How to create an EC2 instance on AWS

Introduction

AWS is today's biggest cloud provider, providing more than 240 services. Today, we will look at one of the most popular services, Amazon EC2. With an EC2 instance, you can host your virtual server in the cloud for a cheap price. What makes EC2 such a nice option is its array of options to customize your virtual server from OS to memory to CPU and more as we will see today. With an EC2 instance, you can use it to host applications, use it as a database, and more.

These instructions will be laid out for anyone with no technical background required. However, it would be best to have some technical knowledge to customize your virtual server for your needs. In this tutorial, I will customize my server based on the needs of a standard virtual server

List of Tools

- A laptop or computer with access to the internet
- Browser of your choice

Prerequisites

- Ensure you have an SSH Client installed, which can be checked by running ssh in your command prompt. If not, you can install OpenSSH for Windows here

Warnings

- If you create an EC2 instance and don't intend to use it, don't forget to turn off your EC2 instance so as not to get billed
- This tutorial is shown using a Windows machine, but all the instructions should remain the same

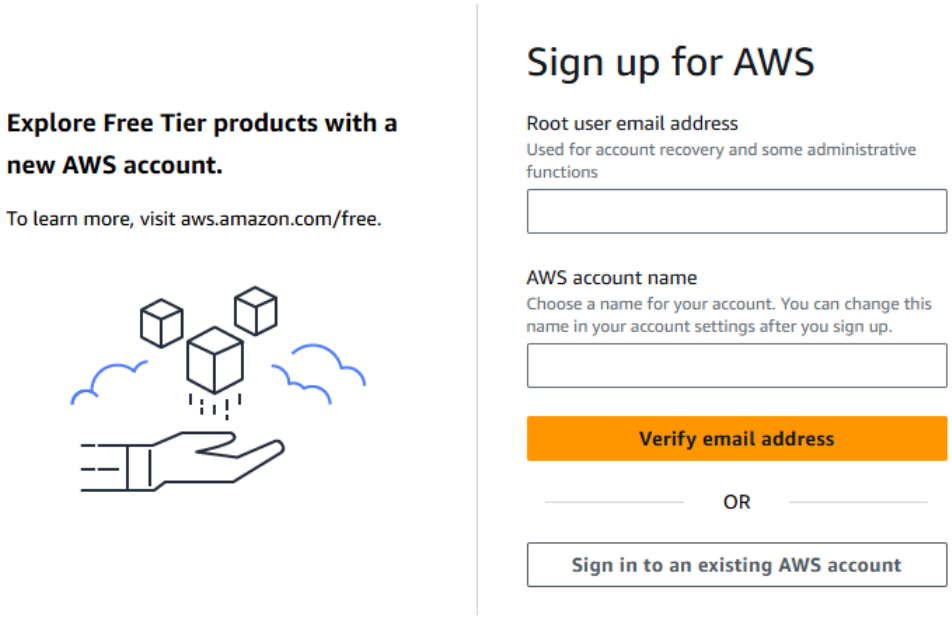
Quick Start Guide

- Create AWS account
- Create EC2 instance
- Verify EC2 instance is up and running
- Terminating EC2 instance

Glossary

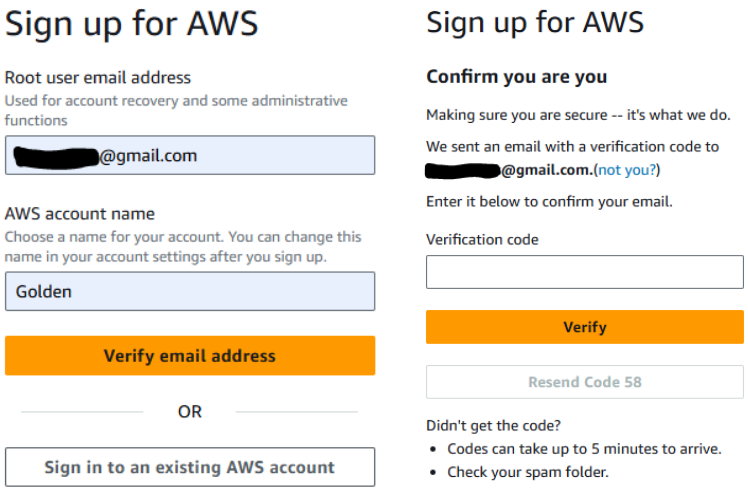
- **Amazon Machine Image (AMI)**: software used to setup an EC2 instance. For more information refer to this article

- **Instance Type:** refers to different kinds of instances with different CPU, memory, storage, networking capacity, and resources. For more information, refer to this article

