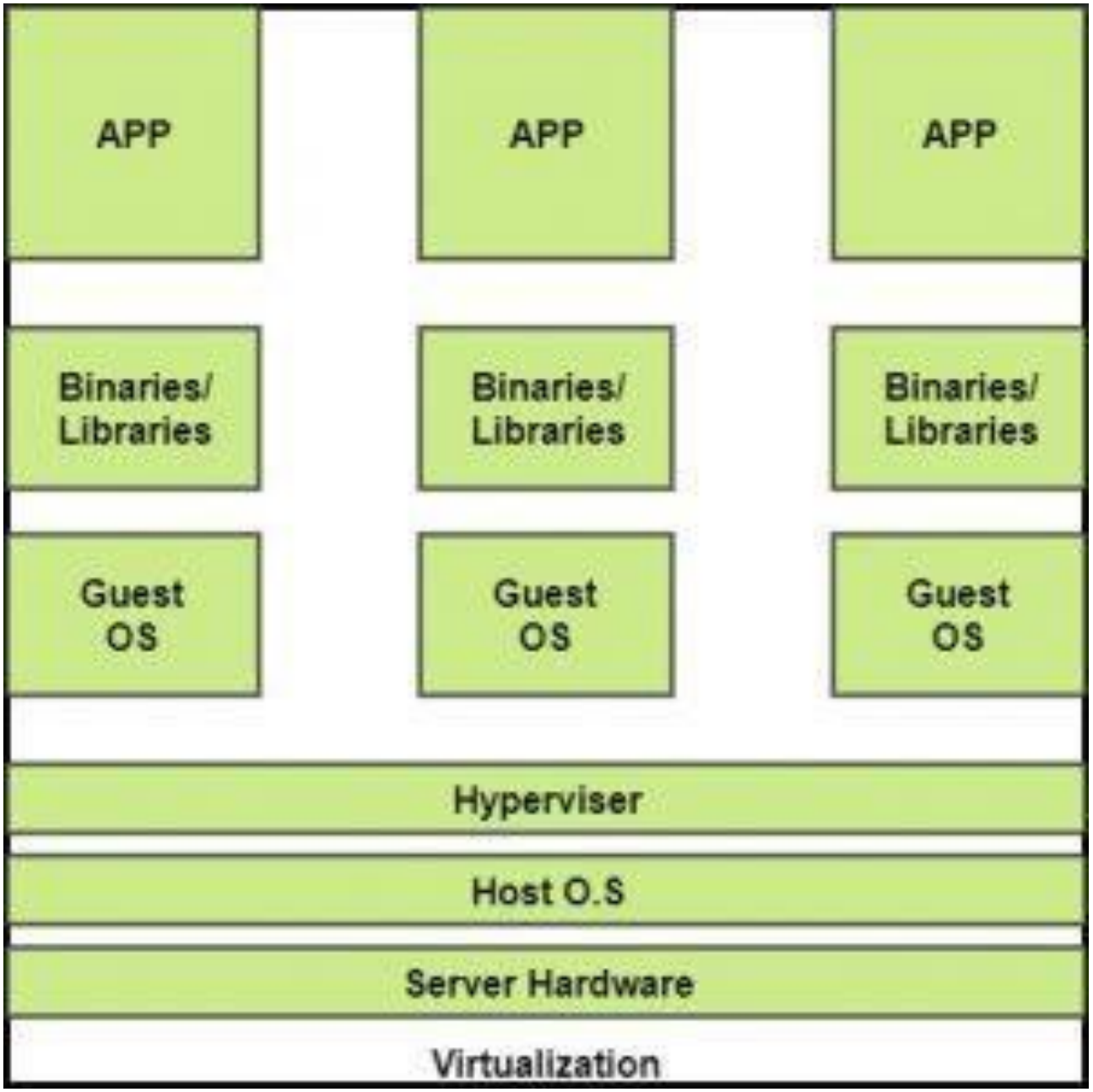
A **hypervisor**, also known as a virtual machine monitor or VMM, is software that creates and runs virtual machines (VMs). A hypervisor allows one host computer to support multiple guest VMs by virtually sharing its resources, such as memory and processing.





(a) Traditional computer

(b) After virtualization


APP
APP
APP
Binaries/
Libraries
Binaries/
Libraries
Binaries/
Libraries
Guest
OS
Guest
OS
Guest
OS
Hyperviser
Host O.S
Server Hardware
Virtualization

Benefits of Virtualization

1. More flexible and efficient allocation of resources.
2. Enhance development productivity.
3. It lowers the cost of IT infrastructure.
4. Remote access and rapid scalability.
5. High availability and disaster recovery.
6. Pay per use of the IT infrastructure on demand.
7. Enables running multiple operating systems.

