

Team name : Cloud-05

Madu Gundarapu ; [madhugundarapu.mca@gmail.com](mailto:madhugundarapu.mca@gmail.com)

Priyanaka Misal : [priyankamisal127@gmail.com](mailto:priyankamisal127@gmail.com)

Swagatika Giri : [swagatikagiri86@gmail.com](mailto:swagatikagiri86@gmail.com)

## **Project Statement: Implementing High Availability with Auto Scaling**

**Objective:** In this project, we'll dive into a cost-effective, yet efficient service that Amazon Web Services (AWS) provides: Amazon EC2 Auto-Scaling Groups (ASG).

Some of today's popular streaming and social media applications utilize AWS to make them highly available. The companies configure their Amazon EC2 Auto-Scaling Groups to increase and decrease resources according to demand or "scaling out" and "scaling in". Since AWS only charges for the resources used, this will save the company plenty of money.

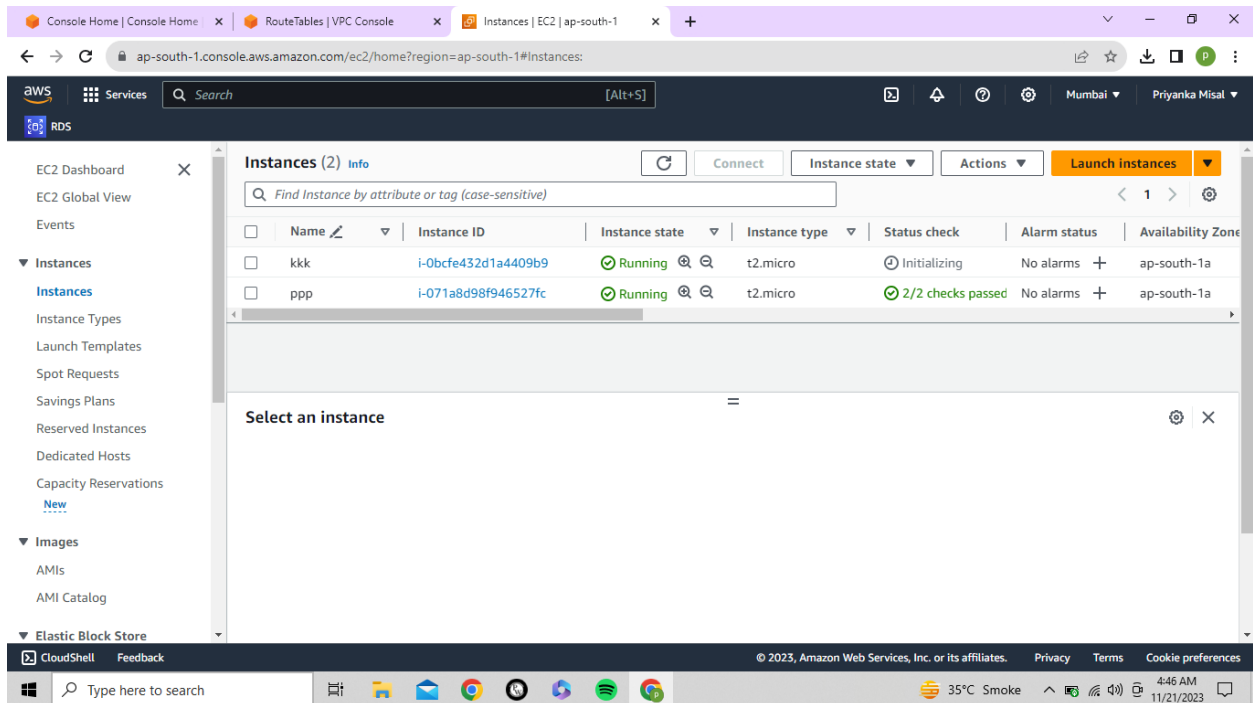
Feel free to follow the steps below as I show how you too, can create an EC2 Auto-Scaling Group!

