AWS – Overview

what is IP address :-- IP means (Internet Protocol) it's a unique identification for a device in the Network .

what is host name :-- name of the server is called the host name.

WEB REQUEST AND RESPONSE FLOW

1. Client opens the Browser and tries to access **google.com** website,

Note :- Remember **"google.com"** is host name / domain name.

2. Browser required two things (domain name and IP address of domain) and without this information browser can't reach to domain

3. so DNS (Domain name server) which keeps the track of (domain name and IP address of domain).

4. The job of domain name server is it keep the track of all the host names and an IP address.

5. so whenever Client tries to access google.com website the first call goes to local DNS and this local DNS is present on your local machines.

6. and local DNS stores all information of (domain name and IP address of domain) in **"etc/host"** file on linux machine.

7. The job of local DNS is get IP address of required (domain name), if the local DNS has IP it will give to browser otherwise local DNS will give call to Root Name server (RNS) and the Job of Root Name server (RNS) is , it will identify the request is comming from which extension of domain **(.com, .edu, .in, .org)** after identifying request, **Root Name server** (RNS) gives call to **Top level Domain** (TLD)

8. **(.com, .edu, .in, .org)** are nothing but **Top level Domain** (TLD)

9. client is accessing **google.com** website so (.com) is TLD and TLD will give call to (NS) **Name server.**

10. now (NS) **Name server** will identify the requested Domain name and will give call to (SOA) **Start of Authority** and SOA will have IP address of requested Domain name and now SOA will give this IP address of requested Domain and now the IP address will get to LOCAL DNS and local DNS will give that IP address to Browser.

11. now using **(domain name & ip address)** browser will send request to google.com website in data packets.

12. and this data packet contains all information like **(request is coming from which IP address, which protocol)** etc.

13. and whenever client request reach to **google.com** website , there firewall will verify your request like [is request is coming with http **(80)** protocol or request is coming with https **(443)**] and it stops the un-authorized access.

14. then all allowed request from firewall goes to **Load Balancer of google.com** website and base on the load / traffic load balancer will distribute the traffic to multiple server using round robin method and gives resposne to client.

****

AWS - 3T'S Elasticity Scalability High Availability

What is elasticity :-- elasticity is also called as **horizontal scaling** and elasticity means flexiible and we are increasing and decreasing the number of EC2 instances based on the load that is called elasticity. elasticity will be for short term, as per the requirement, it will increase and decrease EC2 instance for specific period.

Note :-- in this case here we are increasing the number of servers we are not increasing the capacity of the server.

With help of Auto Scaling we can increase or decrease the EC2 instances and with the help of load balancer, we distributes the traffic to these EC2 instances.

What is auto scaling :-- Autos scaling meaning scale out and scal in. **scale out** meaning increasing the number of EC2 instances. And **scale in** meaning decreasing the number of EC2 instances.if you want to increase the more performance in the run time then go for autoscaling.

What is scalability :-- scalability is also called as **vertical scaling** and increasing the capacity of the server is called scalability. in other word we can say upgradtion of server.
for example, if your server with (8 GB RAM & 500 GB HDD) in future, this server configuration is not suitable to handle the workload,
so you increase the this server configuration means you increase the capacity of the server to (16 GB RAM & 1 TB HDD) is called scalability.
Scale up and Scale down is called scalability. **scale up** meaning what increasing the capacity of the server and **scale down** meaning what decreasing the capacity of the server

Note :-- scalability will be for long term , In real time **you can not increase the capacity of the server while server is running,** for that you have to take downtime.means you need to stop running server and then increase the capacity of the server.

scalability can be achieved by changing the instance type of EC2 machine in AWS.

What is Auto Scaling Group (ASG) :-- increasing and decreasing the number of EC2 instances based on the load that is called horizontal scaling or elasticity.
that increasing and decreasing the number of EC2 instances where does , that place called **Auto Scaling Group** (ASG) . in Auto Scaling Group (ASG) we set the rule , that what will be minimum and maximum EC2 instances count and base on that increasing and decreasing happens.

What is high availability :-- if the service is avilable for any time that is called high availability and some time if the service is not avilable that is called down time. high availability always measured in percentage (%). with help of load balancer we can make the service highly available using **round robin method.**

What is load balancer :-- load balancer distributes the request traffic to servers using round robin method. you create the load balancer in Regional level and load balancer has a power to send the traffic across availability zones. Whenever you create the Load balancer , this load balancer provides the **DNS URL to** acccess web application. In AWS load balancer called (ELB) Elastic load balancer. ELB distributes the traffic to the multiple EC2 instances across the Availability Zones. ElB is service not server.

What is failover :-- load balancer routes client requests across all servers. Load balancer will monitor the apllication is reachable or not by checking health of application and if the one application gets down or found unhealthy then load balancer will send request to another server , that is called failover.

What is Fault tolerance :-- If you don't want down time , then you go for auto scaling , and this is called Fault tolerance. and **Fault tolerance is nothing but zero downtime.**

What is redundancy :-- keeping the same application in multiple servers is called redundancy.

To achieve High availability what are the three things are important :--

We need **(RMF)** redundancy monitoring and failover,

Redundancy :-- keeping the same application on multiple servers is called redundancy.

Monitoring :-- load balancer monitors the health check of application and send request to server using round robin policy.

Failover :-- if one server goes down other servers is picking up that is called failover.

how do you achieve zero downtime :-- using Auto scaling

****

AWS - Regions and Availability Zones

AWS terminologies :-- AWS terminologies are Regions and Availability Zones. AWS has globale infrastructure.

AWs has region and region is a geographical area where AWS has its own data center or infrastructure and every region has Availability Zones. region will have only one data center or multiple data centers in different different Availability Zones. Availability Zone are nothing but data center or infrastructure on different different locations. regions and availability Zones are completely managed by AWS.

one availability zone is nothing but a group of data centers.

Same region base Availability zones are come interconnected with each other so, they can easily communicate each other.

We create the servers in Availability Zones. that servers called in AWS **"EC2 instances"**

Note :-- Availability zones founds inside the (VPC) Virtual Private Cloud. so whatever you **setup or create (EC2 instances /server)** under the VPC of each Availability zone.

VERY IMP :-- you can create **maximum 5 VPC** in one region.
And you can create **maximum 20 EC2** instances in on region.

What is the default region of AWS :-- The AWS default region is **North Virginia** because whenever AWS want to implement any new service first AWS Implement that in **North Virginia only** and slowly that new services will get implemented in all other regions.

two regions never communicate to each other by default but if required they can able to communicate each using VPC peering.

What is latency and types of latency :-- The time it takes for a server to respond to a call is called latency and there are two types of latencies

1 Low latency :-- means When you get a quick response from the server it is called low latency. (response time is quick means low latency)

2 High latency :-- means When the response from the server is delayed, it is called high latency. (response time is more means high latency)

****

AWS - Services EC2, Elastic Beanstalk, Light Sail, Lambda

What is EC2 :-- EC2 means Elastic Compute Cloud, a EC2 is a service from AWS where you can go and create VMS (Virtual machines).

we create **virtual machines** in EC2 service and in AWS **we called those (Virtual machines) as instances.**

Every service in AWS should be either **Regional or Global.**

Elastic Beanstalk :-- it is **PAAS** means platform as service, using Elastic Beanstalk service , it is very easy to deploy web applicaiton in AWS.
The backbone of **Beanstalk** is EC2 means whatever application you depoly on Elastic Beanstalk, so Elastic Beanstalk will launch EC2 instances automatically and deploy the provided web application on EC2 instance and gives the URL to you to access web application.

In General, in **platform as service** (PAAS), you don't have any control on servers but in AWS **(Elastic Beanstalk).**

you have full control on EC2 instances which are lanunched by Elastic Beanstalk. Elastic Beanstalk handles EC2 instances (operating system) behalf of you.

Elastic Beanstalk ask you what is your applicaiton platform like **(java, .net, python)** and we need to select required platform which is suitable for your application then Elastic Beanstalk will deploy the application automatically and take care of applicaiton.

Elastic Beanstalk gives supports to auto scaling option.

Mainly Elastic Beanstalk is used for small web application.

Light Sail :-- Lightsail provides pre-desinged templates like **(wordpress, joomla, drupal, cpanel, gitlab,redmine, etc)** to developer and developer just select required templates and does required configuration and starts to use.

The drawback of Light Sail is it does not support auto scaling.

Lambda :-- AWS Lambda is a serverless and lambda is regional.

A **serverless architecture** is a way to build and run applications and services without having to manage infrastructure.

We create functions in Lambda and that created function in lambda is called Lambda function.

you can write these funciton in many languages like **(java, python ,ruby, .net , etc).**

whatever you do in AWS like **(create,delete,update,stop,terminate)** any service then **event generates at the backend side in AWS and that events stores in Event bridge (Event bridges holds the all events of AWS services)** and base on that events, we can create the rule and execute the lambda function.

events meaning whatever we do in AWS every single click and event will be generated in the back end and Event bridge catch and hold those events.

Example :-- Suppose if you want to stop to others to launch or create EC2 instance. for that you will create the lambda function , which will automatically stop the EC2 instances which are created by others.

Flow :--

1. when others will create or lanuch instance , automatically events will be triggerd in AWS Event bridge.

2. we create the lambda funciton and writes the rule in Event bridge, if any other person create or launch EC2 instance then execute the lambda function to stop newly created EC2 instances which are created by others. So we can say lambda is invoked based on Event trigger.

Note :- Lambda is used for automation.

Real time example of LAMBDA :-- suppose there are 50 EC2 instances of DEVOPS team, and our requirement is all 50 EC2 instances should be stop automatically sharp 9PM and automatically start 9AM every day. so for that , we can create two lambda funciton,

1. Start_funciton :- we will create Start funciton in lambda to start EC2 instances sharp 9 AM.

2. Stop_funciton :- we will create Stop funciton in lambda to Stop EC2 instances sharp 9 PM.

then we will create schudular in Event bridge to execute **(Start_funciton and Stop_funciton)** lambda function.

Common Lambda application types and use cases

- **File processing –** Suppose you have a photo sharing application.
- **Data and analytics –** Suppose you are building an analytics application and storing raw data in a DynamoDB table.
- **Websites –** Suppose you are creating a website and you want to host the backend logic on Lambda.

COMMON NOTE :-- in AWS all the services will start with **simple word** end with **service word**.

Example :- SNS (Simple Notification Service), **SQS** (Simple Queue Service), **SES** (Simple Email Service), **S3** (Simple Storage Service)

****

AWS - Storage S3, EBS

S3 :- S3 is **Simple Storage Service**. and S3 has **unlimited storage** , by default S3 storage is **private**. S3 is **serverless**.

S3 is use to stores any **kind of the files**. you can not **execute any file in S3.** you can not install **(Operating system, database)** or any application in S3.

S3 is a **object based service**. and it is **object base storage**.

we can create the **bucket** in S3 and i**nside the bucket, we stores the objects.** so **bucket is a container of objects.** and **name of the object is called key.**

Note :-- Bucket is nothing but **folder** and **object** is nothing but **files**.

S3 is global and buckets are Regional.

S3 supports **static website hosting.** S3 provides as a static website hosting that meaning , you just create bucket and put all your static HTML files inside the bucket and just enable **"enable static website hosting"** option. then S3 Will give you URL to access static website.

EBS :-- EBS means **(Elastic Block Storage)** EBS is a centralized storage. EBS is nothing but volume **(HDD)**, which we attach to EC2 instance.

in AWS we called EBS volume.

Whevenr we Launch EC2 instance, automatically on**e volume that is called default volume will attach to EC2 instance.** this Default volume is nothing but root volume of EC2 instance. and on this Volume We install the Operating system, so the **volume which has operating system is called default/Root volume.**

Whenver you launch **Window based EC2 instance** , then default EBS volume size will be **30 GB** and

Whenver you launch **linux based EC2 instance** , then default EBS volume size will be **8 GB or 10 GB.**

If you need any extra volumes then you create it and then attach it to EC2 instance. **this volume is called additional volume.**

Note :- additional volumes can be created and attached so volumes can be attached and detached. you can attach multiple volumes to EC2 instance.

difference between S3 and EBS :-- S3 is object (file) base storage and EBS is block base storage , where you can install (Operating system, database) or any application.

NOTE :- there are two types of Operating system (Client OS & Server OS) and EC2 supports **only Server OS**

Client OS :-- Windows 10, 11
Server OS :-- Windows server 2021, redhat, linux, ubuntu , centOS, SUSE

EC2 instance has only **one root volume** and EC2 instance **can have multiple additional volumes.** and for each volume **maximum size is 16 terabyte (TB).** so maximum size of the EBS volume is 16 terabyte (TB).

one volume cannot be attached to multiple EC2 instances at the same time.

YOU CAN INCREASE THE VOLUME SIZE ON FLY , means **without stopping** EC2 instance you in increase the volume size of EC2 instance.

you can not **decrease the volumen size**. and if you want to decrease the volume, then you need to delete that volume and create new volume and attach it to EC2 instance.

EC2 instance and volume should be in the same availability Zone.

Attach volume means **mount the volume** and **Deattach** Volume means **unmount the volume.**

you cannot **detach the root volume while EC2 is running** but you can **detach the aditional volume while EC2 is running.**

you cannot delete the volume while it is attached , you need to do **detach first and then delete.**

FOR WINDOWS AND LINUX
root volume device name is :-- /dev/sda1
aditional volume device name is :-- /dev/sdb1 , here you can put any name but name should be start with (/dev/sd) +**<NAME>**
Example :-- (/dev/sdb1, /dev/sdb2,/dev/sdf3).

FOR UBUNTU

root volume device name is :-- /dev/xvda
aditional volume device name is :-- /dev/xvdb1 , here you can put any name but name should be start with (/dev/xvd) +**<NAME>**
Example :-- (/dev/xvdb, /dev/xvdc2,/dev/xvdf3).

****

AWS - EFS, Data Services

EFS means **(Elastic File System)** is used for share storage , so that share storage can be attached to all EC2 instances.

suppose if you want all EC2 instance need to have a common shared storage for that you can use (EFS) service.

EFS is **file based storage** , you can create (EFS) and just mount or attach (EFS) to EC2 system as shared storage. **EFS has unlimited storage.**

EFS can be **mounted to multiple EC2 instances across availability zones.**

when you create an EFS and mount the EFS to the EC2 instance and you go inside that mount point and put some files that files will be indirectly stored in EFS.

EFS is only for Linux EC2 instances. EFS works on NFSv4 **(NET WORK FILE SYSTEM VERSION 4)** protocol.

FSX is a share storage for Windows server and **FSX is for Window based EC2 instances.**

What is SNOW FAMILY IN AWS :-- There are three devices in the **SNOW FAMILY.**

snow family devices
1. snow cone :-- it can store 8 TB data
2. snow edge :-- it can store 100 TB data
3. snow mobile :- it can store data in PETA BYTES (PB)

snow family is physical data transfer device . As per your requirement, you need to put order on AWS console, and AWS will send you SNOW FAMILY device.

Practicle example :-- suppose , if you have 5 (TB) data, and you want put data on AWS ,then you will buy (snow cone device) from AWS and AWS will send you that device, you will take backup of your 5 (TB) data on this **(snow cone device)** and will return back it to AWS and AWS will copy all that data on their S3 Service.

Note :-- All SNOW FAMILY device data will store in S3 Service.

If you are are not using your data frequently and you want to store it somewhere with cheaper price , for that **AWS Provides Glacier service.**

in **Glacier**, we store all data in **(.zip) archived format.** it is **cheaper than S3** and all infrequently data will be STORED in Glacier service in (.zip) archived format and all **frequently data** will be STORED in S3.

Note :-- using storage gateway , you can **mount/ attach** (S3,EBS, Glacier, FSX) storage services to your EC2 instance.

RDMBS :-- Relational Database Management Service.

RDS is a service where you can (set up, configure , manage and maintain) all rdbms databases and RDS supports only RDBMS databases.

1. RDS has **6 Engine and We call them RDS DB instance.**
(MySql, Oracle, MS SQL, postgres, maria db, Arrora) are RDS DB instance.

Note :- AWS provides DMS (Database Migration Service) , and using this service, we can transfer our local database into AWS

NOSQL :--
1. NOSQL is used for Non-Relational Database system.
2. Using JSON data, unstructured data can be stored
3. The Schemas are dynamic
4. Schemas are non-regid, they are flexible
5. No interface to prepare complex queries
6. Here we call Collections and collections has documents
7. MongoDB, BigTable, Redis,RavenDB,Cassandra, Hbase, Neo4j and CouchDB are good example of NOSQL Databases.

AWS provides DynamoDB is no SQL service (no SQL means Not Only SQL)
NoSQL databases are non-relational databases (no fixed columns) and are distributed (horizontal scaling)
NoSQL databases do not support joins
NoSQL databases do not perform aggregations like (sum, min, max)
NoSQL databases scale horizontally
NoSQL databases do not have fixed SCHEMA.
You can't write complex query in NoSQL databases.

DATA WAREHOUSE :-- RED SHIFT is warehouse of AWS, where we can put large amount of data.

Elastic Cache :-- It is in **memory caching service.**

Cache :-- All frequentyly accessed data is stored at this place. and it will give you best performance.

Elastic Cache support two engine **(memcached & redis)**

****

AWS - VPC, Route53, Cloud Front

Route 53 :-- Route 53 is Global.
Loadbalancer will give you (DNS) url to access website, and this URL will be nasty means **(not user friendly , not meaningful)**
so to get (User friendly & meaning ful) URL, we use Route53 service.
Route 53 is a DNS service from AWS.

(**53** is port number of DNS), so this service is called **Route53.**
Route 53 contains records.

What is record :- record is nothing but routing to destination using (hosted zone/domain name).

Example :-- book.com is your **(hosted zone/domain name)** and whenever user will hit this url, then internally book.com will give call to ELB **(Elastic Load balancer ->> Application Load balancer)** and will fetch your web application.

VPC :-- (VPC) means Virtual Private Cloud and whatever resources that we create in AWS everything should be within the VPC.

VPC is regional and maximum 5 (VPC) we can create per region. VPC is like a virtual data center on the cloud.

HIGH INTERNET SECURITY :-- normally , you always take internet connection from Internet provider and internet provider provides you shared internet, and it is secured but many sectors like **(NASA, ARMY)** want more secured internet connection, so AWS provide **"direct internet"** and **direct internet is not shared internet**, it is super fast very securied **internet connection.**

CloudFront :-- Amazon **CloudFront** is a **(CDN) content delivery network** operated by Amazon Web Services. The content delivery network was created to provide a **globally-distributed network** of proxy servers to cache content, such as web videos or other bulky media, more locally to consumers, to improve access speed for downloading the content.

CloudFront has Edge loation and Every region have edge location and Edge location will cache the application. in **cloudFront** you **create "Distribution"**, so we just need to go in cloudFront service, and need to **to create "Distribution".**

Cloud front have TTL **(Time to leave)** for cache , means if you set (TTL) for 12 hours, then data will be cache for 12 hours. and after 12 hours, data will be updated in cache.

Example :-- suppose , You advertise on a website that an iPhone is sold for ten thousand rupees so that information **will be cached in Edge**

location for next 12 hours (because you have set TTL for 12 hours). before this 12 hours , **if change the price of iphone , that updated price will never reflect immediately on website page it will reflect after 12 hours.** but if you want to **immediately reflect this updated price** of iphone on website page, then you just go in **CloudFront** and do **"invalidate valid cache" so all previous cache will be removed and new information will be cached for next 12 hours.** now at this time user will able to see updated price of iphone.

Note :- CloudFront cache the static and dynamic data. Cloud front has edge locations and edge locations are connected with CDN.

When a user requests content that you're serving with CloudFront, the request is routed to the edge location that provides the lowest latency (time delay), so that content is delivered with the best possible performance.

- 1) If the content is already in the edge location with the lowest latency, CloudFront delivers it immediately.
- 2) If the content is not in that edge location, CloudFront retrieves it from an origin that you've defined—such as an Amazon S3 bucket, a MediaPackage channel, or an HTTP server (for example, a web server) that you have identified as the source for the definitive version of your content.

****

AWS - IAM, Cloud Watch, Cloud Trail

IAM :- (IAM) means **Identity and Access Management.** IAM is **Global service and IAM service totally free.**

IAM allows you to **manage users and their level of access** to the aws console.

It is used to set **users, permissions and roles.** It allows you to grant access to the different services of the aws.

There are **two types of Accounts / Users** in IAM

1. Root Account/ User :- Root user is a Admin user and This Admin user have full access of AWS services and this Admin user can create IAM user and gives the permision / policies to IAM user to access required service. (permission are nothing but policies)

2. IAM Account/ User :- IAM users are created by Admin User and they have limited access of AWS services. for IAM users Admin/Root user can attach and detach permissions or policies.

Note :- If you are login with (email address) means you are Root user and if you are login with (user name) means you are IAM user.
Login url will be different for root user and different for IAM user.

What is Organization in AWS :-- (AWS has ORGANIZATION service)

suppose you have root account (Management Account) and under this root account, there are lot of departments like **(HR, ADMIN, BILLING, DEVELOPER).** so it is not possible to handle so many department with single root account.

Note :- using (SCP) **Service Control Policy** , you can gives required AWS service access permission to each Department.

For example, for (ADMIN) department, we can give all AWS service access permission, for (DEVELOPER) department, we can give only ECS service access permission.

so using Organizaion service of AWS we can create the root account for each department and each ROOT user of Department will create IAM user under his department.

So there **will be Two accounts**

1. **Management Account :-** means main root account
2. **Department/Member Account :--** Member accounts are nothing but all department wise created Root user.

CloudWatch :-- cloud watch is used to monitor AWS resources. cloud watch will monitor all AWS services.
CloudWatch service is reginal service and used for monitoring purpose, **Cloud watch** is used to monitor performance of AWS resources.

(Alarams, Events & Logs) are important three things in Cloudwatch.

We create the Alaram in Cloud watch and in alarm, we set the metrix
Example :-If CPU utilization goes more than 90 % then notify to Admin using SNS service.

There are **two types of monitoring** in cloudwatch
1. **basic monitoring :-** It is free, and will monitor every 5 minutes and generate log.

2. **detailed monitoring :-** It is not free, and will monitor every 1 minute and generate log.

Cloud Trail :-- whatever is happening in the AWS environment will getting recorded or tracked by cloud trail service. and it is used for security purpose also it is used for tracking , auditing purpose.

Coning :-- Coning is separate service which is used to **monitor the changes of the AWS resources.**

Aws inspector :-- Aws inspector which is used for **security purpose**, it monitor all illegal activites like **(who is hacking, who is fishing).**

Trusted Advisors :-- Trusted Advisors is AWS service, which gives advices like **(How we can save the money while using AWS resources).**

AWS support :-- AWS provides AWS support , (AWS support) is nothing but customer support, there are four types of support.

