

Lesson 03 Demo 10

Configuring Manual and Dynamic Scaling

Objective: To configure manual and dynamic scaling for an application using Amazon Web Services (AWS) tools and services for optimized resource management and performance

Tools required: AWS Workspace

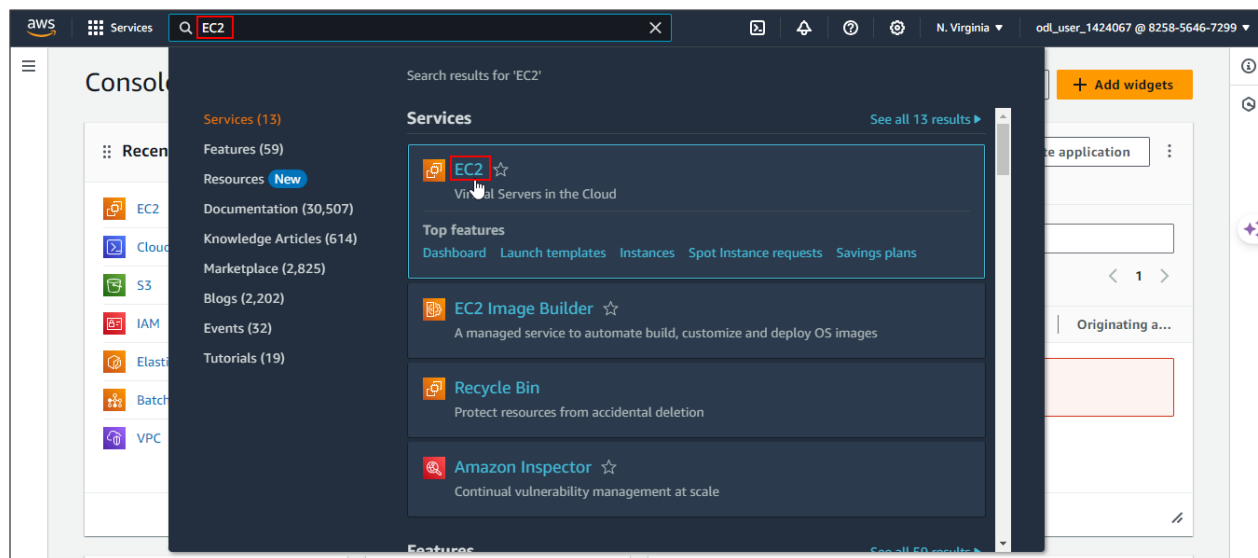
Prerequisites: Create an EC2 instance named S3

Steps to be followed:

1. Set up a predefined auto-scaling group
2. Set up EC2 Auto Scaling with a Load Balancer

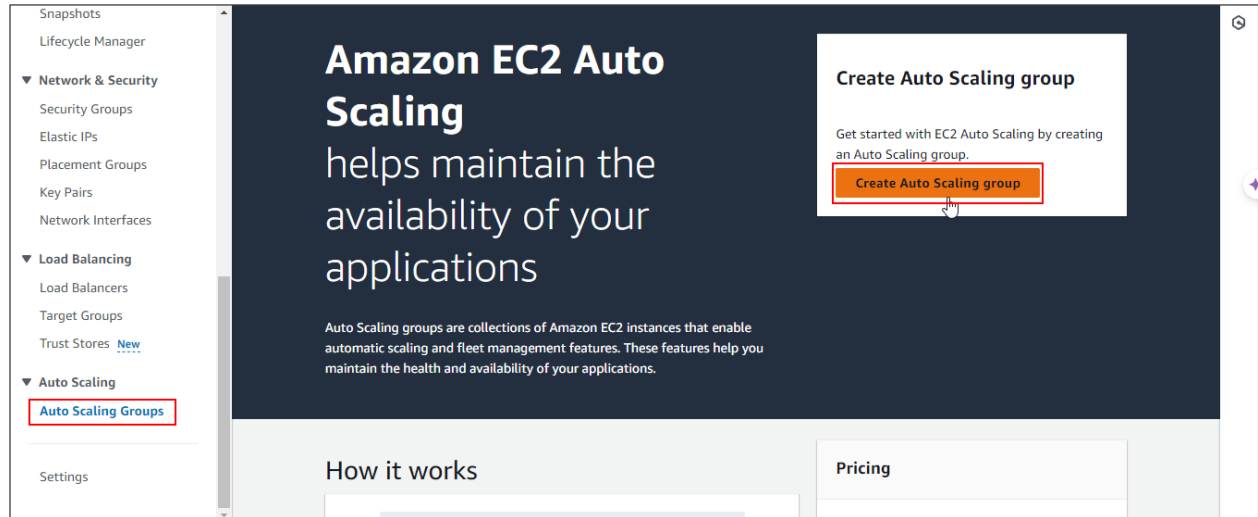
Step 1: Set up a predefined auto-scaling group

