

Create a CloudFormation stack

1. Open the CloudFormation Sample Templates page with your region

2. Scroll down, choose Single EC2 Instance with local MySQL database and download the template

AWS CloudFormation Sample Templates - US West (N. California) region

AWS CloudFormation gives developers and systems administrators an easy way to create a collection of related AWS resources and provision them in an orderly and predictable fashion. The AWS CloudFormation samples package contains a collection of templates that illustrate various usage cases. Stacks can be created from the templates via the [AWS Management Console](#), the AWS CLI, the AWS CloudFormation APIs, or the AWS Toolkits. You can use the templates as-is or you can use them as a starting point for creating your own templates. All the sample templates can be downloaded [here](#).

AWS CloudFormation is available in all AWS regions. For more details click [here](#).

Microsoft Windows Server samples

Template URL	Description	Launch in US West (N. California) Region
Windows_Single_Server_SharePoint_Foundation.template	SharePoint® Foundation 2010 running on Microsoft Windows Server® 2008 R2	Launch Stack
Windows_Single_Server_Active_Directory.template	Create a single server installation of Active Directory running on Microsoft Windows Server® 2008 R2.	Launch Stack
Windows_Roles_And_Features.template	Create a single server specifying server roles running on Microsoft Windows Server® 2008 R2.	Launch Stack

* Microsoft, Windows Server, and SharePoint are trademarks of the Microsoft group of companies.

Open Source Applications

The following sample templates show you how to provision existing open source application stack. Each of the samples shows how to deploy and install the application at launch time using the AWS CloudFormation application bootstrap support described [here](#).

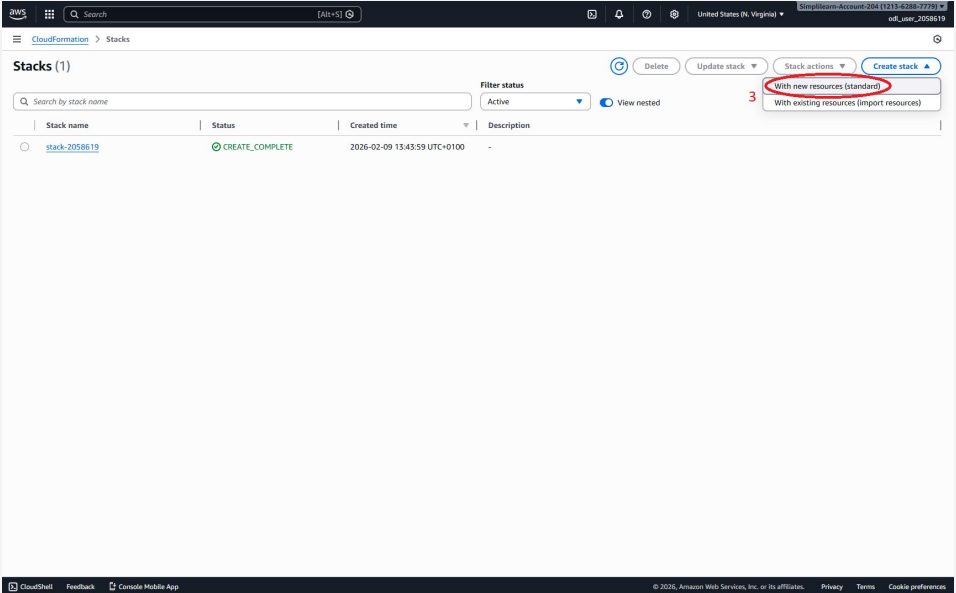
Description	Template URL	Launch in US West (N. California) Region
Drupal is an open source content management platform powering millions of websites and applications. For more details click here .	Single EC2 Instance with local MySQL database	Launch Stack
	Single EC2 Instance web server with Amazon RDS database instance	Launch Stack

Application Framework Examples

The following sample templates show you how to provision application frameworks by setting up *Hello World* applications:

Description	Template URL	Launch in US West (N. California) Region
A simple LAMP stack running a PHP "Hello World" application.	Single EC2 Instance with local MySQL database	Launch Stack
	Single EC2 Instance web server with Amazon RDS database instance	Launch Stack
	Highly Available Web Server with Multi-AZ Amazon RDS Instance	Launch Stack
A simple Ruby on Rails "Hello World" application.	Single EC2 Instance with local MySQL database	Launch Stack
	Single EC2 Instance web server with Amazon RDS database instance	Launch Stack
	Highly Available Web Server with Multi-AZ Amazon RDS Instance	Launch Stack