Virtualization Structures/Tools And Mechanisms

Before virtualization, the operating system manages the hardware.

After virtualization, a virtualization layer is inserted between the hardware and the operating system. In such a case, the virtualization layer is responsible for converting portions of the real hardware into virtual hardware.

Therefore, different operating systems such as Linux and Windows can run on the same physical machine, simultaneously.

Depending on the position of the virtualization layer, there are several classes of VM architectures, namely the hypervisor architecture, para-virtualization, and host-based virtualization.

Hypervisor Types

The hypervisor supports hardware-level virtualization like CPU, memory, disk and network interfaces. The hypervisor provides hypercalls for the guest OSes and applications.

TYPE-1 Hypervisor:
The hypervisor runs directly on the underlying host system. It is also known as a "Native Hypervisor" or "Bare metal hypervisor". It does not require any base server operating system. It has direct access to hardware resources. (Example: Xen Architecture)

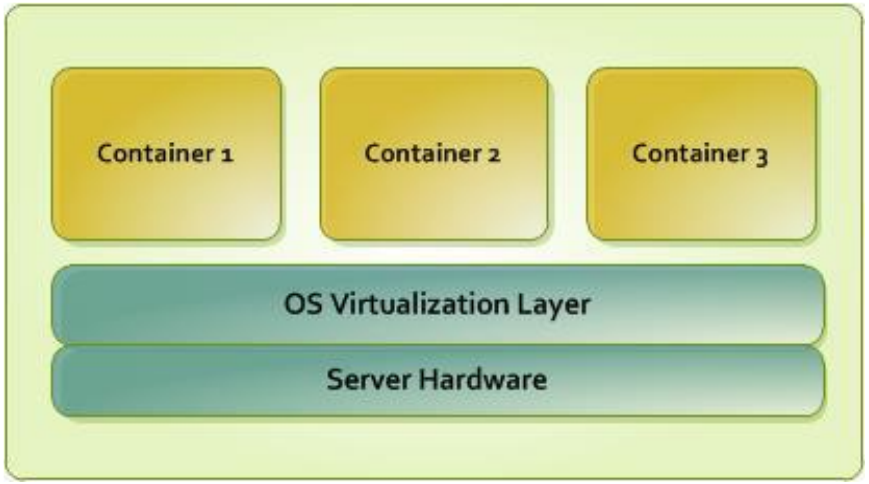
TYPE-2 Hypervisor:
A Host operating system runs on the underlying host system. It is also known as 'Hosted Hypervisor". Such kind of hypervisors doesn't run directly over the underlying hardware rather they run as an application in a Host system (physical machine). (Example: VMware Workstation Pro)

Operating System Virtualization

This refers to an abstraction layer between traditional OS and user applications.

OS-level virtualization creates isolated containers on a single physical server and the OS instances to utilize the hardware and software in data centers.

The containers behave like real servers.



Components of Modern Virtualization

Components of Modern Virtualization

Three main components of modern virtualization come together to create the virtual machines and associated infrastructure that power the modern world.

Virtual Compute: When we talk about virtualized compute resources, we mean the CPU – the "brain" of the computer.

In a virtualized system, the Hypervisor software creates virtual CPUs in software and presents them to its guests for their use.

The guests only see their own virtual CPUs, and the Hypervisor passes their requests on to the physical CPUs.

Components of Modern Virtualization

The benefits of **Virtual Compute** are immense.

Because the Hypervisor can create more virtual CPUs (often creating many more than the number of physical CPUs), the physical CPUs spend far less time in an idle state, making it possible to extract much more performance from them.

In a non-virtualized architecture, many CPU cycles (and therefore electricity and other resources) are wasted because most applications do not run a modern physical CPU at full tilt all the time.

Components of Modern Virtualization

Goal of CPU virtualization: Run all instructions of Guest OS.

All instructions either trap or execute identically. Instructions that access privileged state trap. Non-critical instructions are executed identically.

To have an efficient CPU virtualization, the VMM should be implemented to Shift traditional OS from Kernel mode to user mode and run the VMM in kernel mode. The VMM is now able to intercept all trapping events.

It emulates the effect of that specific instruction or action without carrying it out. In this way, the host OS is not affected by the guest's actions. This is called trap and emulate.

Components of Modern Virtualization

- The VMM still need to:
  - Multiplex VMs on CPUs.
    - Time-slice the VMs, each VM runs OS/Apps
      - Use simple CPU scheduler:
        - Round robin, work-conserving (give extra to other VM)
        - Can oversubscribe and give more #VCPUs than actual

Components of Modern Virtualization

Virtual Networking: Network virtualization uses software to abstract away the physical network from the virtual environment.

