Cloud Computing Final Assessment Questionnaires

Topic Virtualizations and Containers,Infrastructure and data security

Name	
Registration No	
Class	**Cloud Computing- Section-A**
Batch	**4**

Cloud Computing Final Assessment Questionnaires

Q1.What is virtualization in cloud computing? Or Why do we use Virtualization in the cloud?

Virtualization isn't merely a tool for creating virtual machines—it's the core technology for enabling cloud computing. Cloud computing is fundamentally based on pooling resources and virtualization is the technology used to convert fixed infrastructure into these pooled resources. Virtualization provides the abstraction needed for resource pools, which are then managed using orchestration.

Q2 What are the various categories of hardware virtualization used in the cloud?

1.Compute:Compute virtualization abstracts the running of code (including operating systems) from the underlying hardware. Instead of running directly on the hardware, the code runs on top of an abstraction layer that enables more flexible usage, such as running multiple operating systems on the same hardware (virtual machines).Compute most commonly refers to virtual machines or Instances.

2.Networks:Software-Defined Networking (SDN) is an approach to networking that uses software-based controllers or application programming interfaces (APIs) to communicate with underlying hardware infrastructure and direct traffic on a network. This model differs from that of traditional networks, which use dedicated hardware devices (i.e., routers and switches) to control network traffic. SDN can create and control a virtual network – or control a traditional hardware – via software.

3.Storage:Storage virtualization is already common in most organizations—Storage Area Network (SAN) and Network-Attached Storage (NAS) are both common forms of storage virtualization—and storage.

- ❑ Object storage: Object storage is similar to a file system. "Objects" are typically files, which are then stored using a cloud-platform specific mechanism. Most access is through APIs, not standard file sharing protocols, although cloud providers may also offer front-end interfaces to support those protocols.
- ❑ Volume storage: This is essentially a virtual hard drive for instances/virtual machines.
- ❑ Database: Cloud platforms and providers may support a variety of different kinds of databases, including existing commercial and open source options, as well as their own proprietary systems.Proprietary databases typically use their own APIs.

Commercial or open source databases by the provider and typically use existing standards for connections. These can be relational or non-relational—the latter includes NoSQL and other key/value storage systems, or file system-based databases (e.g. HDFS).

- Application/platform: Examples of these would be a content delivery network (CDN), files stored in SaaS, caching, and other novel options.

4.Containers:Containers are code execution environments that run within an operating system for sharing and leveraging resources of that operating system. While a VM is a full abstraction of an operating system, a container is a constrained place to run segregated processes while still utilizing the kernel and other capabilities of the base OS.

- Multiple containers can run on the same virtual machine or be implemented without the use of VMs at all and run directly on hardware.
- The container provides code running inside a restricted environment with only access to the processes and capabilities defined in the container configuration. This allows containers to launch incredibly rapidly, since they don't need to boot an operating system or launch many (sometimes any) new services; the container only needs access to already-running services in the host OS and some can launch in milliseconds.
- Containers are the important part of cloud native applications deployments
- Examples Dockers ,RKT or Rocket ,Linux Containers(LXC) and CRI-O is an implementation of the Kubernetes Container Runtime Interface (CRI)Kubernetes

Q3:Explain Kubernetes Containers ?

- Kubernetes:Kubernetes Service allows you to run and manage your infrastructure in a kubernetes environment. Kubernetes is an open-source tool to manage or orchestrate your containers in the form of worker nodes designed to automate, deploying, scaling, and operating containerized applications. It is designed to grow from tens, thousands, or even millions of containers. There are 2 main components kubernetes control plane and worker nodes manage the overall flow for EKS.
- *Kubernetes Control Plane*

  There are number of different components that make up the control plane and these include a number of different APIs. It has a job to manage and decide the clusters and responsible to communication for your nodes.
- *Worker Nodes*

  Kubernetes Clusters are composed of nodes. A node is a worker machine in Kubernetes and runs as an on-demand EC2 instance and includes software to run containers managed by the Kubernetes control plane.

Q4:What is Hypervisor? What is the need of Hypervisor in Cloud Computing?