(1.basic support ,2.developer support, 3.business support, 4.Enterprise support) and **Basic support is free.**

****

AWS - IAM Service (PRACTICAL)

Note :-- AWS **Multi-Factor Authentication** (MFA) is an AWS Identity and Access Management (IAM) best practice that requires a second authentication factor in addition to user name and password sign-in credentials. MFA is highly recommended for root and IAM user.

There are two ways to Access AWS Account

1. console access :-- console means using AWS GUI (Graphical User Interface). and using (email and password) or (user name and password) we login to console.
means Root & IAM user both can login to console.

2. programmatical access :-- Through CLI (Command line interface) You can access AWS programmatically. programmatical access means, using AWS (SDK) software development kit, we can write the program using (JAVA, PYTHON). we Enables access key and secret key and use to login for programmatical access.

if we lost (access key and secret key) ,then you can regenerate it.

Note :-- Every Root & IAM user will have their own **(access key and secret key).**

****

AWS - IAM Groups IAM Roles

IAM Groups :-- collection of IAM users is called a **IAM Groups**. Instead of going and giving individual permissions (policies) to each and every IAM user, we can group them in "IAM Groups" and set the permission to **"IAM Groups"** , so automatically every IAM user will get **permissions (policies).**

We can attach policies to each and every I**AM user or to "IAM Groups".**

Note :- one user can be in multiple groups. we can attach multiple policies to a IAM user and IAM group, **Maximum 10 policies we can attach to a IAM user and IAM group.**

You can attach and detach the policies to a IAM user and IAM group.
IAM Group are used to assign policies to the bunch of IAM users at the same time.

newly created IAM user will not have any permissions /policies, we need to attach permissions /policies to newly created IAM user or we need to add **(newly created IAM user)** in IAM Group.

permission are nothing but policies and this policies contain Policy Document and policy document contains permission, so we can say policies contain permission and policies are nothing but permission. **in AWS we call permission as policies and all polices are written in JSON format.**

There are two types of policies

1. managed policy :-- managed policy are predefined policies in AWS, that are created and managed by AWS.

2. inline policy :-- As per our requirement we create our own policies and those policies are called inline policies
so Root User / IAM user can create inline policies. inline policy is also called customer managed policy.

IAM Roles :-- IAM Roles are temporary access of AWS resources. with IAM Roles we don't need to login to AWS to access AWS resources.
with IAM Roles AWS provides temporary access of AWS resources. so IAM role is very similar to a IAM user.

Example :-- suppose , if you want to give temporary access of EC2 service to external user, then

1. first you will create the role
2. then you will attach EC2 service access related permission/policies to this role then finally
3. you will attach that role to EC2 service and will give that details to external user.

so external user will get temporary access of EC2 service .

Note :-
1. you can attach maximum 10 permission/policies to this role.
2. AWS Resources can have only one ROLE.
3. one ROLE can attach to multile AWS Resources.

An IAM role is an IAM identity that you can create in your account that has specific permissions. IAM roles define the set of permissions to access AWS service.

REAL TIME EXAMPLE :-- Suppose, there is one person "vishal", and he wants to access my AWS account and have to use S3 service, but Vishal is not a employeee of my company, he is from third party resource. but Vishal has it's own AWS account and vishal has also have his AWS account id.

for that reason, I will do the following steps to give Vishal access to my AWS account's S3 service

1. I will create the role as "VISHAL_S3_ROLE"
2. I will assign (S3 access) permission/policy to this **"VISHAL_S3_ROLE"** and will attach vishal'S AWS account id.

Note :- while creating the IAM ROLE , it will ask to whom you want to give the permissions then you will set or attach vishal'S AWS account id.

3. after that , IAM ROLE will generate a URL.
4. so i can give this URL to Vishal to access my AWS account's S3 service for temorary purpose

(By default, this url will be live for 1 hour, after 1 hour it will expired). you can increase it upto 12 hours.

IDENTITY PROVIDER OR FEDERATION :-- Identity federation allows users outside of the AWS environment to access AWS resources without the need for creating individual IAM users. For example, just take example of SSO, single sign on.

suppose if you don't want to create IAM user instead of that you want your external user should login through (Facebook, gmail, ldap, OKTA) and can access AWS resource. that time Identity federation is useful.
To use Identity provider, you can create an IAM Identity provider entity to create relationship between AWS and your Identity provider. Identity provider supports to (OIDC) OpenId Connect & SAML2.0 (Security Assertion Markup Language).

identity providers can be used when you want to give access to your users who are outside of your AWS account.

PRACTICLE Example URL:--

1.https://www.youtube.com/watch?v=ltizXF-3Zrw&list=PLEr0Q0hLeaQcfrxLJmMDQs7ccGohI-xl1&index=26

2. https://www.youtube.com/watch?v=gxqP9L40Sko

****

AWS - IAM TAGS, Trusted Access Advisor, Inspector

IAM TAGS :- IAM TAGS are key value pair and IAM TAGS is used for identification and automation purpose. in AWS , whatever service you create , you can set the tag to this service. you can maximum 50 tags to each AWS Resource. tags are important but tags are optional.

REAL LIVE EXAMPLE of AUTOMATION with the help of TAG :--

Suppose, we have 100 EC2 instances and we set tag name to each EC2 instaces as "Devops". now i want to stop and start this EC2 instances every day.

every day 9 PM stop EC2 instances and start 9 AM every day. for that you can write the lamdba function and can add the logic to stop and start EC2 instances which tag name is **"Devops".**

Trusted Access Advisor:-- Trusted Access Advisor is a AWS service and With the help of "Trusted Access Advisor" , we can see the all user's usages of AWS Resources and when they did last logins.
means using **Trusted Access Advisor** , we can keep the eye on User's activity.

REAL TIME EXAMPLE :-- Suppose "XYZ" is a user, and I want to know when "XYZ" last logged in and what he did in the last 7 days, so I can use trusted access advisor and get "XYZ" user activity details **(Excel) format.**

Access Analyzer :-- Access Analyzer is an AWS service and with the help of "Access Analyzer" we can collect information about which user has which permissions to access which AWS resources.

ALIAS NAME :-- whenever you create an account in AWS so every account will have 12 digit AWS Account ID and it is difficult to remember this 12 digit Account Id, so we can give the **ALIAS NAME** to this Account ID.

****

AWS - IAM Practicals and AWS sign in Console

How to Create IAM user.

1. you do AWS Console login with Root account
2. then in search box just search "IAM" service
3. then you will get IAM dashboard
4. at left hand side, you will get "Identity and Access Management (IAM)" menu
5. then from "Access management" just select and click on "Users"

option.
6. then just click on "Add users" button to create or add new user and provide below user information
   a) User name :-- MVR
   b) select "Provide user access to the AWS Management Console"
   c) then just select "I want to create an IAM user" option
   d) then set Console password by selecting **"Custom password"** option
   e) then just click on "next" button.
   f) then just select "Attach policies directly" option from **"Permission options".**
   g) then select "AdministratorAccess" policy
   h) then just click on "next" button.
   i) then just review and then add tag (Team :-- DevOps)
   j) then just click on "Create user" button.
   k) then you will get "Reterive password" dashboard and from here
   l) you just download (.csv) file , (.csv) file have login details and login url content.

==========

How to Create user Group.

1. you do AWS Console login with Root account
2. then in search box just search "IAM" service
3. then you will get IAM dashboard
4. at left hand side, you will get "Identity and Access Management (IAM)" menu
5. then from "Access management" just select and click on "Users Group" option.

6. then just click on "Create Group" button to create new Group user and provide below Group information
   a) Name of the Group :-- **DEVOPS**
   b) select listed users from list and Add/ attach or assign this selected user(s) to this "DEVOPS" group.
   c) Attach permisson from policies list :-- (Example :-- select AmozonRouth53FullAccess)
d) then just click on **"Create Group"** button.

==========

How to reset "User" Password

1. you do AWS Console login with Root account
2. then in search box just search "IAM" service
3. then you will get IAM dashboard
4. at left hand side, you will get "Identity and Access Management (IAM)" menu
5. then from "Access management" just select and click on "Users" option.
6. then form user list , just select user row and click
7. then select "Security credentials" tab
8. then just go in "Console sign-in" option and click on **"Manage console access"** button
9. then you can create new password by selecting **"Autogenerated password"** option or
10. by providing custom password
11 after re-setting new password just click on "Apply" button.

==========

How to disable "User" access.

1. you do AWS Console login with Root account
2. then in search box just search "IAM" service
3. then you will get IAM dashboard
4. at left hand side, you will get "Identity and Access Management (IAM)" menu
5. then from "Access management" just select and click on "Users" option.
6. then form user list , just select user row and click
7. then select "Security credentials" tab
8. then just go in "Console sign-in" option and click on **"Manage console access"** button
9. then select "Disable" option from Console acces and just click on **"Apply"** button.

==========

How to Configure (MFA) Multi-factor authentication for IAM USER login

Note :-- for that you need to download and install "Google Authenticator" mobile app on your mobile and then

1. you do AWS Console login with Root account
2. then in search box just search "IAM" service
3. then you will get IAM dashboard
4. at left hand side, you will get "Identity and Access Management (IAM)" menu
5. then from "Access management" just select and click on "Users" option.
6. then form user list , just select user row and click
7. then select "Security credentials" tab
8. then just go to Multi-factor authentication (MFA) tab and just click on "Assign MFA device" and configure below properties
   a) Specify MFA device name :-- mayur-device
   b) select MFA device :- select Authenticator app
   c) then just click on "Next" button.
   d) then open "Google Authenticator" mobile app from your mobile
   e) then just click on "show QR code"
   f) then from your "Google Authenticator" mobile app scan QR code then you will get MFA code (get it twice)
   g) and provide that MFA code in AWS (MFA) Multi-factor authentication window
   h) then just click on **"Add MFA"** button.

==========

How to give programattic access to user ?

1. you do AWS Console login with Root account
2. then in search box just search "IAM" service
3. then you will get IAM dashboard
4. at left hand side, you will get "Identity and Access Management (IAM)" menu
5. then from "Access management" just select and click on "Users" option.
6. then form user list , just select user row and click
7. then select "Security credentials" tab
8. then just go to "Access keys" tab and click on "create access key" button.
9. then select "Command Line interface (CLI)" option from Access key best practices & alternative option
10. then just click on "Next" button and finally click on "Create Access key" button.

Note :-- you can can disable the programmatical axis you can disable the programmatical access by deactivating keys.

****

AWS - IAM Roles and Policies

There are two types of policies

1. managed policy :-- managed policy are predefined policies in AWS, that are created and managed by AWS.

2. inline policy :-- As per our requirement we create our own policies and those policies are called inline policies
so Root User / IAM user can create inline policies. inline policy is also called customer managed policy.

How to create Inline policies.

REAL TIME EXAMPLE :-- Suppose I want to give IAM access to user, but i don't want to give full access to user
only selected access like (create IAM User, create Group) to this user. for that i will create "policy" and will assign selected IAM permission/policies to this policy.

1. you do AWS Console login with Root account
2. then in search box just search "IAM" service
3. then you will get IAM dashboard
4. at left hand side, you will get "Identity and Access Management (IAM)" menu

5. then from "Access management" just select and click on "Policies" option.
6. then just click on "create policy" button.
7. then you will get "Specify permissions" dashboard
8. then from "Policy editor" tab just search "IAM" policy from "Select a Service" List and just click on selected Policy.
9. then just select required "Access level" option like **(List,Read,Write, Permission Management, Tagging)**

   a) select List "Access level" and select (ListUsers, ListGroups) option.

   b) select Read "Access level" and select (GetUser, GetGroup) option

   c) select Write "Access level" and select **(CreateUser,UpdateUser,DeleteUser, CreateGroup,UpdateGroup,DeleteGroup)** option.

10. then click on "Resource" option and select "Specify resource ARNs for this actions" , here

a) if you want to bind this policy to all User and Group then select "All" option or
b) select "Specific " option to attach this policy to (Group, user).

Inside the "Specific" option , If you want to set this policy to User then select "User" option from "Resource" group and just click on "Add Arn" link and configure below properties.

a) **Resource** :-- select (This account) option
b) **ARN** :-- user/mayur (add User name in (ARN) option)
c) after adding ARN you will able to see added ARN like (arn:aws:iam:::user/mayur)
d) then just click on **"Add ARNs"** button.

Inside the "Specific" option , If you want to set this policy to Group then select "Group" option from "Resource" group and just click on "Add Arn" link and configure below properties.

a) **Resource** :-- select (This account) option
b) **ARN** :-- user/devops (add group name in (ARN) option)
c) after adding ARN you will able to see added ARN like (arn:aws:iam:::user/devops)
d) then just click on **"Add ARNs"** button.

11. then just click on **"Next"** button

==========

How to create Role for EC2

1. you do AWS Console login with Root account
2. then in search box just search "IAM" service
3. then you will get IAM dashboard
4. at left hand side, you will get "Identity and Access Management (IAM)" menu
5. then from "Access management" just select and click on "Roles" option.
6. then just click on "create Role" button.
7. then you will able see (Select trusted entity) dashboard
8. then select "AWS service" as trusted entity
9. then select "EC2" option from "Common use cases"
10. then just click on "Next" button
11. then select "permissions" from "permissions policies" list
    a) AmazonS3ReadOnlyAccess
12. then just click on "Next" button
13. then set Role details
    a) Role Name :--6PMROLE

b) Description :-- 6PMROLE

13.then just click on "Create Role Button" button.

==========

I have the ec2 instances, and I want to stop all this E2 instance automatically, so how i can do this.

1. you do AWS Console login with Root account
2. then in search box just search "IAM" service
3. then you will get IAM dashboard
4. at left hand side, you will get "Identity and Access Management (IAM)" menu
5. then from "Access management" just select and click on "Roles" option.
6. then just click on "create Role" button.
7. then you will able see (Select trusted entity) dashboard
8. then select "AWS service" as trusted entity
9. then select "Lambda" option from "Common use cases"
10. then just click on "Next" button
11. then select "permissions" from "permissions policies" list
    a) AmazonEC2FullAccess
12. then just click on "Next" button
13. then set Role details
    a) Role Name :--AUTOMATIC-6PM-EC2-STOP-ROLE
    b) Description :-- AUTOMATIC-6PM-EC2-STOP-ROLE

14. then just click on "Create Role Button" button.

==========

(REQUIREMENT) :--suppose , if you want to give temporary access of EC2 service to external user.

REAL TIME EXAMPLE :-- Suppose, there is one person "vishal", and he wants to access my AWS account and have to use S3 service, but Vishal is not a employeee of
my company, he is from third party resource. but Vishal has it's own AWS account and vishal has also have his AWS account id.

for that reason, I will do the following steps to give Vishal access to my AWS account's S3 service

1. you do AWS Console login with Root account
2. then in search box just search "IAM" service
3. then you will get IAM dashboard
4. at left hand side, you will get "Identity and Access Management (IAM)" menu

5. then from "Access management" just select and click on "Roles" option.
6. then just click on "create Role" button.
7. then you will able see (Select trusted entity) dashboard
8. then select "AWS account" as trusted entity
9. then select "Another AWS account" option and provide (Vishal's AWS account ID)
10. then just click on "Next" button
11. then select "permissions" from "permissions policies" list
    a) AmazonS3ReadOnlyAccess
12. then just click on "Next" button
13. then set Role details
    a) Role Name :--VISHAL_S3_ROLE
    b) Description :-- VISHAL_S3_ROLE

14. then just click on "Create Role Button" button.
15. then just go IAM dashboard and then from "Access management" just select and click on "Roles" option.
16. just search newly created "VISHAL_S3_ROLE" role and just click and open it
17. in summary option :-- you will see (Link to Switch Roles in Console) option, and there you will find URL.
18. just copy this URL and gives to Vishal to access my AWS account's S3 service for temorary purpose
    (By default, this url will be live for 1 hour, after 1 hour it will expired). you can increase it upto 12 hours.

****

AWS - EC2 Service, EC2 Families/Instance Type

EC2 Service :-- EC2 means (Amazon Elastic Compute Cloud) and Ec2 is a web service from AWS that provides resizable compute services from in the cloud.
Ec2 is regional service. (**EC2 is pay as you use module).**

AWS Pricing model :-- for first year , each month 750 hours are free.

On demand instances Pricing model :-- on demand meaning whenever the customer Required EC2 instance that time customer will launch EC2 instance and will use it for certain period and will terminate and will pay as per his usages. On demand instances will have fixed price and this fixed price will be for hourly bases. and conecpt of On demand instances is "Pay for what you have used" and "pay for hour" , in On demand instances model, customer don't give any commitment to AWS that he will use AWS resources for how many days,month or years. also don't give any advance charges **(upfront changes)** to AWS.

Reserve Instances Pricing model :-- There are 3 types of Reserve Instances.

a) Standard Reserved Instance :- in AWS.In the Standard instance pricing model, customers commit to using AWS EC2 instances for 1 or 3 years (long-term commitment) and contract with AWS and pay AWS upfront payment means (advance charges , down payment, full or partial payment) and get a very good discount. AWS gives 75 % discount to customer on their hourly price. means (suppose If you're paying let's say 10 Rupees per hour, so you get 75% discount and you just need to pay 2.5 RS).

b) Convertable Reserved Instance :- here you can change your hardware configuration any time. means **(In Convertable Reserved Instance , suppose your EC2 machine have 4 GB ram, but after some days you came to know that this 4GB ram is insufficent , so you can increase the ram as per requirement [8GB, 16GB]. whatever you want.)** so Hardware upgradation is possible in Convertable Reserved Instance model.

c) Scheduled Reserved Instance :- if you are planning to hold EC2 instance for short period like (for a day, for a week or for month) then you must go for Scheduled Reserved Instance model.

Note :- If you are planning to use AWS EC2 instances for long period like (1 year or 3 years) , then Reserve Instances Pricing model is best choice, for you, to get good discount and to save money.

Spot Instances model :-- Spot means (bidding / auctioning). so in Spot Instances model, many people started bidding to access AWS EC2 instances.

Example :-- suppose there is Bidding of one (I-CORE 5 , 160GB RAM, 50 TB HDD) High configuration machine, and many people partcipate in bidding and so one people is ready to pay 10 dolloar for this machine and another is ready to pay 20 dollar for this machine, so for highest bidding price will consider, and another person will get this machine.

Note :-- if you want to get a huge capacity machine for cheaper price then go for spot Instances model.

Very imp :-- Spot Instances can be used for temporary purposes. you will get almost 90% discount it for Spot Instances model.

Dedicated Host machine model :-- If customer does not want the E2 instance to be shared in the host machine, so customer will go for Dedicated Host machine model. means if you need a physical machine with Virtual Machine then go for Dedicated Host machine model.

Saving plan model :-- it is same as Standard Reserved Instance but have different strategy

==========

EC2 Families/Instance Type :- Instance type is comibination of **(CPU+Memory),** There are **4 types of Instance Types.**

1. General Instance Type :-- for all general purpose you can go with **"General Instance Type"**

2. Memory Instance Type :-- If your machine want more memory then you go for **"Memory instance Type"**

3. CPU instance Type :- If your machine want more CPU then you go for **"CPU instance Type"**

4. Storeage instance Type :-- If your machine want more Storeage then you go for **"Storeage instanceType"**

4. GPU instance Type :-- This advance machine and If your machine want more Graphic for gaming purpose then you go for **" GPU instanceType"**

Some Instance type :-- Instance type is comibination of (CPU+Memory)

1. T2.Nano (0.5GB Ram + 1 Virtual CPU)
2. T2.Micro (1GB Ram + 2 Virtual CPU) :-It is free tier machine.
3. T2.Small (2GB Ram + 2 Virtual CPU)
4. T2.Medium (4GB Ram + 2 Virtual CPU)
5. T2.Large (8GB Ram + 4 Virtual CPU)
6. T2.Xlarge (16GB Ram + 8 Virtual CPU)

==========

What is Brustable performance Instances :-- It is not free service, it is billable, in Brustable performance Instances , AWS will give you some CPU on credit bases on runtime. so there will be no down time and it will give high performance for limited period of time.

Note :- CPU credits is depend on instance type. only **(T2 and T3 Instance types)** support for Brustable performance Instances.

****

AWS - EC2 Volumes, EBS Volumes, Instance Store Volume, EBS Central Storage.

EC2 Volumes :-- volumes are nothing but hard disk. and they are classified in two types

1. EBS volume :-- It is persistent storage or permanent storage. EBS is block base storage , where you can install (Operating system, database) or any application. If you stop and start the EC2 instance data

will never lost from EBS. EBS is billable and EBS has different types of volume.

EBS have below volume types.

a. General purpose (GP2, & GP3) :-- general purpose have SSD storage , SSD means (Solid State Disc) it is used for General purpose. GP3 has higher performance than GP2

b. Provisioned purpose (IOPS1, & IOPS2) :-- IO means (Input Output per second) and it is used for High performance. If you need high performance for (Database, Graphic) then go with Provisioned purpose .

c. Throughput (st1) :-- It comes with HDD support, and used for frequently access data with cheaper price. you don't need performance and have to use for regular purpose with cheaper price then select Throughput (st1).

d. Cold (sc1) :-- It comes with HDD support, and used for not frequently access data with cheaper price.

e. Magentic (standard) :-- It comes with HDD support and is previous generation.

Note :-

1. GP2 is a default volume type. and GP2 has default IOPS (Input Output per second) with 1:3 ratio means -> 1GB will have 3 IOPS

2. IOPS1, IOPS2 and GP3 are IOPS (Input Output per second) base configurable.

3. Root volume supports (GP2,GP3,IOPS1, IOPS2 and Standard) volume.

4. Root volume does not support for (ST1,SC1) volume, because (ST1,SC1) volume are not good in performace.

5. Additional volume supports ALL Types :-- if you need if you need any extra volumes what you need do you create it and then attach it to EC2 instance. this volume is called additional volume.

Note :- additional volumes can be created and attached so volumes can be attached and detached so volumes can be attached and volumes can be detached.

2. Instance Store volume (ISV) :-- Instance Store type are also called Emphemeral Storage. Instance Store type are temporary storage. If you stop and start the EC2 instance data will never lost from Instance Store type and it is free volume.

Note :-- By Default, If you terminate the EC2 instance, then all Root Volumes will be automatically deleted.
There is a option called "delete on termination" , if you check/select it then Root Volumes will be automatically deleted when Ec2 instance will terminate .

Note :-- If you terminate the EC2 instance ,then additional volume will not be deleted. because "delete on termination" is not selected.

REAL TIME EXAMPLE :-- Whenver you launch EC2 instance,but how AWS will come to know that EC2 instance is properly launched or not ?
There are two types of status to check the EC2 instance status.

1. Instance status check :-- It will check your EC2 instance (your Virtual machine is properly lanunch or not)

2. System status check :-- It Will check IP address is assigned or not to your EC2 instance, network is proper or not.
Note :-- if above Two status check are passed, then you will able to login to your EC2 instance.

if status check gets failed then your stop and start your EC2 instance.

