**Ec2**

**Amazon Elastic Compute Cloud** (**EC2**) is a service given by [Amazon.com](https://en.wikipedia.org/wiki/Amazon.com" \o "Amazon.com)'s cloud-computing platform, Amazon web Services (AWS), that allows users to rent virtual computers on which they can run their own computer applications. A user can create, launch, and terminate server-instances as needed. EC2 provides users with control over a wide variety of regions of instances that allows for latency optimization and redundancy. EC2 has a secure and Resizable compute capacity in the cloud.

EC2 allows deployment of applications by providing a service through which a user can load an Amazon Machine Image (AMI) to configure a virtual machine, which Amazon calls an "instance", containing any software desired.

There are different types of Instances offered by the AWS:

* General Purpose: A1, T3, T2, M5, M5a, M4, T3a
* Compute Optimized: C5, C5n, C4
* Memory Optimized: R5, R5a, R4, X1e, X1, High Memory, z1d
* Accelerated Computing: P3, P2, G3, F1
* Storage Optimized: H1, I3, D2

Ec2 instances are offered according to the Demand i.e. User has to pay by the hour.

Features of EC2:

1. Operating Systems
2. Cost Friendly :

User has to pay for he/she use.

1. Persistent Storage
2. Auto Scaling

A user can scale the Instances up or down accordingly i.e. When there is a high demand or requirement, It can be Scaled up and vice versa.

1. Integrated with other services

EC2 is integrated with other services like S3, RDS etc to give users ease in

1. Elastic IP addresses
2. Secure
3. Flexible

Different types of Operating System ( Linux, Windows etc.)

1. Reliable

Secure Network to all the resources and services

**Load Balancer**

A load balancer **serves as the single point of contact for clients**. The load balancer distributes incoming application traffic across multiple targets, such as EC2 instances, in multiple Availability Zones. This increases the availability of your application.

A load balancer is a device that acts as a reverse proxy and distributes network or application traffic across a number of servers.

Elastic Load Balancing supports the following types of load balancers:

1. **Application Load Balancers**
2. **Network Load Balancers**
3. **Classic Load Balancers**
4. **Gateway Load Balancers**

You can add and remove resources from your load balancer as your needs change, without disturbing the overall flow of requests to your applications.

Advantages of Load Balancer :

1. Security
2. Performance
3. Resilience
4. Scalability

**Target Groups**

A target group tells a load balancer where to direct the traffic to : EC2 instances, fixed IP addresses, etc. When creating a load balancer, you create one or more listeners and configure listener rules to direct the traffic to one target group.

Target Group is a way of getting network traffic routed via specified protocols and ports to specified instances. It's basically load balancing on a port level. This is used mostly to allow accessing many applications running on different ports but the same instance.

This enables multiple applications on an instance to use the same port.

**IAM**

Identity and Access Management

IAM is a Global service, It does not require any Region or zone.

* Users are people in an organization.
* Groups only contain users, not other groups.
* Users don’t have to belong to a group.
* User can belong to multiple groups.

**IAM Permissions :**

* Users or groups can be assigned to JSON documents called policies.
* These policies define the permissions of the user.
* Should Follow Least Privilege Principle i.e. give the user permissions only what the user needs not more.

**MFA (Multi-Factor Authentication):**

Password + MFA = Very much Secure Login

MFA devices :

* **Virtual MFA Device**

1. Google Authenticator
2. Authy
3. [Duo Mobile](https://apps.apple.com/us/app/duo-mobile/id422663827)
4. [LastPass Authenticator](https://apps.apple.com/us/app/lastpass-authenticator/id1079110004)
5. [Microsoft Authenticator](https://apps.apple.com/us/app/microsoft-authenticator/id983156458)

* **Universal 2nd Factor (U2F) Security Key**
* **Hardware Key Fob MFA Device**
* **Hardware Key Fob MFA Device for AWS Gov Cloud(US)**

Different Methods to Access the AWS :

* AWS Management Console

(Password + MFA )

* AWS Command Line Interface(CLI)

Protected by Access Keys

* AWS Software Developer Kit(SDK)

Protected by Access Keys

**Access Keys :**

**Access Keys are generated through the AWS Console**

**Access are secret just like passwords, they shouldn’t be shared.**

**Access Key ID ~ Username**

**Secret Access Key ~ Password**

**AWS CLI**

* It is the Command line interface
* Through AWS CLI we can access the AWS Services using the commands in the command line shell.
* AWS CLI provides the direct access to the public API’s of AWS Services.