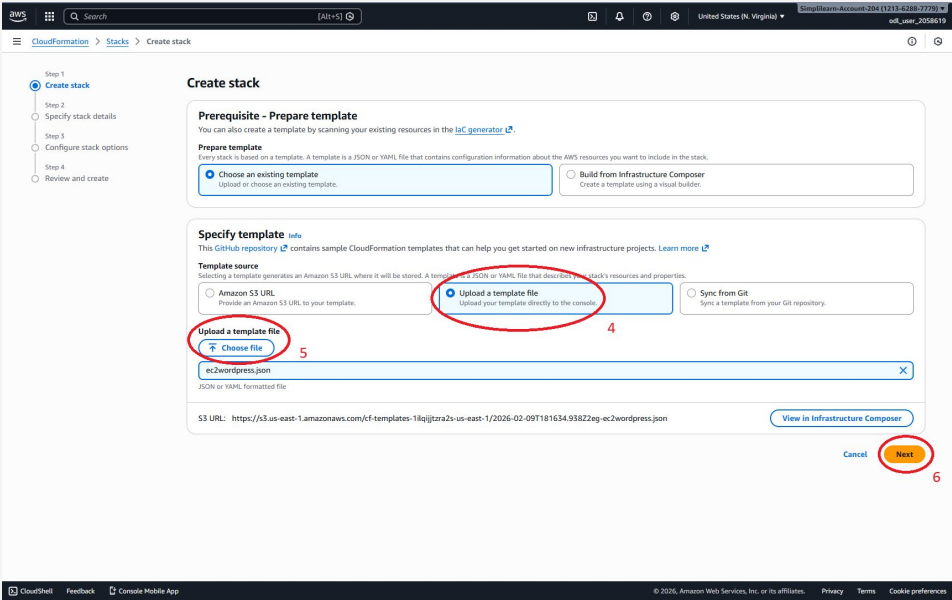
3. Open the CloudFormation page and click on Create stack with new resources



4. Select Upload a template file

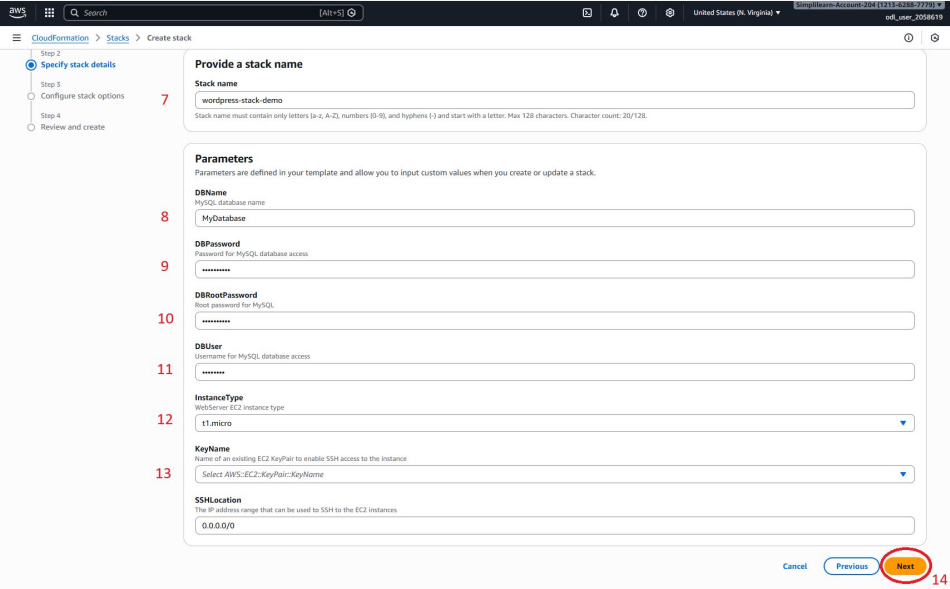
5. Choose your template file

6. Click on Next



Provide the following:

- 7. Stack name
- 8. Database name
- 9. Database password
- 10. Database root password
- 11. Database user
- 12. Instance type
- 13. Select KeyName
- 14. Click on next

