

PART-1

STEP 1-

LAUNCH AN INSTANDCE

▼ 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

Recents

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE L

SUSE

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

ami-0f403e3180720dd7e (64-bit (x86), uefi-preferred) / ami-0237525b5672165b3 (64-bit (Arm), uefi)

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

Free tier eligible

▼ Summary

Number of instances [Info](#)

Software Image (AMI)

Amazon Linux 2023 AMI 2023.3.2...[read more](#)

ami-0f403e3180720dd7e

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Cancel

Launch instance

[Review commands](#)

© 2024 Amazon Web Services, Inc. or its affiliates. Privacy Policy Terms Cookies

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*

Proceed without a key pair (Not recommended)

Default value ▼

🔄 [Create new key pair](#)

▼ Network settings [Info](#)

VPC - *required* [Info](#)

vpc-04643f4dea97e29fa
172.31.0.0/16

(default) ▼



Subnet [Info](#)

subnet-02c5d405a7b08401b

VPC: vpc-04643f4dea97e29fa Owner: 905418179079 Availability Zone: us-east-1f
IP addresses available: 4090 CIDR: 172.31.64.0/20



[Create new subnet](#)

Auto-assign public IP [Info](#)

Enable ▼

▼ Summary

Number of instances [Info](#)

1

Software Image (AMI)

Amazon Linux 2023 AMI 2023.3.2...[read more](#)
ami-0f403e3180720dd7e

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Cancel

Launch instance

Services [Search](#)

[Alt+S]



N. Virginia ▼

Vicky Omprakash

EC2 VPC S3 AWS Auto Scaling Simple Queue Service Simple Notification Service Key Management Service

[EC2](#) > [Instances](#) > Launch an instance



Success

Successfully initiated launch of instance [\(i-05973d02531f0b1ef\)](#)

► Launch log

Next Steps

[Q](#) What would you like to do next with this instance, for example "create alarm" or "create backup"

< 1 2 3 4 5 6 >

Create billing and free tier usage alerts

To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.

Connect to your instance

Once your instance is running, log into it from your local computer.

[Connect to instance](#)

Connect an RDS database

Configure the connection between an EC2 instance and a database to allow traffic flow between them.

[Connect an RDS database](#)

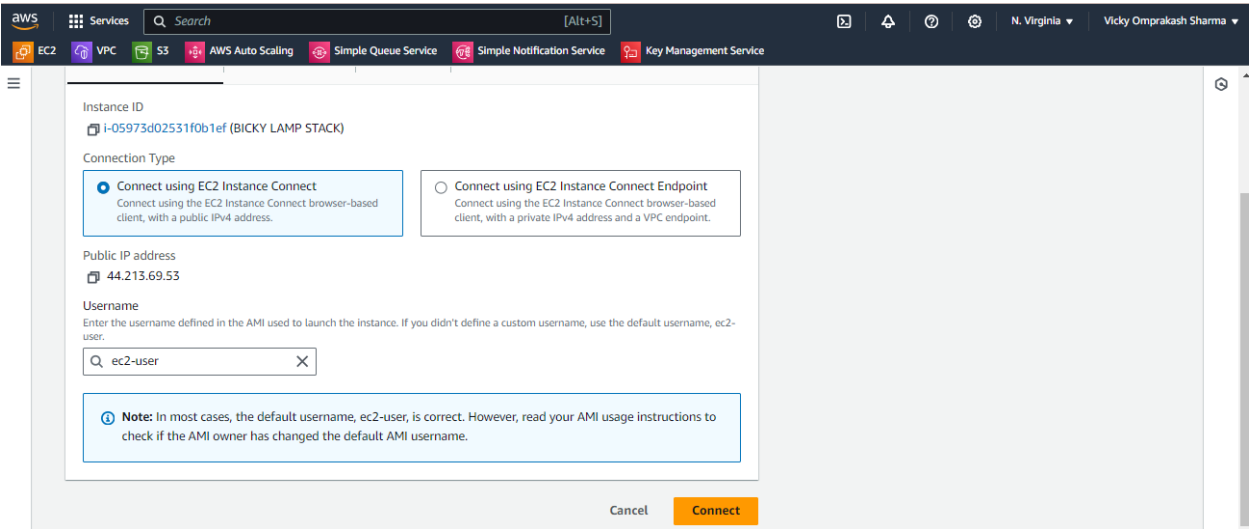
Create EBS snapshot policy

Create a policy that automates the creation, retention, and deletion of EBS snapshots