**AWS SDK**

* It is the Software Developer kit
* Language specific API’s (Set of libraries)
* Access & Manage AWS services programmatically.
* Supports SDK’s (JS, PHP, Ruby, Python, .NET, Java, Nodejs)
* Eg AWS CLI is built on AWS SDK for Python

**VPC**

VPC : Virtual Private cloud

* You can have multiple VPC in an AWS region
* Since it is private , only private IPV4 range is allowed.
* Your VPC CIDR should not overlap with your other networks.

Tenancy : Default or Dedicated

Default means shared hardware or dedicated hardware , dedicated hardware are costly.

We can add more IPV4 CIDR’s in the VPC , max 5.

Subnets :

* Adding Subnets to the VPC we created.
* AWS reserves 5 IP’s addresses ( first 4 and last 1) in each subnet.
* Created 2 subnets in the VPC ( Private and Public)
* 2 Private subnets in an Availability zone(Larger Subnet size)
* 2 Public subnets in an Availability zone( smaller subnet size)

Internet Gateway (IGW)

* Allows resources in a VPC to connect to the internet.
* Created separately from a VPC.
* One VPC can only be allowed to one IGW and vice versa
* Internet gateways do not allow internet access on their own.
* Route tables must also be edited.

Region

VPC

Public Subnet Private Subnet

Internet gateway Router

Route table

Public EC2 instance

* First need to launch an EC2 instance on the VPC network on the public subnet.
* To allow the Internet access create a Internet gateway and Attach it to the VPC created.
* But now we have to edit the route table to allow internet activity on the IGW
* Create 2 route table
* One private route table with private subnets in it
* One public route table with public subnets in it.
* Now editing the public route table since we created the EC2 instance in the public subnet, So we edit the routes and add 0.0.0.0/0 and set the target to Internet gateway that we created. And after saving the changes we can now access the internet on the EC2 instance on the public subnet.

Bastion host

* To connect to the EC2 instances in the private subnet through the public subnet , Users can SSH into the Bastin host in the public subnet and then SSH into the Private subnet.
* Basically Bastion host is used to SSH into our private EC2 instances
* Bastion host security group must be tightened.

NAT Instance

* NAT Network Translation
* Allows EC2 instances in private subnets to connect to the Internet.
* It should be launched in a public subnet.
* It must have elastic IP attached to it.
* Route table must be configured to route traffic from private subnets to the NAT instance.

NAT gateway

* NAT gateway much better , Managed by AWS , Higher bandwidth, high availability
* Nat gateway is created in a specific AZ, uses an Elastic IP
* Cannot be used by the EC2 in the same subnet.
* Requires an IGW since(Private Subnet => NATGW => IGW)
* Security groups aint required much.
* Can create multiple NAT gateway in multiple AZ for fault tolerance

When creating a NAT gateway , select the Subnet you want to create the NAT Gateway in.

Then allocate an Elastic IP to it.

Now you can edit the routes in the route table and select the target to the NAT gateway you have created.

NAT gateway is working now, we can connect to the internet through the private subnet using the NAT gateway.

Security Groups & NACL

NACL : Network Access Control List

* NACL are stateless, Security Groups are stateful.
* SG operates at the instance level while NACL operates at the subnet level
* NACL are like firewall which control the traffics to and from the subnets
* One NACL per subnet.
* New Subnets are assigned with default NACL

Ephermal Ports

* Client connect to a defined ports and expects a response on an ephemeral ports.
* Ephermal ports are basically a random port assigned for the connection life.

VPC Peering

* Privately connect two VPCs using the AWS network
* The VPC’s must not have overlapping CIDRs
* VPC peering connection is not transitive
* VPC peering connection can be between different AWS account/regions

Creating a Peering Network

Select a requester,

Select the account and region

Select an Accepter

Now edit the route tables to connect both the instances in the VPC