AWS EC2

=> Elastic Compute Cloud

=> It is Most demanded service in AWS

=> It is Used to create Virtual Machines in AWS cloud

=> EC2 VM is called as EC2 instance

EC2 Instance = Computer / Server / VM / Virtual Machine / V Box

=> EC2 instance is re-sizable (we can change configuration based on demand)

=> EC2 is a paid service (hourly billing)

=> EC2 VM Minimum billing period is 1 hour

9:00 AM to 9:15 AM => 15 mins => 1 hour billing

9:30 AM to 9:50 AM => 20 mins => 1 hour billing

Note: To encourage beginners, AWS provided "t2.micro/t3.micro" for 1 year free (monthly 750 hours)

=> Ec2 is Regional service.

EC2 Instance Types

=> Amazon EC2 (Elastic Compute Cloud) offers a variety of instance types to suit different use cases.

1) General Purpose

- T series : t2, t3, t4...
- M series : m6g, m5, m5a, m5n...

2) Compute Optimized (High-performance processors)

- C series : c7gm c6i, c6g, c5, c5a

3) Memory Optimized (High memory capacity)

- R series : r6g, r5, r5a, r5n
- X series : x2gd, x1e

4) Storage Optimized (High-performance local storage using NVMe SSDs or HDDs)

- I series : i4i, i3, i3en
- D series : d2

5) Accelarated Computing (Powerful GPU and FPGA accelerators)

- P series : p4, p3, p2
- G series : g5, g4ad, g4dn
- inf series : inf1

6) High Performance Computing (HPC) (massive computational power & latest-generation processors)

- H series : hpc6id

What is AMI

=> AMI stands for Amazon machine image

=> An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance.

a) Quick Start AMIs

Ex: amazon linux, ubuntu, red hat, suse, windows, mac os...

b) Third Party AMIs (Paid)

c) Community Based (Free, Public Access)

Note: To launch EC2 instance AMI is required.

Note: We can also create our own AMIs

Process : Select EC2 VM => Actions => Images and Templates => Create Image

Key Pair

=> In AWS, a Key Pair is a set of security credentials used to securely connect to EC2 instances.

=> It consists of two parts:

Private Key â€“ Kept by the user and used to connect to the instance.

Public Key â€“ Stored in AWS and attached to the EC2 instance.

Note-1: When we are connecting with EC2 instance, private key and public key handshake will happen. If handshake is successfull then only we can connect with that machine.

Note-2: One key pair we can use to launch multiple EC2 instances.

Security Groups

=> A Security Group (SG) in AWS acts as a virtual firewall that controls inbound and outbound traffic for EC2 instances

=> In Security Group we can define 2 types of rules

1) In-Bound Rules : Control Traffic comming into the instance

2) Out-Bound Rules : Control traffic going out from instance.

SSH	22
HTTP	80
HTTPS	443
RDP	3389
MySQL	3306

=> Security group rules are stateful. If we allow inbound traffic, then it is applicable for outbound also by default.

=> In Security group we can configure only 'Allow' rules. By default all are 'Deny'.

=> Deny-All by Default â€“ By default, all incoming traffic is blocked, and outgoing traffic is allowed.

Note: One Security group we can attach with Multiple Ec2 Instances and One Ec2 instance can have multiple security group also.

=====================

What is VPC in EC2

====================

=> VPC stands for Virtual Private Cloud

=> VPC provides network required to launch ec2 instance

Note: To encourage beginners, AWS provided Default VPC to launch EC2 instances.

Note: Based on requirement, we can create custom VPC and we can use custom vpc to launch Ec2 instance.

==============

What is EBS

==============

=> EBS stands for Elastic Block Store

=> It is block level storage device (Hard Disc / SSD)

=> For EC2 instances storage will be provided by EBS.

=> For windows VM, we will get 30 GB as default volume size.

=> For Linux VM, we will get 8 GB as default volume size.

Note: EBS volume can have upto 16 TB (16000 GB)

==============================

Lab Task-1 : Create EC2 VM with Windows AMI and connect to it using RDP client

Lab Task-2 : Create EC2 VM with Linux AMI and connect to it using SSH client

==============================

===============================

Types of IP's in AWS Cloud

===============================

=> We have 3 types of IP's in AWS cloud

1) private ip

2) public ip

3) elastic ip

=> Private IP is a fixed IP in AWS. It is used for internal communication (With in VPC).
=> Even if we restart the EC2 instance, private IP will not change.

=> public ip is a dynamic IP in aws. It is used to connect with EC2 VM from outside.
=> When we restart our EC2 instance then public ip gets changed.

=> If we want fixed public ip then we need to use Elastic IP.

=> Elastic IPs are commercial (bill will be generated).

What is user-data in EC2 VM ?

=> It is used to execute script while launching ec2 machine.

=> User data will execute only once

=> Create EC2 VM (amazon linux) with below user data

```
#! /bin/bash

sudo su
yum install httpd -y
cd /var/www/html
echo "<html><h1>Life Insurance Server - 1</h1></html>" > index.html
service httpd start
```

What is Load Balancer ?

=> When we run our application in single server then we have to face below challenges

1) One server should handle all reqs

2) Burden will increase on server

3) Response will be delayed for clients

4) Server can crash

5) Single Point Of Failure

=> To avoid above problems, we will run our application in multiple servers.

=> We will use Load Balancer to distribute load to all servers in the round robbin fashion.