The Hypervisor may provide its guest systems with many virtual network devices (switches, network interfaces, etc.) that have little to no correspondence with the physical network connecting the physical servers.

Within the Hypervisor's configuration, there is a mapping of virtual to physical network resources, but this is invisible to the guests.

If two virtual computers in the same environment need to talk to each other over the network, that traffic may never touch the physical network at all if they are on the same host system.

Components of Modern Virtualization

Virtual Storage: In the beginning, there were disk drives. Later, there was RAID (Redundant Array of Independent Disks). RAID allowed us to combine multiple disks into one, gaining additional speed and reliability.

Later still, the RAID arrays that used to live in each server moved into specialized, larger storage arrays which could provide virtual disks to any number of servers over a specialized high-speed network, known as a SAN (Storage Area Network).

All these methods are simply different ways to virtualize storage.

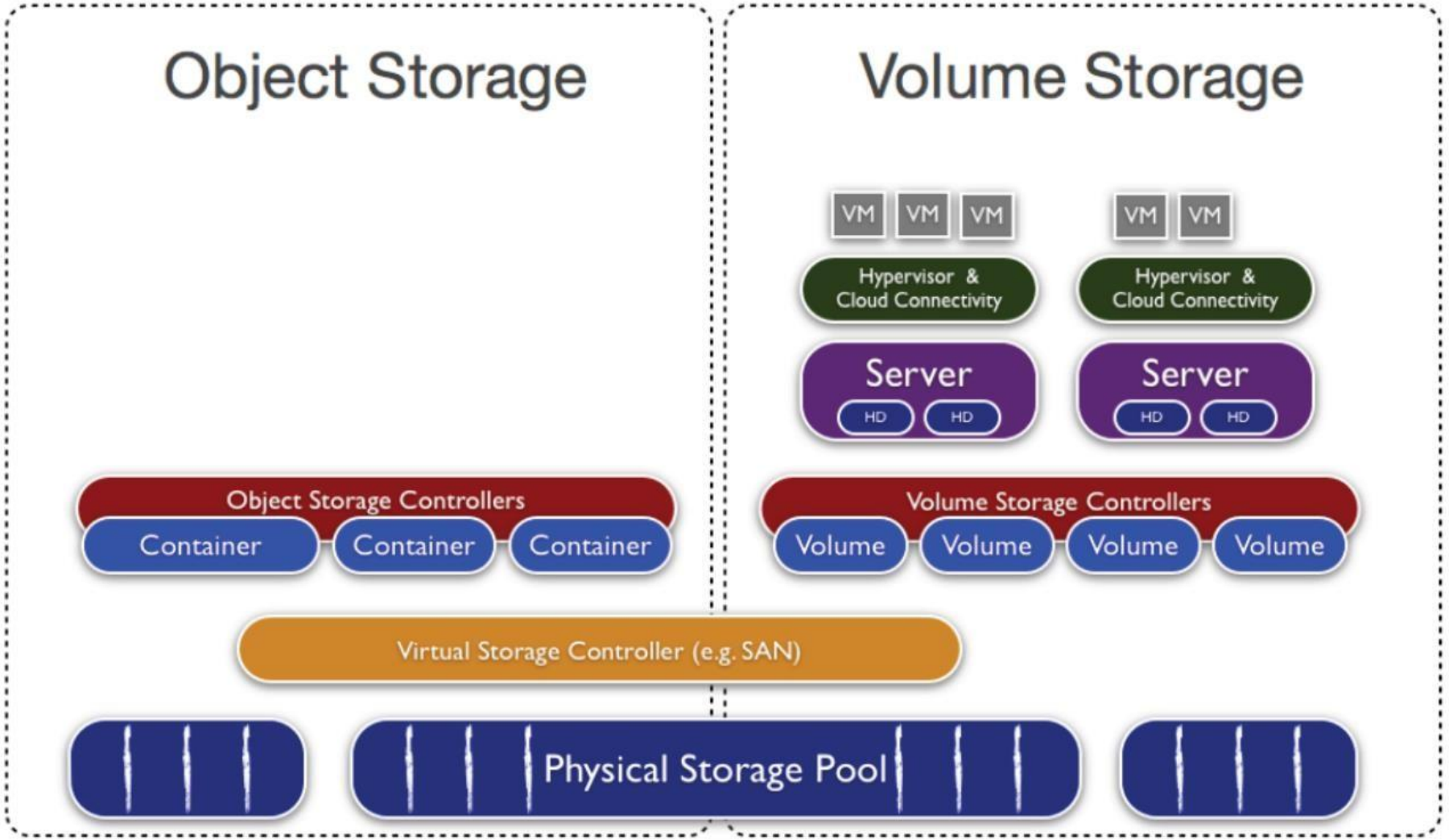
Over the years, the trend has been to abstract away individual storage devices, consolidate them, and distribute work across them.

