Date
Page

③ Module 3 → Introduction to AWS.

● Objectives.

a) History of AWS.

b) AWS Global infrastructure.

c) Customer & AWS shared responsibility model.

d) Architected framework of AWS.

e) Cost & billing considerations.

ⓐ History of AWS

⇒ Today AWS offers more then 175+ services. which includes → computing, storage, Analytics, IoT, (Networking & content). Developer tools, Business app, ML, Management & Security

⇒ Amazon → 1994

Amazon simple queue service is released as first IT service → 2004

AWS → 2006

AWS releases AI services → 2017

⇒ Amazon (AWS), Microsoft (Azure), Google (Cloud), Alibaba cloud, Oracle, IBM.

$\underbrace{\text{Amazon (AWS), Microsoft (Azure), Google (Cloud)}}_{\text{top 3}}$.

ⓑ AWS Global Infrastructure.

World into ⇒ Regions (A place where multiple Availibity zone are placed) into → Availability zone (A place where one or more data centers for providing services) ⟷ Edge locations For linking different regions (around the world) together.

→ Planning for failure is necessary & amazon takes care of that.

∴ If a file is stored in Amazon S3 (storage facility), it's copied & stored in multiple data centers.

Having multiple database & computing server is a good practice.

(c) ⇒ AWS Shared Responsibility

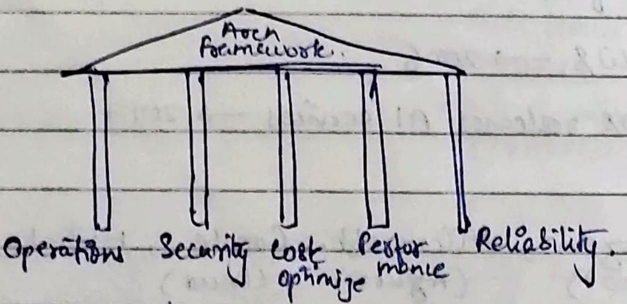
→ Divided into 2 parts

AWS responsible for security OF the cloud [Physical security of centers, → infrastructure, services].

Customer is responsible for their security IN the cloud. [what you store, who has access, services used, admin rights].

(d) AWS Well Arch Framework

⇒ 5 pillars



→ AWS provides AWS Arch FW tool → to test your system.

(e) Cost & Billings

→ Total Cost of ownership (TCO) includes :-

→ Procurement, Management, Maintainence & decommision of hardware resources.

Pricing models

i) Pay as you go.

ii) Save when you reserve. (1 yr, 3yr contracts)

iii) Pay less by using more. (more use, less pay of a service.)

AWS free tier → Always free
→ 12 months free
→ Trials.
} Ranges from service to service.

Billing dashboard also availiable.

Module 4 → AWS core services.

•) Objectives.

a) Different types of services.

b) Main categories of services

c) Each service description.

d) Monolithic vs Microservices Architecture.

Monolithic → If one component fails, app as a whole might fail.

Microservices → High reliability, loosely coupled, one failure doesn't stop the whole application.

a) Types of Services

→ Managed Services
Fully managed Services
Serverless Services
} Three types in AWS

↑ more flexible & customizable

Ⓒ Each Service Description

i) Amazon VPC (Virtual Private Cloud).

→ Have your AWS resources in a isoloted virtual network.
↓
Own IPs, Subnets, etc

→ Add security layers to your network.

→ Create your own virtual data center. by connecting it with EC2 & RDS.

(ii) Amazon EC2 (Elastic Compute Cloud).

→ A web server providing computing capacity in cloud.

→ Predicts capacity & volume needs, makes it easier to scale up as well as scale down when needed. (seasonal needs).

→ Used with VPC & RDS.

→ Host databases & Applications, On demand computing → Backup & Recovery.

(iii) Amazon RDS (Relational Database Service).

→ As name suggests it's a relational database service. Administ tasks and setup are managed automatically.

→ Acc to database needs can scale up & down, hence managing your costs.

→ web & mobile app, games, ecommerce applications.

If customization is needed RDS can be run with EC2. VPC is used anyway.

Support ticket needed when expecting high traffic.

(iv) Amazon Cloudwatch

→ Used for monitoring & observation by DevOps enggs, developers & security engineers.

→ It provides actionable insights & data, to monitor your applications.

→ Shows in form of logs & metrics. An overview shown.

(v) Amazon SNS (Simple Notification Service)

→ web service which makes it easy to setup & operate (or) send notifications from the cloud. Eg:- ~~from~~ developers of app to subscribers.

→ Subscribers should be aware of changes happening to their apps regularly.

→ Must be working with a monitoring service such as Lamdba (or) Cloudwatch.

(vi) Lambda AWS

→ A serverless compute service that you can use to run func code without provisioning (or) managing servers.

→ whatever service your code needs is in auto given to it by lambda. Eg:- Scaling up of EC2.

→ Used with S3 & SNS.

→ Real time file & stream processing, web apps.

(vii) **AWS IAM (Identity & Access Management).**

- A centralized security management system that is included in every AWS account, to control access to cloud (AWS) services.
- IAM User → Given to each account.
- Controls access with accordance to what is paid for by the user.

(viii) **Amazon DynamoDB**

- Serverless non-relational database.
- Rest same as RDS.

(ix) **Amazon S3 (Simple Storage Service).**

- Object storage built to retrieve and store data of any amount from anywhere at anytime.
- No estimation of storage space is required, you can store as many files as needed in a bucket.
- files uploaded are replicated into multiple availability zones to avoid loss of data.
- Not build for block storage.