****

AWS - EC2 Instance, EBS Snapshot Standard & Archive Tier, FSR (Fast Snapshot Restore)

Snapshot :- snapshot meaning backup, means backup of (Root Volume or additional volume) is called snapshot.

Note :-- Remember AWS is always take backup of incremental data, so snapshot are incremental backup

REAL TIME EXAMPLE :--

What is incremental backup means, for example suppose if you have 50GB Storage (SSD) drive and SSD have 10GB Data.

now suppose if you have taken a backup of 10 gb on 1-march-2024 after 2 months, SSD have additional new 10GB Data, so total SSD have 20GB Data. (previous 10GB + additional new 10GB Data).

now if you taken a backup of additional new 10GB Data on 1-june-2024 , then AWS Will never take a backup of all 20GB data,
it will only take a backup of additional new 10GB Data and this process called incremental backup.

in other word we can say snapshot is **point in time copy** , means when you're taking the backup at that time whatever the data
you have only that data will be backed up that is called point in time.

IMPLEMENTATION :-- snapshot meaning backup, means backup of (Root Volume or additional volume) is called snapshot and we can restore the snapshot to volume you can take snapshot from availability zone of one region to another availability zone of another region.
we store the snapshot in S3 and snapshot are regional. by default snapshot are private , but as per requirement we can make it public.

SOME IMPORTANT POINT OF SNAPSHOT

1. Snapshot is a point in time copy of the volume
2. Backup of the Volume is called snapshot
3. EBS snapshots are created from EBS volume
4. You can create snapshots from Volumes
5. **EBS Volume---> EBS snapshots--> EBS Volumes**
6. You cannot attach a snapshot directly to the EC2 instance, you have to create a volume using this snapshot and attach it to the EC2 instance.
7. It is not possible to login to the snapshot directly.
8. snapshots are stored in S3 (providers S3)
9. snapshots are visible from the EC2 console.
10. snapshots doesnt have availability zone.
11. Snapshots are Regional
12. By default, snapshots are private, if required we can make it public.
13. You can copy the snapshot from one region to another reginon in the same account
14. Snapshots can be shared from one AWS account to another AWS acount (private).
15. EBS volumes cannot be moved directly to any availability zone, instead create snapshot.
16. EBS Volumes are created from snapshots
17. Instance store volumes are created from a template stored into S3
18. To Create a snapshot , we no need to STOP the EC2 instance.
19. Data Life Cycle Manager :-- **Data Life Cycle Manager** is service , and using this service we can take snapshot automatically by creating schedule.

EBS Snapshot have two types

1. Standard tier :-- our current taken snapshot are called standard tier.

2. Archive tier :-- we can move the old snapshot to Archive tier in (compress/ archive) format, it will save money , because it cheaper.

Note :--

1. whichever snapshot is not necessary and whichever snapshot is not used you that snapshot you can put it into archive mode.
2. Move the snapshot to an archive tier is 75% cheaper.

Noe :- if you want create volume from archied snapshot, then you need first move/transfer archied snapshot to standard snapshot
then you will able to create volume from standard snapshot.

Note :-- KMS Key Management Service is used to store all encryption keys.

By default snapshot are not encrypted , and encrypted snapshot can not be shared.

Recycle Bin :-- All deleted snapshots stores in Recyle Bin. deleted any snapshot it will go and sit in recycle bin but in recycle bin also how many days will it sit (you mention this period example 7 days, 10 days, 1 year etc) that is called retention period.

FSR (Fast Snapshot Restore) :-- suppose, you have a snapshot of 100 GB from that snapshot we need to create a volume , so it will
take lot of time, becuase snapshot size to big. for that we can use FSR (Fast Snapshot Restore) service is used to take a big
snapshot quick and fast also we can create volume using this kind of big snapshot quick and fast.
FSR (Fast Snapshot Restore) is not free, it is paid service.

AMI IMAGES :--- Take image of pre-configured machine and use this this image to configure another machine that is call image and this image is called in AWS as **(AMI) Amazon Mahine Image.** AMI is copy of the entire EC2 instance Including volume. In simple word, AMI is duplicate copy of EC2 machine in image format.

Note :- all AMI stored in S3 only.

REAL TIME EXAMPLE :-- just assume I am your manager for now and I say that there are 10 people are joining in our project, so do setup of 10 machine. So it is not possible to configure 10 mahine within short time, so what we can do , we will configure single machine (by installing OS,required software, also custom application) and will take image of this machine and will install this image on another 9 machine. that is call image and this image is called in AWS as (AMI) Amazon Machine Image.

****

AWS - EC2 Instance Key-Pair, Cluster Networking Instance

KEY-PAIR :-- combination of public and private Keys is called Key-Pair.

AWS will carry the public key and customer will carry the private key.
cust will get (.pem) file and it is private file.

Key-Pair used to retrieve the password of the E2 instance. we don't have any Key-Pair by default ,
we need to create Key-Pair and attach it EC2 instance and created (Key-Pair) file have (.pem) extension.
one Key-Pair can be attached to multiple E2 instances. (We don't need to create separate Key-Pair for each EC2 instance).
but it is not good practice ,you need to create each Key-Pair for each EC2 instance.

KEY-PAIR REAL TIME EXAMPLE :-- using KEY-PAIR , we get the password to login (Window base EC2 instance or LInux base EC2 instance)

The default usename to Login Window base EC2 instance :-- adminstrator
and using KEY-PAIR , we get the password to login (Window base EC2 instance)

RDP (Remote Desktop protocol) is used to connect Window base EC2 instance and the port is 3389
We use RDP tool to login on Window base EC2 instance.

The default usename to Login LInux base EC2 instance :-- ec2-user
and using KEY-PAIR , we get the password to login (LInux base EC2 instance)

SSH is used to connect LInux base EC2 instance and the port is 22, We use putty tool to login on LInux base EC2 instance.
putty doesn't suppor (.pem) file, it support (.ppk) file
We use **puttgen tool to convert (.pem) file to (.ppk) file.**

Cluster Networking Instance :-- Group of (EC2 instances)/ servers is called Cluster and this Group called placement group.
if you want best performance then you make a group of EC2 instances and that is called placement group.

There are 3 types of placement group.

1. Cluster placement group :-- For better performance Cluster placement group means Grouping the EC2 instances in same rack, same availability zone

2. spread placement group :--in spread placement group Ec2 instances are spread across EC2 instances . and it is used for critical applications and for high performance.

3. partition placement group :-- in partition placement group Ec2 instances are spread across EC2 instances . and it is used for critical applications and for high performance. in partition placement group we will have same configuration EC2 machine. and in each partition , we can **keep 100 EC2 machine.**

****

AWS - EC2 Security Groups, NACL

Security Groups are used to stop un authorized access to AWS EC2 instances. in Security Groups, we create inbound and outbound rules. Security Groups acts like a firewall to the EC2 instances

Security Groups has two types of rules

inbound rule :- inbound meaning the traffic which allows inside to AWS.

Note :-- by default inbound rule deny all incoming traffic. whichever protocol with protocol is required that only you allow inside inbound rule. like **(ssh :22,rdp : 3389,http :80,https:443 , tcp:all).**

outbound rule :- outbound meaning the traffic which allows outside from AWS , by default outbound rule are allow all traffic.

Every EC2 will have a default Security Group.

what is stateful :-- if you allow any inbound rule then you no need to allow that on outbound rule , it will automatically allow from outbound rule . **that is called stateful**

what is stateless :-- if you allow any inbound rule then you must need to allow that on outbound rule , **that is called stateless** for example (NACL) are stateless

NACL :-- Network Access Control List , and this is used control your network. NACL is same like a Security Group but it is a another layer on Security Group for EC2 instance. If you want high security , then you go for NACL. in NACL, there are also inbound and outbound rules.

so first request will hit to NACL and then it will reach to Security Groups.

whatever we create our infrastructure we put everything in AWS everything should be inside the VPC.
when you Create the VPC with network range (Network range is nothing but IP address) .

in VPC we create partition , that partition called subnet. we create private and public subnet in VPC.

subnet rule

1. One subnet is associated to one availability zone.
2. One subnet can not be in multiple availability zone at the same time.
3. availability zone can have multiple subnets
4. Security Group is EC2 instance level.
5. NACL is subnet level.
6. one NACL can have multiple subnet
7. one subnet could not be in multiple NACL at the same time.

REAL TIME EXAMPLE of NACL:-- suppose you have more than 100 subnets, and you want to block/denay ssh in all subnets from inbound rules. so it is very difficilut to go in each subnet and remove ssh entry from Inbound rule.

instead of that, we add ssh entry in NACL, so automatically this ssh entry will be blocked/denay in subnets beause,
first request will hit to NACL and then it will reach to Security Groups.
when it found that SSH entry is block/denay in NACL, then
request will never reach to subnet.

Difference between Security Group and NACL

Security Group	NACL
have Inbound rules and Outbound Rule	have Inbound rules and Outbound Rule
have default Security group	have default Security group
Security Group will hit after NACL	NACL will hit before Security Group
By default, Inbound rules are deny	By default, Inbound rules are allow
You can not DENY on security group	You can DENY on NACL and allow also.
Security Group are instance level	NACL are subnet level
If you create new Security Group then inbound rules are DENY and outbound rules are ALLOWED	If you create new Security Group then inbound rules are DENY and outbound rules are DENY
Security Group are STATEFUL	Security Group are STATELESS
If you allow any inbound rule, you no need to allow on outbound rule.	If you allow any inbound rule, you MUST allow on outbound rule also.

****

AWS - EC2 Instance Auto-Scaling, Elastic Load Balancer

Auto-Scaling

What is auto scaling :-- Autos scaling meaning scale out and scal in. scale out meaning what increasing the number of EC2 instances. And scale in meaning what decreasing the number of EC2 instances.if you want to increase the more performance in the run time then go for autoscaling.

Whenever there is demand on the traffic Auto Sclaing Group (ASG) will scale out and scale in EC2 instances automatically.

Note :

1. Without Load Balancer scale out and scal in will never happen
2. ELB does health checks to the application.
3. CloudWatch will monitor the EC2 instances.

What is what is that desired capacity :-- we have three things here minimum , maximum and desire capacity.

Suppose , you want to setup the application, initially you need to have the number of E2 instance.

so how much EC2 instances you need that is called desired capacity

Example :--

1. first i have set 4 EC2 instances should be launch initially (this is DESIRED CAPACITY).
2. then i have set (MINIUM EC2 instances) **2**
3. then i have set (MAXIMUM EC2 instances) **6**

Now :--
1. when traffic increase then automatically scale out will happen and automatically EC2 instances will increased up to MAXIMUM capacity.

2. when traffic goes down then automatically scale in will happen and automatically EC2 instances will decreased up to MINIUM capacity.

So **DESIRED CAPACITY meaning**, at the time of launching your your setup how many instances that you need initially that is called DESIRED CAPACITY

Short note :--
MIN :-- The min number of EC2 instances that ASG should have (Example min=2).

MAX :-- The max number of EC2 instances that ASG should have (Example max=6)

DESIRED CAPACITY :- The number of EC2 instances that you wish/desired to launch initially.

There are 3 types of scaling options

1. Manual Scaling :- manual meaning if you are manually modifying minimum ,maximum , desire capacity is called manual scaling.

2. Schedule Scaling :-- certain period of time we want more EC2 instances (scale out) automatically increased, and after that EC2 instances should (scale in) automatically decreased that is called Schedule scaling.

Example :-- suppose every friday, there is huge rush in a mall, so we required more EC2 instances should automatically increased and for another days EC2 instances should automatically decreased , for that purpose , we can go for Schedule Scaling.

3. Dynamic Scaling :-- Dynamic scaling happen base on load/traffic and we can use metric to check the load/traffic.
Note :- metric are used to check load/traffic and the metric are **(CPU, NETWORK, REQUEST COUNT etc)** and base on that , we can do dynamic scalling. and these metric comes under the cloudwatch.

base on the traffic Dynamic scaling will automatically increase (scale out) EC2 instances and automatically decrease (scale in) EC2 instances.

Example :- in ASG , we select Metric type (Average CPU utilization) and set Target Value in percentage.

Example :-- if we set Target Value (60) percentage , then if CPU Utilization goes upto and above 60 percentage, the **(Scale Out)** will be occured.

when CPU Utilization will be below 60 percentage then **(Scale in)** will be occured.

Note :-- When (Average CPU utilization) get increased upto and above 60 percentage, means traffic is increased, so (scale out) will happen and When (Average CPU utilization) get decreased below 60 percentage, means traffic is decreased, so (scale in) will happen

What is Launch Template/ Launch Configuration :-- A Launch Template/ launch configuration is an instance configuration template that an Auto Scaling group uses to launch EC2 instances. When you create a launch configuration, you specify information for the instances.

In Launch Template/Launch Configuration , we put the all EC2 configuration details like (Amazon Machine Image (AMI), the instance type, a key pair, security groups, and other parameters used to launch EC2 instances.

Auto Scaling group **usage a launch template** , so using this template autoscaling groups will go and launch the machines.

REAL TIME EXAMPLE OF LAUNCH TEMPLATE AND AUTO SCALING GROUP:--

you can create your custom AMI and AMI content (instance type, volumes, security group, key-pair, tags, operating system,required software, custom application).

then use this AMI in Launch template. and auto scaling group usage this launch template for auto scaling.

Note :-- Auto scaling GROUP is combination of (ELB, EC2 instances, Launch Template) and , if you want notification, then you can use SNS also.

ELASTIC LOAD BALANCER :-- Elastic Load balancer which distribute the traffic to multiple EC2 instances across availability zone.
ELB is a service not server
ELB can be access using DNS name or URL.
ELB send traffic to healthy instances.

if any instance get un-healthy then Auto scaling group launch the new instance.
ELB follow the round robin policy.
ELB has the IP address but these IP are dynamic not static , so AWS always recommend to use the ELB DNS name not IP address.

TYPES OF LOAD BALANCER :--

1. classic load balancer :-- This classic load balancer is previous generation load balancer and it work on HTTP https and TCP.

2. Application load balancer :-- This Application load balancer is later generation load balancer and it (works on http, https) protocol and it works on Layer-7 , layer 7 means (Application Layer). This is Default load balancer.

Application load balancer have routing features like
a) host base routing
b) path base routing
c) String parameter base routing.

in Load Balancer , we create the Rules like

1. Path base routing
2. Host Header base routing
3. Http Header base routing
4. Http Request base routing
5. Query String base routing
6. Source IP base routing

and base on this rules , we route the request to respective Target Group and this target will communicate with backend application endpoint.

EXPLANATION :--
1. Load balancer will accept request through url and will send request to rule

2. as per the rule for requested url, it will send traffic to target group

3. Target group is nothing but group of EC2 Instacne, which will manage the request.

For example , there are two url (https://swami.com & https://om.com)

In load balancer, for each url will have different rules and target group

1. suppose if user hit https://swami.com ,
2. then load balancer send this url to Rules
3. Rule check the url and check the request, and it send this request to SWAMI-TG (Target Group)
4. SWAMI-TG (Target Group) Will have group of EC2 Instacne, which will manage the request. and will response to request

1. now suppose if user hit https://om.com ,
2. then load balancer send this url to Rules
3. Rule check the url and check the request, and it send this request to OM-TG (Target Group)
4. OM-TG (Target Group) Will have group of EC2 Instacne, which will manage the request. and will response to request

NOte :-- so, we don't need create separate load balancer for each URL, with Single load balancer, we can manage the traffic for different different URL.

3. Network load balancer :-- (works on tcp, tls, udp) protocol, it works on Layer-4 , layer 4 means (Network Layer). It is used for networking purpose. If you want extreme high performance, then you go for Network load balancer. this Network load balancer provides 1 static ip per availability zone.

4. Gateway load balancer :-- (woks on GENEVE protocol with 6081 port) and it used for third party like firewall and it works on Layer-3 , layer 3 means (Gateway Layer).

****

AWS - Types of IP's, GA (Global Accelerator)

What are the **7 steps to Launch EC2 instance.**

1. Select AMI (Amazon Machine Image), AMI is copy of the Operating

system.
2. Select Instance Type (Example t2.micro)
3. set instance Configuration (means, how many instances you want to launch,do you want to Public IP or not, do you want to provide
   User data or not, etc)
4. select storage / volumes (EBS volume) and also attach
   additional volume if required
5. set Security Group
6. set Tag (key- value pair). Example :-- **name=LinuxServer**
7. Review and create and attach (.pem) file.

Types of IP's :-- we have three types of IP address

1. Public IP **:--** This ip is not mandatory , with Public IP we can connect from out side world to EC2 instance.or can access any AWS resource from out side world. The problem is , when you start and stop EC2 instance, you get new public address. means public IP is not static IP means Public IP is dynamic. Public IP is optional.

2. Private IP **:--** This ip is mandatory ,whenever you launch the EC2 instance ,by default you will get private IP, you can not connect to the private IP directly from out side world to EC2 instance. or can't access any AWS resource from out side world. In AWS within the VPC , if two EC2 instance want to talk to each other, then we use the Private IP, means internal communication within the VPC, we used private IP. Private IP is static and it never change.

3. Elastic IP **:--** Public IP and Elastic IP both are same but Elastic IP is static IP and fixed, once it assigned to EC2 machine,it never change and with Elastic IP we can connect from out side world to EC2 instance.or can access any AWS resource from out side world. Elastic IP is optional.

Note **:--** whenever you get an elastic IP you attach it to the E2 instance but if you don't assign it and keep it ideal then you will need to pay for that , because you are keeping the Elastic IP ideal.

what is instance metadata **:--** data about EC2 instance is called instance metadata.

Example of instance metadata **:--** The all below type of information is called instance metadata.
1. who has launched the E2 instance.
2. when it has launched.
3. what type of volume it has.
4. which Security Group is attached.
5. which VPC is attached
6. which subnet it is attached.

Note : --

1. From your AWS Console , The " detail section" will give you all information about metadata.
2. If you want all information about metadata from CLI , then you run bellow URL from CLI.

URL :-- http://169.254.169.254/latest/meta-data/

What is USER DATA :-- Which data is stored , which application is installed on EC2 instance, that is called USER DATA.

Example :- if you launch EC2 instance, and on this instance, if you install Apache server, this Apache server will called (USER DATA). suppose if you upload any data on S3, that is also called (USER DATA).

suppose your manager told you to configure 100 E2 instance and install APACHE WEB SERVER and also install your web application on it.

so in this case , (downd load and install APACHE WEB SERVER and configure your web application on APACHE WEB SERVER) is called USER DATA.

so it is time consuming to download apache web server and install it on 100 EC2 instance, so good practice is, you can write a nice shell script with required USER DATA (downd load and install APACHE WEB SERVER and configure your web application on APACHE WEB SERVER) and then launch 100 EC2 instance at a same time, automatically this shell script will execute, when your EC2 instance will start and will install all required application, which you mentioned inside the shell script.

Note :-- so this USER-DATA is also called BootStrap scripting file, which execute only once, when your EC2 Instance will boot/start.

There are two type of IP

1. Unicast IP :-- One server hold one IP address
2. Anycast IP :-- All server holds same IP address, and the client is routed to the nearest one.

Global Accelerator :-- AWS Global Accelerator is a networking service that helps you improve the availability, performance, and security of your public applications. Global Accelerator is a global service that supports endpoints in multiple AWS Regions. By default, Global Accelerator provides you with static IP addresses that you associate with your accelerator. **AWS Global Accelerator is not a free service.**

Global Accelerator is a global service that supports endpoints in multiple AWS Regions. Global Accelerator uses Edge Locations to find an optimal pathway to the nearest regional endpoint.

AWS Global Accelerator vs Amazon CloudFront -
CloudFront uses Edge Locations to cache content while Global Accelerator uses Edge Locations to find an optimal pathway to the nearest regional endpoint.

****

AWS EC2 Practicals, AWS Console

NOTE :--

1. EC2 Dashboard :-- remember after doing our practicals ,come back to EC dashboard and make sure everything should be zero except key pair and Security Group.

2. (IAM , key pair & Security Group) services are free and all others services are not free.

3. EC2 Global View :-- This is service is used to check, in which regions ec2 instances are in running state.
so, All the regions will be listed here and will show per region wise how many ec2 instances are in running state.

4. changes to the security group will be affected immediately.you no need to stop the ec2 instance to modify the security group.

What is Service Quota :-- limits are called now Service Quota and remember **maximum 20 E2 instance you can launch in your fresh account** .

WHAT is EFA :-- (EFA) is **Elastic Fabric Adapter,** and it is a network device that you can attach to your EC2 instance to reduce latency and increase throuhput for distributed High Performance Computing (HPC) and Machine Learning (ML) applications.

Note :-

1. While creating / Launching EC2 instance, in **"Advance network configuration"** tab, we can enable (EFA)

2. We can't enable (EFA) for t2.micro machines.

WHAT IS NITRO ENCLAVE :-- A Nitro Enclave is a **trusted execution environment** (TEE) in which you cn securly process the sensitive data.

Note :-

1. While creating / Launching EC2 instance, in "Advance Setting" tab, we can enable (NITRO ENCLAVE)

2. We can't enable (NITRO ENCLAVE) for t2.micro machines.

3. Nitro enclaves will support which has more than two CPUs instance

WHILE CREATING EC2 INSTANCE HOW MANY INSTANCE STATES ARE THERE ?

There are 6 (SIX) instance state

1. Pending state :-- whenever you launch the ec2, it will be in Pending state.

2. Running state :-- When EC2 starts , then it will be in Running state.

3. stopping state :-- whenever you stop the E2 istance , then it will be in Running state.
4. stopped state :-- When EC2 instance stops, then it will be in stopped state.

5. shutting down state :-- whenever you shutting down the E2 istance , then it will be in shutting down state.

6. terminated state :-- When EC2 instance shutdown, then it will be in terminated state.

IMPORTANT NOTE --

1. If you stop and start the EC2 instance, then **EC2 instance will get new Public IP address means Public IP address**
is not static, it is dynamic and if Public IP address changed then Public DNS URL will also change.

2. for virtualization , we normally use VMware but in AWS , we don't use VMware, In AWS, we use (XEN) for virtualization. means
AWS usages (XEN) for virtualization.

3. While creating the image we cannot change root volume properties except size.

4. while creating IMAGE (AMI) we can also add additional volume [Elastic Block Storage Volume (EBS), Instance Storage Volume (isv)].

5. If your EC2 instance root volume is not encrypted then while creating image root volume will not encrypted.

6. If your EC2 instance root volume is encrypted then while creating image root volume will encrypted.

7. While creating IMAGE (AMI), additional volumes can be encrypted.

8. During the image creation process E2 creates a snapshot of each volume.

9. D-register AMI means deleting the AMI.

10.in AMI (Amazon Machine Image) dashboard, you just click on **"Action"** button and select **"Edit Ami permissions"** option, with "Edit Ami permissions", we can do two things what are those you can make the Ami public and you can share it to others AWS Account.