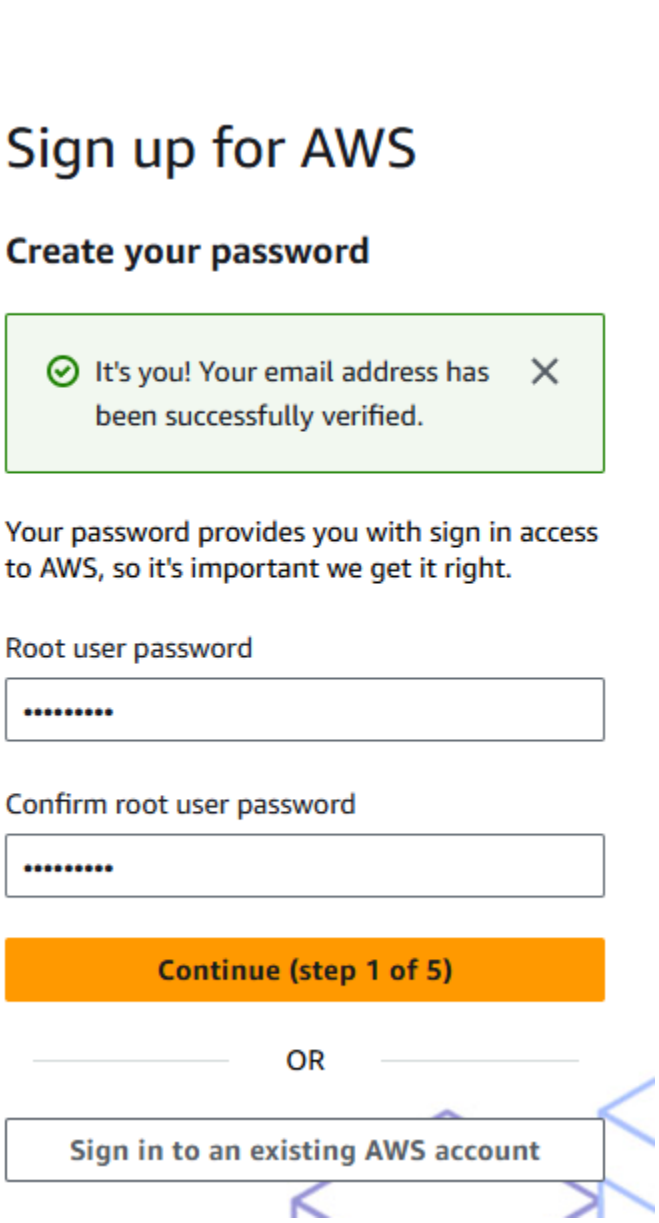


Create AWS Account

1. Go to this link to create an AWS account
2. Enter your email address and choose an AWS account name
3. Press “Verify email address”
4. Enter the verification code sent to your email



5. Create your password and press “Continue”
6. Enter your contact information and press “Agree and Continue”
7. Enter your billing information and press “Verify and continue
8. Enter your phone number to confirm your identity and press “Send SMS”
9. Enter your verification code sent to your phone
10. Select the “Basic support - Free” plan and press “Complete sign up”



Sign up for AWS

Billing Information

Billing country
Your billing country determines the payment methods available to you to pay for AWS services.

United States

Credit or Debit card number

[redacted]

AWS accepts most major credit and debit cards. To learn more about payment options, review our FAQ

Expiration date

[redacted] [redacted]

Security code

[redacted]

Cardholder's name

[redacted]

Billing address

- (•) Use my contact address

[redacted]

- ( ) Use a new address

Verify and continue (step 3 of 5)

You might be redirected to your bank's website to authorize the verification charge.

Sign up for AWS

Confirm your identity

Before you can use your AWS account, you must verify your phone number. When you continue, the AWS automated system will contact you with a verification code.

How should we send you the verification code?

- (•) Text message (SMS)
- ( ) Voice call

Country or region code

United States (+1)

Mobile phone number

[redacted]

Send SMS (step 4 of 5)

