



AWS
re:Start
LAB

Scaling and Load Balancing



WEEK 9





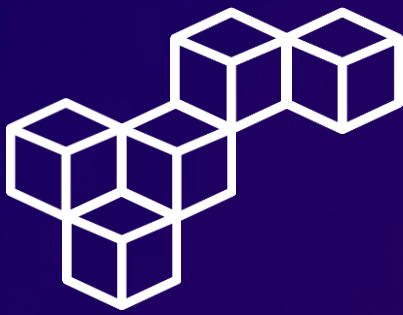
Overview

In this lab, you use the Elastic Load Balancing (ELB) and Amazon EC2 Auto Scaling to load balance and automatically scale your infrastructure. ELB automatically distributes incoming application traffic across multiple Amazon Elastic Compute Cloud (Amazon EC2) instances. ELB provides the amount of load balancing capacity needed to route application traffic to help you achieve fault tolerance in your applications.

Auto Scaling helps you maintain application availability and gives you the ability to scale your Amazon EC2 capacity out or in automatically according to conditions that you define. You can use auto scaling to help ensure that you are running your desired number of EC2 instances. Auto scaling can also automatically increase the number of EC2 instances during spikes in demand to maintain performance and can decrease capacity during lulls to reduce costs. Auto scaling is well suited to applications that have stable demand patterns or that experience hourly, daily, or weekly variability in usage.

Topics covered

- Create an AMI from an EC2 instance.
- Create a load balancer.
- Create a launch template and an Auto Scaling group.
- Configure an Auto Scaling group to scale new instances within private subnets.
- Use Amazon CloudWatch alarms to monitor the performance of your infrastructure.

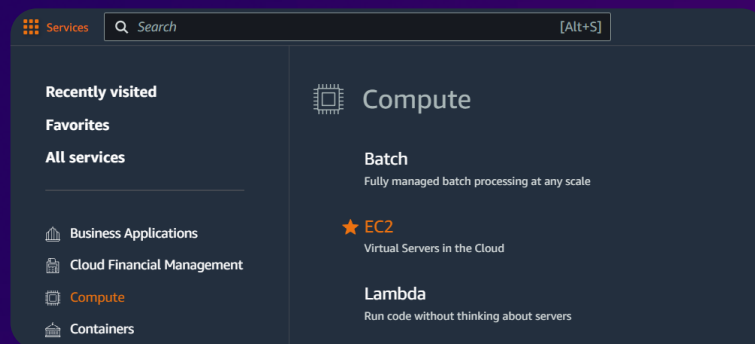


Task 1

Creating an AMI for auto scaling

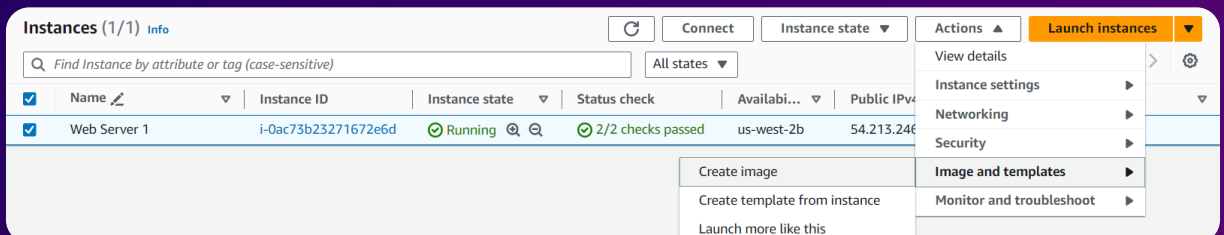
Step 1: Access the EC2 Management Console

Open the AWS Management Console, and select EC2.



Step 2: Create image

Navigate to the **Instances** section, select the **Web Server 1** instance, and choose Actions > Image and templates > [Create image](#) to create an AMI based on this instance.





