

Home / AWS / Guided Lab / Creating an Application Load Balancer and Auto Scaling Group in AWS

Creating an Application Load Balancer and Auto Scaling Group in AWS

Level: **Intermediate**

Amazon EC2 Amazon Web Services Elastic Load Balancing Amazon EC2 Auto Scaling



1h 29m 12s left



End Lab

Open Console

Validation

Lab Credentials

User Name ⓘ

Whiz_User_80425.64332336



Password ⓘ

fb5d3310-99fd-4b39-b872-2a84056c8bd1



Access Key ⓘ

AKIA4BHUHZ5XGVPAADOZ



Secret Key ⓘ

Abdkw38ZudU4U045iwupegjEjGuBOK/y+4/pyoLp






Lab Resources

No Lab Resources Found

Support Documents

No Support Documents Found

Need help?

-  How to use Hands on Lab
-  Troubleshooting Lab
-  FAQs

[Submit Feedback](#)[Share](#)[Lab Overview](#)[Lab Steps](#)[Lab Validation](#) Cloud Architect Compute

Lab Steps

Task 1: Sign in to AWS Management Console

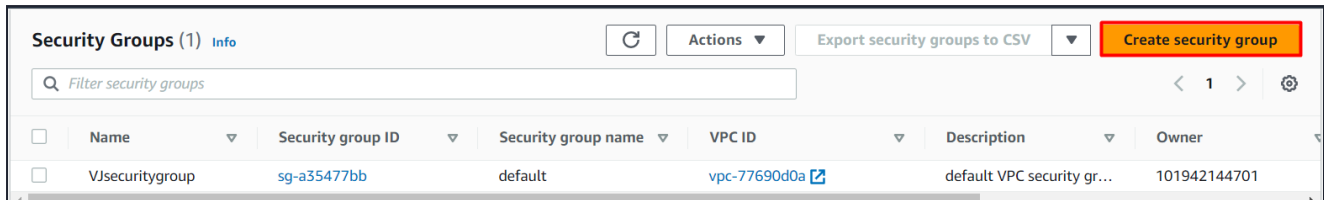
1. Click on the **Open Console** button, and you will get redirected to AWS Console in a new browser tab.
2. On the AWS sign-in page,
 - Leave the Account ID as default. Never edit/remove the 12 digit Account ID present in the AWS Console. otherwise, you cannot proceed with the lab.
 - Now copy your **User Name** and **Password** in the Lab Console to the **IAM Username**



Task 2: Create a Security Group for the Load balancer

1. Make sure you are in the **N.Virginia** Region.
2. Navigate to **EC2** by clicking on the **Services** menu available under the **Compute** section.
3. On the left panel menu, select the **Security Groups** under the **Network & Security** section.

4. Click on the **Create security group** button.



5. We are going to create a Security group for the Launch template with port 80 number enabled.

- Security group name: Enter **Load-balancer-SG**
- Description: Enter **Security group for Load balancer**
- VPC: Select **Default VPC**

The screenshot shows the 'Basic details' section of the AWS Security Groups 'Create' wizard. Three fields are highlighted with red boxes: 'Security group name' with the value 'Load-balancer-SG', 'Description' with the value 'Security group for Load balancer', and 'VPC' with the value 'vpc-6d289d17 (Default VPC)'.

- Click on the **Add rule** button under **Inbound rules**.
- Type : Select **HTTP**
- Source : Select **Custom**
- In the textbox add **0.0.0.0/0**

The screenshot shows the 'Inbound rules' section of the AWS Security Groups 'Create' wizard. A new rule is being added. The 'Type' dropdown is set to 'HTTP', the 'Protocol' is 'TCP', and the 'Port range' is '80'. The 'Source' dropdown is set to 'Custom', and the 'Source' text box contains '0.0.0.0/0'. The 'Add rule' button is visible at the bottom left.

6. Leave everything as default and click on the **Create security group** button.

Task 3: Create a Security Group for Launch template

1. Click on the **Create security group** button.

2. We are going to create a Security group for the Launch template with port 80 number enabled.

- Security group name: Enter **Launch-template-SG**
- Description: Enter **Security group for Launch template**
- VPC: Select **Default VPC**



Basic details

Security group name [Info](#)
Launch-template-SG
Name cannot be edited after creation.