### **STEP 1 : Create an 2 EC2 instances**

1. Open the EC2 Console. I confirmed my region : Mumbai
2. In the navigation pane, choose **EC2 Dashboard**, and then choose **Launch instance**. Under **Names and tags**, for **Name**, enter a name for your instance
3. Choose an **Amazon Machine Details(AMI)**
4. Choose an Instance Type: Select "**t2.micro**" instance type
5. Configure Instance Details: You can leave most of the default settings as is , You can specify the number of instances network settings and more if needed. Click "Next" when you are ready
6. Add Storage
7. Configure Security Group
8. Create a New "Key Pair" or use Existing One

9. Click On “Launch Instance”

10. After success it will appear As below



**STEP 2:** In the search bar, I typed in **VPC** then clicked on “Create VPC” on the dashboard

1. I did the following as seen in the picture below :o choose VPC only for resources create,named the VPC ,Entered 10.10.0.0/16 for the IPv4 CIDR block ,then click create VPC.

2. In order to connect my custom VPC to the internet, I created an internet gateway and attached it to the custom VPC.
3. I simply named the internet gateway and clicked create internet gateway.

VPC > Internet gateways > Create internet gateway

## Create internet gateway [Info](#)

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

### Internet gateway settings

**Name tag**  
Creates a tag with a key of 'Name' and a value that you specify.

### 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 or track your AWS costs.

Key	Value - optional	
<input type="text" value="Name"/>	<input type="text" value="acsb-wk7-igw"/>	<input type="button" value="Remove"/>

You can add 49 more tags.

4. I Was prompted to attach the internet gateway to a VPC as seen in the success message.
5. I clicked attach to a VPC ,selected my custom VPC created ,then clicked attach internetgateway.

**VPC**  
Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

**Available VPCs**  
Attach the internet gateway to this VPC.

► **AWS Command Line Interface command**

Cancel **Attach internet gateway**

6. Once Successfully attached ,you'll see the change of state on the internet gateway dashboard

### STEP 3:

1. I navigated back to the left panel and clicked **subnets**. On the new screen, I clicked **create subnet**, selected my custom **VPC**, then created **three public subnets** (in 3 different availability zones) with the following settings

#Subnet Name

Availability zone a

IPv4 CIDR Block : 10.10.1.0/24

#Subnet Name

Availability zone b

IPv4 CIDR Block : 10.10.2.0/24

#Subnet Name

Availability zone c

IPv4 CIDR Block : 10.10.3.0/

2. Below is an example of what my screen looked like after filling out the sections for my public subnets

### Subnet 1 of 3

#### Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

#### Availability Zone [Info](#)

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

#### IPv4 CIDR block [Info](#)

#### ▼ Tags - optional

##### Key

##### Value - optional

You can add 49 more tags.

### Subnet 2 of 3

#### Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

#### Availability Zone [Info](#)

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

#### IPv4 CIDR block [Info](#)

#### ► Tags - optional

- Once I created the public subnets and received a success message, I clicked on each individual subnet, clicked actions, edit subnet settings and enabled auto-assign of public IPv4 addresses. This way, any EC2 instances launched within the subnet will automatically be assigned an IPv4 address.

**Auto-assign IP settings** [Info](#)  
Enable the auto-assign IP settings to automatically request a public IPv4 or IPv6 address for a new network interface in this subnet.

☒ **Enable auto-assign public IPv4 address** [Info](#)

☐ **Enable auto-assign customer-owned IPv4 address** [Info](#)  
Option disabled because no customer owned pools found.

**Resource-based name (RBN) settings** [Info](#)  
Specify the hostname type for EC2 instances in this subnet and optional RBN DNS query settings.