Task 1

Creating an AMI for auto scaling

Step 3: Configure the image

In the **Create image** section, configure the following settings.

Instance ID


 i-0ac73b23271672e6d (Web Server 1)

Image name

Can't be modified after creation.

Image description - optional

Step 4: Review image

Navigate to the **AMIs** section, and review the newly created Amazon Machine Image **Web Server AMI**.

Amazon Machine Images (AMIs) (1) Info

Owned by me

Find AMI by attribute or tag

Recycle Bin

EC2 Image Builder

Actions

Launch instance from AMI

	Name	AMI name	AMI ID	Source	Status	Platform
<input type="checkbox"/>	Web Server AMI		ami-028a4387f02edf7bb	851725358876/Web Server AMI	Available	Linux/UNIX

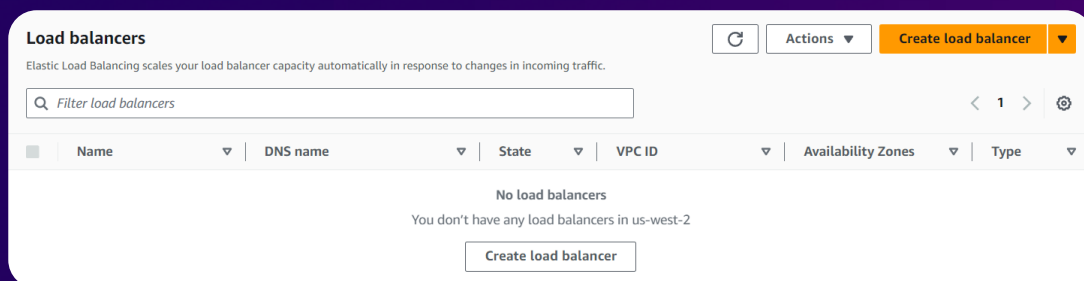


Task 2

Creating a load balancer

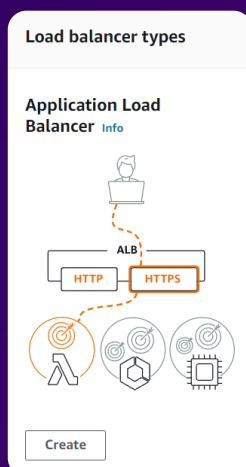
Step 1: Create load balancer

Navigate to the **Load balancers** section, and select [Create load balancer](#).



Step 2: Load balancer types

In the **Load balancer types** section, select **Application Load Balancer**, and choose [Create](#).





Task 2

Creating a load balancer

Step 3: Basic configuration

In the **Basic configuration** section, for **Load balancer name**, enter **LabELB**.

Basic configuration

Load balancer name
Name must be unique within your AWS account and can't be changed after the load balancer is created.

LabELB

Step 4: Network mapping

In the **Network mapping** section, configure the following settings.

Network mapping [Info](#)
The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC [Info](#)
Select the virtual private cloud (VPC) for your targets.

Lab VPC
vpc-0764acff325bcd2d9
IPv4 VPC CIDR: 10.0.0.0/16

Mappings [Info](#)
Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only.

☒ **us-west-2a (usw2-az1)**
Subnet
subnet-0b072b8d2b627ccb0 Public Subnet 1

☒ **us-west-2b (usw2-az2)**
Subnet
subnet-07a277895651a2413 Public Subnet 2



Task 2

Creating a load balancer

Step 5: Security groups

In the **Security groups** section, select the [Web Security Group](#), which permits HTTP access.

Security groups [Info](#)
A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#) [↗](#).

Security groups

Select up to 5 security groups ▼

Web Security Group X
sg-00422672ec413ba8e VPC: vpc-0764acff325bcd2d9

Step 6: Create target group

In the **Listeners and routing** section, choose the [Create target group](#) link. On the new **Target groups** tab, configure the following settings, and choose [Create target group](#).

Basic configuration
Settings in this section can't be changed after the target group is created.

