What is cloud Computing

. Cloud computing is consuming IT Infrastructure over the internet where we use pay as you go model.

Cloud computing is cheap , and no upfront investment.

customers no need to maintain data center, it is actually done by your cloud provider

You can access your resources from anywhere

Cloud Service offerings:

* IaaS (Infrastructure as a service) : Ex Virtual Machines
* PaaS ( Platform as a service)
* SaaS ( Software as a Service) : Ex : gmail

AWS Global Infrastructures : Explains how Amazon maintains data centers across the globe

. AWS maintain their data centers in all regions

If infra (servers) close to you ..some sort of performance is good.

. Regions :.Are geographical locations across globe.

. Availability Zones : (Short form is AZs)

Availability zone is data center,

each region is a combinations of 2 or more availability zone.

. Edge Locations: It is network of cache servers, it is used by cloud-front

**EC2**: Elastich compute Cloud

It is a virtual server (virtual machine) in the cloud .

While creating EC2 , it will ask you few inputs like what OS you want ,what is your cpu ,what is ram size etc.

Once you feed the data click on submit it will go to datacenter and then it will aunch your VM

Write steps and info to create EC2

What is pem file : The PEM format is often used to represent certificates, certificate requests, certificate chains, and keys

SSH Client :

Putty and mobexterm tools are used to access the SSH client

AMI : Amazon machine Image , it is VM template, which is preconfigured with certain software’s and operating system

**Region** : AWS always maintain mini 2 data centers in each region, we call them as availability zones.

Suppose Mumbai 2 data cennters go down, in such cases we need to keep backup in other regions

In india Mumbai we have 2 data centers, in USA Amazon have six data centers

**VPC :**  Virtual Private cloud :

Using VPC we can build our own virtual network in cloud

Using VPC all your servers isolated from others. It helps us to secure our resources.

We should always launch our resources in to our vpc. Vpc is free

VPC spans a region when you create vpc it spans a region.

**Subnets :** we can not directly launches aws resources in to VPC. Subnet is a smaller network

VPC must be subdivide in to subnets.Subnet spans availability zone (AZ).

We have to utilize all zones.

How many subnets we need to create : The rule is Number of subnets is equal to number of AZ’s in a region. Mumbai has 3 zones we need to have three subnets.

If you want to have private and public subnets, if there are 3 zones we must have three public subnets and three private subnets that is standard rule.

What is public Subnet : If subnet is exposed to internet its public.

What is private subnet: If subnet is not reachable to internet it is private subnet.

All internet phasing applications like web applications ,websites etc. should go in to public subnet

Database servers, and any application servers which are not internet phasing should go in to private subnet.

**Design Networking in Amazon :**

To design VPC and subnet we must understand CIDR block // inter domain routing.

We have 2 types of ip address , ip4 (32 bit) and ip6(128 bit)

Number of bit are more number of IPS we get more

**Creating Public Subnet:**

To make subnet public we have to expose that to internet.

To configure internet we need internet gateway.

Internet Gateway : It is a virtual router that connect a vpc to the internet.

As we discussed by default , subnets are private to make them public we need to configure the subnet to internet thorough internet gateway.

Create VPC with one public subnet and one private subnet.

..Step1 : Create one VPC

….Step2: Create Subnet 1

Step4: Create internetgateway

Step5: Attach it

What is the private ip and punlic ip in ec2..which one you are using in realtime

Graphical user interface

Description automatically generated with medium confidence

Suppose you are using GMAIL ..it comes under which category. Its SaaS ..you are not maintaining any thing.

For gmail team it is IaaS or PaaS, depends on contest.

IaaS is means we have to maintain something, where as in PaaS maintenance is drastically reduced.

NAT instance falls under Infrastructure as a Service -- more maintenance

NAT gateway falls under Platform as a Service. -- less maintenance

Egress means –talks about outgoing traffic, and ingress talks about incoming traffic.

Private IP : It is an IP reachable within vpc. The IP will never change for EC2.

Public IP : It is used for connecting to internet and vice versa. Public IP changes when we stop and start EC2 instance

My EC2 start and running see the public IP address



When I stop EC2 instance.

A picture containing table

Description automatically generated

We have to assign public ip at the time launching EC2 , assigning of public ip cant be done after EC2 is launched.

