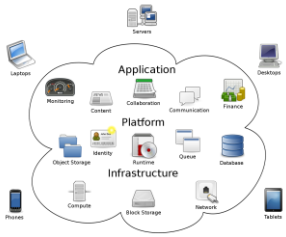


UNIT II

Cloud Computing

Learning Objectives

- Virtualization in Cloud
  - Virtualization
  - Applications and Advantages of Virtualization
  - Implement virtualization and Techniques of virtualizations
  - Middleware virtualization
  - Hardware Virtualizations
  - Types of Virtualization
- Security Issues in Cloud Computing
  - Security in cloud computing
  - Security Challenges in Cloud Computing
  - Information Security
  - Privacy and Trust in Cloud Computing

Virtualization

- Virtualization uses **software to create an abstraction layer over computer** hardware that allows
  - the hardware elements of a single computer—
    - Processors
    - Memory
    - storage and more
  - to be divided into multiple virtual computers, commonly called virtual machines (VMs)
- Each VM runs its own operating system (OS) and behaves like an independent computer
- it is running on just a portion of the actual underlying computer hardware

Virtual Machine

- Virtual machines (VMs) are virtual environments that simulate a physical compute in software form
  - VM's configuration
  - the storage for the virtual hard drive
  - some snapshots of the VM that preserve its state at a particular point in time
- A virtual machine is a virtual representation, or emulation, of a physical computer. They are often referred to as a guest while the physical machine they run on is referred to as the host.
- Virtualization makes it possible to create multiple virtual machines, each with their own operating system (OS) and applications, on a single physical machine

Virtual Machine

- A VM cannot interact directly with a physical computer. Instead, it needs a lightweight software layer called a *hypervisor* to coordinate between it and the underlying physical hardware
- The **hypervisor allocates physical computing resources**—
  - Processors
  - Memory
  - Storage
- ***Hypervisor*** keeps each VM separate from others so they don't interfere with each other

Hypervisors

- A hypervisor is the software layer that coordinates VMs
- It serves as an interface between the VM and the underlying physical hardware, ensuring that each has access to the physical resources it needs to execute
- Hardware Resources
  - Processors
  - Memory
  - Storage
- It also ensures that the VMs don't interfere with each other by impinging on each other's memory space or compute cycles.

Hypervisors -Types

- **Type 1 or "bare-metal" hypervisors**
  - interact with the underlying physical resources
  - replacing the traditional operating system altogether
  - They most commonly appear in virtual server scenarios.
  - For Example, VMWare ESXi 6.5
- **Type 2 hosted hypervisors**
  - 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 resources.

Hypervisors -Type 1 vs. Type 2

- **Type 1 or "bare-metal" hypervisors**
  - **Pros**: highly efficient (direct access to physical hardware), secure (nothing in between them and the CPU that an attacker could compromise)
  - **Cons**: Often needs a separate management machine to administer different VMs and control the host hardware.
- **Type 2 hypervisors**
  - **Pros**: quick and easy access to an alternative guest OS, great for end-user productivity, allow consumer to access their favorite Linux-based development tools.
  - **Cons**: access computing, memory, and network resources via the host OS, which has primary access to the physical machine, latency issues, affecting performance, potential security risks

Hypervisors -Characteristics

- Performance:
  - bare-metal hypervisors should support guest OS performance close to native speeds
- Ecosystem:
  - good documentation and technical support to implement and manage hypervisors across multiple physical servers at scale
  - a healthy community of third-party developers that can support the hypervisor with their own agents and plugins that offer capabilities, such as backup and restore capacity analysis and fail-over management
- Management tools:
  - provision the VMs, maintain them, audit them, and clean up
  - the vendor or third-party community supports the hypervisor architecture with comprehensive management tools
- Live migration:
  - This enables you to move VMs between hypervisors on different physical machines without stopping them
- Cost:
  - the cost and fee structure involved in licensing hypervisor technology

Hypervisors -Examples

- VM-Ware
  - ESXi hypervisor (T1)
  - VSphere hypervisor (T1)
  - VMware Fusion (T2)-MacOS
  - Workstation (T2)
  - VirtualBox (T2)
- Microsoft
  - Hyper-V hypervisor (T1)
- IBM Xen Server (Open Source)
  - Citrix Hypervisor under Xen Server (T1)
- Linux Based (Open Source Hypervisor)
  - KVM (Kernel-based VM)

