

Experiment – 4

- ❖ **Aim:** Learn the basic Elastic Load Balancing (ELB) services to load balance the infrastructure. To create an Amazon Machine Image (AMI) from a running instance. Create a load balancer. Automatically scale new instances. Create Amazon CloudWatch alarms and monitor the performance of your infrastructure.

➤ Task 1: Create an AMI for Auto Scaling

In this task, you will create an AMI from the existing Web Server 1. This will save the contents of the boot disk so that new instances can be launched with identical content.

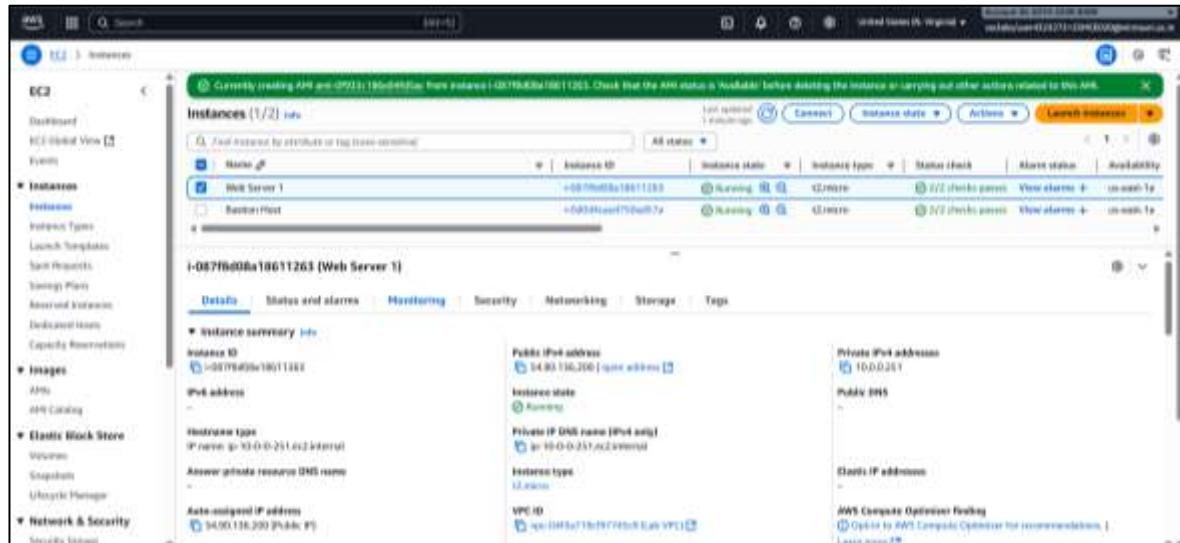
1. In the **AWS Management Console**, in the search box next to Services, search for and select **EC2**.
2. In the left navigation pane, choose **Instances**.

First, you will confirm that the instance is running.

3. Wait until the **Status Checks** for **Web Server 1** displays 2/2 checks passed. If necessary, choose refresh to update the status.

You will now create an AMI based upon this instance.

4. Select **Web Server 1**.



5. In the **Action** menu, choose **Image and templates > Create image**, then configure:
 - o **Image name:** WebServerAMI
 - o **Image description:** Lab AMI for Web Server

[EC2](#) > [Instances](#) > [i-0879d36a18611263](#) > Create Image

Create image [Info](#)

An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.

Image details								
Instance ID	i-0879d36a18611263 [Web Server 1]							
Image name	<input type="text" value="WebServerAMI"/>							
Image description - optional	<input type="text" value="Lab AMI for Web Server"/>							
<input checked="" type="checkbox"/> Reboot instance	When selected, Amazon EC2 reboots the instance so that it turns off when snapshots of the attached volumes are taken. This ensures data consistency.							
Instance volumes								
Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/dev/sda1	Create new snapshot	<input type="text" value="8"/>	SSD General Purpose (SSD)	<input type="text" value="3000"/>	<input type="text" value="10000"/>	<input checked="" type="checkbox"/> Enable	<input type="checkbox"/> Enable
Add volume								

6. Choose Create Image

A confirmation banner displays the **AMI ID** for your new AMI.

You will use this AMI when launching the Auto Scaling group later in the lab.

Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/dev/sda1	Create new snapshot	<input type="text" value="8"/>	SSD General Purpose (SSD)	<input type="text" value="3000"/>	<input type="text" value="10000"/>	<input checked="" type="checkbox"/> Enable	<input type="checkbox"/> Enable
Add volume								
<p>AMI ID: ami-0f3a18611263</p> <p>During the image creation process, Amazon EC2 creates a snapshot of each of the above volumes.</p> <p>Tags - optional: A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources to track your AWS costs.</p> <p><input checked="" type="radio"/> Tag image and snapshots together Tag the image and the snapshots with the same tag.</p> <p><input type="radio"/> Tag image and snapshots separately Tag the image and the snapshots with different tags.</p> <p>No tags associated with this resource.</p> <p>Add new tag</p> <p>You can add up to 10 more tags.</p>								
								Cancel Create Image

➤ Task 2: Create a Load Balancer

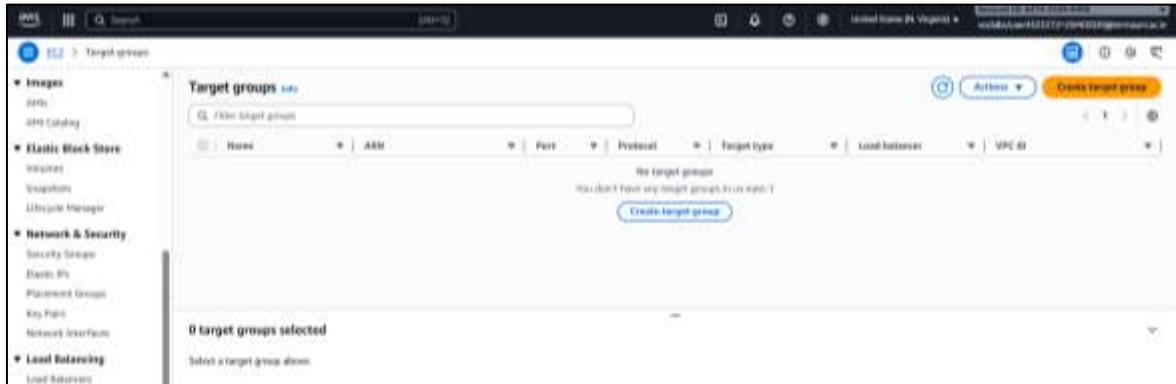
In this task, you will first create a target group and then you will create a load balancer that can balance traffic across multiple EC2 instances and Availability Zones.