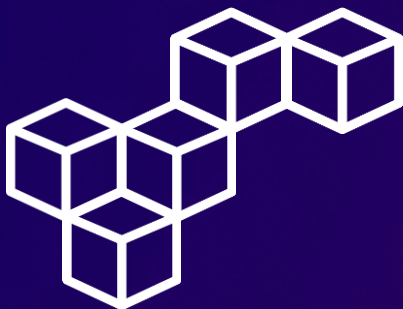
Choose a target type

☒ Instances

- Supports load balancing to instances within a specific VPC.
- Facilitates the use of [Amazon EC2 Auto Scaling](#) [↗](#) to manage and scale your EC2 capacity.

Target group name

lab-target-group



Task 2

Creating a load balancer

Step 7: Listeners and routing

In the **Listeners and routing** section, for the **Listener** tab, in the **Forward to** dropdown list, choose the **lab-target-group**.

Listeners and routing [Info](#)
A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

▼ Listener HTTP:80

Protocol

Port

Default action

[Info](#)

HTTP ▼

:

80

1-65535

Forward to

lab-target-group

Target type: Instance, IPv4

HTTP ▼

[Create target group](#) [↗](#)

Step 8: Review load balancer creation

Navigate to the **Load balancers** section, and review the newly created load balancer **LabELB**, and copy the DNS name.

Load balancers (1)							
Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.							
<input type="text" value="Filter load balancers"/>							
<input type="checkbox"/>	Name ▼	DNS name ▼	State ▼	VPC ID ▼	Availability Zones ▼	Type ▼	
<input type="checkbox"/>	LabELB	LabELB-313115008.us-west-2.elb.amazonaws.com	Active	vpc-0764acff325...	2 Availability Zones	application	



Task 3

Creating a launch template

Step 1: Create launch template

Navigate to the **Launch Templates** section, and select [Create launch template](#).

New launch template

Create launch template

Step 2: Launch template name and description

In the **Launch template name and description** section, configure the following settings.

Launch template name and description

Launch template name - *required*

lab-app-launch-template

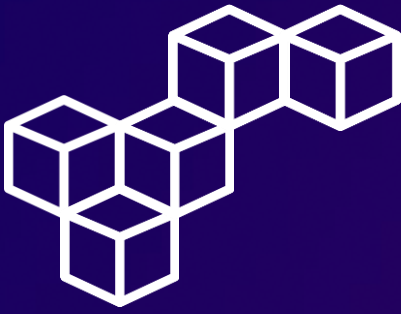
Template version description

A web server for the load test app

Auto Scaling guidance [Info](#)

Select this if you intend to use this template with EC2 Auto Scaling

☒ Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

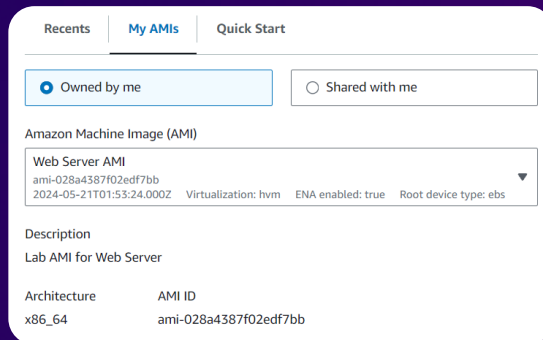


Task 3

Creating a launch template

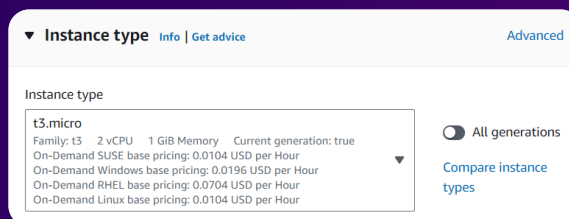
Step 3: Application and OS Images

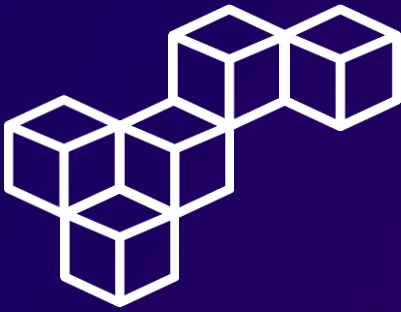
In the **Application and OS Images (Amazon Machine Image)** section, choose the **My AMIs** tab, and select the **Web Server AMI**.