1.1 Navigate to the AWS console home dashboard, search for and click on **EC2**

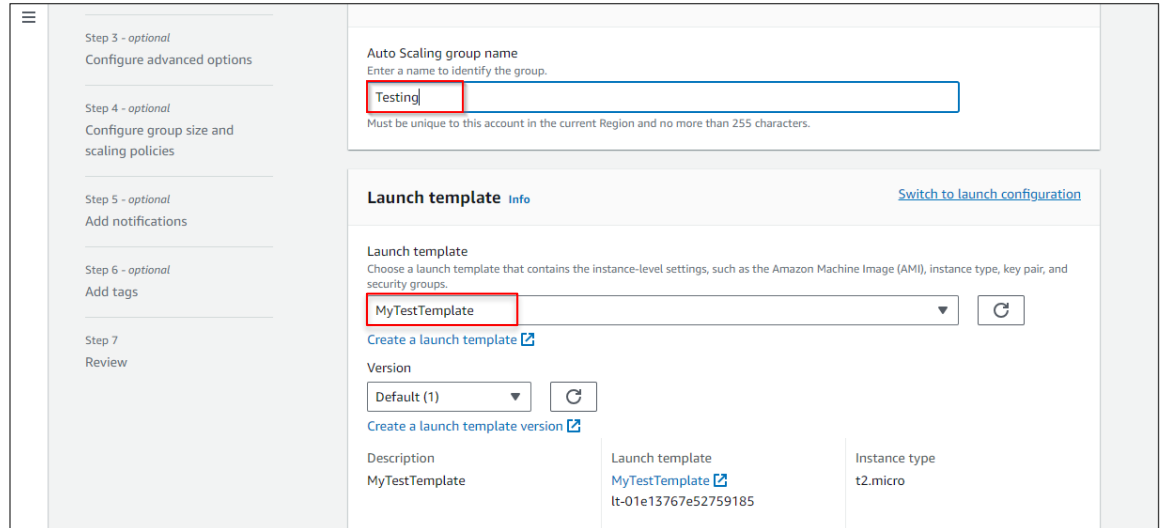


Note: Set the region to **US East (N. Virginia) us-east-1** in all demos

1.2 Navigate to Auto Scaling Groups in the Auto Scaling section, and click on Create Auto Scaling group



1.3 Add the name as Testing, select MyTestTemplate in the Launch template, and click Next



1.4 Select the availability zones and subnets as **us-east-1a** and **us-east-1b**, then click **Next**

Network [Info](#)

☒ **us-east-1a** | subnet-08219e2328d8d7c3b
172.31.16.0/20 Default

☒ **us-east-1b** | subnet-0081290ff8abfd41f
172.31.32.0/20 Default

☐ us-east-1c | subnet-0fb409c9b72c9bc00
172.31.0.0/20 Default

☐ us-east-1d | subnet-07353efa96c73ea4f
172.31.80.0/20 Default

☐ us-east-1e | subnet-0c67178539eae4e88
172.31.48.0/20 Default

☐ us-east-1f | subnet-0986291fa39c821f3
172.31.64.0/20 Default

Select Availability Zones and subnets

us-east-1a | subnet-08219e2328d8d7c3b X
172.31.16.0/20 Default

us-east-1b | subnet-0081290ff8abfd41f X
172.31.32.0/20 Default

[Create a subnet](#)

Cancel Skip to review Previous **Next**

1.5 Click on **Next**

Elastic Load Balancing monitors whether instances are available to handle requests. When it reports an unhealthy instance, EC2 Auto Scaling can replace it on its next periodic check.

☐ Turn on VPC Lattice health checks
VPC Lattice can monitor whether instances are available to handle requests. If it considers a target as failed a health check, EC2 Auto Scaling replaces it after its next periodic check.

Health check grace period [Info](#)
This time period delays the first health check until your instances finish initializing. It doesn't prevent an instance from terminating when placed into a non-running state.

