



VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Fall Semester 2025-26

Lab Assignment – 4

Slot: L13+L14

Class: VL2025260105679

Branch: B.tech CSBS

Course code & title: CBS3005

Cloud, Microservices and Applications

Faculty name: Nithya K

DA by:

Kartikey Gupta

22BBS0105

1) A company wants to deploy a secure, scalable, and highly available web application on AWS for global users. Perform the following tasks in AWS and submit screenshots of each step as evidence:

- (i) Launch multiple EC2 instances (web servers) and configure them in different Availability Zones.
- (ii) Create an Application Load Balancer (ALB) to distribute traffic across these instances.
- (iii) Configure health checks so that faulty instances are automatically removed from load balancing.
- (iv) Enable Auto Scaling to add/remove instances based on traffic demand.
- (v) Configure path-based routing: /auth requests go to the authentication service, /order requests go to the order processing service. Integrate the Load Balancer with Route 53 so that global users are routed to the nearest AWS region.

(i) Launch Multiple EC2 Instances in Different Availability Zones

1. **Log in to AWS Management Console → go to EC2.**

The screenshot shows the AWS Management Console EC2 dashboard for the Europe (Stockholm) region. The left sidebar includes links for Dashboard, AWS Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, and Network & Security, Security Groups. The main content area displays the following sections:

- Resources:** Shows 0 Instances (running), 0 Auto Scaling Groups, 0 Capacity Reservations, 0 Dedicated Hosts, 0 Elastic IPs, 0 Instances, 3 Key pairs, 0 Load balancers, 0 Placement groups, 4 Security groups, 0 Snapshots, 0 Volumes.
- Launch instance:** Buttons for "Launch instance" and "Migrate a server". Note: Your instances will launch in the Europe (Stockholm) Region.
- Service health:** Region: Europe (Stockholm). Status: This service is operating normally.
- Zones:** Shows three availability zones: eu-north-1a (Zone ID: eun1-az1), eu-north-1b (Zone ID: eun1-az2), and eu-north-1c (Zone ID: eun1-az3).
- Account attributes:** Default VPC: vpc-0f13ebc96a105497c. Settings: Data protection and security, Allowed AMIs, Zones, EC2 Serial Console, Default credit specification, EC2 console preferences.
- Explore AWS:** Get Up to 40% Better Price Performance, Optimize EC2 Cost with Spot Instances and EC2 Auto Scaling, 10 Things You Can Do Today to Reduce AWS Costs.

2. Click **Launch Instance** → choose Amazon Linux 2 / Ubuntu AMI.

VIT Vellore - VTOP

Launch an instance | EC2 | eu-north-1

eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#LaunchInstances:

Search [Alt+S]

EC2 > Instances > Launch an instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name Add additional tags

Application and OS Images (Amazon Machine Image) 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

Recents Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux Debian

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI
ami-043339ea831b48099 (64-bit (x86), uefi-preferred) / ami-0d5ee4c9faade005 (64-bit (Arm), uefi)
Free tier eligible

Virtualization: hvm ENA enabled: true Root device type: ebs

CloudShell Feedback

3 cm of rain Sunday

Summary

Number of instances Info

1

Software Image (AMI)

Amazon Linux 2023 AMI 2023.8.2... [read more](#)
ami-043339ea831b48099

Virtual server type (instance type)

t3.micro

Firewall (security group)