Step 4: Instance type

In the **Instance type** section, choose **t3.micro**.





Task 3

Creating a launch template

Step 5: Key pair (login)

In the **Key pair (login)** section, for **Key pair name**, select **Don't include in launch template**.

▼ **Key pair (login)** [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name

Don't include in launch template ▼

[Create new key pair](#)

Step 6: Network settings

In the **Network settings** section, for **Security groups**, select the **Web Security Group**.

▼ **Network settings** [Info](#)

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Select existing security group

☐ Create security group

Security groups [Info](#)

Select security groups ▼

Web Security Group sg-00422672ec413ba8e ✕
VPC: vpc-0764acff325bcd2d9

[Compare security group rules](#)

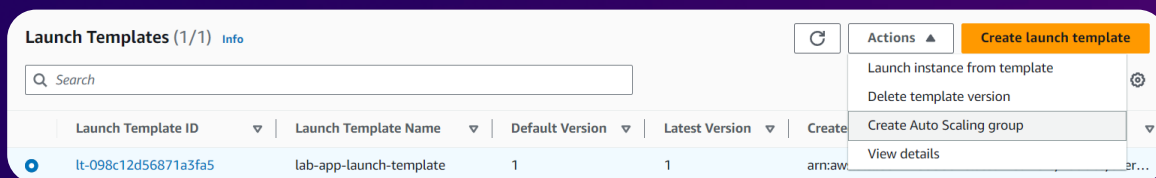


Task 4

Creating an Auto Scaling group

Step 1: Create Auto Scaling group

In the **Launch Templates** section, choose the newly created **lab-app-launch-template** launch template, and from the Actions dropdown list, select **Create Auto Scaling group**.



Step 2: Name

In the **Name** section, for **Auto Scaling group name**, enter **Lab Auto Scaling Group**.

Name

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

Lab Auto Scaling Group



Task 4

Creating an Auto Scaling group

Step 3: Network

In the **Network** section, configure the following settings.

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-0764acff325bcd2d9 (Lab VPC)
10.0.0.0/16

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

us-west-2a | subnet-0fa825c69a50781e0
(Private Subnet 1)
10.0.1.0/24

us-west-2b | subnet-0f8498ee975cfb5bb
(Private Subnet 2)
10.0.3.0/24

Step 4: Configure advanced options

For the **Load balancing** and **Attach to an existing load balancer** sections, configure the following settings.

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.

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

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.

Existing load balancer target groups
Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

lab-target-group | HTTP
Application Load Balancer: LabELB



Task 4

Creating an Auto Scaling group

Step 5: Health checks

In the **Health checks** section, for **Additional health check types**, select **Turn on Elastic Load Balancing health checks**.

Health checks
Health checks increase availability by replacing unhealthy instances. When you use multiple health checks, all are evaluated, and if at least one fails, instance replacement occurs.

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.

Step 6: Configure group size and scaling

For the **Group size** and **Scaling** sections, configure the following settings.

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.

2

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

2

Equal or less than desired capacity
Max desired capacity

4

Equal or greater than desired capacity



Task 4

Creating an Auto Scaling group

Step 7: Automatic scaling

In the **Automatic scaling** section, configure the following settings.

Automatic scaling - optional
Choose whether to use a target tracking policy [Info](#)
You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

☐ No scaling policies
Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

☒ Target tracking scaling policy
Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

Scaling policy name

Metric type [Info](#)
Monitored metric that determines if resource utilization is too low or high. If using EC2 metrics, consider enabling detailed monitoring for better scaling performance.