300 seconds

Additional settings

Monitoring [Info](#)

☐ Enable group metrics collection within CloudWatch

Default instance warmup [Info](#)
The amount of time that CloudWatch metrics for new instances do not contribute to the group's aggregated instance metrics, as their usage data is not reliable yet.

☐ Enable default instance warmup

Cancel Skip to review Previous **Next**

1.6 Add the **Desired capacity**, **Minimum capacity**, and **Maximum capacity** as **2**, and click on **Next**

Choose instance launch options

Step 3 - optional
[Configure advanced options](#)

Step 4 - optional
Configure group size and scaling

Step 5 - optional
[Add notifications](#)

Step 6 - optional
[Add tags](#)

Step 7
[Review](#)

Group size Info
Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.

Desired capacity type
Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances) ▼

Desired capacity
Specify your group size.

Scaling Info
You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits
Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity **Max desired capacity**

Equal or less than desired capacity Equal or greater than desired capacity

Automatic scaling - optional

Control your Auto Scaling group's availability during instance replacement events. This includes health checks, instance refreshes, maximum instance lifetime features and events that happen automatically to keep your group balanced, called rebalancing events.

Choose a replacement behavior depending on your availability requirements

Mixed behavior (selected)
☒ **No policy**
For rebalancing events, new instances will launch before terminating others. For all other events, instances terminate and launch at the same time.

Prioritize availability
☐ **Launch before terminating**
Launch new instances and wait for them to be ready before terminating others. This allows you to go above your desired capacity by a given percentage and may temporarily increase costs.

Control costs
☐ **Terminate and launch**
Terminate and launch instances at the same time. This allows you to go below your desired capacity by a given percentage and may temporarily reduce availability.

Flexible
☐ **Custom behavior**
Set custom values for the minimum and maximum amount of available capacity. This gives you greater flexibility in setting how far below and over your desired capacity EC2 Auto Scaling goes when replacing instances.

Instance scale-in protection
Scale-in protection prevents newly launched instances from being terminated by scaling activities. Make sure to remove scale-in protection for the group or individual instances when instances are ready to be terminated.

☐ Enable instance scale-in protection

Cancel Skip to review Previous **Next**

1.7 Review the steps, and click **Create Auto Scaling group**

Instance scale-in protection

Instance scale-in protection

☐ Enable instance protection from scale in

Step 5: Add notifications Edit

Notifications

No notifications

Step 6: Add tags Edit

Tags (0)

Key	Value	Tag new instances
No tags		

Cancel Previous Create Auto Scaling group

EC2 > Auto Scaling groups

Auto Scaling groups (1) [Info](#) Refresh Launch configurations Launch templates Actions Create Auto Scaling group

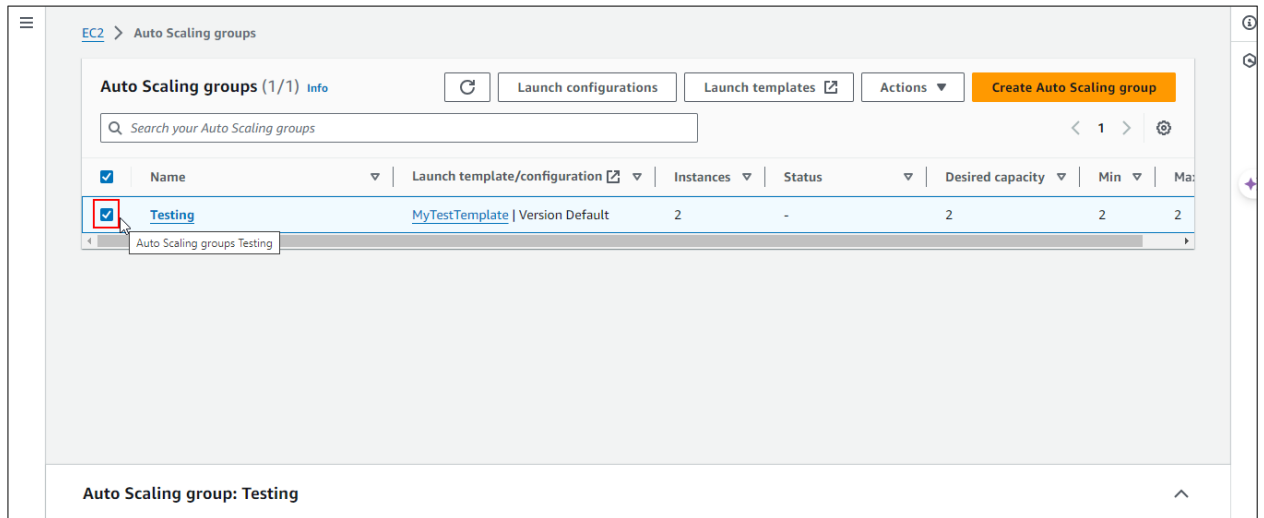
<input type="checkbox"/>	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max
<input type="checkbox"/>	Testing	MyTestTemplate Version Default	2	-	2	2	2