Storage Type in IaaS

The data in the cloud are in most cases present in the storage.

Three types of storage exist in the cloud:

- **Object storage**, which is a file repository and is used to store individual files ('objects'). Object storage is accessed through an API rather than a network share, which opens a wealth of new uses.
- **Volume Storage**, which is a virtual hard drive and where the volumes attach to virtual machines for use just like a physical hard drive or array.
- **File Storage**, Data is organized as files inside directories which are inside other directories.


Object Storage
Volume Storage
VM
VM
VM
VM
VM
Hypervisor &
Cloud Connectivity
Hypervisor &
Cloud Connectivity
Server
HD
HD
Server
HD
HD
Object Storage Controllers
Container
Container
Container
Volume Storage Controllers
Volume
Volume
Volume
Volume
Virtual Storage Controller (e.g. SAN)
Physical Storage Pool

Storage Type in IaaS

- **Physical storage** might be hard drives in servers or SSD drives in datacenter servers.
- It is pooled by a *virtual storage controller,* like a SAN (Storage Area Network).
- On top of the physical storage, we build object and volume storage(block) and File Storage (Network File Storage)

Object Storage

- **Object storage controllers** connect to assigned physical or virtual storage and manage connectivity. The access to objects is through APIs and not a file system .

- Object storage controllers create virtual *containers* which are assigned to cloud users.

- A container is a pool of storage in which you can place object (files). Every container stores each bit in multiple locations. This is called *data dispersion*.

Object Storage

- ✓ Data itself
- ✓ Metadata
- ✓ A unique id
- ✓ Ideal for unstructured data
- ❖ Use cases:
  - Storing images, videos, documents
  - Backups and archives
  - Big data analytics
  - Static website hosting
- ❖ Examples:
  - Amazon S3
  - Google Cloud Storage
  - Azure Block storage

Volume Storage

- **Volume storage controllers** connect to assigned physical (or virtual) storage to manage connectivity.
- The controller creates volumes on request and assigns them to specific cloud instances.
- In traditional virtualization terms, it creates a virtual hard drive and connects it to VM, usually over the network.
- A **volume** is essentially a virtual hard drive assigned to a VM until it is destroyed and automatically deallocates its volumes and returns their capacity to the free storage pool.

Volume Storage

- Physical servers typically have local hard drives which can be assigned to the volume controller to expand the storage pool or used locally for non-persistent storage (**Local ephemeral storage** [which is temporary tied to the virtual machine. When the workload stops, crashes, or is rescheduled, all data stored on it is automatically deleted]).

- When you shutdown your instance this data is always lost, although it might be recoverable until overwritten.

- Volumes may be distributed, with data dispersed across multiple physical drives. They are connected to virtual machines over the network.

Volume Storage

- ✓ Fixed size blocks
- ✓ Acts as a raw disk (like C or D drive)
- ✓ Requires a file system (NTFS)
- ✓ Data is accessible only when it is attached to a running VM
- ❖ Use cases:
  - DB (MySQL, ..)
  - OS storage
  - Applications fast read/write access
- ❖ Examples:
  - Amazon EBS (Elastic Block Store)
  - Google Persistent Disks
  - Azure Managed Disks

How Object and Volume Storage Interact

Most clouds include both object and volume storage; Here are the key uses:

- A ***snapshot*** is an instant backup of a volume, moved into object storage.

- A snapshot effectively copies a complete set of the storage blocks in your volume into a file in an object container.

- Generally, every block in your volumes is stored in multiple physical locations, typically 3 or more times, so taking a snapshot tells the volume controller to copy a complete set of blocks over to object storage.

How Object and Volume Storage Interact

- ***Images*** are pre-defined storage volumes in object storage, which contain operating systems and other virtual hard drives used to launch instances.

- When you launch an instance the volume controller creates a volume of the required size, then copies the requested image from the object controller into the virtual machine.

