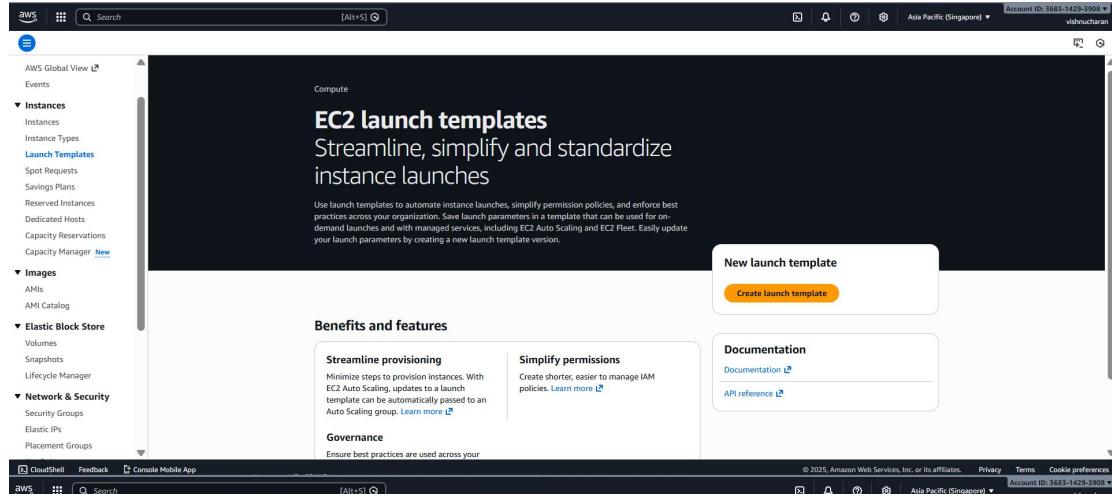


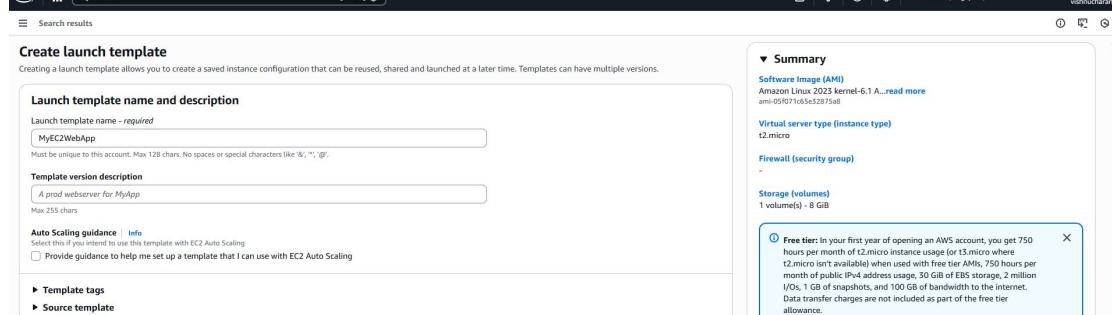
Scalable Web App with ALB & Auto Scaling – Use EC2, ALB, and Auto Scaling for high availability.

1. Create a Launch Template (Blueprint for EC2):-

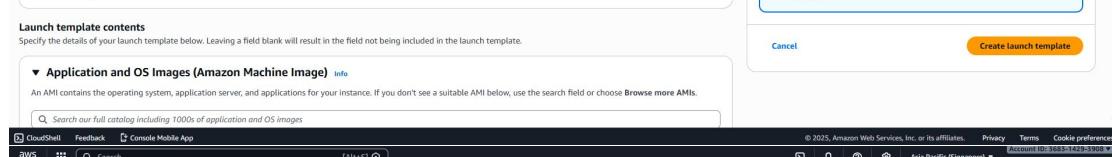
2. AWS Console → EC2 → Launch Templates → Create launch template



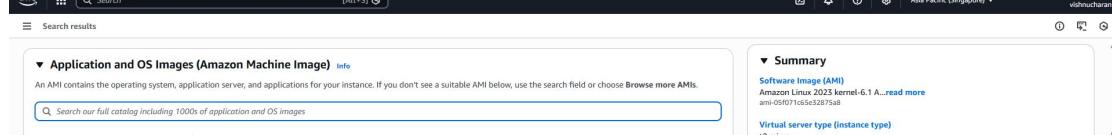
The screenshot shows the AWS Global View interface with the Compute section selected. Under Instances, 'Launch Templates' is highlighted. A modal window titled 'New launch template' is open, containing a single button: 'Create launch template'.