0 Auto Scaling groups selected

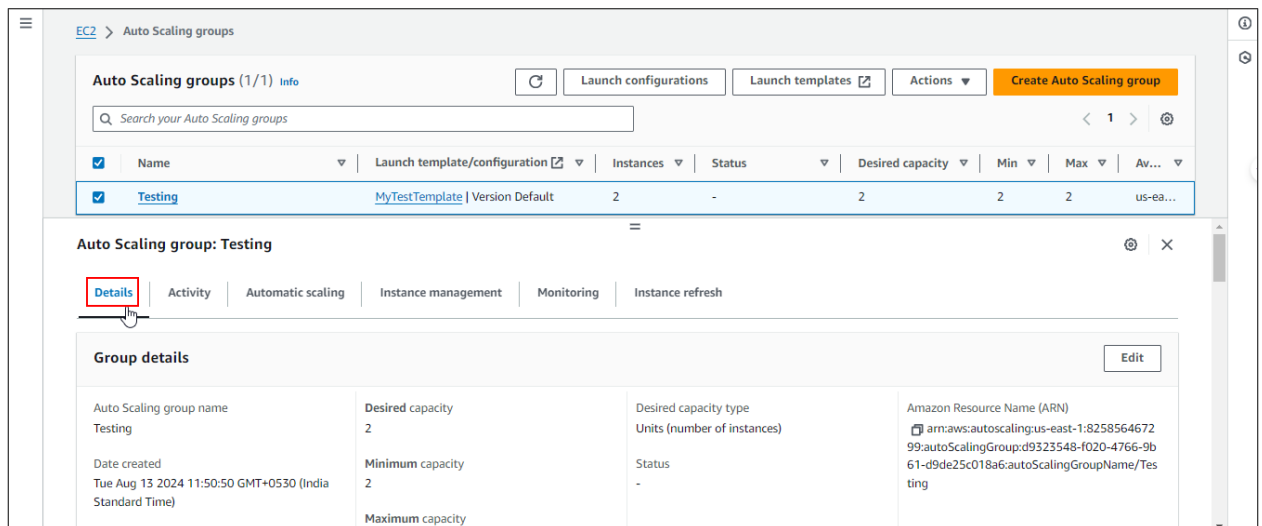
Auto-scaling groups have been created successfully.

Step 2: Set up EC2 Auto Scaling with a Load Balancer

2.1 Select the previously created auto-scaling group as shown:



2.2 Click on **Details** to verify the group details



2.3 Navigate to the **Instances**, and click **Launch instances** to create a new instance named **Testing**

The screenshot shows the AWS Management Console's EC2 Instances page. The left sidebar contains navigation links for EC2 Dashboard, EC2 Global View, Events, Console-to-Code, and a list of instance categories including Instances, Images, and Elastic Block Store. The 'Instances' link is highlighted. The main panel shows 'Instances (2)' with a table of two running instances. The 'Launch instances' button in the top right is highlighted with a red box.