[Create EBS snapshot policy](#)

STEP 2)

NOW CONNECT THE INSTANCE



PART 2-

BRIEF SUMMARY WHAT WE GONNA DO NOW

Updating system packages for a smooth setup

Installing the Apache web server for web hosting. ➦ PHP version 8

INSTALLING PHP 8 and its extensions for dynamic web content.

Installing MySQL 8 for data-driven web applications.

- Activating the PHP_CURL module for enhanced data communication. Update the package repository:

[illegible]

Step 3)

Cmd - sudo yum install httpd -y

Cmd - `sudo systemctl start httpd`

```
Complete!  
[root@ip-172-31-68-81 ec2-user]# sudo systemctl start httpd  
[root@ip-172-31-68-81 ec2-user]#
```

Step 5) Enable Apache to start on boot:

Cmd -sudo systemctl enable httpd

```
[root@ip-172-31-68-81 ec2-user]# sudo systemctl start httpd
[root@ip-172-31-68-81 ec2-user]# sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service -> /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-68-81 ec2-user]#
```

i-040ed14f3998075de (DEPAK-LAMP-EC2)
PublicIPs: 44.197.116.125 PrivateIPs: 172.31.68.81

2) PHP version 8

To install PHP version 8 on Amazon Linux 2023, you can use the Remi repository, which provides up-to-date versions of PHP. Here's how you can do it:

1. Enable the Remi repository:

```
[root@ip-172-31-68-81 ec2-user]# sudo amazon-linux-extras enable epel
sudo yum install https://rpms.remirepo.net/enterprise/remi-release-7.rpm
sudo: amazon-linux-extras: command not found
Last metadata expiration check: 0:16:59 ago on Mon Mar 11 20:37:44 2024.
remi-release-7.rpm
Error:
  Problem: conflicting requests
   - nothing provides epel-release = 7 needed by remi-release-7.9-6.el7.remi.noarch
   - nothing provides redhat-release needed by remi-release-7.9-6.el7.remi.noarch
(try to add '--skip-broken' to skip uninstallable packages)
[root@ip-172-31-68-81 ec2-user]#
```