=> We have below advantages with Load Balancer

1) App will run in multiple severs

2) Load will be distributed

3) Burden will be reduces on servers

4) Fast Performance

5) High Availability

=> In AWS we have 4 types of Load Balancers

1) Application Load Balancer (ALB)
2) Network Load Balancer (NLB)
3) Gateway Load Balancer (GLB)
4) Classic Load Balancer (previous generation)

Load Balancer Lab Task

Step-1:: Create EC2 VM-1 with below user data

```
#! /bin/bash

sudo su
yum install httpd -y
cd /var/www/html
echo "<html><h1>Life Insurance Server - 1</h1></html>" > index.html
service httpd start
```

Step-2 :: Create EC2 VM-2 with below user data

```
#! /bin/bash

sudo su
yum install httpd -y
cd /var/www/html
echo "<html><h1>Life Insurance Server - 2</h1></html>" > index.html
service httpd start
```

Step-3 :: Add these 2 instances to one "Target Group"

Step-4 :: Create Load Balancer with Target Group (ALB)

Step-5 :: Access Load Balancer DNS in browser

============
Assignment
============

Microservices Load Balancing with multiple Target Groups : https://www.youtube.com/watch?v=QvEJ8--zneU

===========
OSI Model
===========

=> OSI stands for Open Systems Interconnection

=> IT represents 7 layers for client-server communication

Layer-7 : Application Layer

- Human and Computer Interaction Layer
- Applications can access network services

Layer-6 : Presentation Layer

- Ensures that data is in usable format
- Data Encryption occurs
- Translates data formats between applications and networks

Layer-5 : Session Layer

- Maintains connections between devices
- Responsible for controlling ports and sessions

Layer-4 : Transport Layer

- Transmits data using tranmission protocols including TCP and UDP
- Ensures end-to-end data delivery process

Layer-3 : Network Layer

- Decide which physical path the data will take
- Manages IP addressing, routing, forwarding of data

Layer-2 : Data Link Layer

- Transfers data between adjacent network nodes

Layer-1 : Physical Layer

- Tranmists raw bits over physical medium

==========================

Application Load Balancer
==========================

=> Works at Layer-7 (Application layer)

=> Protocols : HTTP and HTTPS

=> Routing will happen based on content(URL, Header, Hostname)

=> Usecases : Web applications, Microservices, APIs

==========================

Network Load Balancer
==========================

=> Works at Layer-4 (Transport layer)

=> Protocols : TCP and UDP

=> Handles millions of requests per second

=> Usecases : High Performance workloads (Ex: Gaming, IoT, real-time streams data)

=======================

Gateway Load Balancer
=======================

=> A gateway works at Layer 3 (Network Layer) and above because it translates protocols between different networks.

=> It can also operate at Layer 7 (Application Layer) when handling application-specific translations (e.g., API Gateways).

=> For global applications requiring traffic distribution across regions.

===============

Auto Scaling
===============

=> It is used to adjust the capacity required to handle the load of our application.

=> If requests are increasing then servers should be increased and if requests are decreasing then servers should be reduced.

=> If we use Auto scaling then it will increase or decrease the no.of servers based on incoming traffic.

=> We have below advantages with Auto Scaling

1) Cost Management

2) High Availability

3) Fault Tolerance

Note: Auto Scaling continuously monitors CPU usage, memory, network traffic, or request count.

Example: If CPU usage exceeds 80%, new instances are launched.

Example: E-commerce sites add more servers during sales events.

=======================
Horizontal Vs Vertical
=======================

Horizontal Scaling : Adding more servers (instances) to handle increased traffic.

Vertical Scaling : Upgrading the existing server (CPU, RAM, storage) instead of adding new ones.

==============
What is EBS
==============

=> EBS stands for Elastic Block Store

=> It is block level storage device (Hard Disc / SSD)

=> When we create ec2 instance then "EBS Volume" gets created automatically.

Note: If we remove EBS volume from EC2 instance then we can't start/use that EC2 instance.

=> In EBS, we have 2 types of volumes (storages)

1) Root Volume

2) Additional Volume

Note: When we launch EC2 instance by default we will get one Root volume.

=> Root volume is mandatory to launch EC2 instance.

=> Additional EBS volumes are optional devices (we can add/remove)

=> For windows VM, we will get 30 GB as default volume size.
=> For Linux VM, we will get 8 GB as default volume size.

Note: EBS volume can have upto 16 TB (16000 GB)

=> One EC2 VM can have multiple EBS volumes.

=> One EBS volume can be attached to only one EC2 VM at a time.

=> EBS volumes are Availability Zone Specific

```
Mumbai : ap-south-1
                        ap-south-1a
                        ap-south-1b
                        ap-south-1c
```

Note: In which availability zone our EC2 VM got created, in same availability zone we have to create

EBS volumes for attaching.

Snapshots

=> Snapshots are used for volumes backup

=> snapshots are region specific

=> From Volume we can create snapshot and from snapshot we can create volume

```
        volume =====> snapshot ========> volume
```

=> Snapshot can't be attached to EC2 instance directley
(volumes can be attached to ec2 instances directley).

Assignment : How to recover EC2 VM when we lost pem file

EC2 Summary

1) What is EC2 and Why

2) EC2 instance types

3) What is AMI

4) What is Key Pair

5) What is Security Group (inbound & outbound)

6) Windows VM launch & RDP Client

7) Linux VM launch & SSH Client

8) What is user data

9) Static Website Hosting using EC2

10) Load Balancer & Types

11) OSI Model

12) What is Auto Scaling Group

13) Types of IPs

14) EBS Volumes

15) EBS Snapshots