

1. Why the Auto Scaling feature is significant in cloud computing?

Autoscaling lets you automatically adjust the number or the lifecycle state of compute instances in an instance pool. This helps you provide consistent performance for your end users during periods of high demand, and helps you reduce your costs during periods of low demand.

2. Describe the following ?

1. Metric-based autoscaling

2. Schedule-based autoscaling

- **Metric-based autoscaling:** An autoscaling action is triggered when a performance metric meets or exceeds a threshold.

Metric-based autoscaling relies on performance metrics that are collected by the Monitoring service, such as CPU utilization. These performance metrics are aggregated into one-minute time periods and then averaged across all instances in the instance pool.

- **Schedule-based autoscaling:** Autoscaling events take place at the specific times that you schedule. Schedule-based autoscaling lets you improve the availability of your workloads by scheduling capacity ahead of anticipated load. If you run your workload on a managed instance group (MIG), you can schedule a required number of virtual machine (VM) instances for recurring load patterns as well as one-off events.

3. Demonstrate Metric based autoscaling or Schedule based auto scaling to cater your organizations business requirements. (Specify the requirements)

1. We will be doing load balancer with auto scaling , so first create load balancer , which we have already discussed in lab3
2. Next we will start Schedule Based auto scaling as metric for scaling , below screen shots demonstrates how to do



\

1. Open launch configuration from side panel and configure accordingly

Before that create AMI of instance by opening actions on existing instance -> images and templates- > create Image-> configure->done

EC2 > Launch configurations > Create launch configuration

Create launch configuration [Info](#)

 Instead of using launch configurations to create your EC2 Auto Scaling groups, we recommend that you use launch templates and make use of the Auto Scaling guidance option. For more information on migrating launch configurations and using launch templates, [see the documentation](#) 

Create launch template

Launch configuration name

Name

Amazon machine image (AMI) [Info](#)

AMI

Choose an AMI

Configuration :

Launch configuration name

Name

Amazon machine image (AMI) [Info](#)

AMI

Instance type [Info](#)

Instance type

Successfully created launch configuration: 2147228lab10

EC2 > Launch configurations

Launch configurations (1) [Info](#)

 Actions  Copy to launch template 

< 1 > 

<input type="checkbox"/>	Name	AMI ID	Instance type	Spot price	Creation time
<input type="checkbox"/>	2147228lab10	ami-01ef84b5f40...	t2.micro	-	Wed Nov 09 2022 22:57:15 GMT+0530 (India Stan...

Select a launch configuration above

2. After creating above step , select name ->Actions-> choose launch template
Do necessary configuration

Choose launch template or configuration [Info](#)

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group. If you currently use launch configurations, you might consider migrating to launch templates.


Name

Auto Scaling group name
Enter a name to identify the group.

2147228lab4

Must be unique to this account in the current Region and no more than 255 characters.

Launch configuration [Info](#) [Switch to launch template](#)

 Instead of using launch configurations to create your EC2 Auto Scaling groups, we recommend that you use launch templates and make use of the Auto Scaling guidance option. For more information on migrating launch configurations and using launch templates, [see the documentation](#)

Launch configuration
Choose a launch configuration that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

2147228lab10

[Create a launch configuration](#)

Launch configuration	AMI ID	Date created
----------------------	--------	--------------

Choosing zones vpc (we can create vpc also)

Network [Info](#)

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC
Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-043e0d8c01d28866c

172.31.0.0/16 Default

[Create a VPC](#)

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets

us-east-1a | subnet-08496a93b6fd32b43

172.31.32.0/20 Default

[Create a subnet](#)

Cancel

Previous

Skip to review

Next

3. Now choose our created load balancer -> attach to existing -> choose classic load balancer -> don't forget to check ELB which I forgot to mention

Configure advanced options Info

Choose a load balancer to distribute incoming traffic for your application across instances to make it more reliable and easily scalable. You can also set options that give you more control over health check replacements and monitoring.

Load balancing - optional Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

☐ No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

☒ Attach to an existing load balancer
Choose from your existing load balancers.

☐ Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to an existing load balancer

Select the load balancers that you want to attach to your Auto Scaling group.

☐ Choose from your load balancer target groups
This option allows you to attach Application, Network, or Gateway Load Balancers.

☒ Choose from Classic Load Balancers

Classic Load Balancers

Select Classic Load Balancers

2147228lab4 Classic Load Balancer

Health checks - optional

Health check type Info

EC2 Auto Scaling automatically replaces instances that fail health checks. If you enabled load balancing, you can enable ELB health checks in addition to the EC2 health checks that are always enabled.