A **hypervisor** is software that creates and runs virtual machines (VMs). A **hypervisor**, sometimes called a virtual machine monitor (VMM).It is a small software layer that enables multiple instances of operating systems to run alongside each other, sharing the same physical computing resources.

In cloud computing Virtualization enables the “resource pooling” characteristics of cloud or abstraction .No virtualization means no cloud.

Q5:What are types of hypervisors?

1.Type 1 Hypervisors:A Type 1 hypervisor runs directly on the underlying computer’s physical hardware, interacting directly with its CPU, memory, and physical storage. For this reason, Type 1 hypervisors are also referred to as bare-metal hypervisors. A Type 1 hypervisor takes the place of the host operating system.

Type 1 hypervisors are highly efficient because they have direct access to physical hardware. This also increases their security, because there is nothing in between them and the CPU that an attacker could compromise. But a Type 1 hypervisor often requires a separate management machine to administer different VMs and control the host hardware.

2.Type 2 Hypervisors:A Type 2 hypervisor doesn’t run directly on the underlying hardware. Instead, it runs as an application in an OS. Type 2 hypervisors rarely show up in server-based environments. Instead, they’re suitable for individual PC users needing to run multiple operating systems.

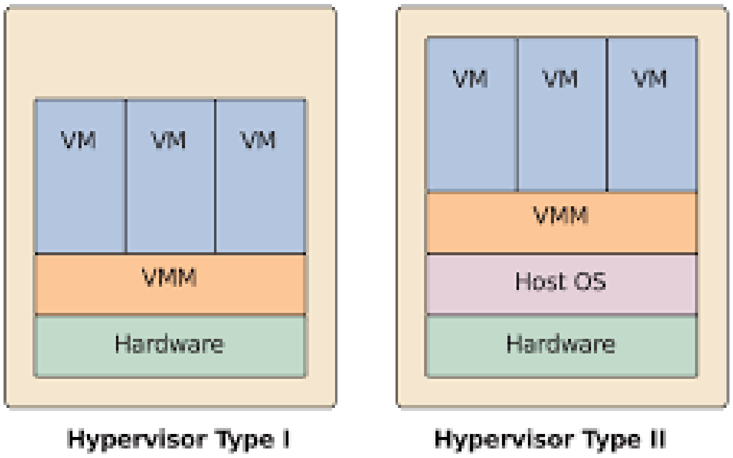


Fig. 2: System VMs.

Q5 What are the container's components?

- Software container systems always include three key components:
  - The execution environment (the container).
  - An orchestration and scheduling controller (which can be a collection of multiple tools).
  - A repository for the container images or code to execute.
  - Together, these are the place to run things, the things to run, and the management system to tie them together.

Q6 What are Dockers?

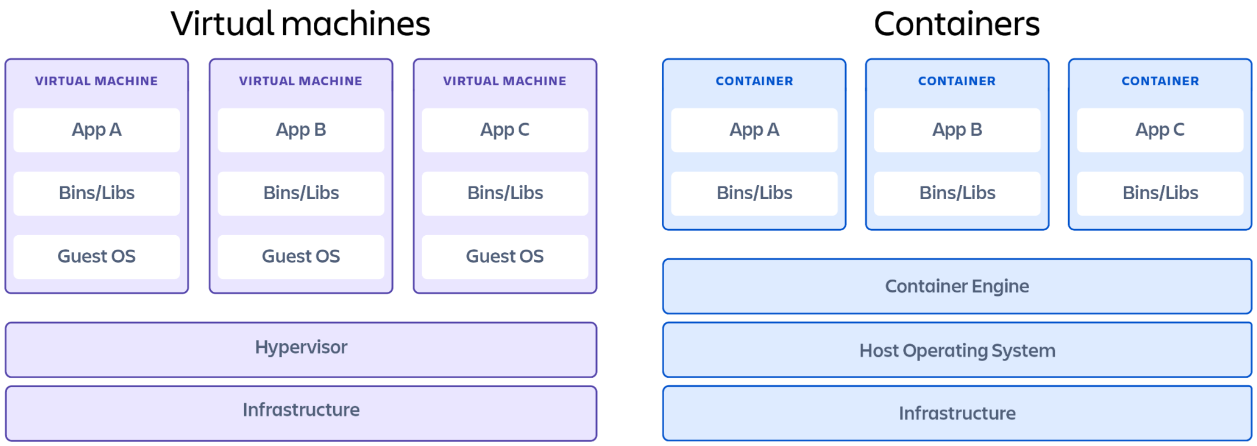
Docker is the most popular and widely used container runtime. Docker Hub is a giant public repository of popular containerized software applications. Containers on Docker Hub can be instantly downloaded and deployed to a local Docker runtime.