New security group

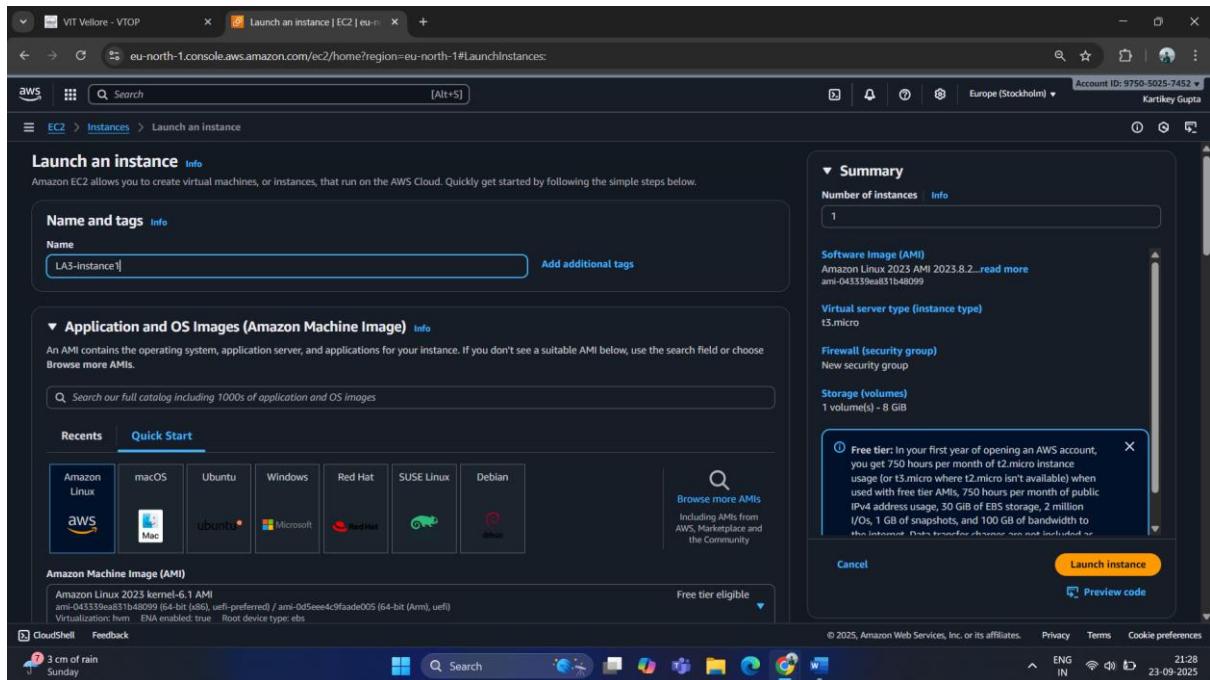
Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 50 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GiB of bandwidth to the Internet. Data transfer charges are not included.

Cancel [Launch instance](#) [Preview code](#)

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 21:28 23-09-2025



VIT Vellore - VTOP

Launch an instance | EC2 | eu-north-1

eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#LaunchInstances:

Search [Alt+S]

EC2 > Instances > Launch an instance

Network settings Info

VPC - required Info

vpc-0f13eb96a105497c (default) [Create new subnet](#) [Create security group](#)

Subnet Info

No preference [Create new subnet](#)

Availability Zone Info

eu-north-1a eun1-az1 [Enable additional zones](#)

Auto-assign public IP Info

Enable

Additional charges apply when outside of free tier allowance.

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.

Create security group Select existing security group

Security group name - required

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _/-/!@#\$%^&*`~`\$^`!

Description - required Info

lauch-wizard-4 created 2025-09-23T15:55:00.675Z

Summary

Number of instances Info

1

Software Image (AMI)

Amazon Linux 2023 AMI 2023.8.2... [read more](#)
ami-043339ea831b48099

Virtual server type (instance type)

t3.micro

Firewall (security group)

New security group

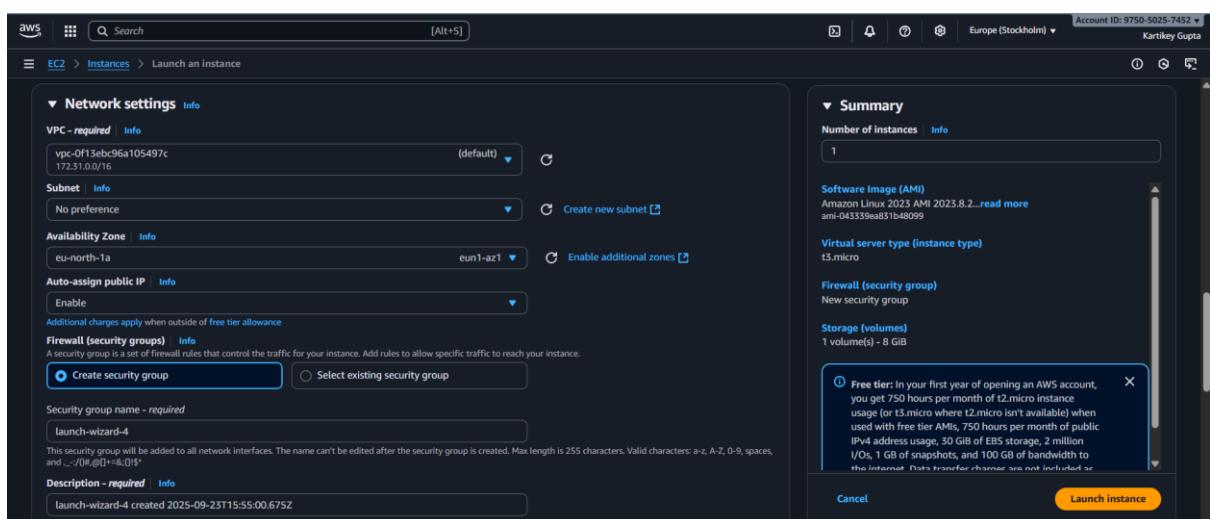
Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 50 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GiB of bandwidth to the Internet. Data transfer charges are not included.