10. In the left navigation pane, choose **Target Groups**.

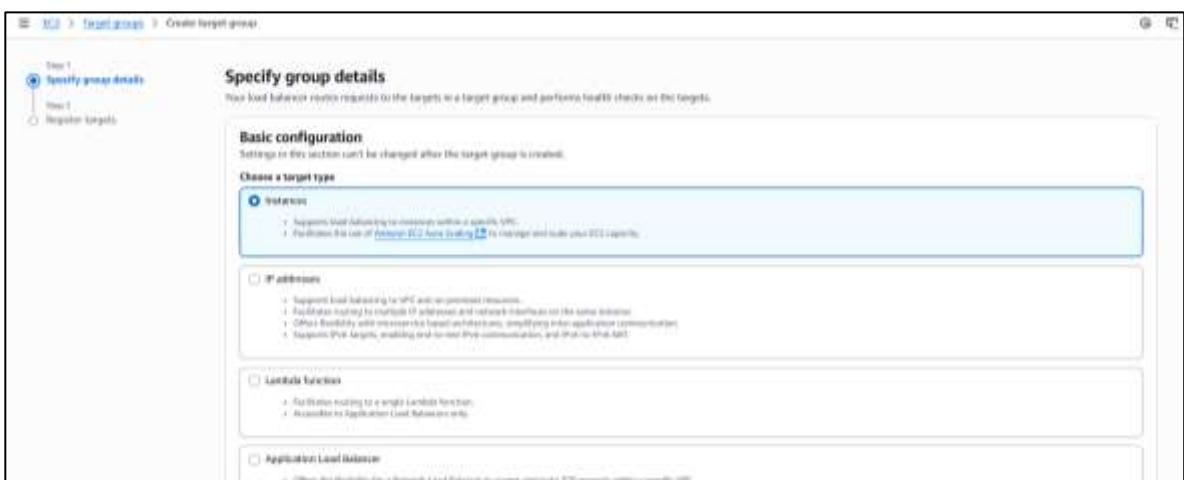
Analysis: Target Groups define where to send traffic that comes into the Load Balancer. The Application Load Balancer can send traffic to multiple Target Groups based upon the URL of the incoming request, such as having requests from mobile

apps going to a different set of servers. Your web application will use only one Target Group.

- Choose **Create Target Group**.



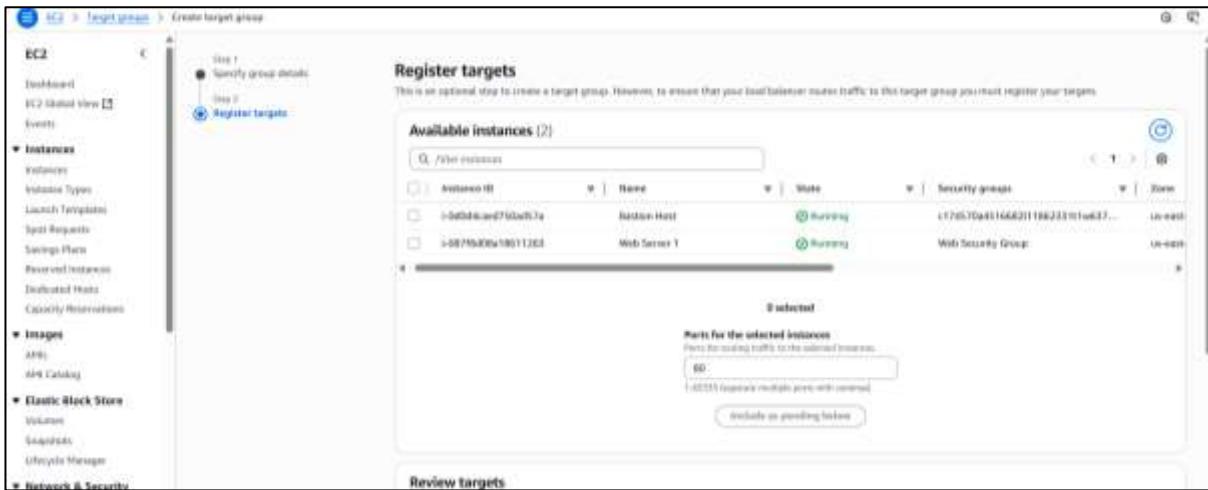
- Choose a target type: **Instances**



- **Target group name**, enter: LabGroup
- Select **Lab** VPC from the **VPC** drop-down menu.