Description [Info](#)
Security group for Launch template

VPC [Info](#)
vpc-6d289d17 (Default VPC) ▼

- Click on the **Add rule** button under **Inbound rules**.
 - Here we will add **SSH from the open internet** and **HTTP from the security group of a Load Balancer**.
 - Type : Select **SSH**
 - Source : Select **Custom**
 - In the textbox add **0.0.0.0/0**
 - Click on the **Add rule** button to add **HTTP**
 - Type : Select **HTTP**
 - Source : Select **Custom**
 - In the textbox type, **Load-balancer-SG**, select it.

Inbound rules [Info](#)

Type Info	Protocol Info	Port range Info	Source Info
SSH	TCP	22	Anywh... 0.0.0.0/0
HTTP	TCP	80	Custom sg-0758812a96d27c1fe

[Add rule](#)

6. Leave everything as default and click on the **Create security group** button.

Task 4: Create a Key Pair for the Launch template

1. In the left navigation pane (scroll down) within **Network & Security**, click on the **Key Pairs**.
2. To create a new key pair, click on the **Create key pair** button.

Key pairs [Info](#)

[Search](#)

[Refresh](#) [Actions](#) [Create key pair](#)

Name	Type	Created	Fingerprint	ID
No key pairs to display				

3. Fill in the details below:

- Name: Enter **WhizKeyPair**
- File format: **pem (Linux & Mac Users)** or **ppk (Windows users)**
- Leave other options as default.
- Click on the **Create key pair** button.

Key pair

A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance.

Name

WhizKeyPair

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

Info

☒ RSA

☐ ED25519

Private key file format

☒ .pem

For use with OpenSSH

☐ .ppk

For use with PuTTY

Tags - optional

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel

Create key pair

4. Key pair will be created.

Task 5: Creating a Launch template

1. In the left navigation pane (scroll down) within **Instances**, click on the **Launch templates**
2. Click on the **Create launch template** button.

Compute

EC2 launch templates

Streamline, simplify and standardize instance launches

Use launch templates to automate instance launches, simplify permission policies, and enforce best practices across your organization. Save launch parameters in a template that can be used for on-demand launches and with managed services, including EC2 Auto Scaling and EC2 Fleet. Easily update your launch parameters by creating a new launch template version.

Benefits and features

New launch template

Create launch template

3. Under **Launch template name and description** section:

- Launch template name: Enter **whizlabsLC**
- Template version description: Enter **Launch template for whizdemo**

- Leave other options as default.

Launch template name and description

Launch template name - *required*

Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'.

Template version description

Max 255 chars

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

► **Template tags**

► **Source template**

4. Under Launch template contents:

- Select **Amazon Linux** from the Quick Start
- Amazon machine image (AMI): Select **Amazon Linux 2 AMI (HVM), SSD Volume Type**

Quick Start

Don't include in launch template

Amazon Linux
aws

macOS
Mac

Ubuntu
ubuntu

Windows
Microsoft

>

Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type
ami-06a0cd9728546d178 (64-bit (x86)) / ami-09e51988f56677f44 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible ▼

Description

Amazon Linux 2 Kernel 5.10 AMI 2.0.20230504.1 x86_64 HVM gp2

Architecture

AMI ID

64-bit (x86) ▼

ami-06a0cd9728546d178

Verified provider

- Under Instance type:
 - Select **t2.micro** from the below list.

Instance type [Info](#)

Instance type

Don't include in launch template ▲

Q |

Don't include in launch template

t1.micro **Free tier eligible**

Family: t1 1 vCPU 0 GiB Memory On-Demand Linux pricing: 0.02 USD per Hour
On-Demand Windows pricing: 0.02 USD per Hour

t2.nano

Family: t2 1 vCPU 0 GiB Memory On-Demand Linux pricing: 0.0058 USD per Hour
On-Demand Windows pricing: 0.0081 USD per Hour

t2.micro **Free tier eligible**

Family: t2 1 vCPU 1 GiB Memory On-Demand Linux pricing: 0.0116 USD per Hour
On-Demand Windows pricing: 0.0162 USD per Hour

t2.micro

- Key pair (Login): Select **WhizKeyPair**

Key pair (login) [Info](#)

Key pair name

WhizKeyPair ▼

- Subnet: Choose **any subnet from the list**.
- Security groups: Select **Launch-template-SG** from the list

▼ **Network settings** [Get guidance](#)

Subnet Info

subnet-a2564def
VPC: vpc-f5563288 Owner: 531961822138 Availability Zone: us-east-1a
IP addresses available: 4091

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

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

Common security groups Info

Select security groups

Launch-template-SG sg-0ec1eca7412870976 X
VPC: vpc-f5563288

Security groups that you add or remove here will be added to or removed from all your network interfaces.

