

Cycle 08 AWS Homework - ASG + ALB Replication

Part A: Setup and Implementation

1. EC2 Instance Setup

- Launch two Ubuntu EC2 instances in the same region.
- Install Apache web server:

```
bash
CopyEdit
sudo apt update
sudo apt install apache2 -y
```

- Add custom content:
 -
 - On Server 1: update `/var/www/html/index.html`
 - On Server 2: update `/var/www/html/server2.html`
- Test in browser using public IPs.

2. Create AMIs

- Stop both instances.
- Create AMIs for each server from the EC2 console.
- Name them distinctly (e.g., `Server1-AMI` , `Server2-AMI`).

3. Launch Templates

- Create two launch templates using each AMI.
- Select instance type (t2.micro for free tier).
- Ensure networking and security groups allow HTTP access.

4. Auto Scaling Groups (ASG)

- Create two ASGs, one for each launch template.
- Set minimum capacity = 1, maximum = 3.
- Use multiple Availability Zones for redundancy.
- Attach each ASG to its respective target group.

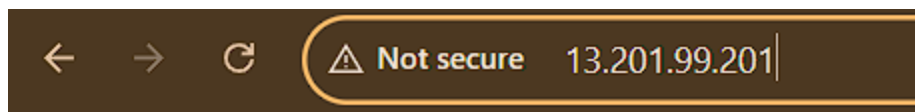
5. Application Load Balancer (ALB)

- Create an ALB (internet-facing).
- Add two target groups: one for each ASG.
- Configure listener rules:
 - Path `/` → Target Group 1
 - Path `/server2.html` → Target Group 2
- Access via ALB DNS to verify routing.

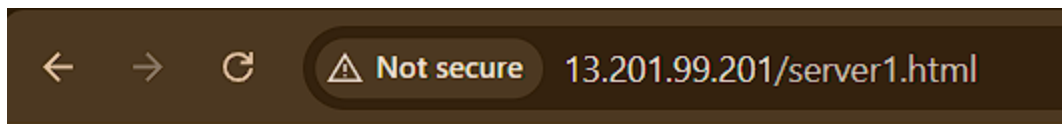
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
<input type="checkbox"/>	instance2	i-0fc9357d47a8b3a60	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1b	ec2-13-20
<input type="checkbox"/>	instance1	i-0db3dc0a122cf6d30	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1b	ec2-13-20

```
root@ip-172-31-2-45:~# cat > /var/www/html/index.html
Welcome to home2
root@ip-172-31-2-45:~# cat > /var/www/html/server2.html
<h1>Welcome to server2</h1>
root@ip-172-31-2-45:~# history
 1  sudo apt-get update
 2  sudo apt-get install apache2
 3  sudo systemctl enable apache2
 4  sudo systemctl restart apache2
 5  sudo systemctl enable apache2
 6  sudo systemctl status apache2
 7  cat > /var/www/html/index.html
 8  cat > /var/www/html/server2.html
 9  history
root@ip-172-31-2-45:~#
```

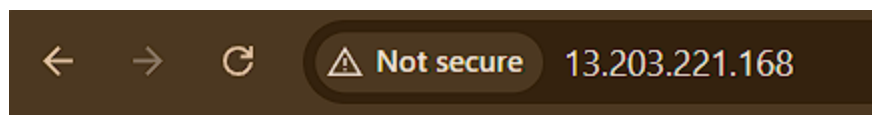
```
root@ip-172-31-11-46:~# cat > /var/www/html/index.html
Welcome to home1
root@ip-172-31-11-46:~# cat > /var/www/html/server1.html
<h1>Welcome to server1</h1>
root@ip-172-31-11-46:~# history
 1 sudo apt-get update
 2 sudo apt-get install apache2
 3 sudo systemctl enable apache2
 4 sudo systemctl restart apache2
 5 sudo systemctl status apache2
 6 cat > /var/www/html/index.html
 7 cat > /var/www/html/server1.html
 8 history
```



