

Simulating Manual Horizontal and Vertical Scaling in AWS

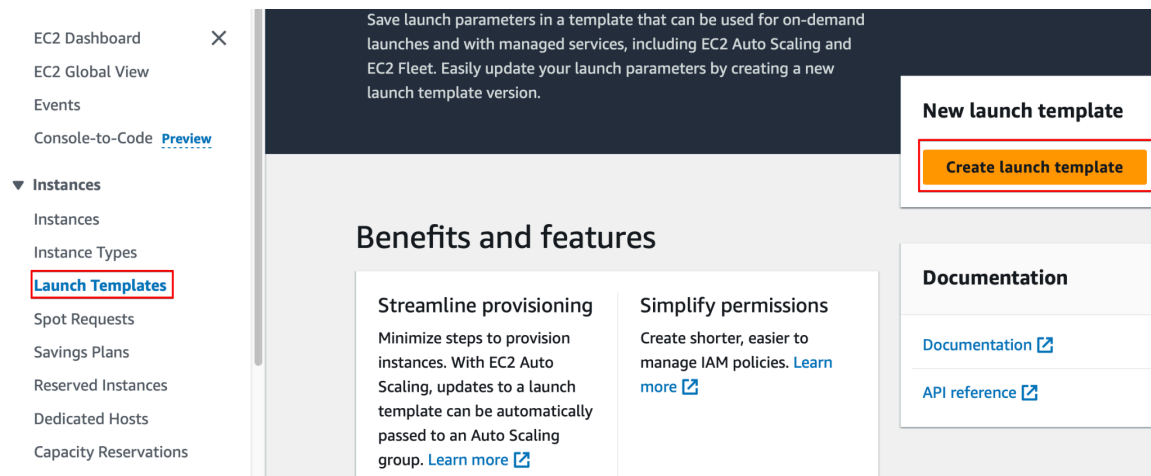
This runbook provides detailed steps to simulate horizontal and vertical scaling in AWS. We will create an EC2 launch template, an Auto Scaling Group (ASG), and configure **manual scaling**. The process simulates scale out and scale in based on hypothetical load, as well as simulating **vertical scaling** by modifying instance types.

1. Create a Launch Template of an Instance

This step involves creating a launch template which defines the configuration details of the EC2 instances.

The template includes details such as the Amazon Machine Image (AMI), instance type, key pair, security groups, and other parameters required to launch an instance.

1. Open the Amazon EC2 console.
2. In the left navigation pane, choose 'Launch Templates'.



3. Choose 'Create launch template'.
4. Provide a name and description and tag(optional) for the launch template.

Launch template name and description

Launch template name - *required*

jjtech-ASG-demo

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

Template version description

version1 template for ASG demo

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

Key	Value	
<input type="text" value="Name"/>	<input type="text" value="demo-launch-template"/>	<input type="button" value="Remove tag"/>

You can add up to 49 more tags.

▼ Source template

5. In the 'Launch template contents' section, specify the AMI, instance type, key pair, security group, and other configurations.

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 server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Quick Start

Amazon Linux

macOS

Ubuntu

Windows

Red Hat

SUSE Linux

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

ami-01b799c439fd5516a (64-bit (x86), uefi-preferred) / ami-Oe1ef59154d415994 (64-bit (Arm), uefi)

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

Free tier eligible ▼

▼ Instance type

Info | Get advice

Instance type

t2.micro

Free tier eligible

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

[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 key pair when you launch the instance.

Key pair name

ASG-launchTemplate

▼ Network settings

Info

Subnet

Info

Don't include in launch template

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 group name - required

asg-template-sg

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Valid characters: a-z, A-Z, 0-9, spaces, and . _ - / () # , @ [] + = & ; { } ! \$ *

Description - required

Info

Sg for instances launch using the ASG

VPC

Info

vpc-d0c9adad

172.31.0.0/16

(default)

Inbound Security Group Rules

No security group rules are currently included in this template. Add a new rule to include inbound traffic.

