

**Ismail basheer**  
**2348521**

## **Significance of Auto Scaling in Cloud Computing**

Auto Scaling is a cloud computing feature that automatically adjusts the number of compute resources (such as virtual machines) in response to changes in demand. This ensures that applications have the right amount of resources at any given time, improving performance and cost efficiency.

- Cost Efficiency: Auto Scaling ensures that you only pay for the resources you need by scaling down during low demand periods and scaling up during high demand periods.
- Improved Availability: By automatically adding resources when demand increases, Auto Scaling helps maintain application availability and prevents performance degradation.
- Optimized Performance: It ensures applications have sufficient resources to handle incoming traffic, maintaining optimal performance and user experience.
- Fault Tolerance: Auto Scaling can replace failed instances, ensuring that the overall system remains healthy and operational.
- Reduced Manual Intervention: It automates the process of scaling resources, reducing the need for manual monitoring and intervention.

## **Types of Auto Scaling**

### **Metric-based Autoscaling**

Metric-based auto scaling adjusts the number of compute resources based on specific performance metrics, such as CPU utilization, memory usage, or network traffic. When the metrics cross predefined thresholds, the system automatically scales resources up or down.

#### **How it Works:**

1. Define Metrics: Set performance metrics that will trigger scaling actions, such as average CPU utilization or request count per instance.
2. Set Thresholds: Establish threshold values for these metrics that will trigger scaling actions.
3. Monitoring: Continuously monitor the specified metrics.
4. Scaling Actions: When metrics exceed or fall below the defined thresholds, the system triggers scaling actions to add or remove resources.

#### **Benefits:**

- Responsive: Quickly adapts to real-time changes in demand.
- Efficient Resource Utilization: Ensures resources are allocated based on actual usage patterns.

### **Schedule-based Autoscaling**

Schedule-based auto scaling adjusts the number of compute resources based on a predefined schedule. This method is useful when the demand is predictable, such as increasing resources during business hours and reducing them after hours.

#### **How it Works:**

1. Define Schedules: Create a schedule specifying when to scale resources up or down.
2. Set Scaling Actions: Specify the number of resources to add or remove at each scheduled time.
3. Automated Execution: The system automatically adjusts resources according to the schedule.

#### **Benefits:**

- Predictable Demand Management: Ideal for applications with predictable usage patterns.
- Cost Savings: Ensures resources are not over-provisioned during low-demand periods.

### 3.CREATING THE AUTO-SCALING GROUP

The screenshot shows the 'Create launch template' page in the AWS EC2 console. The left sidebar lists 'Launch templates' and 'Create launch template'. The main area has a 'Summary' section on the right and a 'Launch template name and description' section on the left.

**Launch template name and description**

- Launch template name - required**: MYTMLPT
- Template version description**: A prod webserver for MyApp
- Auto Scaling guidance**: Info
- Provide guidance to help me set up a template that I can use with EC2 Auto Scaling**:
- Template tags**
- Source template**

**Launch template contents**

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

**Summary**

- Software Image (AMI)**: Amazon Linux 2023 AMI  
ami-0b72821e2f351e396
- Virtual server type (instance type)**: t2.micro
- Firewall (security group)**: launch-wizard-1
- Storage (volumes)**: 1 volume(s) - 8 GiB

**Free tier**: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

**Create launch template**

The screenshot shows the 'Create launch template' page in the AWS EC2 console, focusing on advanced configuration options. The left sidebar lists 'Launch templates' and 'Create launch template'. The main area has a 'Summary' section on the right and several sections on the left.

**Don't include in launch template**

- Nitro Enclaves**: Nitro Enclaves are not compatible with instance types that have less than 2 vCPUs.
- License configurations**: Info
- Specify CPU options**: The selected instance type does not support CPU options.
- Metadata accessible**: Info
- Metadata IPv6 endpoint**: Info
- Metadata version**: Info
- Metadata response hop limit**: Info
- Allow tags in metadata**: Info
- User data - optional**: Info

**Summary**

- Software Image (AMI)**: Amazon Linux 2023 AMI  
ami-0b72821e2f351e396
- Virtual server type (instance type)**: t2.micro
- Firewall (security group)**: launch-wizard-1
- Storage (volumes)**: 1 volume(s) - 8 GiB

**Free tier**: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

**Create launch template**

You are signed in as 2348... | Home | Launch AWS Academy | Create Auto Scaling group | Create launch template | christ university Mail | +