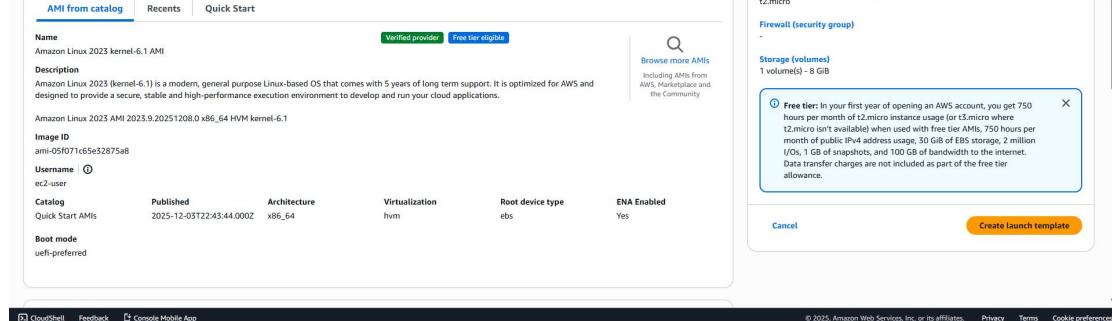
This step requires entering a launch template name ('MyEC2WebApp') and a template version description ('A prod webserver for MyApp'). It also includes sections for 'Auto Scaling guidance' (checkboxes for enabling EC2 Auto Scaling and providing guidance), 'Template tags', and 'Source template'.



This step shows the 'Application and OS Images (Amazon Machine Image)' section. It lists the selected AMI ('Amazon Linux 2023 kernel-6.1 AMI') and provides options to search for more AMIs or browse Marketplace.



The summary page displays the selected AMI ('Amazon Linux 2023 kernel-6.1 AMI'), virtual server type ('t2.micro'), storage ('1 volume(s) - 8 GB'), and a note about the free tier. It also shows the 'Create launch template' button.



This screenshot shows the 'Application and OS Images (Amazon Machine Image)' catalog for the selected AMI. It lists details like Name ('Amazon Linux 2023 kernel-6.1 AMI'), 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.'), and technical specifications (Catalog: Quick Start AMIs, Published: 2025-12-03T22:43:44.000Z, Architecture: x86_64, Virtualization: hvm, Root device type: ebs, ENA Enabled: Yes). It also includes links to 'Verified provider' and 'Free tier eligible'.

Instance type

t2.micro
 Family t2
 1 vCPU | 1 GiB Memory | Current generation: true | On-Demand Ubuntu Pro base pricing: 0.0164 USD per Hour
 On-Demand Linux base pricing: 0.0146 USD per Hour | On-Demand Windows base pricing: 0.0192 USD per Hour
 On-Demand RHEL base pricing: 0.0249 USD per Hour | On-Demand SUSE base pricing: 0.0146 USD per Hour

Additional costs apply for AMIs with pre-installed software

Key pair (login)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name
 mydebian13

Create new key pair

Network settings

Subnet

subnet-06db806dactedb4d7
 VPC: vpc-05073fe1fb42c0fb1b | Owner: 368314299008 | Availability Zone: ap-southeast-1b (apse1-az2)
 Zone type: Availability Zone | IP addresses available: 4091 | CIDR: 172.31.32.0/20

Create new subnet

Availability Zone

Don't include in launch template

Firewall (security group)

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

Select security groups

default sg-092317afe7b4943a1 X
 VPC: vpc-05073fe1fb42c0fb1b

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

Advanced network configuration

Storage (volumes)

EBS Volumes

Volume 1 (AMI Root) - 8 GiB, EBS, General purpose SSD (gp3), 3000 IOPS
 AMI Volumes are not included in the template unless modified

Metadata response hop limit

Don't include in launch template

Allow tags in metadata

Don't include in launch template

User data - optional

Upload a file with your user data or enter it in the field.