Cancel [Launch instance](#) [Preview code](#)

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 21:28 23-09-2025



Inbound Security Group Rules

- Security group rule 1 (TCP, 22, 0.0.0.0/0)**
 - Type: ssh
 - Protocol: TCP
 - Port range: 22
 - Description - optional: e.g. SSH for admin desktop
- Security group rule 2 (TCP, 80, 0.0.0.0/0)**
 - Type: HTTP
 - Protocol: TCP
 - Port range: 80
 - Description - optional: e.g. SSH for admin desktop

⚠️ Rules of source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[Add security group rule](#)

Summary

Number of instances: 1

Virtual server type (instance type): t3.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

[Cancel](#) [Launch instance](#) [Preview code](#)

Network settings

VPC - required

- VPC: vpc-0f13eb96a105497c (default)
- Subnet: No preference
- Availability Zone: eu-north-1b
- Auto-assign public IP: Enable

Firewall (security groups)

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.

Create security group Select existing security group

Security group name - required

Security group rule 1 (TCP, 22, 0.0.0.0/0)

Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.8.2 [read more](#)

Virtual server type (instance type): t3.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

[Cancel](#) [Launch instance](#) [Preview code](#)

3. Select an **instance type** (e.g., t2.micro for free tier).
4. In **Network settings**:
 - Choose a **VPC** (default or custom).
 - Place **Instance 1** in **Availability Zone A** (e.g., ap-south-1a).
 - Place **Instance 2** in **Availability Zone B** (e.g., ap-south-1b).
5. Configure **Security Group**:
 - Allow **HTTP** (port 80) and **SSH** (port 22).
6. Launch instances and note **public IPs**.

| Instances (2) Info | | | | | | | | |
|--|---------------|---------------------|--|---------------|--|-------------------------------|-------------------|----------------------------------|
| <input type="text" value="Find Instance by attribute or tag (case-sensitive)"/> <input type="button" value="Clear filters"/> All states ▾ | | | | | | | | |
| | Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 DNS |
| <input type="checkbox"/> | LA5-Instance2 | i-0fffdc42a22068e31 | Running | t3.micro | 3/3 checks passed | View alarms + | eu-north-1b | ec2-16-170-245-220.eu... 16.170 |
| <input type="checkbox"/> | LA5-Instance1 | i-02490c1b088537c1f | Running | t3.micro | 3/3 checks passed | View alarms + | eu-north-1a | ec2-51-21-219-2.eu-no... 51.21.2 |

(ii) Create Application Load Balancer (ALB)

Step 1: Create Target Group(s) first

1. Go to EC2 → Target Groups → Create target group.
2. Configure:
 - **Target type:** Instances
 - **Name:** e.g., MyApp-TG
 - **Protocol:** HTTP
 - **Port:** 80
 - **VPC:** Select the same VPC where your EC2 instances are launched
 - **Health check:** HTTP, path /

The screenshot shows the 'Specify group details' step of the 'Create target group' wizard. The 'Basic configuration' section is visible, showing the selected target type as 'Instances'. Other options like 'IP addresses', 'Lambda function', and 'Application Load Balancer' are also listed. The interface includes standard AWS navigation elements like CloudShell, Feedback, and a bottom navigation bar with links for Rain warning, Search, and various AWS services.

Target group name
MyApp-TG
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.
80
1-65535

IP address type
Only targets with the indicated IP address type can be registered to this target group.
 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.

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.
vpc-0f1Seb96a105497c (default)

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

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 perform health checks on the root, or specify a custom path if preferred.
/

Up to 1024 characters allowed.