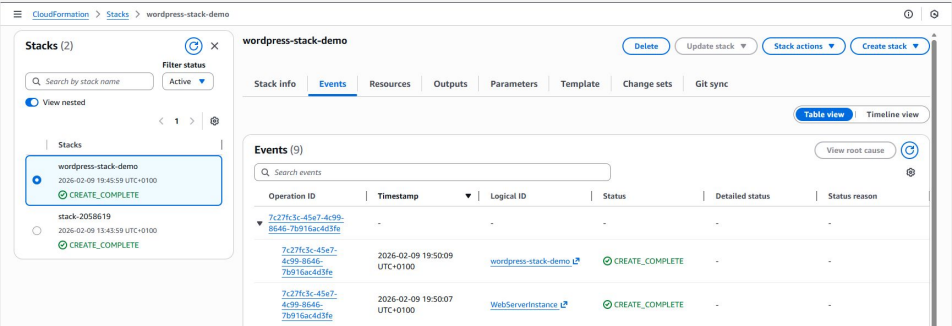
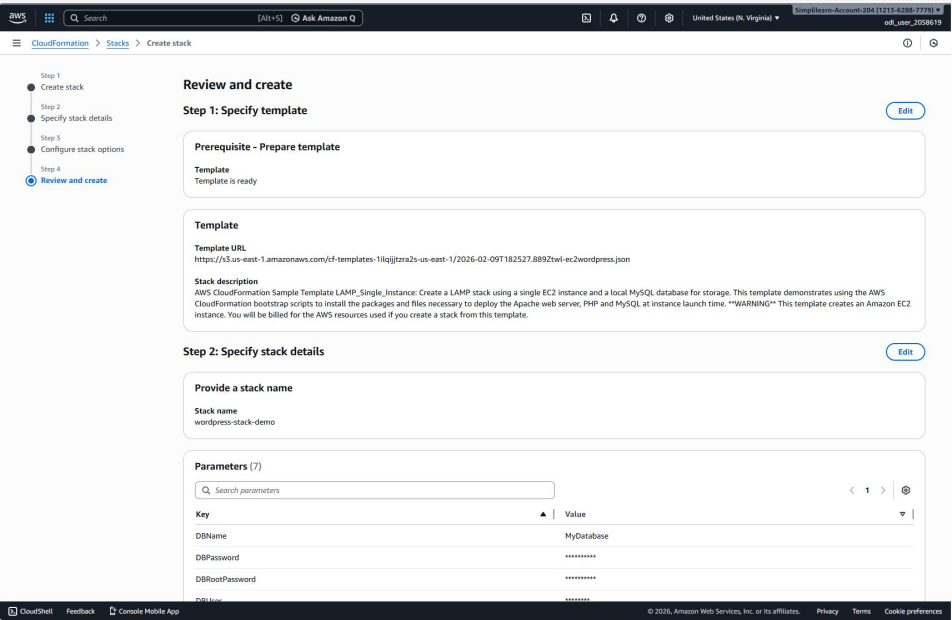


15. Click next

15

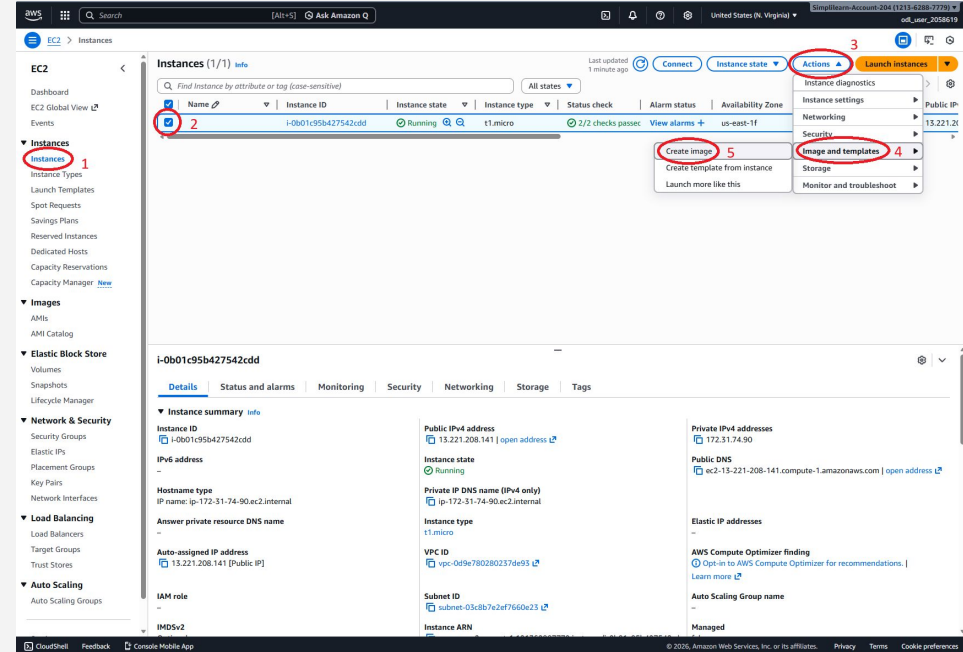
16. Review all the stack configuration details and click Submit

After a few seconds, the stack will be created.