Virtualization-Types

- Desktop virtualization:
  - run multiple desktop operating systems, each in its own VM on the same computer.
- Network virtualization
  - abstracts hardware elements and functions (e.g., connections, switches, routers, etc.) and abstracts them into software running on a hypervisor
  - software-defined networking (SDN), which virtualizes hardware that controls network traffic routing, and network function virtualization (NFV), which virtualizes one or more hardware appliances that provide a specific network function
- Storage virtualization
  - enables all the storage devices on the network— whether they're installed on individual servers or standalone storage units—to be accessed and managed as a single storage device
- Data virtualization
  - data from multiple applications, using multiple file formats, in multiple locations, ranging from the cloud to on-premise hardware and software systems
- Application virtualization
  - Local application virtualization
  - Application streaming:
  - Server-based application virtualization

Virtualization-Types

- Data center virtualization
  - abstracts most of a data center's hardware into software, effectively enabling an administrator to divide a single physical data center into multiple virtual data centers for different clients
- CPU virtualization
  - the fundamental technology that makes hypervisors, virtual machines, and operating systems possible
  - It allows a single CPU to be divided into multiple virtual CPUs for use by multiple VMs.
- GPU virtualization
  - GPU virtualization lets multiple VMs use all or some of a single GPU's processing power for faster video, artificial intelligence (AI), and other graphic- or math-intensive applications
- Linux virtualization
  - Linux includes its own hypervisor, called the kernel-based virtual machine (KVM), which supports Intel and AMD's virtualization processor extensions so you can create x86-based VMs from within a Linux host OS
- Cloud virtualization
  - IaaS, PaaS, and SaaS

Virtualization-Virtualization vs. containerization

- Server virtualization reproduces an entire computer in hardware, which then runs an entire OS. The OS runs one application. That's more efficient than no virtualization at all, but it still duplicates unnecessary code and services for each application you want to run.
- Containers take an alternative approach. They share an underlying OS kernel, only running the application and the things it depends on, like software libraries and environment variables. This makes containers smaller and faster to deploy.

Containerization - Containers

- Containers are executable units of software in which application code is packaged, along with its libraries and dependencies, in common ways so that it can be run anywhere, whether it be on desktop, traditional IT, or the cloud
- containers take advantage of a form of operating system (OS) virtualization in which features of the OS (in the case of the Linux kernel, namely the namespaces and cgroups primitives) are leveraged to both isolate processes and control the amount of CPU, memory, and disk that those processes have access to.
- Containers are small, fast, and portable because unlike a virtual machine, containers do not need include a guest OS in every instance and can, instead, simply leverage the features and resources of the host OS.
- Containers are Lightweight, Portable and platform independent, Supports modern development and architecture, and Improves utilization

Containerization – Use Cases

- Microservices:
  - Containers are small and lightweight, which makes them a good match for microservice architectures where applications are constructed of many, loosely coupled and independently deployable smaller services.
- DevOps:
  - The combination of microservices as an architecture and containers as a platform is a common foundation for many teams that embrace DevOps as the way they build, ship and run software.
- Hybrid, multi-cloud:
  - Because containers can run consistently anywhere, across laptop, on-premises and cloud environments, they are an ideal underlying architecture for hybrid cloud and multicloud scenarios where organizations find themselves operating across a mix of multiple public clouds in combination with their own data center.
- Application modernizing and migration:
  - One of the most common approaches to application modernization starts by containerizing them so that they can be migrated to the cloud.

Implement Virtualization and Containerization

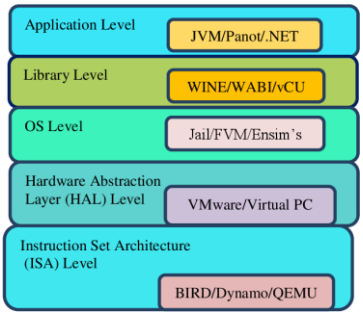
- Install VM ESXi Hypervisor
- Implement the KVM in Ubuntu
- Implement Microsoft Hypervisor-V
- Install docker in Windows or Ubuntu