Ports for the selected instances
Ports for routing traffic to the selected instances.
80
1-65535 (separate multiple ports with commas)

Review targets

| Targets (2) | Remove all pending | | | | | | | | | | |
|---------------------|--------------------|------|---------|-----------------|-------------|----------------------|----------------------|--|--|--|--|
| Filter targets: | | | | | | | | | | | |
| Instance ID | Name | Port | State | Security groups | Zone | Private IPv4 address | Subnet ID | | | | |
| i-0ffdc42a22068e31 | LA3-Instance2 | 80 | Running | launch-wizard-5 | eu-north-1b | 172.31.32.212 | subnet-07a84a48af02e | | | | |
| i-02490c1b088537c1f | LA3-Instance1 | 80 | Running | launch-wizard-4 | eu-north-1a | 172.31.23.94 | subnet-0f44bdb978301 | | | | |

2 pending

Cancel Previous Create target group

3. Click **Next** and **register your EC2 instances** in this Target Group.

4. Complete Target Group creation.

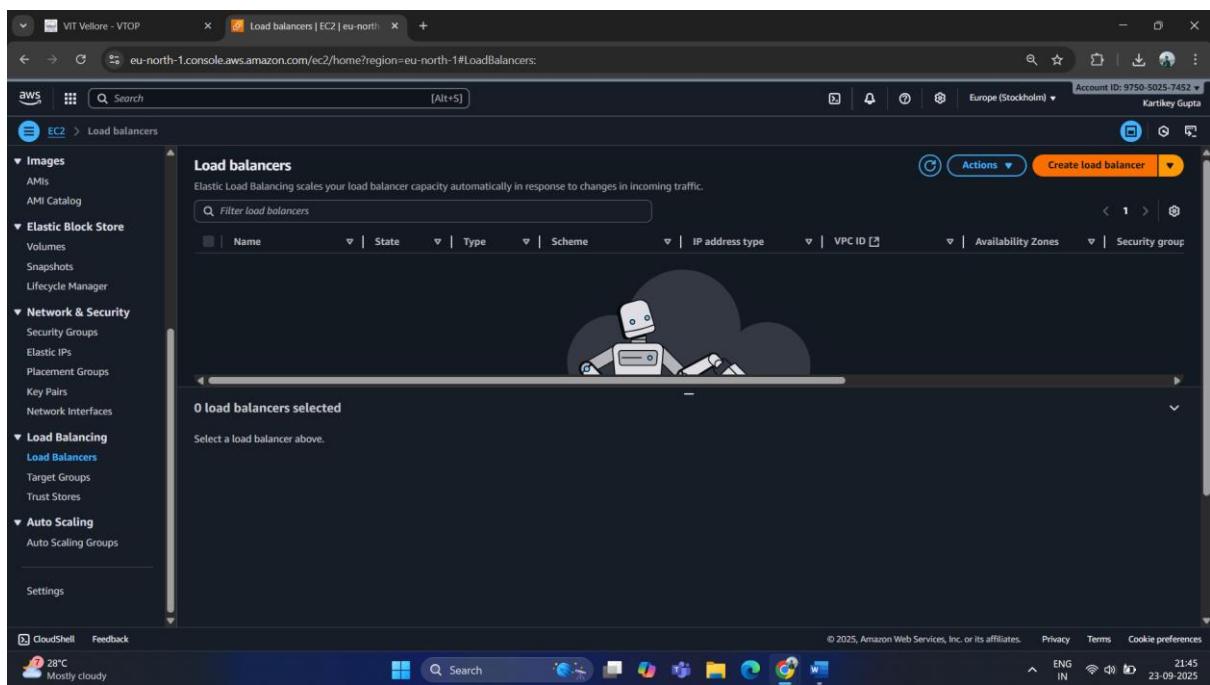
5. Install servers

- sudo yum update -y
- sudo yum install -y httpd
- sudo systemctl start httpd

- sudo systemctl enable httpd
- echo "Hello from \$(hostname -f)" > /var/www/html/index.html

Step 2: Create ALB

1. Go to **EC2 → Load Balancers → Create Load Balancer → Application Load Balancer.**
2. Configure:
 - **Name:** MyAppALB
 - **Scheme:** Internet-facing
 - **IP address type:** IPv4
 - **Listeners:** HTTP (80)
3. Select **Availability Zones:** choose at least 2 public subnets in different AZs.



Compare and select load balancer type

A complete feature-by-feature comparison along with detailed highlights is also available. [Learn more](#)

| Load balancer types |
|---|
| Application Load Balancer Info |
| |
| Choose an Application Load Balancer when you need a flexible front end for your applications using HTTP and HTTPS traffic. Optimize at the application level. Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers. |
| Create |
| Network Load Balancer Info |
| |
| Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, or fast deployment, support for UDP, and static IP addresses for your applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies. |
| Create |
| Gateway Load Balancer Info |
| |
| Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls. |
| Create |

CloudShell Feedback 28°C Mostly cloudy © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 21:46 23-09-2025

1. Configure:

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 Application Load Balancers work

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

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

 Internal

- Serves internal traffic.
- Has private IP addresses.
- DNS name resolves to private IPs.
- Compatible with the IPv4 and Dualstack IP address types.