File Storage

- ✓ **File Storage** stores data in a hierarchical structure of files and folders just like on your pc.
- ✓ It uses NFS (Network File Storage) or SMB (Server Message Block)
- ✓ Data is organized as files inside directories which are inside other directories.
- ✓ It can be mounted by multiple servers at once.
- ✓ You access data with a file path
- ❖ Use Cases :
  - Shared content repositories
  - Media and CMS
  - Home directories and user file shares
  - Web servers needing shared file access
- ❖ Examples:
  - Amazon EFS (Elastic File System)
  - Azure Files
  - Google FileStore

Challenges in IaaS

IaaS Risks

The architecture of cloud storage models create new and interesting risks:

- Cloud administrators can access any data stored in the cloud over the network.
- Snapshots are accessible over the network. They pose a significantly increased risk of exposure compared to traditional infrastructure.
- All this is managed with networks and APIs. Someone accessing a cloud administrator's or developer's system could access an entire (virtual) datacenter.
- There is very little visibility into where things are actually stored, although some cloud platforms are beginning to offer more transparency.

IaaS Risks

- A snapshot is a near-instant backup of a (virtual) hard drive that is incredibly portable and easily made public.
- It is possible to write a script that, if run on a cloud administrator's computer, would snapshot every single volume that administrator could access and make the snapshots public.
- A few API calls from an unprotected developer or administrator system could expose all the data in your cloud.
- If you allow instances to store data in local ephemeral storage, sensitive data such as encryption keys may be left behind when you move or terminate an instance.

IaaS Risks

Here are a few examples of how life changes:

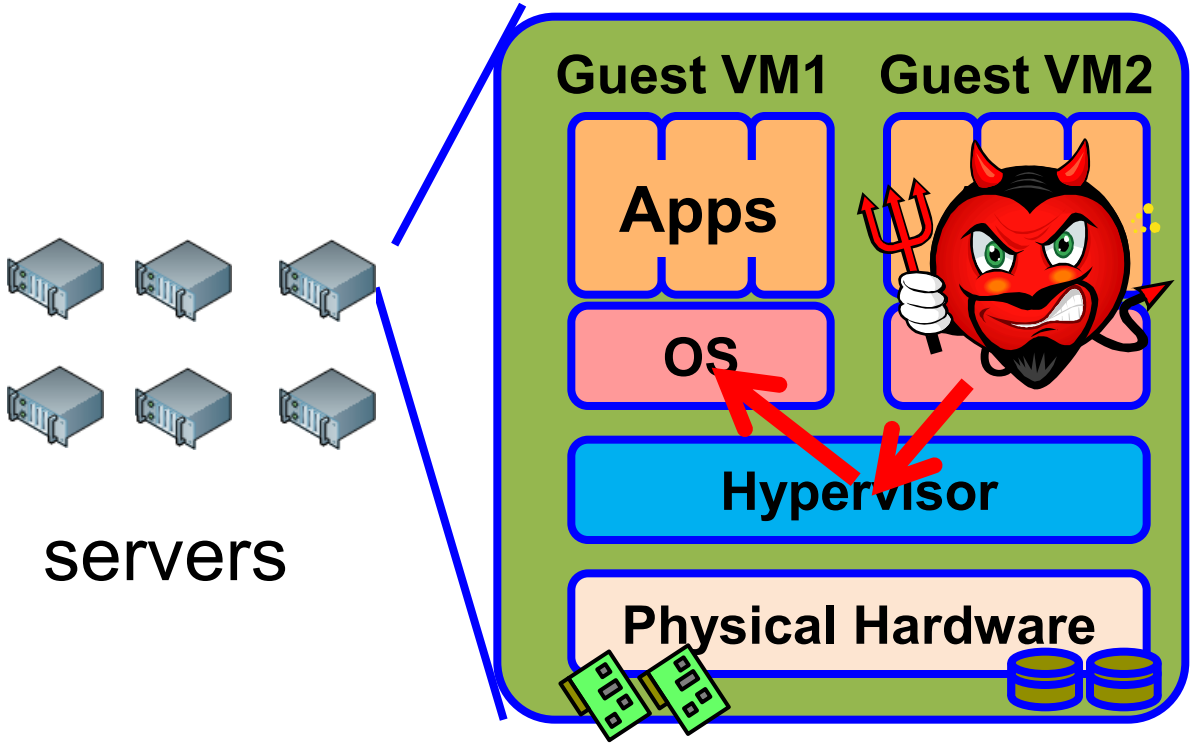
- In public clouds, an administrator at your cloud provider *could* access your virtual hard drives. This would violate all sorts of policies and contracts, but it is still technically possible.
- In most IaaS clouds, a single command or API call can make an instant copy (snapshot) of an entire virtual hard drive, and then move it around your environment or make it public on the Internet.
- If your data is on the same hard drive as a criminal organization using the same cloud provider, and 'their' hardware are considered as part of an investigation, your data could be exposed. Yes, this has happened.