6. Leave other settings (e.g storage) as default.

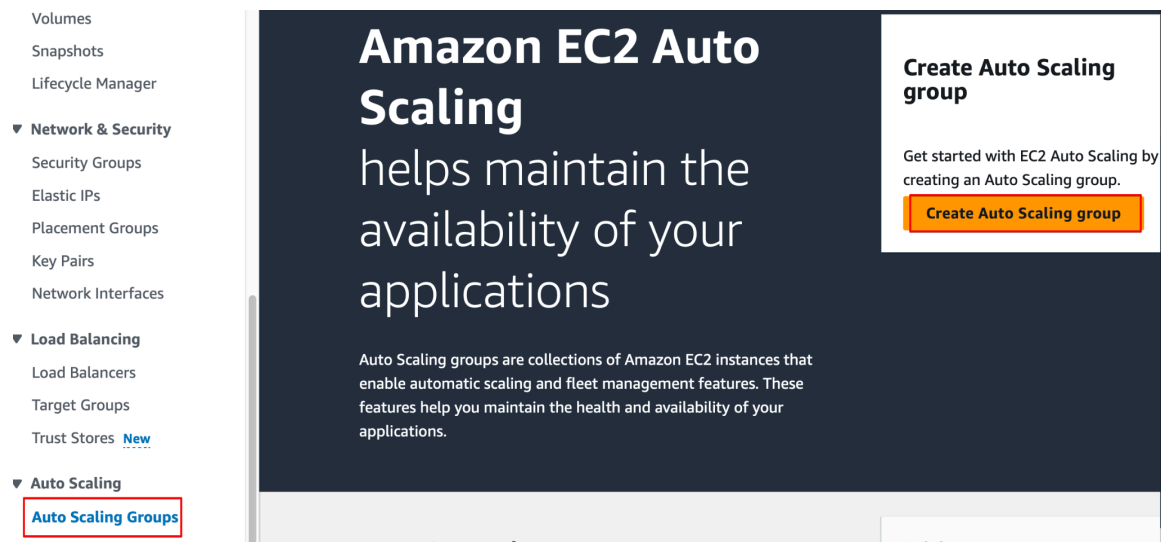
Review the settings and choose '**Create launch template**'.

2. Create an Auto Scaling Group

This step involves creating an Auto Scaling Group (ASG) using the launch template. The ASG ensures that the number of instances scales automatically based on the defined policies and rules.

1. Open the Amazon EC2 console.

2. In the left navigation pane, choose 'Auto Scaling Groups'.



3. Choose 'Create Auto Scaling group'.

4. Select the launch template created in the previous step and click on **Next**

The screenshot shows the 'Choose launch template or configuration' step in the AWS Management Console. On the left is a sidebar with steps 1 through 7. Step 1 is 'Choose launch template or configuration'. The main area has a title 'Choose launch template or configuration' with an 'Info' link. Below the title is a description: 'Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group. If you currently use launch configurations, you might consider migrating to launch templates.' There are two main sections. The first is 'Name', which contains a text input field for 'Auto Scaling group name' with the value 'demo-ASG' highlighted by a red box. The second is 'Launch template', which has a search bar with 'jjtech-ASG-demo' entered and highlighted by a red box. Below the search bar is a list of launch templates, with 'jjtech-ASG-demo' selected and highlighted by a red box. There is also a 'Version' dropdown set to 'Default (1)'.

Step 1
Choose launch template or configuration

Step 2
Choose instance launch options

Step 3 - optional
Configure advanced options

Step 4 - optional
Configure group size and scaling

Step 5 - optional
Add notifications

Step 6 - optional
Add tags

Step 7
Review

Choose launch template or configuration [Info](#)

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group. If you currently use launch configurations, you might consider migrating to launch templates.

Name

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

demo-ASG

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

Launch template [Info](#) [Switch to launch configuration](#)

Search launch templates

jjtech-ASG-demo

jjtech-ASG-demo

Create a launch template

Version

Default (1)

In the *Choose instance launch options* section,

5. For instance type requirements, leave as default

6. Configure the VPC and subnets for the Auto Scaling group. Select subnets in different AZs in order to enable the ASG to launch instances in that AZ

The screenshot shows the 'Network' section of the AWS Management Console. It has a title 'Network' with an 'Info' link. Below the title is a description: '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.' There are two main sections. The first is 'VPC', which contains a dropdown menu for 'vpc-d0c9adad' with the value '172.31.0.0/16 Default' and a refresh button. The second is 'Availability Zones and subnets', which contains a dropdown menu for 'Select Availability Zones and subnets' and a list of three subnets: 'us-east-1a | subnet-e0dc8ac1', 'us-east-1b | subnet-61b9a22c', and 'us-east-1c | subnet-562f7b09'. The first two subnets are highlighted by a red box. There is also a 'Create a subnet' link at the bottom.

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-d0c9adad
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-e0dc8ac1
172.31.80.0/20 Default

us-east-1b | subnet-61b9a22c
172.31.16.0/20 Default

us-east-1c | subnet-562f7b09
172.31.32.0/20 Default

Create a subnet

Cancel Skip to review Previous **Next**

In the *Configure Advanced Options* section, leave everything as default

In *Configure the group size and scaling* section,

7. Set the minimum capacity to **2**, desired capacity to **4**, and maximum capacity to **6**

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 group.

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 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 group configured with a set of instance attributes.

Units (number of instances) ▼

Desired capacity
Specify your group size.

4

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	Max desired capacity
2	5

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

8. Leave other settings as default, scroll down and click on **Skip to review**

9. Review the settings and choose '**Create Auto Scaling group**'.

3. Manual Scaling

Manual scaling allows you to manually adjust the desired capacity of the Auto Scaling group. This step demonstrates how to change the desired capacity to simulate scaling events.