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateTemplate:autoScalingGuidance=true

Services | Search [Alt+S]

Metadata response hop limit: [Info](#)  
Don't include in launch template

Allow tags in metadata: [Info](#)  
Don't include in launch template

User data - optional: [Info](#)  
Upload a file with your user data or enter it in the field.

```
yum update -y  
yum install -y httpd  
systemctl start httpd  
systemctl enable httpd  
Echo "<h1>Hello World from ${hostname}-f</h1>">/var/www/html/index.html
```

User data has already been base64 encoded

Software Image (AMI): Amazon Linux 2023 AMI ami-0b72821e2f351e396

Virtual server type (instance type): t2.micro

Firewall (security group): launch-wizard-1

Storage (volumes): 1 volume(s) - 8 GiB

**Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

You are signed in as 2348... | Home | Launch AWS Academy | Create Auto Scaling group | SecurityGroups | EC2 | Create launch template | christ university Mail | +

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateAutoScalingGroup:

Services | Search [Alt+S]

**Instance maintenance policy:** [Info](#)  
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**

**No policy**  
For rebalancing events, new instances will launch before terminating others. For all other events, instances terminate and launch at the same time.

**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

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the 'Instance maintenance policy' configuration step in the AWS Auto Scaling wizard. It includes sections for 'Instance maintenance policy' (info), 'Choose a replacement behavior depending on your availability requirements' (Mixed behavior, Prioritize availability, Control costs, Flexible), and 'Instance scale-in protection' (checkbox for 'Enable instance scale-in protection'). Navigation buttons at the bottom include 'Cancel', 'Skip to review', 'Previous', and 'Next'.

Successfully created an auto-scaling group 2348534AutoShop.

The screenshot shows the 'Group size' configuration step in the AWS Auto Scaling wizard. It includes sections for '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), and 'Scaling' (info). The 'Desired capacity type' section shows 'Units (number of instances)' set to 2. The 'Scaling limits' section shows 'Min desired capacity' at 1 and 'Max desired capacity' at 4. The 'Automatic scaling - optional' section shows 'No scaling policies' selected. Navigation buttons at the bottom include 'CloudShell', 'Feedback', '© 2024, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

You are signed in as [REDACTED] Home Launch AWS Academy Create Auto Scaling SecurityGroups | EC2 Create launch template christ university Mail

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateAutoScalingGroup:

aws Services Search [Alt+S]

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

### Load balancing 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 a new load balancer

Define a new load balancer to create for attachment to this Auto Scaling group.

Load balancer type Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the Load Balancing console.

Application Load Balancer HTTP, HTTPS

Network Load Balancer TCP, UDP, TLS

Load balancer name Name cannot be changed after the load balancer is created.

2348521-1

Load balancer scheme Scheme cannot be changed after the load balancer is created.

Internal

Internet-facing

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## Instance type is selected as t2.micro

You are signed in as [REDACTED] Home Launch AWS Academy Create Auto Scaling SecurityGroups | EC2 Create launch template christ university Mail

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateAutoScalingGroup:

aws Services Search [Alt+S]

### VPC

vpc-0e2f3196551c18677

#### Availability Zones and subnets

You must select a single subnet for each Availability Zone enabled. Only public subnets are available for selection to support DNS resolution.

us-east-1d subnet-0ff6634f23be93f29

us-east-1f subnet-01bab4cac24b68911

us-east-1e subnet-0f5036ec0b0cba84e

us-east-1c subnet-09d21217d3bb997f0

us-east-1b subnet-0e74059b272bfff838

us-east-1a subnet-096dfde4148d66ae1

#### Listeners and routing

If you require secure listeners, or multiple listeners, you can configure them from the Load Balancing console after your load balancer is created.

Protocol	Port	Default routing (forward to)
HTTP	80	Create a target group

New target group name An instance target group with default settings will be created.

2348521-1

#### Tags - optional

Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

You are signed in as [Profile] Home Launch AWS Academy Create Auto Scaling SecurityGroups | EC2 Create launch template christ university Mail N. Virginia vodlabs/user3346276=ismail.basheer@msam.christuniversity.in @... [Alt+S]

EC2 health checks  
**Always enabled**

Additional health check types - optional [Info](#)  
 Turn on Elastic Load Balancing health checks [Recommended](#)  
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.

