**EC2**

**What is EC2..?**

* EC2 stands for Elastic Compute Cloud
* EC2 provides on demand, scalable compute capacity on the AWS cloud. By using EC2 we can reduce the hardware costs. EC2 launches instances based on the available AMI types.
* In EC2 we can deploy as many as virtual server, In those virtual servers we can temporarily store the data after terminating the instance(virtual server) the data will deleted.
* We can increase the capacity to the virtual server or decrease the capacity of the virtual server based on our availability. We can attach or detach or modify the storage of the virtual server.
* EC2 instance can be deployed within a VPC(Virtual Private Cloud) with in the available location or region. The EC2 instance is secured inside the security group.

**Components used in lauching the EC2 instance**

* Instances
* Amazon machine images(AMIs)
* Instance types
* Security Groups
* Key pairs
* Storage volumes
* VPC
* Region
* Elastic IP addresses
* Tags

**Instances:**

* Instances are the virtual server which has its own CPU, memory, storage resources, network etc. on these basis the performance of the instance is calculated.

**Amazon Machine Images(AMIs):**

* AMI is a predefined template which is used to launching the virtual machine some predefined templates such as Amazon Linux, Windows, ubuntu, Red hat, macOS, etc.…
* AMIs can be selected based on the categories such as operating system(Windows, ubuntu, etc.) and architecture(X32 bits, X64 bits).
* AMI is used to deploy a virtual server.
* We can create our customized AMI using the snapshots.
* We can use the same AMI for the multiple EC2 instances.

**Instance Types:**

* Instances are selected based up on work load and the type of application.
* Instances types are the various configurations of the CPU, memory, storage, networking capacity.
* Types of instances:

1. General purpose instances
2. Compute optimized instance
3. Memory optimized instance
4. Storage optimized instance
5. Accelerated computing instance

**Security Groups:**

* Security groups which is like a virtual firewall which is used to control the traffic we can specify the particular user access or deny by adding the inbound and outbound rules.
* Security groups have the components like types of protocols, port ranges, source and destination

**Key pair:**

* Key pair which is used for the secure login. Key pair has the two keys one is the private key and another key is the public key.
* The local system has the private key and the AWS has the private key when we have the both keys then only we can able to access the instance.
* Key pair is used for limited access towards the EC2 instance. It is used for the security purposes.

**Storage Volumes:**

* EBS(Elastic block storage) is one of the storage volumes used in the AWS. EBS is the block level storage device that you can attach or detach or modify the storage volumes.
* We can attach multiples storage volumes to an particular EC2 instance on one condition i.e., both the storage volumes and instances are need to available in the same location.

**VPC(Virtual Private Cloud):**

* Amazon VPC enables you to define the virtual network in the AWS cloud it is known as Virtual Private Cloud.
* We can create a VPC for particular range of IP addresses and we can add them to the EC2 instances, we can create subnets, configure subnets, create routing tables, internet gateways, NACL ect.
* There will be a default VPC created automatically or we want we can create an VPC manually.

**Region:**

* Region where is the location we are operating and creating the EC2 instances and accessing other AWS services.

**Elastic IP address:**

* Elastic IP address is an static IP address which is used for the dynamic cloud computing. Elastic IP address is yours until you release it. It is charged based on the hourly based.
* Elastic IP address does not change like the dynamic public IP address. AWS allocates the Elastic IP address from the pool of the IP addresses.
* Elastic IP address is created in specific location it cannot to transferred to another location. It was attached to single instances.
* If you release it we cannot get back the released elastic IP address if you to allocation it AWS will assign the new elastic IP address from the pool of the IP addresses.

**Tags:**

* Tags are used to manage the resources what we have to assign the metadata in the form of tags.
* Tags are stored in the form of the Key value pairs.

**Launching an instance in EC2:**

1. Login to the AWS account.
2. Go to the search bar, search for EC2 and select the EC2.
3. After selecting go to the instance and click on the launch instance.
4. Name the instance(Virtual Server) .
5. Select the AMI(Amazon machine image) for launching the instance.
6. Select the type of instance(default t2.micro is selected which is free tier don’t change it).
7. Select the Key pair select the existing Key pair if there no existing key pair then create one new Key pair.
8. Edit the Network Settings select the VPC and Subnet in which we are going to launch the instance and also select the Security group and add the inbound rules.
9. Edit the storage or Enter the customized storage, Click on the Launch instance, the instance will be launched.

**Launch Templates:**

* When you launch an instance (Virtual Server), It has the specific configuration. When we need to launch instance we need to give the configuration details again and again to avoid these problems we use the launch templates.
* Each launch templates has the hundred of the version we can select any version we want. If any version is not selected the default version is selected.
* Launch templates have configurations like instance type, AMI ID, subnet, Key pair, Security groups.
* Each AWS account have 5000 launch templates per regions, 10,000 versions per one launch templates.

**Creating a Launch Templates:**

1. Login into the AWS account.
2. Search for the EC2 in the search bar, select the EC2 in EC2 select the Lauch templates.
3. Name the launch template and give the description of the launch template. Check the auto scaling guidance if you want to use for the AWS EC2 Auto Scaling.
4. Select the AMI, Instance types, Key pair, Edit the Network Settings for the Subnet of the VPC and select the Security Group.
5. Add the Storage volumes and Resource tags.
6. Now click on the create launch templates. The Launch template is created.

**Creating a Instance using the Launch Template:**

1. Search for the EC2 in the search bar, select the EC2 select the launch templates.
2. Select the existing launch templates and click on the action.
3. Click on the Launch instance from template then the new instance will be launched.

**Creating a Customized AMI using the launched instance:**

1. Go to the EC2. Select the Launched instance which is similar to the Customized AMI you want to launch.
2. Select the instance click on the action, Select the “Image and Template”, Click on the create image.
3. After Clicking give the name for the image and description make the any changes you want click on the create image.
4. In the EC2 dashboard select the AMI you will see the AMI image which you have created.
5. Select the AMI image which is created, Select it and click on the Launch instance from AMI. The Instance will be launched.
6. Now you have successfully launched the instance with your customized AMI.

**Creating the Load Balancers:**

1. Go to the EC2 dashboard and select the load balancers.
2. Select the Classic Load Balancers
3. After selecting Classic Load balancers give the name of the load balancer in the basic configuration.
4. In the Network Mapping select the VPC and the Locations.
5. In security groups add the Security groups.
6. If you want to add any existing instances you can add them in the attributes give the timeout (draining time) is 10secs
7. Click on the create Load balancers

**Load balancers:**

* Load balancers distributes the incoming traffic to the instances.
* Load balancers distributes the traffic only to the healthy instances.
* Load balancers distributes the traffic equally to all of the instances.

**Types of Load balancers:**

1. Application Load balancers: Operates application layer and supports HTTP and HTTPS.
2. Network Load Balancers: Operates at the transport layer and supports TCP,UDP.
3. Gateway Load Balancers : Operates at the network layer.
4. Classic Load balancers

**Auto Scaling Groups:**

* An Auto Scaling Group contain collection of EC2 instances that are treated as the logical grouping for the purpose of automatic scaling and management.
* The size of the Auto Scaling depends up on the desired capacity, you can adjust the desired capacity when meet the conditions.
* Auto Scaling Group Terminates the instance or add other instances based on the minimum and maximum value.
* If any instances known as the unhealthy it will terminate the unhealthy instance and add the instance in place of it.
* Scaling policy will increase or decrease the number of instances based on the health conditions.