IaaS Risks

- In IaaS, we still manage traditional virtualized networks, computers, and storage.
- We ‘boot’ computers (launch instances), assign IP addresses, and connect (virtual) hard drives.
- While the IaaS resembles to traditional infrastructure, the architecture behind is different.
- For both public and private clouds, the architecture of the physical infrastructure is completely different so that the security must be differently managed.
- The cloud is not inherently *less* secure than traditional infrastructure, but it is very *different*.

IaaS Risks

- Malicious software can run on the same server:
  - Attack hypervisor
  - Access/Obstruct other VMs



Cloud Challenges at Communication level

1. Shared communication infrastructure

Users on the cloud are usually granted with the super-user access for the purpose of managing their VMs. The access capability empowers the malicious user to acquire system IP or MAC addresses and make malicious use of IaaS network interfaces, such as, sniffing and spoofing over the real network.

2. Virtual networks

In cloud computing systems, the communication takes place not only on real networks but virtualized networks. Security and protection mechanisms over the physical network cannot monitor the traffic over virtualized network. The share among multiple VMs causes the possibility of certain attacks, such as, Denial of Service (DoS), spoofing and sniffing of virtual network.

3. Security misconfigurations

Cloud Challenges at Virtualization level

1. **VM image sharing**
   - Users are allowed to upload and download images from the repository
   - A malicious user can investigate the code of the image to look for probable attack point
   - A malicious user can upload an image that contains a malware

2. **VM isolation**
   - Isolation is not only needed on storage devices, but Network, memory and computational hardware also needs isolation of VMs

Cloud Challenges at Virtualization level

3. **VM escape**
   - VM escape is a situation in which a malicious user or VM escapes from the control of VMM or hypervisor, which is a software component that manages all the VMs and their access to the hardware.
   - The VM escape situation can provide attacker access to other VMs or can bring the VMM down.
   - This can happen by exploiting vulnerabilities in the VMM software

4. **Hypervisor issues**
   - The metadata of the VMs, kept by the VMM, may also be exposed to an attacker if the attacker takes control of a VMM.
   - There are many reported bugs in the VMM that let the attacker to take control of the VMM or bypass security restrictions.
   - For example, vulnerabilities in the Xen, Microsoft Virtual PC, and Microsoft Virtual Server can be abused by attackers to gain privileged rights.

Cloud Challenges at Storage level

1. **Data privacy and integrity.**
   - In a shared environment, the security strength of the cloud equals its weakest entity. A **successful attack on a single entity will result in unauthorized access to the data of all the users**.
   - **Employee of SaaS providers**, having access to information may also act as a potential risk.
   - Due to virtualization physical resources are shared among multiple tenants. This eventually may allow malicious users to launch attacks on the data of other users while in processing phase.

Cloud Challenges at Storage level

2. **Data recovery vulnerability**
   - The resource allocated to a particular user may be assigned to the other user at some later point of time.
   - In case of memory and storage resources, a malicious user can employ data recovery techniques to obtain the data of previous users.
   - Some ethical hackers were able to recover Amazon machine images files 98% of the times.

3. **Data backup**
   - A regular data backup is needed at the CSP side to ensure the availability and recovery of data in case of intentional and accidental disasters.
   - the backup storage also needs to be protected against unauthorized access and tampering

IaaS Security Solutions

❖ ***Encryption*** is the best tool available for most cloud data security.

Implemented properly, encryption protects data as it moves through your environment.

It doesn't matter if there are 3 versions of a particular block exposed on multiple hard drives because without the key, they are *all* meaningless.

It doesn't matter if someone makes a snapshot of an encrypted volume public.

Only exposure of data *and* its associated keys is problematic.

Encryption System

❖ Three major components define the overall structure of an encryption system:

- **The data:** The objects to encrypt.
- **The encryption engine:** This component handles actual encryption and decryption operations.
- **The key manager:** This handles keys and passes them to the encryption engine as authorized



Volume Encryption

Encrypting Entire Volumes

As a reminder, volume encryption protects from the following risks:.

