

Part 1: EC2 with ELB and ASG

Objective: Learn how to create a scalable and highly available web application environment using Amazon EC2 instances, ELB, and ASG.

Approach:

1. **Launch EC2 Instances:** Start by launching two or more EC2 instances. These instances will run a simple web application (e.g., a "Hello World" page or any basic web service).
2. **Configure Load Balancer:** Set up an Elastic Load Balancer (ELB) to distribute incoming web traffic across your EC2 instances. This step ensures high availability and fault tolerance.
3. **Set Up Auto Scaling Group (ASG):** Create an ASG that uses the launched EC2 instances. Configure ASG policies to automatically scale the number of instances up or down based on criteria like CPU usage or network traffic.
4. **Test Your Setup:** Simulate traffic to test the scaling policies and the load balancer. Observe how ASG adds or removes instances and how ELB distributes traffic.
5. **Verify Website Functionality:** Ensure that the website hosted on EC2 instances remains accessible and functional during scaling operations.

Goal: By the end of this lab, students will have a hands-on understanding of setting up a load-balanced and auto-scaled web application using AWS services.

Steps Involved:

- 1) **Launch EC2 instance**
Provide the name of instance

[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 the simple steps below.

Name and tags [Info](#)

Name

Sonu's server

[Add additional tag](#)

▼ Application and OS Images (Amazon Machine Image) [Info](#)


An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q Search our full catalog including 1000s of application and OS images


Recents

Quick Start


Amazon Linux




macOS




Ubuntu




Windows




Red Hat



SUSE Linux




Browse more
Including AMIs
AWS, Marketplace
the Commu

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

Free tier elig

ami-0440d3b780d96b29d (64-bit (x86), uefi-preferred) / ami-0f93c02efd1974b8b (64-bit (Arm), uefi)

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

- 2) Select t2.micro instance type. Add new keypair or from an existing key pair.

▼ Instance type [Info](#) | [Get advice](#)

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true

On-Demand Windows base pricing: 0.0162 USD per Hour

On-Demand SUSE base pricing: 0.0116 USD per Hour

On-Demand RHEL base pricing: 0.0716 USD per Hour

On-Demand Linux base pricing: 0.0116 USD per Hour

Free tier eligible

☒ All generations

[Compare instance types](#)

Additional costs apply for AMIs with pre-installed software

▼ Key pair (login) [Info](#)

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 - *required*

keypair

[Create new key pair](#)

3) Network Settings

Security Group launch-wizard-14 is created allowing SSH, HTTPS and HTTP traffic from anywhere.

▼ Network settings [Info](#) [Edit](#)

Network [Info](#)

vpc-0b7272288b62dddec

Subnet [Info](#)

No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)

Enable

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

We'll create a new security group called 'launch-wizard-4' with the following rules:

☒ Allow SSH traffic from

Helps you connect to your instance

Anywhere
0.0.0.0/0

☒ Allow HTTPS traffic from the internet

To set up an endpoint, for example when creating a web server

☒ Allow HTTP traffic from the internet

To set up an endpoint, for example when creating a web server

⚠ Rules with 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.

×

4) Two instances with same configuration are to be created.

▼ Summary

Number of instances

[Info](#)

When launching more than 1 instance, consider [EC2 Auto Scaling](#)

Software Image (AMI)

Amazon Linux 2023 AMI 2023.3.2...[read more](#)
ami-0440d3b780d96b29d

Virtual server type (instance type)



t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB


 **Free tier:** In your first year includes  750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Launch instance

5) At Advanced Detail section below script is attached and rest configuration are left as it is.

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

 Choose file


```
#!/bin/bash
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
echo "<h1> Hello from $(hostname -f)<a/h1>" > /var/www/html/index.html
S
```

☐ User data has already been base64 encoded

6) Click on “Launch Instance”

7) Successful instance creation

EC2 > Instances > Launch an instance

 **Success**
Successfully initiated launch of instances (i-0b1b6b8e33ac8c423, i-004a05a5dfa56a27c)

► Launch log

Next Steps

8) Update the created security group inbound rule as the ec2 instance is not working while trying to run from the Public IPv4 address of the created EC2 instance.