Average CPU utilization ▼

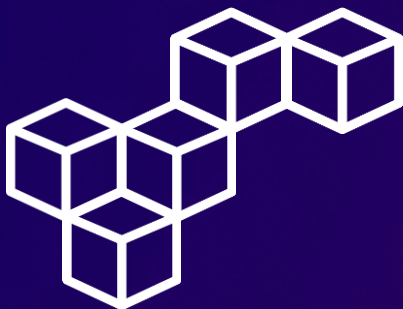
Target value

Step 8: Tags

In the **Add tags** section, configure the following Tags settings.

Tags (1)

Key	Value - optional	Tag new instances
<input type="text" value="Name"/>	<input type="text" value="Lab Instance"/>	<input checked="" type="checkbox"/>



Task 5

Verifying that load balancing is working

Step 1: Review Auto Scaling

Navigate to the **Instances** section, you should see two new instances named **Lab Instance**. These instances were launched by auto scaling.

Instances (3) Info								Refresh Connect Instance state Actions Launch instances	
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>				All states		< 1 > Settings			
<input type="checkbox"/>	Name ↗	Instance ID	Instance state ↕	Status check ↗	Availabi... ↕	Public IPv4 ... ↕	Security group name ↕		
<input type="checkbox"/>	Web Server 1	i-0ac73b23271672e6d	Running ↗ ↗	2/2 checks passed	us-west-2b	54.213.246.30	Web Security Group		
<input type="checkbox"/>	Lab Instance	i-09e46a18b548c41fc	Running ↗ ↗	2/2 checks passed	us-west-2b	–	Web Security Group		
<input type="checkbox"/>	Lab Instance	i-09ce109e2b099f304	Running ↗ ↗	2/2 checks passed	us-west-2a	–	Web Security Group		

Step 2: Review Target Groups

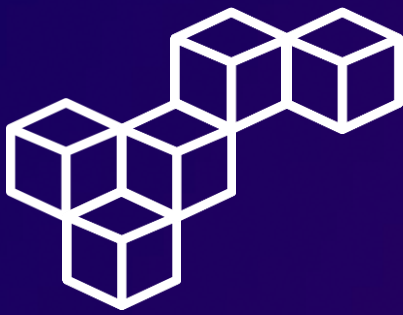
Navigate to the **Target Groups** section, and select the **lab-target-group** target group.

Target groups (1) [Info](#)

Filter target groups

< 1 >

</



Task 5

Verifying that load balancing is working

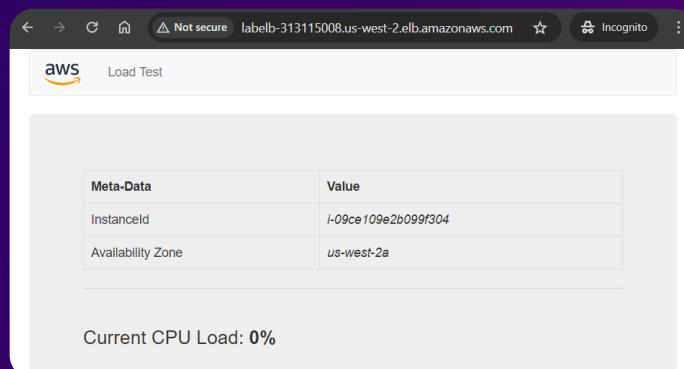
Step 3: Review Health status

In the **Registered targets** section, two Lab Instance targets should be listed for this target group. Review the **Healthy** Health status of both instances.