[Create new subnet](#) [Compare security group rules](#)

- Leave all other options as default.
- Expand the option of **Advanced details**, Go to the **User data**, and paste the below script.

```
#!/bin/bash
sudo su
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
echo "Hello World from $(hostname -f)" > /var/www/html/index.html
echo "Healthy" > /var/www/html/health.html
```



9. To create, click on the **Create launch template** button. Upon successful creation, it will create a Launch template.

Task 6: Create Target group and The Load Balancer

(i) Create Target group :

1. In the EC2 console, navigate to **Target Groups** from the left navigation panel.
2. Click on **Create Target Group** button.

Target groups [Info](#)

[Create target group](#)

Search or filter target groups

Name	ARN	Port	Protocol	Target type	Load balancer
No target groups					
You don't have any target groups in us-east-1					

[Create target group](#)

- **Target Type:** Select **Instances**

- Name: Enter **web-server-TG**
- Protocol: Choose **HTTP**
- Port : Enter **80**

Target group name

web-server-TG

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol

Port

HTTP

:

80

VPC

Select the VPC with the instances that you want to include in the target group.

Default VPC
vpc-77690d0a
IPv4: 172.31.0.0/16

Protocol version

☒ HTTP1
Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.

☐ HTTP2
Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.

☐ gRPC
Send requests to targets using gRPC. Supported when the request protocol is gRPC.

- Note: The target group is used to route requests to one or more registered targets

3. Health check:

- Protocol: Select **HTTP**
- Path: Enter **/health.html**
- Note: The load balancer periodically sends pings, attempts connections, or sends requests to test the EC2 instances. These tests are called health checks.

Health checks

The associated load balancer periodically sends requests, per the settings below, to the registered targets to test their status.

Health check protocol

HTTP

Health check path

Use the default path of "/" to ping the root, or specify a custom path if preferred.

/health.html

Up to 1024 characters allowed.

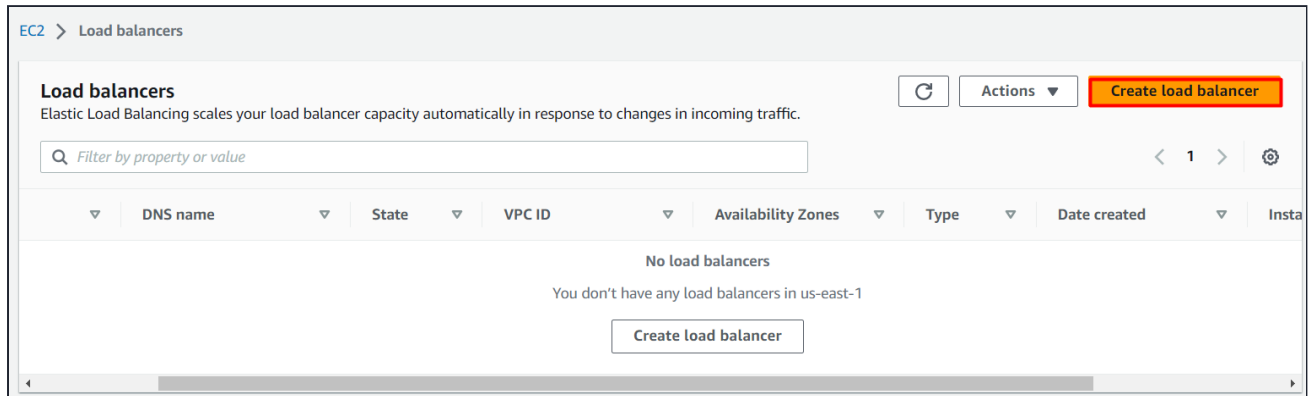
► Advanced health check settings

4. Click on the **Next** button.

5. Leave this page as default and click on **Create target group** button.

(ii) Create Load Balancer :

1. In the EC2 console, navigate to **Load balancers** in the left-side panel.
2. Click on **Create load balancer** button at the top-left to create a new load balancer for our web servers.



3. On the next screen, choose **Create** button under **Application Load Balancer** since we are testing the high availability of the web application.