Edit inbound rules [info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules [info](#)

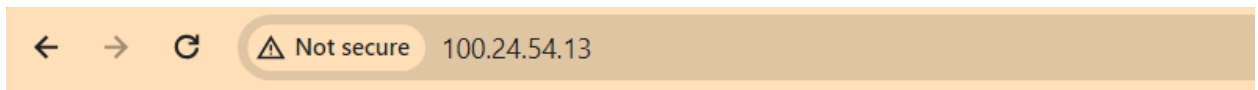
Security group rule ID	Type info	Protocol info	Port range info	Source info	Description - optional info	
sg-09e42abda8a9868dc	HTTPS	TCP	.443	Anywhere-IPv4	Q, 0.0.0.0/0 X	Delete
sg-0111cebdea06a012d	SSH	TCP	22	Anywhere-IPv4	Q, 0.0.0.0/0 X	Delete
sg-0ca823145f0b1ec8d	HTTP	TCP	80	Anywhere-IPv4	Q, 0.0.0.0/0 X	Delete

Add rule

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

Cancel Preview changes Save rules

9) Both instances public IPv4 addresses worked in web browser successfully.



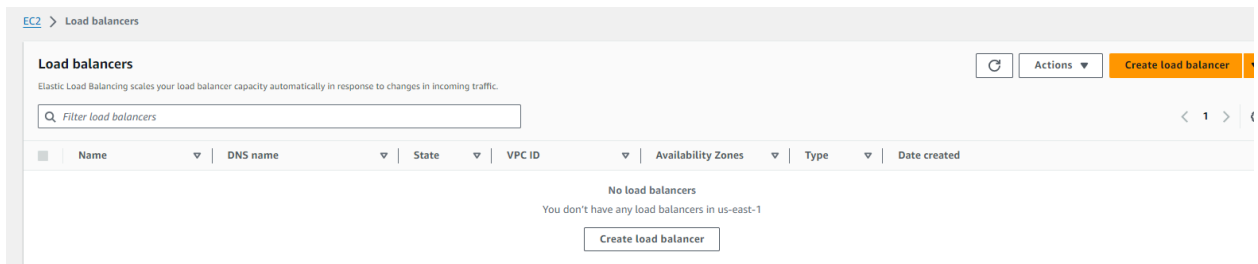
Hello from ip-172-31-89-214.ec2.internal



Hello from ip-172-31-91-82.ec2.internal

Configuring Load Balancer

10) Creating Load Balancer



11) We select Application Load Balancer and provide the required configuration.

[EC2](#) > [Load balancers](#) > Create Application Load Balancer

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. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply. If a rule is 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.

myloadbalancer

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

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

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

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

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

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

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

12) Select all Network Mapping options.

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 or you can [create a new VPC](#). Only VPCs with an internet gateway are enabled for selection. The selected VPC must have at least one public subnet. To confirm the VPC for your targets, view your [target groups](#).

vpc-b727238b:cidr-block:172.31.0.0/16

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 available for selection.

☒ **us-east-1a (use1-az6)**
Subnet: subnet-020aceaf4a3e074e
IPv4 address: Assigned by AWS

☒ **us-east-1b (use1-az1)**
Subnet: subnet-0d2e8b94ac8ac629a
IPv4 address: Assigned by AWS

☒ **us-east-1c (use1-az2)**
Subnet: subnet-0c8a67d85572795f7
IPv4 address: Assigned by AWS

☒ **us-east-1d (use1-az4)**
Subnet: subnet-030e7321e75aff44e
IPv4 address: Assigned by AWS

☒ **us-east-1e (use1-az5)**
Subnet: subnet-032651b5942c27e5c
IPv4 address: Assigned by AWS

☒ **us-east-1f (use1-az5)**
Subnet: subnet-032651b5942c27e5c
IPv4 address: Assigned by AWS