Choose file

```
#!/bin/bash
yum update -y
yum install httpd -y
systemctl start httpd
systemctl enable httpd
echo '<h1>Welcome to Scalable Web application</h1>' > /var/www/html/index.html
```

User data has already been base64 encoded

Summary

Software Image (AMI)
 Amazon Linux 2023 kernel-6.1 A... [read more](#)
 ami-05071c65e32875ab

Virtual server type (instance type)
 t2.micro

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 GiB of snapshots, and 100 GiB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

Create launch template

Cancel

Network settings

Subnet

subnet-04bf478a0e7c6dd7
 VPC: vpc-05073fe1fb42c0fb1b | Owner: 368314299008 | Availability Zone: ap-southeast-1a (apse1-az1)
 Zone type: Availability Zone | IP addresses available: 4091 | CIDR: 172.31.16.0/20

Create new subnet

Availability Zone

Don't include in launch template

Firewall (security group)

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

Select security groups

default sg-092317afe7b4943a1 X
 VPC: vpc-05073fe1fb42c0fb1b

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

Advanced network configuration

Storage (volumes)

EBS Volumes

Volume 1 (AMI Root) - 8 GiB, EBS, General purpose SSD (gp3), 3000 IOPS
 AMI Volumes are not included in the template unless modified

Metadata response hop limit

Don't include in launch template

Allow tags in metadata

Don't include in launch template

User data - optional

Upload a file with your user data or enter it in the field.

Choose file

```
#!/bin/bash
yum update -y
yum install httpd -y
systemctl start httpd
systemctl enable httpd
echo '<h1>Welcome to Scalable Web application</h1>' > /var/www/html/index.html
```

User data has already been base64 encoded

Summary

Software Image (AMI)
 Amazon Linux 2023 kernel-6.1 A... [read more](#)
 ami-05071c65e32875ab

Virtual server type (instance type)
 t2.micro

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 GiB of snapshots, and 100 GiB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

Create launch template

Cancel

Storage (volumes)

EBS Volumes

Volume 1 (AMI Root) - 8 GiB, EBS, General purpose SSD (gp3), 3000 IOPS
 AMI Volumes are not included in the template unless modified

Metadata response hop limit

Don't include in launch template

Allow tags in metadata

Don't include in launch template

User data - optional

Upload a file with your user data or enter it in the field.

Choose file

```
#!/bin/bash
yum update -y
yum install httpd -y
systemctl start httpd
systemctl enable httpd
echo '<h1>Welcome to Scalable Web application</h1>' > /var/www/html/index.html
```

User data has already been base64 encoded

Summary

Software Image (AMI)
 Amazon Linux 2023 kernel-6.1 A... [read more](#)
 ami-05071c65e32875ab

Virtual server type (instance type)
 t2.micro

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 GiB of snapshots, and 100 GiB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

Create launch template

Cancel

Launch Templates (1/1)

Launch Template ID	Launch Template Name	Default Version	Latest Version	Create Time	Created By	Managed	Operator	Status
lt-0d3f20338e5800f42	MyEC2WebApp	1	1	2025-12-22T13:49:35.000Z	arn:aws:iam::368314293908:root	false	-	-

MyEC2WebApp (lt-0d3f20338e5800f42)

Launch template details

Launch template ID	Launch template name	Default version	Owner
lt-0d3f20338e5800f42	MyEC2WebApp	1	arn:aws:iam::368314293908:root

Launch template version details

Version	Description	Date created	Created by
1.0 (Default)	-	2025-12-22T13:49:35.000Z	arn:aws:iam::368314293908:root

3. Create Target Group (Health & Routing)

Target groups

Name	ARN	Port	Protocol	Target type	Load balancer	VPC ID
No target groups						

Create target group

Step 1: Create target group

A target group can be made up of one or more targets. Your load balancer routes requests to the targets in a target group and performs health checks on the targets.

Settings - immutable

Choose a target type and the load balancer and listener will route traffic to your target. These settings can't be modified after target group creation.

Target type

Indicate what resource type you want to target. Only the selected resource type can be registered to this target group.

- Instances**: Supports load balancing to instances in a VPC. Integrate with Auto Scaling Groups or ECS services for automatic management. Suitable for: ALB, NLB, GVLB.
- IP addresses**: Supports load balancing to IP addresses and on-premises resources. Suitable for: ALB, NLB, GVLB.
- Lambda function**: Supports load balancing to a single Lambda function. ALB required as traffic source. Suitable for: ALB.
- Application Load Balancer**: Allows use of static IP addresses and PrivateLink with an Application Load Balancer. NLB required as traffic source. Suitable for: NLB.