11. Recycle bin can be implemented on Snapshot and AMI. not every AMI and every snapshot will go to Recycle bin.

****

AWS - EC2 Practicals 4 = EC2 Instance Console

work on volumes and life cycle manager

Volumes creates base on EC2 instance. **per EC2 instance will have 1 volume and this is called root volume.**

We can modify the volume and when we take the backup of volume that is called snapshot.

to delete root volume, you need to first detach it from EC2 then you can delete it. so we need to first stop the EC2 instance
after that you can detach it from EC2 then you can delete root volume.

without root volume EC2 instance will never work, because operating system founds in root volume, so to run or work EC2 instance
you need to attach root volume.

Note :-

1. root volume name is :-- /dev/sda1

2. root volume can be available base on availbility zone, you can't use another availbility zone's (root volume) in another availbility zone.

3. Every EC2 instance will have only 1 root volume and you can attach additional volume to EC2 instance to store data and additional volume are chargable and additional volume name is :-- xvdf

4. you can increase the volume size on fly, means you don't need to stop EC2 instance.

5. instance type combination of memory and CPU and it can be **(t2.micro, t2.small, t2.large, t2.exlarge)** etc.

6. instance type and volumes are totally different.

7. default volume type of **root volume is (GP2) General purpose SSD**

Note :-- Recylebin can be used only for snapshot and AMI.

Create snapshot lifecycle policy :-- The lifecyle manage is used to automatically create images and snapshot. for that you need to create CRON (snapshot schedule time, day). snapshot are incremental.

****

AWS - 32 Ec2 Instance Load Balancer

We have 4 types of Load Balancer

1. classic load balancer this is deprecated and out dated
2. Application load balancer (works on http, https) protocol
3. network load balancer (works on tcp, tls, udp) protocol
4. gateway load balancer (woks on GENEVE) protocol used for third party like firewall.

We create load balancer with target group and using DNS name we can use load balancer.

load balancer distribute the traffice to EC2 and using health check Load balancer do monitoring.

Load balancer usages (http/ https) listeners and it usages internate facing.

in Load Balancer , we create the Rules like

1. Path base routing
2. Host Header base routing
3. Http Header base routing
4. Http Request base routing
5. Query String base routing
6. Source IP base routing

Note :-

1. https is secured , for that we required SSL certificate, in AWS, we use (ACM) Amazon Certificate Manager service to get SSL certificate, so using ACM , you can purchase the (SSL certificate) for HTTPS url. and in Load balancer , you can attach (SSL certificate) and make url secured (HTTPS).

ACM Service :-- Go to ACM service and click on Request Certificate, then select Request a public Certificate , then provide Domain name : **(Example : *.boom.com)**, then click on the Certificate and Create Records in **ROUTE53** (Make sure you purchase the domain first and then create record)

2. Route 53 is DNS service in AWS.

3. Load balancer send traffic to availbility zone.
4. Load balancer are reginal.

Important Note :- Advanced health check in load balancer

1. **Healthy threshold :--** here we can set count , for example (5 five) , so Load balancer will ping to the application (5 five) time to check the application is running or not . means it will check (5 five) time the application is running well or not.

2. Unhealthy threshold :-- here we can set count , for example (2 two) time,so Load balancer will ping to the application (2 two) time and for first try we don't get response and also for second try , if we don't get response from application, then it will be consider as Unhealthy. means we don't get response (2 two) time, then load balancer will consider the application is Unhealthy.

3. Time out :-- here we can set seconds, for example (5 second) , if we don't get response from application within (5 second) then load balancer will consider the application is Unhealthy.

4. Interval :-- here we can set seconds, for example (30 second) , every thirty second , load balancer will check health check of application.

5. Success code :-- here we can set success status code, for example (200 status code) , load balancer excepet 200 status code for healthy application.

Important security point :--

We can modify the Load balancer attributes, and can enable the (WAF) Web application firewall, WAF will stop all kind of hacking activites and it prevent application from hacking.

so whenever user hit the endpoint request before request goes to load balancer,WAF will check the request is valid and safe or not. so WAF will going to secure the applicaiton from hacking, sql enjection etc and stops (DDOS attack).

for that , **you just enable :--** WAF fail open option in Load balancer attributes .

TARGET Group :-- we create Target group and attach to Load balancer and it follows the (Round robin) Load balancing algorithm.

There are 3 types of scaling types (manual, schedule & dynamic). ASG is dynamic auto scaling type and we use ASG with load balancer.

Auto Scaling with Load Balancer :-- Using Auto scaling , we can avoid the fault tolerance issue. fault tolerance means zero down time.

1. create Elastic Load Balancer with emty Target Group

2. Launch Template(LT) and add boostrap script (user data) for EC2 instance.

3. Using with Launch Template (LT) create Auto Scaling Group (ASG)

4. in ASG, we configure (Minimum,Maximum & desire) Capacity of EC2 machine.

For Example , if you set
Minimum Capacity =2 (Ec2 instance)
Maximum Capacity =6 (Ec2 instance)
Desire Capacity =4 (Ec2 instance) desire capacity the number of E2 instance that you wish or desire to launch initially that is called desire capacity.

then ASG will automatically create 4 (Ec2 instance) and will keep minimum 2 and max 6.

5. ASG follow th Scale out and scale in process

a) Scale Out :-- means increase the (Ec2 instance)

b) Scale in :-- means decrease the (Ec2 instance)

Note :- in ASG , we select Metric type (Average CPU utilization) and set Target Value in percentage.

Example :-- if we set Target Value (60) percentage , then if CPU Utilization goes upto and above 60 percentage, the (Scale Out) will be occured and when CPU Utilization will be below 60 percentage then (Scale in) will be occured.

Note :-- in SNS there are below event types and it executes when instance

1. Launch
2. Terminate
3. Fail to Launch
4. Fail to terminate

or normally while (Scale Out or Scale in) something goes wrong means (Scale Out or Scale in) not working properly then you will get notification, and for getting notification we use SNS service (simple notification service).

There are two thing in **SNS.**

1. Topic :-- you can create any topic
2. Subscription :- and you can subscribe the created topic. using email Id or Phone number. you can subscribe the topic and get notification.

Note :--

LAUNCH TEMPLATE :-- If you want to have a big applicaiton such as (tomcat, java, python etc) on EC2 instances, then create EC2 instance first, deploy or install all required software and application and create (AMI) Amazon Machine Image of that EC2 instance and use it this custom (AMI) with Launch Template.

ASG (Auto scaling Group) :-- now create the ASG and set launch template (Which was created using custom AMI).

What is draining :-- draining means to removing the ec2 instances from from the target group.

****

CloudWatch Alarms Events & Logs

CloudWatch service is reginal service and used for monitoring purpose, Cloud watch is used to monitor performance of AWS resources. **(Alarams, Events & Logs)** are important three things in Cloudwatch.

Remember this Cloud watch can monitor only this host level metrics.

There are two types of monitoring in cloudwatch

1. basic monitoring :- It is free, and will monitor every 5 minutes and generate log.

2. detailed monitoring :- It is not free, and will monitor every 1 minute and generate log.

To monitor the AWS resources, we usages **"HOST LEVEL METRICS"** and **"HOST LEVEL METRICS"** are (1.CPU, 2.Network, 3.Disk, 4.status checks). this host level Matrix are nothing but default matrics.

Alarms :--

Alarms can can send you the notifications and also can also do some actions what are the actions.

it can stop EC2 instance, it can terminate EC2 instance, it can recover EC2 instance or It can reboot EC2 instance.

Alarms has **three state (1. In Alaram, 2. Ok, 3. Insufficient)**

alarm's actions can be enabled and disabled any time. Alaram actions are (stop, terminate, reboot, stop).

Suppose you have configured the CPU utilization if if it is greater than 90% notifications should come then,

1.If CPU utilization greater than 90 % in Alaram state will be :-- (in Alaram) state.

2.If CPU utilization less than than 80 % in Alaram state will be :-- (Ok) state.

3.If EC2 instance stopped due some issue then Alaram state will be :-- (Insufficient) state.

Cloud Watch alarms are on single metric.
Composite alarms :--- Composite alarms are monitoring the states of multiple alarms.
Example : - AND Conditions , OR Conditions.

you have multiple alarms and one alarm is depend on another alarm then must go with Composite alarms.

==========

Events :--

Events are stored in Event Bridge.

EC2 have 6 states (1. pending 2. running 3. stopping 4. stopped 5. shutting down 6. terminated). and based on these states
Event can be occour.

So What is event Bridge :--

I can create the rule base on event and execute respective target that is called event bridge.

Example :--

1. suppose , someone **stopped EC2** instance means stopping state will start and this state will called event and here we create the RULE, that if stopping state occured then send SNS. this process called event bridge.

2. suppose , someone **starts the stopped EC2 instance** means pending state will start and this state will called event and here we create the RULE, that if pending state occured then execute Lambda Function to Terminate the newly started EC2 instance. this process called event bridge.

WE can create the CRON job or schedule which will act as events.

Example :--

1. every day sharp 9 am, EC2 machine should start :-- for this we will create Lambda function , which will automatically start EC2 machine sharp 9 am

2. every day sharp 9 pm, EC2 machine should stop :-- for this we will create Lambda function, which will automatically STOP EC2 machine sharp 9 Pm.

so event bridge means :-- , if you want to schedule something , then you need to choose event Bridge there you can schedule something and invoke the services.

What is names space :-- names space is nothing but group of matrics group of matrics or collection of related Matrix.

==========

Logs :-- All AWS Resources log will come in centralized place that place is cloudwatch logs.

you need to install cloudwatch agent in all EC2 instances , this cloudwatch agent will push all logs to cloudwatch logs.

You need to create IAM Role and attach **"Cloud watch permission"** and attach this role to EC2 instance then cloudwatch agent will able to send log to cloudwatch logs.

Canary is a feature that is implemented in AWS and Canary is a feature from cloud watch where it is used to do the application monitoring.

Canary is related to application and status are related to EC2 instance

==========

Lambda Limit :--

Lambda usage default memory (128 MB) to excute the program and maximum 10 GB.

Lambda usage default execution time 900 seconds (15 minutes)
Lambda usage Environment variable up to 4KB

Lambda usage default Disk capacity (512 MB) to excute the program and maximum 10 GB.

Lambda usage concurrency Execution =1000 time (and it can be creased)

Deployment :-- Lambda function deployment sie (compressed .zip, .rar) =50 MB
Lambda function uncompressed deployment sie =250 MB

Lambda will be billed only for the execution time.

==========

Reference Link :--
https://k21academy.com/amazon-web-services/blue-green-deployment-in-aws/

What is Blue green deployment model :- there will be two envrionment, using swap URL , we can achieve Blue green deployment.

A blue/green deployment is a deployment strategy in which you create two separate, but **identical environments**. One environment **(blue)** is running the current application version and one environment **(green)** is running the new application version.
Blue/green deployment, sometimes referred to as red/black deployment, is a technique for releasing applications by shifting traffic between two identical environments running differing versions of the application.

The basic idea is to shift traffic between two identical environments, running different versions of your application. the **blue colour usually signifies the live version**, and the **green colour signifies the version that needs to be tested.** These can be swapped too.

After the green environment is ready and tested, production traffic is redirected from blue to green.We can use a load balancer to route traffic between them.

****

AWS - S3 Storage Classes, Glacier, LCM Rules, CORS, CRR, SRR, Encryption.

S3 Storage Classes :--
whenever we upload any objects into S3 , we must need to first select the storage class.

There are different types of storage classes (Standard Frequently Access , Standard Infrequently Access ,
Reduced Redundancy Storage, One Zone Infrequently Access, Intelligent Tire , Glacier, Deep Glacier).

Standard Frequently Access storage classes :-- It is default storage class and it is used for general purpose, whoever have a requirement to use the data frequently go for Frequently storage classes. There is no Reterival Charges for Frequently Access storage classes.

Standard Infrequently Access storage classes :-- This is used for infrequently access data, then whoever have a requirement to use the data not frequently go for Standard Infrequently Access storage classes.
There is Reterival Charges for Infrequently Access storage classes. and it cheaper than (Standard Frequently Access storage classes).

Reduced Redundancy Storage classes :-- (same copy of same uploaded object in multiple locations are called Redundancy).
If you go with (Reduced Redundancy Storage classes) then AWS will never keep any copy of same uploaded object in multiple locations.
It is used to Frequently Access data but data will not critical/ importanat. There is no Reterival Charges for Reduced Redundancy Storage classes.

It is cheaper but AWS suggest to don't use this Reduced Redundancy Storage classes.

One Zone Infrequently Access :-- It is used for Infrequently Access data but data will not critical/ importanat. if you go with (One Zone Infrequently Access) storage then, you can save your data in single availability zone. It is cheaper but AWS suggest to don't use this.
There is Reterival Charges for One Zone Infrequently Access storage classes.

Intelligent Tire Access :-- Intelligent Tire means unknown access pattern, Whenever user doesn't know that how he will access S3 Storage (frequently or Infrequently) then user can go with Intelligent Access storage. So base on data usage it will automatically shift to **(frequently or Infrequently)** storage classes.

Glacier Access :-- Glacier is used for Infrequently data Access and it stores data in compressed (archieve [.zip]) format.
in Glacier , we create walt and in vault AWS stores data in in compressed (archieve) format, so we can say vault is noting but the one kinde of container of all compressed (archieve [.zip]) files. one (archieve [.zip]) file size can be 40 TB (Terra bytes).
There is Reterival Charges will be applied for Glacier Access.

Glacier has below Reterival options.

1. **Expedited :--** Whenever you want to reterive the data within 1 to 5 minutes.
2. **standard :--** Whenever you want to reterive the data within 3 to 5 hours.
3. **Bulk :--** Whenever you want to reterive huge the data within 5 to 12 hours.

Deep Glacier :-- Deep Glacier is same as Glacier only different is here data will be available for 180 days.
Minimum duration :- 180 days

Every Storage classes with (availability & durability) for data.

availability :- means any time
durability :- means long time

for Standard frequently Access storage classes (availability & durability) will be
availability :- 99.99 %
durability :- 999999999 (11 nine)
Minimum object size :- 0 Bytes

for Standard Infrequently Access storage classes (availability & durability) will be
availability :- 99.9 %
durability :- 999999999
Minimum object size :- 128 KB
Minimum duration :- 30 days

for Reduced Redundancy Storage classes (availability & durability) will be
availability :- 99.99 %
durability :- 99.99 %

for One Zone Infrequently Access (availability & durability) will be
availability :- 99.5 %
durability :- 999999999
Minimum object size :- 128 KB
Minimum duration :- 30 days

for Intelligent Tire Access (availability & durability) will be
availability :- 99.9 %
durability :- 999999999
Minimum duration :- 30 days

for Glacier Access (availability & durability) will be
availability :- 99.99 %
durability :- 999999999
Minimum duration :- 90 days

What is life cycle management :-- life cycle management is nothing but Life cycle rules and this is used to move your objects
from one storage class to another storage class automatically.
It is possible to move the objects from one storage class to another storage class automatically by created life cycle rules.
This Life cycle rules can be create on S3 bucket level and it will be applicable for object.
there are two thing (Tranistion and Expiration) in life cyle managment

What is **Tranistion and Expiration** in S3 life cyle managment

1. Tranistion :-- Transition actions – These actions define when objects transition to another storage class. For example, you might choose to transition objects to the S3 Standard-IA storage class 30 days after creating them, or archive objects to the S3 Glacier Flexible Retrieval storage class one year after creating them.

2. Expiration :-- S3's new Object Expiration function allows you to define rules to schedule the removal of your objects after a pre-defined time period. The rules are specified in the Lifecycle Configuration policy that you apply to a bucket. You can update this policy through the S3 API or from the AWS Management Console.

What is Athena in S3 :-- Athena is serverless service and Athena will analyze the logs directly from S3, Amazon Athena is an interactive query service that makes it easy to analyze data directly from Amazon S3 using standard SQL.

How many types of Encryptions in S3? ->
by default S3 Bucket Encryption is enabled and Amazon S3 has 3 types of Encryptions

1. Server Side Encryption :-- Encryption can managed by Server and in server side encryption , there are 3 TYPES (S3, KMS, C)

a) S3 :-- AWS managed key [mostly S3 usage this encryption] and this encryption usage Advance Encryption standard (AES 256) algorithm.

b) KMS :-- AWS KMS key
C) C :-- Customer provided key

2. client Side Encryption :-- Encryption can managed by client

3. Transit Side Encryption :-- Encryption can managed by (HTTPS) protocol

What is Pre-Signed URL in S3 ?
Pre-Signed URL is used for to give temporary access for certian period of time to user.

What is the use of S3 transfer Acceleration?
Amazon S3 Transfer Acceleration is a bucket-level feature that enables fast, easy, and secure transfers of files over long distances between your client and an S3 bucket. Transfer Acceleration is designed to optimize transfer speeds from across the world into S3 buckets.

Special Acceleration fee: $0.04 per 1GB downloaded. Standard Data Transfer fee: $0.085 per 1GB, given that this 10 TB download was not the only time you have used Amazon S3 within the same month.

what is transfer acceleration in aws ??
Transfer Acceleration uses the globally distributed edge locations in CloudFront for data transport. The AWS edge network has points of presence in more than 50 locations. Today, it is used to distribute content through CloudFront and to provide rapid responses to DNS queries made to Amazon Route 53.

what is crr and srr in aws s3

Cross-Region Replication (CRR)—copies S3 objects across multiple Amazon Regions (ARs), representing geographically separate Amazon data centers.

Example :--
Suppose if you created 1 bucket (my-bucket1) in one region and also created 1 bucket (my-bucket2) in another region now the requirement is , whenever you upload any objects in (my-bucket1) automatically make a copy this uploaded object in another region.

for that we can use Cross-Region Replication (CRR) option of S3 , so in S3 create the Replication Role and set destination of another region.

Same-Region Replication (SRR)—copies S3 objects between buckets in different availability zones (AZs), which are separate data centers in the same AR.

Example :--
Suppose if you created 1 bucket (my-bucket1) in one region and also created another bucket (my-bucket2) in same region
now the requirement is , whenever you upload any objects in (my-bucket1) automatically make a copy this uploaded object in same region.

for that we can use **Same-Region Replication** (SRR) option of S3 , so in S3 create the Replication Role and set destination of same region.

****

AWS - S3 Requester Pay S3 Event Notification S3 Batch operations S3 Access Point

S3 Requester Pays:-- In General, Bucket owner pay all S3 storage data transfer cost associated to their Bucket.

With Requester Pays Buckets, the requester instead of the bucket owner pay the cost of the request and the data download from the bucker.

The requester must be authenticated in AWS, so that AWS knows where to charge (in their AWS account). Can not be anonymous.

S3 Event Notifications :-- S3 will send Notifications using **(SNS,Lambda, SQS)** generally S3 Event send Notifications using SNS.

S3 Batch Operations :-- It perform bulk operations on exisiting S3 objects with a single Request. S3 Batch Operations is a managed solution for performing storage actions like copying and tagging objects at scale, whether for one-time tasks or for recurring, batch workloads. S3 Batch Operations can perform actions across billions of objects and petabytes of data with a single request.

S3 Access Point :-- An Access Point can support a single user or application, or groups of users or applications within and across accounts, allowing separate management of each access point. Every access point is associated with a single bucket and contains a network origin control, and a Block Public Access control.

Example :--

Suppose in your S3 bucket, you have created three subfolders/(prefix) (1. devops , 2. Developers, 3. Sales) and i want to give access to each sufolder/(prefix), means devops team can access only (devops folder) not other, Developers team can access only (Developers folder) not other, Sales team can access only (Sales folder) not other.

To achieve this, we can use S3 Access Point, we can create (S3 Access Point) for each subfolders/(prefix) (1. devops , 2. Developers, 3. Sales) so user will access respective folder/(prefix) using this (S3 Access Point).

Instead of writing critical bucket policies, you can create access points to each subfolder (prefix) and give the DNS Names to the users to access their respective folders in buckets.

Access Points can be public (internet) or Private (VPC).

Note :-- If you create private buckect then all uploaded or stored objects in this private bucket will private access If you create public buckect still all uploaded or stored objects in this private bucket will private access , but you can make it public if required. so default object nature is private.

****

AWS - EFS Service

EFS Service :-- EFS means **Elastic File System** and EFS is regional service, EFS is for Linux operating system.

We mount the EFS to EC2 instance(Linux)

EFS usages the NFS (Network File System) protocol. EFS is a file storage service for use with Amazon compute (EC2, containers, serverless) and on-premises servers. EFS provides a file system interface, file system access semantics (such as strong consistency and file locking), and concurrently accessible storage for up to thousands of EC2 instances.

Note :--

1. you can select availability zone and enable EFS for selected availability zones.

2. after creating EFS, go to the "Replication" Tab

Benifit of "Replication" :-- suppose you are in MUMBAI region and you want to replicate the same EFS in another Region, then you can use "Replication". for that and do following action.

1. Click on "Create replication" button.
2. select "Destination AWS Region" From dropdown.
3. and then Click on "Create replication" button to create "Replication".

Using **"Replication"** option , we can transfer data from one region to another region automatically.

Note :--

1. after creating EFS , then launch EC2 instance and attach created (EFS).

2. then go to "Security Group" and add NFS protocol in "IN BOUND" rule.

FSX service :-- for Windows we can use FSX service. FSX means (File systems, backups, and file shares.)

FSX is regional service, We mount the FSX to EC2 instance (Windows)
Amazon FSx lets you easily and securely backup, archive, or replicate your on-premises file storage to AWS in order to meet regulatory, data retention, or **disaster recovery** requirements.

****

AWS - CLI Tutorial & Practical in AWS Console

We can access AWS with two way
1. console access
2. programmatical access

AWS CLI :-- AWS CLI is AWS Command Line Interface. using CLI , you can control multiple AWS services from the command line and automate them through scripts.

The AWS Command Line Interface (AWS CLI) is an open source tool that enables you to interact with AWS services using commands in your command-line shell.

First install AWS CLI using below command on Linux/ubuntu

curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"

unzip awscliv2.zip

sudo ./aws/install

Then configure AWS using below AWS CLI command

1. aws --version :-- This command will show the installed version of AWS.

2. Create Access key in AWS so Follow below steps to create the ACCESS KEY in AWS

1. Go to the AWS and select IAM service

2. then select User , which you are created.

(If user is not created, then create it.)

3. after seleting User, go to "Security Credentials" tab

4. then select "Access Keys" option.

5. then select "Command Line Interface (CLI)" option.

6. then just click on "Next" button and after click on "create Access Key" button.

2. aws configure :- This command will ask below question

1. access key = Provode the <ACCESS KEY>
2. secret key = Provode the <SECRET KEY>
3. region   = Provode the <REGION>
4. output format = table

To get the help , that how to use AWS CLI command run following command on terminal

1. aws help
2. aws ec2 describe-instances help :-- This command shows EC2 DescribeInstances operation, including descriptions of its input parameters, filters, and output.