13) Application Load Balancer Security Groups

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

Q |

☒ launch-wizard-4
sg-05ba46fec929a603c VPC: vpc-0b7272288b62dddec

☐ launch-wizard-2
sg-09f1ef525ad2c21a9 VPC: vpc-0b7272288b62dddec

☐ launch-wizard-1
sg-0eae42968ba66701a VPC: vpc-0b7272288b62dddec

☐ launch-wizard-3
sg-0a5fe7bf18c57ed79 VPC: vpc-0b7272288b62dddec

☒ default
sg-012cf31c7f3a994e7 VPC: vpc-0b7272288b62dddec

↻

↕

etermine how the load balanc

14) Creating a Target Group

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

15) Provide the group details name for group -1 and for group-2.

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 can't be changed after the target group is created.

Choose a target type

☒ Instances

- Supports load balancing to instances within a specific VPC.
- Facilitates the use of [Amazon EC2 Auto Scaling](#) to manage and scale your EC2 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

myTGs

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

Protocol : Port

Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation

HTTP 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-0b7272288b62dddec

IPv4: 172.31.0.0/16

16) Configuration Review and Target Group Creation.

Successful Target Group Creation

EC2 > Target groups

Target groups (4) Info

Filter target groups

	Name	ARN	Port	Protocol	Target type	Load balancer	VPC ID
<input type="checkbox"/>	myTG2	arn:aws:elasticloadbalanci...	80	HTTP	Instance	None associated	vpc-0b7272288b62dddec
<input type="checkbox"/>	myTGs	arn:aws:elasticloadbalanci...	80	HTTP	Instance	None associated	vpc-0b7272288b62dddec

17) Now, the load balancer configuration can be continued. In Listeners and routing section, a recently created target group is selected.

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

myTGs

Target type: Instance, IPv4

HTTP ▼

⌂

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

18) Summary and Load Balancer Creation. Click on Create load balancer.

Review

Review the load balancer configurations and make changes if needed. After you finish reviewing the configurations, choose Create load balancer.

Summary

Review and confirm your configurations. [Estimate cost](#)

Basic configuration [Edit](#)

Security groups [Edit](#)

Network mapping [Edit](#)

Listeners and routing [Edit](#)

myloadbalancer

- Internet-facing
- IPv4

- launch-wizard-4
- sg-05ba46fec929a603c
- default
- sg-012cf31c7f3a994e7

VPC [vpc-0b7272288b62dddec](#)

- us-east-1a
- subnet-020aceaf44a3e074e
- us-east-1b
- subnet-0d2e8b94acbac629a
- us-east-1c
- subnet-0c8a67d85372793f7
- us-east-1d
- subnet-030e7321e75aff44e
- us-east-1e
- subnet-032651b5942c27e5c
- us-east-1f
- subnet-00b9d4caa6246652d

- HTTP:80 defaults to [myTGs](#)

Service integrations [Edit](#)

Tags [Edit](#)

AWS WAF: None

AWS Global Accelerator: None

None

Attributes

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

Creation workflow and status

► Server-side tasks and status

After completing and submitting the above steps, all server-side tasks and their statuses become available for monitoring.

Cancel

Create load balancer

19) Successful creation of Application Load Balancer.

Successfully created load balancer: myloadbalancer
It might take a few minutes for your load balancer to fully set up and route traffic. Targets will also take a few minutes to complete the registration process and pass initial health checks.

EC2 > Load balancers > myloadbalancer

myloadbalancer

▼ Details

Load balancer type
Application

Scheme
Internet-facing

Status
Provisioning

Hosted zone
Z355XDOTRQ7X7K

VPC
vpc-0b717228b6124dddc

Availability Zones
subnet-0c8a67d8537279377 us-east-1c (use1-az2)
subnet-000f64c4a65246652d1 us-east-1f (use1-az5)
subnet-032651b5942c27e5c us-east-1e (use1-az3)
subnet-043e8b94a8a8a6279a us-east-1b (use1-az1)
subnet-030e7521e75aff4de us-east-1d (use1-az4)
subnet-020ac0af64a3e074e us-east-1a (use1-az6)