Target group name

Name must be unique per Region per AWS account.
Mywebapptargetgroup
Accepts a-z, A-Z, 0-9, and hyphen (-). Can't begin or end with hyphen. 1-32 total characters; Count: 19/32

Protocol

Protocol for communication between the load balancer and targets.
HTTP

Port

Port number where targets receive traffic. Can be overridden for individual targets during registration.
80
1-65535

EC2 > Target groups > Create target group

Step 1
 Step 1: Create target group
 Step 2 - recommended
Register targets
 Step 3
 Review and create

Register targets - recommended
 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 (0)

Ports for the selected instances
 Ports to route traffic to the selected instances:
 80
 T-45555 (separate multiple ports with comma)
 Include as pending below

Review targets

Targets (0)

Step 1: Target group details

Name: Mywebapptargetgroup	Target type: Instance	Protocol : Port: HTTP: 80	Protocol version: HTTP1
VPC: vpc-05273fe1b42c061b	IP address type: IPv4	Target control port: 1	

Health check details

Health check protocol: HTTP	Health check path: /	Health check port: traffic-port	Interval: 30 seconds
Timeout: 5 seconds	Healthy threshold: 2	Unhealthy threshold: 2	Success codes: 200

Step 2: Register targets

Targets (0)

Create target group

EC2 > Target groups > Create target group

Step 1
 Step 1: Create target group
 Step 2 - recommended
Register targets
 Step 3
 Review and create

Review and create
 Review your target group configuration before creating

Step 1: Target group details

Name: Mywebapptargetgroup	Target type: Instance	Protocol : Port: HTTP: 80	Protocol version: HTTP1
VPC: vpc-05273fe1b42c061b	IP address type: IPv4	Target control port: 1	

Health check details

Health check protocol: HTTP	Health check path: /	Health check port: traffic-port	Interval: 30 seconds
Timeout: 5 seconds	Healthy threshold: 2	Unhealthy threshold: 2	Success codes: 200

Step 2: Register targets

Targets (0)

Create target group

EC2 > Target groups > Mywebapptargetgroup

Instances
 Instances
 Instance Types
 User-defined Roles
 Spot Requests
 Savings Plans
 Reserved Instances
 Dedicated Hosts
 Capacity Reservations
 Capacity Manager

Images
 AMIs
 AMI Catalog

Elastic Block Store
 Volumes
 Snapshots
 Lifecycle Manager

Network & Security
 Security Groups
 Elastic IPs
 Placement Groups
 Key Pairs
 Network Interfaces

Load Balancing
 Load Balancers
 Target Groups
 Trust Stores

Auto Scaling
 Auto Scaling Groups
 Settings

Mywebapptargetgroup

Details
 arn:aws:elb:ap-southeast-1:3681429908:targetgroup/Mywebapptargetgroup/B02285401626242

Target type: Instance	Protocol : Port: HTTP: 80	Protocol version: HTTP1
IP address type: IPv4	Load balancer: <input checked="" type="radio"/> None associated	VPC: vpc-05273fe1b42c061b
0 Total targets	0 Healthy	0 Unhealthy
	0 Anomalous	
		0 Unused
		0 Initial
		0 Draining