4. In **Basic configuration**:

- Name: Enter **Web-server-LB**
- Scheme: Select **Internet-facing**
- IP address type: Choose **IPv4**

Basic configuration

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

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme [Info](#)
Scheme can't be changed after the load balancer is created.

☒ **Internet-facing**
An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)

☐ **Internal**
An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type [Info](#)
Select the type of IP addresses that your subnets use.

☒ **IPv4**
Recommended for internal load balancers.

☐ **Dualstack**
Includes IPv4 and IPv6 addresses.

- **Availability Zones**

- VPC: Choose **Default**
- Availability Zones : Select **us-east-1a** and **us-east-1b**.
- **Security Groups:**
 - Remove the default one and choose **Load-balancer-SG**

Security groups [Info](#)

A security group is a set of firewall rules that control the traffic to your load balancer.

Security groups

Select up to 5 security groups ▼

[Create new security group](#)

default sg-a35477bb X
VPC: vpc-77690d0a

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 th to its registered targets.

- **Note:** We must specify the availability zones in which the load balancer needs to be enabled, making it route traffic only two targets launched in those availability zones. You must include **subnets from a minimum of two Availability zones** to make our Load balancer **Highly-Available**.

5. In the listener part select the Target group that you have created earlier. if it is not visible click on refresh button.

▼ Listener HTTP:80 [Remove](#)

Protocol HTTP ▼

Port 80
1-65535

Default action [Info](#)

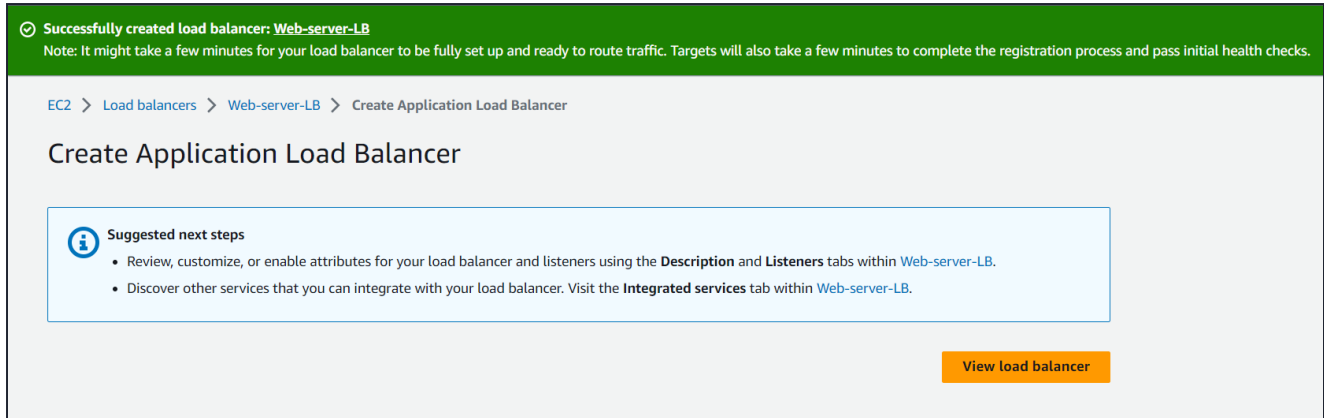
Forward to web-server-TG
Target type: Instance, IPv4

HTTP ▼

[Create target group](#)

6. Click on **Create Load Balancer** button.

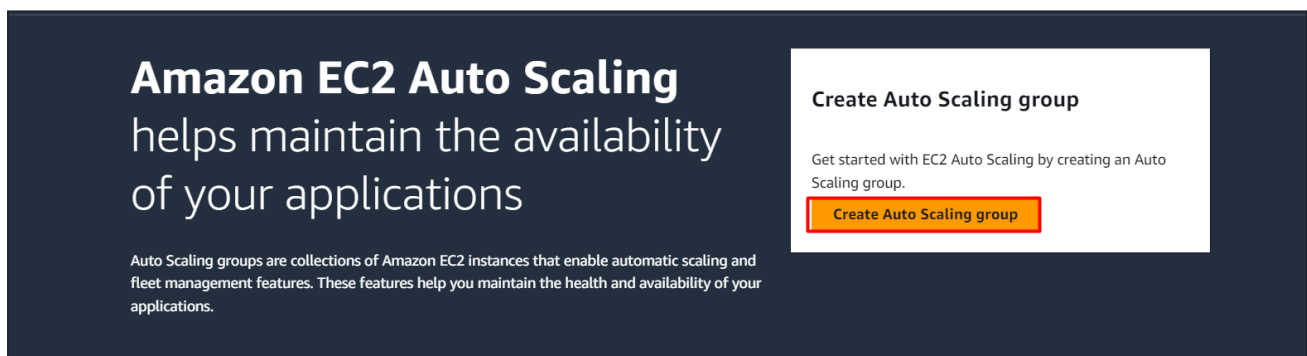
7. **You have successfully created the Application Load balancer.**