IPv4
Data
Febr

Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:304064102356:loadbalancer/app/myloadbalancer/91c28870bba54e87

DNS name
myloadbalancer-324725830.us-east-1.elb.amazonaws.com (A Record)

Listeners and rules

Network mapping

Security

Monitoring

Integrations

Attributes

Tags

Listeners and rules (1) info

A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

Filter listeners

Protocol:Port

Default action

Rules

ARN

Security policy

Default SSL/TLS certificate

mTLS

Trust store

Tags

HTTP:80

Forward to target group

myTGs 1 (100%)

Group-level stickiness: Off

1 rule

ARN

Not applicable

Not applicable

Not applicable

Not applicable

0 tags

20) Registering targets

EC2 > Target groups > myTGs > Register targets

Register targets

Select instances, specify ports, and add the instances to the list of pending targets. Repeat to add additional combinations of instances and ports selections, click Register pending targets.

Available instances (2/2)

Filter instances

<input checked="" type="checkbox"/>	Instance ID	Name	State	Security groups
<input checked="" type="checkbox"/>	i-0b1b6b8e33ac8c423	Sonu's server	Running	launch-wizard-4
<input checked="" type="checkbox"/>	i-004a05a5dfa56a22c	Sonu's server	Running	launch-wizard-4

2 selected

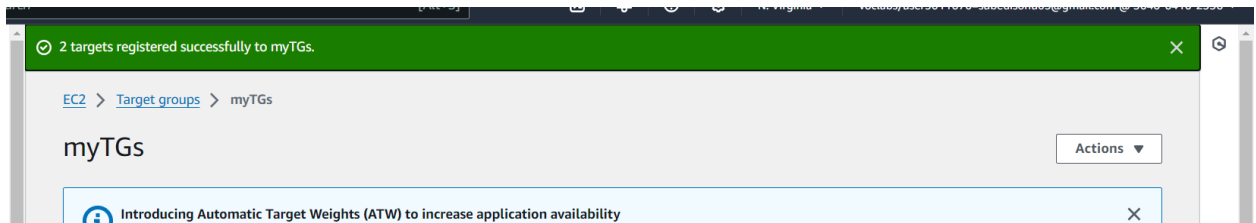
Ports for the selected instances
Ports for routing traffic to the selected instances.

80

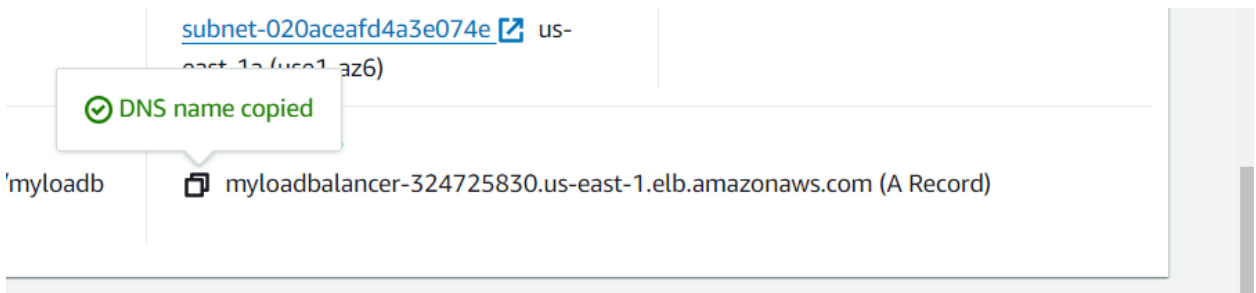
1-65535 (separate multiple ports with commas)

Include as pending below

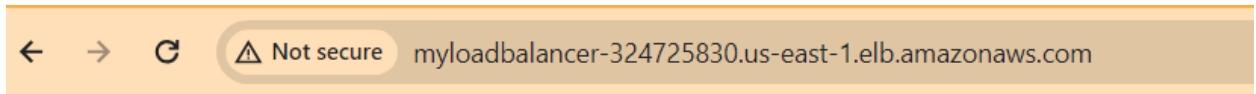
21) The targets has been successfully registered.