Targets | Monitoring | Health checks | Attributes | Tags

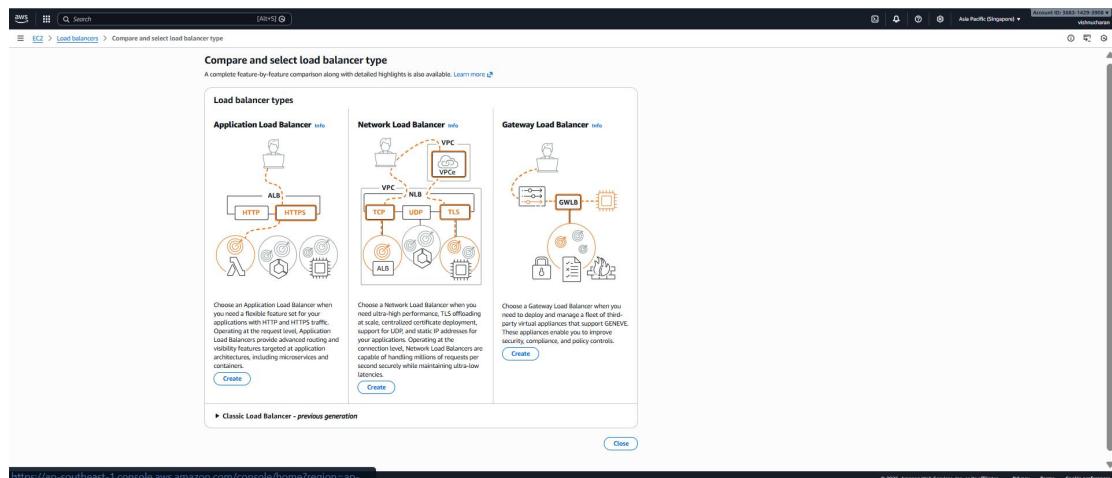
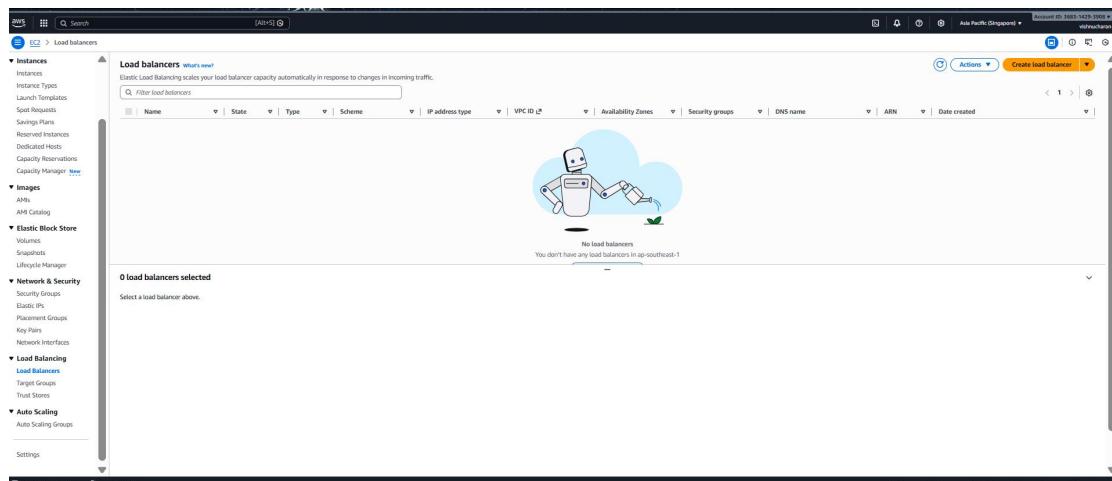
Registered targets (0) info
 Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.

Filter targets

Instance ID	Name	Port	Zone	Health status	Health status details	Administrative override
-------------	------	------	------	---------------	-----------------------	-------------------------

No registered targets
 Register targets

4. Create Application Load Balancer (ALB)



► 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.

mywebapploadbalancer

A maximum of 52 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme

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 public access.

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.

Dualstack

Includes IPv4 and IPv6 addresses.

Dualstack without public IPv4

Includes a public IPv6 address, and private IPv4 and IPv6 addresses. Compatible with **internet-facing** load balancers only.

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-05073f61b42c06fb

172.31.0.0/16

(default)



Create VPC

IP pools

Info

You can optionally choose to configure an IPAM pool as the preferred source for your load balancer 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, IPv6 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.

ap-southeast-1a (apsec1-a21)

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-0491478a0e7c8d07

IPv4 subnet CIDR: 172.31.1.0/20

✓ ap-southeast-1b (apsec1-a22)

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-056bb064ac8e0d47

IPv4 subnet CIDR: 172.31.32.0/20

□ ap-southeast-1c (apsec1-a23)

Security groups

Info

A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

Security groups

Select up to 5 security groups

Launch-w2d-1

sg-055c23c84a59e40c VPC: vpc-05073f61b42c06fb



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

HTTP

Port

80

1-65535

Remove

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

Mywebapptargetgroup

Target type: Instance, IPv4 | Target stickiness: Off

HTTP



Weight

1

Percent

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.

□ Turn on target group stickiness

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

You can add up to 49 more listeners.

Screenshot of the AWS EC2 Load Balancer console showing the configuration of a new load balancer named "mywebapploadbalancer".