Middleware Support for Virtualization

- Library-level virtualization is also known as user-level Application Binary Interface (ABI) or API emulation
- Create execution environments for running alien programs on a platform rather than creating a VM to run the entire operating system
- Several library-level virtualization systems: namely the Windows Application Binary Interface (WABI), lxrun, WINE, Visual MainWin, and vCUDA

Middleware or Runtime Library and References or Web Link	Brief Introduction and Application Platforms
WABI (http://docs.sun.com/app/docs/doc/802-6306)	Middleware that converts Windows system calls running on x86 PCs to Solaris system calls running on SPARC workstations
Lxrun (Linux Run) (http://www.ugcs.caltech.edu/~steven/lxrun/)	A system call emulator that enables Linux applications written for x86 hosts to run on UNIX systems such as the SCO OpenServer
WINE (http://www.winehq.org/)	A library support system for virtualizing x86 processors to run Windows applications under Linux, FreeBSD, and Solaris
Visual MainWin (http://www.mainsoft.com/)	A compiler support system to develop Windows applications using Visual Studio to run on Solaris, Linux, and AIX hosts
vCUDA (Example 3.2) (IEEE *IPDPS* 2009 [57])	Virtualization support for using general-purpose GPUs to run data-intensive applications under a special guest OS

Implementation levels of virtualization



Implementation levels of virtualization

- Instruction Set Architecture Level
  - virtualization is performed by emulating a given ISA by the ISA of the host machine
  - Example: MIPS(C) binary code can run on an x86-based host machine with the help of ISA emulation
- Hardware Abstraction Level (HAL)
  - manages the hardware using the process of virtualization
  - the hardware components, the input-output device, the memory, the processor
- Operating System Level
  - It is capable of creating a layer that is abstract between the operating system and the application
- Library Level
  - APIs, vCUDA and WINE
- Application Level
  - a process that deceives a standard app into believing that it interfaces directly with an operating system's capacities when, in fact, it does not.

Advantages of Virtualization

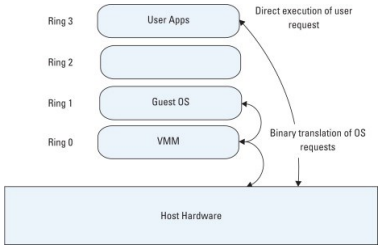
- Cut off IT expenses
- Reduce downtime and enhance resiliency in disaster recovery situations
- Increase efficiency and productivity
- Control independence and DevOps
- Move to be more green-friendly (organizational and environmental)
- Allows the running of legacy apps
- Enables cross-platform operations
- Prevents conflicts with other virtualized apps
- Permits users to run multiple app instances

Implementation Techniques of Virtualization

- Full virtualization
  - used to provide a VME that completely simulates the underlying hardware
  - Software are capable of execution on the physical hardware can be run in the VM and any OS supported by the underlying hardware can be run in each individual VM
  - the VM simulates enough hardware to allow an unmodified guest OS to be run in isolation
  - The *hypervisor* provides each VM with all the services of the physical system, including a virtual BIOS, virtual devices, and virtualized memory management
  - The guest OS is fully disengaged from the underlying hardware by the virtualization layer
  - Full virtualization is achieved by using a combination of binary translation and direct execution
  - hypervisors, the physical CPU executes non-sensitive instructions at native speed
  - OS instructions are translated on the fly and cached for future use, and user level instructions run unmodified at native speed
  - Full virtualization offers the best isolation and security for VMs
  - It simplifies migration and portability as the same guest OS instance can run on virtualized or native hardware

Implementation Techniques of Virtualization

- Full virtualization



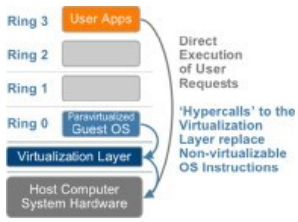
[source: https://www.sciencedirect.com/topics/computer-science/full-virtualization]