21) After target group has been registered,copy the DNS name of load balancer and access in web browser.

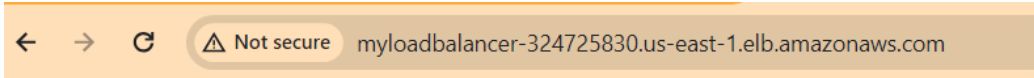


IP from one of the instance,



Hello from ip-172-31-91-82.ec2.internal

Ip from next instance



Hello from ip-172-31-89-214.ec2.internal

22) Creating an Auto Scaling Group

Create Auto Scaling group

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

Create Auto Scaling group

23) In the launch template section, Choose launch template, “Create a launch template” is selected and new template is created.

Choose launch template [info](#)

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.

Name

Auto Scaling group name
Enter a name to identify the group.

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

Launch template [info](#)

i For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

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.

[Create a launch template](#)

Cancel Next

[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 a later time. Templates can have multiple versions.

Launch template name and description

Launch template name - *required*

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

Template version description

Max 255 chars

Auto Scaling guidance | [Info](#)
Select this if you intend to use this template with EC2 Auto Scaling

☒ Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

► Template tags

► Source template

Application and OS Image

24) AMI in current use is selected

Launch template contents

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

▼ Application and OS Images (Amazon Machine Image) - required [Info](#)

An AMI is a template that contains the software configuration (operating system, application software, etc.) required to launch your instance. Search or Browse for AMIs if you don't see what you need below.

Recents

Quick Start

☐ Recently launched

☒ Currently in use

25) For instance type, t2.micro is selected and for key pair, existing key pair is selected

▼ Instance type [Info](#) | [Get advice](#) [Adv](#)

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true

On-Demand Windows base pricing: 0.0162 USD per Hour

On-Demand SUSE base pricing: 0.0116 USD per Hour

On-Demand RHEL base pricing: 0.0716 USD per Hour

On-Demand Linux base pricing: 0.0116 USD per Hour

Free tier eligible

☒ All generations

[Compare instance types](#)


[Additional costs apply for AMIs with pre-installed software](#)

▼ Key pair (login) [Info](#)

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

keypair ▼

 [Create new key pair](#)


26) Network Settings

Existing Security Groups are selected and other configurations are kept the same as it.

▼ Network settings [Info](#)

Subnet [Info](#)

Don't include in launch template ▼

 [Create new subnet](#)

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

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Select existing security group


☐ Create security group

Security groups [Info](#)

Select security groups ▼

launch-wizard-4 sg-05ba46fec929a603c ✕

VPC: vpc-0b7272288b62dddec

 [Compare security group rules](#)

► Advanced network configuration

27) Bash script is added as per the previous instance configurations

```
#!/bin/bash
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
echo "<h1> Hello from $(hostname -f)<a/h1>" > /var/www/html/index.html
```

User data - optional | [Info](#)


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


 **Choose file**

```
#!/bin/bash
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
echo "<h1> Hello from $(hostname -f)<a/h1>" > /var/www/html/index.html
```

28) Launch template successfully created.

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

 **Success**
Successfully created [mytemplate\(lt-0ef3d8e4a8bda3c78\)](#).

 Actions log

Next Steps

29) Now, the configuration of auto scaling is continued. Here a newly created launch template is selected.

Launch template [Info](#)

i For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

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.

mytemplate ▼

↺

[Create a launch template](#)

Version

Default (1) ▼

↺

[Create a launch template version](#)

30) VPC along with all Availability Zones and subnets are selected .

Network [Info](#)

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling select 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-0b7272288b62dddec
172.31.0.0/16 Default ▼

↺

[Create a VPC](#)

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets ▼

↺

us-east-1a | subnet-020aceafd4a3e074e ✕
172.31.32.0/20 Default

us-east-1b | subnet-0d2e8b94ac8ac629a ✕
172.31.0.0/20 Default

us-east-1c | subnet-0c8a67d85372793f7 ✕
172.31.80.0/20 Default

us-east-1d | subnet-030e7321e75aff44e ✕
172.31.16.0/20 Default

us-east-1e | subnet-032651b5942c27e5c ✕
172.31.48.0/20 Default

us-east-1f | subnet-00bfd4caa6246652d ✕
172.31.64.0/20 Default

31) Configuring Advanced Options

Existing load balancer is chosen and it's target group is selected. Others advanced configuration is kept as default.