Sign up for AWS

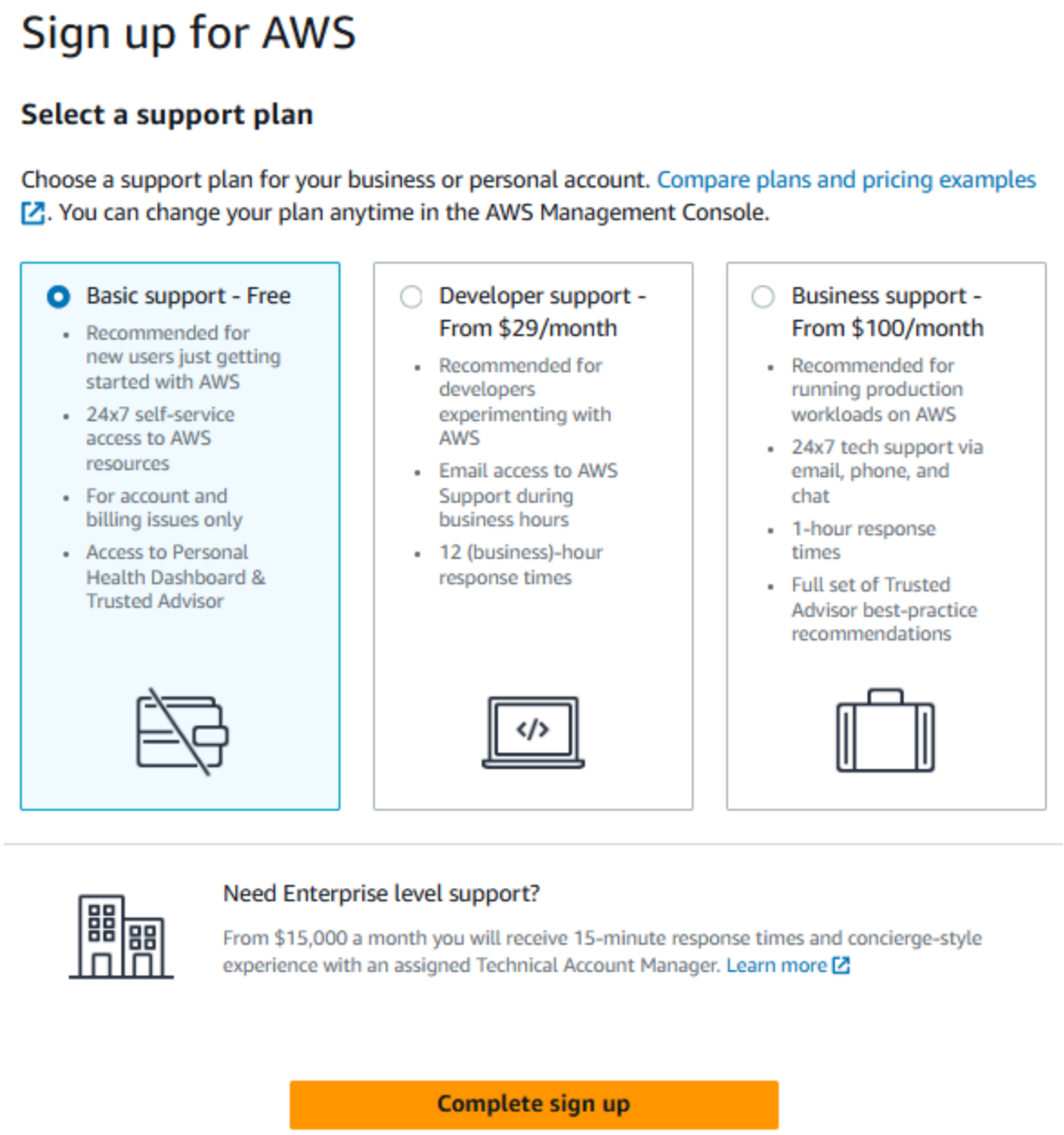
Confirm your identity

Verify code

5201

Continue (step 4 of 5)

Having trouble? Sometimes it takes up to 10 minutes to retrieve a verification code. If it's been longer than that, return to the previous page and try again.



