6181]-177

B.E. (E & TC)

CLOUD COMPUTING

(2019 Pattern NOV-DEC 2023) (End-Semester) (Semester - VII) (404183)

Question 1

(a) Describe the structure and types of virtualizations.

Virtualization is a cornerstone technology in modern computing, enabling the creation of virtual representations of physical resources. It permits multiple operating systems to run concurrently on a single physical system by abstracting hardware resources and presenting them as independent, logical units.

Detailed Structure of Virtualization:

1. **Host Machine**: The physical hardware that houses the virtualization layer and provides foundational resources such as CPU, memory, storage, and network.
2. **Hypervisor**: A critical software layer responsible for managing virtual machines. It allocates resources dynamically and ensures isolation and efficient operation of VMs.
3. **Virtual Machines (VMs)**: These are the encapsulated environments that act as separate logical computers, each running its own operating system and applications.
4. **Guest Operating System**: The OS installed and operational inside a VM, independent of the host system.
5. **Virtual Resources**: Includes virtual CPUs (vCPUs), virtual memory, virtualized network adapters, and emulated storage devices, which simulate real hardware components for the VMs.

Expanded Types of Virtualization:

1. **Server Virtualization**:
   - Facilitates partitioning a single physical server into multiple logical servers.
   - Enhances resource utilization and reduces hardware costs.
   - Examples: VMware vSphere, Microsoft Hyper-V.
2. **Storage Virtualization**:
   - Combines various physical storage resources into a unified, manageable storage pool.
   - Benefits include simplified management, improved scalability, and disaster recovery capabilities.
3. **Network Virtualization**:
   - Abstracts networking resources, allowing dynamic allocation and management.
   - Uses technologies like VLANs (Virtual Local Area Networks) and SDNs (Software-Defined Networking).
4. **Desktop Virtualization**:
   - Enables users to access virtual desktop environments remotely.
   - Example: Virtual Desktop Infrastructure (VDI), which facilitates secure and centralized desktop deployment.
5. **Application Virtualization**:
   - Isolates applications from the operating system, allowing them to run in independent virtual environments.
   - Examples include Citrix XenApp and VMware ThinApp.

(b) Explain in brief Virtual Clusters.

Virtual clusters are logical groupings of virtual machines interconnected via a network and configured to function as a cohesive unit. Unlike traditional clusters, virtual clusters do not require physical proximity and can span multiple servers or even geographic locations.

Comprehensive Characteristics of Virtual Clusters:

1. **Scalability**:
   - New nodes (virtual machines) can be dynamically added to the cluster as needed, providing elastic scalability.
   - Resources are allocated based on workload demands.
2. **Fault Tolerance and High Availability**:
   - Redundancy mechanisms within the cluster ensure that failures of individual nodes do not disrupt overall operations.
3. **Resource Sharing and Efficiency**:
   - Optimizes the utilization of hardware resources by dynamically allocating them to virtual nodes.

Applications in Real-World Scenarios:

- High-performance computing (HPC) workloads.
- Enterprise applications requiring load balancing.
- Disaster recovery setups in cloud environments.

(c) Write short notes on:

(i) Desktop Virtualization:

Desktop virtualization revolutionizes the way users interact with desktop environments by enabling them to access their computing resources remotely, irrespective of the underlying hardware.

Expanded Benefits:

1. **Remote Accessibility**:
   - Users can securely access their desktops from any device, enhancing mobility and productivity.
2. **Enhanced Security**:
   - Centralized management of virtual desktops ensures tighter security controls, reducing risks from data breaches.
3. **Cost-Effectiveness**:
   - Reduces the need for high-spec end-user devices by offloading processing to the server side.

(ii) Network Virtualization:

Network virtualization abstracts the physical network infrastructure, creating logical virtual networks that operate independently of the underlying hardware.

Expanded Benefits:

1. **Dynamic Resource Allocation**:
   - Simplifies network reconfiguration and scaling.
2. **Improved Security**:
   - Segmentation of virtual networks enhances security by isolating sensitive traffic.
3. **Increased Flexibility**:
   - Facilitates rapid deployment of new applications and services.

Question 2

(a) Describe the components of a virtual machine.

A virtual machine (VM) encapsulates a complete computing environment, providing users with the illusion of a dedicated physical machine while operating on shared physical resources.