Q7 What is the difference between Virtual machines(VMs) and Containers?

The key differentiator between containers and virtual machines is that virtual machines virtualize an entire machine down to the hardware layers and containers only virtualize software layers above the operating system level.

- Containers don’t necessarily provide full security isolation, but they do provide task segregation.That said, virtual machines typically do provide security isolation.
- Different host operating systems and container technologies offer different

security capabilities. This assessment should be included in any container platform selection process.



Q8:Deploying Containers(dockers,Kubernetes or LCX) on Windows or Linux environments provide the same level of security ?
No Windows and Linux have different security features or capabilities therefore container deployment will provide different security levels.Do the proper platform assessment before deploying any containers.

Q9:Who is responsible for virtualization or hypervisors security in the cloud?
The cloud provider is also responsible for securing the underlying infrastructure and the virtualization technology from external attack or internal misuse. This means using patched and up-to-date hypervisors that are properly configured and supported with processes to keep them up to date and secure over time. The inability to patch hypervisors across a cloud deployment could create a fundamentally insecure cloud when a new vulnerability in the technology is discovered.

Q10:What will happen? if the hypervisor or O/S is not patched properly?
If Hypervisor is not patched then total Cloud and its deployments become insecure.
If the O/S is not patched then only the deployments on that O/S will be insecure or vulnerable.

Q11: what are the categories of Network virtualizations?

- Virtual Local Area Networks (VLANs):
  - VLANs leverage existing network technology implemented in most network hardware. VLANs are extremely common in enterprise networks, even without cloud computing. They are designed for use in single-tenant

networks (enterprise data centers) to separate different business units, functions, etc. (like guest networks). VLANs are not designed for cloud-scale virtualization or security and shouldn’t be considered, on their own, an effective security control for isolating networks. They are also never a substitute for physical network segregation.

❏ Software-Defined Networking (SDN) is an approach to networking that uses software-based controllers or application programming interfaces (APIs) to communicate with underlying hardware infrastructure and direct traffic on a network.This model differs from that of traditional networks, which use dedicated hardware devices (i.e., routers and switches) to control network traffic. SDN can create and control a virtual network – or control a traditional hardware – via software.

Q12:Virtual Local Area Networks (VLANs): can be used for network virtualization in the cloud ?
No They are designed for use in single-tenant networks (enterprise data centers) to separate different business units, functions, etc. (like guest networks). VLANs are not designed for cloud-scale virtualization or security and shouldn’t be considered, on their own, an effective security control for isolating networks.

Q13:What kind of firewalls are used in the cloud?

❏ SDN firewalls (e.g., security groups) can apply “TAG” to assets or Virtual machines or Instances based on more flexible criteria than hardware-based firewalls, since they aren’t limited based on physical topology. SDN firewalls are typically policy sets that define ingress and egress rules that can apply to single assets or groups of assets, regardless of network location (within a given virtual network).For example, you can create a set of firewall rules that apply to any asset with a particular tag. For example, if virtual machines in an auto-scale group are automatically deployed in multiple subnets and load balanced across them, then you can create a firewall ruleset that applies to these instances or TAGS ,regardless of their subnet or IP address. It is a key enabling feature of secure cloud networks that use architectures quite differently from traditional computing.

Q14:We can deploy Traditional or On-premises Firewalls in the Cloud?

No Traditional firewalls will not worked in cloud due to auto-scaling group are automatically deployed in multiple subnets and load balanced across them, The IP addresses will change much quickly and with high velocity the traditional firewalls operates on IP-based policies may treat this regular cloud feature as incident or cant operate properly while with SDN based firewalls you can create a firewall ruleset that applies to these instances or TAGS ,regardless of their subnet or IP address to operate normally under cloud.