1. Open the Amazon EC2 console.
2. In the left navigation pane, choose 'Auto Scaling Groups'.
3. Select the Auto Scaling group created earlier.
4. Choose 'Edit'.

EC2 > Auto Scaling groups

Auto Scaling groups (1/1) Info

Launch configurations Launch templates Actions Create Auto Scaling group

Search your Auto Scaling groups

<input checked="" type="checkbox"/>	Name	Launch template/configuration	Instances	Status
<input checked="" type="checkbox"/>	ASG-demo	jitech-ASG-demo Version Default	4	-

Auto Scaling group: ASG-demo

Details | Activity | Automatic scaling | Instance management | Monitoring | Instance refresh

5. Change the desired capacity to the desired value (e.g., 2).

EC2 > Auto Scaling groups > ASG-demo

Edit ASG-demo Info

Group size Info

Specify the size of the Auto Scaling group by changing the desired capacity. You can also specify the maximum capacity.

Desired capacity type

Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only available for instance types configured with a set of instance attributes.

Units (number of instances)

Desired capacity

Specify your group size.

2

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

5

Equal or greater than desired capacity

6. Scroll down and click on **Update** to save the changes. Observe the scaling activity.

Vertical Scaling Simulation

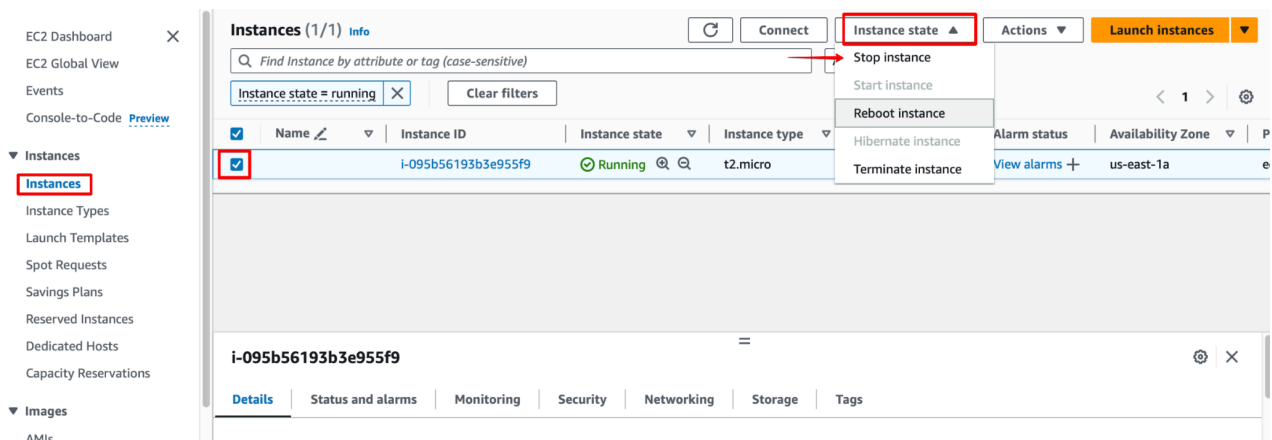
Vertical scaling involves changing the instance type to a larger or smaller instance to increase or decrease the resources (CPU, memory) available to the application. This simulation will demonstrate how to modify the instance type directly on an existing EC2 instance.

Prerequisites:

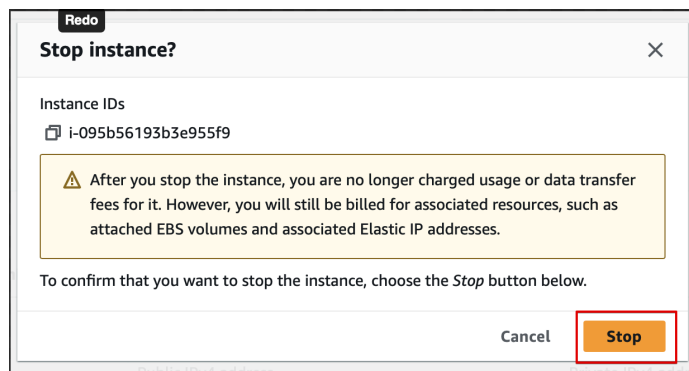
- Existing EC2 instance running.

Steps to Simulate Vertical Scaling:

1. **Open the Amazon EC2 Console**
 - Navigate to the [Amazon EC2 console](#).
2. **Stop the Instance**
 - In the left navigation pane, choose 'Instances'.
 - Select the instance you want to scale vertically.
 - Choose 'Instance State' > 'Stop Instance'.