2. Install PHP 8:

Cmd - `sudo yum-config-manager --enable remi-php80`

`sudo yum install php php-cli php-fpm php-mysqld`

```
sudo yum install php php-cli php-fpm php-mysqld
Error: No matching repo to modify: remi-php80.
Last metadata expiration check: 0:18:18 ago on Mon Mar 11 20:37:44 2024.
Dependencies resolved.
=====
Package                                Architecture      Version           Repository        Size
-----
Installing:
php8.2                                x86_64            8.2.15-1.amzn2023.0.1  amazonlinux      11 k
php8.2-fpm                            x86_64            8.2.15-1.amzn2023.0.1  amazonlinux      1.9 M
php8.2-mysqld                         x86_64            8.2.15-1.amzn2023.0.1  amazonlinux      148 k
Installing dependencies:
libsodium                             x86_64            1.0.18-13.amzn2023.0.1  amazonlinux      166 k
libxslt                               x86_64            1.1.34-5.amzn2023.0.2  amazonlinux      241 k
nginxfilesystem                       noarch            1:1.24.0-1.amzn2023.0.2  amazonlinux      9.1 k
php8.2-cli                            x86_64            8.2.15-1.amzn2023.0.1  amazonlinux      3.6 M
php8.2-common                         x86_64            8.2.15-1.amzn2023.0.1  amazonlinux      704 k
php8.2-pdo                            x86_64            8.2.15-1.amzn2023.0.1  amazonlinux      89 k
php8.2-process                        x86_64            8.2.15-1.amzn2023.0.1  amazonlinux      44 k
php8.2-xml                            x86_64            8.2.15-1.amzn2023.0.1  amazonlinux      148 k
=====
aws  Services  Search  [Alt+S]
=====
EC2  VPC  S3  AWS Auto Scaling  Simple Queue Service  Simple Notification Service  Key Management Service
Verifying : php8.2-xml-8.2.15-1.amzn2023.0.1.x86_64 1/14
Verifying : php8.2-mysqld-8.2.15-1.amzn2023.0.1.x86_64 2/14
Verifying : php8.2-8.2.15-1.amzn2023.0.1.x86_64 3/14
Verifying : php8.2-pdo-8.2.15-1.amzn2023.0.1.x86_64 4/14
Verifying : php8.2-common-8.2.15-1.amzn2023.0.1.x86_64 5/14
Verifying : libsodium-1.0.18-13.amzn2023.0.1.x86_64 6/14
Verifying : php8.2-cli-8.2.15-1.amzn2023.0.1.x86_64 7/14
Verifying : php8.2-fpm-8.2.15-1.amzn2023.0.1.x86_64 8/14
Verifying : libxslt-1.1.34-5.amzn2023.0.2.x86_64 9/14
Verifying : php8.2-sodium-8.2.15-1.amzn2023.0.1.x86_64 10/14
Verifying : php8.2-process-8.2.15-1.amzn2023.0.1.x86_64 11/14
Verifying : php8.2-opcache-8.2.15-1.amzn2023.0.1.x86_64 12/14
Verifying : php8.2-mbstring-8.2.15-1.amzn2023.0.1.x86_64 13/14
Verifying : nginxfilesystem-1:1.24.0-1.amzn2023.0.2.noarch 14/14
Installed:
libsodium-1.0.18-13.amzn2023.0.1.x86_64  libxslt-1.1.34-5.amzn2023.0.2.x86_64  nginxfilesystem-1:1.24.0-1.amzn2023.0.2.noarch
php8.2-8.2.15-1.amzn2023.0.1.x86_64  php8.2-cli-8.2.15-1.amzn2023.0.1.x86_64  php8.2-common-8.2.15-1.amzn2023.0.1.x86_64
php8.2-fpm-8.2.15-1.amzn2023.0.1.x86_64  php8.2-mbstring-8.2.15-1.amzn2023.0.1.x86_64  php8.2-mysqld-8.2.15-1.amzn2023.0.1.x86_64
php8.2-opcache-8.2.15-1.amzn2023.0.1.x86_64  php8.2-pdo-8.2.15-1.amzn2023.0.1.x86_64  php8.2-process-8.2.15-1.amzn2023.0.1.x86_64
php8.2-sodium-8.2.15-1.amzn2023.0.1.x86_64  php8.2-xml-8.2.15-1.amzn2023.0.1.x86_64
Complete!
[root@ip-172-31-68-81 ec2-user]#
```

NOW VERIFY INSTALLMENT OF PHP OCCUR OR NOT
CMD-`php --version`

```
Complete!
[root@ip-172-31-68-81 ec2-user]# php --version
PHP 8.2.15 (cli) (built: Jan 16 2024 12:19:32) (NTS gcc x86_64)
Copyright (c) The PHP Group
Zend Engine v4.2.15, Copyright (c) Zend Technologies
    with Zend OPcache v8.2.15, Copyright (c), by Zend Technologies
[root@ip-172-31-68-81 ec2-user]#
```

i-040ed14f3998075de (DEPAK-LAMP-EC2)

NOW ADDITIONALLY ALSO DOWNLOAD PHP ALL EXTENSION FILE NOW

Install PHP 8.0 with all extensions:

