**AWS**

Day: 1 Introduction & IAM User Service

Day: 2 EC2 & VPC

Day: 3 VPC (SG + NACL) & Route 53

Day: 4 Project

Day: 5 S3 Bucket

Day: 6 AWS CLI

Day: 7 AWS CLI & CFT

Day: 8 CI/CD (Code Commit)

Day: 9 Project on CI and Building Application in AWS

Day: 10 AWS Ultimate CICD Pipeline (Skipped)

Day: 10 AWS CLOUD WATCH & LAMBDA

Day: 11 AWS LAMBDA, COST OPTIMIZATION

Day: 12 AWS Cloud Front, ECR vs Docker Hub

Day: 13 ECS🡺Elastic Container Service

Day: 14 ECS, Secret Management, AWS Config

Day: 15 Load Balancer

**DAY:1**

**1.CLOUD**🡺Instead of Buying the Physical Server to Deploy the Application we can use Virtualisation and can use virtual Server with IP Addresses and can work accordingly with lesser Costs.

**2.PRIVATE CLOUD**🡺If Team A uses Cloud with their Instance and that Team A are not allowed to give the credentials of instance to other organizations.

**3.PUBLIC CLOUD**🡺Anyone with AWS OR AZZURE Account Irrespective of any organization can create instances Ex: -Amazon, Microsoft, Google

**\***Can create a Virtual Private cloud in a Public Cloud

-----------------------------------------------------------------------

DAY:1

**IAM: --**

IAM Service is used to create IAM user or child User from Root user

In AWS, **IAM is a combination of Both Authentication & Authorization** for child or IAM User

**IAM helps to create & describe the user details, user permissions**

**(USERS + POLICIES + GROUPS) + Roles ==IAM**

Users=User Details (Credentials, Info of User)

Policies= What are the actions can user do

groups= Category of user (Helps to Set the Policies in smart way in single step) by creating the category of people only can-do particular actions

Roles = Giving Temporary Access to AWS Account with [Access Key + Secret Access key + Session Token]

DAY:2

**EC2🡺Elastic Cloud Compute**

Elastic= in AWS Many Services have prefix "Elastic" that can increase or decrease the service

Cloud= in AWS cloud

Compute= Request Instance (CPU + RAM + Disk)

**So EC2=Give me a Virtual Server in AWS cloud which is in Elastic Nature**

EC2 Instance: --**(**EC2 Creation, EC2 Instance Connection, to deploy application in EC2 Instance,

**How to access the application from outside world)**

**Accessing the Application from Outside is done through MobaXterm**

-----------------------------------------------------------------------

DAY:2

**VPC🡺Virtual Private Cloud:**

if a person needs to reach the Application it should go through the below path

**Person🡺Instance [Internet Gateway 🡺Public Sub Net 🡺 (NAT Gateway) 🡺Load Balancer 🡺Route Table 🡺Security Group🡺Private Sub Net or Application]**

NAT Gateway is useful to hide the IP Address of the Server or Application

DAY: - 3

**VPC (SG = Security Groups + NACL)**

* **VPC is requested by DevOps Engineer** based on the range in Ip Address Based

[Ex: - IP Address (10.1.0.0/16) = 65536 Components or Subnets]

* In VPC can have multiple projects and in each project can have Minor Projects.

**Workflow: --**

* User Enters into VPC through Internet gateway and Access the Load Balancer through Public Subnet
* Load Balancer have rights to access the Private Sub Net.

🡺We can **Add more Security in Each Subnet Level using NACL** or **instead we can add Security at Application in EC2 Instance using Security Groups** and NACL helps to increase the Security

**Primary Difference B/W SG and NACL: -**

SG==> Security Groups manages the traffic to the Application by maintaining Particular Ports.