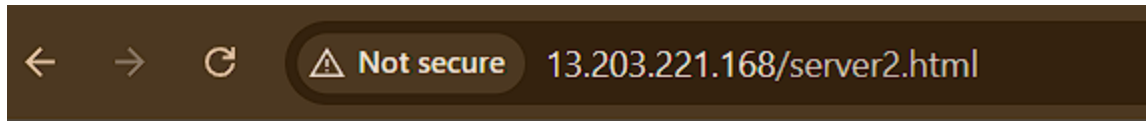
Welcome to home1



Welcome to server1



Welcome to home2



Welcome to server2

Amazon Machine Images (AMIs) (2) [Info](#)

Owned by me ▼

Find AMI by attribute or tag

<input type="checkbox"/>	Name	AMI name	AMI ID
<input type="checkbox"/>		AMI 1	ami-0d633ae518143881a
<input type="checkbox"/>		AMI 2	ami-0c7945cff581030eb

[EC2](#) > [Launch templates](#) > Create launch template

Create launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and have multiple versions.

Launch template name and description

Launch template name - *required*

Template1

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

Recents


My AMIs

Quick Start

☐ Don't include in launch template

☒ Owned by me

☐ Shared with me



Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

AMI 1

ami-0d633ae518143881a

2025-07-23T10:31:37.000Z Virtualization: hvm ENA enabled: true Root device type: ebs Boot mode: uefi-preferred

Description

Create launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared, and have multiple versions.

Launch template name and description

Launch template name - *required*

Template2

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

Template version description

Recents


My AMIs

Quick Start

☐ Don't include in launch template

☒ Owned by me

☐ Shared with me



Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

AMI 2

ami-0c7945cff581030eb

2025-07-23T10:32:02.000Z Virtualization: hvm ENA enabled: true Root device type: ebs Boot mode: uefi-preferred

Launch Templates (2) [Info](#)

<input type="checkbox"/>	Launch Template ID ▾	Launch Template Name ▾
<input type="checkbox"/>	lt-0b2f1913c495aece4	Template1
<input type="checkbox"/>	lt-0c78191ef41ba6795	Template2

Actions ▾

Launch instances ▲

Launch instances

Launch instance from template

Migrate a server

status

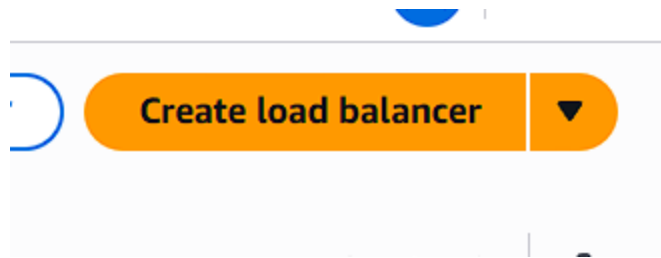
> Create target group

Register targets

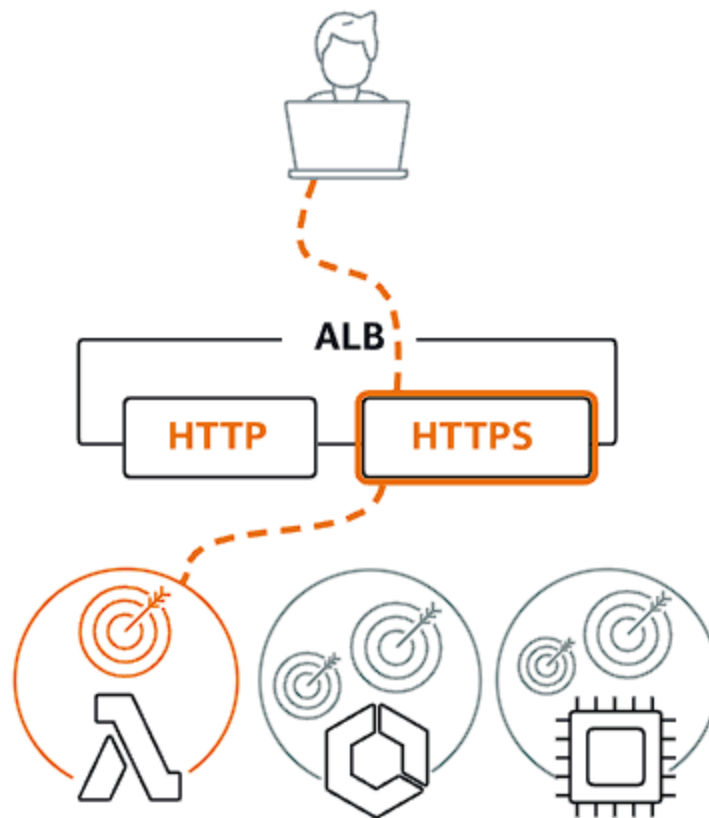
This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group, you must register at least one target.