Q15:What are the different types of compute?

1.Virtual Machines or Instances:**Virtual machines:** Virtual machines are the most-well known form of compute abstraction, and are offered by all IaaS providers. They are commonly called instances in cloud computing since they are created (or cloned) off a base image. The Virtual Machine Manager (hypervisor) abstracts an operating system from the underlying hardware. Modern hypervisors can tie into underlying hardware capabilities now commonly available on standard servers (and workstations) to reinforce isolation while supporting high-performance operations.

- ❏ Virtual machines are potentially open to certain memory attacks, but this is increasingly difficult due to ongoing hardware and software enhancements to reinforce isolation. VMs on modern hypervisors are generally an effective security control, and advances in hardware isolation for VMs and secure execution environments continue to improve these capabilities.

2.Containers:Containers are code execution environments that run within an operating system

(for now), sharing and leveraging resources of that operating system. While a VM is a full
abstraction of an operating system, a container is a constrained place to run segregated
processes while still utilizing the kernel and other capabilities of the base OS.

- ❏ Multiple containers can run on the same virtual machine or be implemented without the use of VMs at all and run directly on hardware.
- ❏ The container provides code running inside a restricted environment with only access to the processes and capabilities defined in the container configuration. This allows containers to launch incredibly rapidly, since they don’t need to boot an operating system or launch many (sometimes any) new services; the container only needs access to already-running services in the host OS and

some can launch in milliseconds.

- ❏ Containers are newer, with differing isolation capabilities that are very platform-dependent.

3.Platform based load: Platform-based workloads:

- ❏ This is a more complex category that covers workloads running on a shared platform that aren't virtual machines or containers, such as logic/procedures running on a shared database platform. Imagine a stored procedure running inside a multitenant database, or a machine-learning job running on a machine-learning Platform as a Service. Isolation and security are totally the responsibility of the platform provider, although the provider may expose certain security options and controls.

4.Serverless Computing:Serverless is a broad category that refers to any situation where the cloud user doesn't manage any of the underlying hardware or virtual machines, and just accesses exposed functions. For example, there are serverless platforms for directly executing application code. Under the hood, these still utilize capabilities such as containers, virtual machines, or specialized hardware platforms. From a security perspective, serverless is merely a combined term that covers containers and platform-based workloads, where the cloud provider manages all the underlying layers, including foundational security functions and controls.

Q16: What is dedicated hosting or Private Tenancy in the cloud?

In some environments dedicated/private tenancy is possible, but typically at a higher cost. With this model only designated workloads run on a designated physical server. Costs increase in public cloud as a consumer since you are taking hardware out of the general resource pool, but also in private cloud, due to less efficient use of internal resources.

Q17 What are Immutable Virtual machines ?

Auto-scaling and containers, by nature, work best when you run instances launched dynamically based on an image; those instances can be shut down when no longer needed for capacity without breaking an application stack. This is core to the elasticity of compute in the cloud. Thus, you no longer patch or make other changes to a running workload, since that wouldn't change the image, and, thus, new instances would be out of sync with whatever manual changes you make on whatever is running. We call these virtual machines immutable.

Q18: Updating Operating system updates is an example of an Immutable machine?
No, it's not completely immutable as changes are applied in the live environment.

Q19: Full Vulnerability assessments and Penetration testing can be .done in the cloud?
No only limited testing with prior notification to cloud providers based on multi tenancy nature of the cloud and other security risks.

Q20: How you protect data at rest in the cloud?

- Encryption
    - Encryption protects data by applying a mathematical algorithm that “scrambles” the data, which then can only be recovered by running it through an unscrambling (decryption) process with a corresponding key. The result is a blob of “ciphertext”.
    - Encryption Components
        - There are three components of an encryption system:
            - Data
                - The data is, of course, the information that you’re encrypting.
            - The encryption engine
                - The engine is what performs the mathematical process of encryption?
            - Key management
                - Finally, the key manager handles the keys for the encryption.
        - The overall design of the system focuses on where to put each of these components.
        - When designing an encryption system, you should start with a threat model. For example, do you trust a cloud provider to manage your keys? How could the keys be exposed? Where should you locate the encryption engine to manage the threats you are concerned with?
- IAAS Encryption
    - IaaS Encryption IaaS volumes can be encrypted using different methods, depending on your data.
- Volume storage encryption
    - Instance-managed encryption: The encryption engine runs within the instance, and the key is stored in the volume but protected by a passphrase or keypair.
    - Externally managed encryption:The encryption engine runs in the instance, but the keys are managed externally and issued to the instance

on request.