☐ **Enable resource name DNS A record on launch** [Info](#)

☐ **Enable resource name DNS AAAA record on launch** [Info](#)

**Hostname type** [Info](#)

☐ Resource name

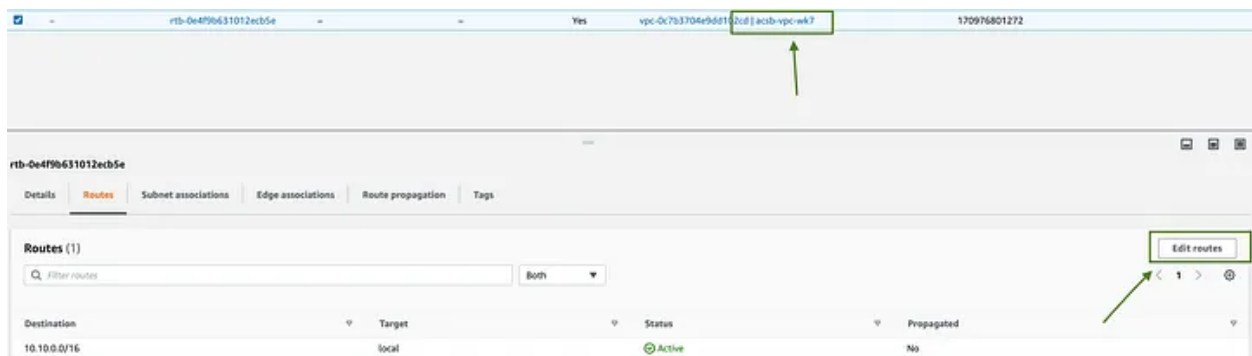
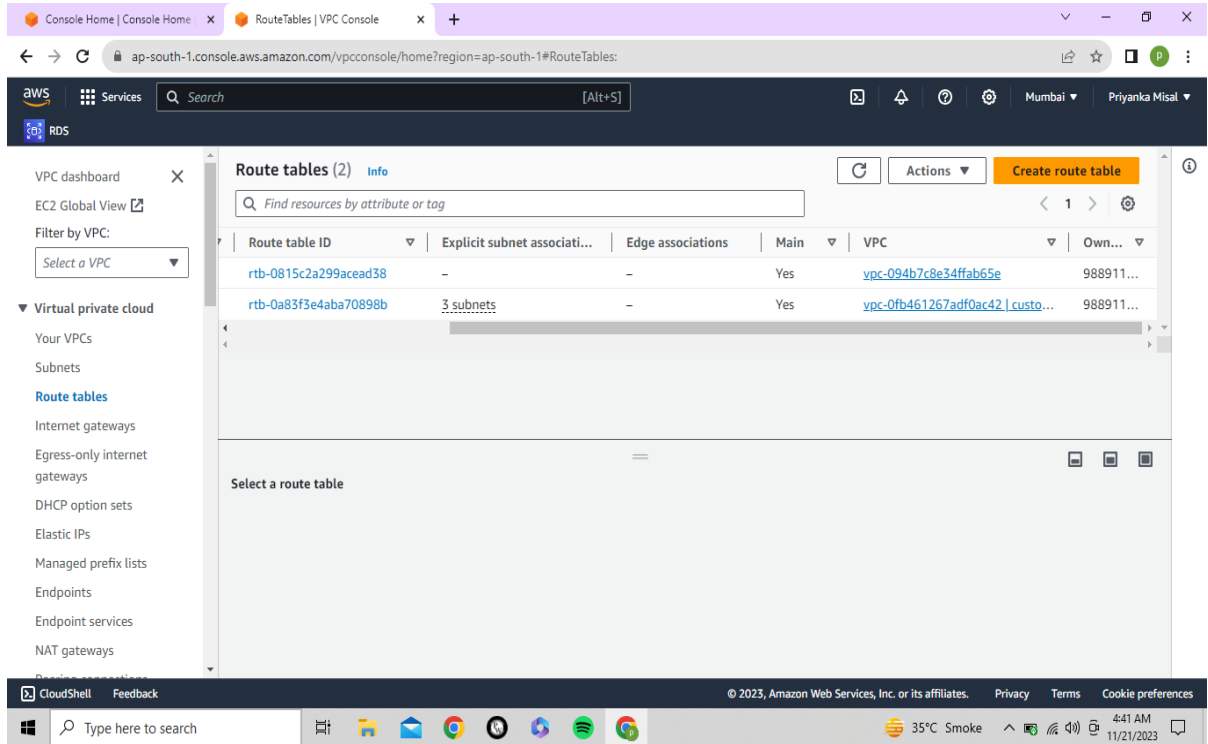
☒ IP name