Load balancing [Info](#)

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

☐ No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

☒ Attach to an existing load balancer
Choose from your existing load balancers.

☐ Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to an existing load balancer
Select the load balancers that you want to attach to your Auto Scaling group.

☒ Choose from your load balancer target groups
This option allows you to attach Application, Network, or Gateway Load Balancers.

☐ Choose from Classic Load Balancers

Existing load balancer target groups
Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

Select target groups ▼

myTGs | HTTP
Application Load Balancer: myloadbalancer ✕

32) Group Size and Scaling Configuration is done. Here desired capacity of Group size is set as 2. In scaling, min desired capacity is set as 2 and max desired capacity as 4.

Group size [Info](#)

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 [Info](#)

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

Equal or less than desired capacity

Max desired capacity

4

Equal or greater than desired capacity

Automatic scaling - optional

Choose whether to use a target tracking policy [Info](#)

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.

33) No Instance Maintenance Policy is selected. No notifications is added and no tags are added.

Instance maintenance policy - new [Info](#)

Control your Auto Scaling group's availability during instance replacement events. instance lifetime features and events that happen automatically to keep your group

Control availability and cost during replacement events

An instance maintenance policy determines how much availability Auto Scaling replaces instances. It also establishes guardrails can be added or removed when replacing instances.

Choose a replacement behavior depending on your availability requirements

Mixed behavior

☒ **No policy**

For rebalancing events, new instances will launch before terminating others. For all other events, instances terminate and launch at the same time.

Prioritize availability

☐ **Launch before terminating**

Launch new instances and wait for them to be ready before terminating others. This allows you to go above your desired capacity by a given percentage and may temporarily increase costs.

Control availability

☐ **Terminate before launching**

Terminate instances before launching new ones. This ensures that the number of instances never exceeds the desired capacity during replacement events.

Instance scale-in protection

Add notifications - *optional* [Info](#)

Send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group.

Add notification

Cancel



Skip to review

Previous

Next

Add tags - *optional* [Info](#)

Add tags to help you search, filter, and track your Auto Scaling group across AWS. You can also choose to automatically add these tags to instances when they are launched.

 You can optionally choose to add tags to instances (and their attached EBS volumes) by specifying tags in your launch template. We recommend caution, however, because the tag values for instances from your launch template will be overridden if there are any duplicate keys specified for the Auto Scaling group. 

Tags (0)

Add tag

50 remaining

Cancel

Previous

Next

34) Now auto scaling group is configured and created.

Tag new instances

No tags

Cancel

Previous

Create Auto Scaling group

35) Now we can notice two Instances are in pending state automatically. In the activity tab of auto scaling, we can observe the activity history of the EC2 instances.

Instances (4) Info									
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>					Any state ▾				
<input type="checkbox"/>	Name ↗ ▾	Instance ID	Instance state ▾	Instance type ▾	Status check	Alarm status	Availability Zone ▾	Public IPv4 DNS ▾	Public IPv4
<input type="checkbox"/>		i-0cf87fc1c7b3958d8	Running 🔍 🔍	t2.micro	🕒 Initializing	View alarms +	us-east-1e	ec2-100-25-153-90.co...	100.25.153.1
<input type="checkbox"/>		i-008ca31035e2742a6	Running 🔍 🔍	t2.micro	🕒 Initializing	View alarms +	us-east-1d	ec2-54-172-93-105.co...	54.172.93.1

Activity history (2)

Status ▾	Description ▾	Cause ▾	Start time ▾
Successful	Launching a new EC2 instance: i-0cf87fc1c7b3958d8	At 2024-02-24T08:06:54Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2024-02-24T08:06:56Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2024 February 24, 01:51:59 PM +05:45
Successful	Launching a new EC2 instance: i-008ca31035e2742a6	At 2024-02-24T08:06:54Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2024-02-24T08:06:56Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2024 February 24, 01:51:58 PM +05:45

36) When the DNS of the load balancer is refreshed even multiple times, it works.

