GAE Laucher

1. What is meant by Google App Engine?

Google App Engine (GAE) is a **Platform as a Service (PaaS)** provided by Google that allows developers to build and host web applications on Google's infrastructure. It automatically handles things like **scaling**, **load balancing**, **monitoring**, and **server management**, so developers can focus on writing code.

---

2. How does Google App Engine work?

- Developers write applications using supported languages like **Python, Java, Node.js, Go**, etc.
- The code is deployed to App Engine using the **Google Cloud SDK**.
- App Engine automatically:
    - **Deploys** the app to Google's data centers.
    - **Scales** the app up or down based on traffic.
    - **Manages** infrastructure (no need to manage servers).
    - Provides features like **automatic updates**, **security patches**, and **monitoring tools**.
- It uses a **sandbox environment** to run applications securely and efficiently.

---

3. What are the different versions of Google App Engine?

Google App Engine has two main **environments** (versions):

1. **Standard Environment:**
    - Uses **predefined runtimes** (e.g., Python, Java, Node.js).
    - **Fast startup time** and **auto-scaling**.
    - **Sandboxed** environment with some restrictions.
2. **Flexible Environment:**
    - Supports **custom runtimes** using Docker.
    - Runs on **Google Compute Engine VMs**.
    - Offers **more control** over the environment (e.g., libraries, system access).
    - Better suited for apps with **higher resource needs** or **custom dependencies**.

File Transfer Between Virtual Machines

Q1. What is virtualization?
A: Virtualization is the creation of virtual servers, devices, or resources that behave like real physical systems. It allows multiple virtual environments to run on a single physical system.

Q2. What is operating-system-level virtualization?
A: It's a server-virtualization method where the OS kernel allows multiple isolated user-space instances, also called containers, VEs, or VPS.

Q3. What is hardware/platform virtualization?
A: It refers to creating virtual machines that act like real computers. Software runs independently of the underlying hardware on these virtual machines.

Q4. What is a hypervisor?
A: A hypervisor (or Virtual Machine Monitor, VMM) is software or firmware that creates and manages virtual machines. It allows multiple guest OSes to run on a host system.

Q5. What are host and guest machines in virtualization?
A: The **host** machine is the actual physical computer, and the **guest** machine is the virtual machine running on it.

- **VirtualBox and Ubuntu Questions**

Q6. What is Oracle VirtualBox?
A: VirtualBox is a free, open-source virtualization software that allows you to run multiple operating systems like Ubuntu, Windows, etc., on a single physical machine.

Q7. What is Ubuntu?
A: Ubuntu is a free and open-source Linux operating system. It can be downloaded and installed on any number of computers.

Q8. How do you create a VM in VirtualBox using a VMDK image?
A:

1. Open VirtualBox → Click "New".
2. Name the VM and choose OS type and version.
3. Allocate memory (1024 MB recommended).
4. Choose "Use an existing virtual hard disk" and select the downloaded VMDK image.
5. Finish creating the VM.

Q9. What is a NAT Network and why do we use it?
A: A NAT (Network Address Translation) Network allows VMs to communicate with each other internally and access external internet via the host. It's needed to transfer files between VMs.

Q10. How do you create a NAT Network in VirtualBox?
A:

1. Go to File → Preferences → Network.
2. Click '+' to add a new NAT Network.
3. Edit its name and CIDR (e.g., 172.168.2.0/24) and save.

Q11. How do you assign this NAT network to your VM?
A:

1. Go to VM Settings → Network.
2. Enable Adapter 1 → Choose "Attached to: NAT Network".
3. Select the created NAT Network from the list.

Q12. How do you check the IP address of your VM?
A: Run the following commands in the Ubuntu terminal:

bash

CopyEdit

sudo apt update

sudo apt install net-tools

ifconfig

Q13. How do you create a file in Ubuntu terminal?
A:

bash

CopyEdit

touch transfer.txt

nano transfer.txt # then write content, press Ctrl+X, then Y to save.

Q14. What is the command to transfer files between two VMs using SCP?
A:

bash

CopyEdit

scp transfer.txt vagrant@<destination_IP>:/home/vagrant

Example:

bash

CopyEdit

scp transfer.txt vagrant@172.168.2.5:/home/vagrant

Q15. What is the default username if you are using Vagrant boxes?
A: The default username is usually vagrant.

OpenStack Virtual Machine Launch Using TryStack

Q1. What is OpenStack?
A1. OpenStack is an open-source cloud computing platform that provides Infrastructure-as-a-Service (IaaS). It allows users to create and manage virtual machines, storage, and networks using a dashboard or API.

Q2. What is TryStack?
A2. TryStack is a free online demo version of OpenStack that lets users explore and learn OpenStack features without installing it on their own machine.

Q3. What are the minimum hardware requirements to install OpenStack?
A3.

- 4 GB RAM
- 4 CPU units
- 30 GB Disk Space

Q4. Why do we use the stack user for installing OpenStack?
A4. The stack user is a non-root user with sudo privileges used to securely install and manage OpenStack. It helps avoid accidental changes as root.