Elastic IP : It is static ip , if we have scenario where public ip should not change even when we stop EC2 instance. We can detach elastic IP from one instance and attach to another instance.

Elastic ip pricing : one elastic ip is free, if it is associated with running EC2 instacne.

Second elastic ip onwards we have to pay

**EBS : Elastic Block Storage**

If you take our pc /laptop ..hard disk is embedded

EBS is a hard disk for your EC2 instance, but it is virtualized.

EBS and EC2 are two different , if my ec2 crashes , my EBS remains same, so that we can spin off new ec2 and attach with EBS

We should use this for frequently read write operations where I/O size is small.

We can use this for workloads like relational and non relational data bases big data analytics even for web base applications etc.

It is highly durable and highly scalable. They maintain two replicas in same availability zone.

EBS is AZ (availability zone) specific , that is this volume can be used by EC2 present in same AZ.

One EC2 can have multiple volumes. We can resize the disk. Resize means we can increase but we cant decrease (if already data is there we can not decrease)

**EBS snapshots** : it is called EBS backups . we can take backups backups are stored in S3.

We can take snap shot of one AZ and restore it in another zone.

Make sure we have EC2 instance

Create EBS volume and make sure AZ of EBS and AZ of EC2 should be same

How do you recover pem file if it is losed

There are several ways,

Option1 :

Detach root volume from EC2 where we don’t have pem file

Attach this volume to another EC2 which has password (pem file)

Generate new public private key pair ,configure public key on ebs volume.

Detach the volume attach it to your actual EC2 instance

Option 2 :

Step1:Create AMI

Step2: From AMI , Launch new EC2 from AMI and select new key while launching.

**IOPS** : Input output operations per second, number of writes (output) and number of reads (input)

Data base perfomrnace depeneds on IOPS, if our disk has more IOPS data base performance will be more.

**Elastic pipeline (ESL) :**  It is a network file system in the cloud. This can be used by multiple EC2 instances.

Do u know all softwares placed in share folder of remote machine in corelogic and every one copy and install in their machine

That sahred flder is referred as ESL in AWS terminology

**Cloud Watch :**

Cloud watch is monitoring tool in AWS.

You can monitor all of your AWS resources by default. You don’t have to setup stuff like agents tools tc..

We can also store application/aws service logs or any logs.

We can monitor applications /resources hosted outside cloud.

There are many buil-in metrics for monitoring.

Cloud watch creating and publishing custom metrics (Important for interviews)

We can configure alarams for example when EC2 CPU >80% send alerts.

When rds is running out of storage sending alerts

We can create dashboards

We can configure billing alerts and get notified

**Clod watch basic monitoring vs Detailed monitoring**

Basic monitoring .

1. It is free of cost
2. Metrics are sent at 5 minutes frequency

Detailed monitoring (Recommended for production Environments)

1. It comes with additional cost
2. Metrics are updated at 1 minute and less than one minute

**AWS S3 (Self storage Service) :**

It is object based storage in AWS

We Can store the files ,Images ,videos,audios, through S3

It maintains multiple replicas of same objects in multiple availability zones

We can store unlimited data on S3 and no need to reserve the size

Maximum size of an object could be 5 TB. If your file is greater than 5TB we have to split and store

IN EBS we can not directly access the data need EC2 ..where as in S3 we can directly store the read and write the data.

S3 exposes restful API’s using which you can store and retrieve data

Through CLI we can access the S3

1. S3 is good place to store artifacts.
2. Zip files, WAR files, Docker images ,exe files.

**Upload Objects to S3**

1. Create S3 bucket (that is mandatory)

Search for S3 🡪 Click on create Bucket 🡪 Give the unique name for S3 bucket

Graphical user interface, application

Description automatically generated

Name should lowercase letters and it should be unique. Buckets are region specific.

Bucket name should follow some validations.

Bucket name accepts only lower cases

Name can contain alpha numeric characters , it doesn’t allow most of special characters.

A bucket name can contain dot and hiphen (-) . but name can not begin with or end with dot and hiphen (-)

1. Upload objects in to bucket
2. We can upload using AWS management console
3. CLI
4. SDKs (java,.net, python etc.)

**Use case of S3 (When and where can we use S3)**

1. We use S3 for storing backups
2. If it is right place to store binaries like pdf excel videos audios photographs etc.
3. Every company (facebook, google etc) has their own object storage
4. If you open Amazon , you can images and text like below ..text stored in and coming from databases like dynomoDB and Images are stored in and coming from S3
5. S3 also used as data lake, S3 bucket can store unlimited amount of big data for analytics .
6. S3 also can be used for storing application logs.
7. S3 is good place to store artifacts.
8. Zip files, WAR files, Docker images ,exe files.