← → ↻ ⚠ Not secure myloadbalancer-324725830.us-east-1.elb.amazonaws.com

Hello from ip-172-31-21-138.ec2.internal

← → ↻ ⚠ Not secure myloadbalancer-324725830.us-east-1.elb.amazonaws.com

Hello from ip-172-31-91-82.ec2.internal

37) Testing of Auto-Scaling

For that, we terminate the instance.

We notice that the other instance is running automatically.

Instances (2/4) Info

Refresh

Connect

Instance state ▼

Actions ▼

Launch

Find Instance by attribute or tag (case-sensitive)

Any state ▼

<input type="checkbox"/>	Name <div>✎</div> ▼	Instance ID	Instance state ▼	Instance type ▼	Status check	Alarm status	Availability Zone ▼	Public IPv4 DNS
<input type="checkbox"/>		i-0cf87fc1c7b3958d8	<div>Running</div> <div>🔍 🔍</div>	t2.micro	<div>⌚</div> Initializing	<div>View alarms</div> <div>+</div>	us-east-1e	ec2-100-25-153-90...
<input checked="" type="checkbox"/>		i-008ca31035e2742a6	<div>Terminated</div> <div>🔍 🔍</div>	t2.micro	<div>⌚</div> Initializing	<div>View alarms</div> <div>+</div>	us-east-1d	–
<input type="checkbox"/>	Sonu's server	i-0b1b6b8e33ac8c423	<div>Running</div> <div>🔍 🔍</div>	t2.micro	<div>✅</div> 2/2 checks passed	<div>View alarms</div> <div>+</div>	us-east-1c	ec2-174-129-140-11
<input checked="" type="checkbox"/>	Sonu's server	i-004a05a5dfa56a22c	<div>Terminated</div> <div>🔍 🔍</div>	t2.micro	<div>✅</div> 2/2 checks passed	<div>View alarms</div> <div>+</div>	us-east-1c	–

38) Now connect to the instance via SSH.

```
PS C:\Users\subed> cd Downloads
PS C:\Users\subed\Downloads> ssh -i "keypair.pem" ec2-user@100.25.153.90
The authenticity of host '100.25.153.90 (100.25.153.90)' can't be established.
ED25519 key fingerprint is SHA256:/Tw0v iw+C761FAp/Ub9GkoPEl5AAhMtwgx6aeVgbHkc.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '100.25.153.90' (ED25519) to the list of known hosts.

      #_
    ~\ _ #####_      Amazon Linux 2023
  ~ ~ \ _ ##### \
  ~ ~ \ _ ### |
  ~ ~ \ _ #/ _ _ _ https://aws.amazon.com/linux/amazon-linux-2023
    ~ ~ V ~ ' ' ->
      ~ ~ ~
      ~ ~ . . _ /
      _ / _ /
      _ / m / '

[ec2-user@ip-172-31-50-88 ~]$
```

Stress test

39) Stress test is installed.

```
~/m/
[ec2-user@ip-172-31-50-88 ~]$ sudo yum install stress -y
Last metadata expiration check: 0:06:45 ago on Sat Feb 24 08:07:51 2024.
Dependencies resolved.
===== Pa
===== Package Information =====
Version      Repository      Size
-----
stress       x86_64          1.0.4-28.amzn2023.0.2      amazonlinux      37 k
===== Inst

Transaction Summary
===== Inst

Total download size: 37 k
Installed size: 78 k
Downloading Packages:
stress-1.0.4-28.amzn2023.0.2.x86_64.rpm                                420 kB/s | 37 kB    00:00
-----Tot
251 kB/s | 37 kB    00:00

Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      :                                1/1
  Installing     : stress-1.0.4-28.amzn2023.0.2.x86_64      1/1
  Running scriptlet: stress-1.0.4-28.amzn2023.0.2.x86_64      1/1
```

40) Now the stress test is performed.

