What is AWS Auto Scaling?

AWS Auto Scaling is a service that helps the user to monitor applications and automatically adjusts the capacity to maintain steady, predictable performance at the lowest possible cost.

Benefits of Auto Scaling

Auto Scaling your application leads to the following benefits:

- Better fault tolerance
- High availability of resources
- Better cost management
- High reliability of resources
- The high flexibility of resources

In Auto Scaling, creating a backup, and restoring the data is an essential part. This can be done by creating an EBS instance. EBS (elastic backup store) is responsible for creating volume backups. It consists of two backups, namely, snapshots and AMI.

Let’s look into the concept in detail.

Snapshots vs. AMI

AWS provides a data storage service along with the Amazon EC2 instance, namely, Elastic Block Store (EBS).

EBS has Snapshots for data storage, whereas AMI is primarily associated with AWS EC2. Now, let’s take a look at the following table to understand how snapshots are different from AMIs.

Snapshots	AMI
It is used as a backup of a single EBS volume attached to the EC2 instance	It is used as a backup of an EC2 instance
Opt for this when the instance contains multiple static EBS volumes	This is widely used to replace a failed EC2 instance
Here, pay only for the storage of the modified data	Here, pay only for the storage that you use
It is a non-bootable image on EBS volume	It is a bootable image on an EC2 instance

Moving forward, let’s understand how Auto Scaling works.

How Does AWS Auto Scaling work?

- Configure a single unified scaling policy per application source.
- Explore the application and create a system that adds and removes EC2 instances in response to the requirement.
- Choose the service that you want to scale up or down.
- Select what to optimize. Based on a schedule, scale your application in response to predictable load changes.
- Keep tracing the scaling load and maintain a steady count of instances.

What Are the Different Scaling Plans?

A scaling plan helps a user to configure a set of instructions for scaling based on software requirements.

- Scaling strategy guides the service of AWS Auto Scaling on how to optimize resources in an application.
- With a scaling strategy, users can create their own strategy based on the required metrics and thresholds.

Types of Scaling Plans

- Manual scaling - This scaling helps in managing the task of building or terminating EC2 instances on its own.
- Scaling based on a schedule - Developers can predict future traffic and can schedule the time for executing AWS Auto Scaling.
- Scaling based on demand - This scaling lets developers define required scaling in response to client demand.

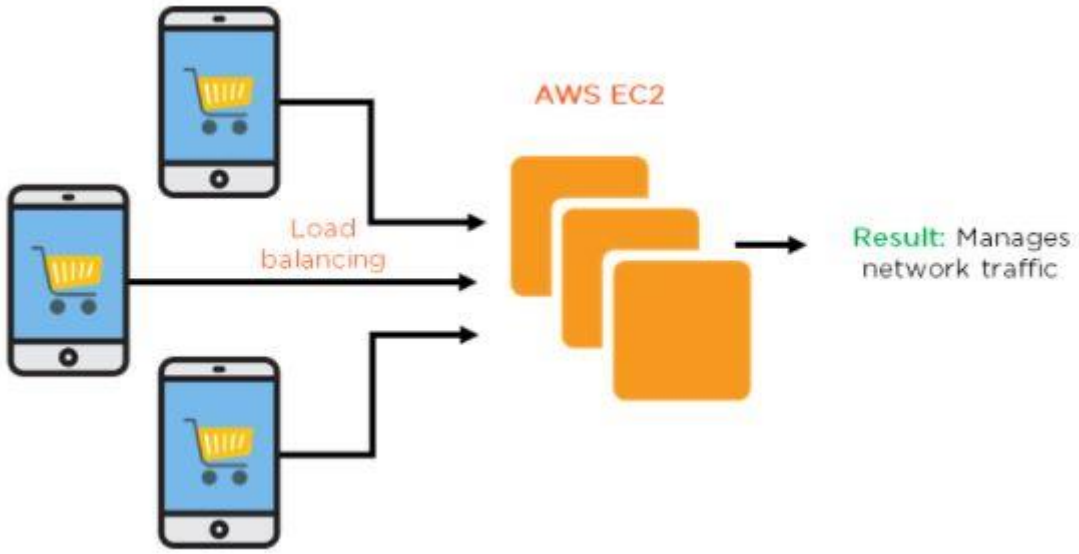
- Maintaining the current instance-level - Developers configure an Auto Scaling group for managing running instances.

This was all about Auto Scaling, now let’s understand what a load balancer is and some of its types.

What is a Load Balancer?

A load balancer behaves as a reverse proxy and performs the responsibility of delivering network traffic to various cloud servers. With the help of a load balancer, flexibility, and fault tolerance of an application increases.

For example, usually, when there is high network traffic application might end up crashing, but the AWS load balancer manages the network traffic and avoids challenges like system crash.



Moving forward, let's understand the different types of load balancers.