8. Wait for **2 to 3 minutes** for the load balancer to become **Active**.

Task 7: Create an Auto Scaling Group

1. An Auto Scaling group is a scalable collection of EC2 instances. When you create an Auto Scaling group, you include information such as the subnets for the instances and the number of instances the group must maintain at all times.
2. Go to the left menu under EC2 and choose **Auto Scaling Groups** under **Auto Scaling**.
3. Click on the **Create Auto Scaling group** button.



4. Step 1: Choose launch template or configuration

- Auto Scaling group name : Enter **Whiz-ASG**
- Select the Launch template **whizlabsLC** from the list and click on the **Next** button.

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.

Whiz-ASG

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

Launch template [Info](#)

[Switch to launch configuration](#)

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.

Select a launch template

Search launch templates

whizlabsLC

Cancel

Next

5. Step 2: Configure settings

- VPC: Select the **Default VPC** from the list.
- Subnet: Select **Subnet of us-east-1a and us-east-1b**
- Click on the **Next** button.

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-77690d0a (Default VPC)

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-f9c9819f X
172.31.0.0/20 Default

us-east-1b | subnet-bd5c0a9c (SubNet_wizlab) X
172.31.80.0/20 Default

[Create a subnet](#)

6. Step 3: Configure advanced options

- Load balancing – *optional*: **Attach to an existing load balancer**

- Attach to an existing load balancer: **Choose from your load balancer target groups**
- Existing load balancer target groups: **web-server-TG**

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 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.

web-server-TG | HTTP

Application Load Balancer: Web-server-LB

✕

- Health check – *optional*:
 - Health check type: EC2 (default) and **Check the** Turn on Elastic Load Balancing health checks **checkbox**.
 - Health check grace period: **60** seconds
- Click on the **Next** button.

6. Step 4: Configure group size and scaling policies

- Under Group size – *optional*
 - Desired capacity : Enter **1**
 - Minimum capacity : Enter **1**
 - Maximum capacity : Enter **4**
- Under Scaling policies – *optional*
 - Select **Target tracking scaling policy**
 - Scaling policy name: Target tracking policy

- Metric value: **Average CPU Utilization**
- Target value: Enter **30**
- Instance need: **60** seconds warm up before including in metric
- Under Instance scale-in protection
 - No changes are needed, click on the **Next** button.

7. Step 5: Add notifications

- No changes are needed in this page, click on the **Next** button.

8. Step 6: Add tags

- Enter tags in key-value pairs to identify your auto scaling group **instances**.
 - Key: Enter **Name**
 - Value: Enter **Whiz**

Tags (1)

Key	Value - optional
Name	Whiz

Tag new instances ☒

Add tag 49 remaining **Remove**

- Click on the **Next** button.

6. Now Review, scroll down and click on the **Create Auto Scaling Group** button.

7. You will be redirected to the autoscaling group page, you will be able to see that one instance is launched by the autoscaling group.

8. Now go to the EC2 instances list. You will see that there are **one running instances** (which were created by your autoscaling group) You can confirm this from their tag name, which you gave at the time of creating the autoscaling group.

9. You have successfully created an autoscaling group with a policy to a minimum of 1 and a maximum of 4 instances.

Task 8: SSH into EC2 Instance

- Please follow the steps in [SSH into EC2 Instance](#).