Creating EC2 Instance

Prereq

- You should have created an AWS account

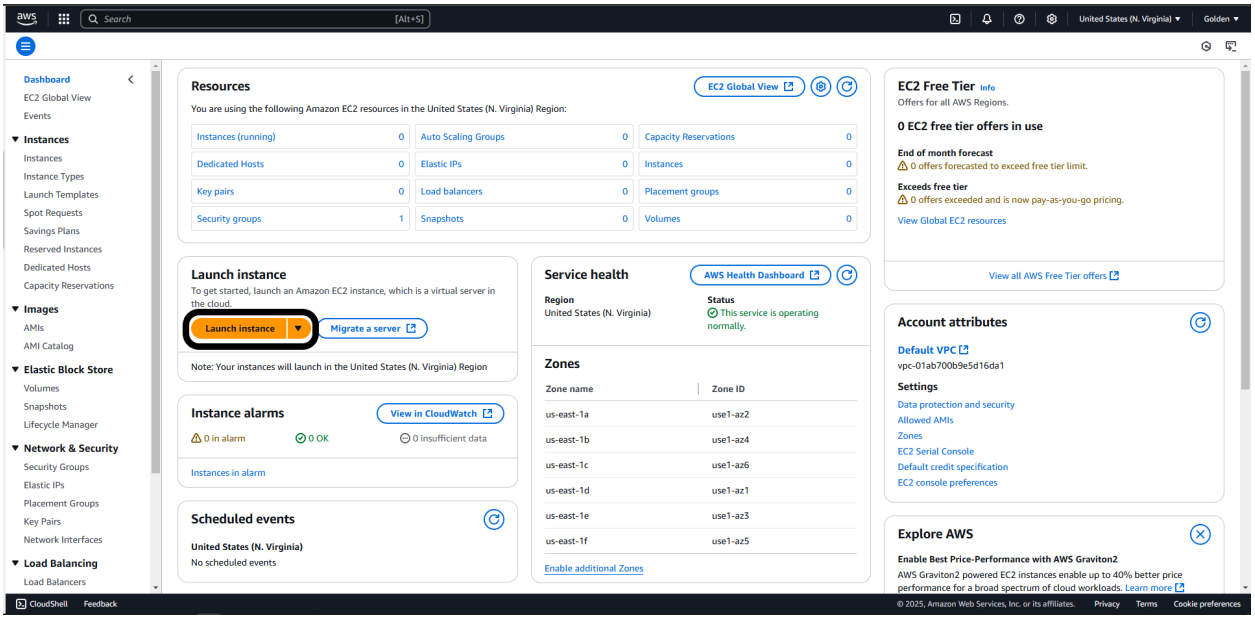
Steps

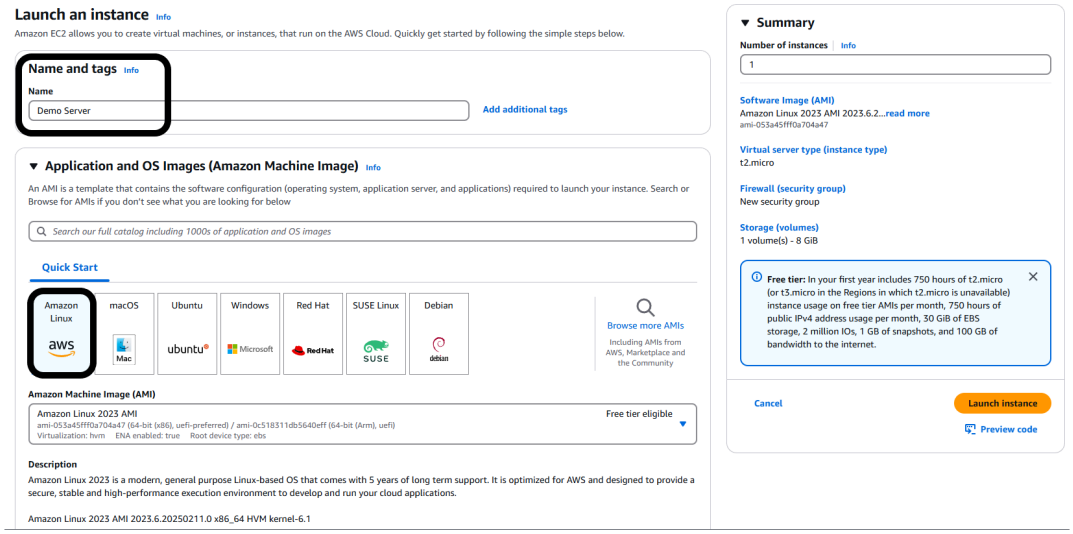
1. Go to this link to open the AWS EC2 console
2. (Optional) Click on the dropdown in the top right corner that specifies a location and choose a location. This will determine where you want your virtual server to

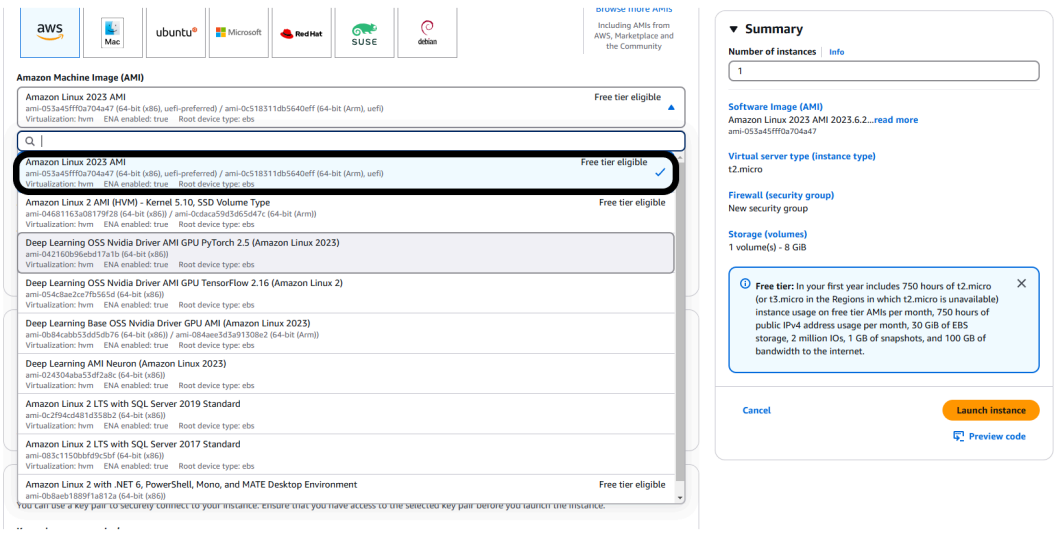
be stored. For me, I will choose the server closest to me which is “United States (N. Virginia)”.

3. Click “Launch instance”
4. Provide a descriptive name for your EC2 instance
5. Under Application and OS Images (Amazon Machine Image)
    a. Choose your preferred Operating System (I chose Amazon Linux)
    b. Select your preferred Amazon Machine Image (AMI) (I chose Amazon Linux 2023 AMI as it’s part of the free tier)
6. Under Instance type, select an instance type that best suits your needs (I chose t2.micro as it’s part of the free tier)
7. Key pair (login)
    a. Select “Create new key pair”
    b. Enter your key pair name
    c. Select .pem if you are on Linux/macOS, otherwise select .ppk if you are on Windows
    d. Select “Create key pair”
    e. Download the key to somewhere secure so you can access your EC2 instance with the key later
8. You can change any values under Network settings and Configure storage, but it is not necessary. (I will not be changing any values.)
9. Preview your summary to make sure everything is as you selected and select “Launch instance”

Now you have successfully launched your EC2 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 following the simple steps below.
Name and tags Info
Name
Demo Server
Add additional tags
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
Search our full catalog including 1000s of application and OS images
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE Linux
Debian
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Amazon Machine Image (AMI)
Amazon Linux 2023 AMI
ami-053a45fff0a704a47 (64-bit (x86), uefi-preferred) / ami-0c518311db5640eff (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs
Free tier eligible
Description
Amazon Linux 2023 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.
Amazon Linux 2023 AMI 2023.6.20250211.0 x86_64 HVM kernel-6.1
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...read more
ami-053a45fff0a704a47
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, 750 hours of public IPv4 address usage 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
Preview code