**EC2 Auto Scaling will start to detect and act on health checks performed by Elastic Load Balancing.**  
To avoid unexpected terminations, first verify the settings of these health checks in the [Load Balancer console](#) [X](#)

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.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

You are signed in as [Profile] Home Launch AWS Academy Create Auto Scaling SecurityGroups | EC2 Create launch template christ university Mail N. Virginia vodlabs/user3346276=ismail.basheer@msam.christuniversity.in @... [Alt+S]

Step 1 [Create launch template](#)

Step 2 **Choose instance launch options** [Info](#)  
Choose the VPC network environment that your instances are launched into, and customize the instance types and purchase options.

**Instance type requirements** [Info](#) [Override launch template](#)  
You can keep the same instance attributes or instance type from your launch template, or you can choose to override the launch template by specifying different instance attributes or manually adding instance types.

Launch template	Version	Description
MYTMPLT <a href="#">Edit</a>	Default	-
lt-0e4bd5d3a6610c16a		

Instance type  
t2.micro

**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-0e2f3196551c18677 [Edit](#) [Create a VPC](#)

Availability Zones and subnets

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

You are signed in as ... | Home | Launch AWS Academy | Create Auto Scaling | SecurityGroups | EC2 | Create launch template | christ university Mail | +

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateAutoScalingGroup:

Configure advanced options

Step 4 - optional  
Configure group size and scaling

Step 5 - optional  
Add notifications

Step 6 - optional  
Add tags

Step 7  
Review

Enter a name to identify the group.  
2348521  
Must be unique to this account in the current Region and no more than 255 characters.

Launch template [Info](#)

For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

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

MYTMPLT

Create a launch template [\[x\]](#)

Version  
Default (1) [\[x\]](#)

Create a launch template version [\[x\]](#)

Description	Launch template	Instance type
-	MYTMPLT <a href="#">[x]</a> lt-0e4bd5d3a6610c16a	t2.micro
AMI ID	Security groups	Request Spot Instances
ami-0b72821e2f351e396	-	No
Key pair name	Security group IDs	

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

You are signed in as ... | Home | Launch AWS Academy | Auto Scaling groups | SecurityGroups | EC2 | Create launch template | christ university Mail | +

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#AutoScalingGroups:

EC2 > Auto Scaling groups

Auto Scaling groups (1) [Info](#)

Search your Auto Scaling groups

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Available...
2348521	MYTMPLT   Version Default	0	Updating capacity...	1	1	1	us-east-1a, ...

0 Auto Scaling groups selected

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

You are signed in as [Profile] Home Launch AWS Academy Load balancers | EC2 SecurityGroups | EC2 Create launch template christ university Mail +

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LoadBalancers:

aws Services Search [Alt+S]

AMIs AMI Catalog

Elastic Block Store Volumes Snapshots Lifecycle Manager

Network & Security Security Groups Elastic IPs Placement Groups Key Pairs Network Interfaces

Load Balancing Load Balancers Target Groups Trust Stores New

Auto Scaling Auto Scaling Groups Settings

CloudShell Feedback

EC2 > Load balancers

Load balancers (1/1)

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Filter load balancers

Name DNS name State VPC ID Availability Zones Type

2348521-1 2348521-1-1734247383.u... Provisioning vpc-0e2f3196551c186... 6 Availability Zones application

Last refreshed seconds ago export

Load balancer: 2348521-1

Listeners (1) Rules (1) Target groups (1) Info Targets

HTTP:80 1 rule Priority default Forward to target group  
Conditions (If)  
If no other rule applies

Instance 2348521-1 1 target | 0 unhealthy  
0 0 0 1 0

Target i-0cd92 Initiation progress

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS Load Balancers console. On the left, a navigation sidebar lists various AWS services. The main pane displays a single load balancer named '2348521-1'. The configuration is visualized as a flowchart: 'Listeners (1)' (HTTP:80) connects to 'Rules (1)' (Priority default: Forward to target group). This rule has a condition 'If no other rule applies'. The 'Target groups (1) Info' section shows one instance, '2348521-1', with 1 target and 0 unhealthy targets. Finally, the instance connects to a 'Target' (Amazon EC2 instance 'i-0cd92') which is in the 'Initiation progress' state. The top right of the main pane shows the last refresh time as 'Last refreshed seconds ago' and includes an 'export' button. The bottom right contains copyright information and links for privacy, terms, and cookie preferences.