Types of Load Balancers

The three types of load balancers are:

- Classic load balancer
- Application load balancer

- Network load balancer

Classic Load Balancer

- It is widely used in EC2 instances and is a basic type of load balancer.
- Based on IP address and TCP port, the classic balancer routes the traffic between backend servers and users.
- This balancer does not support host-based routing and results in low efficiency of resources.

Application Load Balancer

- It is responsible for performing the tasks in the application layer of the OSI model and is the advanced form of load balancing.
- It is used when there are HTTP and HTTPS traffic routing.
- It supports host-based and path-based routing.

Network Load Balancer

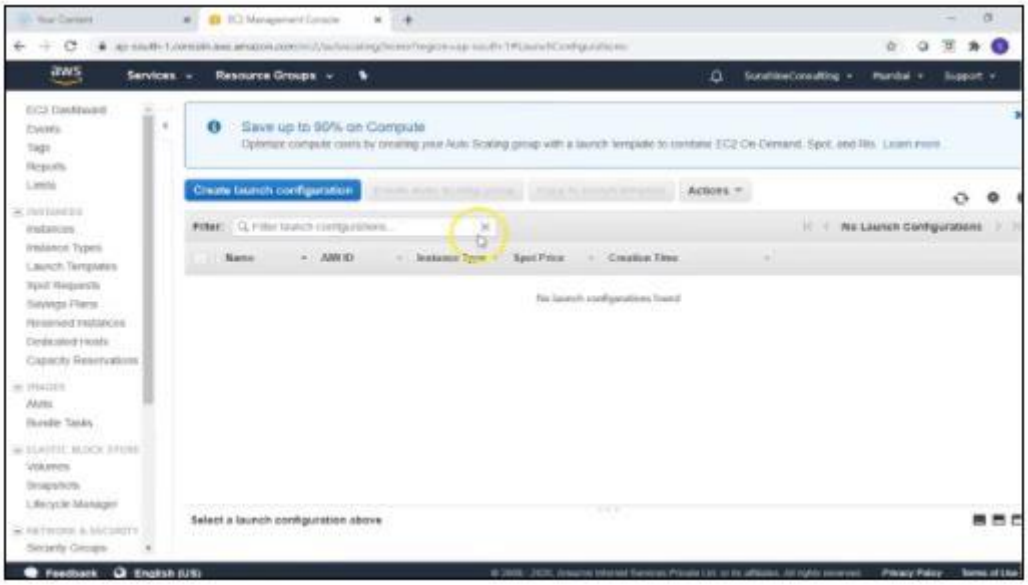
- It performs the task at layer 4 of the connection level in the OSI model.
- Its primary purpose is to route TCP traffic.
- It can manage a massive amount of traffic and is suitable for managing low latencies.

Demo - Configuring Auto Scaling Properties and Attaching AMIs

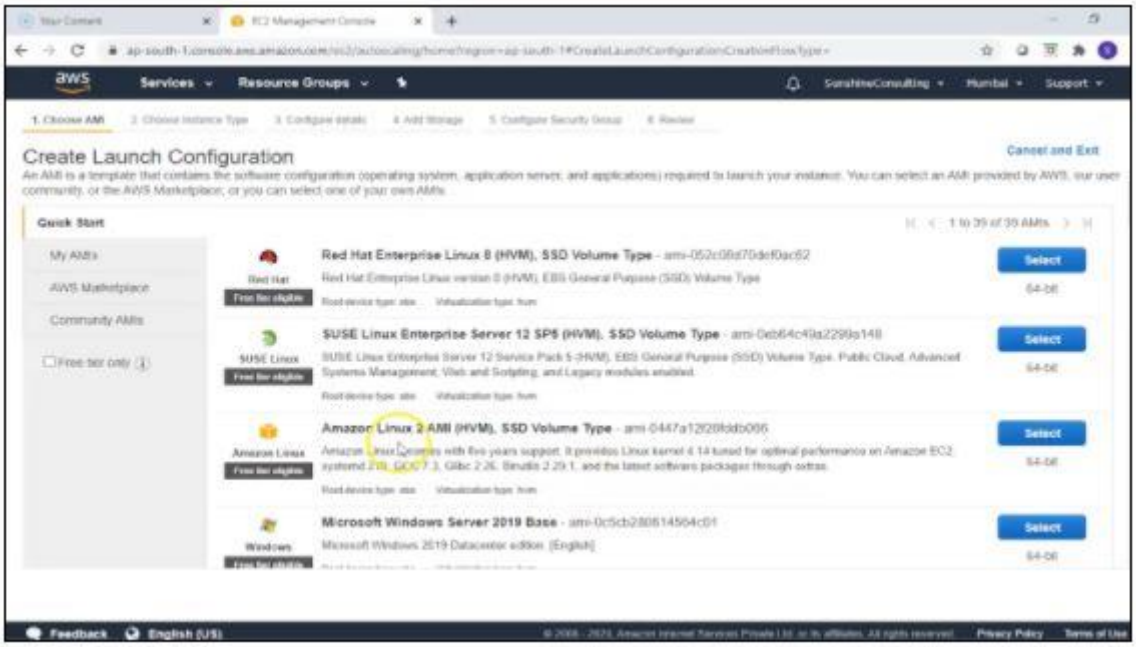
In this demo, we will be creating a launch configuration and an AWS Auto Scaling group.