Q5. What is the purpose of the local.conf file in DevStack?
A5. The local.conf file is used to automate the OpenStack installation by pre-defining passwords for components like admin, database, RabbitMQ, and services.

Q6. Which command is used to install OpenStack using DevStack?
A6.

bash

CopyEdit

./stack.sh

Q7. After successful installation, where can you access the OpenStack dashboard?
A7. The dashboard can be accessed via a web browser at:

perl

CopyEdit

http://<your_IP_address>/dashboard/

Q8. Why is creating a network necessary before launching an instance?
A8. Without a network, the virtual machine (instance) cannot communicate or access the internet. OpenStack requires at least one network to launch and connect instances.

Q9. What is Cirros in OpenStack?
A9. Cirros is a minimal Linux distribution used primarily for testing OpenStack instances. It is lightweight and quick to boot.

Q10. What is a "Flavor" in OpenStack?
A10. A Flavor in OpenStack defines the compute, memory, and storage capacity of an instance. It specifies resources like CPU, RAM, and disk space.

Q11. What are security groups in OpenStack?
A11. Security groups in OpenStack act like virtual firewalls. They control inbound and outbound traffic to instances based on defined rules.

Q12. How do you access the virtual machine's command line after launching it?
A12. You can access the VM's command line by clicking on the **"Console"** tab of the instance in the OpenStack dashboard.

Q1. What is PaaS?
A1. PaaS stands for *Platform as a Service*. It is a cloud computing model that provides a platform and environment to allow developers to build, deploy, and manage applications without managing the underlying infrastructure.

Q2. Give some examples of PaaS providers.
A2.

- Heroku
- Google App Engine
- Microsoft Azure App Services
- AWS Elastic Beanstalk
- PythonAnywhere (for Python apps)

Q3. What are the key features of PaaS?
A3.

- Automatic scaling
- Application hosting
- Built-in security
- Integrated development tools
- Database integration
- Deployment automation

Q4. What is the advantage of deploying applications using PaaS?
A4.

- No need to manage servers
- Faster deployment
- Easy scalability
- Supports multiple programming languages
- Continuous integration and delivery support

Q5. Which programming languages are commonly supported by PaaS platforms?
A5.

- Python

- Java
- Node.js
- PHP
- Ruby
- .NET

Q1. What is Render.com?
A1. Render.com is a modern cloud platform that offers PaaS services for developers to build, deploy, and host web applications, static websites, databases, and more with ease.

Q2. Why is Render.com considered a PaaS platform?
A2. Render abstracts infrastructure management and provides developers with a platform to deploy and manage applications without worrying about underlying servers or scaling, which makes it a PaaS.

Q3. What types of applications can you deploy using Render?
A3.

- Static sites (HTML, CSS, JS)
- Web services (Node.js, Python, Go, etc.)
- Background workers
- Cron jobs
- PostgreSQL databases
- Docker containers

Q4. What are the steps to deploy a web app using Render?
A4.

1. Push your code to GitHub or GitLab
2. Sign up or log in to https://render.com
3. Click on **"New Web Service"**
4. Connect your GitHub/GitLab repository
5. Choose the branch and build command
6. Define the start command (like npm start, python app.py)
7. Click **"Create Web Service"**
8. Render builds and deploys your app automatically

Q5. What is the difference between Static Site and Web Service in Render?
A5.

- **Static Site**: Used for HTML/CSS/JS-only projects with no backend.
- **Web Service**: Used for dynamic apps with a backend (e.g., Node.js, Django, Flask).

Q6. What kind of build and start commands are used in Render for Node.js apps?
A6.

- **Build Command:** npm install
- **Start Command:** npm start or node app.js

Firebase Auth + GAE + Cloud Datastore

Q1. What is Firebase Authentication?
A1. Firebase Authentication is a service that enables user authentication using passwords, phone numbers, and identity providers like Google, Facebook, and Twitter.

Q2. Why is Firebase Authentication used in your assignment?
A2. Firebase Authentication is used to securely verify user identity and manage sign-up/sign-in flows, which is critical for protecting user data.

Q3. What is Google App Engine (GAE)?
A3. Google App Engine is a serverless platform that allows you to deploy web applications and APIs without managing the underlying infrastructure.

Q4. What is the standard environment in Google App Engine?
A4. The standard environment is a sandboxed environment that supports specific languages (like Python, Node.js) with automatic scaling and built-in security.

Q5. What is Google Cloud Datastore?
A5. Cloud Datastore is a NoSQL document database built for automatic scaling, high performance, and ease of application development.

Q6. What is the flow of your assignment?
A6.

1. User signs in using Firebase Authentication
2. App Engine receives the Firebase ID token
3. Token is verified on the backend (GAE)
4. On successful verification, user data is stored in Google Cloud Datastore

---

Q7. How is the Firebase ID token verified in App Engine?
A7. The Firebase Admin SDK is used in the backend (deployed on GAE) to verify the ID token using Firebase's public keys.

Q8. What kind of data is stored in the Datastore?
A8. User credentials like UID, email, display name, and timestamp are stored as entities in the Datastore.