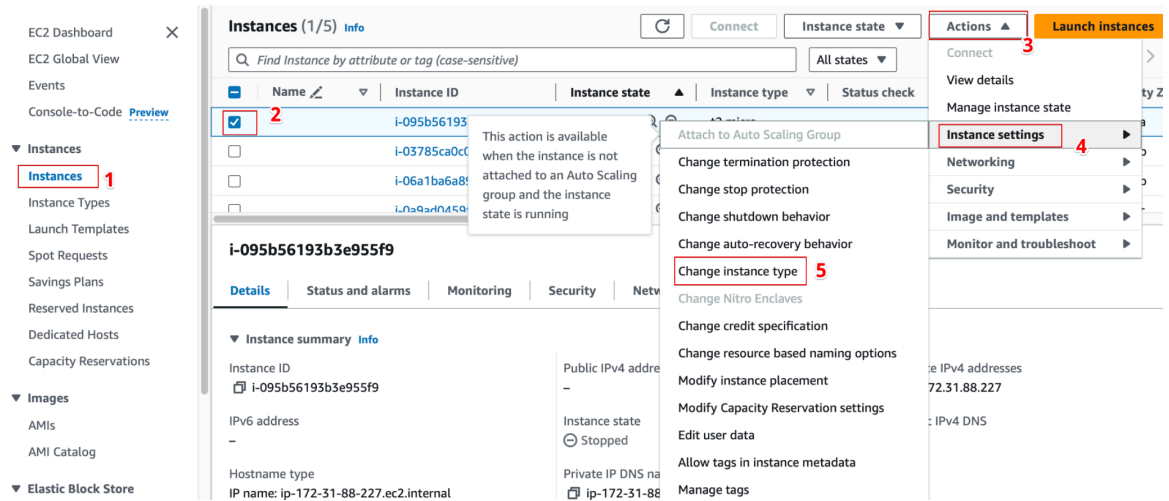


- Confirm the action to stop the instance.



3. Change the Instance Type

- After the instance is stopped, select the instance again.
- Choose 'Actions' > 'Instance Settings' > 'Change Instance Type'.



- In the 'Change Instance Type' dialog box, select the new instance type you want to use (e.g., change from **t2.micro** to **t2.medium**).

Change instance type [Info](#) | [Get advice](#)

You can change the instance type only if the current instance type and the instance type that you want are compatible.

Instance ID
i-095b56193b3e955f9

Current instance type
t2.micro

New instance type

☐ EBS-optimized
EBS-optimized is not supported for this instance type

▼ Instance type comparison

Attribute	t2.micro	t2.medium
On-Demand Linux pricing	0.0116 USD per Hour	0.0464 USD per Hour
On-Demand Windows pricing	0.0162 USD per Hour	0.0644 USD per Hour
vCPUs	1 (1 core)	2 (2 core)
Memory (MiB)	1024	4096
Storage (GB)	–	–
Supported root device types	ebs	ebs

- Choose 'Apply'.

4. Start the Instance

- With the instance still selected, choose 'Instance State' > 'Start Instance'.
- Confirm the action to start the instance.

The screenshot shows the AWS Management Console 'Instances' page. On the left is a navigation sidebar with links like 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Console-to-Code', and 'Instances'. The main content area shows a table of instances. One instance, 'i-095b56193b3e955f9', is selected and its state is 'Stopped'. A dropdown menu for 'Instance state' is open, showing options: 'Stop instance', 'Start instance' (highlighted with a red box), 'Reboot instance', 'Hibernate instance', and 'Terminate instance'. Below the table, the details for the selected instance are shown, including tabs for 'Details', 'Status and alarms', 'Monitoring', 'Security', 'Networking', 'Storage', and 'Tags'. The 'Instance summary' section is visible, showing fields like 'Instance ID', 'Public IPv4 address', and 'Private IPv4 address'.

- Wait for the instance to enter the 'running' state.

5. Verify the Changes

- Select the instance to view its details.
- Verify that the 'Instance Type' has been updated to the new instance type.

The screenshot shows the AWS Management Console 'Instances' page after the instance has started. The instance 'i-095b56193b3e955f9' is now in the 'Running' state. The 'Instance state' dropdown menu is open, showing options: 'Stop instance', 'Start instance', 'Reboot instance', 'Hibernate instance', and 'Terminate instance'. A red box highlights the 'Launch instances' button in the top right corner. Below the table, the details for the selected instance are shown, including tabs for 'Details', 'Status and alarms', 'Monitoring', 'Security', 'Networking', 'Storage', and 'Tags'. The 'Instance summary' section is visible, showing fields like 'Instance ID', 'Public IPv4 address', 'Private IPv4 addresses', 'Instance state', 'Public IPv4 DNS', 'Private IP DNS name (IPv4 only)', 'Instance type', and 'Elastic IP addresses'.

- Ensure that the instance is running and operating as expected.