Ex: -8080 or Any port

SG🡺 i) Inbound Traffic ii) Outbound Traffic

User🡺Inbound Traffic🡺 [Instance [Application]]🡺Outbound Traffic🡺Google.com

**Ex: -**

* User🡺Inbound Traffic🡺[Amazon.com]🡺Outbound Traffic **🡺** Razo Pay or Amazon Pay
* User reaches Amazon Website then Amazon Website reaches Payment Website

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**BY DEFAULT:**

**AWS Restricts all Inbound Traffic (All Ports) to Application**

**AWS Allows All Outbound Traffic (All ports Except Port:25) from Application**

**NACL🡺Network Access Control List**

SG is Applied at EC2 Instance Level

**SG is only for allowing**

where

NACL is Applied at Subnet Level

**NACL is for both Allowing and Denying**

* Created a VPC and Instance with Subnet
* Connected the instance in Mobaxterm, now trying to access the python Application which is EC2 in VPC

command to run in Mobaxterm to create a server and to run it is "python3 -m http.server 8000"

-------------------------------------------------------------------------

Day:3

ROUTE 53: -

Route 53 in AWS Provides DNS as a Service.

DNS=>Domian Name System

DNS Service is the one that maps the domain Name with IP Address

Every Application contains IP Address

Instead of Saying IP Address to others to Access the Application

we can Simply say the Domain Name like Google.com, pavan.com or pavan.in or pavan.xyz

So, Route 53 lies between Internet gateway and Load Balancer

This Route 53 helps to map the Domain Name to IP Address of a Load Balancer and after that the process is same

To Purchase the Domain Name, we can go to GoDaddy or AWS too

AWS Has Option that Domain Name can buy in GoDaddy and can integrate that Domain Name in Route 53

Before Request got held the Route 53 goes to Header Section to find the Domain Name

If the Domain Name Found then IP Address got reserved for it and the Request will go from Header Section through IP Address

-----------------------------------------------------------------------

Day: 4

**Project:("D:\AWS\VPC Architecture with Private and Public Subnets with AutoScaling.png")**

1. Create a VPC with 4 subnets (2Private, 2Public) and 2 Availability Zones with NAT Gateway for each Availability Zone
2. Create Auto Scaling in Ec2 Service by creating the template, After creating the template use the template for creating Auto-Scaling and assign the Instances for both private subnets for Auto-Scaling

-Increase the desired size and Maximum size up to 4 if required to increase

-Instead of creating Load Balancer directly, Create the Load Balancer later

# Note Point’s

NACL and SG plays very crucial role for the Security Purpose . These are the last point of Security for a Application

\_\_\_\_

AUTO-SCALING is the process that

If no. of Requests are increased then the Auto-Scaling increases the instances(Servers) Automatically

\_\_\_\_

LOAD BALANCER helps to divide the reqs Equally to all servers & this load Balancers helps to send req to Server in VPC

\_\_\_\_

BASTION HOST OR JUMP SERVER

This bastion Host lies in Public Subnet and no one can reach the Server or EC2 instance in Private Subnet Directly but We can Access our Server or EC2 Instance through Bastion Host which lies in Public Subnet

\_\_\_\_

Ques:If a Application Need to be Highly Available and Scalable, What do u do for this?

Ans: For Availability ill deploy my applications in both Availability Zones

For Scalability I'll use Auto-Scaling for the Application

Ques:If a Application is Two Tier how will u design it?

Ans: Two tier Application is created by adding a Public and Private Subnet. Server lies in the Private Subnet.

Day :1

DAY:1

1.CLOUD==>Instead of Buying the Physical Server to Deploy the Application we can use Virtualisation and can use virtual Server with IP Addresses and can work accordingly with lesser Costs.

2.PRIVATE CLOUD===>If Team A uses Cloud with their Instance and that Team A are not allowed to give the instance to other organizations.

3.PUBLIC CLOUD===>Anyone with AWS OR AZZURE Account Irrespective of any organization can create instances Example : - Amazon, Microsoft, Google

->Can create a Virtual Private cloud in a Public Cloud

-----------------------------------------------------------------------

DAY:1

IAM: --

IAM Service is used to create IAM user or child User from Root user

IAM in AWS is a combination of Both Authentication & Authorization for child or IAM User

IAM helps to create & describe the user details, user permissions

(USERS + POLICIES + GROUPS) + Roles ==IAM

Users=User Details (Credential’s , Info of User)

Policies= What are the actions can user do

groups= Category of user (Helps to Set the Policies in smart way in single step) by creating the category of people only can-do particular actions

Roles = Giving Temporary Access to AWS Account with [Access Key + Secret Access key + Session Token]