****

AWS - Transfer Family Service

AWS Transfer Family service is new service , it fully managed service and used for file transfer and it support (FTP, FTPS, SFTP) protocol.
Data can be transfer in and out of S3 BUCKETS and EFS.

AWS Transfer Family is a secure transfer service that stores your data in Amazon Simple Storage Service or Amazon Elastic File System and simplifies the migration of Secure File Transfer Protocol (SFTP), File Transfer Protocol Secure (FTPS), File Transfer Protocol (FTP) , and Applicability Statement 2 (AS2)

AWS Transfer Family supports transferring data over the following protocols:

a) Secure Shell (SSH) File Transfer Protocol (SFTP): version 3

b) File Transfer Protocol Secure (FTPS)

c) File Transfer Protocol (FTP)

d) Applicability Statement 2 (AS2)

PRACTICAL DEMO :-- (Transfer Family Service)

1. Create a Public Bucket
2. Create a Bucket Policy and add/ attach to S3 bucket
3. create (Transfer Family server)
4. add user , while creating user create ROLE with S3 permission
5. generate the public and private keys using putty generator.
6. Open filezilla from your local machine and push th files.

---

1. in S3 Create a Public Bucket and provide below information
   a) Bucket Name :-- 6PM-TRANSFER-DEMO
   b) AWS Region :-- Asia Pacific (Mumbai) ap-south-1
   c) Object Ownership :- enable (ACLS enabled) option.
   d) Block Public Access settings for this bucket :- [uncheck] Block all public access
   e) then just click on "Create Bucket" button.

---

2. Create a Bucket Policy and add/ attach to S3 bucket

   a) To create the Bucket Policy , first select "6PM-TRANSFER-DEMO" bucket from S3 bucket list and just select "Permission" tab.
   b) now go to "Bucket Policy" section and just click on "Edit" button.
   c) just click on Policy generator Button.
   d) In AWS POLICY Generator configure below information
      a) Select Type of Policy :-- S3 Bucket Policy
      b) Principle:- *
      c) Select Action :-- GetObject, ListBucket, PutObject
      d) Amozon Resource Name (ARN) :- <HERE YOU select the ARN of BUCKET> , set ARN of "6PM-TRANSFER-DEMO" bucket
         Note :- To get the ARN of "6PM-TRANSFER-DEMO" bucket, go and select "6PM-TRANSFER-DEMO" from S3 bucket list and just click on "Copy content" button and paste it as ARN in S3 Bucket Policy.

      e) just click on "Add condition" button and configure below information
         a) select Condition :- Ipaddress
         b) select key   :- Source Ip
         c) set value   :- here set the <PUBLIC IP ADDRESS OF CLIENT MACHINE>/32
            Note :- get the PUBLIC IP ADDRESS OF CLIENT MACHINE , ask client to open browser and search "WHAT IS MY IP" in google search bar.
            Example :- 183.82.125.5 (This is public ip address of client)
            We need set this ip along with (32)
            Full example :--183.82.125.5/32
         d) then just click on "Add condition" button.
         e) then just click on "Add statement" button.
         f) then just click on "Generate Policy" button.
         g) then copy policy and paste add it in S3 bucket (in EDIT BUCKET POLICY) section.

---

3. create (Transfer Family server) :-
1. Search and open "AWS Transfer Family" service from Amazon Services.
2. then just click on "create server" button.
3. select the protocols you want to enable :-- SFTP (SSH file Transfer Protocol)
4. then just click on "next" button.
5. then select Identity Provider for (SFTP, FTPS, or FTP) :-- Service managed
   Note :-- Identity Provider means here we are selecting the user.
6. then just click on "next" button.
7. select Endpoint configuration :-- Publicly accessible
   Note :-- in real time must select (VPC hosted) option.
8. select hostname :-- None
9. then just click on "next" button.
10. choose a domain :-- select Amazon S3 option
11. then just click on "next" button.
12. in Configure additonal details :- don't select any thing
13. then just click on "next" button.
14. then just click on "create" button to Transfer Family server.

---

4. add user , while creating user create ROLE with S3 permission :--

Note :- first you need to create ROLE first, so go IAM service and select Roles option and just click on "Create Role" button and complete below configuration.

1. select Trusted entity type :- AWS service
2. select Use cases for other AWS services :-- Transfer option
   Note :-- here Transfer means (AWS Transfer Family).
3. then just click on "next" button.
4. select Permission Policies :-- AmazonS3FullAccess
5. then just click on "next" button.
6. set Role Name :-- TransferRole
7. then just click on "Create Role" button to create Role.

In Transfer Family server , select newly created (Transfer Family server) and just click on "Add user" button and complete below configuration in User configuration section

User configuration
1. User Name :- demoUser (Note , you can give any name).
2. select Role :-- select newly create Role <TransferRole>
3. Home Directory :-- here select Bucket name <6PM-TRANSFER-DEMO>
4. set User :-- demouser
5. SSH public keys:-- using putty create public key and paste it here.
6. then just click on "next" button to add user.

---

6. download filezilla and install it on your local machine and push th files. and do below action

a) add site manager and select SFTP
b) host :-- set <TRANSFER HOST NAME> for that go to Transfer Server and and select newly created "Transfer server" and copy the "custom host name" and paste it here.

c) port :-- 22
d) select Logon Type :-- select key file
e) username :-- set <USER NAME> which you are created In Transfer Family server (demouser)
f) set the download (.ppk) file of AWS and just click on "connect" button.

****

AWS - Storage Gateway Service Tutorial & Practicals

AWS Storage Gateway is a hybrid cloud storage service that gives you on-premises access to virtually unlimited cloud storage. Storage Gateway provides a standard set of storage protocols such as iSCSI, SMB, and NFS, which allow you to use AWS storage without rewriting your existing applications.

This storage Gateway is the combination of your on premises (local machine (EC2) storage means HDD drive) storage and then AWS storage(S3,EBS ,Glacier and FSX) .
It is used to transfer data from on on premises (local machine (EC2) storage means HDD drive) to AWS storage (S3,EBS ,Glacier and FSX) and from AWS storage(S3,EBS ,Glacier and FSX) to on premises (local machine (EC2) storage means HDD drive) .

AWS storage Services are (S3,EBS ,Glacier and FSX).

You can mount the AWS storage Services to your on premises (local machine (EC2) storage means HDD drive) for backup purpose.
Note :- on premises (local machine (EC2) storage means HDD drive) will have limited storage capacity but with the help of AWS storage Services are (S3,EBS ,Glacier and FSX) we can resolve our storage issue.

Practicle :-- We will mount our S3 to on premises (local machine (EC2) storage means HDD drive) for backup purpose.

Storage gateway will provide the image you download the image from storage gateway and create the virtual machine on premises (local machine (EC2) storage means HDD drive) for backup purpose.

Note :--
1. if you have so many files , photos, videos, on your on premises (local machine (EC2) storage means HDD drive)

then you will select S3 Storage Gateway .

2. suppose if you want run or execute some application then you will use EBS Storage Gateway .

3. suppose you have so many archive (compressed .zip files) on your on premises (local machine (EC2) storage means HDD drive) then you will select Glacier.

Base on your requirement , storage gateway has mainly four types of gateways how many types of gateways.

1. file gateway :-- S3 (Simple storage service) file Gateway support NFS protocol and SMB

a) NFS means :-- Network file system, it is for Linux environment

b) SMB means :-- Server Message Block, it is used for windows environment.

2. volume gateway :-- EBS (Elastic Block Storage)

Volume gateway have two volumes (Cached volume and Stored volume)
Cached volume :-- Cache meaning which is used frequently used access data will be saved there. Cache Volume is used for fast response

3. tape gateway :-- Glacier , Glacier usages (iSCSi) protocol

4. fsx gateway :-- it is only for windows operating system. fsx gateway usages the VTL (Virtual tape library)

Storage Gateway Practical :---

1. First , We will create the Storage Gateway on AWS.
2. then, we will create (Virtual machine) EC2 Instance on AWS.
3. then, we will install the Storage Gateway Appliance on (Virtual machine) EC2 Instance.
4. then, using file gatway from (Storage Gateway) , will create S3 bucket.
5. from the Storage Gateway (file gateway), we will share the file to S3 bucket. by creating connection between Storage Gateway (file gateway) and S3.
6. finally we need create connection between on premises (local machine (EC2) and Storage Gateway (file gateway)

so connections will be

(local machine (EC2)----> Storage Gateway (file gateway) -----> S3

PRACTICAL

1. Go to the AWS and search [Storage Gateway Hybrid Storage

Integration] Service and just click on "Create Gateway" button. and configure below properties.

a) Gateway name :-- FileGatewaydemo
b) select Gateway Type :- Amazon S3 File Gateway
c) Choose Volumen Type :- Cached volumes
d) select Host Platform Option :-- Amazon EC2
e) select Launch EC2 instance :-- customize your setting

f) then click on "Launch an instance" button and provide below information.
   a) Name :- StorageGatewayAppliacne
   b) Image :-- will be automatically selected (m5.xlarge)
   c) Key pair :- select key pair
   d) subnet :- select subnet
   e) select Firewall (Security groups) :- Select existing security group
   f) add volume (storage capacity) :-- 150 GB GP2 :-- It is EBS volume.
   g) then click on "Launch instance" button to create File Storage.
   f) in Security group add inbound rule for (NFS PORT 20048) or you can allow All traffic in inbound rule.

g) then select /checkout Confirm setup gateway option
h) then just click on "next" button
i) select Connection option :-ip address
j) set Ip address :-- set the Ip address of EC2 machine.
k) select Endpoint option :- publicly accessible.
l) then just click on "next" button
m) then just click on "Activate gateway" button
n) then just click on "configure" button

2. then create S3 private Bucket, go to AWS and create S3 bucket with below options

1. Bucket Name :- 6PM-FG-S3-DEMO

2. select Region :- Asia Pacific (Mumbai) ap-south-1

3. Object Ownership :- ACLs disabled

4. then just click on "create bucket" button to create bucket.

3. Now again goto Storage Gateway and select newly created "FileGatewaydemo" storage gateway and peform following actions.

1. just click on **"Create file share"** button and configure below properties in **"Create file share"**

   a) select Gateway :-- FileGatewaydemo

   b) select File share protocol :- NFS (for linux environment select NFS and for window select SMB)

c) select S3 bucket :- 6PM-FG-S3-DEMO

d) then just click on "Create file share" button.

e) in File access setting set below properties
a) just click on **"Add"** client button (to add NFS client)
b) add Allow clients :-- 0.0.0.0/0 (it is for all NFS client)

C) then just click on "next" button.

4. then just click on "Create" button to Create file share.

5. In Storage Gateway just select "File shares" option and then do following option
a) just click on newly created "File share"
b) you will able to see mount points for [Linux, windows & mac]
Example : for Linux :- MOUNT COMMAND
(mount -t nfs -o nolock,hard 172.31.24.32:PM-FG-S3-DEMO)

As per your on-premises machine , you can copy the "mount point"

6. we will create on premises (local machine (EC2)) instance.

7. do login in on premises (local machine (EC2)) instance.

8. then install NFS on (local machine (EC2)) instance, using below command.
Command :-- yum install -y nfs-utils

9. then create directory using belo command
Command :-- mkdir filesystem

10 now use "mount point" (which you seen in In Storage Gateway just select "File shares" option) for mounting , using below command

Command :-- <MOUNT COMMAND> <DIRECTORY NAME>/
Example :--
mount -t nfs -o nolock,hard 172.31.24.32:6PM-FG-S3-DEMO filesystem/

Explanation :--
1. mount command is :--
mount -t nfs -o nolock,hard 172.31.24.32:6PM-FG-S3-DEMO
2. Direcotry is :-- filesystem

3. at the end of the command put /

4. Complete command :--
mount -t nfs -o nolock,hard 172.31.24.32:6PM-FG-S3-DEMO filesystem/

11. now goto the filesystem using below command
Command :-- cd filesystem
12. run below command to see all mounted drive
Command :-- df -h

Note :- now whatever data you will store in this folder, it will automatically sotre in mounted drive means in S3 bucket (6PM-FG-S3-DEMO).

****

AWS - RDS, RDS Proxy

RDS is a service where you can (set up, configure , manage and maintain) all rdbms databases.
RDS :- Relational Database Services and RDS supports only RDBMS databases.

RDS has 6 Engine and We call them RDS DB instance.
(MySql, Oracle, MS SQL, postgres, maria db, Arrora) are RDS DB instance.

in each Database Engine can have one database or multiple databases.
means, suppose if you are using mysql Database Engine , then in this (Mysql Database Engine) can have one database or multiple databases.

There are two important topic is RDS

1. Read Replica :-- A read replica is a read-only copy of a DB instance. You can reduce the load on your primary DB instance by routing queries from your applications to the read replica. In this way, you can elastically scale out beyond the capacity constraints of a single DB instance for read-heavy database workloads.your read replicas can be in multiple regions,

All the write operation will be done on master server (Primary server) and replication will be happening to the read replica.
read replicas is used to increase the performance . you can have maximum 5 read replicas. Read replica can be in different (availability zone) and cross region also.
Read replica can use for read purpose not for write purpose.
Read replica can have their own endpoints (data base connection url).

2. Multi AZ (multi availability zone) / cluster :-- Multi AZ (multi availability zone) is used for high availability purpose and read replica used for performance purpose.

Group of db instances are called cluster, if you select [Multi AZ] option while creating database, then parallelly one more DB instance will created but this DB instance not visible for us. and AWS will take care of data replicatin from one instance to another instance using synchronously.

If any thing goes wrong with DB Instance means (net work issue, not able to connect to machine) then Fail over will happen in another [Multi AZ] DB Instance.

Note :-

1. DB OPERATION means (delete table, delete column of table, remove rows) will not cause for Fail over.

2. In Multi AZ endpoints (data base connection url) will not be change , it will be same.

3. you can enable Read Replica for Multi AZ.

RDS Feature :--

1. Database Backup :- Database backup are called snapshot.

Note :--

1. This snapshot means backup will all database of Db Engine ,and AWS will take a backup/snapshot

2. DB Level Operations :- DB level operation means perform action on single database like (create Table, Procedures, triggers, database backup) will be done by User/ customer.

3. Performance Insights :-- Performance Insights is one kind of dashboard and from there, we can monitor the performance of database.

4. Retention period :-- Retention period means keep the backup of datbase for how many days, normally you can set RDS backup retention period to between 0 and 35 days. Backup retention period. You can set the backup retention period when you create a DB instance. If you don't set the backup retention period, the default backup retention period is one day if you create the DB instance using the Amazon RDS API or the AWS CLI. The default backup retention period is seven days if you create the DB instance using the console

5. you can export the Database backup /snapshot into S3 also and you can restore it also.

6. Database backup /snapshot can be copied from one region to another region

7. Database backup /snapshot can be shared from account to another AWS account.

8. you can **scale up** the DB INSTANCE , but for scale up , you need to stop DB INSTANCE, so you will face down time issue.

scale up :- scale up means increase the database storage capacity, suppose , while creating database, if you set 100 GB HDD

storage for database but after some days, if you notice there is no space available , then you will increase the HDD capacity , that is called scale up. but the problem is you will face down time issue, so while creating database, you can do the auto scaling option on Database Storage level.

9. AWS proprietary engine is Arrora and Aurora is compatible with your MySQL and postgress and it 5 time faster than MySQL and postgress.

10. in Nromal Database , you can create maximum 5 Read replicas but in Aurora you can create 15 Read replicas.

11. this Aurora is giving you a concept of load balancer that meaning load balancer for read replicas. so your application will be connected to the load balancer and load balancer will distribute to among that 15 servers. means for read replica Aurora provides load balancer.

12. Aurora supports database engine and also Aurora is serverless

RDX PROXY :-- RDS Proxy is a fully-managed, highly available, and easy-to-use database proxy feature of Amazon RDS that enables your applications to: 1) improve scalability by pooling and sharing database connections; 2) improve availability by reducing database failover times by up to 66% and preserving application connections.

so many applications will be connected to your DB engine correct that mean there will be so many connections will be there and it will increase the load on database. To avoid that load, AWS RDS have RDS Proxy and you will connect to RDS Proxy instead of Database. it will collect all the connections and it will give it will give less number of connections to the DB instance and also removes the all unwanted connections and then it will connect to database.

Note :-- RDS PROXY is fully managed database proxy for RDS , which allows apps to pool an share the DB connections establised with the database improving database efficiency by reducing the stress on the database resources like (CPU,RAM) and minimize open connections and timeouts. RDS proxy is serverless and useful for autoscaling , High availability. RDS proxy support (Mysql, Postgres, MariaDB, MS SQL, and Aurora).

Secret manager is a AWS service where you can store all secrets like (username , passwords). and RDS proxy can read these secret. RDS Proxy is private and not publically accessible. It must be access through (VPC).

VERY IMP :-- RDS Proxy will allow your applications to pool and share the DB connections established with the DB instead of having single app connect to RDS Instance, they will be instead connecting to the proxy and proxy will pool these connections together into less connections to the RDS DB instance.

==========

RDS PRACTICLE :-- RDS is Regional service

1. Go to the Amazon RDS dashboard and just click on "Create database" button.
2. Choose a database creation method (Standard create / Easy create) :-- Standard create
3. select database :-- postgres
4. select PostgreSQL Engine Version :-- LATEST
5. select Database template (Production / Dev-test / Free tier)
   Note :-- if you select (Production template) then you will able to select below [Availability and durability]

   Availability and durability

   1. Multi-AZ DB Cluster NEW :-- Group of db instances are called cluster, so here one primary database and two replica will be created.

   2. Multi-AZ DB instance :-- Multi-AZ DB instance meaning a parallel another DB instance will get created.

   3. Single DB instance :-- will have only single db instance.

6. select Multi-AZ DB instance

7. set DB instance identifier name :-- postgresinstance

8. master user name :- postgres

9. master password :- postgres

10. confirm password :- postgres

11. select instance configuration (standard classes, Memory optimized classes, Burstable classes) As per the standard practice you can go with (standard classes, Memory optimized classes)
    so select :------ **Memory optimized classes**

12. Storage type :-- it is automatically selected

13. Allocate storage :-- 400 GIB

14. set provisioned IOPS :-- 3000 Default value

15. enable auto storage autoscaling in [STORAGE AUTO SCALING SECTION]

16. Maximum storage thresold :-- 1000 Default value

17. in Additional configuration set database name : demodb

18. then click on "create database" button.

VERY IMP :-- disaster recovery :-- means suppose your databse backup is in one region for example (Mumbai region) and due to some technical issue database backup get destroyed , to avoid this issue , we go with disaster recovery means your database backup will be in more than one regions.

What is Retention period :-- How many days backup you need , that is called Retention period.

==========

Elastic Cache, Redis

Elastic Cache :-- It is in memory caching service and it will give you best performance.
Cache :-- All frequentyly accessed data is stored at this place.

suppose, your customers are accessing data frequently from database ,
so it is very costly to trigger a data base to get (frequent data), because it will un necessary create the load on database,
instead of triggering the database to get (frequent data) , we can put this (frequent data) in Elastic Cache and can increase the performance and reduce the load on database.

Benefit of Elastic Cache and work flow

1 Application will hit the request for data

2.fist request will come to Elastic Cache and it will check required/ request data is cache or not

3.if required/ request data will be available in Elastic Cache ,the Elastic Cache will return to Application as response.

4.and if required/ request data will not be available in Elastic Cache , the request goes to RDS and reterive data from RDS

5.after getting data from RDS , Application will write this data in to Elastic Cache.

Note :-

1. Elastic Cache is mostly used for read purpose and it increase the performance of application
2. Elastic Cache can handle the session data.
3. Elastic Cache support encryption (In-Transit and Data at Rest)
4. Elastic Cache is regional

Elastic Cache support two engine (memcached & redis)

redis :-

1. redis supports high availability.
2. redis support backups and failover.

3. Data is Persistent means permanent in redis.

so you stop and start the redis engine, data will never lost.

redis architecture :-- In Redis, we have two things
Cluster Mode Enabled :-- here you will get almost 500 shards.

1 shard means (1 Primary Node and 5 Replica Nodes) * 500

Cluster Mode Disabled :-- here you will get 1 shard means (1 Primary Node and 5 Replica Nodes)

Note :-

In redis Cluster meaning :-- Collection of shards and
Shard means :-- Shard is Collection of Nodes/ server and each shard has 6 nodes (1 Primary Node and 5 Replica Nodes)

==========

memcached :--

1. memcached not supports high availability
2. memcached not support backups and failover.
3. Data is not Persistent means permanent in memcached.
so you stop and start the memcached engine, data will lost.

Note :-- memcached is cheaper than redis.

Developer Strategies :-

1. Lazy loading :-- Load your data, when it required, We don't insert data parallelly in RDS database and Elastic Cache. Data will insert first in RDS and when Application will fetch , it will fetch from RDS and after it will insert in Elastic Cache.

2. Write Through :-- Whenever Applicaiton will insert data in RDS database parallelly data will be store in Elastic Cache , so data will never lost , it will always available.

==========

SERVICE QUOTAS :-- AWS provide "Service Quotas" service, using this service , you can find soft limit of AWS resources service wise. if required you can increase the soft limit by creating a ticket with AWS.

For example :-- you will get 5 default Elastic IPS, so 5 is soft limit of Elastic IP, if you want more Elastic IP , then you can create ticket with AWS and increase the Elastic IP.

Elastic Cache, Redis practicle

1. Go to AWS and search for "Amazon ElastiCache" , you will get "Amazon ElastiCache" dashboard.
2. you can create Redis Clusters or memcached Clusters

==========

50 DynamoDB, Indexes

DynamoDB is no SQL service in AWS (no SQL means Not Only SQL)
NoSQL databases are non-relational databases (no fixed columns) and are distributed (horizontal scaling)
NoSQL databases do not support joins
NoSQL databases do not perform aggregations like (sum, min, max)
NoSQL databases scale horizontally
NoSQL databases do not have fixed SCHEMA.
You can't write complex query in NoSQL databases.

SQL VS NOSQL databases

SQL :--

1. SQL is generally used in RDBMS
2. Structured data can be stored in tables
3. The Schema are static
4. Schemas are rigid and bound to relationships
5. Helpful to design complex queries
6. here , we call tables, rows and columns
7. Mysql, Oracle, Sqlite, Postgres and MS-SQL are good example of SQL Databases.

NOSQL :--

1. NOSQL is used for Non-Relational Database system.
2. Using JSON data, unstructured data can be stored
3. The Schemas are dynamic
4. Schemas are non-regid, they are flexible
5. No interface to prepare complex queries
6. Here we call Collections and collections has documents
7. MongoDB, BigTable, Redis,RavenDB,Cassandra, Hbase, Neo4j and CouchDB are good example of NOSQL Databases.