Available instances (1/2)

<input type="checkbox"/>	Instance ID ▾	Name ▾	State ▾
<input checked="" type="checkbox"/>	i-0e1bec17f1dc4ce53	Instance2	✓ Running
<input type="checkbox"/>	i-048b2b77172e5c69a	Instance 1	✓ Running



Application Load Balancer [Info](#)



Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.

[Create](#)

Basic configuration

Load balancer name

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

LD

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must start with a letter and end with a letter or digit.

Scheme | [Info](#)

Scheme can't be changed after the load balancer is created.

- ☒ **Internet-facing**
 - Serves internet-facing traffic.
 - Has public IP addresses.
 - DNS name resolves to public IPs.
 - Requires a public subnet.

Load balancer IP address type | [Info](#)

Specify group details

Your load balancer routes requests to the targets in a target group and performs health


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 c



IP addresses

- Supports load balancing to VPC and on-premises resources.
- Facilitates routing to multiple IP addresses and network interfaces on the same ins
- Offers flexibility with microservice based architectures, simplifying inter-applicatio
- Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6



Lambda function

- Facilitates routing to a single Lambda function.
- Accessible to Application Load Balancers only.

Listeners and routing [Info](#)

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that y

▼ Listener HTTP:80

Protocol

HTTP



Port

80

1-65535

Default action | [Info](#)

Forward to

Select a target group

[Create target group](#) 

Register targets

This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group you must register your targets.

Available instances (1/2)

< 1 >

<input type="checkbox"/>	Instance ID	Name	State	Security groups	Zone
<input checked="" type="checkbox"/>	i-0e1bec17f1dc4ce53	Instance2	Running	Security	ap-south-1b
<input type="checkbox"/>	i-048b2b77172e5c69a	Instance 1	Running	Security	ap-south-1b

1 selected

> Create Auto Scaling group

Choose launch template [Info](#)

Specify a launch template that contains settings common to all EC2 instances that are lau

ptions

ices

scaling

Name

Auto Scaling group name

Enter a name to identify the group.

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 launch configurations is not recommended but still available via the CLI and AP

Launch template

Choose a launch template that contains the instance-level settings, such as the Amazon Machine Im

[Create a launch template](#)

Version



[Create a launch template version](#)

Choose instance launch options [Info](#)

Choose the VPC network environment that your instances are launched into, and customize the instance types and

ptions

as

caling

Instance type requirements [Info](#)

You can keep the same instance attributes or instance type from your launch template, or you can choose to o attributes or manually adding instance types.



Specify instance attributes

Provide your compute requirements. We fulfill your desired capacity with matching instance types based on your allocation strategy selection.



Manually add ins

Add one or more in: your desired capacit

Required instance attributes

Enter your compute requirements in virtual CPUs (vCPUs) and memory.

vCPUs

Enter the minimum and maximum number of vCPUs per instance.

minimum

maximum



No minimum



No maximum

Maximum vCPUs is required and must be greater than

Memory (GiB)

Enter the minimum and maximum GiBs of memory per instance.

minimum

maximum

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



aps1-az1 (ap-south-1a) | subnet-00bb156050e0ef090

172.31.32.0/20 Default

aps1-az2 (ap-south-1c) | subnet-090e19b6a34692968

172.31.16.0/20 Default

aps1-az3 (ap-south-1b) | subnet-0a14d154b1b33442a

172.31.0.0/20 Default

[Create a subnet](#)

Create Auto Scaling group

Health checks

Health checks increase availability by replacing unhealthy instances when a failure occurs.