Implementation Techniques of Virtualization

- Paravirtualization
  - involves modifying the OS kernel
  - The OS kernel acts as a bridge between the applications and the processing done at the hardware level
  - replaces nonvirtualizable instructions with hypercalls that communicate directly with the virtualization layer hypervisor
  - A hypercall is based on the same concept as a system call
  - Hypercalls work with hypervisor likely System calls are used by an application to request services from the OS and provide the interface between the application or process and the OS
  - The hypervisor also provides hypercall interfaces for other kernel operations including memory management and interrupt handling
  - the host OS boots, the VM emulator is launched
  - the emulator uses either the VMLAUNCH (Intel) or the VMRUN (AMD) instruction to start execution of the VM
  - there are two copies of the OS in existence
  - introduce support issues in production environments because it requires deep OS kernel modifications, it is relatively easy compared with full virtualization
  - The open source Xen project is an example of paravirtualization

Implementation Techniques of Virtualization

- Paravirtualization



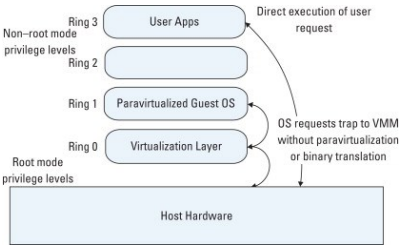
[source: https://www.sciencedirect.com/topics/computer-science/paravirtualization]

Implementation Techniques of Virtualization

- Hardware-Assisted or Emulator Virtualization
  - Hardware-assisted virtualization is also called native virtualization, accelerated virtualization, or hardware VM, depending on the vendor
  - a technology that allows for a CPU instruction set communication in which the VMM runs in a new root level mode below the OS kernel level
  - Intel and AMD processors became available in 2006 but was first introduced on the IBM System/370 in 1972
  - In this type of virtualization, privileged and sensitive calls are set to automatically trap to the hypervisor
  - The binary translation used in full virtualization or the use of hypercalls in paravirtualization is no longer needed
  - Depending on the CPU manufacturer, the guest state is stored in either VM Control Structures (Intel) or VM Control Blocks (AMD)
  - First-generation hardware-assisted technologies still lag behind in performance when compared to the full virtualization, but development of second-generation hardware-assisted technologies will improve virtualization performance while reducing memory overhead

Implementation Techniques of Virtualization

- Hardware-Assisted or Emulator Virtualization



[source: https://www.sciencedirect.com/topics/computer-science/assisted-virtualization]

Further Reading:
https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/techpaper/VMware_paravirtualization.pdf

Hardware Virtualization

- Hardware virtualization
  - virtualization of computers as complete hardware platforms
  - certain logical abstractions of their componentry
  - or only the functionality required to run various operating systems.
  - reducing equipment and labor costs associated with equipment maintenance
  - reducing energy consumption and the global footprint in environmental-ecological sectors of technology
  - more easily controlled than a physical machine
  - useful in kernel development and for teaching operating system courses, including running legacy operating systems that do not support modern hardware
  - no need for an up-front hardware purchase
  - relocated from one physical machine to another as needed

Hardware Virtualization

- Hardware virtualization
  - Networking Virtualization
  - Storage Virtualization
  - Guest OS images

Network Virtualization

- Network Virtualization (NV)
  - abstracting network resources that were traditionally delivered in hardware to software
  - combine multiple physical networks to one virtual, software-based network, or it can divide one physical network into separate, independent virtual networks.
  - allows network administrators to move virtual machines across different domains without reconfiguring the network
- Need of Network Virtualization
  - rewriting the rules for the way services are delivered
    - from the software-defined data center
      - to the cloud
      - to the edge
    - moves networks from static, inflexible, and inefficient to
      - Dynamic
      - Agile
      - Optimized
- With network virtualization, you can forget about spending days or weeks provisioning the infrastructure to support a new application.

Network Virtualization

- Network Virtualization (NV)
  - abstracting network resources that were traditionally delivered in hardware to software
  - combine multiple physical networks to one virtual, software-based network, or it can divide one physical network into separate, independent virtual networks.
  - allows network administrators to move virtual machines across different domains without reconfiguring the network
- Need of Network Virtualization
  - rewriting the rules for the way services are delivered
    - from the software-defined data center
      - to the cloud
      - to the edge
    - moves networks from static, inflexible, and inefficient to
      - Dynamic
      - Agile
      - Optimized
- With network virtualization, you can forget about spending days or weeks provisioning the infrastructure to support a new application.