DynamoDB vs RDS

1. DynamoDB offers "push button" scaling, meaning that you can scale your database on the fly, without any down time.

2. DynamoDB is Serverless , in DynamoDB there is no server, in DynamoDB, you directly creates the tables.

3. RDS is not so easy and you usually have to use bigger instance size or to add a read replica.

Amazon DynamoDB is a serverless and in DynamoDB, tables, items and attributes are the core components.

A table is a collection of data.
An item is a group of attributes will act a row
Attributes in DynamoDB are similar in many ways to fields or columns

In DynamoDB there are two Consistency models to read data

1. **Eventual Consistent Reads** (ECR) : Eventual meaning slowly, This is default reading way to read data of DynamoDB.

2. **Strongly Consistent Reads** (SCR) : Strongly meaning immediately/ quick , it is used to read data quickly.

In DynamoDB you no need to design the whole table , it is flexible, While creating table in DynamoDB,
you need to create only one column and that column should be primary key, means in DynamoDB, initally
you can create a table with single column and that column should be primary key.

In DynamoDB primary key is mandatory while creating table and In DynamoDB primary key is called partition key

What is composite key in DynamoDB ??

composite key is combination of Primary key and sort key in DynamoDB

if you want duplicate value in primary, then you need to create sort key in DynamoDB
and sort key will have unique value. sort key is optional in DynamoDB

Note :--

1. We generally called primary key is harsh Attribute.
2. We generally called sort key is Range Attribute.

What is DynamoDB Streams ?? : DynamoDB Streams is nothing but change log, so whatever you change in DynamoDB that will capture here , means if you

update any item (row) in DynamoDB , it will take backup of previous Item (row) information and will put it in DynamoDB Streams (change log). so will able to verify , what was the previous data in previous row and what is the newly updated data.

Note :-- if you add, update, delete any such operation, it will maintain the **[added, updated, deleted]** history in DynamoDB Streams (change log).

DynamoDB indexes :--
Indexes helps to increase the performance of the table and reterving data. base on indexes, search record should fast.

in DynamoDB there are two types of indexes.

1. Local Secondary Index (LSI) :-- using composite key (primary key [partition key] + sort key) we can create the index
LSI can be created only at the time of creating the table later LSI cannot be modified cannot be deleted.

2. Global Secondary Index (GSI) :-- using (Any column as Primary key + Any column as sort key) we can create index.
GSI can be created anytime it can be created, modified , deleted anytime.

What is Provisioned Capacity Units in DynamoDB :--

With provisioned capacity mode, you specify the number of data reads and writes per second that you require for your application.
You can use auto scaling to automatically adjust your table's capacity based on the specified utilization rate to ensure application performance while reducing costs.

There are two types of Capacity Units
1. Read Capacity Units (RCU) :-- by default (1 RCU= 5.2 MILLONS READS)
2. Write Capacity Units (WCU) :-- by default (1 WCU= 2.5 MILLONS WRITES)

DynamoDB performance depend upon (RCU) and (WCU).

Provisioned Capacity Units Formula :--

For **Eventual Consistent Reads** (ECR),
    1 Read Capacity Units (RCU) = 2 reads per second of 4KB size
For **Strongly Consistent Reads** (SCR),
    1 Read Capacity Units (RCU) = 1 reads per second of 4KB size

For 1 Write Capacity Units (WCU), = 1 write per second of 4KB size

Provisioned Capacity Units Formula Explanation :--

Read Capacity Requirements :--- If you have a table and you want to read 100 items per second with (SCR) **strongly consistent reads**

and your items are 8KB in size, you would calculate the required provisioned capacity as follows.

8KB/4KB=2 capacity units.

2 read capacity unitis per item x 100 reads per second = 200/1 = 200 read capacity units

Note :-- Eventual consistend Reads would require 200/2 = 100 read capacity units.

Write Capacity Requirements :--- If you have a table and you want to write 50 items per second and your items are 4KB in size,
you would calculate the required provisioned capacity as follows.

4KB/1KB= 4 capacity units

4 write capacity units per item x 50 writes per second = 200/1 = 200 write capacity units.

EXAMPLE :

Read Capacity Requirements

Question 1 : 10 strongly consistent reads per seconds of 4 KB each ?
Answer :-- We need 10 x 4KB /4KB= 10 RCU (For SCR , 1 RCU= 1 read per second for 4 KB size).

Question 2 : 16 Eventually consistent reads (ECR) per seconds of 12 KB each ?
Answer :-- We need 16/2 x 12 KB / 4kb =24 RCU (For SCR , 1 RCU= 2 read per second for 4 KB size).

Question 3 : 10 Strongly consistent reads per seconds of 6 KB each ?
Answer :-- We need 10 x 8 KB / 4kb =20 RCU (For SCR , 1 RCU= 2 read per second for 4 KB size, 6 round off to 8KB because ech is 4KB and next size is 8KB).

NOTE :-- in DynamoDB, we can SET TTL (Time to live) for each item. TTL (Time to live) means we can schedule to delete particular ITEM.
mean you can se the expire of the particular row that is call TTL **(Time to live).**

Example :-- suppose you 1 month subscriptions for netflix after 1 month your subscription will automatically expire, how it is possible,
it is possible due to TTL (Time to live) setting.

==========

Cloud Front :-- Amazon CloudFront is a **content delivery network** operated by Amazon Web Services.

The content delivery network was created to provide a globally-distributed network of proxy servers to cache content, such as web videos or other bulky media, more locally to consumers, to improve access speed for downloading the content.

Amazon CloudFront is a web service that speeds up distribution of your static and dynamic web content, such as **.html, .css, .js, and image files,** to your users. CloudFront delivers your content through a worldwide network of data centers called edge locations.

CloudFront has Edge loation and Every region have edge location and Edge location will cache the application. in cloud front you create **"Distribution"**, so we just need to go in cloud front service, and need to to create **"Distribution".**

CloudFront have TTL **(Time to leave)** for cache , means if you set (TTL) for 12 hours, then data will be cache for 12 hours. and after 12 hours, data will be updated in cache.

Example :-- suppose , You advertise on a website that an iPhone is sold for ten thousand rupees so that information will be cached in Edge location for next 12 hours (because you have set TTL for 12 hours). before this 12 hours , if change the price of iphone , that updated price will never reflect immediately on website page it will reflect after 12 hours. but if you want to immediately reflect this updated price of iphone on website page, then you just go in Cloud front and do **"invalidate valid cash"** so all previous cache will be removed and new information will be cached for next 12 hours. now at this time user will able to see updated price of iphone.

Note :- Cloud front cache the static and dynamic data. Cloud front has edge locations and edge locations are connected with CDN.

When a user requests content that you're serving with CloudFront, the request is routed to the edge location that provides the lowest latency (time delay), so that content is delivered with the best possible performance.

1) If the content is already in the edge location with the lowest latency, CloudFront delivers it immediately.

2) If the content is not in that edge location, CloudFront retrieves it from an origin that you've defined—such as an Amazon S3 bucket,a MediaPackage channel, or an HTTP server (for example, a web server) that you have identified as the source for the definitive version of your content.

Cloud Front practicle :--

First create S3 BUCKET

1. set bucket name :-- swami-master-video-storage
2. select "Block all public access" to disable public access
3. enable "Bucket Versioning"
4. set Default encryption as :-- Server-side encryption with Amazon S3 managed keys (SSE-S3)
5. enable Bucket Key
6. then click on "Create Bucket" button.

Second now configure CLOUDFRONT

search "Amazon CloudFront" and just click on "Create a CloudFront distribution" button

1. First, we need to configure "Origin access" , so just open left hand sided "cloudFront" panel menu and select "Origin access" option from "security" menu and click on "Create control setting" button and set below properties.

   a) Name :-- swami-video-streaming-OriginAccessControl
   b) set Signing behavior as "Sign requests (recommended)"
   c) select Origin Type "S3"
   d) then just click on "create" button to create "Create control setting"

2. Then select "Distributions" option from "security" menu and click on "Create distribution" button and set below properties.

   a) **Origin domain :-** here select newly created S3 BUCKET NAME (swami-master-video-storage.s3.ap-south-1.amazonaws.com)

   b) **Origin access :--** select "Origin access control settings (recommended)" option and select newly created Origin access control "swami-video-streaming-OriginAccessControl"

   c) you will get below notification You must update the S3 bucket policy CloudFront will provide you with the policy statement after creating the distribution.

   d) set Enable Origin Shield :- No

   e) set Path pattern :- Default (*)

   f) set Compress objects automatically :- yes

g) set Viewer protocol policy :-- Redirect HTTP to HTTPS

h) set Allowed HTTP methods :-- GET, HEAD

i) set Restrict viewer access :-- No

k) set Cache key and origin requests :-- Cache policy and origin request policy (recommended)

l) set Cache policy :- CachingOptimized (Recommended for S3)

m) set Web Application Firewall (WAF) :-- Enable security protections.

n) and just click on "Create distribution" button to create distribution.

Note :-- you will get below notification , you just click on **"Copy policy"** button and open S3 and update S3 bucket policy.

The S3 bucket policy needs to be updated Complete distribution configuration by allowing read access to CloudFront origin access control in your policy statement.Go to S3 bucket permissions to update policy.

3. update S3 bucket policy. for that go to S3 and select created bucket "swami-master-video-storage".

a) then go to "Permissions" option

b) then select "Bucket policy" and just click on "Edit" button

c) pase copied policy which you copied while creating "distribution".

d) and then just click on "Save changes" button.

e) then again go to CloudFront > Distributions option and reload the page.

4. upload video on S3 bucket , for that go to S3 and select created bucket "swami-master-video-storage".
   a) now upload video file in bucket Example (abc.mp4) .
   b) then open the uploaded file and copy the key name (abc.mp4).

5. now again go to **CloudFront > Distributions option** and copy the **"Domain name"** of created **"Distribution".**

Example :-- d33776ej7j1v9z.cloudfront.net

6. open this url on browser :--
   https://d33776ej7j1v9z.cloudfront.net/abc.mp4

7. How to Disable CloudFront Distribution / How to Delete CloudFront Distribution ?
   Refer below link for better understanding.
https://www.youtube.com/watch?v=zfj9nB1e20k&pp=ygUoaG93IHRvIGRlbGV0ZSBkaXN0cmlidXRpb24gaW4gY2xvdWRmcm9udA%3D%3D

****

AWS ROUTE 53

AWS ROUTE 53 :-- Route 53 is Global service

Amazon Route 53 is a scalable and highly available Domain Name System service. Route 53 is a DNS (Domain Name System) service in AWS.

DNS (Domain Name System) **PORT number is 53,** so **it is called ROUTE 53.**

Amazon Route 53 service that works by translating human-readable domain names into IP addresses that computers use to communicate with each other.

It manages the routing of internet traffic to the appropriate resources based on the domain names entered by users.

Amazon Route 53 - Routing Policies

Simple routing policy – Use for a single resource that performs a certain role for your domain, for example, a web server that provides content to the example.com site.

Failover routing policy – Use when you want to configure active-passive failover.

Route 53 has three main functions: (domain registration, DNS routing, and health checking, Routing policies)

Note :--
DNS is all about records, In Route 53 the first thing that you have to create is hosted Zone and hosted zone contains records, so
hostage Zone hostage zone is a container of Records and what is the name of the "hostage Zone" that we create is domain name.
hosted zone is noting but domain name (hosted zone and domain name both are same).

There are two types of hosted zone

1. **public hosted zone :--** It is publically available and will work on internet.

2. **private hosted zone :--** It is private and will work on VPC.

What is record :- record is nothing but routing to destination using (hosted zone/domain name).

Example :-- book.com is your (hosted zone/domain name) and whenever user will hit this url, then internally book.com will give call to ELB **(Elastic Load balancer ->> Application Load balancer)** and wil fetch your web application.

Reason :-- (**Elastic Load balancer ->> Application Load balancer**) will have nasty URL , so it is difficult to read and give to end customer so we use the ROUTE 53 and create the record and within record we create the relation of domain name with (ELB) url.

Example :-- user hit [DOMAIN NAME] book.com --------> and book.com will call (ELB) url and ELB will execute the WEB Application.

Note:- Whever you create public hosted zone Two records will get created automatically. (NS and SOA) records**(NS and SOA)** records are automatically created and handled by AWS.

1. NS Record :-- NS records tell the Internet where to go to find out a domain's IP address.NS stands for 'nameserver,' and the
name server record indicates which DNS server is authoritative for that domain.

2. SOA Record :-- Start of authority, used to designate the primary name server and administrator responsible for a zone.

Important :-- You can purchase a domain in ROUTE 53 and whenever you purchase a domain in ROUTE 53, automatically hosted Zone will be created and (NS and SOA) records will be automatically created and handled by AWS.

Whenever you purchase a domain from outside area like (Godady) then you need to update (NS) records in Godady
so when any reaques will come in Godady then Godady will route the request to ROUTE 53.

==========

ROUTE 53 Records :-- Records is realated to DNS. there are some below important records

1. A Record :--It is (URL to IPV4)

A record maps a domain to the physical IP address of the computer hosting that domain. Internet traffic uses the A record to find the computer hosting your domain's DNS settings. The value of an A record is always an IP address, and multiple A records can be configured for one domain name.

Example :-- suppose we have (https://swami.com) website, and whenever we hit the request to this (https://swami.com) website,

1. first request will go to ROUTE 53
2. then from ROUTE 53 request will go to Hosted zone (swami.com)
3. then from Hosted zone (swami.com) request will go to records and in this record will have routing information
4. records content [swami.com ->> EC2 IP address IPV4].

Note :- if you have requirement that bind IP address (IPV4) with URL in record , then go for A record.

2. AAAA Record :-- It is (URL to IPV6) , We don't use it because it is related to IPV6.

A and AAAA records are equally important when it comes to resolving DNS. The difference lies in that A records is used to resolve a hostname which corresponds to an IPv4 address, while AAAA records are used to resolve a domain name which corresponds to an IPv6 address.

3. CNAME Record :-- It is (URL to URL)

A CNAME record can be created for your zone apex. An Amazon Route 53 CNAME record can point to any DNS record hosted anywhere.

CNAME means (Canonical Name) and A Canonical Name or CNAME record is a type of DNS record that maps an alias name to a true or canonical domain name.

CNAME records are typically used to map a subdomain such as www or mail to the domain hosting that subdomain's content.

Example :-- suppose we have (https://swami.com) website, and whenever we hit the request to this (https://swami.com) website,

1. first request will go to ROUTE 53
2. then from ROUTE 53 request will go to Hosted zone (swami.com)
3. then from Hosted zone (swami.com) request will go to records and in this record will have routing information
4. records content [swami.com ->> ELB (domain name) nasty URL].

Note :- if you have requirement that ELB (domain name) nasty URL with URL in record , then go for CNAME record.

Note :-- CNAME Record free

4. ALIAS Record :-- It is (URL to Any Resource) ALIAS means nickname.

The ALIAS record is similar to a CNAME record, which is used to point subdomains to a hostname. The CNAME record only can be used for subdomains, **so the ALIAS record fills this gap.**

Example :-- suppose we have (https://swami.com) website, and whenever we hit the request to this (https://swami.com) website,

1. first request will go to ROUTE 53
2. then from ROUTE 53 request will go to Hosted zone (swami.com)
3. then from Hosted zone (swami.com) request will go to records and in this record will have routing information
4. records content [swami.com ->> ELB (domain name) nasty URL]. or
5. records content [swami.com ->> EC2 IP address IPV4].

Note :- if you have requirement that ELB (domain name) nasty URL with URL in record or bind IP address (IPV4) with URL , then go for ALIAS record.

Note :-- ALIAS Record free

5. MX Record :-- It is for (Emails)

A mail exchanger record (MX record) is a configuration that specifies which mail servers can accept email that's sent to your domain.

A DNS 'mail exchange' (MX) record directs email to a mail server. The MX record indicates how email messages should be routed in accordance with the Simple Mail Transfer Protocol (SMTP, the standard protocol for all email). Like CNAME records, an MX record must always point to another domain.

Note :- In Real World we commonly use (A Record + ALIAS Record) means combination of (A Record + ALIAS Record) .

What is https://swami.com , so (swami.com) is nothing but domain , so we called (main domain or naked domain or zone apex record).

suppose , if you have (https://admin.swami.com or https://hr.swami.com) so it is called sub domains.
because, (admin, hr) will be sub domain of main domain (swami.com).

VERY IMPORTANT :--

1. CNAME Record are billable and ALIAS Record free and for naked domains you cannot use CNAME.
2. you can not use nake domain with CNAME, instead use ALIAS.
3. for sub domains you can use CNAME.

so always choose ALIAS over the CNAME.

==========

Routing policies :--

Example :-- suppose we have (https://swami.com) website, and whenever we hit the request to this (https://swami.com) website,

1. first request will go to ROUTE 53
2. then from ROUTE 53 request will go to Hosted zone (swami.com)
3. then from Hosted zone (swami.com) request will go to records and in this record will have routing information

4. records content [swami.com ->> ELB (domain name) nasty URL].
so the final request goes to ELB and your application is getting monitored through the load balancer and assume , if your instance destination gets down, so application level it will be manage using (Health check). but we are not doing any Health check for domain (https://swami.com) level, means you implemented (SIMPLY ROUTING POLICY in RECORD) and there is no health check so, in this situation , we can with (FIRST & SECOND) solution

FIRST SOLUTION :--- >

we will going to replicate the applicaiton in different regions add "FAIL OVER ROUTING POLICY in RECORD" .

Example :-- suppose , if you have created applicaiton in (MUMBAI REGION) and in mumbai region , have created ELB and will have instance destination.

so we will create the same copy of applicaiton in (ANOTHER REGION) and will add this (ANOTHER REGION) as "FAIL OVER ROUTING POLICY in RECORD"

so we need to create 2 records

1. SIMPLY ROUTING POLICY :-- this is for MUMBAI (Default PRIMARY region)

2. FAIL OVER ROUTING POLICY :-- this is for ANOTHER (SECONDARY region)

SO benefit is , if we found the (Default PRIMARY region) (https://swami.com) website gets down, we will get the response from (SECONDARY region).

SECOND SOLUTION :--- >

Or another solution is , we will going to create "Maintaince.html" and will put it in S3 bucket,
and will create static website and will add it as "FAIL OVER ROUTING POLICY in RECORD".

so we need to create 2 records

1. SIMPLY ROUTING POLICY :-- this is for MUMBAI (Default PRIMARY region)

2. FAIL OVER ROUTING POLICY :-- this is for S3 STATIC WEB SITE ADDRESS (SECONDARY region)

SO benefit is , if we found the (Default PRIMARY region) (https://swami.com) website gets down, we will get the response from (S3 STATIC WEB SITE).

==========

Reference URL :-- https://jayendrapatil.com/aws-route-53-routing-policy/

SIMPLE ROUTING POLICY

1. Simple routing policy is a simple round-robin policy and can be applied when there is a single resource doing the function for the domain e.g. web server that serves content for the website.

2. Simple routing helps configure standard DNS records, with no special Route 53 routing such as weighted or latency.

3. Route 53 responds to the DNS queries based on the values in the resource record set e.g. IP address in an A record.

4. Simple routing does not allow the creation of multiple records with the same name and type, but multiple values can be specified in the same record, such as multiple IP addresses.

5. Route 53 displays all the values to resolve it recursively in random order and the resolver displays the values for the client. The client then chooses a value and resends the query.

6. Simple routing policy does not support health checks, so the record would be returned to the client even if it is unhealthy.

With Alias record enabled, only one AWS resource or one record can be specified in the current hosted zone.

==========

GEOLOCATION BASED ROUTING POLICY :-- Geolocation routing enables customers to choose the resources that serve their traffic based on the geographic location of their users. Customers can use this feature to localize content or restrict distribution of content to only the locations for which they have distribution rights.

Route 53 will automatically detect from which country requst is coming and base on the request based IP address it will be redirect to the appropriate record .

Example :-

you have created (https://book.com) website with many laguage support like (HINDI (MUMBAI) , ENGLISH (CANADA), IRISH (IRELAND), JAPANESE (TOKEYO))

1. you deploy it in MUMBAI region with HINDI language
2. you deploy it in CANADA region with ENGLISH language
3. you deploy it in IRELAND region with IRISH language
3. you deploy it in TOKEYO region with JAPANESE language

so in ROUTE 53, you will create 4 RECORDS for four countries,

RECORD Example :--

1. https://book.com ---> MUMBAI ELB / IPV4
2. https://book.com ---> CANADA ELB / IPV4
3. https://book.com ---> IRELAND ELB / IPV4
4. https://book.com ---> TOKEYO ELB / IPV4

so whenever anybody hit the (https://book.com) website from any country ROUTE 53 will automatically detect the (Geo location) and route the request to respective RECORD.

==========

LATENCY BASED ROUTING POLICY :-- Requests get connected based on the latency doesn't matter about the regions.

1. Latency-based Routing Policy helps respond to the DNS query based on which data center gives the user the lowest network latency.

2. Latency-based routing policy can be used when there are multiple resources performing the same function and Route 53 needs to be configured to respond to the DNS queries with the resources that provide the fastest response with the lowest latency.

3. A latency resource record set can be created for the EC2 resource in each region that hosts the application. When Route 53 receives a query for the corresponding domain, it selects the latency resource record set for the EC2 region that gives the user the lowest latency.

Route 53 then responds with the value associated with that resource record set for e.g., you might have web servers for example.com in the EC2 data centers in Ireland and in Tokyo. When a user browses example.com from Singapore, Route 53 will pick up the data center (Tokyo) which has the lowest latency from the user’s location.

4. Latency between hosts on the Internet can change over time as a result of changes in network connectivity and routing.

Latency-based routing is based on latency measurements performed over a period of time, and the measurements reflect these changes for e.g. if the latency from the user in Singapore to Ireland improves, the user can be routed to Ireland.

5. Latency-based routing cannot guarantee users from the same geographic will be served from the same location for any compliance reason Latency resource record sets can be created using any record type that Route 53 supports except NS or SOA.

6. Latency-based routing policy supports health checks.

==========

MULTI-VALUE BASED ROUTING POLICY :-- It is same as SIMPLE ROUTING POLICY but MULTI-VALUE POLICY has health checks.

1. Multivalue routing helps return multiple values, e.g. IP addresses for the web servers, in response to DNS queries.

2. Multivalue routing also helps check the health of each resource, so only the values for healthy resources are returned.

3. Route 53 responds to DNS queries with up to eight healthy records and gives different answers to different DNS resolvers.

4. Multivalue answer routing is not a substitute for a load balancer, but the ability to return multiple health-checkable IP addresses
is a way to use DNS to improve availability and load balancing.

5. To route traffic approximately randomly to multiple resources, such as web servers, one multivalue answer record can be created
for each resource and, optionally, associate a Route 53 health check with each record. If a web server becomes unavailable after
the resolver caches a response, client software can try another IP address in the response.