**S3 Storage Tiers**

1. Standard storage
2. S3 maintains three copies of every object across multiple AZ’s.
3. It gives you 99.999999999% of durability (we call it 11 nine durability)

Durability means loosing of data chances are very very less.

1. It gives you 99.99% availability.
2. It is for frequently accessed objects
3. Storage cost is higher than standard IA
4. Standard IA (Infrequently Accessed )
5. S3 maintains three copies of every object across multiple AZ’s.
6. It gives you 99.999999999% of durability (we call it 11 nine durability)
7. Durability means loosing of data chances are very very less.
8. It gives you 99.99% availability.
9. It is designed for infrequently accessed objects
10. Storage access cost is less than Standard storage.
11. Reduce Redundancy.
12. name it self indicates instead of 3 copies it maintain 2 copies.
13. It gives you 99.99% of durability
14. Durability means loosing of data chances are very very less.
15. Storage cost is less
16. It designed for frequently accessed objects.
17. Use this for less critical objects.
18. Intelligent Tiering
19. If you have objects with unknown access pattern then prefer this
20. It will be automatically decide the right storage type based on object access pattern.
21. One Zone IA (Infrequently Accessed)
22. Low cost option for infrequently access data.
23. It will not store data in multiple AZ’s
24. Secondary backup is good use case.

**S3 Versioning ( interview and certification )**

Versioning helps objects from accidental deletion.

If versioning is enabled , deleting objects can be restored by deleting a delete marker.

By default versioning is disabled.

Once versioning is enabled we can suspend it but we can not disable it.

If we upload 1gb file 10 times it will have 10 versions which occupying 10 gb storage.

**S3 Encryption :**

In general if we are storing sensitive data in S3 that needs to be encrypted at rest which increases

security for our data.

By default encryption is disabled we can enable it at the time of upload files t S3

Create S3 Bucket and open it 🡪 Upload a file 🡪 Click on Next 🡪

|  |  |
| --- | --- |
| S3 | EBS-Elastic block storage |
| Object level storage | Block level storage |
| It distributes across several machines and allows user to access the storage through internet | Not accessbale via internet and it AZ based |
| No need of EC2 access S3 | Need EC2 to access |

**Lamda functions :**

1. Lamda is serverless compute service (Run the code without thinking of servers)

Requirment is : Lets say our script executes 10 minutes in a day.

Scenario 1: In IaaS (Infrastructure as service) ..need to create EC2 + We need to maintain the service +performance issue we need to take care and we need to pay for TOTAL 30 days in a month

Scenario2: Serverless (Lamda) scenario : We pay only for number of seconds our script executed; we are not charged for idle time (we can reduce our cost upto 90%). We do not need to maintain the servers everything is taken care by AWS.

To implement this one should know the python (Boto3 is python sdk , which is used to to interact with AWS services)

Scenarios where we use lamda functions:

To upload the images in to S3

To send the alerts when particular EC2 stops

**IAM (Identity and Access Management) –**

IAM is used to manage access to our AWS account and resources.

1. We can share access to user by creating IAM user.
2. We can share access to services .i.e.

for example ELB want to upload access logs to S3,then ELB should have permission to

upload logs.

Application running EC2 wants to access S3

Application running on EC2 wants to access DynamoDB

1. Managing MFA (Multi factor authentication)

Federating AWS account access with third part identity providers like microsoft Active directory or LDAP

**AWS Data bases :**

1. AWS database services supports many varieties of databases.

For us most important is

1. RDS (Relational database service)
2. Redshift (Datawarehouse)
3. DynamoDB (No SQL database)
4. Elastic Cache(Distributed cache Service)

**RDS :**

1. RDS is highly durable, scalable, fully managed database in the cloud.
2. It falls under PaaS (less maintenance)
3. RDS will not give IP address (Important point)
4. RDS supports Multi AZ deployments.
5. In multi AZ Setup AWS will provision two databases instances in two different AZ’s
6. One is primary
7. Other one is standby (If primary fails, it will automatically route traffic to standby)
8. Multi AZ increases the availability of our database servers in case of zone failures.
9. Multi AZ for availability not for scalability (performance)
10. RDS Replicas
11. Along with master DB we can add additional db instance to increase application performance.
12. RR allow only select queries (read only)
13. Insert update and delete operations are not supported.
14. “RR” will have sepearte end point .
15. You can configure RR in same region or different regions this option is setting up disaster recovery of databases.
16. In case of failures RR can be promoted as master DB and your application can be pointed to new DB.