Load balancer IP address type [Info](#)
Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses have an additional cost.
 IPv4
 Includes only IPv4 addresses.

CloudShell Feedback 28°C Mostly cloudy © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 21:47 23-09-2025

Network mapping Info

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC Info

The load balancer will exist and scale within the selected VPC. The selected VPC is also where the load balancer targets must be hosted unless routing to Lambda or on-premises targets, or if using VPC peering. To confirm the VPC for your targets, view [target groups](#).

vpc-0f15ebc96a105497c
172.31.0.0/16 (default)

IP pools Info

You can optionally choose to configure an IPAM pool as the preferred source for your load balancers IP addresses. Create or view Pools in the [Amazon VPC IP Address Manager console](#).

Use IPAM pool for public IPv4 addresses

The IPAM pool you choose will be the preferred source of public IPv4 addresses. If the pool is depleted IPv4 addresses will be assigned by AWS.

Availability Zones and subnets Info

Select at least two Availability Zones and a subnet for each zone. A load balancer node will be placed in each selected zone and will automatically scale in response to traffic. The load balancer routes traffic to targets in the selected Availability Zones only.

eu-north-1a (eun1-az1)
Subnet
Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.
subnet-0f44b9db97830fa751
IPv4 subnet CIDR: 172.31.16.0/20

eu-north-1b (eun1-az2)
Subnet
Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.
subnet-0784a8af026ef07
IPv4 subnet CIDR: 172.31.32.0/20

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 21:47 ENG IN 23-09-2025

Step 3: Configure Security Group

1. Either select an existing SG or create a new one.
2. **Inbound rule:** HTTP, port 80, source 0.0.0.0/0
3. Outbound can remain default (All traffic).

Step 4: Attach Target Group

1. Under **Listener → Default action**, select **Forward to → Your Target Group** (e.g., MyApp-TG).

Listeners and routing Info

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

Listener HTTP:80

| | |
|-----------------|---------------|
| Protocol | Port |
| HTTP | 80 1-65535 |

Default action Info

The default action is used if no other rules apply. Choose the default action for traffic on this listener.

Routing action

Forward to target groups Redirect to URL Return fixed response

Forward to target group Info

Choose a target group and specify routing weight or [create target group](#).

| Target group | Weight | Percent |
|--|------------|---------|
| MyApp-TG Target type: Instance, IPv4 Target stickiness: Off | 1 0-999 | 100% |

+ Add target group
You can add up to 4 more target groups.