Target group name	<input type="text" value="LabGroup"/>	
A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.		
Protocol	Protocol for load balancer-to-target communication. Can't be modified after creation.	
HTTP	Port	Port number where targets receive traffic. Can be overridden for individual targets during registration.
IP address type		
Only targets with the indicated IP address type can be registered to this target group.		
<input checked="" type="radio"/> IPv4	Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.	
<input type="radio"/> IPv6	Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). Learn more	
VPC	Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.	
<input type="text" value="vpc-04f5a719cf97745c9 (Lab VPC) 10.0.0.0/16"/>		Create VPC

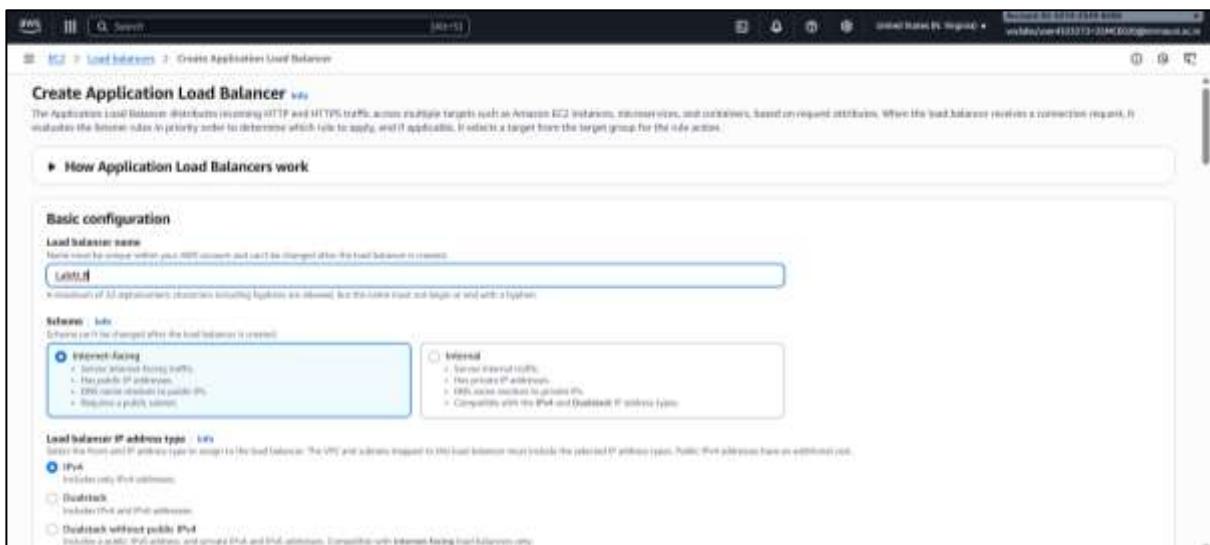
11. Choose **Next**. The **Register targets** screen appears.
12. Review the settings and choose **Create target group**



13. In the left navigation pane, choose **Load Balancers**.
14. At the top of the screen, choose **Create load balancer**.

Several different types of load balancer are displayed. You will be using an Application Load Balancer that operates at the request level (layer 7), routing traffic to targets — EC2 instances, containers, IP addresses and Lambda functions — based on the content of the request. For more information, see: [Comparison of Load Balancers](#).

15. Under **Application Load Balancer**, choose **Create**.
16. Under **Load balancer name**, enter: **LabELB**



17. Scroll down to the **Network mapping** section, then:

- For **VPC**, choose **Lab VPC**

You will now specify which subnets the Load Balancer should use. The load balancer will be internet facing, so you will select both Public Subnets.

- Choose the **first** displayed Availability Zone, then select **Public Subnet 1** from the Subnet drop down menu that displays beneath it.
- Choose the **second** displayed Availability Zone, then select **Public Subnet 2** from the Subnet drop down menu that displays beneath it.

You should now have two subnets selected: **Public Subnet 1** and **Public Subnet 2**.

The screenshot shows the 'Network mapping' section of a CloudFormation stack. At the top, it says 'VPC: Lab VPC'. Below that, there's a note about traffic targets and IP pools. Under 'Availability Zones and subnets', two subnets are selected: 'us-east-1a [subnet-0a34]' and 'us-east-1b [subnet-0a35]'. Both subnets are associated with 'Public Subnet 1'.

18. In the **Security groups** section:

- Choose the Security groups drop down menu and select **Web Security Group**
- Below the drop down menu, choose the **X** next to the default security group to remove it.

The **Web Security Group** security group should now be the only one that appears.

The screenshot shows the 'Security groups' section. A dropdown menu at the top is set to 'Web Security Group'. Below the dropdown, a list of security groups is shown. The 'default' group is listed with a checked checkbox, while the 'Web Security Group' is listed with an unchecked checkbox. The 'Web Security Group' row is highlighted with a blue background.

19. For the Listener HTTP:80 row, set the Default action to forward to **LabGroup**.

The screenshot shows the 'Listeners and routing' section of the CloudFront console. A single listener is defined for port 80 using the 'HTTP' protocol. The 'Default action' is set to 'Forward to target groups'. A target group named 'LabGroup' is selected, which is configured to use the 'HTTP' protocol and has a weight of 1 and a percentage of 100%. There is also an option to turn on target group stickiness.

20. Scroll to the bottom and choose **Create load balancer**

The load balancer is successfully created.

- o Choose **View load balancer**

The load balancer will show a state of provisioning. There is no need to wait until it is ready. Please continue with the next task.

The screenshot shows the 'LabELB' load balancer details page. The load balancer type is 'Application', and the status is 'Provisioning'. The VPC is 'vpc-0475a7191077745c', and the IP address type is 'IPv4'. The DNS name is 'LabELB-096209730.us-east-1.elb.amazonaws.com (A Record)'. The page includes tabs for 'Listeners and rules', 'Network mapping', 'Resource map', 'Security', 'Monitoring', 'Integrations', 'Attributes', 'Capacity', and 'Tags'. At the bottom, there are buttons for 'Manage rules', 'Manage listener', and 'Add listener'.

➤ **Task 3: Create a Launch Template and an Auto Scaling Group**

In this task, you will create a launch template for your Auto Scaling group. A launch template is a template that an Auto Scaling group uses to launch EC2 instances. When you create a launch template, you specify information for the instances such as the AMI, the instance type, a key pair, and security group.

21. In the left navigation pane, choose **Launch Templates**.
22. Choose **Create Launch Template**.

The screenshot shows the AWS EC2 Launch Templates page. At the top, it says "Compute" and "EC2 launch templates: Streamline, simplify and standardize instance launches". Below this is a descriptive text about using launch templates to automate instance launches, simplify permission policies, and enforce best practices across your organization. To the right, there's a call-to-action button "New launch template" with "Create launch template" inside it. On the left, under "Benefits and features", there are two boxes: "Streamline provisioning" (describing how it minimizes steps to provision instances) and "Simplify permissions" (describing how it creates shorter, easier-to-manage IAM policies). To the right, there's a "Documentation" section with links to "Documentation" and "API reference".