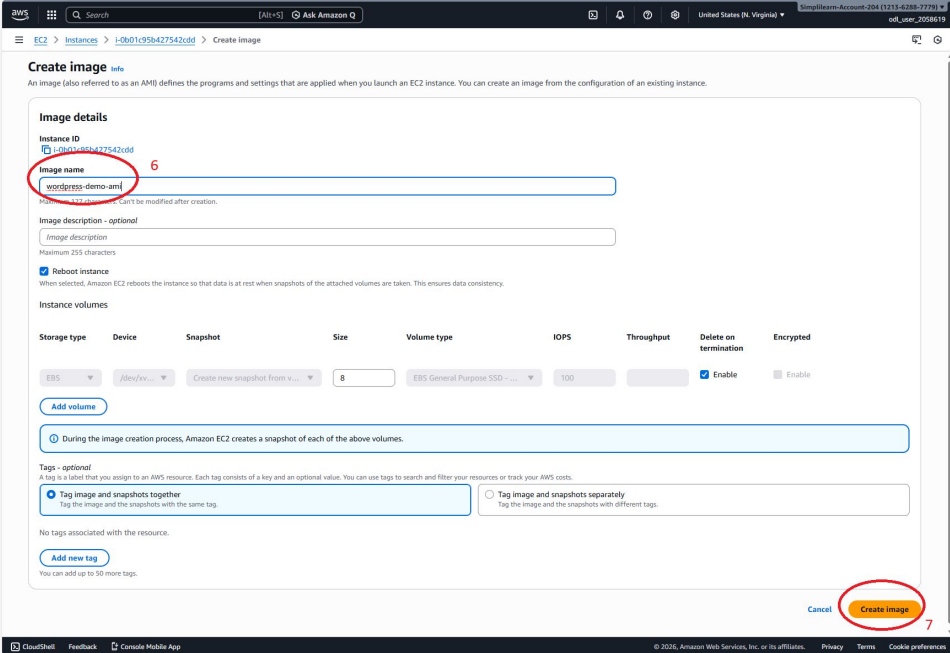
Create an AMI of the WordPress instance

1. Open the EC2 Instances page
2. Select the instance
3. Click on Actions
4. Select Image and templates
5. Click on Create image