Amazon Machine Image (AMI)
Amazon Linux 2023 AMI
ami-053a45fff0a704a47 (64-bit (x86), uefi-preferred) / ami-0c518311db5640eff (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs
Free tier eligible
Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type
ami-04681163a08179f28 (64-bit (x86)) / ami-0cdaca59d3d65d47c (64-bit (Arm))
Deep Learning OSS Nvidia Driver AMI GPU PyTorch 2.5 (Amazon Linux 2023)
ami-042160b96ebd17a1b (64-bit (x86))
Deep Learning OSS Nvidia Driver AMI GPU TensorFlow 2.16 (Amazon Linux 2)
ami-054c8ae2ce7fb565d (64-bit (x86))
Deep Learning Base OSS Nvidia Driver GPU AMI (Amazon Linux 2023)
ami-0b84cabb53dd5db76 (64-bit (x86)) / ami-084aee3d3a91308e2 (64-bit (Arm))
Deep Learning AMI Neuron (Amazon Linux 2023)
ami-024304aba53df2a8c (64-bit (x86))
Amazon Linux 2 LTS with SQL Server 2019 Standard
ami-0c2f94cd481d358b2 (64-bit (x86))
Amazon Linux 2 LTS with SQL Server 2017 Standard
ami-085c1150bbfd9c3bf (64-bit (x86))
Amazon Linux 2 with .NET 6, PowerShell, Mono, and MATE Desktop Environment
ami-0b8aeb1889f1a812a (64-bit (x86))
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...read more
ami-053a45fff0a704a47
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, 750 hours of public IPv4 address usage 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
Preview code

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 Ubuntu Pro base pricing: 0.0134 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour On-Demand RHEL base pricing: 0.026 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

All generations

Compare instance types

Get advice on instance type selection...

t2.nano
Family: t2 1 vCPU 0.5 GiB Memory Current generation: true On-Demand Linux base pricing: 0.0058 USD per Hour
On-Demand SUSE base pricing: 0.0058 USD per Hour On-Demand Windows base pricing: 0.0081 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0076 USD per Hour

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 Ubuntu Pro base pricing: 0.0134 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour On-Demand RHEL base pricing: 0.026 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

t2.small
Family: t2 1 vCPU 2 GiB Memory Current generation: true On-Demand Windows base pricing: 0.032 USD per Hour
On-Demand Linux base pricing: 0.023 USD per Hour On-Demand RHEL base pricing: 0.0376 USD per Hour
On-Demand SUSE base pricing: 0.053 USD per Hour On-Demand Ubuntu Pro base pricing: 0.025 USD per Hour

ou launch the instance.