23. Configure the launch template settings and create it:
- o **Launch template name:** LabConfig

The screenshot shows the "Create launch template" wizard. The first step is "Launch template name and description". It has a field for "Launch template name - required" containing "LabConfig", a note that it must be unique to the account and have up to 128 characters, and a "Template version description" field containing "A prod webserver for MyApp". Below these are sections for "Auto Scaling guidance" (with a checkbox for "Provide guidance to help me set up a template that I can use with EC2 Auto Scaling" which is checked), "Template tags", and "Source template". The second step is "Launch template contents", with a note that it specifies the details of the launch template. The entire form is contained within a light gray border.

- Under **Auto Scaling guidance**, select Provide guidance to help me set up a template that I can use with EC2 Auto Scaling
- In the Application and OS Images (Amazon Machine Image) area, choose My AMIs.
- **Amazon Machine Image (AMI)**: choose Web Server AMI

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.

▼ Application and OS Images (Amazon Machine Image) - required [Info](#)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

Search our full catalog including 1000s of application and OS images

Recent | **My AMIs** | Quick Start

Owned by me | Shared with me

[Browse more AMIs](#) Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

WebServerAMI ami-0f933c186e94fd5ac 2025-09-29T11:24:36.000Z	Virtualization: hvm	ENI enabled: true	Root device type: ebs	Boot mode: uefi-prepared
---	---------------------	-------------------	-----------------------	--------------------------

Description
Lab AMI for Web Server

Architecture	AMI ID
x86_64	ami-0f933c186e94fd5ac

- **Instance type**: choose t2.micro
- **Key pair name**: choose vockey

▼ Instance type [Info](#) | Get advice

Advanced

Instance type

t2.micro

Family: t2 | 1 vCPU | 1 GiB Memory | Current generation: true | On-Demand Windows base pricing: 0.0162 USD per Hour | On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour | On-Demand SUSE base pricing: 0.0116 USD per Hour | On-Demand RHEL base pricing: 0.0126 USD per Hour | On-Demand Linux base pricing: 0.0116 USD per Hour

All generations

[Compare instance types](#)

Additional costs apply for AMIs with pre-installed software

▼ 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

vockey

[Create new key pair](#)

- **Firewall (security groups)**: choose Select existing security group
- **Security groups**: choose Web Security Group
- Scroll down to the **Advanced details** area and expand it.

Network settings

Subnet | [Info](#)

Don't include in launch template

When you specify a subnet, a network interface is automatically added to your template.

Create new subnet

Availability Zone | [Info](#)

Not applicable for EC2 Auto Scaling

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-0ac3ba75391be4cdf X

VPC: vpc-0af5a719cf97745f9

Compare security group rules

- Scroll down to the **Detailed CloudWatch monitoring** setting. Select Enable
- Choose **Create launch template**

Next, you will create an Auto Scaling group that uses this launch template.

Success

Successfully created LtConfig(lt-01ea93e68ab661755)

Actions log

Next Steps

Launch an instance

With On-Demand instances, you pay for compute capacity by the second (for Linux, with a minimum of 60 seconds) or by the hour (for all other operating systems) with no long-term commitments or upfront payments. Launch an On-Demand instance from your launch template.

Launch instance from this template

Create an Auto Scaling group from your template

Amazon EC2 Auto Scaling helps you maintain application availability and allows you to scale your Amazon EC2 capacity up or down automatically according to conditions you define. You can use Auto Scaling to help ensure that you are running your desired number of Amazon EC2 instances during demand spikes to maintain performance and decrease capacity during lulls to reduce costs.

Create Auto Scaling group

Create Spot Fleet

A Spot Instance is an unusual EC2 instance that is available for less than the On-Demand price. Because Spot Instances enable you to request unused EC2 instances at steep discounts, you can lower your Amazon EC2 costs significantly. The hourly price for a Spot instance (of each instance type in each Availability Zone) is set by Amazon EC2, and adjusted gradually based on the long-term supply of and demand for Spot Instances. Spot instances are well-suited for data analysis, batch jobs, background processing, and spooling tasks.

Create Spot Fleet

View launch template

24. In the Success dialog, choose the **LabConfig** launch template.

LabConfig

LabConfig (lt-01ea93e68ab661755)

Launch template details

Launch template ID lt-01ea93e68ab661755	Launch template name LabConfig	Default version 1
--	-----------------------------------	----------------------

Details **Versions** **Template tags**

Launch template version details

Version 1 (Default)	Description	Date created 2025-09-29T11:44:56.000Z	Created by arn:aws:sts::637423498408:assumed-role/v-E020@nirmauni.ac.in
Instance details		Actions	
AMI ID ami-0f933c196e94fd5ac	Instance type t3.micro	Availability Zone	Availability Zone Id
Key pair name rocky	Security groups	Security group IDs sg-0ac3ba75391be4cdf	

25. From the **Actions** menu, choose Create Auto Scaling group.
26. Configure the details in Step 1 (Choose launch template or configuration):
 - **Auto Scaling group name:** Lab Auto Scaling Group
 - **Launch template:** confirm that the LabConfig template you just created is selected.

Step 1 Choose launch template

Choose launch template

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.

Name

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

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

Launch template

For accounts created after May 11, 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, 2025.