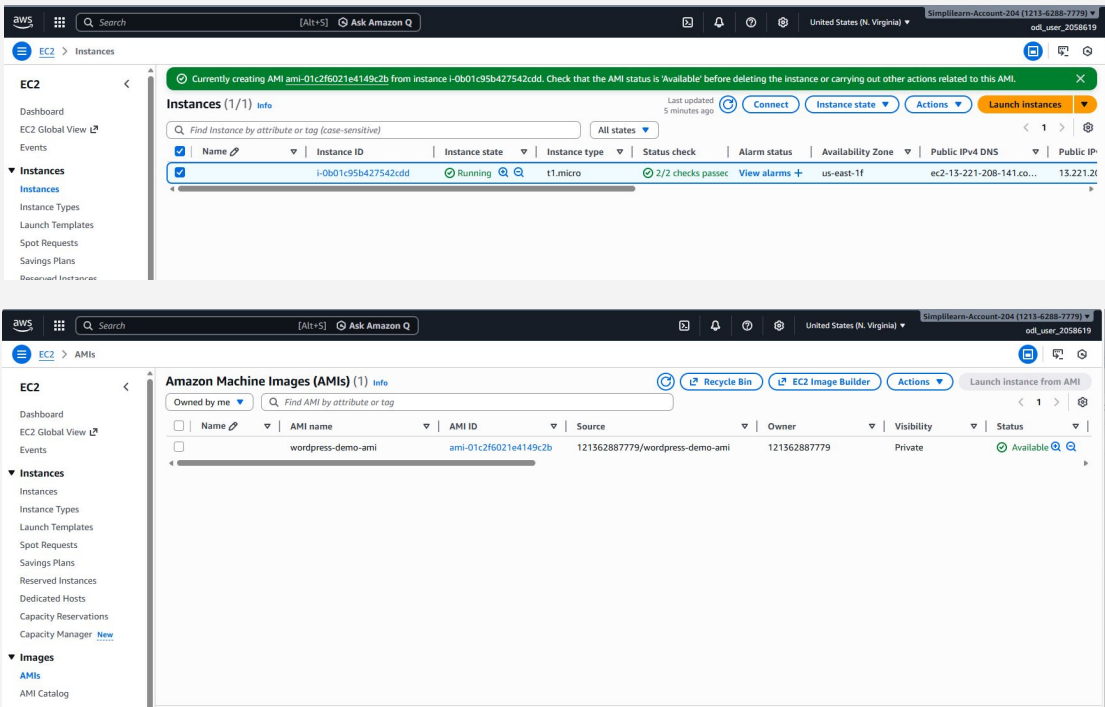


6. Type image name

7. Click on Create Image

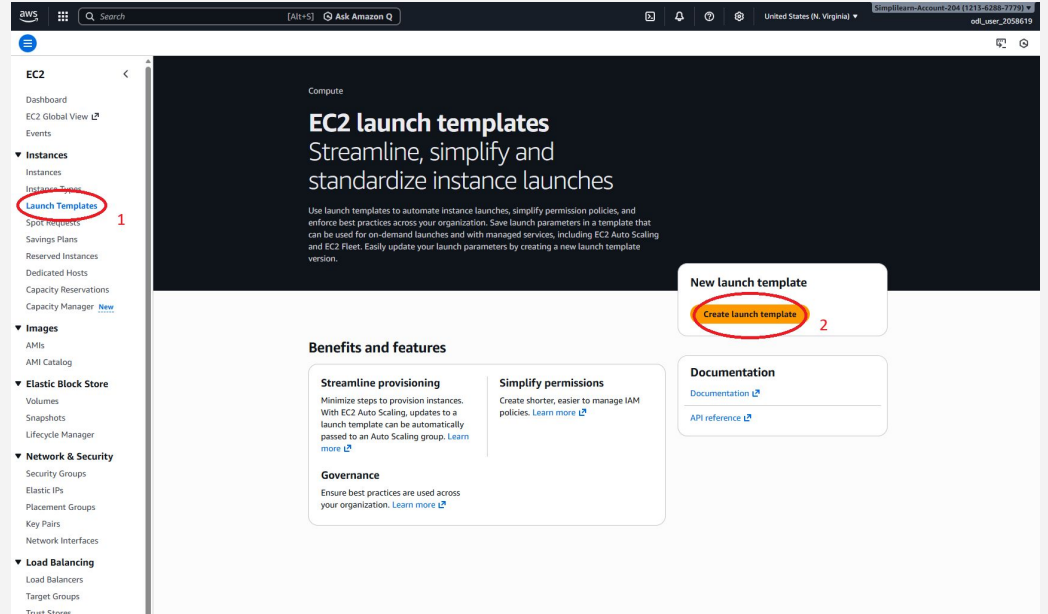


By following steps the AMI will be successfully created

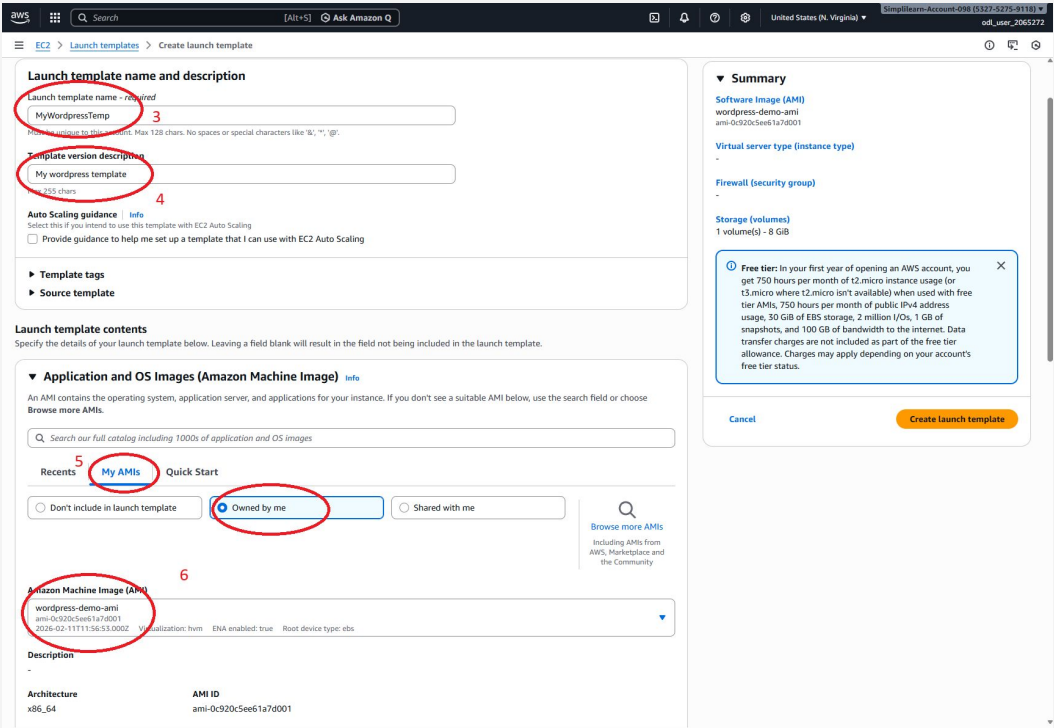


Configure Auto Scaling to launch a new WordPress instance

1. Select Launch Templates
2. Click on Create Launch template button



- 3. Name the Launch template
- 4. Provide a description
- 5. Select My AMIs
- 6. Select the previously created AMI