**DNS64 settings**  
Enable DNS64 to allow IPv6-only services in Amazon VPC to communicate with IPv4-only services and networks.

☐ **Enable DNS64** [Info](#)

Cancel **Save**

- After creating my VPC, internet gateway and three public subnets, I needed to configure my route table. I navigated back to the panel on the left side of the VPC dashboard and clicked the tab route tables.
- Below is the route table that was created for my custom VPC named “my first igw”. I clicked routes, then edit routes.



6. I added the destination 0.0.0.0/0 to allow any IP address as the source, selected my new internet gateway as the target, then saved the changes.
7. Then, I edited my route table's explicit subnet associations.

Details Routes **Subnet associations** Edge associations Route propagation Tags

**Explicit subnet associations (0)**

Find subnet association

Subnet ID IPv4 CIDR IPv6 CIDR

No subnet associations  
You do not have any subnet associations.

**Subnets without explicit associations (3)**

The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

Find subnet association

Subnet ID IPv4 CIDR IPv6 CIDR

subnet-0702dce8b57f9c57 / acb-pubsub-2	10.10.2.0/24	=
subnet-0bfe02ccdda39e4f1 / acb-pubsub-1	10.10.1.0/24	=
subnet-0bc89c22de7b41f6 / acb-pubsub-3	10.10.3.0/24	=

8. I selected all three of my public subnets, as seen below, and saved the associations.

**Edit subnet associations**

Change which subnets are associated with this route table.

**Available subnets (3/3)**

Filter subnet associations

<input checked="" type="checkbox"/>	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/>	acb-pubsub-2	subnet-0702dce8b57f9c57	10.10.2.0/24	=	Main (rtb-0e4ff0a631013ac054)
<input checked="" type="checkbox"/>	acb-pubsub-1	subnet-0bfe02ccdda39e4f1	10.10.1.0/24	=	Main (rtb-0e4ff0a631013ac054)
<input checked="" type="checkbox"/>	acb-pubsub-3	subnet-0bc89c22de7b41f6	10.10.3.0/24	=	Main (rtb-0e4ff0a631013ac054)