**RDS Snap shots (Database backups) :**

Taking DB backup is common activity for DB administrators

Snapshots are stored in S3

RDS supports two types of snapshots

Automated snapshots

Manual snap shots

Automated snapshots

Minimum could be zero (disabling automated backups)

Default is 7 days (retains 7 days old backups)

Max can be 35 days (Retains 35 days backups)

Manual Backups

We initiate this backup

We use backups to store database in case of failures

We can copy snapshot to different regions , this could be helpful for planning disaster recovery.

**Changing Database Attributes** :

You can change RDS attributes at any point of time.

The following are the supported database engines in RDS

1. My SQL 2. Oracle 3. Microsoft SQL 4. Arora 5.Postgre SQL 6.MariaDB

If you want to setup a database which is not in the above list then you have launch EC2 instance and setup your favourite database on it.

Aurora is amazon own database :

It delivers high performance and availability with up to 15 low-latency read replicas (Certifcation)

It has serverless capabilities and autoscaling.

**DynomoDB :**

1. DynamoDB is fully managed NOSQL database with single digit millisecond latency at any scale.
2. No size limitations , you can store and process unlimited amount of data.
3. NoSQL characteristics
4. They can handle unlimited amount of data.
5. They follow dynamic schema at runtime clients can add new column at the time of inserting data.
6. NoSQL database do not support join so there will not be performance bottlenecks.
7. No SQL supports clustering we can combine n number of compute nodes to logically form a single database.
8. DynamoDB falls under serverless.

From AWS console navigate to DynamoDB 🡪 Click on Create Table

**ELSATIC LOAD BALANCER :** It distributes traffic across multiple backend ec2 instances.

It increase high availability of and tolerance of application.

Third party load balances like nginx ,f5,haproxy etc

AWS has three types of load balances.

1. Elastic load balancer
2. Application load balancer
3. Network load balancer

Elastic Load Balancer ( ELB) :

It supports internal (private) load balancing and external load balancing. Using security group we can secure load balancer

Load balancer support http and https

**Auto Scaling:**

Auto scaling adds/removes EC2 instances based on scaling policies.

Auto scaling feature is free in amazon.

If autoscaling launches ec2 , we have to pay.

Auto scaling performs horizontal scaling.(increase /decrease number of ec2 instances)

Vertical scaling means resize the instance (resize means increase CPU and RAM memory)

**TerraForm :** Terraform is a tool for automating creation of AWS resources.

Terraform has its own syntaxes which is easy to learn and use

Terraform is widely use option for infrastructure provisioning.

Competitive tools for terraform cloud formation templates,

cloud formation is amazon native tool,there we can create resources using json and yaml .

Cloud formation works in aws cloud only not other clouds like azure

Terrform supports multiple clouds. Like aure gcp,digital portion,vmware etc.

Terraform is more dynamic and more maintainable.

**IAC**: Infra structure as Code

It is practice of creating infrastructure as code. Terrform is IAC

**TERRAFORM:**

Setting up terraform workspace

In above cloud formation model , we are uploading doing some actions in AWS console.

But in terraform its not like that through command prompt we are going to upload the files.

Grant AWS permissions to terraform, for managing resources.

Create IAM user and get access keys and secret keys which is going to be use by terraform to access AWS.

From global search ,..search for IAM

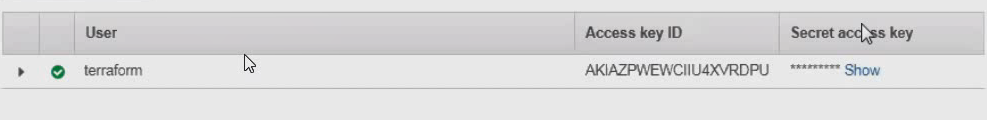
Graphical user interface, text, application

Description automatically generated

Graphical user interface, application, Word

Description automatically generated

LEAVE tags page blank click next –review page –click create user



There are diff. ways to to get the access of AWS for terraform once u get above access ky and secret key

Type1 : Create a file in Visual studio

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application, Word

Description automatically generated

Terrform supports diff clouds , so you need metion explicitly aws , so that it can recognize.