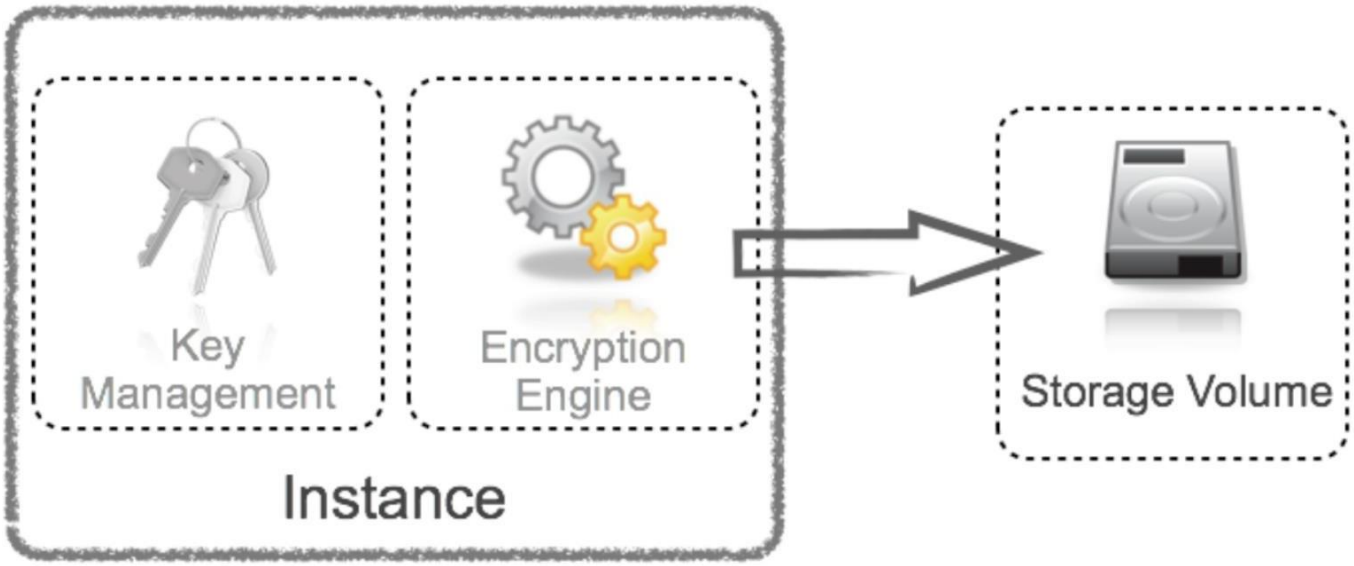
- Protects volumes from snapshot cloning/exposure.
- Protects volumes from examination by the cloud provider, including cloud administrators.
- Protects volumes from being exposed by physical drive loss.

Volume Encryption

IaaS volumes can be encrypted three ways:

- **Instance-managed encryption:** The encryption engine runs within the instance, and keys are stored in the volume protected by a passphrase or key pair.
- **Externally-managed encryption:** The encryption engine runs in the instance, but keys are managed externally and issued to instances on request.
- **Proxy encryption:** You connect the volume to a special instance or appliance/software to perform the encryption operations; then you connect the application instance to the proxy for the unencrypted data. The proxy handles all cryptographic operations and may keep keys either onboard or externally.

Volume Encryption: Instance-managed encryption



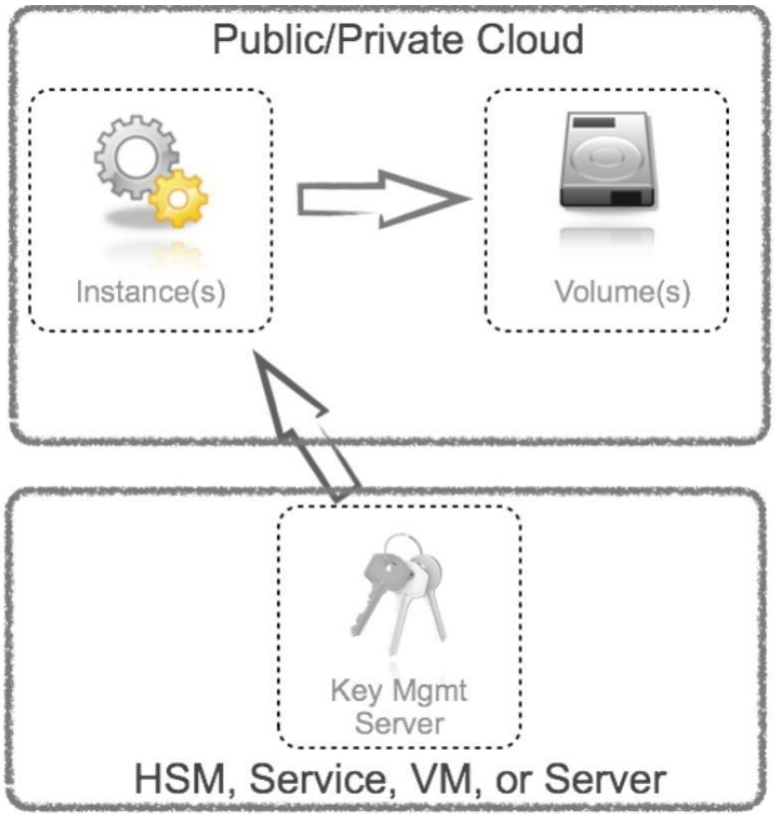
The encryption engine runs inside the instance

You connect a second *new* storage volume

You log into your instance, and using the encryption engine you encrypt the new storage volume

Volume Encryption: Externally-managed encryption

Externally-managed encryption is like instance-managed except that the keys are handled outside the instance in a key management server, service, or Hardware Security Module (HSM).