- ❑ Object and file storage
  - ❑ Client-side encryption: When object storage is used as the back-end for an application (including mobile applications), encrypt the data using an encryption engine embedded in
    the application or client.
  - ❑ Server-side encryption: Data is encrypted on the server (cloud) side after being transferred in. The cloud provider has access to the key and runs the encryption engine.
  - ❑ Proxy encryption: In this model, you connect the volume to a special instance or appliance/ software, and then connect your instance to the encryption instance. The proxy handles all crypto operations and may keep keys either
    onboard or externally.
- ❑ PaaS Encryption
  - ❑ PaaS encryption varies tremendously due to all the different PaaS platforms.
    - ❑ Application layer encryption: Data is encrypted in the PaaS application or the client accessing the platform.
    - ❑ Database encryption: Data is encrypted in the database using encryption that’s built in and is supported by a database platform like Transparent Database Encryption (TDE) or at the field level.
    - ❑ Other: These are provider-managed layers in the application, such as the messaging queue. There are also IaaS options when that is used for underlying storage.
- ❑ SaaS Encryption
  - ❑ SaaS providers may use any of the options previously discussed. It is recommended to use per-customer keys when possible, in order to better enforce multitenancy isolation. The following options are for SaaS consumers:
    - ❑ Provider-managed encryption: Data is encrypted in the SaaS application and generally managed by the provider.
    - ❑ Proxy encryption: Data passes through an encryption proxy before being sent to the SaaS application.

- ❑ Tokenization, on the other hand, takes the data and replaces it with a random value. It
  then stores the original and the randomized version in a secure database for later recovery.

- ❑ Tokenization is often used when the format of the data is important (e.g. replacing credit card numbers in an existing system that requires the same format text string).

- Format Preserving
  - Encryption encrypts data with a key but also keeps the same structural format as tokenization.
  - but it may not be as cryptographically secure due to the compromises.

Q21: what are the main consideration for key management?

- The main considerations for key management are
  - performance, accessibility, latency, and security
  - Can you get the right key to the right place at the right time while also meeting your security and compliance requirements?
- There are four potential options for handling key management:
  - HSM/appliance: Use a traditional hardware security module (HSM) or appliance-based key manager, which will typically need to be on-premises, and deliver the keys to the cloud over a dedicated connection.
  - Virtual appliance/software: Deploy a virtual appliance or software-based key manager in the cloud.
  - Cloud provider service: This is a key management service offered by the cloud provider. Before selecting this option, make sure you understand the security model and SLAs to understand if your key could be exposed.
  - Hybrid: You can also use a combination, such as using a HSM as the root of trust for keys but then delivering application-specific keys to a virtual appliance that's located in the cloud and only manages keys for its particular context.

Q22:what is customer-managed Key?

- A customer-managed key allows
  - a cloud customer to manage their own encryption key
  - while the provider manages the encryption engine.
- For example, using your own key to encrypt SaaS data within the SaaS platform. Many providers encrypt data by default, using keys completely in their control. Some may allow you to substitute your own key, which integrates with their encryption system. Make sure your vendor's practices align with your requirements.
- Some providers may require you to use a
  - service within the provider to manage the key.
  - Thus, although the key is customer-managed, it is still potentially available to provider. This doesn't necessarily mean it is insecure: Since the key management and data storage systems can be separated, it would require collusion on the part of multiple employees at the provider to potentially compromise data.
  - However, keys and data could still be exposed by a government request, depending on local laws. You may be able to store the keys externally from the provider and only pass them over on a per-request basis.

Q23: Why enable versioning in the cloud storage?
To protect data against intentional or unintentional modifications or destruction .Backup and versioning will help in recovering data and the object or document version before modification.