Details:

- Load balancer type: Application
- Status: Provisioning
- Scheme: Internet-facing
- Health zone: ZLUM5919CMLES
- VPC: vpc-050773fa11b43c061b
- Availability Zones: subnets-06d006decbe0d7, ap-southeast-1b (ap-northeast-1), subnets-07a078a7f0a442, ap-southeast-1a (ap-northeast-1)
- Load balancer IP address type: IPv4
- Date created: December 22, 2025, 20:21 (UTC+05:30)
- Load balancer ARN: arnaws:elasticloadbalancing:ap-southeast-1:368114291908:loadbalance/app/mywebapploadbalancer/bc77d931907547
- DNS name: mywebapploadbalancer-202465462.ap-southeast-1.elb.amazonaws.com (A Record)

Listeners and rules (1) Info

ProtocolPort	Default actions	Rules	ARN	Security policy	Default SSL/TLS certificate	mTLS	Trust store
HTTP:80	Forward to target group: mywebapptargetgroup (100%) Target group stickiness: Off	1 rule	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable

5. Create Auto Scaling Group

Screenshot of the AWS Auto Scaling Groups console.

Amazon EC2 Auto Scaling helps maintain the availability of your applications.

An Auto Scaling group is a collection of Amazon EC2 instances that are treated as a logical unit. You configure settings for a group and its instances as well as define the group's minimum, maximum, and desired capacity. Setting different minimum and maximum values forms the bounds of the group, which allows the group to scale up or down as needed. When launching instances in the group, you can specify what percentage of your capacity should be fulfilled by On-Demand instances, and what percentage with Spot Instances, to save up to 90% on EC2 costs. Amazon EC2 Auto Scaling lets you provision and balance capacity across Availability Zones to optimize availability. It also provides lifecycle hooks, instance health checks, and scheduled scaling to automate capacity management.

Create Auto Scaling group

Get started with EC2 Auto Scaling by creating an Auto Scaling group.

Pricing

Amazon EC2 Auto Scaling charges under additional fees beyond the hourly fee for Amazon EC2, CloudWatch (for scaling policies), and the other AWS resources that you use. Visit the pricing page of each service to learn more.

Getting started

What is Amazon EC2 Auto Scaling? Info
Getting started with Amazon EC2 Auto Scaling Info
Set up a scaled and load-balanced application Info
FAQ Info

Screenshot of the "Create Auto Scaling group" wizard.

Step 1: Choose launch template or configuration

Choose 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: MyWebApp

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.

MyEC2WebApp

Description

AMI ID: ami-050773fa11b43c061b
Key pair name: mydbakan1

Security group

Security group IDs: sg-09287a0fb0442a1

Additional details

Storage (volume):
Date created: Mon Dec 22 2025 19:19:35 GMT+0530 (India Standard Time)

Next Step

Step 1

- Choose launch template or configuration

Step 2

Choose instance launch options

- Step 3 - optional
- Step 4 - optional
- Configure group size and scaling
- Step 5 - optional
- Add notifications
- Step 6 - optional
- Add tags
- Step 7 - optional
- Review

Choose instance launch options Info

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

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 attributes.

Launch template MyEC2LaunchTemplate Override launch template

Version Latest

Description

Instance type t2.micro

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-00973f96 (1a2c201b) 172.31.0.0/16 Default

Create a VPC

Availability Zones and subnets Info

Select the availability zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets

apn1-1a2c201b-1a1 [subnet-04b478a0e70ddc07] 172.31.1.0/24 Default

apn1-1a2c201b-1b [subnet-06d80606edcbed47] 172.31.32.0/24 Default

Create a subnet

Availability Zone distribution new Info

Auto Scaling automatically launches instances across Availability Zones. If launch failures occur in a zone, select a strategy.

Balanced best effort If launcher fails in one Availability Zone, Auto Scaling will attempt to launch in another healthy Availability Zone.

Subnet only If launches fail in one Availability Zone, Auto Scaling will continue to attempt to launch in the unhealthy Availability Zone to preserve balanced distribution.

Integrate with other services - optional Info

Use a load balancer to distribute network traffic across multiple servers. Enable service-to-service communications with VPC Lattice. Shift resources away from impaired Availability Zones with zonal shift. You can also customize health check replacements and monitoring.

