CLOUD COMPUTING LAB

ASSIGNMENT 3 :

OPENSTACK

NAME : SWAPNADEEP MISHRA

ROLL NO : 002211001115

SECTION NO : A3

a) What is Openstack (Introduction).

OpenStack is an open-source cloud computing platform that allows users to create and manage both public and private clouds. It provides a set of software tools to manage and deploy cloud computing resources, such as compute (virtual machines), storage, and networking, through a web-based dashboard, command-line tools, or RESTful APIs.

Key Components of OpenStack:

1. **Nova**: Manages and provisions compute resources, including virtual machines and bare metal servers.
2. **Neutron**: Provides networking as a service, managing networks, IP addresses, and routers.
3. **Cinder**: Manages block storage, similar to a hard drive, which can be attached to running instances.
4. **Swift**: Offers object storage, allowing users to store and retrieve large amounts of unstructured data.
5. **Glance**: Manages and stores disk images, enabling the creation of instances from these images.
6. **Keystone**: Handles identity services for authentication and authorization across all OpenStack services.
7. **Horizon**: Provides the web-based dashboard for users and administrators to manage resources visually.

Use Cases:

- **Private Cloud**: Organizations use OpenStack to set up a cloud environment within their own data centers.
- **Public Cloud**: Service providers use OpenStack to offer cloud services to external customers.
- **Hybrid Cloud**: OpenStack can be used to bridge private and public clouds, allowing workloads to move seamlessly between them.

OpenStack is highly customizable, making it popular among enterprises, telecommunications companies, and research institutions looking for flexible cloud solutions.

b) Identify which level of virtualization it provides (SaaS, PaaS, IaaS, any other XaaS?)

OpenStack primarily provides ****Infrastructure as a Service (IaaS)**.**

Explanation:
- ****IaaS (Infrastructure as a Service)**:** OpenStack enables users to create and manage virtualized computing resources, such as virtual machines, storage, and networking, through a cloud platform. It allows users to provision, manage, and scale these resources on demand. This is the core functionality of OpenStack, making it an IaaS platform.

Comparison with Other XaaS:

- **PaaS (Platform as a Service)**: While OpenStack itself is not a PaaS, it can be used as the underlying infrastructure for a PaaS platform. PaaS provides a higher level of abstraction by offering development platforms and environments for building, deploying, and managing applications.

- **SaaS (Software as a Service)**: OpenStack is not SaaS. SaaS delivers software applications over the internet, typically accessed through a web browser. OpenStack, on the other hand, provides the infrastructure on which such applications could run.

So, OpenStack's primary role is in the IaaS category, providing the foundational cloud infrastructure that other services and applications can be built upon.

c) The role of each one of the components

1. Nova (Compute)

- **Role**: Nova is responsible for provisioning and managing compute resources in the cloud. It handles the creation, scheduling, and management of virtual machines (VMs), enabling users to deploy instances on-demand.
- **Key Functions**:
  - Manages the lifecycle of compute instances (VMs).
  - Supports various hypervisors (e.g., KVM, Xen, VMware).
  - Manages resource allocation and scaling.

2. Neutron (Networking)

- **Role**: Neutron provides "Networking as a Service" (NaaS), allowing users to create and manage networks, subnets, routers, and firewalls. It handles the networking infrastructure and ensures communication between compute instances.
- **Key Functions**:
  - Manages networking, including IP address allocation, routing, and firewalling.
  - Supports complex network topologies (e.g., VLANs, VXLANs).
  - Integrates with software-defined networking (SDN) controllers.

3. Cinder (Block Storage)

- **Role**: Cinder manages block storage, providing persistent storage volumes that can be attached to and detached from instances. It is analogous to traditional hard drives in cloud environments.
- **Key Functions**:
  - Provides persistent storage volumes to instances.
  - Manages storage pools from various backends (e.g., LVM, Ceph).
  - Supports volume snapshots and backups.

4. Swift (Object Storage)

- **Role**: Swift is responsible for storing and retrieving unstructured data objects via a RESTful API. It is designed to store massive amounts of data that can be accessed and managed over the network.
- **Key Functions**:
    - Provides highly available, distributed, and scalable object storage.
    - Stores unstructured data like backups, images, and archives.
    - Handles data replication and integrity.