Creating a Launch Configuration

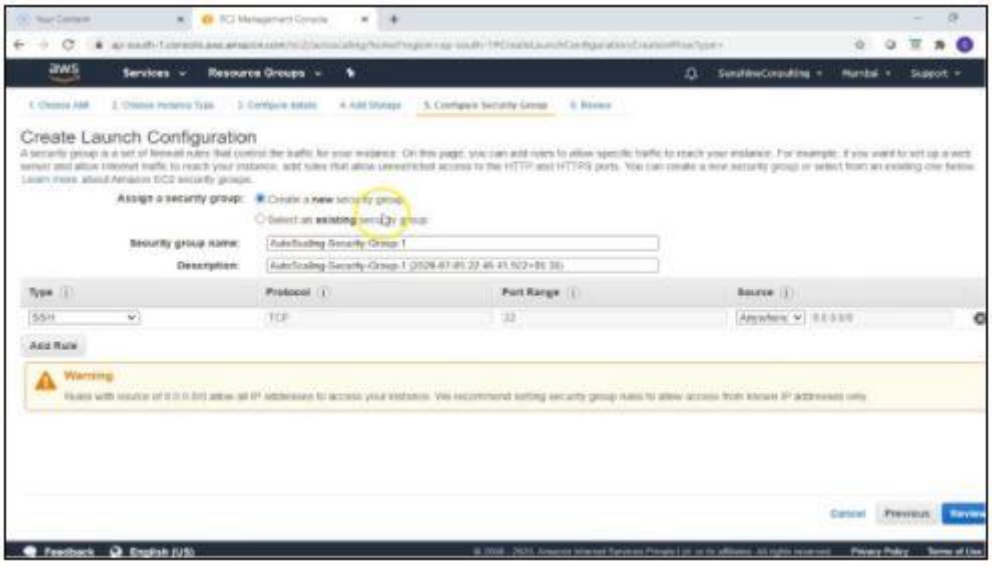
- Go to the Launch Configurations section and click on Create launch configuration.



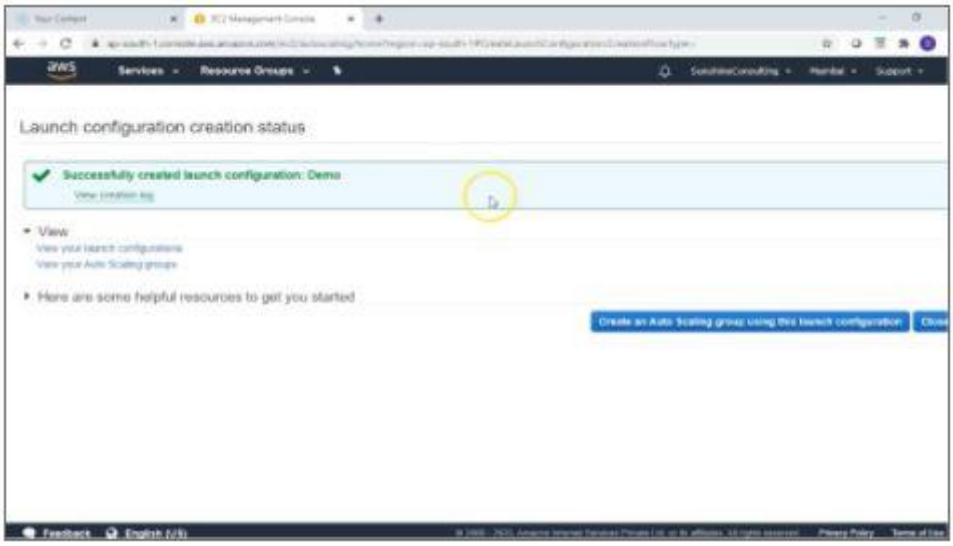
- Go to My AMIs, name the launch configuration, and move to the next option.
- Now, click on Create Launch Configuration after reviewing all configurations of Amazon Linux 2 AMI.



- Now, assign an existing security group or configure it by creating a new one.



- Next, choose a key pair before you launch the configuration, also review the entire status at the end.



Congratulations, you have created a launch configuration successfully.

Moving forward, we will be creating an Auto Scaling group in AWS.

Creating Auto Scaling Groups in AWS

- First, click on the create an auto-scaling group option using a launch configuration.
- Now, enter a suitable name for the auto-scaling group and leave the rest of the default settings.

- Now, choose at least two default subnets and click on ‘Next’ to configure scaling policies.
- Select Use scaling policies in order to assign the capacity of this group and set the desired number in the Target value.
- Now, click on Next: Configure Notifications.
- Next, keep going to the last option of creating an autoscaling group in the same section make sure to review all configurations status that you have set.