Registered targets (2) Info									
<small>Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.</small>									
<input type="text" value="Filter targets"/>									
<div>< 1 > ⚙</div>									
<input type="checkbox"/>	Instance ID	Name	Port	Zone	Health status	Launch time	Anomaly detection result		
<input type="checkbox"/>	i-09e46a18b548c41fc	Lab Instance	80	us-west-2b	Healthy	May 20, 2024, ...	Normal		
<input type="checkbox"/>	i-09ce109e2b099f304	Lab Instance	80	us-west-2a	Healthy	May 20, 2024, ...	Normal		

Step 4: Review Load Balancing

Access the instances launched in the Auto Scaling group using the load balancer DNS name. The Load Test application should appear in your browser, which means that the load balancer received the request, sent it to one of the EC2 instances, and then passed back the result.



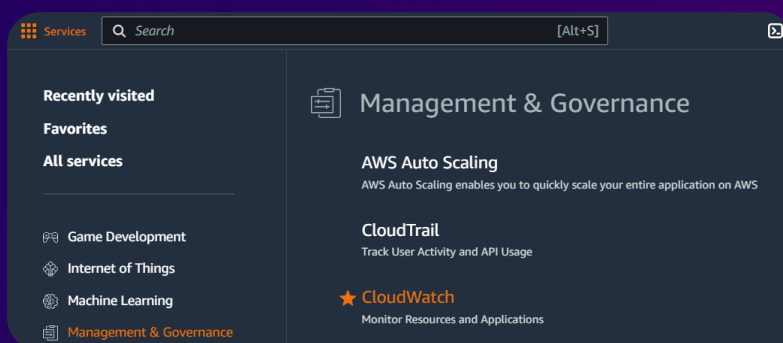


Task 6

Testing auto scaling

Step 1: Access the CloudWatch Console

In the AWS Management Console, select CloudWatch.



Step 2: Review CloudWatch alarms

Navigate to the **All alarms** section. Two alarms are displayed. The Auto Scaling group automatically created these two alarms. These alarms automatically keep the average CPU load close to 50 percent while also staying within the limitation of having 2–4 instances. The **AlarmHigh** alarm should have a **State** of **OK**.

Alarms (2)						
<input type="checkbox"/> Hide Auto Scaling alarms		Clear selection		Create composite alarm	Actions ▼	Create alarm
<input type="text" value="Search"/>		Alarm state: Any ▼	Alarm type: Any ▼	Actions status: Any ▼	< 1 > ⌂	
<input type="checkbox"/>	Name ▼	State ▼	Last state update (UTC) ▼	Conditions	Actions	
<input type="checkbox"/>	TargetTracking-Lab Auto Scaling Group-AlarmLow-cc9a3429-6237-410f-b266-2a2ef90b889f	In alarm	2024-05-21 03:06:45	CPUUtilization < 35 for 15 datapoints within 15 minutes	✔ Actions enabled	
<input type="checkbox"/>	TargetTracking-Lab Auto Scaling Group-AlarmHigh-a56b98c7-45ae-41b2-ae73-76f6d945b4aa	OK	2024-05-21 02:56:07	CPUUtilization > 50 for 3 datapoints within 3 minutes	✔ Actions enabled	

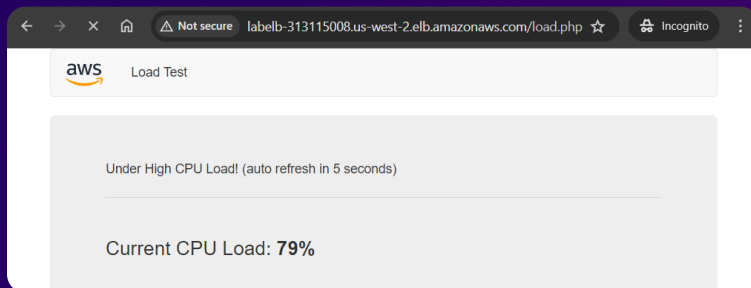


Task 6

Testing auto scaling

Step 3: Load Test

In the Load Test application, choose **Load Test**, this causes the application to generate high loads that raise the CPU level.