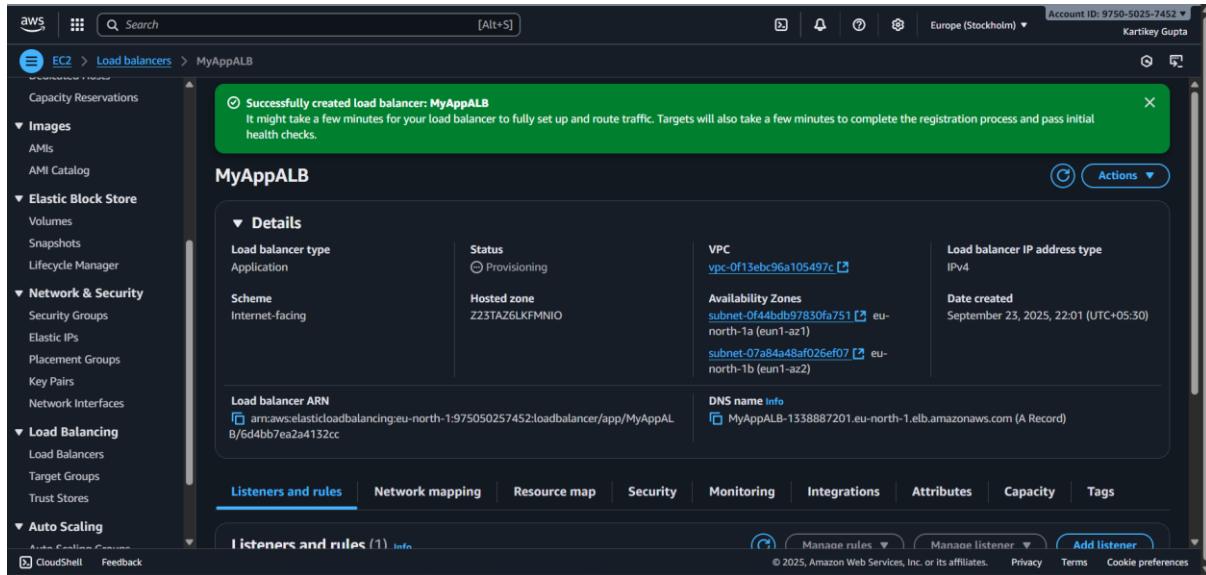
Target group stickiness Info

Enables the load balancer to bind a user's session to a specific target group. To use stickiness the client must support cookies. If you want to bind a user's session to a specific target, turn on the Target Group attribute Stickiness.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 5: Review and Create

- Review all settings and click **Create Load Balancer**.
- Note the **ALB DNS name** — this is what users will access.



Step (iii): Configure Health Checks

Step 1: Open Target Groups

1. Go to **EC2 → Target Groups** in the AWS console.
2. Select the **Target Group** you created (e.g., MyApp-TG).

Step 2: Go to Health Checks

1. In the Target Group details, click the **Health checks** tab.

Step 3: Configure Health Check Settings

- **Protocol:** HTTP
- **Path:** / (or /health if you have a dedicated health endpoint)
- **Healthy threshold:** 2 → instance must pass 2 consecutive checks to be marked healthy
- **Unhealthy threshold:** 2 → instance must fail 2 consecutive checks to be marked unhealthy
- **Timeout:** 5 seconds → how long to wait for a response
- **Interval:** 30 seconds → time between 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 perform health checks on the root, or specify a custom path if preferred.
/

Up to 1024 characters allowed.

Advanced health check settings

Healthy threshold
The number of consecutive health checks successes required before considering an unhealthy target healthy.
2

Unhealthy threshold
The number of consecutive health check failures required before considering a target unhealthy.
2

Timeout
The amount of time, in seconds, during which no response means a failed health check.
5 seconds

Interval
The approximate amount of time between health checks of an individual target.
30 seconds

Success codes
The HTTP codes to use when checking for a successful response from a target. You can specify multiple values (for example, "200,202") or a range of values (for example, "200-299").
200

Restore defaults

Cancel **Save changes!**

Step 4: Save

- Click **Save** to apply health check settings.

Step 5: Verify

- Go back to **Targets** tab in the Target Group.
- After a few seconds/minutes, the **Status** column should show **healthy** for your EC2 instances.
- If an instance fails, check:
 - Security group allows inbound HTTP from ALB
 - Web server is running and responding at the path you configured

EC2 > **Target groups** > **MyApp-TG**

MyApp-TG

Details

arn:aws:elasticloadbalancing:eu-north-1:975050257452:targetgroup/MyApp-TG/bf29b7ca95e8265f

| Target type | Protocol : Port | Protocol version |
|-----------------|-----------------|-----------------------|
| Instance | HTTP: 80 | HTTP |
| IP address type | Load balancer | VPC |
| IPv4 | MyAppLB | vpc-0f13ebc96a105497c |

2 Total targets | 0 Healthy | 0 Unhealthy | 0 Unused | 0 Initial | 0 Draining | 0 Anomalous

Distribution of targets by Availability Zone (AZ)

| Availability Zone | Count |
|-------------------|-------|
| eu-north-1a | 1 |
| eu-north-1b | 1 |

Targets | Monitoring | Health checks | Attributes | Tags

Registered targets (2) Info

Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group settings.

Anomaly mitigation: Not applicable | Deregister | Register targets

© 2025, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences

Step (iv): Enable Auto Scaling

Step 1: Prepare a Launch Template

1. Go to **EC2** → **Launch Templates** → **Create Launch Template**.
2. Configure template:
 - o **Name:** MyApp-LT
 - o **AMI:** Use the same AMI as your EC2 instances

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 launched at a later time. Templates can have multiple versions.

Launch template name and description

Launch template name - required
MyApp-LT

Template version description
A prod webserver for MyApp

Summary

Software Image (AMI)
Amazon Linux 2023 AMI 2023.8.2... [read more](#)
ami-04539ea8a31b48099

Virtual server type (instance type)

Firewall (security group)

Storage (volumes)
1 volume(s) - 8 GiB

Application and OS Images (Amazon Machine Image) 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](#).