5. Horizon (Dashboard)

- **Role**: Horizon provides a web-based graphical user interface (GUI) for managing OpenStack services. It allows users to administer and monitor their cloud resources without needing to use the command line.
- **Key Functions**:
    - Offers a user-friendly interface for managing instances, storage, and networks.
    - Provides access to project and user management features.
    - Displays metrics and logs for monitoring resources.

6. Keystone (Identity Service)

- **Role**: Keystone handles authentication, authorization, and service discovery within OpenStack. It acts as a centralized identity service for managing users and assigning roles to control access to resources.
- **Key Functions**:
    - Manages user accounts, roles, and permissions.
    - Provides token-based authentication for other OpenStack services.
    - Supports multi-tenant environments and integrates with external identity systems (e.g., LDAP).

7. Glance (Image Service)

- **Role**: Glance manages disk images, which are used to provision compute instances. It provides services for discovering, registering, and retrieving virtual machine images.
- **Key Functions**:
    - Stores and manages disk images and metadata.
    - Allows users to create and manage custom images.
    - Supports image formats like QCOW2, VHD, and RAW.

8. Heat (Orchestration)

- **Role**: Heat provides orchestration services, enabling users to automate the deployment of cloud applications and infrastructure. It uses templates to describe the infrastructure needed for an application.
- **Key Functions**:
    - Automates the creation and management of cloud resources (compute, storage, networking).
    - Uses templates written in YAML to define resource provisioning.
    - Supports scaling and auto-healing of resources.

9. Ceilometer (Telemetry)

- **Role**: Ceilometer collects and monitors metering data from the various OpenStack services, providing insights into usage for billing, monitoring, and benchmarking.
- **Key Functions**:
  - Collects metrics, events, and logs across the cloud infrastructure.
  - Supports usage reporting for billing and monitoring purposes.
  - Integrates with other services for performance tracking and alerts.

10. Barbican (Key Management)

- **Role**: Barbican provides secure storage, provisioning, and management of secrets such as encryption keys, certificates, and passwords.
- **Key Functions**:
  - Manages and stores encryption keys and secrets securely.
  - Provides APIs for retrieving and managing secrets.
  - Integrates with other OpenStack services to provide encryption at rest.

11. Trove (Database as a Service)

- **Role**: Trove provides scalable and reliable database as a service for OpenStack. It simplifies the management of relational and non-relational databases.
- **Key Functions**:
  - Automates the deployment and management of database instances.
  - Supports multiple database engines (e.g., MySQL, PostgreSQL).
  - Provides backup, restore, and scaling capabilities for databases.

12. Magnum (Container Orchestration)

- **Role**: Magnum offers container orchestration services, allowing users to manage containers as first-class resources within OpenStack.
- **Key Functions**:
  - Integrates with container orchestration platforms like Kubernetes, Docker Swarm.
  - Simplifies the deployment of containerized applications.
  - Provides multi-tenancy and security for container environments.

13. Sahara (Data Processing)

- **Role**: Sahara enables the provisioning and management of big data clusters like Hadoop or Spark within OpenStack.
- **Key Functions**:
  - Simplifies the deployment of data processing frameworks.
  - Provides integration with compute, storage, and networking resources.
  - Supports scaling and management of big data clusters.

14. Manila (Shared File System)

- **Role**: Manila provides shared file systems as a service, allowing instances to share storage resources.
- **Key Functions**:
    - Manages the provisioning and management of shared file systems.
    - Supports various file system backends (e.g., NFS, CIFS).
    - Enables multi-tenancy and access control for shared storage.

These components work together to provide a complete cloud infrastructure platform, offering users flexibility, scalability, and control over their cloud environment.

d) What are the mandatory components for the simplest deployment?

1. Nova (Compute)

- **Why It's Needed**: Nova is the core service that provides compute resources (virtual machines). It's essential for running and managing the virtual instances that users deploy in the cloud.