te new key pair

Edit

vpc-01ab700b9e5d16da1

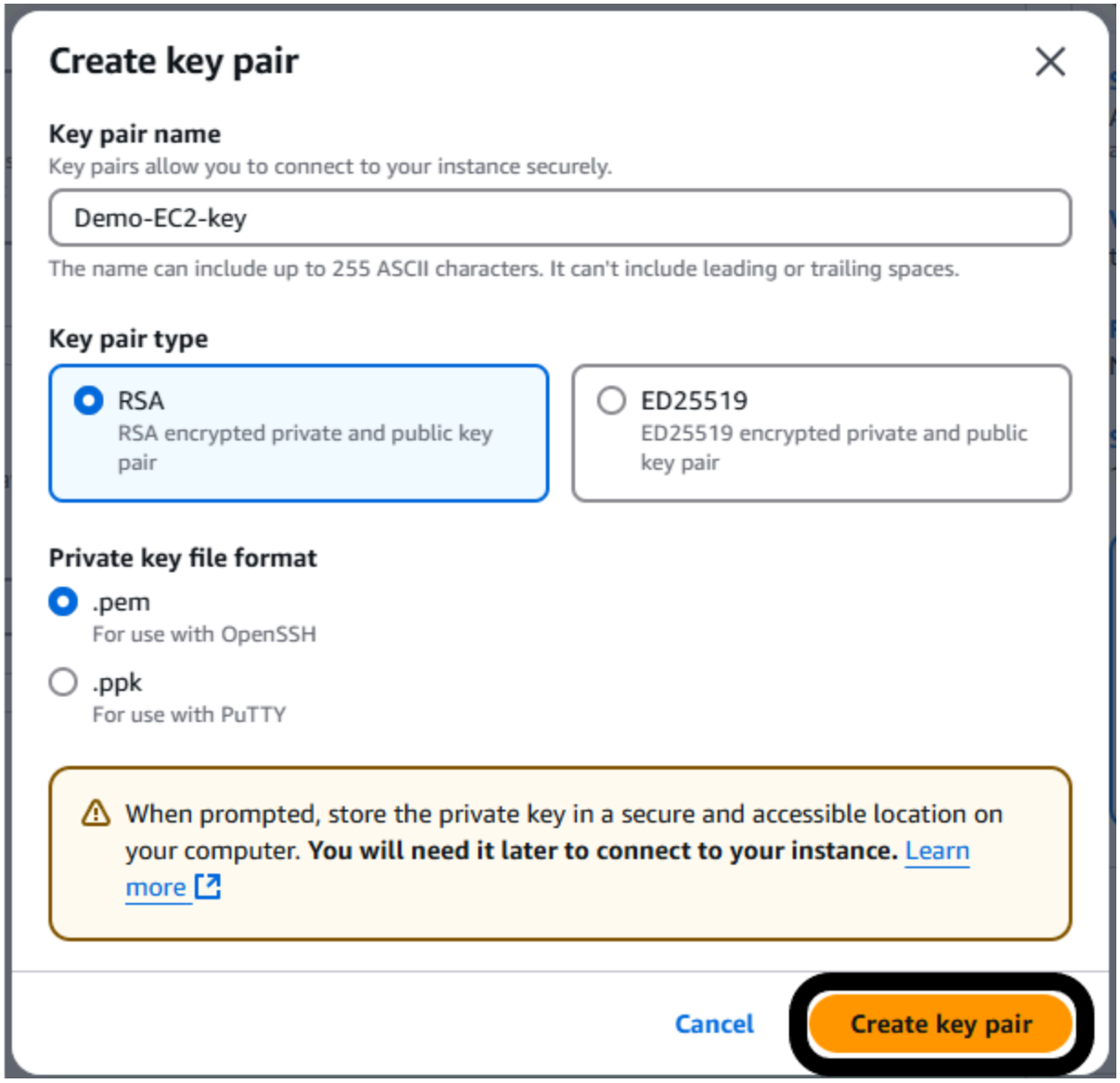
Key pair (login) Info

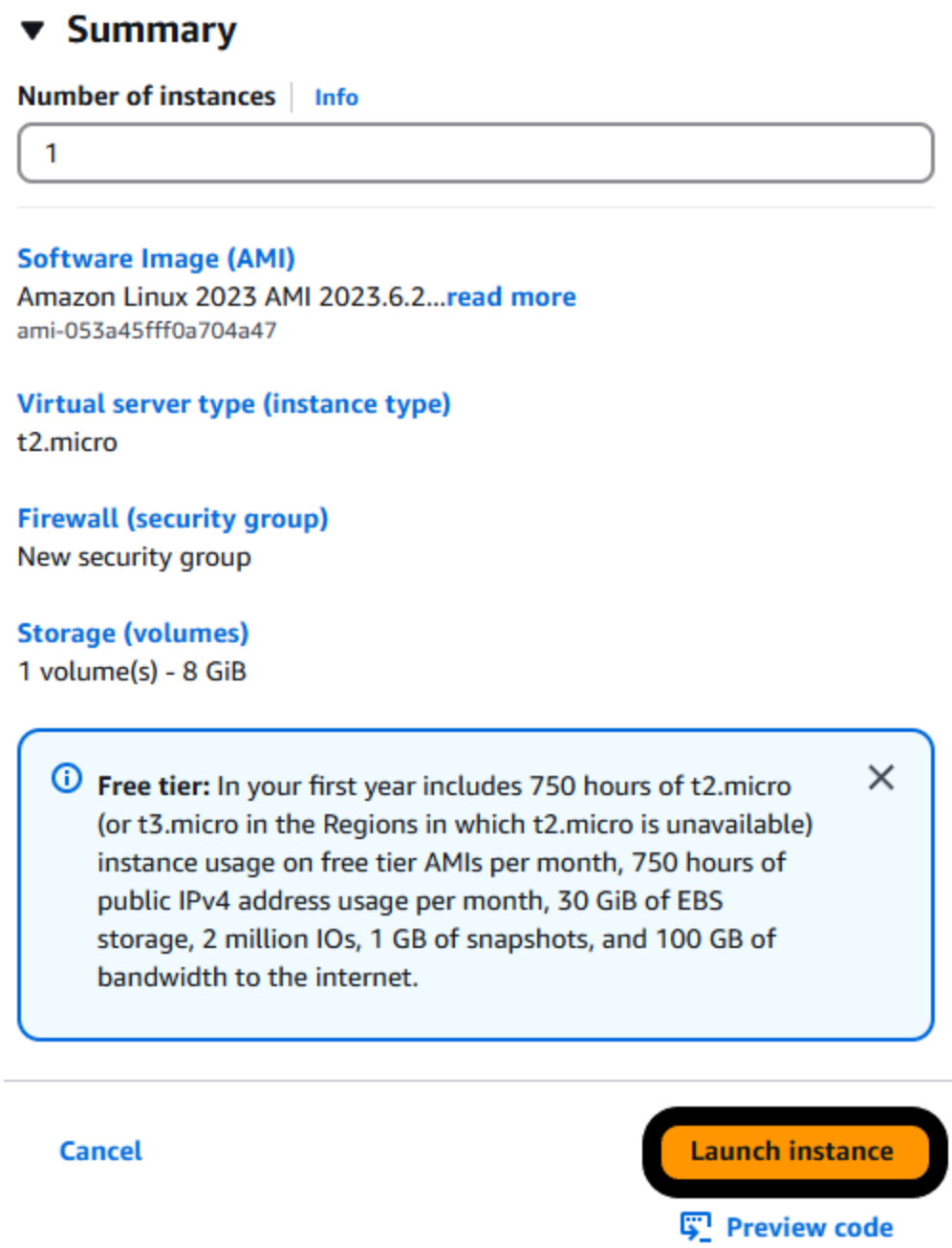
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

Select

Create new key pair


Create key pair
Key pair name
Key pairs allow you to connect to your instance securely.
Demo-EC2-key
The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.
Key pair type
RSA
RSA encrypted private and public key pair
ED25519
ED25519 encrypted private and public key pair
Private key file format
.pem
For use with OpenSSH
.ppk
For use with PuTTY
When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. Learn more
Cancel
Create key pair