**Selected subnets**

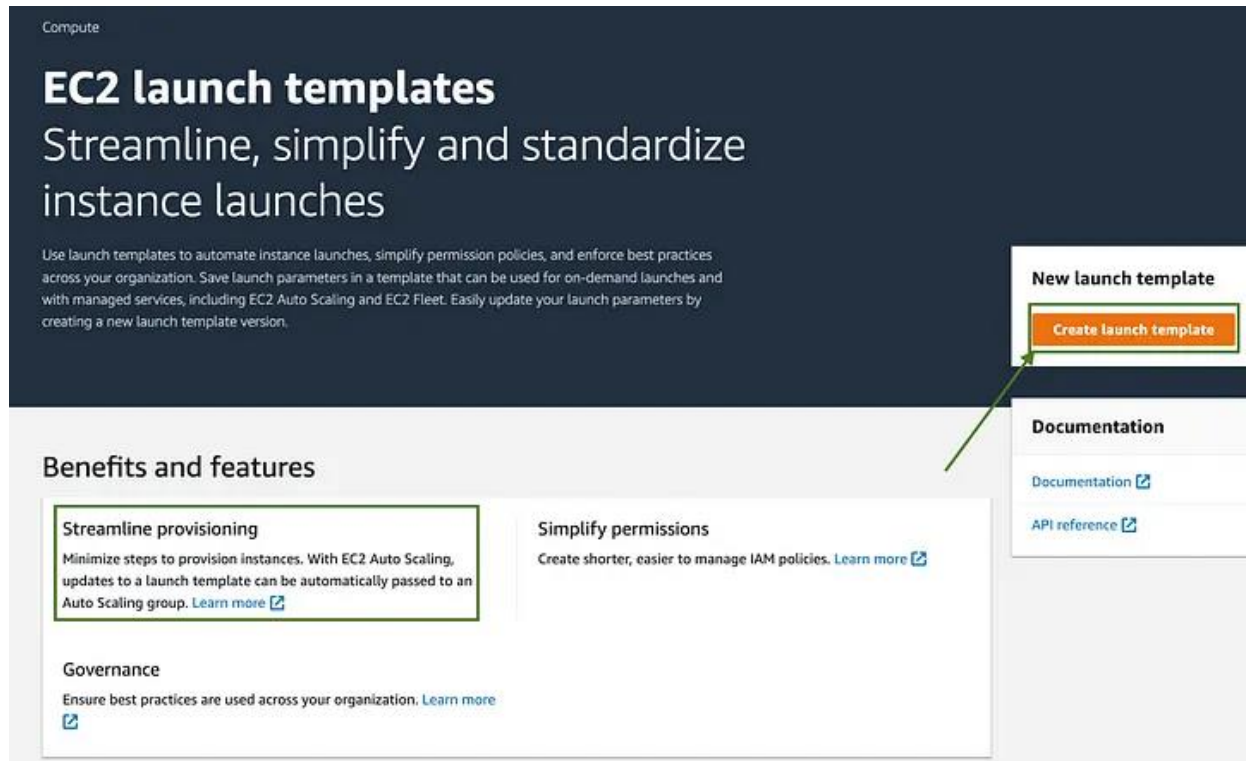
subnet-0bc89c22de7b41f6 / acb-pubsub-3 X subnet-0bfe02ccdda39e4f1 / acb-pubsub-1 X subnet-0702dce8b57f9c57 / acb-pubsub-2 X