Step 4: Review the In alarm State

The **AlarmLow** alarm status should change to OK, and the **AlarmHigh** alarm status should change to **In alarm**.

Alarms (2)

☐ Hide Auto Scaling alarms

Clear selection

Create composite alarm

Actions ▾

Create alarm

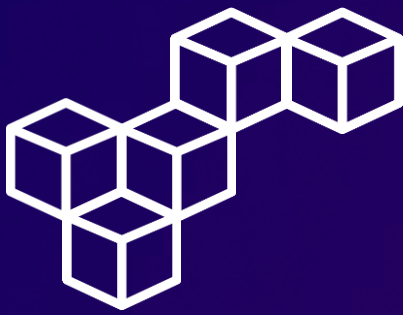
Alarm state: Any ▾

Alarm type: Any ▾

Actions status: Any ▾

< 1 >

<input type="checkbox"/>	Name ▾	State ▾	Last state update (UTC) ▾	Conditions	Actions
<input type="checkbox"/>	TargetTracking-Lab Auto Scaling Group-AlarmHigh-a56b98c7-45ae-41b2-ae73-76f6d945b4aa	In alarm	2024-05-21 03:19:07	CPUUtilization > 50 for 3 datapoints within 3 minutes	Actions enabled
<input type="checkbox"/>	TargetTracking-Lab Auto Scaling Group-AlarmLow-cc9a3429-6237-410f-b266-2a2ef90b889f	OK	2024-05-21 03:15:45	CPUUtilization < 35 for 15 datapoints within 15 minutes	Actions enabled

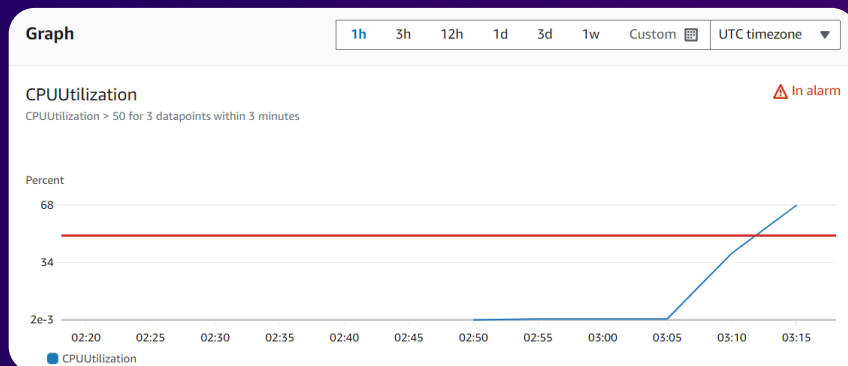


Task 6

Testing auto scaling

Step 5: Review the CPUUtilization Graph

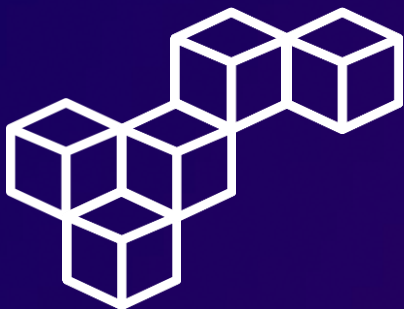
You should see the **AlarmHigh** chart indicating an increasing CPU percentage. Once it crosses the 50 percent line for more than 3 minutes, it initiates auto scaling to add additional instances.



Step 6: Review alarm response

Navigate to the **Instances** section, more than two instances named **Lab Instance** should now be running. Auto scaling created the new instances in response to the alarm.