Connecting to your EC2 instance

Prereq

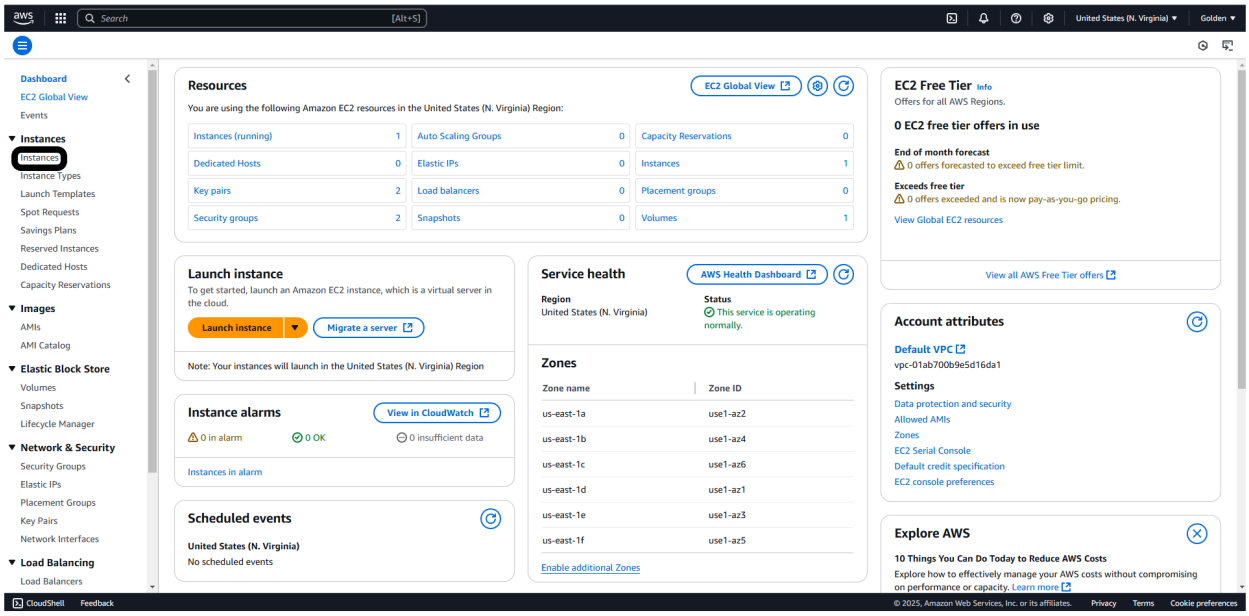
- Launched EC2 instance
- You should be able to use the ssh command in your terminal

Steps

1. Go to your AWS console
2. Click “Instances” located on the left-hand side of the page

3. Click on your EC2 instance
4. Click “Connect”
5. Select “SSH client”
6. Open your terminal
7. Go into your search bar and open Command Prompt
8. Go to the directory where you stored your EC2 instance key (In my case, I had it in downloads)
9. Paste the command shown in the SSH client, highlighted in the picture below into your terminal
10. Enter “yes” into your terminal when prompted “Are you sure you want to continue connecting”

You have now successfully connected to your EC2 instance!



Instance summary for i-04a33d631f26086d0 (Demo Server) Info

Connect | Instance state | Actions

Updated less than a minute ago

Instance ID
i-04a33d631f26086d0

IPv6 address
–

Hostname type
IP name: ip-172-31-85-61.ec2.internal

Answer private resource DNS name
IPv4 (A)

Auto-assigned IP address
3.84.119.136 [Public IP]

IAM Role
–

IMDSv2
Required

Operator
–

Public IPv4 address
3.84.119.136 | open address

Instance state
Running

Private IP DNS name (IPv4 only)
ip-172-31-85-61.ec2.internal

Instance type
t2.micro

VPC ID
vpc-01ab700b9e5d16da1

Subnet ID
subnet-0e433d1113e477dc6

Instance ARN
arn:aws:ec2:us-east-1:354918380194:instance/i-04a33d631f26086d0

Private IPv4 addresses
172.31.85.61

Public IPv4 DNS
ec2-3-84-119-136.compute-1.amazonaws.com | open address

Elastic IP addresses
–

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more

Auto Scaling Group name
–

Managed
false

Connect to instance Info

Connect to your instance i-04a33d631f26086d0 (Demo Server) using any of these options

EC2 Instance Connect | Session Manager | SSH client | EC2 serial console

Instance ID
i-04a33d631f26086d0 (Demo Server)