7. Select Instance type

8. Select Key pair name

9. The security group needs to be the same as the first instance security group

10. Click on Create launch template

▼ Instance type

Info | Get advice

Advanced

Instance type

7

t1.micro
Family: t1 1 vCPU 0.512 GiB Memory Current generation: false On-Demand SUSE base pricing: 0.02 USD per Hour
On-Demand Windows base pricing: 0.02 USD per Hour On-Demand Linux base pricing: 0.02 USD per Hour
On-Demand ARM64 base pricing: 0.03 USD per Hour

Additional costs apply for AMIs with pre-installed software

All generations

Compare instance types

▼ 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

8

MyDemoKey

Create new key pair

Security groups

Info

9

Select security groups

wordpress-stack-demo-WebServerSecurityGroup-v7sDO3mbNhpds g-080a8f5438268f5ec X
VPC: vpc-0cae8f928ddb086cf

Compare security group rules

► Advanced network configuration

10. Select Auto Scaling Groups

11. Click on Create Auto Scaling group button

EC2

Dashboard

EC2 Global View

Events

Instances

Instance Types

Launch Templates

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

Amazon EC2 Auto Scaling

helps maintain the availability of your applications

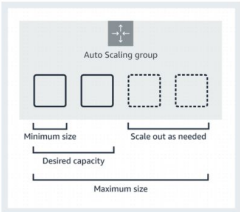
Auto Scaling groups are collections of Amazon EC2 instances that enable automatic scaling and fleet management features. These features help you maintain the health and availability of your applications.

Create Auto Scaling group

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

Create Auto Scaling group

How it works



The diagram illustrates an Auto Scaling group as a container holding several EC2 instances. A bracket at the bottom indicates the 'Desired capacity'. Another bracket below it shows the 'Minimum size' and 'Maximum size' limits. A label 'Scale out as needed' points to the right side of the group, indicating automatic expansion.

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 capacity values forms the bounds of the group, which allows the group to scale as the load on your application spikes higher or lower, based on demand. To scale the Auto Scaling group, you can either make manual adjustments to the desired capacity or let Amazon EC2 Auto Scaling automatically add and remove capacity to meet changes in demand.

When launching fleets of instances, 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.

Benefits and features

Pricing

Amazon EC2 Auto Scaling features have no additional fees beyond the service fees 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?

Getting started with Amazon EC2 Auto Scaling

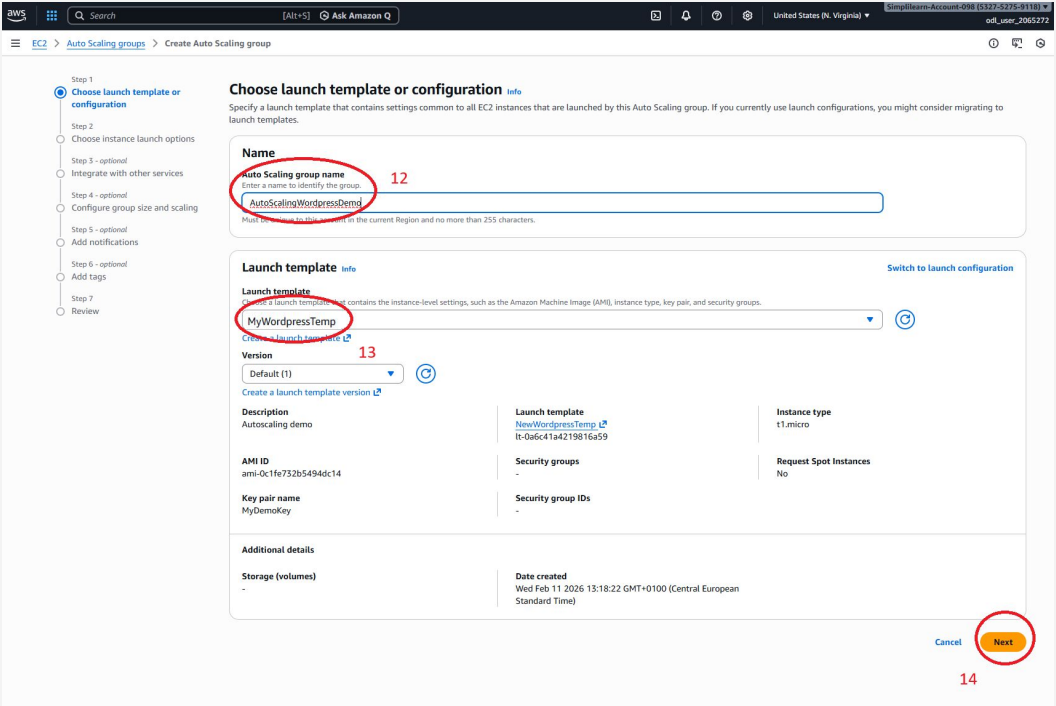
Set up a scaled and load-balanced application