Launch template

Create a launch template

- Choose **Next.**
- 27. Configure the details in Step 2 (Choose instance launch options):
 - **VPC:** choose Lab VPC
 - **Availability Zones and subnets:** Choose Private Subnet 1 and then choose Private Subnet 2.
 - Choose **Next.**

Network

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-0af5e719cf97745c9 (Lab VPC)

10.0.0.0/16

Create a VPC

Availability Zones and subnets

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

Select Availability Zones and subnets

use1-az4 (us-east-1a) | subnet-05f14a0df87deedb7 (Private Subnet 1)

use1-az6 (us-east-1b) | subnet-063b5038aa7a9d5c8 (Private Subnet 2)

Create a subnet

28. Configure the details in Step 3 (Configure advanced options):
 - Choose **Attach to an existing load balancer**
 - **Existing load balancer target groups:** select LabGroup.
 - In the **Additional settings** pane:
 - Select **Enable group metrics collection within CloudWatch**

This will capture metrics at 1-minute intervals, which allows Auto Scaling to react quickly to changing usage patterns.

- Choose **Next.**

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 forced 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

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

Select target groups ▾

LabGroup | HTTP
Application Load Balancer: LabWeb

29. Configure the details in Step 4 (Configure group size and scaling policies - optional):

- Under **Group size**, configure:
 - **Desired capacity:** 2
 - **Minimum capacity:** 2
 - **Maximum capacity:** 6

This will allow Auto Scaling to automatically add/remove instances, always keeping between 2 and 6 instances running.

Configure group size and scaling - optional Info

Define your group's desired capacity and scaling limits. You can optionally add automatic scaling to adjust the size of your group.

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 6 Equal or greater than desired capacity
---	--

- Under **Scaling policies**, choose Target tracking scaling policy and configure:
 - **Scaling policy name:** LabScalingPolicy
 - **Metric type:** Average CPU Utilization
 - **Target value:** 60

This tells Auto Scaling to maintain an average CPU utilization across all instances at 60%. Auto Scaling will automatically add or remove capacity as required to keep the

metric at, or close to, the specified target value. It adjusts to fluctuations in the metric due to a fluctuating load pattern.

- Choose **Next**.

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](#) Standard 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

Instance warmup [Info](#) 500 seconds

Disable scale-in to create only a scale-out policy

30. Configure the details in Step 5 (Add notifications - optional):

Auto Scaling can send a notification when a scaling event takes place. You will use the default settings.

- Choose **Next**.

Instances will attempt to launch into a Capacity Reservation first. If capacity isn't available, instances will run in On-Demand capacity.

Additional settings

Instance scale-in protection If protect from scale in is enabled, newly launched instances will be protected from scale in by default.

Enable instance scale-in protection

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

31. Configure the details in Step 6 (Add tags - optional):

Tags applied to the Auto Scaling group will be automatically propagated to the instances that are launched.

- Choose **Add Tag** and Configure the following:
 - **Key:** Name
 - **Value:** Lab Instance
- Choose **Next**.

Add tags - optional Info

Add tags to help you search, filter, and track your Auto Scaling group across AWS. You can also choose to automatically add these tags to instances when they are launched.

ⓘ You can optionally choose to add tags to instances (and their attached EBS volumes) by specifying tags in your launch template. We recommend caution, however, because the tag values for instances from your launch template will be overridden if there are any duplicate keys specified for the Auto Scaling group.

Tags (1)	
Key	Value - optional
Name	Lab Instance
<input checked="" type="checkbox"/> Tag new instances	
Add tag 45 remaining	
Remove	

32. Configure the details in Step 6 (Review):

- Review the details of your Auto Scaling group
- Choose **Create Auto Scaling Group**

Your Auto Scaling group will initially show an instance count of zero, but new instances will be launched to reach the **Desired** count of 2 instances.

Additional settings

Instance scale-in protection: Enabled

Monitoring: Enabled

Default instance warmup: Enabled

Capacity Reservation preference

Preference: Default

Capacity Reservation ID:

Resource groups:

Step 5: Add notifications

Notifications

No notifications

Step 6: Add tags

Tags (1)

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

[Preview code](#) [Next Step](#) [Previous Step](#) [Create Auto Scaling group](#)

➤ Task 4: Verify that Load Balancing is Working

In this task, you will verify that Load Balancing is working correctly.

33. In the left navigation pane, choose **Instances**.

You should see two new instances named **Lab Instance**. These were launched by Auto Scaling.

If the instances or names are not displayed, wait 30 seconds and choose refresh in the top-right.

Next, you will confirm that the new instances have passed their Health Check.

34. In the left navigation pane, choose **Target Groups**.

35. Select LabGroup.

36. Choose the **Targets** tab.

Two target instances named Lab Instance should be listed in the target group.

37. Wait until the **Status** of both instances transitions to healthy.

Choose Refresh in the upper-right to check for updates if necessary.

Healthy indicates that an instance has passed the Load Balancer's health check. This means that the Load Balancer will send traffic to the instance.

You can now access the Auto Scaling group via the Load Balancer.

Target group: LabGroup										
Registered targets (2) Info										
Target group 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 5 healthy targets.										
Anomaly mitigation: Not applicable Edit Deregister Register targets										
Filter targets	Instance ID	Name	Port	Zones	Health status	Health status details	Admin... Overrides	Overrid...	Last check	Launch...
Filter targets	603fe32d6645cd54	Lab Instance	80	us-east-1b (us...)	Healthy	-	<input type="checkbox"/> No override	No overrid...	September...	
	b0d1fb24c06ac9873b	Lab Instance	80	us-east-1a (us...)	Healthy	-	<input type="checkbox"/> No override	No overrid...	September...	