© Bharati Vidyapeeth's Institute of Computer Applications and Management, New Delhi-63, by Dr.Arpita

Network Virtualization - Benefits

- Network Virtualization (NV)
  - Network virtualization decouples network services from the underlying hardware and allows virtual provisioning of an entire network
  - programmatically create, provision, and manage networks all in software
- Advantages
  - Reduce network provisioning time from weeks to minutes
  - Achieve greater operational efficiency by automating manual processes
  - Place and move workloads independently of physical topology
  - Improve network security within the data center
  - Networking and security policies defined for each connected application

Storage Virtualization

- Storage Virtualization (SV)
  - Storage virtualization or a virtual Storage Area Network (SAN) is the pooling of multiple physical storage arrays from SANs and making them appear as a single virtual storage device
  - The pool can integrate unlike storage hardware from different networks, vendors, or data centers into one logical view and manage them
  - separates the storage management software from
    - the underlying hardware infrastructure in order to provide more flexibility
    - scalable pools of storage resources
  - abstract storage hardware (arrays and disks) into virtual storage pools
- Storage Network Industry Association (SNIA) defines
  - The application of virtualization to storage services or devices for the purpose of aggregating functions or devices, hiding complexity, or adding new capabilities to lower level storage resources.

Storage Virtualization

- Challenges in Physical SANs
  - Vendor lock-in
  - Data migration across arrays
  - Scalability
  - Redundancy
  - Performance
  - High costs
  - Management
- Storage Virtualization Deployment Options
  - Storage Virtualization Node
    - consists of a storage virtualization controller node
    - interconnect your existing SAN arrays via iSCSI or FC connections to the new storage virtualization controller
  - Converged Server SAN
    - mixing new internal disks and existing external SAN arrays under the same virtual pool
    - More redundancy, Lower costs, Managed storage from a single pane of glass, Performance and scalability

Guest Operating System

- A guest OS is the software installed on either a virtual machine (VM) or partitioned disk that describes an operating system that is different than the host operating system.
- Virtualization technology allows a computer to run more than a single OS at the same time. A guest OS on a virtual machine can be different from the host OS while a guest OS on a partitioned disk must be the same be the same as the host OS.
- A Type 1 hypervisor (bare-metal hypervisor) runs directly on the hardware of a server, which takes the place of the host OS. The Type 1 hypervisor can create virtual machines, which can run guest OSs. A physical server can have multiple virtual machines and each VM can run its own guest OS. One guest OS can run Linux while another could run Windows.
- Guest operating systems can be of great benefit to administrators. Administrators are able to run programs and applications that aren't compatible with the host OS on a guest operating system. Admins can also run more than one application that requires different operating systems on the same physical hardware.

Cloud Data and Network Security

- Cloud security involves the procedures and technology that secure cloud computing environments against both external and insider cybersecurity threats
- Cloud computing, which is the delivery of information technology services over the internet, has become a must for businesses and governments seeking to accelerate innovation and collaboration
- Cloud security and security management best practices designed to prevent unauthorized access are required to keep data and applications in the cloud secure from current and emerging cyber-security threats.

Cloud security challenges

- Visibility into cloud data
  - In many cases, cloud services are accessed outside of the corporate network and from devices not managed by IT
- Control over cloud data
  - In a third-party cloud service provider's environment, IT teams have less access to data than when they controlled servers and applications on their own premises. Cloud customers are given limited control by default, and access to underlying physical infrastructure is unavailable.
- Access to cloud data and applications
  - Users may access cloud applications and data over the internet, making access controls based on the traditional data center network perimeter no longer effective. In addition, privileged access by cloud provider personnel could bypass your own security controls.
- Compliance
  - Use of cloud computing services adds another dimension to regulatory and internal compliance. Your cloud environment may need to adhere to regulatory requirements such as HIPAA, PCI and Sarbanes-Oxley, as well as requirements from internal teams, partners and customers. Cloud provider infrastructure, as well as interfaces between in-house systems and the cloud are also included in compliance and risk management processes.