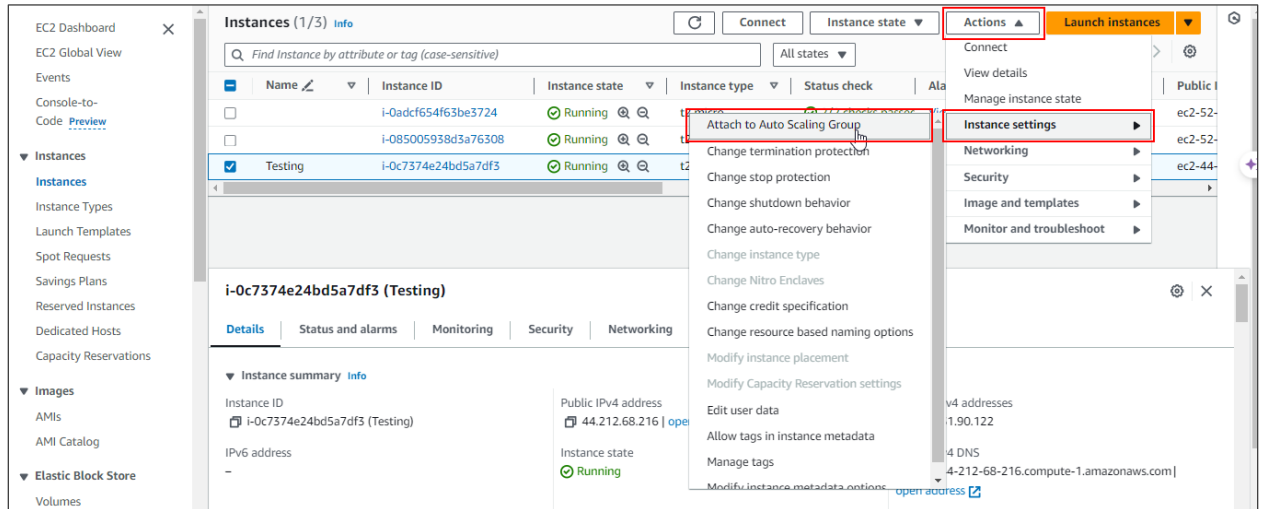
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
	i-0adcf654f63be3724	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-52-
	i-085005938d3a76308	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-52-

The screenshot shows the AWS Management Console's EC2 Instances page after launching a new instance. The table now contains three instances. The third instance, named 'Testing', is in a 'Pending' state. The 'Launch instances' button is still highlighted.

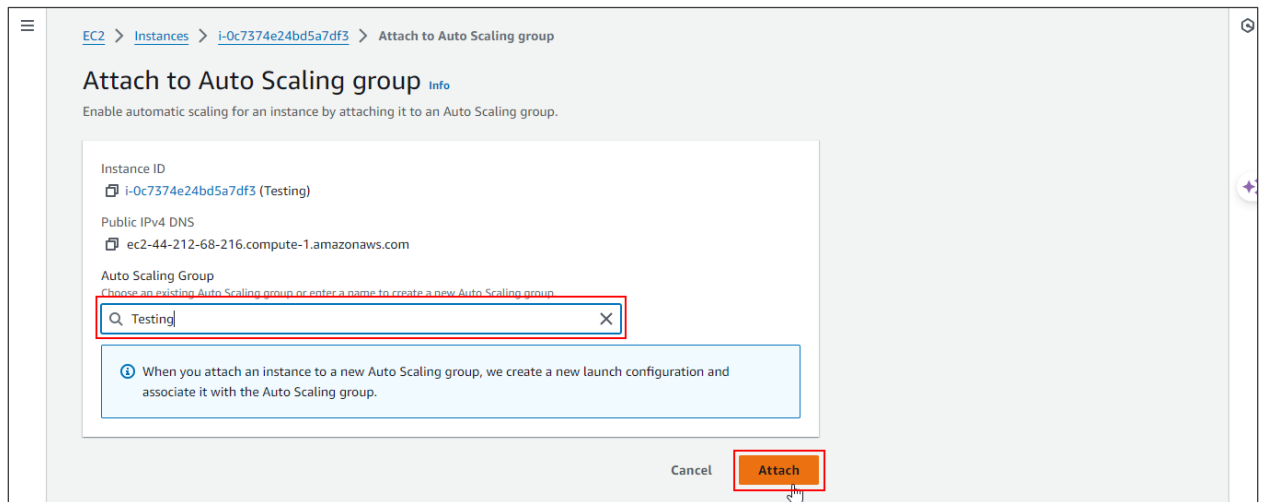
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
	i-0adcf654f63be3724	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-52-
	i-085005938d3a76308	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-52-
Testing	i-0c7374e24bd5a7df3	Pending	t2.micro	-	View alarms +	us-east-1d	ec2-44-

Instances have been created successfully; refer to the previous demos for instructions on creating instances.

2.4 Click on **Actions**, then select **Instance settings**, and choose **Attach to Auto Scaling Group**



2.5 Select the **Auto Scaling Group** name **Testing**, and click **Attach**



By following these steps, you have successfully configured manual and dynamic scaling for your application using AWS tools and services, ensuring optimized resource management and performance.