CMD- `sudo yum-config-manager --enable remi-php80`

`sudo yum install php php-cli php-fpm php-mysqlnd php-zip php-devel php-gd php-mcrypt php-mbstring php-curl php-xml php-pear php-bcmath php-json php-xmldr`

```
[root@ip-172-31-68-81 ec2-user]# sudo yum-config-manager --enable remi-php80
sudo yum install php php-cli php-fpm php-mysqlnd php-zip php-devel php-gd php-mcrypt php-mbstring php-curl php-xml php-pear php-bcmath php-json php-xmldr
Error: No matching repo to modify: remi-php80.
Last metadata expiration check: 0:26:45 ago on Mon Mar 11 20:37:44 2024.
Package php8.2-8.2.15-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-cli-8.2.15-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-fpm-8.2.15-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-mysqlnd-8.2.15-1.amzn2023.0.1.x86_64 is already installed.
No match for argument: php-mcrypt
Package php8.2-mbstring-8.2.15-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-common-8.2.15-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-xml-8.2.15-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-common-8.2.15-1.amzn2023.0.1.x86_64 is already installed.
No match for argument: php-xmldr
Error: Unable to find a match: php-mcrypt php-xmldr
[root@ip-172-31-68-81 ec2-user]#
```

i-040ed14f3998075de (DEPAK-LAMP-EC2)

PublicIPs: 44.197.116.125 PrivateIPs: 172.31.68.81

3)
Installing MySQL 8 for data-driven web applications.

FOR THESE we will need to download the RPM file using this

Cmd -

```
[root@ip-172-31-68-81 ec2-user]# sudo wget https://dev.mysql.com/get/mysql80-community-release-el9-1.noarch.rpm
--2024-03-11 21:16:22-- https://dev.mysql.com/get/mysql80-community-release-el9-1.noarch.rpm
Resolving dev.mysql.com (dev.mysql.com)... 23.34.160.157, 2600:1408:7400:493::2e31, 2600:1408:7400:4b7::2e31
Connecting to dev.mysql.com (dev.mysql.com)|23.34.160.157|:443... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: https://repo.mysql.com//mysql80-community-release-el9-1.noarch.rpm [following]
--2024-03-11 21:16:23-- https://repo.mysql.com//mysql80-community-release-el9-1.noarch.rpm
Resolving repo.mysql.com (repo.mysql.com)... 23.43.169.39, 2600:1408:ec00:886::1d68, 2600:1408:ec00:88f::1d68
Connecting to repo.mysql.com (repo.mysql.com)|23.43.169.39|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10534 (10K) [application/x-redhat-package-manager]
Saving to: 'mysql80-community-release-el9-1.noarch.rpm'

mysql80-community-release-el9-1.noarch.r 100%[=====>
2024-03-11 21:16:23 (208 MB/s) - 'mysql80-community-release-el9-1.noarch.rpm' saved [10534/10534]

[root@ip-172-31-68-81 ec2-user]#
```

You can confirm the file has been downloaded by running

Cmd -ls -lrt

```
[root@ip-172-31-68-81 ec2-user]# ls -lrt
total 12
-rw-r--r--. 1 root root 10534 Jul 25  2022 mysql80-community-release-el9-1.noarch.rpm
[root@ip-172-31-68-81 ec2-user]#
```

After downloading the next step is to install the file then install mysql-community-server using the below commands

```
Cmd- sudo dnf install mysql80-community-release-el9-1.noarch.rpm -y
sudo dnf install mysql-community-server -y
```

Services

EC2VPCS3AWS Auto ScalingSimple Queue ServiceSimple Notification ServiceKey Management Service

total 12

-rw-r--r--. 1 root root 10534 Jul 25 2022 mysql80-community-release-el9-1.noarch.rpm

[root@ip-172-31-68-81 ec2-user]# sudo dnf install mysql80-community-release-el9-1.noarch.rpm -y

sudo dnf install mysql-community-server -y

Last metadata expiration check: 0:07:51 ago on Mon Mar 11 21:11:12 2024.

Dependencies resolved.

Package	Architecture	Version	Repository	Size
Upgrading:				
mysql80-community-release	noarch	el9-1	@commandline	10 k

Transaction Summary

Upgrade 1 Package

Total size: 10 k

Downloading Packages:

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing : 1/1

i-040ed14f3998075de (DEPAK-LAMP-EC2)

PublicIPs: 44.197.116.125 PrivateIPs: 172.31.68.81

Next

ERROR: GPG CHECK FAILED

[root@ip-172-31-68-81 ec2-user]# sudo systemctl start mysqld

Failed to start mysqld.service: Unit mysqld.service not found.

[root@ip-172-31-68-81 ec2-user]#

step is to start mysql