FAQ

12. Name the Auto Scaling group

13. Select Launch template

14. Click on Next button



Select Availability Zones and click on Next button

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-0c2a74bc545fb3b28

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

use1-az1 (us-east-1c) | subnet-0a6c6051576d0e681

172.31.0.0/20

Default

use1-az2 (us-east-1d) | subnet-01adf0571c288c5ce

172.31.80.0/20

Default

use1-az3 (us-east-1e) | subnet-04fcf353af118208e

172.31.48.0/20

Default

use1-az4 (us-east-1a) | subnet-03901e2a2b4218969

172.31.16.0/20

Default

use1-az5 (us-east-1f) | subnet-00b014264077a13a2

172.31.64.0/20

Default

use1-az6 (us-east-1b) | subnet-00bc0f3b1bfb131d7

172.31.32.0/20

Default

[Create a subnet](#)

Availability Zone distribution - new
Auto Scaling automatically balances instances across Availability Zones. If launch failures occur in a zone, select a strategy.

☒ **Balanced best effort**
If launches fail in one Availability Zone, Auto Scaling will attempt to launch in another healthy Availability Zone.

☐ **Balanced 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.

Your requested instance type (t1.micro) is not available in 1 Availability Zone. You may need to change the instance type or choose other Availability Zones for better resiliency. [Learn more](#)

Cancel

Skip to review

Previous

Next

Click on Attach to a new load balancer and create a target group

Click Next and skip the other steps. In step 7, click the Create Auto Scaling Group button.

Integrate with other services

Step 4 - optional

Configure group size and scaling

Step 5 - optional

Add notifications

Step 6 - optional

Add tags

Step 7

Review

Use the options below 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

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 a new load balancer

Load balancer type

Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the [Load Balancing console](#).

☒ Application Load Balancer

HTTP, HTTPS

☐ Network Load Balancer

TCP, UDP, TLS

Load balancer name

Name cannot be changed after the load balancer is created.

AutoScalingWordpressDemo-1

Load balancer scheme

Scheme cannot be changed after the load balancer is created.

☒ Internal

☐ Internet-facing

Network mapping

Your new load balancer will be created using the same VPC and Availability Zone selections as your Auto Scaling group. You can select different subnets and add subnets from additional Availability Zones.

VPC

vpc-0c2a74bc545fb3b28

Availability Zones and subnets

You must select a single subnet for each Availability Zone enabled. Only public subnets are available for selection to support DNS resolution.

☒ use1-az1 (us-east-1c)

Select a subnet
subnet-0a6c6051576d0e681

☒ use1-az2 (us-east-1d)

Select a subnet
subnet-01adf0571c288c5ce

☒ use1-az5 (us-east-1f)

Select a subnet
subnet-009b014264077a13a2

☒ use1-az4 (us-east-1a)

Select a subnet
subnet-03901e2a2b4218969

☒ use1-az6 (us-east-1b)

Select a subnet
subnet-00bc0f3b1bf0131d7

☒ use1-az3 (us-east-1e)

Select a subnet
subnet-04fcf553af118208e

Listeners and routing

If you require secure listeners, or multiple listeners, you can configure them from the [Load Balancing console](#) after your load balancer is created.

Protocol

HTTP

Port

80

Default routing (forward to)

Create a target group

New target group name

An internet target group with default settings will be created.

AutoScalingWordpressDemo-1

Tags - optional

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

Configure the new WordPress instance to shut down automatically

1. Select the Auto Scaling Group
2. Select Automatic scaling
3. Click on Create scheduled action

The screenshot displays the AWS Management Console interface for configuring an Auto Scaling group. The left-hand navigation pane shows the 'EC2' section expanded, with 'Auto Scaling Groups' selected under the 'Instances' category. The main content area shows the 'Auto Scaling groups (1/1)' page. A table lists the existing group, 'AutoScalingWordPressDemo', which is selected with a red circle and labeled '1'. Below the table, the 'Automatic scaling' tab is selected and circled in red, labeled '2'. This tab displays information about scaling policies and dynamic scaling policies. At the bottom of the console, the 'Scheduled actions (0)' section is visible, where the 'Create scheduled action' button is circled in red and labeled '3'.