Detailed Components of a Virtual Machine:

1. **Virtual CPU (vCPU)**:
   - Acts as a software-based representation of a physical CPU.
   - Handles computation tasks for applications and the guest OS.
2. **Virtual Memory**:
   - Simulated memory allocated from the host machine's RAM.
   - Allows efficient memory management through paging and allocation techniques.
3. **Virtual Storage**:
   - Provides a virtualized representation of physical storage devices, often using files or partitions on the host.
   - Enables features like snapshots and cloning.
4. **Virtual Network Interface**:
   - Connects the VM to a virtual or physical network, enabling communication with other devices.
5. **Hypervisor**:
   - The layer of software that manages and monitors the VM, ensuring efficient resource allocation and isolation.
6. **Guest Operating System**:
   - The OS installed within the VM, operating independently of the host OS.

(b) Describe the various methods of implementing storage virtualization.

Storage virtualization abstracts and consolidates multiple physical storage devices into a single logical storage unit, improving manageability and efficiency.

Expanded Methods:

1. **Block-Level Virtualization**:
   - Operates at the block layer, aggregating storage blocks from multiple devices into a unified pool.
   - Used in SAN (Storage Area Network) environments to improve performance and scalability.
2. **File-Level Virtualization**:
   - Abstracts file storage by consolidating files and directories across multiple storage systems.
   - Simplifies access and management in NAS (Network Attached Storage) environments.
3. **Host-Based Virtualization**:
   - Implemented at the host operating system level.
   - Ideal for smaller deployments or environments without specialized hardware.
4. **Network-Based Virtualization**:
   - Centralizes storage management within the network layer, enhancing scalability and resource sharing.
5. **Array-Based Virtualization**:
   - Integrates virtualization functionality directly within storage arrays, enabling seamless data movement and replication.

Question 3

(a) Explain the design principles of cloud computing services.

Cloud computing services are designed with specific principles to ensure scalability, flexibility, and reliability.

Expanded Design Principles:

1. **Resource Pooling**:
   - Aggregates resources like storage, computing power, and network bandwidth into a shared pool.
   - Example: Multi-tenancy in AWS, where different users share the same physical resources securely.
2. **On-Demand Self-Service**:
   - Users can provision and manage resources automatically without human intervention.
   - Example: Allocating virtual machines in Microsoft Azure through its web interface.
3. **Elasticity and Scalability**:
   - Automatically scales resources up or down based on demand.
   - Example: Auto-scaling groups in AWS EC2 allow for resource elasticity.
4. **Measured Service**:
   - Resource usage is monitored, controlled, and reported, providing transparency.
   - Example: Google Cloud provides real-time dashboards showing resource usage metrics.
5. **Broad Network Access**:
   - Services are accessible over standard networks using various devices such as phones, laptops, or tablets.
   - Example: Accessing Dropbox files through mobile applications.
6. **Resilience and Fault Tolerance**:
   - Ensures minimal downtime and data protection during system failures.
   - Example: Data replication in Amazon S3 across multiple regions.

(Diagram Required)

(b) Draw and explain the cloud computing life cycle.

The cloud computing life cycle outlines the stages from planning to retirement of cloud resources.

Stages:

1. **Planning**:
   - Define objectives, estimate costs, and choose the appropriate cloud model (public, private, or hybrid).
2. **Design**:
   - Architect solutions, considering scalability, security, and compliance.
3. **Deployment**:
   - Implement cloud infrastructure and configure applications.
4. **Operation and Management**:
   - Monitor performance, optimize resource utilization, and ensure security.
5. **Decommissioning**:
   - Safely retire outdated or unnecessary resources to optimize costs.

(Illustrate with a lifecycle diagram)

(c) Enlist the elements of cloud security architecture with a suitable diagram.

The cloud security architecture consists of tools, policies, and processes to secure cloud environments.

Expanded Elements:

1. **Identity and Access Management (IAM)**:
   - Ensures that only authorized users can access resources.
   - Example: AWS IAM enables granular access control.
2. **Data Security**:
   - Encrypts data in transit and at rest to prevent unauthorized access.
   - Example: Google Cloud offers default encryption for all stored data.
3. **Network Security**:
   - Implements firewalls, VPNs, and intrusion detection systems.
   - Example: Azure Network Security Groups (NSGs) filter traffic.
4. **Application Security**:
   - Protects applications from vulnerabilities like SQL injection and cross-site scripting.
   - Example: Web Application Firewalls (WAFs) in AWS.
5. **Compliance Management**:
   - Ensures adherence to regulatory frameworks like GDPR and HIPAA.
   - Example: AWS Artifact provides access to compliance documents.