Recent AMIs: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, Debi

Search: Q Amazon linux

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI
ami-043339ea831b48099 (64-bit (x86), uefi-preferred) / ami-0d5eee4c9faade005 (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2023 (kernel-6.1) is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

Amazon Linux 2023 AMI 2023.8.20250915.0 x86_64 HVM kernel-6.1

Summary

Software Image (AMI)
Amazon Linux 2023 AMI 2023.8.2... [read more](#)
ami-043339ea831b48099

Virtual server type (instance type)

Firewall (security group)

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. Data transfer

Create launch template

- **Instance type:** e.g., t2.micro
- **Key pair:** Your existing key
- **Security group:** Same as your running EC2 (allow HTTP 80 from ALB SG)

Network settings Info

Subnet | Info

Don't include in launch template

Availability Zone | Info

Don't include in launch 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

Security groups | Info

Select security groups

launch-wizard-4 sg-07d2c65d95f4334a6 X
VPC: vpc-0f13ebc96a105497c

launch-wizard-5 sg-0c146159be9988032 X
VPC: vpc-0f13ebc96a105497c

default sg-0039c6133d65e96a1 X
VPC: vpc-0f13ebc96a105497c

Compare security group rules

Hide all selected

Advanced network configuration

Summary

Software Image (AMI)
Amazon Linux 2023 AMI 2023.8.2... [read more](#)
ami-043339ea831b48099

Virtual server type (instance type)

Firewall (security group)
3 security groups

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. Data transfer

Create launch template

- **User data (optional):** Script to auto-install Apache/web server on launch:

```
#!/bin/bash

yum update -y

yum install -y httpd

systemctl start httpd

systemctl enable httpd

echo "Hello from $(hostname -f)" > /var/www/html/index.html
```

The screenshot shows the 'Create launch template' wizard in the AWS EC2 console. In the 'User data - optional' section, a shell script is pasted:

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

A tooltip in the summary pane provides information about the free tier benefits.

3. Save the Launch Template.

The screenshot shows the success message: 'Successfully created MyApp-LT(lt-0b579cc5cb9d05c0a).'

Step 2: Create Auto Scaling Group

1. Go to **EC2 → Auto Scaling Groups → Create Auto Scaling Group**.
2. **Select launch template: MyApp-LT**
3. **Name: MyApp-ASG**

The screenshot shows the 'Choose launch template' step of the 'Create Auto Scaling group' wizard. The 'Name' field is set to 'MyApp-ASG'. The 'Launch template' dropdown shows 'MyApp-LT' selected. A note indicates that accounts created after May 31, 2023, can only use launch templates.

Step 3: Configure Network

1. **VPC:** Same as your instances/ALB
2. **Subnets:** Select at least 2 AZs for high availability

The screenshot shows the 'Network' configuration step. Under 'VPC', the 'Default' option is selected. Under 'Availability Zones and subnets', two subnets are chosen: 'eun1-az2 (eu-north-1b) | subnet-07a84a48af026ef07' and 'eun1-az1 (eu-north-1a) | subnet-0f44bdb97830fa751'. In the 'Availability Zone distribution - new' section, the 'Balanced best effort' option is selected, which is highlighted with a blue border.

Step 4: Attach to Target Group

1. Select “Attach to an existing load balancer”
2. Choose your **ALB Target Group** (MyApp-TG)

The screenshot shows the 'Integrate with other services - optional' step. The 'Attach to an existing load balancer' option is selected. A target group named 'MyApp-TG | HTTP' is chosen from the dropdown. Other options like 'No load balancer' and 'Attach to a new load balancer' are also shown.

Step 5: Configure Group Size

- **Desired capacity:** 2 → initially 2 instances
- **Minimum:** 2 → ensures at least 2 instances are always running

- **Maximum:** 5 → allows scaling up to 5 instances if needed

Configure group size and scaling - optional

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

Group size

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

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

Max desired capacity

5

Equal or less than desired capacity Equal or greater than desired capacity

Step 6: Add Scaling Policies

1. **Target tracking / simple scaling**
2. Example policy:
 - **Scale out:** CPU > 70% → add instance(s)

Automatic scaling - optional

Choose whether to use a target tracking policy

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

Target Tracking Policy

Metric type

Monitored metric that determines if resource utilization is too low or high. If using EC2 metrics, consider enabling detailed monitoring for better scaling performance.