Task 9: Install the stress

1. Switch to root user:

```
sudo su
```



2. Now run the updates using the following command:

```
yum -y update
```



3. Once completed, let's install and run an stress

- Install the required packages and libraries

```
amazon-linux-extras install epel -y
```



```
yum install stress -y
```



```
stress --cpu 8 --timeout 300s
```



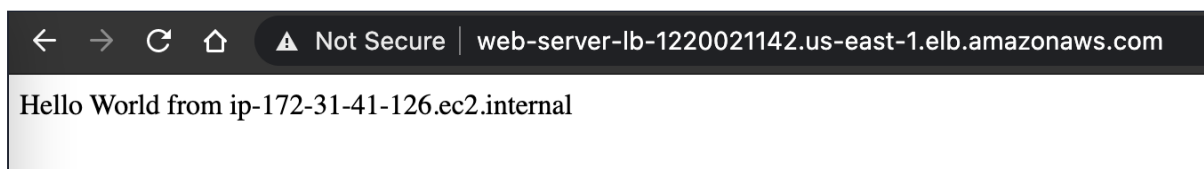
Task 10: Test Auto Scaling Group and Elastic Load Balancer

1. For testing the auto-scaling policy, go to the EC2 instance list. You will see more new instances are getting launched.
2. That we can see in Auto scaling page also. Capacity is now incremented by 3.

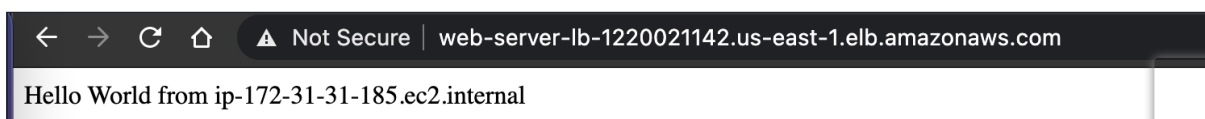
Note: It will take 10-15 min to launch all instances

Auto Scaling groups (1)							Create an Auto Scaling group	
Search your Auto Scaling groups							< 1 > ⚙	
<input type="checkbox"/>	Name ▾	Launch template/configuration ▾	Instances ▾	Status ▾	Desired capacity ▾	Min ▾		
<input type="checkbox"/>	Whiz-ASG	whizlabsLC Version Default	4	-	4	1		