38. In the left navigation pane, choose **Load Balancers**.

39. Select the LabELB load balancer.

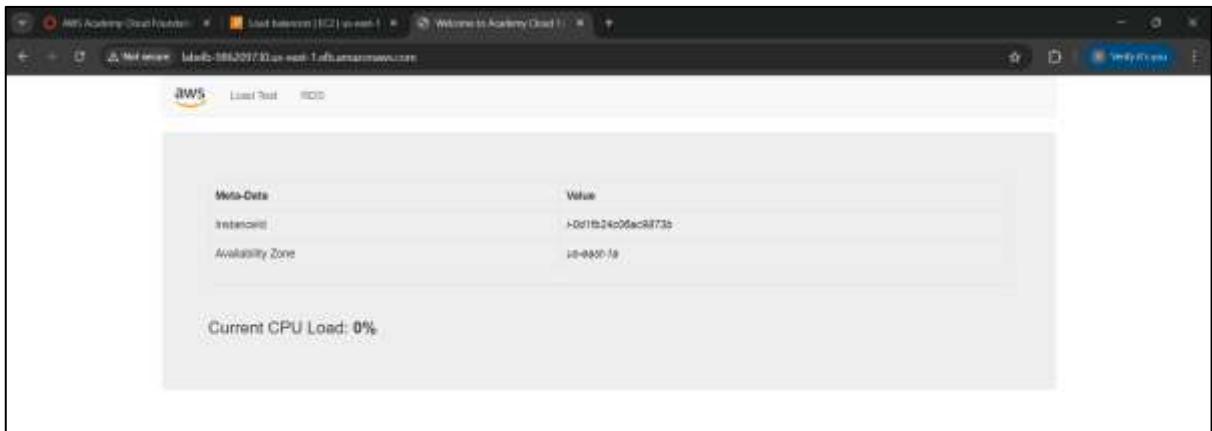
Load balancers (1/1)									
Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.									
Actions Create load balancer									
Filter load balancers									
Name	State	Type	Scheme	IP address type	VPC ID	Availability Zones	Security groups		
LabELB	Active	application	Internet-facing	IPv4	vpc-04f3a719c197745c9	2 Availability Zones	sg-0e3ba7539f1		

40. In the Details pane, copy the **DNS name** of the load balancer, making sure to omit "(A Record)".

It should look similar to: LabELB-1998580470.us-west-2.elb.amazonaws.com

41. Open a new web browser tab, paste the DNS Name you just copied, and press Enter.

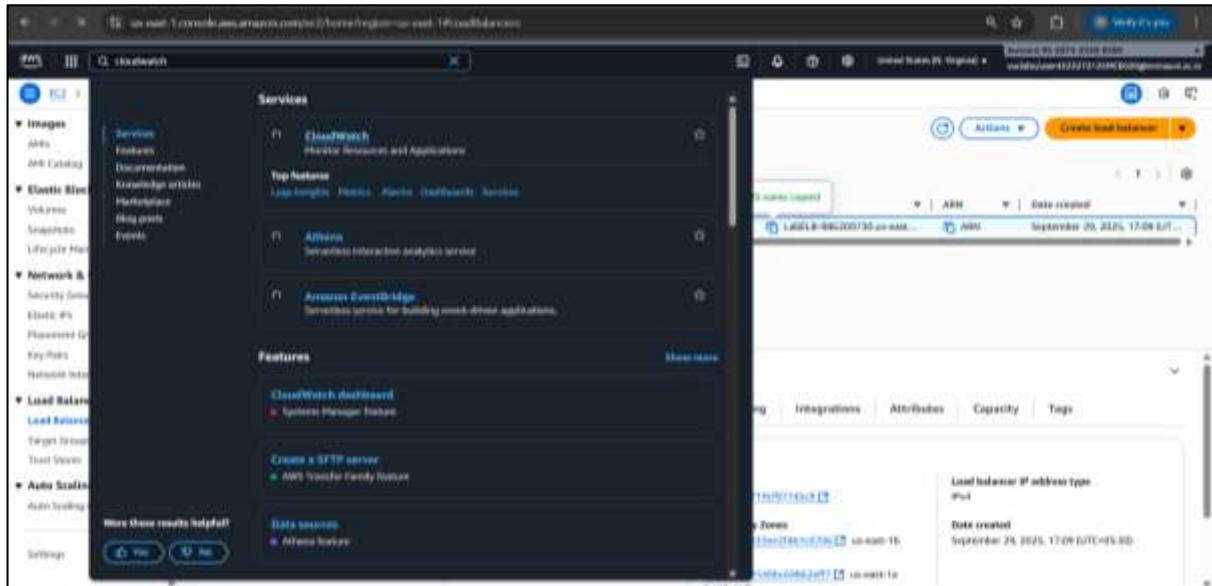
The application should appear in your browser. This indicates that the Load Balancer received the request, sent it to one of the EC2 instances, then passed back the result.



➤ Task 5: Test Auto Scaling

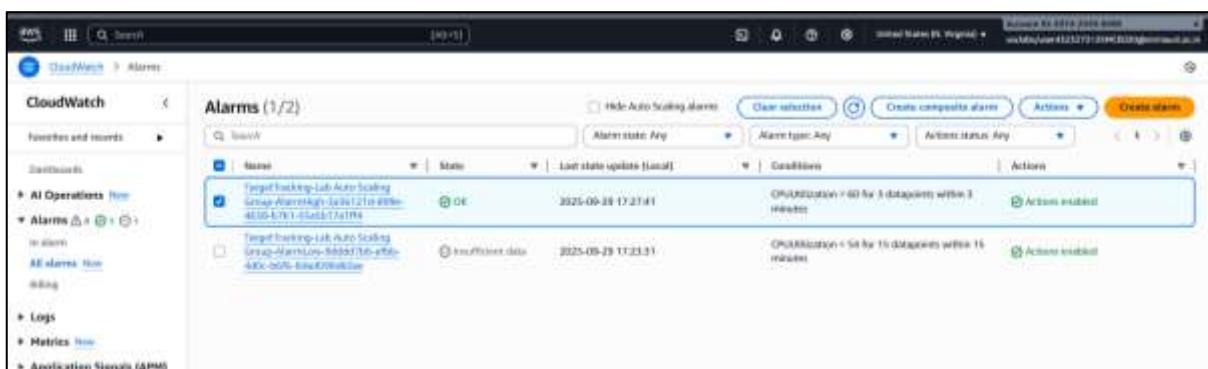
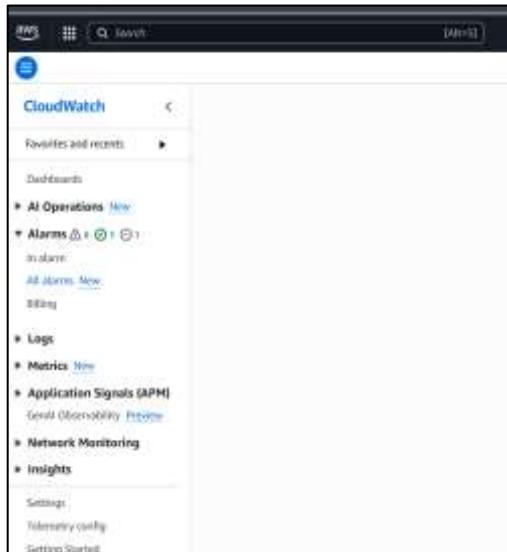
You created an Auto Scaling group with a minimum of two instances and a maximum of six instances. Currently two instances are running because the minimum size is two and the group is currently not under any load. You will now increase the load to cause Auto Scaling to add additional instances.