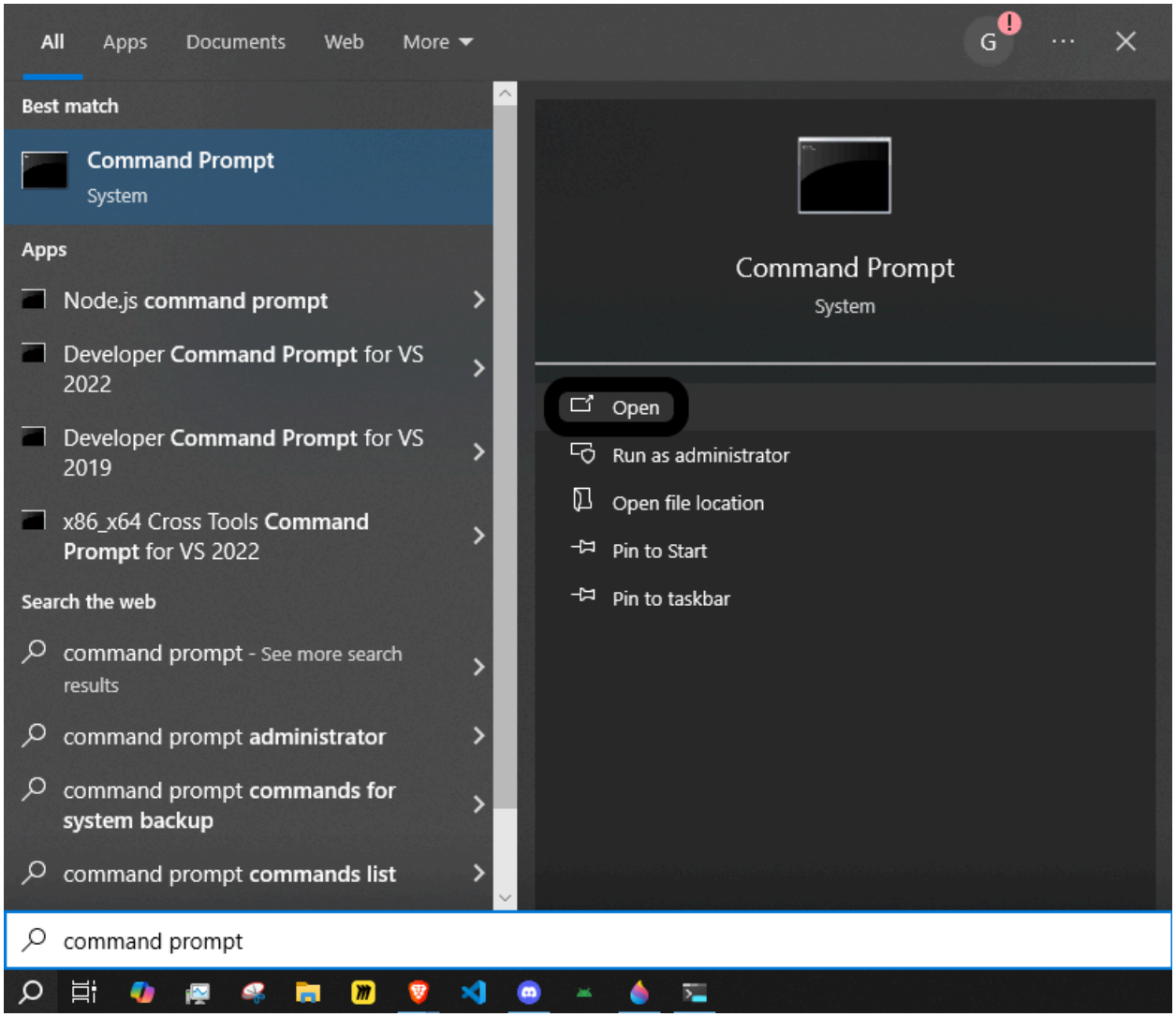
1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is Demo-EC2-key.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
   chmod 400 "Demo-EC2-key.pem"
4. Connect to your instance using its Public DNS:
   ec2-3-84-119-136.compute-1.amazonaws.com

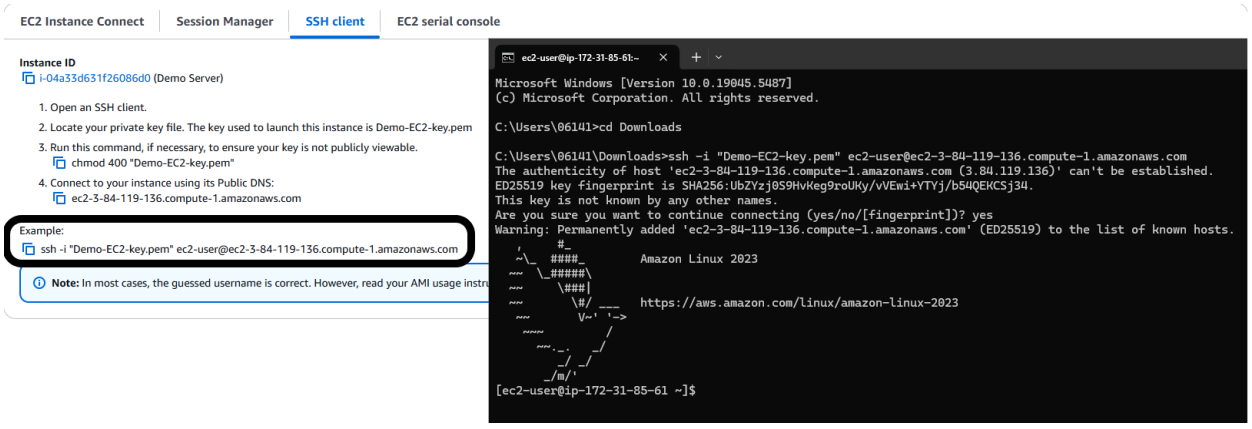
Example:

ssh -i "Demo-EC2-key.pem" ec2-user@ec2-3-84-119-136.compute-1.amazonaws.com

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel





Terminating EC2 Instance

Prereq: Must have an EC2 instance running

1. Go to your AWS console

2. Click “Instances” located on the left-hand side of the page
3. Select the instance(s) you want to terminate by selecting the checkbox next to the instances
4. Click on the “Instance State” dropdown and select “Terminate (delete) instance”

Terminating your EC2 instance(s) would ensure that you would not incur any more charges. Alternatively, you stop