EC2 > Auto Scaling groups

Auto Scaling groups (1/1) Info

Launch configurations Launch templates Actions Create Auto Scaling group

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
AutoScalingWordPressDemo	MyWordPressTemp Version Default	1	-	1	1	1	6 Availability Zones

1

Auto Scaling group: AutoScalingWordPressDemo

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

2

Scaling policies resize your Auto Scaling group to meet changes in demand. With reactive dynamic scaling policies, you can track specific CloudWatch metrics and take action when the CloudWatch alarm threshold is met. Use predictive scaling policies along with dynamic scaling policies in the following situations: when your application demand changes quickly, but with a recurring pattern, or when your EC2 instances require more time to initialize.

Dynamic scaling policies (0) Info

Actions Create dynamic scaling policy

1

Network Interfaces

Load Balancing

Auto Scaling

Auto Scaling Groups

Scheduled actions (0) Info

Filter scheduled actions

Name	Start time	End time	Recurrence	Time zone
No scheduled actions are currently specified				

Create scheduled action

3

4. Type the action name

5. Type the desired instance capacity

6. Choose Cron and the action time (in this case 09:00 Monday-Friday)

7. Select the start date and time

8. Create action

Create scheduled action

Name
start-wordpress-dev

Desired capacity 1 **Min** 1 **Max** 1

Recurrence
Schedule a specific date and time for the first scheduled action to run. Interpreted in recurrence time zone: Etc/UTC
Cron 0 9 ** MON-FRI

Time zone
Etc/UTC
Current time in selected time zone is 2026-02-11/17:18 UTC

Start date 2026/02/12 **Start time** 08:00 Etc/UTC
Format: YYYY/MM/DD. Format: hh:mm

Set End Time

[Learn more about scheduled scaling](#)

Cancel Create

Create scheduled action

Name

stop-wordpress-dev

Provide at least one value for Desired, Min, or Max Capacity

Desired capacity

Min

Max

0

0

0

Recurrence

Schedule a specific date and time for the first scheduled action to run. Interpreted in recurrence time zone: Etc/UTC

Cron

0 17 ** MON-FRI

Time zone

Etc/UTC

Current time in selected time zone is 2026-02-11/17:20 UTC

Start date

Start time

2026/02/12

16:00

Etc/UTC

Format: YYYY/MM/DD

Format: hh:mm.

Set End Time

Learn more about scheduled scaling

Cancel

Create

9. Type the action name

10. Type the desired instance capacity

11. Choose Cron and the action time (in this case 17:00 Monday-Friday)

12. Select the start date and time

13. Create action

Two scheduled actions are created: **start-wordpress-dev**, which starts the development instance, and **stop-wordpress-dev**, which shuts down the development instance.

Network Interfaces

▼ Load Balancing

Load Balancers

Target Groups

Trust Stores

▼ Auto Scaling

Auto Scaling Groups

Settings

Scheduled actions (2) [Info](#)

⌕

Filter scheduled actions

⌂

Actions ▼







Create scheduled action

< 1 > ⚙






<input type="checkbox"/>	Name	Start time	End time	Recurrence	Time zone
<input type="checkbox"/>	start-wordpress-dev	2026 February 12, 09:00:00 AM +...		0 9 * * MON-FRI	Etc/UTC
<input type="checkbox"/>	stop-wordpress-dev	2026 February 12, 05:00:00 PM +...		0 17 * * MON-FRI	Etc/UTC

Output Screenshots

Instances

Instance ID	Instance state ▾	Instance type ▾	Status check	Alarm status	Availability Zone ▾	Public IPv4 DNS ▾	Public IPv4 ... ▾	Elastic IP ▾	IPv6 IPs ▾	Monitor... ▾	Security group name ▾
i-0113204912c0e9e40	Running  	t1.micro	 2/2 checks passed	View alarms +	us-east-1b	ec2-54-174-118-91.co...	54.174.118.91	–	–	disabled	wordpress-stack-demo...
i-0f786ec2f03795f1e	Running  	t1.micro	 2/2 checks passed	View alarms +	us-east-1f	ec2-44-192-69-206.co...	44.192.69.206	–	–	disabled	wordpress-stack-demo...

Health checks

<input type="checkbox"/>	ID ▲	Name ▾	Details ▾	Status in last 24 hours	Current s... ▾	Alarm ▾	State ▾	Actions
<input type="checkbox"/>	087f7f6e-067d-4687-b83d-9cd4101bbcca	wordpress-dev	http://44.192.69.206:80/	<input type="text"/>	 Healthy	None, Create alarm	 Enabled	
<input type="checkbox"/>	96ed5e70-26ca-41e7-a9ca-e1230b6d228e	wordpress-prod	http://54.174.118.91:80/	<input type="text"/>	 Healthy	None, Create alarm	 Enabled	