Cancel **Save associations**

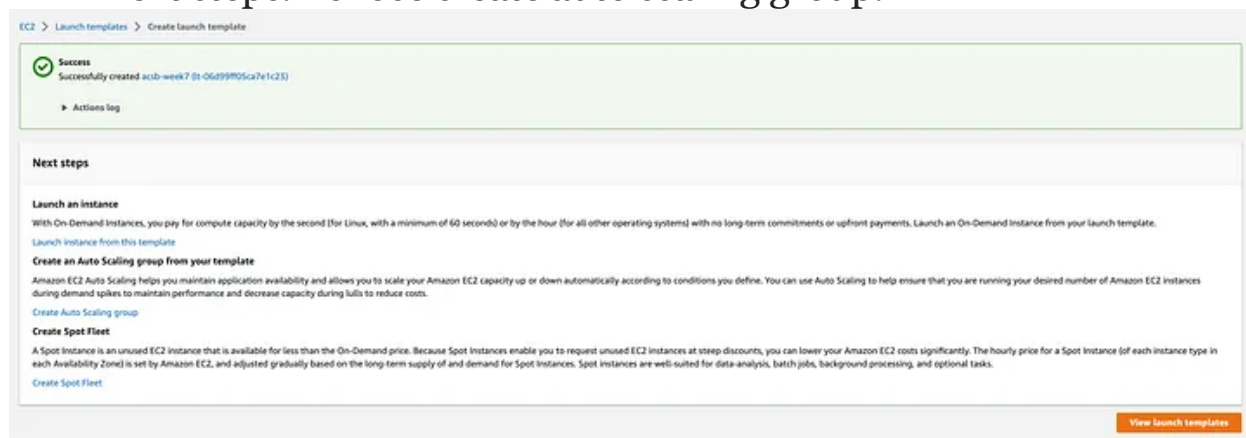


## STEP 4:

1. I typed EC2 in my search bar and navigated back to the EC2 dashboard. Under instances, I clicked launch templates.



2. I clicked create launch template, then received suggestions for next steps. I chose create auto scaling group.



## STEP 5

1. I did the following to create an ASG:
2. **Choose launch template or configuration (step 1):** named auto scaling group and chose the launch template I created in earlier steps.
3. **Choose instance launch options (step 2):** selected my custom VPC and selected all three public subnets created specifically for this project.
4. **Configure group size and scaling policies (step 4):** For purposes of this project, I entered desired capacity (2), minimum capacity (2), and maximum capacity (5).
5. Here is what to expect from each section, when creating an auto scaling group

**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.

acsb\_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.

acsb-week7

[Create a launch template](#)

**Version**  
Default (1)

[Create a launch template version](#)

<b>Description</b> -	<b>Launch template</b> <a href="#">acsb-week7</a> lt-06d99ff05ca7e1c23	<b>Instance type</b> t2.micro
<b>AMI ID</b> ami-0b5eea76982371e91	<b>Security groups</b> -	<b>Request Spot Instances</b> No
<b>Key pair name</b> luitproject	<b>Security group IDs</b> -	

**Additional details**

<b>Storage (volumes)</b> -	<b>Date created</b> Mon Dec 26 2022 15:06:44 GMT-0500 (Eastern Standard Time)
-------------------------------	---

Cancel **Next**

**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-0c7b3704e9dd102cd (acsb-vpc-wk7)  
10.10.0.0/16

[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-0bfe02ccdda39e4f1 (acsb-pubsub-1)  
10.10.1.0/24

us-east-1b | subnet-0702dce8b57f9ccb7 (acsb-pubsub-2)  
10.10.2.0/24

us-east-1c | subnet-0bc89c622de7b41f6 (acsb-pubsub-3)  
10.10.3.0/24

[Create a subnet](#)

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

**Launch template**  
acsb-wk7 [Info](#)  
lt-06d99ff05ca7e1c23

**Version**  
Default

**Description**  
-

**Instance type**  
t2.micro

[Override launch template](#)

[Cancel](#) [Previous](#) [Skip to review](#) [Next](#)

## Configure group size and scaling policies [Info](#)

Set the desired, minimum, and maximum capacity of your Auto Scaling group. You can optionally add a scaling policy to dynamically scale the number of instances in the group.

**Group size - optional** [Info](#)

Specify the size of the Auto Scaling group by changing the desired capacity. You can also specify minimum and maximum capacity limits. Your desired capacity must be within the limit range.

**Desired capacity**  
2

**Minimum capacity**  
2

**Maximum capacity**  
5

**Scaling policies - optional**

Choose whether to use a scaling policy to dynamically resize your Auto Scaling group to meet changes in demand. [Info](#)

☐ **Target tracking scaling policy**  
Choose a desired outcome and leave it to the scaling policy to add and remove capacity as needed to achieve that outcome.

☒ **None**

**Instance scale-in protection - optional**

**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**

[Cancel](#) [Previous](#) [Skip to review](#) [Next](#)

**Step 4: Configure group size and scaling policies** Edit

Group size		
Desired capacity	Minimum capacity	Maximum capacity
2	2	5

**Scaling policy**

No scaling policy

**Instance scale-in protection**

Instance scale-in protection  
☐ Enable instance protection from scale in

**Step 5: Add notifications** Edit

**Notifications**

No notifications

**Step 6: Add tags** Edit

**Tags (0)**

Key	Value	Tag new instances
No tags		

Cancel Create Auto Scaling group

6. The ASG has been successfully created

Console Home | [RouteTables | VPC Co](#) | [Auto Scaling groups](#) | [Auto Scaling groups](#) | [Target group details](#) | [\(2\) WhatsApp](#)

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#AutoScalingGroups

**Auto Scaling group updated successfully**

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

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
<input type="checkbox"/>	<a href="#">autoscaling group</a>	<a href="#">myfirsttemplate</a>   Version Default	2	-	2

0 Auto Scaling groups selected

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

5:28 AM 11/21/2023

## STEP 6 :

1. Next, I needed to create an application load balancer (ALB), so I went back to the EC2 panel on the left, clicked load balancers, then clicked on create load balancer.
2. **For load balancer types:** I selected Application Load Balancer
3. **Basic configuration:** named my load balancer, selected internet-facing for the scheme, and kept IPv4 for IP address type.