Load balancing Info

Use this section to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

Select Load balancing options

No load balancer This option allows your Auto Scaling group will not be framed 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 balancer to attach

Choose from your load balancer target groups This option allows you to attach Applications, Networks, or Gateway Load Balancers.

Select target groups

Mywebappsgroup1 (HTTP Application Load Balancer: mywebapploadbalancer)

VPC Lattice integration options Info

To improve networking capabilities and scalability, integrate your Auto Scaling group with VPC Lattice. VPC Lattice facilitates communications between AWS services and helps you connect and manage your application across compute services in AWS.

Select VPC Lattice service to attach

No VPC Lattice service This option allows you to manage your Auto Scaling group's network access and connectivity with other services.

Attach to VPC Lattice Requesting insights associated with specified VPC Lattice target groups will be routed to your Auto Scaling group.

Create new VPC Lattice service

Application Recovery Controller (ARC) zonal shift - new Info

During an Availability Zone impairment, target instance launches towards other healthy Availability Zones.

Configure group size and scaling - optional Info

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

Group size Info

Set the number of instances in your 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(GB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances)

Desired capacity Specify your group size.

2

Scaling Info

You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits Set the limits in which your desired capacity can be increased or decreased.

Min desired capacity 1

Max desired capacity 4

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

Automatic scaling - optional Info

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 You can set up a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

Instance maintenance policy Info

Control your Auto Scaling group's availability during instance replacement events. This includes health checks, instance refreshes, maximum instance lifetime features and events that happen automatically to keep your group balanced, called rebalancing events.

Choose a replacement behavior depending on your availability requirements

Health behavior **No policy** If a previous event, new instances will

Prioritize availability **Launch before terminating** When launching new instances for them to be

Control costs **Terminate and launch** Terminating old instances before launching new ones at the same time

Flexibility **Custom behavior** Define your own logic for the instance and

Screenshot of the AWS Auto Scaling Group creation process:

- Step 1: Choose launch template or configuration**
 - Choose launch template or configuration
 - Choose instance launch options
 - Integrate with other services
 - Configure group size and scaling
 - Add notifications
 - Add tags
 - Review
- Step 2: Choose instance launch options**
 - Network**: VPC: MyEC2WebAppsg
 - Availability Zones and subnets**: Two subnets: aps1-a1 (ap-southeast-1a) and aps1-a2 (ap-southeast-1a).
 - Instance type requirements**: Balanced best effort.
- Step 3: Integrate with other services**
 - Load balancing**: None selected.

Auto Scaling groups (1/1)

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones	Creation time
Mywebappsg	MyEC2WebAppsg Version Latest	0	Updating capacity	2	1	4	2 Availability zones	Mon Dec 22 2025 20:30:25 GMT+0530 (India Standard Time)

Auto Scaling group: Mywebappsg

Details | Integrations | Automatic scaling | Instance management | Instance refresh | Activity | Monitoring | Tags - moved

Mywebappsg Capacity overview

Desired capacity	Scaling limits	Desired capacity type	Status
2	1 - 4	Units (number of instances)	Updating capacity

Date created: Mon Dec 22 2025 20:30:25 GMT+0530 (India Standard Time)

Launch template

Launch template	AMI ID	Instance type	Owner
MyEC2WebAppsg	ami-05f071c65c32875a8	t2.micro	arn:aws:iam::368314293908:root
Version	Security groups	Security group IDs	Create time
Latest	-	sg-052011a7b79445a1	Mon Dec 22 2025 19:19:35 GMT+0530 (India Standard Time)
Description	Storage (volumes)	Key pair name	Request Spot Instances
-	-	mydbian13	No

[View details in the launch template console](#)

Now my EC2 instances are created automatically because of Auto Scaling Group.

Screenshot of the AWS EC2 Instances page:

Instances (2)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 IP	Elastic IP	IPv6 IP	Monitoring	Security group name	Key name
i-0f558ad0fffae1116e	i-0f558ad0fffae1116e	Running	t2.micro	Initiating	View alarms	ap-southeast-1b	ec2-54-251-227-25.ap...	54.251.227.25	-	-	disabled	default	mydbian13
i-017051b0485b5e673	i-017051b0485b5e673	Running	t2.micro	Initiating	View alarms	ap-southeast-1a	ec2-13-213-5-155.ap...	13.213.5.155	-	-	disabled	default	mydbian13

Select an instance



Welcome to Scalable Web App