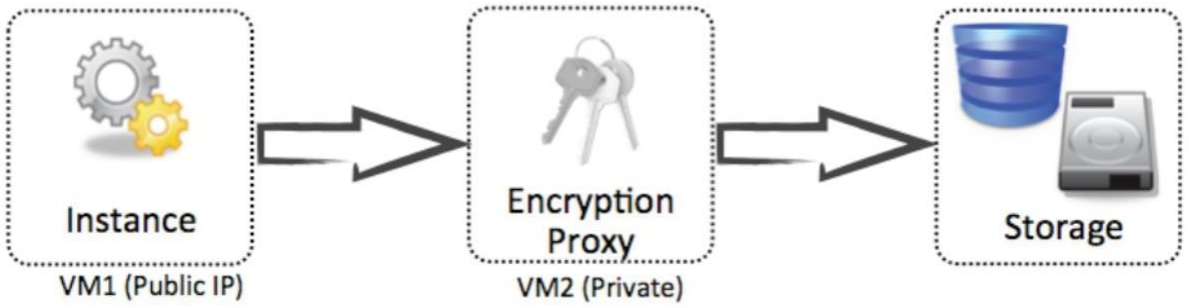


Deployment and feature options

1. An HSM or other hardware key management appliance
2. Key management software
3. Key management Software as a Service (SaaS)

Volume Encryption: Proxy encryption

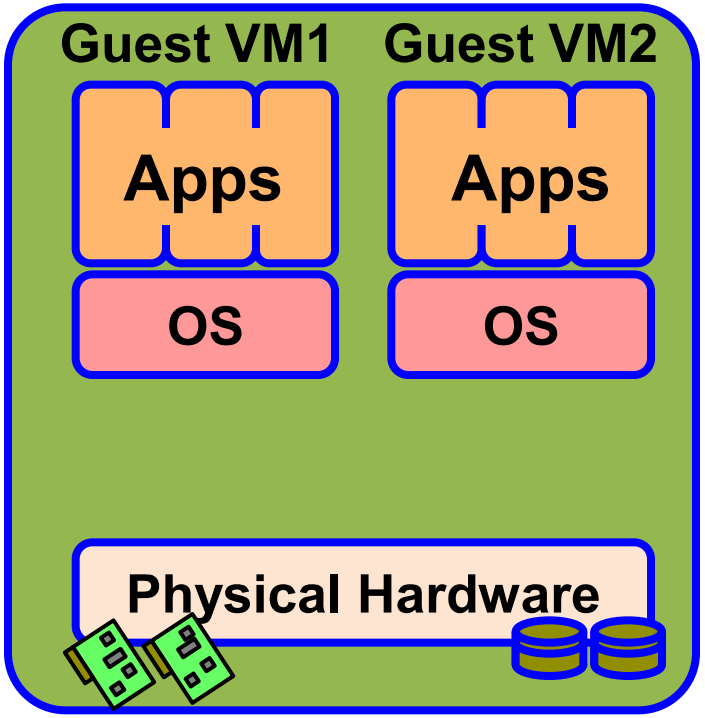
The proxy encryption option uses an inline software encryption proxy to encrypt and decrypt data.



IaaS Security Solutions

- NoHype removes the hypervisor
  - One VM per Core: Each processor core is dedicated exclusively to a single virtual machine during its runtime.
  - There's nothing to attack but creates other vulnerabilities.
  - Direct Hardware Access: VMs interact directly with pre-allocated CPU, memory, and I/O devices without going through a software translation/arbitration layer.

No hypervisor →



IaaS Security Solutions

Data dispersion: A technique used to improve data security, but without the use of encryption mechanisms.

This sort of algorithms can provide high availability and assurance for data stored in the cloud, by means of data fragmentation, and are common in many cloud platforms.

In a fragmentation scheme, a file **f** is split into **n** fragments, all of these are signed and distributed to **n** remote servers. The user then can reconstruct **f** by accessing **m** arbitrarily chosen fragments.

When fragmentation is used along with encryption, data security is enhanced: an adversary must compromise **m** cloud nodes to retrieve **m** fragments of the file **f**, and then must break the encryption mechanism being used.