### Create Application Load Balancer [Info](#)

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and containers, based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply, and if applicable, it selects a target from the target group for the rule action.

► **How Elastic Load balancing works**

---

#### Basic configuration

**Load balancer name** [Info](#)  
Name must be unique within your AWS account and cannot 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 cannot 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.

4. **Network mapping:** Used the dropdown menu to choose my VPC otherwise it would've used the default VPC. Selected each of my public subnets that I created earlier.

**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. Only VPCs with an internet gateway are enabled for selection. The selected VPC cannot be changed after the load balancer is created. To confirm the VPC for your targets, view your [target groups](#).

acsb-vpc-wk7  
vpc-0c7b5704e9dd102cd  
IPv4: 10.10.0.0/16

**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. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

☒ **us-east-1a (use1-az2)**

Subnet  
subnet-0bfe02ccdda39e4f1 acsb-pubsub-1

**IPv4 settings**  
Assigned by AWS

☒ **us-east-1b (use1-az4)**

Subnet  
subnet-0702dce8b37f9ccb7 acsb-pubsub-2

**IPv4 settings**  
Assigned by AWS

☒ **us-east-1c (use1-az6)**

Subnet  
subnet-0bc89c622de7b41f6 acsb-pubsub-3

**IPv4 settings**  
Assigned by AWS

5. **Security Groups:** Selected the security group I created earlier from the dropdown menu.

**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 [Info](#)

acsb-sg sg-09f05aa04e0a94183 X  
VPC: vpc-0c7b5704e9dd102cd

6. **Listeners and routing:** I changed the protocol to HTTP , then clicked create target group. I was taken to a separate window to create the target group so that I could come back to the page and refresh it. If you have a target group created already, you should be able to select it from the dropdown menu under default action.

### 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

Remove

Protocol

Port

Default action

[Info](#)

HTTP

:

80

Forward to

Select a target group

1-65535

Create target group

Listener tags - optional

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

Add listener tag

You can add up to 50 more tags.

Add listener

7. **To create a target:** I selected instances for the target type, named the target group, selected HTTP for the protocol, my custom VPC, and protocol version HTTP1.

Specify group details

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

Basic configuration

Settings in this section cannot 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.

☐ IP addresses

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

☐ Lambda function

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

☐ Application Load Balancer

- Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
- Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

Target group name

ACSB-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.

acsb-vpc-wk7  
vpc-0c7b3704e9dd102cd  
IPv4: 10.10.0.0/16

Protocol version

☒ HTTP1

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

8. **Health checks:** I selected HTTP for health check protocol and used the default path / for 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.

/

Up to 1024 characters allowed.

▶ Advanced health check settings

### Attributes

ⓘ Certain default attributes will be applied to your target group. You can view and edit them after creating the target group.

▶ Tags - optional

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

CancelNext

9. I clicked next and needed to select both of the instances to create the target group.

Console Home | Con... x | RouteTables | VPC Co... x | Auto Scaling groups... x | Auto Scaling groups... x | Target group details | x | (2) WhatsApp... x | + | - | x

← → ↻ ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#TargetGroup:targetGroupArn=arn:aws:elasticloadbalancing:ap-south-1:988911902283:targetg... ☆ ⬇️ Ⓜ️

aws Services 🔍 Search [Alt+S] Mumbai Priyanka Misal

RDS

EC2 Dashboard x

EC2 Global View

Events

▼ Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

New

▼ Images

AMIs

AMI Catalog

▼ Elastic Block Store

CloudShell Feedback

🟢 Successfully created the target group: firsttg.