42. Return to the AWS Management Console, but do not close the application tab — you will return to it soon.
43. In the search box next to Services, search for and select **CloudWatch**.



44. In the left navigation pane, choose **All alarms**.

Two alarms will be displayed. These were created automatically by the Auto Scaling group. They will automatically keep the average CPU load close to 60% while also staying within the limitation of having two to six instances.



- On the **Services** menu, choose **EC2**.
- In the left navigation pane, choose **Auto Scaling Groups**.
- Select **Lab Auto Scaling Group**.
- In the bottom half of the page, choose the **Automatic Scaling** tab.
- Select **LabScalingPolicy**.
- Choose **Action** and **Edit**.
- Change the **Target Value** to **50**.
- Choose **Update**.
- On the **Services** menu, choose **CloudWatch**.
- In the left navigation pane, choose **All alarms** and verify you see two alarms.

45. Choose the **OK** alarm, which has AlarmHigh in its name.

Name	Status	Last state update (Local)	Conditions	Actions
TargetTracking-Lab Auto Scaling	OK	2025-09-29 17:41:31	CPUUtilization = 54 for 15 datapoints within 15 minutes	Actions enabled
TargetTracking-Lab Auto Scaling Group-AlarmHigh-1a8121d-099e-4638-6fb3-55a5b17a1f84	OK	2025-09-29 17:27:41	CPUUtilization = 60 for 3 datapoints within 3 minutes	Actions enabled

If no alarm is showing **OK**, wait a minute then choose refresh in the top-right until the alarm status changes.

The **OK** indicates that the alarm has not been triggered. It is the alarm for **CPU Utilization > 60**, which will add instances when average CPU is high. The chart should show very low levels of CPU at the moment.

You will now tell the application to perform calculations that should raise the CPU level.

46. Return to the browser tab with the web application.

47. Choose **Load Test** beside the AWS logo.

This will cause the application to generate high loads. The browser page will automatically refresh so that all instances in the Auto Scaling group will generate load. Do not close this tab.

48. Return to browser tab with the **CloudWatch** console.

In less than 5 minutes, the **AlarmLow** alarm should change to **OK** and the **AlarmHigh** alarm status should change to In alarm.

You can choose Refresh in the top-right every 60 seconds to update the display.

You should see the **AlarmHigh** chart indicating an increasing CPU percentage. Once it crosses the 60% line for more than 3 minutes, it will trigger Auto Scaling to add additional instances.

49. Wait until the **AlarmHigh** alarm enters the In alarm state.

You can now view the additional instance(s) that were launched.

The screenshot shows the CloudWatch Metrics Alarms page with two entries:

- TargetTracking-Lab Auto Scaling Group-AlarmLow**: State: OK, Last state update: 2025-09-29 17:41:31, Condition: CPUUtilization <= 42 for 15 datapoints within 15 minutes, Actions enabled.
- TargetTracking-Lab Auto Scaling Group-AlarmHigh**: State: In alarm, Last state update: 2025-09-29 17:42:41, Condition: CPUUtilization >= 60 for 3 datapoints within 3 minutes, Actions enabled.

50. In the search box next to **Services**, search for and select **EC2**.

51. In the left navigation pane, choose **Instances**.

More than two instances labeled **Lab Instance** should now be running. The new instance(s) were created by Auto Scaling in response to the CloudWatch alarm.

The screenshot shows the AWS EC2 Instances page with the following instance details:

Name	Instance ID	Instance state	Instance-type	Status check	Alarm status	Availability
Bastion Host	i-0f9d4caed750ad57a	Running	t2.micro	2/2 checks passed	View alarms	us-east-1a
Lab Instance	i-048fa319a2f940c39	Running	t3.micro	Initializing	View alarms	us-east-1b
Lab Instance	i-05ef12af6fcbed54	Running	t3.micro	2/3 checks passed	View alarms	us-east-1b
Lab Instance	i-0d1fb24cd6ac9875b	Running	t3.micro	3/3 checks passed	View alarms	us-east-1a

➤ Task 6: Terminate Web Server 1

In this task, you will terminate Web Server 1. This instance was used to create the AMI used by your Auto Scaling group, but it is no longer needed.