3. Copy the DNS of your load balancer, and paste it into the browser.



4. The load balancer will now try to route to a new instance every time.

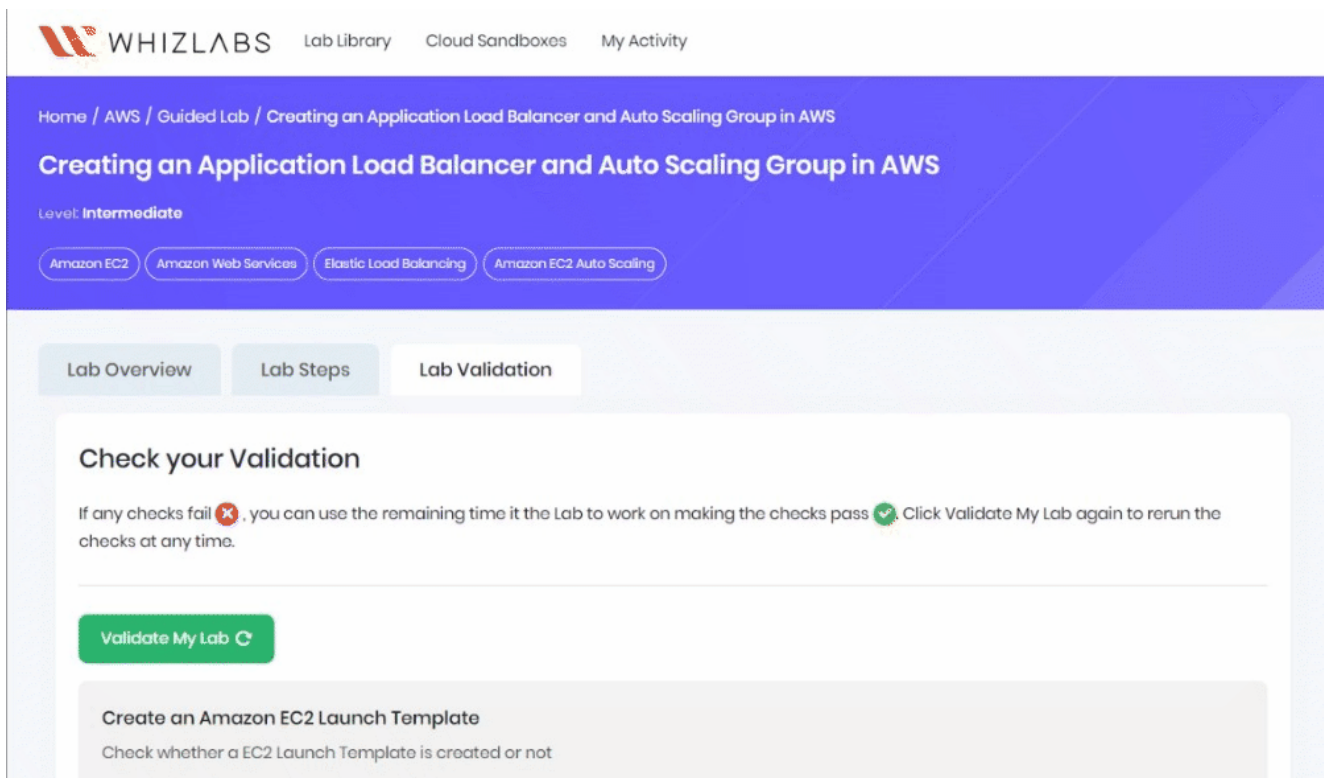


Do you know ?

Autoscaling groups support rolling updates, which allow you to update your application without downtime. When updating the launch configuration or the Amazon Machine Image (AMI) used by the autoscaling group, it can gradually replace instances one-by-one, ensuring that your application remains available throughout the update process.

Task 11: Validation of the Lab

1. Once the lab steps are completed, please click on the **Validation** button on the right-side panel.
2. This will validate the resources in the AWS account and displays whether you have completed this lab successfully or not.
3. Sample output :



Task 12: Delete AWS Resources

Deleting Auto scaling group:

- In the EC2 console, navigate to **Auto scaling groups** in the left-side panel.
- **Whiz-ASG** will be listed here.
- To **delete** the Auto scaling group, need to perform the following actions:

- **Select** the auto scaling group, **Whiz-ASG**
- Click on the **Actions** dropdown and then click on **Delete**,
- Confirm by entering **delete** and click on the **Delete** button when a pop-up is shown.
- **Whiz-ASG's** status will be shown as Deleting immediately.
- It can take up to 3 minutes to delete because it will terminate the EC2 instances.
- You can confirm the termination of EC2 instances by visiting the **Instances** page.

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
<input type="checkbox"/>	Whiz	i-0025fcd0439206fe	Terminated	t2.micro	-	No alarms
<input type="checkbox"/>	Whiz	i-0b2d0b72af4cc0e5d	Terminated	t2.micro	-	No alarms
<input type="checkbox"/>	Whiz	i-07341713c94ad906c	Terminated	t2.micro	-	No alarms

Delete Launch templates

- In the EC2 console, navigate to **Launch templates** in the left-side panel.
- **WhizlabsLC** will be listed here.

Launch template ID	Launch template name	Default version	Latest version
lt-0794b584c2d4ef7bc	whizlabsLC	1	1

- To delete the **Launch template**, need to perform the following actions:
 - **Select** the Launch template,
 - Click on the **Actions** button,
 - choose the **Delete template** option

Launch template ID	Launch template name	Default version	Latest version
lt-0794b584c2d4ef7bc	whizlabsLC	1	1

- Confirm by **entering Delete** and clicking on the **Delete** button when a pop-up is shown.

Delete lt-0794b584c2d4ef7bc

×

You cannot undo this action. Any Auto Scaling groups or Spot fleet requests currently using this launch template may be affected.

Are you sure you want to delete (lt-0794b584c2d4ef7bc) and all its versions?

To confirm deletion, type *Delete* in the field

Delete

▶ CLI commands

Cancel

Delete

- **Whiz-ASG** will be deleted immediately.

Delete Load balancer

- In the EC2 console, navigate to **Load Balancers** in the left-side panel.
- **Web-server-LB** will be listed here.
- To **delete** the load balancer, need to perform the following actions:
 - **Select** the load balancer,
 - Click on the **Actions** button,
 - Select the **Delete** option.
- Confirm by clicking on the **Yes, Delete** button when a pop-up is shown.
- MyNetwork-LB will be deleted immediately.

Delete Target group

- In the EC2 console, navigate to **Target Groups** in the left-side panel.
- **web-server-TG** will be listed here.