==========

WEIGHTED ROUTING POLICY :--

1. Weighted routing policy helps route traffic to different resources in specified proportions (weights) e.g., 75% to one server and 25% to the other during a pilot release.

2. Weights can be assigned between any number from 0 to 255 inclusive.

3. Weighted routing policy can be applied when there are multiple resources that perform the same function e.g., webservers serving the same site

4. Weighted resource record sets allow associating multiple resources with a single DNS name.

5. Weighted routing policy use cases include
   a) load balancing between regions
   b) A/B testing and piloting new versions of software

6. To create a group of weighted resource record sets, two or more resource record sets can be created that has the same combination of DNS name and type, and each resource record set is assigned a unique identifier and a relative weight.

7. When processing a DNS query, Route 53 searches for a resource record set or a group of resource record sets that have the specified name and type.

Route 53 selects one from the group. The probability of any one resource record set being selected depends on its weight as a proportion of the total weight for all resource record sets in the group for e.g., suppose www.example.com has three resource record sets with weights of 1 (20%), 1 (20%), and 3 (60%)(sum = 5).

On average, Route 53 selects each of the first two resource records sets one-fifth of the time and returns the third resource record set three-fifths of the time.

8. Weighted routing policy supports health checks.

==========

Route 53 Traffic Flow

1. Route 53 Traffic Flow helps easily manage traffic globally through a variety of routing types combined with DNS Failover in order to enable a variety of low-latency, fault-tolerant architectures.

2. Traffic Flow provides a simple visual editor, to easily manage how the end-users are routed to the application’s endpoints – whether in a single AWS region or distributed around the globe.

3. Traffic Flow routes traffic based on multiple criteria, such as endpoint health, geographic location, and latency.

4. Traffic Flow’s versioning feature maintains a history of changes to the traffic policies to allow easy rollback to the previous version.

==========

Route 53 PRACTICLE :--

1 . CREATE 2 EC2 instances in Mumbai Region
     a) first instance in (Availbiity zone a) ap-south-1a
     b) second instance in (Availbiity zone b) ap-south-1b

Note :- while creating EC2 instances add below data script in (user data)

```
#!/bin/bash
apt get install httpd -y
service httpd start
chkconfig httpd on
mkdir /var/www/html
echo 'Hey!! This is WEBSITE on EC2!' > /var/www/html/index.html
```

2. in Security Group add below entries in (INBOUND Rules)

Type	**Protocol**	**Port range**	**Source**	
HTTP	TCP	80	MY IP	
RDP	TCP	3389	CUSTOM	PUBLIC IP
SSH	TCP	22	CUSTOM	PUBLIC IP
ALL TRAFFIC	ALL	ALL	CUSTOM	SECURITY GROUP

3. Create target group
     Choose a target type :- instances
     Target group name :- MYTG
     IP address type :- IPV4
     Health check path :- /index.html

then click on **"Next"** button add select "2 Instances" and Register it as targets. then cick on "Create target Group" button, and create target.

4. CREATE LOAD BALANCER (Application Load balancer)

STEP 1 (Basic Configuration)
     Name :- myelb
     Ip Address Type :- ipv4
     Load Balancer protocol :- HTTP
     Load Balancer prot :- 80

Availibility Zones :-- select (ap-south-1a, ap-south-1b)

Listeners and routing :-- select Target group
[newly created target group :--**MYTG**]

then cick on **"Create load balancer"** button, to create load balancer.

5. CREATE RECORD IN ROUTE 53

Note :-- you need to purchase domain name , after that you will perform below task.

1. Go to Route 53 and under the "Hosted zone" create record

     Section Quick create record
          1. Record name   :- test
          2. Record type   :- A - Routes traffic to an IPv4 address
                                    and some AWS resources
     Note :-- Enable Alias and configure below properties

3. Route Traffic to :-- select Alias to Application
                        and Classic load balancer

4. Choose Region   :-- Asia pacfic (Mumbai)

5. Choose Load Balancer   :-- select newly created Load balancer (myelb)

6. Routing policy   :-- here , you can select
                          (Simple routing, Failover) routing.

7. finally just create on "create record" button.

****

AWS – VPC

AWS – VPC :-- VPC means virtual private cloud and everything whatever the infrastructure that we created in AWS everything should be within the VPC. we can say VPC is Virtual data center of the cloud. VPC is reginal Service, and we can create maximum 5 VPC per region.

==========

VPC Order :--

1. VPC (Virtual Private Cloud)

2. IG (Internet Gateway) **:-** Internet Gate Way provides the internet access to VPC. We create the (Internet Gateway) and we attach it to VPC.

Note : one internet gateway can be attached to only one VPC.

3. Public and Private Subnets :--

Public subnet :-- Public subnet means Which Expose to the Intenet. Anyone can connect from outside to AWS Resources like (EC2) using Public subnet.

A public subnet is a subnet that is associated with a route table that has a route to an Internet gateway.
This connects the VPC to the Internet and to other AWS services.
All your public subnet traffic is routed to internet gateway

Private subnet :-- Private subnet means Which is not Expose to the Intenet. Private Subnet. A private subnet is a subnet that is associated with a route table that doesn't have a route to an internet gateway.

Private subnet does not have a direct route to an internet gateway. A private subnet is used for instances that do not need to be directly reachable from the internet. For the best security, it's important to keep backend instances and databases private, so those would go in a private subnet.
Private subnet get the internet through (NAT) Gateway.private subnet traffic is routed to Natgateway.

4. NAT (Network Address Translation Gateway) :-- NAT Gateway is a highly available AWS managed service that makes it easy to connect to the Internet from instances within a private subnet in an Amazon Virtual Private Cloud (Amazon VPC). NAT will convert private IP to Public IP.

you can create only one NAT Gateway in each AZ (availability zone) within a VPC.

Note :-

- 1. you need to create (NAT) Gateway inside the public Subnet not in private subnet.
- 2. In side the NAT, you need to create Elastic IP address and assign it to NAT Gateway.
- 3. NAT Gateway is required , when you create private subnet and , you don't want to access private network base resources to out side world but you want internet for private network base resource, then you will create NAT Gateway in (PUBLIC SUBNET) and will use it in with private route table with (PRIVATE SUBNET).

5. **Router with Routing Table :--** There are (pubic &) Routing tables.

public routing table :-- here all the traffic is routed to internet gateway and public subet is associated to public routing table.

private routing table :-- here all the traffic is routed to NAT gateway and private subet is associated to private routing table.
but you need to create (NAT) Gateway inside the public Subnet not in private subnet.

6. Security Group :-- Security Groups act as virtual firewalls.
A security group controls the traffic that is allowed to reach and leave the resources that it is associated with. For example, after you

associate a security group with an EC2 instance, it controls the inbound and outbound traffic for the instance. When you create a VPC, it comes with a default security group.

VPC Creation steps :--

1. Create VPC
2. Create Internate Gateway (IG) and attach it to the VPC
3. Create Public and Private Subnets.
4. Create Network Address Translation Gateway (NAT) Gateway inside the public subnect
5. Create Public Routing table and attach it to public subnet and all traffic routed to Internate Gateway (IG).
6. Create Private Routing table and attach it to private subnet and all traffic routed to Network Address Translation Gateway (NAT) Gateway.
7. Create Security Group and create (Inbound and Outbound) rules in Security Group.

Note :-- Whenever you create the EC2 instance in public subnet ,then it is called Bastion / jump server.

==========

What is vpc endpoint :-- Without public internet , user can access the AWS Services using vpc endpoint.
We don't need to public internet , we would need to create a VPC endpoint and then VPC endpoints are kind of mapped to a particular AWS service so if you want to talk to different AWS
Services then you need to create separate VPC endpoints.

means we don't want to have internet access but want to access only AWS services and for this AWS has a another concept called **VPC endpoints.**

VPC endpoints is used to access only AWS Services without NAT and IG.

You create a VPC endpoint and attach it to your private subnect and then can use it

Note:- If you want to use S3 , then you create VPC endpoint only for S3 , If you want to use RDS , then you create VPC endpoint only for RDS. means for each AWS Services , you will create diffrenet different VPC endpoint.

If you don't want the complete internet access in AWS and if you want to access only AWS Services then go with the VPC endpoint.

There 3 types of VPC Endpoint

1. Interface endpoints :- has private link and uses (ENI)

Elastic Network Interface.

2. **Gateway Load Balancer endpoints :--** has private link and uses (ENI) Elastic Network Interface.

3. **Gateway endpoints :--** Works with Routing tables

==========

What is CIDR :-- CIDR means (class less inter domain routing).

CIDR **(Classless Inter-Domain Routing or supernetting)** is a method of assigning IP addresses that improves the efficiency of address distribution and replaces the previous system based on Class A, Class B and Class C networks.

A VPC must have an associated IPv4 CIDR block.

what is subnet in aws

A subnet is a range of IP addresses in your VPC. You launch AWS resources, such as Amazon EC2 instances, into your subnets. You can connect a subnet to the internet, other VPCs, and your own data centers, and route traffic to and from your subnets using route tables. each subnet is associated to one availability Zone.

Note :-- anyone working in inside the company IP address will start with (192 , 10 , 172) series

Example :-- 192.168.1.0 , 10.168.1.0 , 172.168.1.0

Note :-- one subnet can not be Associated to multiple availability zones. one availability Zone can have multiple subnets.

What is subnet mask :-- subnet masks are used internally within a network. A subnet mask is a 32-bit address that segregates an IP address into network bits that identify the network and host bits that identify the host device operating on that network.
subnet mask will decide how many IP addresses **you will get base on 2 ^ (32 - n) formula.**

Example :-- 192.168.1.0/24 :-- note (24) is subnet mask , and will decide , how many ip address will create using 2 ^ (32 - n) formula.

Every subnet have 5 IP reserved

1. (.0) you can not assign , it is internally used for network related

2. (.1) you can not assign , it is internally used for routing purpose

3. (.2) you can not assign , it is internally used for DNS

4. (.3) you can not assign , it is internally used for Future purose

5. (.255) you can not assign , it is internally used for broadcasting

==========

NOTE for (IPv4 CIDR = 10.0.0.0/26) :-- how IP Address will be generated.

START IP - 10.0.0.0

calculation formula - 2 ^ (total ip - given range of ip/range)

Example - 2 ^ (32 - 26) = 64
Explanation - 32-26 = 6 , so mulitiply 6 times with 2 figure
2*2*2*2*2*2 = 64
Detail Explanation -
2*2=4
4*2=8
8*2=16
16*2=32
32*2=54

END IP - 10.0.0.63

so from (10.0.0.0 to 10.0.0.63) IP will be generated and

1. (.0) you can not assign , it is internally used for network related
2. (.1) you can not assign , it is internally used for routing purpose
3. (.2) you can not assign , it is internally used for DNS
4. (.3) you can not assign , it is internally used for Future purose actually you can use from (10.0.0.4 to 10.0.0.63) IP.

==========

VPC Practical :--

After login on AWS, search "VPC" service and open the "VPC" dashboard and follow below steps.

STEPS :-

0. select ASIA PACIFIC (Mumbai) ap-south-1 Region.
1. click on Create VPC
2.select (VPC only option) from "VPC setting"
and set below properties

3. Name tag   = myVPC
4. IPv4 CIDR   = 10.0.0.0/26
5. Tenancy   = Default
6. and click on **"Create VPC"** button.

==========

CREATE (3) PRIVATE SUBNETS

7. now create the "subnets" and point this "myVPC" created VPC to
  subnet.
8. select (Create subnet) and set below properties
9. VPC ID   = myVPC   (Create subnets in "myVPC" VPC).
10. Subnet name = private-myVPC-1
11. Availability Zone = Asia Pacific (Mumbai)/ ap-south-1a
12. IPv4 CIDR block   = 10.0.0.0/28
13. and click on "Create subnet" button.
14. now create the "subnets" and point this "myVPC" created VPC
    to subnet.
15. select (Create subnet) and set below properties
16. VPC ID   = myVPC   (Create subnets in "myVPC" VPC).
17. Subnet name = private-myVPC-2
18. Availability Zone = Asia Pacific (Mumbai)/ ap-south-1b
19. IPv4 CIDR block   = 10.0.0.8/28
20. and click on "Create subnet" button.
14. now create the "subnets" and point this "myVPC" created VPC
    to subnet.
15. select (Create subnet) and set below properties
16. VPC ID   = myVPC   (Create subnets in "myVPC" VPC).
17. Subnet name = private-myVPC-3
18. Availability Zone = Asia Pacific (Mumbai)/ ap-south-1c
19. IPv4 CIDR block   = 10.0.0.16/28
20. and click on "Create subnet" button.

==========

CREATE (3) PUBLIC SUBNETS

7. now create the "subnets" and point this "myVPC" created VPC to
  subnet.
8. select (Create subnet) and set below properties
9. VPC ID   = myVPC   (Create subnets in "myVPC" VPC).
10. Subnet name = public-myVPC-1
11. Availability Zone = Asia Pacific (Mumbai)/ ap-south-1a
12. IPv4 CIDR block   = 10.0.0.24/28
13. and click on "Create subnet" button.
7. now create the "subnets" and point this "myVPC" created VPC to
   subnet.
8. select (Create subnet) and set below properties
9. VPC ID   = myVPC   (Create subnets in "myVPC" VPC).

10. Subnet name = public-myVPC-1
11. Availability Zone = Asia Pacific (Mumbai)/ ap-south-1a
12. IPv4 CIDR block = 10.0.0.32/28
13. and click on "Create subnet" button.
14. now create the "subnets" and point this "myVPC" created VPC to subnet.
15. select (Create subnet) and set below properties
16. VPC ID   = myVPC   (Create subnets in "myVPC" VPC).
17. Subnet name = public-myVPC-2
18. Availability Zone = Asia Pacific (Mumbai)/ ap-south-1b
19. IPv4 CIDR block = 10.0.0.40/28
20. and click on "Create subnet" button.
21. after creating subnets automatically Route tables will be created

==========

ROUTE TABLES Create [PUBLIC & PRIVATE] Route tables.

7. now select and click on the "Route tables" and search by "myVPC".
8. and modify below prorperties of "Route Table".
9. name   = public-myVPC-Route-Table
7. now click on "Create Route table" button and create private Route table under "myVPC" VPC.
8. and set below prorperties of "Route Table".
9. name   = private-myVPC-Route-Table
9. VPC   = myVPC
10. now select "private-myVPC-Route-Table" and click on "subnet associations" button and click on "Edit" button.
11. and associate ("private-myVPC-1" ,"private-myVPC-2" , "private-myVPC-3") subnets to this "private-myVPC-Route-Table" and click on "Save associations" button.

10. now select "public-myVPC-Route-Table" and click on "subnet associations" button and click on "Edit" button.
11. and associate ("public-myVPC-1" ,"public-myVPC-2" , "public-myVPC-3") subnets to this "public-myVPC-Route-Table" and click on "Save associations" button.

==========

Note :- Remember, a subnet can have only one route table, but there can be one route table Associated with multiple subnets.

==========

INTERNET GATEWAYS

Now , we need to configure "INTERNET GATEWAYS" . "INTERNET GATEWAYS" is entity which allows connectivity from VPC to outside internet, so

1. select and Click on Internet Gateways and click on "Create Internet Gateways" and set below properties of "Internet Gateways".
2. name = myVPC-Internet-Gateway
3. and click on "Create Internet Gateways" button.
4. now attached created ("myVPC-Internet-Gateway") Internet gateway to created ("myVPC") VPC. for that
5. Click on Internet Gateways and comes on Internet Gateways dashboard
6. then click on "Internet gateway ID" column of ("myVPC-Internet-Gateway") table row.
7. then click on "Actions" and select and click on ("Attach to VPC") button and set below properties.
8. Available VPCs = myVPC

==========

Now , again select and click on the "Route tables" and search by "myVPC" and do following steps.

1. select " public-myVPC-Route-Table" Route table and just click on ("Route") tab.
2. then click on ("Edit Routes") button, and add below route properties.
3. Destination = 0.0.0.0/0
4. Target = (here you just type "igw-") it will show you created Internet Gateway ("myVPC-Internet-Gateway").
5. then click on "save changes" button.

==========

NAT GATEWAYS :-- A NAT gateway is a Network Address Translation (NAT) service. You can use a NAT gateway so that instances in a private subnet can connect to services outside your VPC but external services cannot initiate a connection with those instances.

now we need to configure NAT Gateways, so just select "NAT Gateways" and click and do follwoing actions.

1. Name = myVPC-NAT-Gateway
2. Subnet = select (public subnet) public-myVPC-1
3. Connectivity type = public
4. Elastic IP allocation ID = just click on "Allocate Elastic IP" Button to assign ("Elastic IP")
5. then click on "Create NAT Gateway" button
6. then again select and click on the "Route tables" and search by "myVPC" and follow below steps
7. select "private-myVPC-Route-Table" Route table. and open "Route" Tab and click on "Edit Routes" button. and add below route properties.
3. Destination = 0.0.0.0/0
4. Target = (here you just type "nat-") it will show you created NAT Gateway ("myVPC-NAT-Gateway").
5. then click on "save changes" button.

==========

SUBNETS

now again select "subnets" and for every public subnets enable "Enable auto-assign public IPv4 address" from "Auto-assign IP settings" option which comes under the "Edit subnet settings".

1. select "public-myVPC1" subnet and just click on "Action" button
2. then select "Edit subnet settings".
3. then see the "Auto-assign IP settings"
4 then select "Enable auto-assign public IPv4 address" checkbox option.
3. and save it

1. select "public-myVPC2" subnet and just click on "Action" button
2. then select "Edit subnet settings".
3. then see the "Auto-assign IP settings"
4 then select "Enable auto-assign public IPv4 address" checkbox option.
3. and save it

==========

VPC

now again go in "VPC" dashboard and select "myVPC" VPC and just click on Action and click on "Edit VPC Settings".

1. then go to "DNS settings"
2. then select "Enable DNS resolution"
3. then select "Enable DNS hostnames"
4. then save it.

==========

SECURITY GROUP

now search "EC2" service and open the "EC2" dashboard and follow below steps.

1.From left side menu , go to "Network & Security" option and just click on "Security Group" link
2.then just click on "Create Security Group" button and configure the below attributes.
3. Security Group Name   = myCustomSG
4. select VPC   = myVPC

5. set Inbound Rules

Type	Protocol	Source
SSH	TCP	MY IP

NOTE :-- in INBOUND RULE ADD RULE FOR SELF SECURITY GROUP , for that add one more rule in INBOUND RUL

Type	Protocol	Source
All Traffic	All	myCustomSG (SELF SECURITY GROUP, means , your adding rule for your self).

6. then just click on "Create Security Group" button to create Security Group.

==========

CREATE KEY PAIRS

now search "KEY PAIRS" service and open the "KEY PAIRS" dashboard and follow below steps.

1. then just click on "Create Key pair" button and configure below properties.

   Key pair :-- 6PMBATCH
   Key pair type :-- RSA
   Private key file format :-- .pem

2. then just click on "Create Key pair" button to create pair
3. whenever you will click on "Create Key pair" button , then it will download the "6PMBATCH.pem" file on your local machine.

==========

CREATE EC2 INSTANCE

Note : -- We will create Two instances (Public & Private), and Whenever you create the EC2 instance in public subnet ,then it is called Bastion / jump server.

CREATE PUBLIC EC2 INSTANCE (Bastion / jump server)

now search "EC2" service and open the "EC2" dashboard and follow below steps, to create EC2 instance.

1.From left side menu , go to "Instances" option and just click on "Instances" link

2.then just click on "Launch Instance" button to create "PUBLIC INSTANCE" and configure the below attributes.

Launch and instance

1. Name and tags = BastionServer

2. Application and OS Images (Amazon Machine Image) [AMI] :-- just click on "Browse mor AMIs" and select free tier base

ubuntu AMI.

3. select KEY PAIR :-- Which you have created KEY PAIR (6PMBATCH)

4. Network settings
   select VPC = myVPC
   select Subnet = select public subnet (public-myVPC1, public-myVPC2)
   select Security Group = myCustomSG
   Auto assign Public IP = select enable option
   (for public EC2 instance always enable Auto assign Public IP)

5. then just click on "launch instance" button to create public EC2 (Bastion / jump) server.

==========

CREATE PRIVATE EC2 INSTANCE

now search "EC2" service and open the "EC2" dashboard and follow below steps, to create EC2 instance

1.From left side menu , go to "Instances" option and just click on "Instances" link

2.then just click on "Launch Instance" button to create "PRIVATE INSTANCE" and configure the below attributes.

Launch and instance
1. Name and tags = PrivateServer

2. Application and OS Images (Amazon Machine Image) [AMI] :--
just click on "Browse mor AMIs" and select free tier base ubuntu AMI.

3. select KEY PAIR :-- Which you have created KEY PAIR (6PMBATCH)

4. Network settings
   select VPC = myVPC
   select Subnet = select public subnet (private-myVPC1, private-myVPC2)
   select Security Group = myCustomSG
   Auto assign Public IP = select disable option (for private EC2 instance always disable Auto assign Public IP)

5. then just click on "launch instance" button to create private EC2 server.

==========

ACCESS PRIVATE EC2 INSTANCE FROM PUBLIC EC2 INSTANCE (Bastion / jump server)

1. Connect to PUBLIC EC2 INSTANCE (Bastion / jump server) from terminal

2. then using SSH command you can connect to PRIVATE EC2 INSTANCE , for that do following step

   a) go to the "EC2" dashboard and just select "PrivateServer" PRIVATE EC2 INSTANCE

   b) then just click on "CONNECT" button and to "Connect to instance" dashboard.

   c) in "Connect to instance" , go to "SSH client" tab

   d) copy (SSH) command from "Connect to instance"

   Example Command :-- ssh -i <PEM FILE> ec2-user@<PRIVATE SUBNET IP ADDRESS>
   sample command :-- ssh -i "6PMBATCH.pem" ec2-user@10.0.0.8

   e) use this command in your PUBLIC EC2 INSTANCE in terminal to connect PRIVATE EC2 INSTANCE.

   Note :--
   1. we required 6PMBATCH.pem file to connect PRIVATE EC2 INSTANCE, for that you just copy the 6PMBATCH.pem file on your PUBLIC EC2 INSTANCE.

   2. then we need to set permission to 6PMBATCH.pem file using below command

   command :-- chmod 777 6PMBATCH.pem

==========

ROLE FOR VPC ENDPOINT :--

1. first you install the AWS cli on your PRIVATE EC2 INSTANCE machine.
2. then you create the ROLE to access AWS Resources

Note:- If you want to use S3 , then you create VPC endpoint only for S3 , If you want to use RDS , then you create VPC endpoint only for RDS. means for each AWS Services , you will create diffrenet different VPC endpoint.