Instances (4) Info								
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/> All states < 1 > ⚙️								
<input type="checkbox"/>	Name ↗	Instance ID	Instance state ▼	Status check ↗	Availabi... ▼	Public IPv4 ... ▼	Security group name ▼	
<input type="checkbox"/>	Web Server 1	i-0ac73b23271672e6d	Running 🔍 🔍	2/2 checks passed	us-west-2b	54.213.246.30	Web Security Group	
<input type="checkbox"/>	Lab Instance	i-09e46a18b548c41fc	Running 🔍 🔍	2/2 checks passed	us-west-2b	–	Web Security Group	
<input type="checkbox"/>	Lab Instance	i-06f66a0e6baa1a5c5	Running 🔍 🔍	2/2 checks passed	us-west-2a	–	Web Security Group	
<input type="checkbox"/>	Lab Instance	i-09ce109e2b099f304	Running 🔍 🔍	2/2 checks passed	us-west-2a	–	Web Security Group	



Task 7

Terminating the Web Server 1 instance

Step 1: Terminate instance

In the **Instances** section, terminate the **Web Server 1** instance.

Instances (1/4) Info

Find Instance by attribute or tag (case-sensitive)

All states

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Status check
<input checked="" type="checkbox"/>	Web Server 1	i-0ac73b23271672e6d	Running	2/2 checks passed
<input type="checkbox"/>	Lab Instance	i-09e46a18b548c41fc	Running	2/2 checks passed
<input type="checkbox"/>	Lab Instance	i-06f66a0e6baa1a5c5	Running	2/2 checks passed
<input type="checkbox"/>	Lab Instance	i-09ce109e2b099f304	Running	2/2 checks passed

Instance state

Stop instance

Start instance

Reboot instance

Hibernate instance

Terminate instance

Actions

Launch instances

13.246.30

Web Security Group

Web Security Group

Web Security Group

Web Security Group

Step 2: Review instance termination

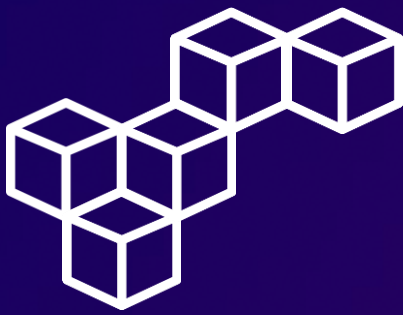
Review the instance termination. Notice how, while the **Web Server 1** instance was being terminated, another **Lab Instance** was launched by the Auto Scaling Group in response to the alarm.

Instances (5) Info

Find Instance by attribute or tag (case-sensitive)

All states

<input type="checkbox"/>	Name	Instance ID	Instance state	Status check	Availabi...	Public IPv4 ...	Security group name
<input type="checkbox"/>	Web Server 1	i-0ac73b23271672e6d	Terminated	-	us-west-2b	-	-
<input type="checkbox"/>	Lab Instance	i-0ce8863549876e450	Running	2/2 checks passed	us-west-2b	-	Web Security Group
<input type="checkbox"/>	Lab Instance	i-09e46a18b548c41fc	Running	2/2 checks passed	us-west-2b	-	Web Security Group
<input type="checkbox"/>	Lab Instance	i-06f66a0e6baa1a5c5	Running	2/2 checks passed	us-west-2a	-	Web Security Group
<input type="checkbox"/>	Lab Instance	i-09ce109e2b099f304	Running	2/2 checks passed	us-west-2a	-	Web Security Group



Conclusions

Application Load Balancers

Application Load Balancers operate at the application layer (HTTP/HTTPS), offering advanced routing and load balancing features.

Listeners

Listeners in load balancers check for connection requests using the configured protocol and port, directing traffic to the appropriate target group.

Target Groups

Target Groups define the destinations for traffic routed by load balancers, grouping EC2 instances or other resources for load distribution.

Launch templates

Launch templates standardize the configuration of EC2 instances, simplifying the process of launching and managing instances consistently.

Auto Scaling Groups

Auto Scaling Groups automatically adjust the number of EC2 instances based on demand, ensuring optimal performance and cost efficiency.



Cristhian Becerra



[cristhian-becerra-espinoza](#)



+51 951 634 354



cristhianbecerra99@gmail.com



Lima, Peru