(Include a diagram highlighting these elements.)

Question 4

(a) Elaborate the cloud computing reference architecture.

Cloud computing reference architecture outlines the components and relationships necessary for delivering cloud services efficiently. The architecture is crucial for ensuring standardized deployments and integration.

Key Components:

1. **Cloud Consumers**:
   - End-users or enterprises utilizing cloud services.
   - Examples: Businesses using AWS S3 for storage or Google Workspace for collaboration.
2. **Cloud Providers**:
   - Entities offering computing resources and services, such as infrastructure, platforms, or software.
   - Examples: Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform.
3. **Cloud Brokers**:
   - Facilitate negotiations and manage service usage between providers and consumers.
   - Example: Cloud aggregators combining IaaS and SaaS solutions for specific industries.
4. **Cloud Auditors**:
   - Evaluate and ensure compliance with regulations and standards.
   - Example: External security audits for SOC 2 compliance.
5. **Service Orchestration**:
   - Coordinates tasks, workflows, and resource provisioning in cloud environments.
   - Example: Kubernetes orchestrating containerized applications.

(Diagram illustrating these interactions can enhance understanding.)

(b) Explain the types and functions of a firewall.

Firewalls are security devices or software designed to monitor and control incoming and outgoing network traffic based on predefined rules.

Types:

1. **Packet Filtering Firewall**:
   - Inspects packets and allows or blocks traffic based on header information.
   - Example: Filtering traffic by IP addresses or ports.
2. **Stateful Inspection Firewall**:
   - Tracks the state of active connections and makes decisions based on context.
   - Example: Allowing HTTP traffic only within established sessions.
3. **Proxy Firewall**:
   - Operates as an intermediary between clients and servers.
   - Example: A web proxy analyzing HTTP requests for threats.
4. **Next-Generation Firewall (NGFW)**:
   - Integrates traditional firewalls with additional features like intrusion prevention and application-layer filtering.
   - Example: Palo Alto Networks NGFW detecting malware within encrypted traffic.

Functions:

- **Access Control**: Enforces policies on allowed traffic.
- **Intrusion Prevention**: Detects and blocks suspicious activities.
- **Logging and Monitoring**: Maintains records of traffic for analysis.

(c) Describe the security challenges for cloud service customers.

Cloud customers face numerous security challenges due to the shared responsibility model and dynamic cloud environments.

Key Challenges:

1. **Data Breaches**:
   - Unauthorized access to sensitive data stored in the cloud.
   - Mitigation: Encryption and access control policies.
2. **Compliance Violations**:
   - Failure to adhere to standards like GDPR, HIPAA, or ISO 27001.
   - Mitigation: Regular audits and compliance tracking tools.
3. **Account Hijacking**:
   - Attackers gaining control of user accounts via phishing or credential theft.
   - Mitigation: Multi-factor authentication (MFA).
4. **Insider Threats**:
   - Malicious or careless actions by authorized personnel.
   - Mitigation: Continuous monitoring and access restrictions.

(Diagram highlighting these challenges and solutions can add clarity.)

Question 5

(a) Explain the steps to create and manage associated objects for Amazon S3 Bucket.

Amazon S3 provides scalable and secure object storage. Managing buckets and objects involves:

1. **Creating a Bucket**:
   - Log in to AWS Management Console.
   - Navigate to S3 and select "Create Bucket."
   - Specify a unique bucket name and region.
2. **Configuring Permissions**:
   - Define access policies (public/private).
   - Example: Restricting access using AWS Identity and Access Management (IAM).
3. **Uploading Objects**:
   - Use the AWS Console or CLI to upload files.
   - Set metadata and storage classes (e.g., Standard, Glacier).
4. **Managing Lifecycle Policies**:
   - Automate transitioning objects between storage classes based on age.
5. **Monitoring Activity**:
   - Enable server access logging and AWS CloudTrail for auditing actions.

(b) Explain Google App Engine application life cycle.

Google App Engine (GAE) is a Platform-as-a-Service (PaaS) that simplifies deploying and scaling applications. Its lifecycle includes:

1. **Development**:
   - Write applications using supported languages like Python, Java, or Go.
   - Test locally with the App Engine SDK.
2. **Deployment**:
   - Deploy applications via the Google Cloud Console or gcloud CLI.
   - Example: "gcloud app deploy" command.
3. **Autoscaling**:
   - GAE automatically adjusts instances based on traffic.
   - Example: Scale to zero during inactivity for reduced costs.
4. **Monitoring and Debugging**:
   - Use Stackdriver for real-time metrics and error diagnostics.
5. **Maintenance and Updates**:
   - Roll out updates using versioning for zero-downtime deployments.

