AutoScaling Lab 2:

Configuring Auto Scaling Group based on Scaling Policy. Basic tasks to create CloudWatch and set alarm for CPU utilization using AWS Management console to scale up

AutoScaling Overview:

Auto Scaling is an AWS service which ensures that we have the correct number of EC2 instances to share the load for running application. Auto Scaling can create number of instances under Auto Scaling Groups.

We can specify minimum or maximum number of instances in each Auto Scaling group and that group will never goes below or above the given number of instances. In case the number of instances are fixed, its mean you mention desired capacity.

We can specify scaling policy also in order to fulfil demand, for example in case of termination of an instance a new instance will be launched. Auto scaling does not start or stop instance, it launches or terminates instances.

For better understanding, we can take one example:

Consider Auto Scaling group has a minimum size (number of instance) 1 and desired size is 2, and maximum capacity could be 4 instances. According to policy criteria we specify instances can vary as per the given range.

Benefits of Auto Scaling:

Here we see some benefits of cloud computing.

a. **Improved Fault Tolerance**: In case of unhealthy instance, Auto Scaling terminates it and launches a new healthy instance. The feature can be used for multiple AZ, if one AZ is not available, Auto Scaling can launch instance in another AZ.

b. **Improved Availability**: To handle application load, accurate number of instances are needed which can be attained with Auto Scaling.

c. **Improved Cost Management**: In a physical environment you need enough time and money to launch machines to handle the situation which is a big challenge. However, Auto Scaling can dynamically increase and decrease instances to manage application load or to maintain a fixed number of running instances.

Within Auto Scaling

Auto Scaling Group:

This group ensures that we have correct number of running instances to handle the load of application. The collection of instances called Auto Scaling Group.

Component of Auto Scaling Group:

a. **Groups**: Instances are organized into groups and it is treated a logical unit.

b. **Launching Configuration**: In launch configuration we specify, AMI ID, instance type, key pair, security groups and block device/s.

c. **Scaling Plans**: It tells when and how to scale.
   i. **Manual Scaling**: In this scaling plan, specify a change in the maximum, minimum or desired capacity of you Auto Scaling Group.
   ii. **Maintain Current Instance Level**: If periodic check of Auto Scaling finds an unhealthy instance within the Auto Scaling Group, it terminates and launches a new instance.
   iii. **Time and Date based scaling**: The scaling will perform automatically as a function of time and date.
   iv. **Demand based scaling**: Whenever utilization of instance resource reaches at its maximum level for a stipulated time, number of instances can be increased.

Auto Scaling Price:

Price is not a constraint for Auto Scaling. AWS does not charge any additional fee or amount for Auto Scaling, prices are only charged for the EC2 instances that it launches.

LAB Overview:

After completion of this lab you will be able to control Auto Scaling Configuration on the basis of CPU utilization, If CPU usage of EC2 Instance shoots up for a certain period of time, auto scale feature can add more number of machines as per configuration and it can continue until instances reach max numbers as given in the configuration. And exactly reduction in number of machines will happen, if average load of CPU goes down.

LAB Task Steps:

To create an Auto Scaling group based on CPU utilization metrics

1. Login into AWS Management Console and open the Amazon EC2 console
2. **Create Auto Scaling group**.

3. **Create a new launch configuration**
4. On the **Configure Auto Scaling group details** page, fill the configurational information:
    a. Enter **Group name**
    b. For **Group size**, type the desired capacity for your Auto Scaling group.
    c. Enter VPC name and subnet information.
5. **Configure scaling policies** page, fill the following information:

    a. Select **Use scaling policies**
    b. Specify the minimum and maximum size for your Auto Scaling group
    c. Specify your scale-out policy under **Increase Group Size**.
    d. Provide a max average CPU utilization for a certain period of time, if alarm triggers for the condition, auto scale will add instances as per configuration.
    e. Similarly specify your scale-in policy under **Decrease Group Size**.
    f. Choose **Review**.
    g. On the **Review** page, choose **Create Auto Scaling group**.

6. Use some steps to verify the scaling policies for configured Auto Scaling group.

a. Open Linux instance using putty if you are taking from Windows machine.
b. And ensure that “stress” packages is installed in Linux instance
    a. # rpm –qa stress
c. If stress rpm is not present, install it by using yum command

    # yum install stress –y

d. Now run stress command to increase load on Linux instance to get alarm triggered to increase running instances automatically through auto scale.

    # stress --cpu 100 --io 100 --timeout 300

    Try and watch, and monitor activities through instance monitoring services.