Cloud security challenges

- Cloud-native breaches
  - Data breaches in the cloud are unlike on-premises breaches, in that data theft often occurs using native functions of the cloud
- Misconfiguration
  - Cloud-native breaches often fall to a cloud customer's responsibility for security, which includes the configuration of the cloud service.
  - Research shows that just 26% of companies can currently audit their IaaS environments for configuration errors. Misconfiguration of IaaS often acts as the front door to a Cloud-native breach, allowing the attacker to successfully land and then move on to expand and exfiltrate data. Research also shows 99% of misconfigurations go unnoticed in IaaS by cloud customers.
- Disaster recovery
  - Cyber-security planning is needed to protect the effects of significant negative breaches. A disaster recovery plan includes policies, procedures, and tools designed to enable the recovery of data and allow an organization to continue operations and business.
- Insider threats
  - A rogue employee is capable of using cloud services to expose an organization to a cybersecurity breach. A recent McAfee Cloud Adoption and Risk Report revealed irregular activity indicative of insider threat in 85% of organizations.

Cloud security solutions

- Visibility into cloud data
  - A complete view of cloud data requires direct access to the cloud service. Cloud security solutions accomplish this through an application programming interface (API) connection to the cloud service. With an API connection it is possible to view:
    - What data is stored in the cloud.
    - Who is using cloud data?
    - The roles of users with access to cloud data.
    - Who cloud users are sharing data with.
    - Where cloud data is located.
    - Where cloud data is being accessed and downloaded from, including from which device.
- Control over cloud data
  - Once you have visibility into cloud data, apply the controls that best suit your organization. These controls include:
    - Data classification (sensitive, regulated, or public)
    - Data Loss Prevention (DLP)
    - Collaboration controls
    - Encryption

Cloud security solutions

- Access to cloud data and applications
  - As with in-house security, access control is a vital component of cloud security. Typical controls include:
    - User access control
    - Device access control
    - Malicious behavior identification
    - Malware prevention
    - Privileged access
- Compliance
  - Existing compliance requirements and practices should be augmented to include data and applications residing in the cloud.
    - Risk assessment
      - Review and update risk assessments to include cloud services. Identify and address risk factors introduced by cloud environments and providers. Risk databases for cloud providers are available to expedite the assessment process.
    - Compliance Assessments
      - Review and update compliance assessments for PCI, HIPAA, Sarbanes-Oxley and other application regulatory requirements.

Cloud security solutions

- Access to cloud data and applications
  - As with in-house security, access control is a vital component of cloud security. Typical controls include:
    - User access control
    - Device access control
    - Malicious behavior identification
    - Malware prevention
    - Privileged access
- Compliance
  - Existing compliance requirements and practices should be augmented to include data and applications residing in the cloud.
    - Risk assessment
      - Review and update risk assessments to include cloud services. Identify and address risk factors introduced by cloud environments and providers. Risk databases for cloud providers are available to expedite the assessment process.
    - Compliance Assessments
      - Review and update compliance assessments for PCI, HIPAA, Sarbanes-Oxley and other application regulatory requirements.

Information Security

- Confidentiality
- Integrity
- Authenticity
- Availability
- Threat
- Vulnerabilities
- Risk
- Security Controls
- Security Mechanism
- Security Policies

Threat Agents

- Anonymous Attacker
- Malicious Service Agent
- Trusted Attacker
- Malicious Insider

Cloud Security Threats

- Traffic Eavesdropping
  - data being transferred to or within the cloud is passively intercepted by a malicious service agent
- Malicious Intermediary
  - Messages are intercepted and altered by malicious agent
- Denial of Service
- Insufficient Authorization
- Virtualization Attack
  - Exploits vulnerabilities about confidentiality, integrity, availability
- Overlapping trust boundaries
- Flaw in implementation
- Security Policy Disparity
- Risk

Cloud Security Mechanisms

- Encryption
- Hashing
- Digital Signature
- Public Key Infrastructure (PKI)
- Identity and Access Management (IAM)
- Single Sign-On (SSO)
- Cloud based security groups

Privacy and Trust

- Privacy
  - the service provider can access the data that is in the cloud at any time
  - It depends upon privacy policies, which users must agree to before they start using cloud services
  - privacy include policy and legislation as well as end-users' choices for how data is stored
  - Encryption and IAM are example of maintaining privacy in the cloud
- Trust is a critical factor in cloud computing; in present practice it depends largely on perception of reputation, and self assessment by providers of cloud services
- Trust mechanisms
  - Evidence
  - Attribute certification
  - Validation

Outcomes

- Now you should be able to implement Virtualization
- Know how of hypervisors
- Types of virtualization
- Levels of implementation of virtaulization
- Understanding of containers
- Get in depth of cloud security mechanism