3. now search "IAM" service and open the "IAM" dashboard and follow below steps, to create Role

4. From left side menu , go to "Access management" option and just click on "Roles" link
5. then just click on "create role" button.
6. then choose trusted entity = AWS services
7. then choose Use case = Common use cases (EC2)
8. then just click on "next" button and go to the "Add permissions" dashboard
9. then Add "Permissions policies" = AmazonS3FullAccess
10. then just click on "next" button and go to "Name Review and creation" dashboard
11. then set Role name = 6PMVPCENDPOINTDEMO
12. then just click on "Create Role" button to create Role.

Now attach this role to PRIVATE EC2 INSTANCE machine for that, go to EC2 Service and select "PRIVATE EC2 INSTANCE"

1. then just click on "Actions" dropdown and select "Security" option and just click on "Modify IAM Role" button.
2. then select the newly created role "6PMVPCENDPOINTDEMO" and just click on "update IAM Role" button.

Note :-- now you disbale the internet access for "PRIVATE EC2 INSTANCE" , for that you go in PRIVATE ROUTING TABLE "private-myVPC-Route-Table" and go to the "Routes" tab and remove the NAT entry.

==========

CREATE VPC ENDPOINT :--

There 3 types of VPC Endpoint

1. Interface endpoints :- has private link and uses (ENI) Elastic Network Interface.

2. Gateway Load Balancer endpoints :-- has private link and uses (ENI) Elastic Network Interface.

3. Gateway endpoints :-- Works with Routing tables

We have to use "Gateway endpoints"

1. now search "IAM" service and open the "IAM" dashboard and follow below steps, to create Role

2. From left side menu , just click on "Endpoints" option and just click on "Create EndPoint" button and configure the below properties.

a) Endpoint setting Name = S3-VPC-ENDPOINT

b) Service Category = AWS services
c) select Service = here just type "S3" and you will get list from list you have to select (Gateway base S3 endpoint).

Example :-- if you type "S3" in select service option , you will get below list table

Service Name	Owner	Type
com.amazonaws.ap.south-1.s3	amazon	Gateway
com.amazonaws.ap.south-1.s3	amazon	Interface
com.amazonaws.ap.south-1.s3-outputs	amazon	Interface

Selection Criteria :- you select that row, which type have (Gateway)
selected ROW/Option :- com.amazonaws.ap.south-1.s3 amazon Gateway.

d) select VPC = myVPC

3. then select Routing table = private-myVPC-Route-Table

4. then select Policy = Full access

5. then just click on "Create endpoint" button to create endpoint.

now you go in PRIVATE ROUTING TABLE "private-myVPC-Route-Table" and go to the "Routes" tab , there you will able to see entry of VPC ENDPOINT.

now go to PRIVATE EC2 INSTANCE terminal and just tye all S3 Related (AWS cli) command to create , update delete S3 buckets,

benifit :-- without internet, using VPC ENDPOINT ,from your "PRIVATE EC2 INSTANCE" using "AWS CLI" command you can accees S3 service.

you can execution all (S3 COMMANDs) from terminal with help of AWS CLI.

COMMANDS :-

```
aws s3 help
aws s3 ls

aws s3api create-bucket --bucket test-bucket -region us-east-1

aws s3 ls
```

==========

NOTE for (IPv4 CIDR = 10.0.0.0/26) :-- how IP Address will be generated.

START IP - 10.0.0.0
calculation formula - 2 ^ (total ip - given range of ip/range)

Example - 2 ^ (32 - 26) = 64
Explanation - 32-26 = 6 , so mulitiply 6 times with 2 figure
2*2*2*2*2*2 = 64

Detail Explanation -
2*2=4
4*2=8
8*2=16
16*2=32
32*2=54

END IP - 10.0.0.63

==========

VPC Peering :-- A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them using private IPv4 addresses or IPv6 addresses.

A VPC peering connection is a networking connection between two VPCs that enables routing using each VPC's private IP addresses as if they were in the same network. VPC peering connections can be created between your own VPCs or with a VPC in another AWS account. VPC peering also supports inter-region peering.

Note :-- VPC peering can be done in the (same region , different region , different account).

VPC PEERING PRACTICAL :-- We will create 2 VPC in different different regions.

in MUMBAI aws region
1. We will create VPC with (1 public subnet & 1 private subnet).
2. and will launch 1 EC2 instance in Bastion (with public subnet)
3. and will launch 1 EC2 instance (with private subnet)

in IRELAND aws region

1. We will create VPC with (1 public subnet & 1 private subnet).
2. and will launch 1 EC2 instance in Bastion (with public subnet)
3. and will launch 1 EC2 instance (with private subnet)

OUR ACTUAL TASK :--
1 We will connect to (MUMBAI Bastion) EC2 then and will connect to PRIVATE EC2 instance of MUMBAI Region
2. and then from (PRIVATE EC2 instance of MUMBAI Region) will directly connect to (PRIVATE EC2 of IRELAND aws region) with the concept of VPC Peering.

TASK AGENDA :- from your Mumbai private server connect to the another region (PRIVATE EC2 of IRELAND aws region).

****

CREATE VPC ASIA PACIFIC (Mumbai) ap-south-1 Region. with following details.

CREATE VPC

VPC NAME :-- MumbaiVPC
IPV4 CIDR :-- 192.168.0.0/16
then click on "Create VPC" button.

CREATE INTERNET GATEWAY

Internet Gateway Name :- MumbaiIGW
then click on "Create Internet Gateway" button.
and attach it with (MumbaiVPC)

CREATE SUBNET

select VPC :- MumbaiVPC
Subnet Name :- public subnet
select Availability Zone :-- Asia Pacific (Mumbai)/ap-south-1a
IPV4 CIDR BLOCK :-- 192.168.1.0/24

then click on "Add Subnet" button to create Private VPC

select VPC :- MumbaiVPC
Subnet Name :- private subnet
select Availability Zone :-- Asia Pacific (Mumbai)/ap-south-1b
IPV4 CIDR BLOCK :-- 192.168.2.0/24

then click on "Create Subnet" button.

CREATE NAT GATEWAY

NAT (Network Address Translation) Name :-- MumbaiNAT
Select subnet :-- public subnet
connectivity type :-- public
Elastic IP allocation ID :- Just click on "Allocate Elastic IP" button (to assign Elastic IP to NAT gate way).

then click on "Create NAT gateway" button.

CREATE ROUTE TABLE (public RT)

Route Table Name :-- Public RT

selec VPC :-- MumbaiVPC

then click on "Create Route Table" button.

then goto "Route tables" dashboard and select "public RT" route table
then goto "Routes" tab and just click on "Edit routes" button and below route rules.

Destination	Target
0.0.0.0/0	select Internet Gateway (MumbaiIGW)

then just click on "Save changes" button.
then goto "Route tables" dashboard and select "public RT" route table
then goto "Subnet associations" tab and just click on "Edit Subnet associations" button
and then from Available subnets list :-- select (public subnet)
and then just click on "Save associations" button.

CREATE ROUTE TABLE (private RT)
Route Table Name :-- private RT
selec VPC :-- MumbaiVPC

then click on "Create Route Table" button.

then goto "Route tables" dashboard and select "private RT" route table
then goto "Routes" tab and just click on "Edit routes" button and below route rules.

Destination	Target
0.0.0.0/0	select NAT Gateway (MumbaiNAT)

then just click on "Save changes" button.
then goto "Route tables" dashboard and select "private RT" route table
then goto "Subnet associations" tab and just click on "Edit Subnet associations" button
and then from Available subnets list :-- select (private subnet)
and then just click on "Save associations" button.

CREATE SECURITY GROUP

Security Group name :-- MyOWNMumbaiSG
Description :-- MyOWNMumbaiSG
select VPC :-- MumbaiVPC
then in "Inbound rules" section, just click on "Add Rule" button add below values

Type	Protocol	Port range	Source
SSH	TCP	22	MY IP

NOTE :-- in INBOUND RULE ADD RULE FOR SELF SECURITY GROUP , for that add one more rule in INBOUND RULE.

Type	Protocol	Source
All Traffic	ALL	MyOWNMumbaiSG (SELF SECURITY GROUP, means , your adding rule for your self).

and then just click on "Create security group" button.

CREATE PUBLIC EC2 INSTANCE (Bastion / jump server) :-- now search "EC2" service and open the "EC2" dashboard and follow below steps, to create EC2 instance

1.From left side menu , go to "Instances" option and just click on "Instances" link

2.then just click on "Launch Instance" button to create "PUBLIC INSTANCE" and configure the below attributes.

Launch and instance
1. Name and tags = BastionServer

2. Application and OS Images (Amazon Machine Image) [AMI] :-- just click on "Browse mor AMIs" and select free tier base ubuntu AMI.

3. select KEY PAIR :-- Which you have created KEY PAIR (6PMBATCH)

4. Network settings
   select VPC = MumbaiVPC
   select Subnet = select public subnet (public subnet)
   select Security Group = MyOWNMumbaiSG
   Auto assign Public IP = select enable option (for public EC2 instance always enable Auto assign Public IP)

5. then just click on "launch instance" button to create public EC2 (Bastion / jump) server.

CREATE PRIVATE EC2 INSTANCE :-- now search "EC2" service and open the "EC2" dashboard and follow below steps, to create EC2 instance

1.From left side menu , go to "Instances" option and just click on "Instances" link

2.then just click on "Launch Instance" button to create "PRIVATE INSTANCE" and configure the below attributes.

Launch and instance
1. Name and tags = PrivateServer

2. Application and OS Images (Amazon Machine Image) [AMI] :-- just click on "Browse mor AMIs" and select free tier base ubuntu AMI.

3. select KEY PAIR :-- Which you have created KEY PAIR (6PMBATCH)

4. Network settings
     select VPC   = MumbaiVPC
     select Subnet   = select public subnet (private subnet)
     select Security Group = MyOWNMumbaiSG
     Auto assign Public IP = select disable option (for private EC2 instance always disable Auto assign Public IP)

5. then just click on "launch instance" button to create private EC2 server.

CREATE VPC EUROPE (Ireland) eu-west-1 Region. with following details.

CREATE VPC

VPC NAME :-- IrelandVPC
IPV4 CIDR :-- 192.169.0.0/16
then click on "Create VPC" button.

CREATE INTERNET GATEWAY

Internet Gateway Name :- IrelandIGW
then click on "Create Internet Gateway" button.
and attach it with (IrelandVPC)

CREATE SUBNET

select VPC :- IrelandVPC
Subnet Name :- public subnet
select Availability Zone :-- Asia Pacific (Ireland)/eu-west-1a
IPV4 CIDR BLOCK :-- 192.169.1.0/24

then click on "Add Subnet" button to create Private VPC

select VPC :- IrelandVPC
Subnet Name :- private subnet
select Availability Zone :-- Asia Pacific (Ireland)/eu-west-1b
IPV4 CIDR BLOCK :-- 192.169.2.0/24

then click on "Create Subnet" button.

CREATE NAT GATEWAY

NAT (Network Address Translation) Name :-- IrelandNAT
Select subnet :-- public subnet
connectivity type :-- public
Elastic IP allocation ID :- Just click on "Allocate Elastic IP" button
                            (to assign Elastic IP to NAT gate way).

then click on "Create NAT gateway" button.

CREATE ROUTE TABLE (public RT)
Route Table Name :-- Public RT
selec VPC :-- IrelandVPC

then click on "Create Route Table" button.

then goto "Route tables" dashboard and select "public RT" route table
then goto "Routes" tab and just click on "Edit routes" button and below route rules.

Destination	Target
0.0.0.0/0	select Internet Gateway (IrelandIGW)

then just click on "Save changes" button.
then goto "Route tables" dashboard and select "public RT" route table
then goto "Subnet associations" tab and just click on "Edit Subnet associations" button
and then from Available subnets list :-- select (public subnet)
and then just click on "Save associations" button.

CREATE ROUTE TABLE (private RT)
Route Table Name :-- private RT
selec VPC :-- IrelandVPC

then click on "Create Route Table" button.

then goto "Route tables" dashboard and select "private RT" route table
then goto "Routes" tab and just click on "Edit routes" button and below route rules.

Destination	Target
0.0.0.0/0	select NAT Gateway (IrelandNAT)

then just click on "Save changes" button.
then goto "Route tables" dashboard and select "private RT" route table
then goto "Subnet associations" tab and just click on "Edit Subnet associations" button
and then from Available subnets list :-- select (private subnet)
and then just click on "Save associations" button.

CREATE SECURITY GROUP

Security Group name :-- MyOWNIrelandSG
Description :-- MyOWNIrelandSG
select VPC :-- IrelandVPC
then in "Inbound rules" section, just click on "Add Rule" button add below values

Type	Protocol	Port range	Source
SSH	TCP	22	MY IP

NOTE :-- in INBOUND RULE ADD RULE FOR SELF SECURITY GROUP , for that add one more rule in INBOUND RUL

Type	Protocol	Source
All Traffic	ALL	MyOWNIrelandSG (SELF SECURITY GROUP, means , your adding rule for your self).

and then just click on "Create security group" button.

==========

CREATE PUBLIC EC2 INSTANCE (Bastion / jump server)

now search "EC2" service and open the "EC2" dashboard and follow below steps, to create EC2 instance
1.From left side menu , go to "Instances" option and just click on "Instances" link

2.then just click on "Launch Instance" button to create "PUBLIC INSTANCE" and configure the below attributes.

Launch and instance
1. Name and tags = BastionServer

2. Application and OS Images (Amazon Machine Image) [AMI] :--
just click on "Browse mor AMIs" and select free tier base ubuntu AMI.

3. select KEY PAIR :-- Which you have created KEY PAIR (6PMBATCH)

4. Network settings
select VPC = IrelandVPC
select Subnet = select public subnet (public subnet)
select Security Group = MyOWNIrelandSG
Auto assign Public IP = select enable option (for public EC2 instance always enable Auto assign Public IP)

5. then just click on "launch instance" button to create public EC2 (Bastion / jump) server.

CREATE PRIVATE EC2 INSTANCE

now search "EC2" service and open the "EC2" dashboard and follow below steps, to create EC2 instance
1.From left side menu , go to "Instances" option and just click on "Instances" link

2.then just click on "Launch Instance" button to create "PRIVATE INSTANCE" and configure the below attributes.

Launch and instance
1. Name and tags = PrivateServer

2. Application and OS Images (Amazon Machine Image) [AMI] :--
just click on "Browse mor AMIs" and select free tier base ubuntu AMI.

3. select KEY PAIR :-- Which you have created KEY PAIR (6PMBATCH)

4. Network settings
select VPC = IrelandVPC
select Subnet = select public subnet (private subnet)
select Security Group = MyOWNIrelandSG
Auto assign Public IP = select disable option (for private EC2 instance always disable Auto assign Public IP)

5. then just click on "launch instance" button to create private EC2 server.

==========

1. Connect to PUBLIC EC2 INSTANCE (Bastion / jump server) from terminal of ASIA PACIFIC (Mumbai) ap-south-1 Region

2. then using SSH command you can connect to PRIVATE EC2 INSTANCE , for that do following step

a) go to the "EC2" dashboard and just select "PrivateServer" PRIVATE EC2 INSTANCE
b) then just click on "CONNECT" button and to "Connect to instance" dashboard.
c) in "Connect to instance" , go to "SSH client" tab
d) copy (SSH) command from "Connect to instance"

Example Command :-- ssh -i <PEM FILE> ec2-user@<PRIVATE SUBNET IP ADDRESS>
sample command :-- ssh -i "6PMBATCH.pem" ec2-user@192.168.2.358

e) use this command in your PUBLIC EC2 INSTANCE in terminal to connect PRIVATE EC2 INSTANCE.

Note :--
a. we required 6PMBATCH.pem file to connect PRIVATE EC2 INSTANCE, for that you just copy the 6PMBATCH.pem file on your PUBLIC EC2 INSTANCE.
b. then we need to set permission to 6PMBATCH.pem file using below command
command :-- chmod 777 6PMBATCH.pem

3. Connect to PUBLIC EC2 INSTANCE (Bastion / jump server) from terminal of EUROPE (Ireland) eu-west-1 Region

4. then using SSH command you can connect to PRIVATE EC2

INSTANCE , for that do following step

a) go to the "EC2" dashboard and just select "PrivateServer" PRIVATE EC2 INSTANCE
b) then just click on "CONNECT" button and to "Connect to instance" dashboard.
c) in "Connect to instance" , go to "SSH client" tab
d) copy (SSH) command from "Connect to instance"

Example Command :-- ssh -i <PEM FILE> ec2-user@<PRIVATE SUBNET IP ADDRESS>
sample command :-- ssh -i "6PMBATCH.pem" ec2-user@192.169.2.358

e) use this command in your PUBLIC EC2 INSTANCE in terminal to connect PRIVATE EC2 INSTANCE.

Note :--
a. we required 6PMBATCH.pem file to connect PRIVATE EC2 INSTANCE, for that you just copy the 6PMBATCH.pem file on your PUBLIC EC2 INSTANCE.
b. then we need to set permission to 6PMBATCH.pem file using below command
command :-- chmod 777 6PMBATCH.pem

==========

NOW WE WILL CREATE VPC PEERING

(we have to connect FROM ASIA PACIFIC (Mumbai) ap-south-1 Region TO EUROPE (Ireland) eu-west-1 Region)

==========

FIRST STEP :-- now go in (EUROPE (Ireland) eu-west-1 Region)
now search "EC2" service and open the "EC2" dashboard and follow below steps.

1.From left side menu , go to "Virtual Private Cloud" option and just click on "Your VPCs New" link
2.just select "IrelandVPC"
3.just click on "Details" tab
4. and copy the VPC ID and paste it notepad

==========

for that , you go in ASIA PACIFIC (Mumbai) ap-south-1 Region
now search "EC2" service and open the "EC2" dashboard and follow below steps.

1.From left side menu , go to "Virtual Private Cloud" option and just click on "Peering connections" link
2.just click on "Create peering connection" button and configure below properties.

a. Peering connection settings Name :- MumbaiToIrelandPeering

b. Select a Local VPC to peer with :- select the MUMBAI VPC (MumbaiVPC)

c. Select another VPC to peer with Account :- My account

d. Select Region :- Another Region and select (EUROPE (Ireland) eu-west-1 Region).

e. set VPC ID (Accepter) :- paste copied VPC ID from your notepad , which you copied from EUROPE (Ireland) eu-west-1 Region.

NOTE :-- From VPC ID (Accepter) stpe, We are sending "VPC peering request" to EUROPE (Ireland) eu-west-1 Region.

then click on "Create peering connection" button.

IMPORTANT :- We have sent "VPC peering request" to EUROPE (Ireland) eu-west-1 Region so go to EUROPE (Ireland) eu-west-1 Region and do following step.

1.From left side menu , go to "Virtual Private Cloud" option and just click on "Peering connections" link

2.here you will able to see "VPC peering request" in "Peering Connections" List

3.just click on that request

4.then click on "Actions" dropdown and just click on "Accept Request" button to accept "VPC peering request".

5.then you will able to see "VPC peering request" status is active in "Peering Connections" List.

==========

SECOND STEP :---

now you go agin in ASIA PACIFIC (Mumbai) ap-south-1 Region now search "EC2" service and open the "EC2" dashboard and follow below steps.

1.From left side menu , go to "Virtual Private Cloud" option and just

click on "Routing tables" link

2.Select Private Route table (Private RT)

3.go to the "Routes" tab and click on "Edit Routes" button

4.then click on "Add Route" button and add Route details

Destination	Target
<SET THE IRELAND VPC IPV4 CIDR>	<SELECT PEERING CONNECTION> Option from List
192.169.0.0/16	MumbaiToIrelandPeering

5.then just click on "Save changes" button.

==========

THIRD STEP :---

now you go in (EUROPE (Ireland) eu-west-1 Region) now search "EC2" service and open the "EC2" dashboard and follow below steps.

1.From left side menu , go to "Virtual Private Cloud" option and just click on "Routing tables" link
2.Select Private Route table (Private RT)
3.go to the "Routes" tab and click on "Edit Routes" button
4.then click on "Add Route" button and add Route details

Destination	Target
<SET THE ASIA PACIFIC (Mumbai) VPC IPV4 CIDR>	<SELECT PEERING CONNECTION> Option from List
192.169.0.0/16	MumbaiToIrelandPeering

5.then just click on "Save changes" button.

==========

FOURTH STEP :---

now you go agin in ASIA PACIFIC (Mumbai) ap-south-1 Region
now search "EC2" service and open the "EC2" dashboard and follow below steps.
1.From left side menu , go to "Network & Security" option and just click on "Security Group" link
2.then just click on "MyOWNMumbaiSG"
3.then just select "Inbound rules" tab and just click on "Edit inbound rules" button.
4.then just click on "Add rule" button and below rule

Type	Protocol	Port range	Source
SSH	TCP	22	192.169.0.0/16 <SET THE IRELAND VPC IPV4 CIDR >

5.then just click on "save rules" button.

==========

FIFTH STEP :---

now you go again in EUROPE (Ireland) eu-west-1 Region
now search "EC2" service and open the "EC2" dashboard and follow below steps.
1.From left side menu , go to "Network & Security" option and just click on "Security Group" link
2.then just click on "MyOWNIrelandSG"
3.then just select "Inbound rules" tab and just click on "Edit inbound rules" button.
4.then just click on "Add rule" button and below rule

Type	Protocol	Port range	Source
SSH	TCP	22	192.168.0.0/16 <SET THE ASIA PACIFIC (Mumbai) VPC IPV4 CIDR>

5.then just click on "save rules" button.

==========

NOW YOU CAN SUCCESSFULLY CONNECT or ACCESS from MUMBAI Region TO IRELAND Region PRIVATE EC2 instance.

****
****

ADDITIONAL INFORMATION:--

now suppose you have lot of VPC and you wish to create "VPC PEERING" in them, then it "VPC peering" conecpt will be
very complexed and it will be very vey hard and difficult to create "VPC peering" in this VPC's.

for that AWS Introduce Advance concept "TRANSIT GATEWAY" service used to avoid this complexity.

A transit gateway is a network transit hub and every VPC will connect to (A transit gateway is a network transit hub).
and based on the routing tables , you can route or communicate with any required VPC.

TRANSIT GATEWAY :-- A transit gateway is a network transit hub that you can use to interconnect your virtual private clouds
(VPCs) and on-premises networks. As your cloud infrastructure expands globally, inter-Region peering connects transit gateways
together using the AWS Global Infrastructure

AWS Transit Gateway connects your Amazon Virtual Private Clouds (VPCs) and on-premises networks through a central hub.
This connection simplifies your network and puts an end to complex peering relationships. Transit Gateway acts as a

highly scalable cloud router—each new connection is made only once.

You can create the connection from COMPANY VPN Connection to AWS VPC, for that first Company will create the VPN conneciton and will give you one IP ADDRESS , you just use this IP ADDRESS in "customer gateway" of AWS service

now search "EC2" service and open the "EC2" dashboard and follow below steps.

1.create VPG
2.create customer gateway :-- here you set the (COMPANY'S VPN IP ADDRESS).