(c) Explain the different cost models in cloud computing.

1. **Pay-As-You-Go**:
   - Pay for actual usage of resources like computing power, storage, or bandwidth.
   - Example: AWS EC2 instance costs billed per second.
2. **Reserved Instances**:
   - Commit to a specific amount of resources over a fixed term for discounted rates.
3. **Spot Pricing**:
   - Bid for unused cloud capacity at reduced rates.
   - Example: Running batch jobs on AWS Spot Instances.
4. **Subscription-Based**:
   - Fixed recurring charges for predictable usage patterns.
   - Example: Microsoft Office 365.

Question 6

(a) Describe Amazon EC2 cloud in brief considering the following points:

Amazon Elastic Compute Cloud (EC2) is a web service provided by AWS that offers scalable computing capacity in the cloud. It allows users to run virtual servers tailored to their requirements.

Key Features:

1. **Amazon Machine Image (AMI)**:
   - AMI serves as a template containing the software configuration (OS, application server, and applications) required to launch an instance.
   - Example: A Linux-based AMI with pre-configured Apache server for web hosting.
2. **Amazon CloudWatch**:
   - A monitoring tool that provides actionable insights into resource utilization, performance, and operational health.
   - Example: Setting alarms to monitor CPU usage of EC2 instances.

Additional Benefits:

- **Scalability**: Auto-scaling groups automatically adjust the number of instances based on traffic.
- **Flexibility**: Offers a variety of instance types optimized for different workloads (e.g., compute, memory, storage).

(b) Write a short note on Microsoft Azure.

Microsoft Azure is a comprehensive cloud computing platform offering IaaS, PaaS, and SaaS solutions. It enables organizations to build, deploy, and manage applications across a global network of data centers.

Key Features:

1. **Compute Services**:
   - Virtual machines, app services, and Kubernetes-based container orchestration.
2. **AI and Machine Learning**:
   - Cognitive services for natural language processing, vision, and anomaly detection.
3. **Hybrid Cloud**:
   - Azure Stack extends Azure services to on-premises data centers for hybrid solutions.
4. **DevOps Tools**:
   - Azure DevOps supports CI/CD pipelines and collaborative development.

(c) Explain different services provided by Google Cloud Platform.

Google Cloud Platform (GCP) delivers various services tailored for modern business needs. Key offerings include:

1. **Compute Services**:
   - Google Compute Engine (virtual machines) and Kubernetes Engine (container management).
2. **Storage Solutions**:
   - Google Cloud Storage for object data, Persistent Disks for VM storage, and BigQuery for analytics.
3. **AI and ML Tools**:
   - TensorFlow integration and pre-trained models for image recognition and language translation.
4. **Networking**:
   - Global load balancers, Cloud CDN for content delivery, and Cloud VPN for secure connectivity.

Question 7

(a) Draw and explain the architecture of the IoT.

The Internet of Things (IoT) architecture consists of multiple layers enabling the collection, processing, and utilization of data from connected devices.

Layers:

1. **Perception Layer**:
   - Comprises sensors and devices that collect data from the environment.
   - Example: Temperature sensors monitoring greenhouse conditions.
2. **Network Layer**:
   - Transmits collected data using protocols like MQTT and HTTP.
3. **Processing Layer**:
   - Involves data processing units such as cloud servers and edge devices.
   - Example: Processing data from IoT sensors in AWS IoT Core.
4. **Application Layer**:
   - Provides user interfaces and services, such as smart home apps for device control.

(Include a diagram illustrating these layers.)

(b) Define Distributed Computing and discuss the different types of Distributed Systems.

Distributed computing refers to a computing environment where tasks are divided among multiple interconnected systems, working collectively to achieve a goal.

Types:

1. **Client-Server Systems**:
   - Clients request services from centralized servers.
   - Example: Web browsers communicating with web servers.
2. **Peer-to-Peer Systems**:
   - All nodes share equal responsibilities for data and tasks.
   - Example: File-sharing networks like BitTorrent.
3. **Distributed Databases**:
   - Data is stored and processed across multiple locations.
   - Example: Google Spanner synchronizing databases globally.

(c) Explain any two enabling technologies for IoT.