EC2 health checks

[i](#) Always enabled

Additional health check types - *optional*

- ☐ Turn on Elastic Load Balancing health checks
Elastic Load Balancing monitors whether instances are healthy.
- ☐ Turn on VPC Lattice health checks
VPC Lattice can monitor whether instances are healthy.
- ☒ Turn on Amazon EBS health checks
EBS monitors whether an instance's root volume is healthy.

[i](#) EBS volumes that are attached to an instance are monitored for I/O errors, or deleted volumes (keeping data intact), or deleted snapshots of any critical EBS volumes.

Health check grace period | [Info](#)

This time period delays the first health check until the specified time.

60



seconds

Auto Scaling groups (2) Info Last updated less than a minute ago [Launch configurations](#) [Launch templates](#) [Actions](#) [Create Auto Scaling group](#)

Search your Auto Scaling groups

<input type="checkbox"/>	Name	Launch template/configuration	Instances	Status	Desired capacity	Min
<input type="checkbox"/>	autoscal2	temp1 Version Default	1	-	1	1
<input type="checkbox"/>	autoscale1	temp1 Version Default	1	-	1	1

Auto Scaling groups (1/2) Info Last updated less than a minute ago [Launch configurations](#) [Launch templates](#) [Actions](#) [Create Auto Scaling group](#)

Search your Auto Scaling groups

<input type="checkbox"/>	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max
<input type="checkbox"/>	autoscal2	temp1 Version Default	1	-	1	1	1
<input checked="" type="checkbox"/>	autoscale1	temp1 Version Default	1	-	1	1	1

[EC2](#) > [Auto Scaling groups](#) > [autoscale1](#) > [Edit](#)

EC2 Catalog

Amazon Elastic Block Store

Amazon EBS

Amazon EC2 Instance Profiles

Amazon EC2 Systems Manager

Amazon VPC

Amazon VPC Subnets

Amazon VPC Elastic IP Addresses

Amazon VPC Internet Gateways

Amazon VPC Network Interfaces

Amazon VPC VPC Peering Connections

Amazon VPC VPC Endpoints

Amazon VPC VPC Flow Logs

Amazon VPC VPC Lattice

Amazon VPC VPC Security Groups

Amazon VPC VPC Subnet Cidr Blocks

Amazon VPC VPC Subnet Filters

Amazon VPC VPC Subnet Tags

Amazon VPC VPC Subnet VPC Peering Connections

Amazon VPC VPC Subnet VPC Peering Connections

Amazon VPC VPC Subnet VPC Peering Connections

Load balancing - optional

Load balancers

☒ Application, Network or Gateway 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

Target1 | TCP
Network Load Balancer: LD

☐ Classic Load Balancers

Part B: Additional Topics

1. Advanced Auto Scaling Policies

- Add target tracking or step scaling:
 - CPU utilization > 70%
 - Request count > 100 per target
- Set cooldown period to avoid excessive scaling.

2. Cost Optimization

- Compare pricing of Linux vs Windows EC2 instances using AWS Pricing Calculator.
- Estimate costs for 1000 requests/hour with:
 - ALB
 - 2 EC2 instances
 - ASG usage
- Analyze cost differences and suggest optimization strategies.

3. High Availability Testing

- Terminate an EC2 instance in an ASG and verify auto-replacement.
- Confirm that instances are launched across two or more AZs.
- Validate traffic continuity through ALB during failover.

4. Security Enhancements

- Create a custom security group:
 - Allow HTTP (80), HTTPS (443)
 - Allow SSH only from your IP
- Use AWS Certificate Manager (ACM):
 - Request a public certificate
 - Attach it to the ALB to enable HTTPS

5. Infrastructure as Code (IaC)

- Use Terraform or AWS CDK to automate the setup:
 - Define VPC, ALB, ASGs, launch templates, and listener rules
- Store code in a structured folder (e.g., `IaC/`)