2. Neutron (Networking)

- **Why It's Needed**: Neutron handles networking, including IP address management, virtual networks, and security groups. It's required to enable communication between instances and external networks.

3. Keystone (Identity Service)

- **Why It's Needed**: Keystone is responsible for authentication and authorization, managing users, and providing security across all OpenStack services. It's necessary to control access to the cloud resources.

4. Glance (Image Service)

- **Why It's Needed**: Glance manages the disk images that are used to create instances. It's required to store and retrieve the base images from which virtual machines are launched.

5. Horizon (Dashboard)

- **Why It's Needed**: Horizon provides a web-based interface for users to manage their cloud resources. While technically not mandatory (as you could use CLI tools), it simplifies the management and monitoring of the cloud.

6. Cinder (Block Storage)

- **Why It's Needed**: Cinder is essential for providing persistent storage to instances. Even in the simplest deployments, most use cases require some form of storage that persists beyond the life of the instance.

Optional but Commonly Included Components:

- **Swift (Object Storage)**: Sometimes included if object storage is required, but not mandatory for basic deployments.
- **Heat (Orchestration)**: Useful for automating the deployment of resources but not mandatory.
- **Ceilometer (Telemetry)**: Used for monitoring and metering, typically added for larger deployments needing detailed usage data.

Minimal Deployment Example:

In the simplest form, the deployment might look like this:

1. **Nova** - Compute resource management.
2. **Neutron** - Networking.
3. **Keystone** - Identity service.
4. **Glance** - Image service.
5. **Cinder** - Block storage.
6. **Horizon** - Web-based dashboard for management.

This minimal set of components provides the essential infrastructure needed to deploy and manage virtual machines in an OpenStack environment.

e) What are the main competitors of Openstack? What are their strong and weak points?

1. VMware vSphere

- **Type**: Proprietary IaaS
- **Strengths**:
  - **Mature and Reliable**: VMware has been a leader in virtualization for years, offering a stable and mature platform.
  - **Enterprise Features**: Advanced features like vMotion, High Availability (HA), Distributed Resource Scheduler (DRS), and strong integration with VMware's broader ecosystem (e.g., NSX for networking, vSAN for storage).
  - **Strong Support and Ecosystem**: VMware provides extensive support and has a large ecosystem of partners and third-party integrations.
- **Weaknesses**:
  - **Cost**: VMware is proprietary software and can be significantly more expensive than open-source alternatives like OpenStack.
  - **Vendor Lock-In**: Organizations may find it challenging to migrate away from VMware once deeply integrated.
  - **Complexity**: Although feature-rich, VMware’s complexity can be a barrier, particularly for smaller organizations.

2. Amazon Web Services (AWS)

- **Type**: Public Cloud IaaS/PaaS
- **Strengths**:

    - **Extensive Services**: AWS offers a vast array of services beyond just IaaS, including databases, machine learning, IoT, and more.
    - **Global Reach**: AWS has a global network of data centers, providing low-latency access to services around the world.
    - **Scalability and Performance**: AWS is built to scale massively, with robust performance and reliability.
    - **Ecosystem and Innovation**: Continuous innovation with new services and features added regularly.
- **Weaknesses**:
    - **Cost**: While flexible, costs can quickly escalate, especially for long-term use without proper cost management.
    - **Complex Pricing**: AWS pricing can be complex and difficult to predict, leading to unexpected costs.
    - **Vendor Lock-In**: Heavy reliance on AWS-specific services can make it difficult to migrate to another provider or on-premises infrastructure.

3. Microsoft Azure

- **Type**: Public Cloud IaaS/PaaS
- **Strengths**:
    - **Integration with Microsoft Products**: Strong integration with Microsoft's ecosystem, including Windows Server, Active Directory, Office 365, and SQL Server.
    - **Hybrid Cloud Capabilities**: Azure Stack allows for seamless integration between on-premises infrastructure and the Azure cloud.
    - **Global Reach and Compliance**: Azure has a large global network with strong compliance and regulatory support for various industries.
    - **PaaS Offerings**: Extensive platform services for developers, including DevOps, AI, and analytics tools.
- **Weaknesses**:
    - **Cost**: Similar to AWS, costs can add up quickly, particularly with extensive use of PaaS offerings.
    - **Complexity**: Managing and configuring Azure services can be complex, especially for organizations without existing Microsoft expertise.
    - **Less Mature Than AWS**: While Azure has rapidly grown, AWS still leads in terms of service maturity and breadth.