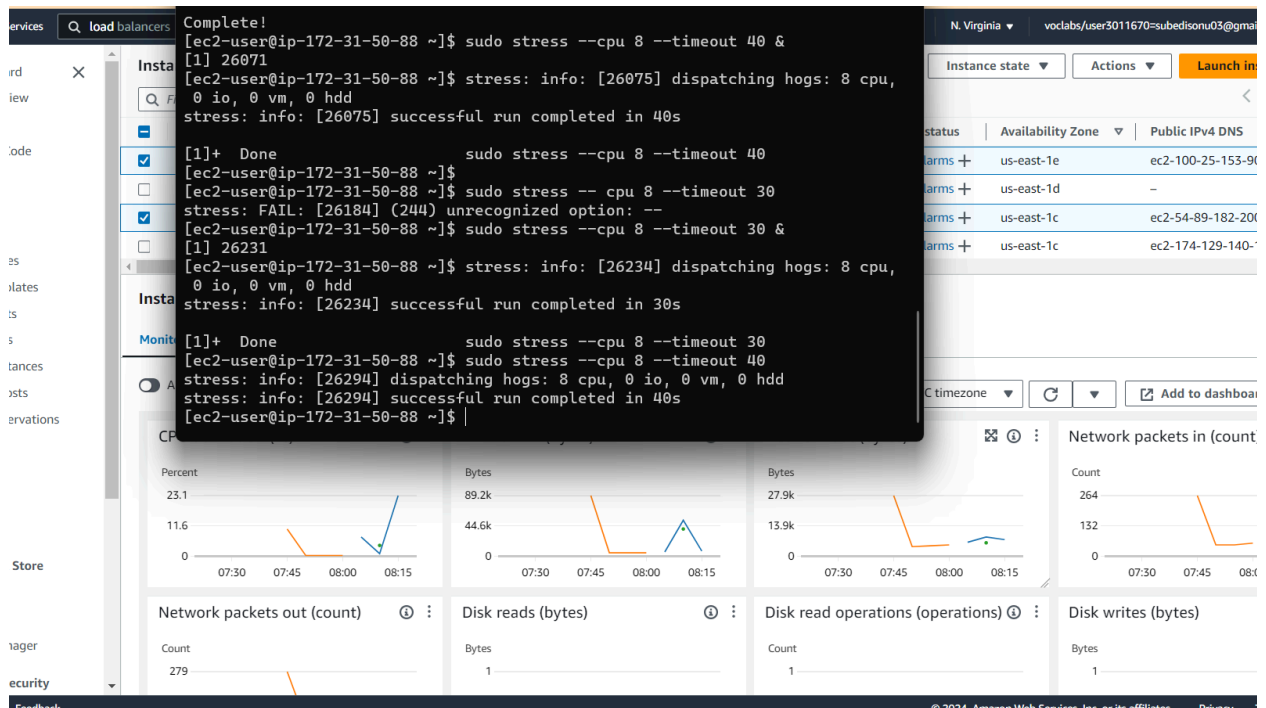
```
Installed.
stress-1.0.4-28.amzn2023.0.2.x86_64

Complete!
[ec2-user@ip-172-31-50-88 ~]$ sudo stress --cpu 8 --timeout 40 &
[1] 26071
[ec2-user@ip-172-31-50-88 ~]$ stress: info: [26075] dispatching hogs: 8 cpu, 0 io, 0 vm, 0 hdd
|
```

41) Completion of stress test

```
stress-1.0.4-28.amzn2023.0.2.x86_64

Complete!
[ec2-user@ip-172-31-50-88 ~]$ sudo stress --cpu 8 --timeout 40 &
[1] 26071
[ec2-user@ip-172-31-50-88 ~]$ stress: info: [26075] dispatching hogs: 8 cpu, 0 io, 0 vm, 0 hdd
stress: info: [26075] successful run completed in 40s
```

42) Monitoring the instances

The CPU utilization graph of two instances is shown as. It's in increasing order.