Average CPU utilization

Target value

70

Instance warmup

300 seconds

Disable scale in to create only a scale-out policy

Step 7: Review and Create

- Review all settings → click **Create Auto Scaling Group**
- Auto Scaling will now automatically launch 2 instances initially, register them with ALB, and scale based on traffic.

The screenshot shows the AWS EC2 Auto Scaling Groups page. On the left, there's a navigation sidebar with sections like EC2, Instances, Images, and Elastic Block Store. The main area displays 'Auto Scaling groups (1) Info' with a table. The table has columns for Name, Launch template/configuration, Instances, Status, Desired capacity, Min, and Max. One row is shown for 'MyApp-ASG' with 'MyApp-LT | Version Default' in the launch template column, 2 instances, and desired capacity of 2. The status is marked with a dash. At the bottom of the table, it says '0 Auto Scaling groups selected'. The top right corner shows account information: Account ID: 9750-5025-7452, Europe (Stockholm), and Kartikey Gupta.

(v) Configure Path-Based Routing + Route 53

1. Go to ALB → **Listeners** → View/Edit rules.
2. Add rules:
 - o If path is /auth → forward to **Target Group: AuthService**.
 - o If path is /order → forward to **Target Group: OrderService**.
3. Ensure you created **separate EC2 groups** (Auth, Order) running respective services.
4. Save listener rules.

The screenshot shows the AWS Load Balancers page for 'MyAppALB'. It's navigating through EC2 > Load balances > MyAppALB > HTTP-80 listener > Add rule. The 'Conditions (1 value)' section contains a condition for 'Path = /auth'. The 'Actions' section shows a 'Forward to target groups' action selected. Below it, there's a 'Target group' table with one entry: 'MyApp-TG' (Target type: Instance, IPv4 | Target stickiness: Off). The 'HTTP' protocol is selected. The 'Weight' for this target group is set to 1, and the 'Percent' is 100%. The bottom right corner shows copyright information: © 2025, Amazon Web Services, Inc. or its affiliates.

aws Search [Alt+S] Europe (Stockholm) Account ID: 9750-5025-7452 Kartikey Gupta

EC2 > Load balances > MyAppALB > HTTP-80 listener > Add rule

Path = /order

Path condition value Case sensitive.

= /order Valid characters are a-z, A-Z, 0-9 and special characters. Path must be 1-128 characters.

+ Add OR condition value

Add condition You can add up to 4 more condition values for this rule.

Actions Requests matching all rule conditions route according to the rule actions.

Routing action

Forward to target groups Redirect to URL Return fixed response

Forward to target group Choose a target group and specify routing weight or [create target group](#).

Target group MyApp-TG Target type: Instance, IPv4 | Target stickiness: Off

HTTP Weight 1 Percent 100%

+ Add target group

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS Lambda function configuration page. The 'Handler' section is visible, containing the code path 'MyAppFunction.handler'. Other sections like 'Environment', 'Triggers', and 'Logs' are partially visible at the bottom.

aws Search [Alt+S] Europe (Stockholm) Account ID: 9750-5025-7452 Kartikey Gupta

EC2 > Load balances > MyAppALB > HTTP-80 listener

HTTP:80

Listener ARN arn:aws:elasticloadbalancing:eu-north-1:975050257452:listener/app/MyAppALB/5d4bb7ea2a4152cc/ea37fbdeaccecc0b

Rules Attributes Tags

Listener rules (3) Info

Traffic received by the listener is routed according to the default action and any additional rules. Rules are evaluated in priority order from the lowest value to the highest value.

Filter rules

| Priority | Name tag | Conditions (If) | Actions (Then) | ARN | Tags | Actions | |
|----------------|----------|--------------------------|--|------------------------------|--------|---------|--|
| 1 | - | Path = /order | Forward to target group MyApp-TG 1 (100%) Target group stickiness: Off | <input type="checkbox"/> ARN | 0 tags | | |
| 2 | - | Path = /auth | Forward to target group MyApp-TG 1 (100%) Target group stickiness: Off | <input type="checkbox"/> ARN | 0 tags | | |
| Last (default) | Default | If no other rule applies | Forward to target group MyApp-TG 1 (100%) Target group stickiness: Off | <input type="checkbox"/> ARN | 0 tags | | |

Rule limits Actions Add rule

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS Lambda function configuration page. The 'Handler' section is visible, containing the code path 'MyAppFunction.handler'. Other sections like 'Environment', 'Triggers', and 'Logs' are partially visible at the bottom.