EC2 > Target groups > firsttg

firsttg Actions

Details

arn:aws:elasticloadbalancing:ap-south-1:988911902283:targetgroup/firsttg/38b414284db23c74

Target type	Protocol : Port	Protocol version	VPC
Instance	HTTP: 80	HTTP1	vpc-0fb461267adf0ac42
IP address type	Load balancer		
IPv4	None associated		

Total targets	Healthy	Unhealthy	Unused	Initial	Draining
0	0	0	0	0	0

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

35°C Smoke 5:28 AM 11/21/2023



10. With the target group created, I went back to the window where I was creating my application load balancer, clicked the refresh button, and selected my target group from the dropdown menu.

**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**

Remove

Protocol

Port

Default action

[Info](#)

HTTP

:

80

Forward to

Select a target group

1-65535

1-65535

Create target

Q |

ACSB-TG

Target type: Instance, IPv4

HTTP

↻

**Listener tags - optional**

Consider adding tags to your listener. Tags enable you to categorize your AWS resources to make them easier to manage.

Add listener tag

You can add up to 50 more tags.

Add listener

11. I clicked create load balancer and successfully created the application load balancer.

Console Home | Console | RouteTables | VPC Co | Auto Scaling groups | Load balancers | EC2 | Target group details | (2) WhatsApp

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

**Load balancers (1)**

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

Filter load balancers

	Name	DNS name	State	VPC ID	Availability Zones
<input type="checkbox"/>	myfirsttargetgroup	myfirsttargetgroup-90101...	Provisioning..	vpc-0fb461267adf0ac42	3 Availability Zones

0 load balancers selected

Select a load balancer above.

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

35°C Smoke 5:24 AM 11/21/2023

12. I checked off the box application, network, or gateway load balancer target groups, selected my target group “ACSB-TG”, then clicked *update* and now we’re ready to roll.

Edit acsb\_asg [Info](#)

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

ACSB-TG | HTTP  
Application Load Balancer: acsb-ALB

☐ Classic Load Balancers

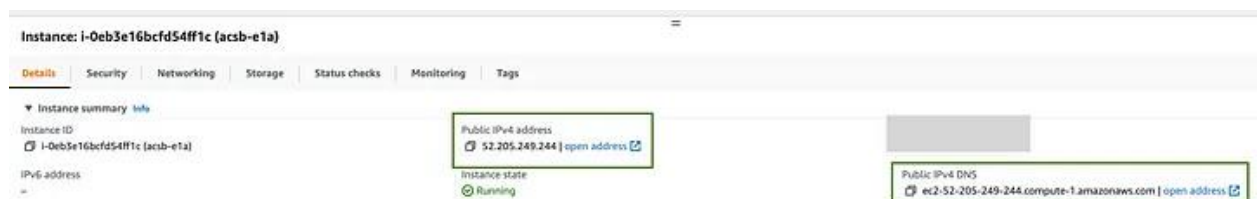
Create and attach new load balancers

Add a new load balancer

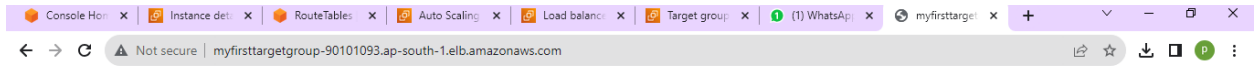
Cancel **Update**

## STEP 7:

1. I moved back to the EC2 dashboard, clicked instances (running), checked off my first EC2 instance, copied the public IPv4 address, pasted it into a new browser window, and was able to see the Apache test page! I repeated the same process for my second EC2 instance as well to confirm its health.



2. For my application load balancer, I copied the DNS name, pasted it into a new browser



**It works!**



**we have successfully created a Implementing High Availability with Auto Scaling**

**THANK YOU!**