4. Google Cloud Platform (GCP)

- **Type**: Public Cloud IaaS/PaaS
- **Strengths**:
    - **Data and Analytics**: Strong capabilities in big data, machine learning, and AI, with services like BigQuery, TensorFlow, and Google Kubernetes Engine.
    - **Performance**: High-performance infrastructure with strong networking capabilities, especially in handling large-scale workloads.
    - **Innovative**: Leading in Kubernetes and containerization, which is crucial for modern cloud-native applications.
- **Weaknesses**:

    - **Market Share and Ecosystem**: GCP lags behind AWS and Azure in market share and ecosystem breadth, which can be a consideration for enterprise adoption.
    - **Enterprise Focus**: GCP's customer base is smaller, and it may have less focus on traditional enterprise workloads compared to AWS or Azure.
    - **Complex Pricing**: Like other public clouds, GCP's pricing model can be difficult to navigate, leading to potential cost management issues.

5. Red Hat OpenShift

- **Type**: Kubernetes-based PaaS
- **Strengths**:
    - **Kubernetes and Containers**: Built on Kubernetes, OpenShift is designed for managing containerized applications, with a focus on DevOps and CI/CD pipelines.
    - **Enterprise Support**: Backed by Red Hat, OpenShift offers strong enterprise support and integrates well with other Red Hat products.
    - **Hybrid and Multi-Cloud**: OpenShift can be deployed on-premises, on public clouds, or in hybrid environments, providing flexibility.
- **Weaknesses**:
    - **Complexity**: OpenShift's enterprise features add complexity, which can be challenging for organizations not familiar with Kubernetes.
    - **Cost**: While open-source at its core, the enterprise version (Red Hat OpenShift) comes with licensing costs, which can add up.
    - **Learning Curve**: The steep learning curve for Kubernetes and container orchestration can be a barrier for some organizations.

6. CloudStack

- **Type**: Open-source IaaS
- **Strengths**:
    - **Simplicity**: CloudStack is considered easier to deploy and manage than OpenStack, with a more straightforward architecture.
    - **Enterprise Features**: It offers robust features like virtual machine management, networking, and storage, similar to OpenStack.
    - **Community Support**: Like OpenStack, it benefits from a strong open-source community, though smaller.
- **Weaknesses**:
    - **Smaller Ecosystem**: CloudStack has a smaller ecosystem and less third-party support compared to OpenStack.
    - **Development Activity**: OpenStack generally sees more active development and innovation, with a larger community driving its growth.
    - **Limited Flexibility**: While simpler, CloudStack might not offer the same level of customization and flexibility as OpenStack.

7. Alibaba Cloud

- **Type**: Public Cloud IaaS/PaaS
- **Strengths**:

    - **Market Leader in Asia**: Strong presence and dominance in the Asia-Pacific region, especially in China.
    - **Wide Range of Services**: Offers a comprehensive suite of cloud services similar to AWS and Azure, including AI, big data, and IoT.
    - **Cost-Effective**: Generally offers competitive pricing, making it an attractive option for businesses in the APAC region.
- **Weaknesses**:
    - **Global Presence**: While strong in Asia, Alibaba Cloud's presence and data center coverage outside of Asia are less extensive compared to AWS and Azure.
    - **Compliance and Trust**: Non-Asian companies may have concerns about data privacy, compliance, and government oversight when using Alibaba Cloud.
    - **Documentation and Support**: English-language documentation and support can be less comprehensive compared to its competitors.

Each of these competitors offers different advantages and challenges depending on an organization's specific needs, such as cost, scalability, ease of use, and ecosystem support. OpenStack remains a popular choice for organizations that want an open-source, customizable solution with strong community support and the ability to avoid vendor lock-in.