Once we get the access we need to create VPC

Terraform provider : It takes few inputs like region secret key access key etc. and also it tells terraform which cloud it has to manage.

Terraform resource : using resource block we create AWS resources.

Graphical user interface, text, application

Description automatically generated

We have to initialize terraform project

Open command prompt

Navigaete to your terraform workspace.

Text

Description automatically generated

Terraform init downloads provider plugins,you no need to run this command again and agina its one time activity.

Above command initialize terraform project by downloading provider plugin

Text

Description automatically generated

Terraform apply : this commands reads the terraform script in current folder and executes them so that it connects to aws and create resources.

Terraform will not consider subfolders

Text

Description automatically generated

Once you prompt yes it will create resources

Text

Description automatically generated

Graphical user interface, application

Description automatically generated

Assume that you need to add another parameter like location ..go and update your terraform script and rerun again ..it will add your new parameter..

Text

Description automatically generated

Text

Description automatically generated

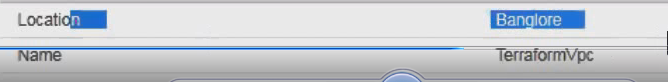
See first running the terraform apply command it shows + symbol means its adding resources

While running second time it shows ~ tilt symbol means it is updating existing symbol.

Graphical user interface, application, website

Description automatically generated

Prompt enter yes



Suppose if you want to destroy ..comment that resource part in code and run the terrfarm apply command..resources will get deleted

Text

Description automatically generated

For commenting use single line comment like java also

**Create Subnet using Terraform:**

Graphical user interface, text, application

Description automatically generated

Uncomment vpc resource and add subnet code and run the file again using terrfarm apply command

Now it will create vpc and subnet.

Every resource takes inputs , and returns outputs, inputs are called arguments and outputs are called attributes

<https://www.terraform.io/docs/providers/aws/r/vpc.html>

How to get outpu of one resource and use as input in other resource using interpolation..

Ex : while creating subnet we need vpc \_id we should get dynimaically so we need to use interpolation…. **“${aws\_vpc.main.id}”**

**Installing Terraform:**

Download terraform, latest version of terraform is **0.12**

**Download windows 64 bit 🡪 downloads zip file 🡪 extract zip 🡪**

Graphical user interface, application, Teams

Description automatically generated

You can see above terrform command as above, put it some location .. it is not installer its command.

Add terraform command to the path,

Go to environment variaoble like java,

Go to user variables and go to path ..edit path ,..at end put semi colon and put the path of terraform command location

Graphical user interface, application, Word

Description automatically generated

How to test terraform path is set or not.

Make sure that no open command prompts , open brand new command session ,

Text

Description automatically generated

You can get the terraform code here



Never add keys information git …for sure some one create instances with ur account …ur credit card will charged.

Configuring Access to Terraform

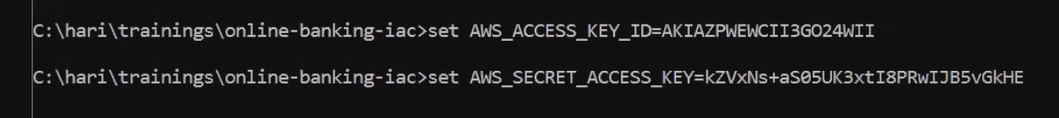
Currently we are suing access key and secret key in the code which is not a security best practice.

We need to decouple code from access key and secret keys

Setting credentials as Environment variables.

Open command prompt …change directory to location whate your terraform file location.

Command is set Envivariablename value



Now go and delete access key and secret key parameters in .tf file

Text

Description automatically generated

Now run below command

Text

Description automatically generated

Approach 2: Shared credentials file. In this approach we are going to store in a file in our laptop, so terraform reads the credentials from that file.

Lets install AWS CLI and configure access keys and secret keys.

Go to <https://aws.amazon.com/cli/>

Downoload windows 64 bit