☒ EC2 ☐ ELB

Health check grace period

The amount of time until EC2 Auto Scaling performs the first health check on new instances after they are put into service.

200 seconds

2147228lab4 created successfully

EC2 > Auto Scaling groups

Auto Scaling groups (1) Info

Search your Auto Scaling groups

< 1 >

<input type="checkbox"/>	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max
<input type="checkbox"/>	2147228lab4	2147228lab10	0	Updating capacity...	1	1	1

3. after creating , scroll down to scheduling -> click create schedule action -> add new
Do necessary configuration

Do note that time is in UTC so adjust time by converting local time to utc for scheduling

Create scheduled action

Name

Provide at least one value for Desired, Min, or Max Capacity

Desired capacity Min Max

Recurrence

Once ▼

Time zone

Etc/UTC ▼

Current time in selected time zone is 2022-11-09/17:32 UTC

Specific start time

Schedule a specific date and time for the first scheduled action to run. Interpreted in recurrence time zone: Etc/UTC

YYYY/MM/DD 00:00 Etc/UTC

I choose 3 instance to be created (desired capacity) max :5 min:1

Create scheduled action

Name

2147228lab4

Provide at least one value for Desired, Min, or Max Capacity

Desired capacity Min Max

3 1 5

Recurrence

Once ▼

Time zone

Etc/UTC ▼

Current time in selected time zone is 2022-11-09/17:36 UTC

Specific start time

Schedule a specific date and time for the first scheduled action to run. Interpreted in recurrence time zone: Etc/UTC

2022/11/09 15:36 Etc/UTC

[Learn more about scheduled scaling](#)

Cancel **Create**

I have set my ec3 to launch 3 minutes after 15:33 so we can see out 3 instance will be created at 15:36 UTC

Now u can see our instance is running : newly created instance

The screenshot displays the 'Instance summary' page for instance **i-0ec8404b7638a56d3**. The instance is in a **Running** state. Key details include:

- Public IPv4 address:** 54.237.212.14
- Private IPv4 addresses:** 172.31.40.58
- Public IPv4 DNS:** ec2-54-237-212-14.compute-1.amazonaws.com
- Instance type:** t2.micro
- VPC ID:** vpc-043e0d8c01d28866c
- Subnet ID:** subnet-08496a93b6fd32b43
- Auto Scaling Group name:** 2147228lab4

The bottom of the console shows a Windows taskbar with various application icons and a system clock indicating 11:11 PM on 11/9/2022.

We can see in below images that 3 new instance is added to our load balancer

The screenshot shows the 'Instances' tab for load balancer **2147228lab4**. It lists four instances, all with a status of **OutOfService**:

Instance ID	Name	Availability Zone	Status	Actions
i-0d2d85e409a7ab61c	2147228_lab4_scaling	us-east-1d	OutOfService	Remove from Load Balancer
i-0d313e3d4bd8edfd0		us-east-1a	OutOfService	Remove from Load Balancer
i-0cf13716c4389231b		us-east-1a	OutOfService	Remove from Load Balancer
i-0ec8404b7638a56d3		us-east-1a	OutOfService	Remove from Load Balancer

Our 3 new instance created automatically at 15:36 UTC apart from one instance already created

<input type="checkbox"/>	2147228_lab4...	i-0d2d85e409a7ab61c	Running	🔍	t2.micro	2/2 checks passed	No alarms	+	us-east-1d
<input type="checkbox"/>	-	i-0ec8404b7638a56d3	Running	🔍	t2.micro	2/2 checks passed	No alarms	+	us-east-1a
<input type="checkbox"/>	-	i-0e255d3b081e08820	Stopped	🔍	t2.micro	-	No alarms	+	us-east-1c
<input type="checkbox"/>	-	i-0d313e3d4bd8edfd0	Running	🔍	t2.micro	2/2 checks passed	No alarms	+	us-east-1a
<input type="checkbox"/>	-	i-0cf13716c4389231b	Running	🔍	t2.micro	2/2 checks passed	No alarms	+	us-east-1a

The following instances are attached to an Auto Scaling group:

- i-0ec8404b7638a56d3 (2147228lab4)
- i-0d313e3d4bd8edfd0 (2147228lab4)
- i-0cf13716c4389231b (2147228lab4)