1. **RFID (Radio Frequency Identification)**:
   - RFID technology uses electromagnetic fields to identify and track tags attached to objects automatically.
   - Tags consist of a microchip and an antenna, which store data and communicate with RFID readers.
   - It facilitates inventory management, asset tracking, and access control.
   - **Real-World Example**: Retailers like Walmart use RFID to streamline supply chain operations, ensuring accurate inventory and reducing losses.
2. **Wireless Sensor Networks (WSNs)**:
   - WSNs consist of spatially distributed sensors that monitor physical or environmental conditions, such as temperature, humidity, or pressure.
   - These sensors communicate wirelessly to transmit data to central nodes for processing.
   - They are integral to IoT applications in smart cities, environmental monitoring, and industrial automation.
   - **Real-World Example**: In agriculture, WSNs are deployed to monitor soil conditions and optimize irrigation, leading to water conservation and increased crop yields.

Question 8

(a) Explain any three innovative applications of IoT.

1. **Smart Agriculture**:
   - IoT technologies in agriculture enhance productivity and optimize resource utilization by providing real-time monitoring and actionable insights.
   - Smart sensors measure parameters such as soil moisture, temperature, humidity, and weather conditions.
   - Data collected helps farmers make data-driven decisions about irrigation schedules, pest control, and crop health.
   - **Example**: IoT-enabled precision irrigation systems ensure water is delivered efficiently to crops, conserving water and improving yields.
   - **Advanced Techniques**: Integration with satellite imaging and AI tools to predict crop diseases and optimize planting schedules.
2. **Connected Healthcare**:
   - IoT devices are revolutionizing healthcare by enabling remote monitoring and improved patient care.
   - Wearable health devices track vital metrics such as heart rate, blood pressure, oxygen levels, and glucose.
   - Smart medication dispensers remind patients to take medicines and alert caregivers in case of anomalies.
   - **Example**: Smartwatches equipped with ECG functionality, like Apple Watch, provide early warning for heart irregularities, potentially saving lives.
   - **Advanced Techniques**: IoT integration with electronic health records (EHR) for continuous data updates and AI-driven diagnosis.
3. **Smart Homes**:
   - IoT enables automation and remote management of various home functions, improving convenience, security, and energy efficiency.
   - Smart home devices include automated lighting, thermostats, security cameras, and appliances connected through centralized apps.
   - **Example**: Alexa-enabled smart thermostats adjust room temperatures dynamically based on occupancy patterns, reducing energy consumption.
   - **Advanced Techniques**: IoT systems integrated with machine learning adapt to user preferences over time, offering personalized experiences.

(b) Differentiate between Distributed Computing and Cloud Computing.

1. **Distributed Computing**:
   - Refers to a system where computational tasks are distributed across multiple interconnected systems, often in different physical locations.
   - Focuses on improving efficiency, fault tolerance, and resource sharing.
   - **Key Features**: Decentralized systems, often with a peer-to-peer structure.
   - **Applications**: Hadoop for big data processing, SETI@home for scientific research.
   - **Example**: In distributed databases like Apache Cassandra, data is partitioned and replicated across nodes to ensure high availability.
2. **Cloud Computing**:
   - Offers on-demand services like storage, compute power, and software applications via the internet.
   - Focuses on scalability, resource efficiency, and centralized management.
   - **Key Features**: Centralized service delivery model, pay-as-you-go pricing.
   - **Applications**: Hosting websites on AWS or deploying AI models using Google Cloud ML Engine.
   - **Example**: Cloud-based SaaS tools like Google Workspace and Microsoft 365 for collaboration and productivity.

(c) Write a short note on Online Social and Professional Networking.

Online networking platforms enable users to connect, collaborate, and share ideas within personal and professional contexts.

Expanded Examples:

1. **Social Networking**:
   - Platforms such as Facebook, Instagram, and Twitter foster personal connections by enabling users to share updates, photos, and engage with communities.
   - Social networking also serves as a channel for brand marketing and awareness campaigns.
   - **Example**: Instagram influencers engage with audiences, driving trends and consumer behavior.
2. **Professional Networking**:
   - Platforms like LinkedIn are tailored for career development, allowing users to showcase skills, seek jobs, and build professional networks.
   - Features such as endorsements, skill assessments, and recommendation systems enhance profile credibility.
   - **Example**: LinkedIn Learning provides courses to improve career-oriented skills and certifications, enriching user profiles.
3. **Emerging Trends**:
   - Cross-platform integrations, AI-driven recommendations for connections and content, and enhanced privacy controls are shaping the future of online networking.