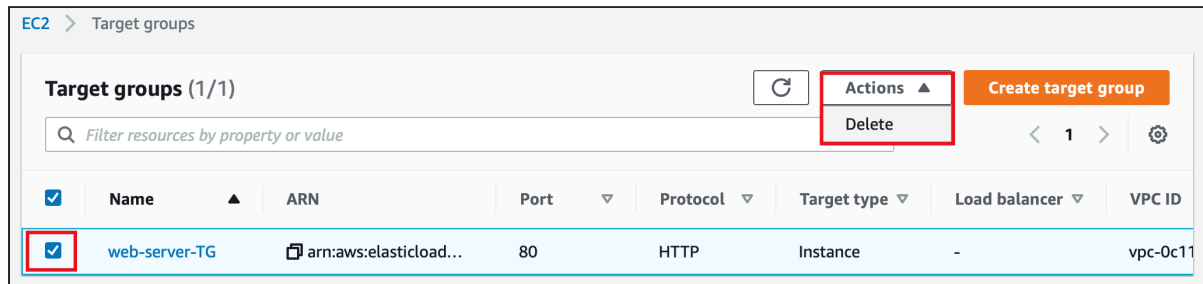
EC2 > Target groups

Target groups (1)							
<div><div>Filter resources by property or value</div><div><div>Actions</div><div>Create target group</div></div></div>							
	Name	ARN	Port	Protocol	Target type	Load balancer	VPC ID
<input type="checkbox"/>	web-server-TG	arn:aws:elasticload...	80	HTTP	Instance	-	vpc-0c11

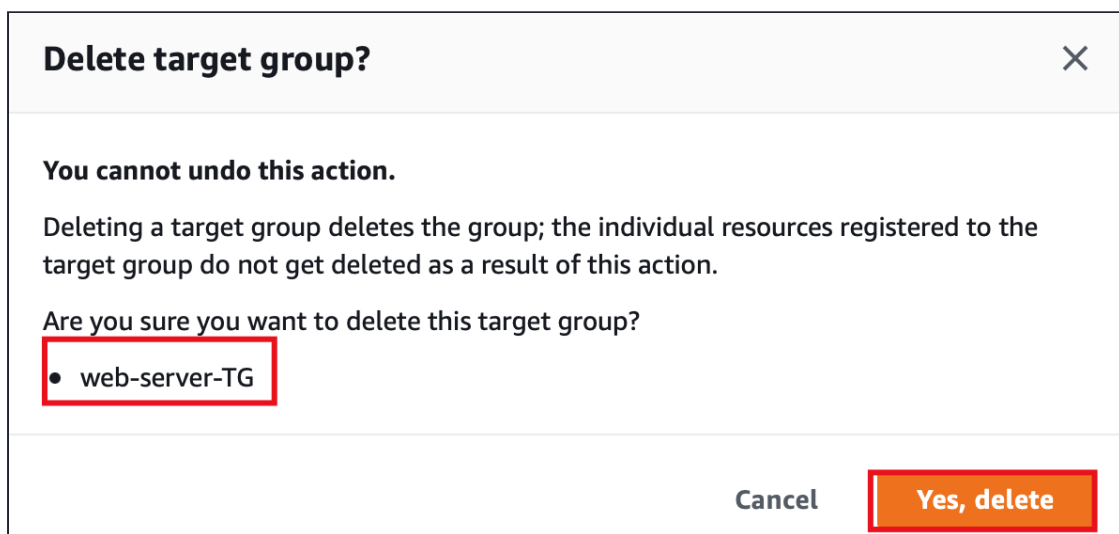
<https://www.whizlabs.com/labs/creating-an-application-load-balancer-and-auto-scaling-group-in-aws>

20/22

- To **delete** the target group, need to perform the following actions:
 - **Select** the load balancer,
 - Click on the **Actions** button,
 - Select the **Delete** option.



- Confirm by clicking on the **Yes, Delete** button when a pop-up is shown.



- Target group will be deleted immediately.

Completion and Conclusion

1. You have created a security group and key pair for the Launch template.
2. You have created a **Launch template** and **Auto Scaling**.
3. You have created an Application load balancer with Target group.
4. You have tested the Autoscaling by stressing the load on the first EC2 instance.

End Lab

1. Sign out of the AWS Account.
2. You have successfully completed the lab.
3. Once you have completed the steps, click on **End Lab** from your whizlabs dashboard.

[About Us](#) [Subscription](#) [Instructions and Guidelines](#) [FAQ's](#) [Contact Us](#)



© 2024, Whizlabs Software Pvt. Ltd.