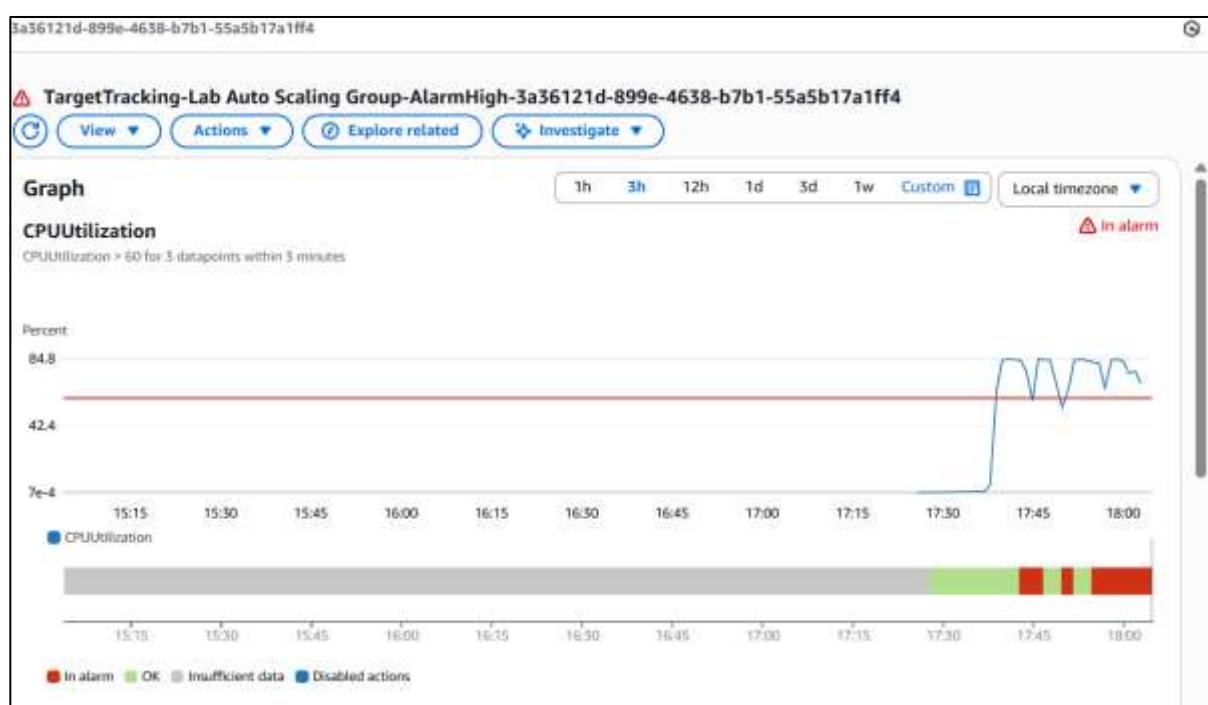
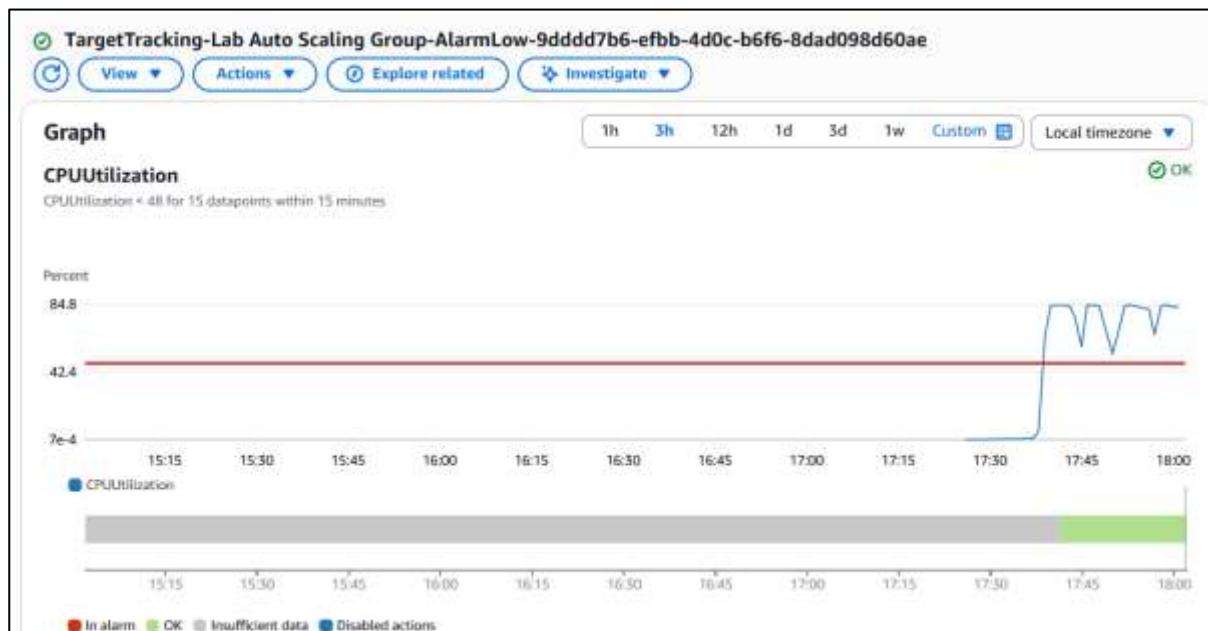
52. Select **Web Server 1** (and ensure it is the only instance selected).

53. In the **Instance State** menu, choose **Instance State > Terminate Instance**.

54. Choose **Terminate**.

The screenshot shows the AWS EC2 Instances page with the following interface elements:

- EC2 Instances** sidebar: Includes Dashboard, EC2 Global View, and Instances (selected).
- Instances (1/5)** table:
 - Header: Name, Instance ID, Instance state, Instance-type, Status check, Alarm status, Availability.
 - Data row: Web Server 1 (selected), i-03f1f50ba10011264, Running, t2.micro, 2/2 checks passed, [View alarms](#), us-east-1a.
 - Other rows: Bastion Host, Lab Instance, Lab Instance, Lab Instance.
- Action Bar**: Contains Connect, Instance state (dropdown set to Stop instance), Actions (dropdown set to Stop instance), and Launch instances.
- Stop instance** dropdown:
 - Start instance
 - Stop instance
 - Reboot instance
 - Termination protection
 - Terminate selected instance (selected)
- Relevant instance** dropdown: Shows the selected instance: Web Server 1 (i-03f1f50ba10011264).
- Actions** dropdown:
 - View alarms + us-east-1a
 - View alarms + us-east-1b
 - 3/3 checks passed
 - View alarms + us-east-1b
 - 3/3 checks passed
 - View alarms + us-east-1a



❖ Conclusion: