

# Terraform Assignment - 1

You have been asked to:

- Create an EC2 service in the default subnet in the ohio region

[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 following the simple steps below.

### Name and tags [Info](#)

Name

Terraform-Assignment

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

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

Recents

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Li

SUSE

Q

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type

Free tier eligible

ami-0f58b397bc5c1f2e8 (64-bit (x86)) / ami-0dda4ba9a42839a4b (64-bit (Arm))

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

Description

Canonical, Ubuntu, 24.04 LTS, amd64 noble image build on 2024-04-23

Architecture

AMI ID

64-bit (x86)

ami-0f58b397bc5c1f2e8

Verified provider

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

### Instance type

t2.medium

Family: t2 2 vCPU 4 GiB Memory Current generation: true  
On-Demand Linux base pricing: 0.0496 USD per Hour  
On-Demand Windows base pricing: 0.0676 USD per Hour  
On-Demand RHEL base pricing: 0.1096 USD per Hour  
On-Demand SUSE base pricing: 0.1496 USD per Hour

☒ 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*

DevOps

[Create new key pair](#)

## ▼ Network settings [Info](#)

[Edit](#)

Network | [Info](#)

vpc-08bde8db1e7da9b9a | Default VPC

Subnet | [Info](#)

No preference (Default subnet in any availability zone)

Auto-assign public IP | [Info](#)

Enable

Additional charges apply when outside of free tier allowance

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

Common security groups [Info](#)

Select security groups

default sg-04d9bdfee395aade5 X

VPC: vpc-08bde8db1e7da9b9a

[Compare security group rules](#)

Security groups that you add or remove here will be added to or removed from all your network interfaces.

▼ **Summary**

Number of instances [Info](#)

1

**Software Image (AMI)**  
Canonical, Ubuntu, 24.04 LTS, ...[read more](#)  
ami-0f58b397bc5c1f2e8

**Virtual server type (instance type)**  
t2.medium

**Firewall (security group)**  
default

**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

×

Cancel **Launch instance**

[Review commands](#)

Instances (1) <a href="#">Info</a>							Launch instances
<input type="text"/> Find Instance by attribute or tag (case-sensitive)							All states
<input type="checkbox"/>	Name ↗	Instance ID	Instance state	Instance type	Status check	Alarm status	
<input type="checkbox"/>	Terraform-Assignment	i-0d81c556e43b0246d	Running	t2.medium	2/2 checks passed	<a href="#">View alarms</a>	+

```
ubuntu@ip-172-31-34-196:~$ sudo apt-get update -y
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease [256 kB]
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 Packages [1401 kB]
Get:5 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/main Translation-en [513 kB]
```

**i-0d81c556e43b0246d (Terraform-Assignment)**

PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

## Installing Terraform

```
wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
```

```
echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
```

```
sudo apt update && sudo apt install terraform
```

```
ubuntu@ip-172-31-34-196:~$ terraform --version
Terraform v1.8.4
on linux_amd64
ubuntu@ip-172-31-34-196:~$
```

**i-0d81c556e43b0246d (Terraform-Assignment)**

PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

## Install AWS CLI

```
ubuntu@ip-172-31-34-196:~$ sudo apt install unzip -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  zip
The following NEW packages will be installed:
  unzip
```

**i-0d81c556e43b0246d (Terraform-Assignment)**

PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

```
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
```

```
unzip awscliv2.zip
```

```
sudo ./aws/install
```

```
ubuntu@ip-172-31-34-196:~$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 57.8M  100 57.8M    0     0  121M      0  --:--:-- --:--:-- --:--:-- 121M
ubuntu@ip-172-31-34-196:~$
```

**i-0d81c556e43b0246d (Terraform-Assignment)**

PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

```
ubuntu@ip-172-31-34-196:~$ ls
aws  awscliv2.zip
ubuntu@ip-172-31-34-196:~$
```

**i-0d81c556e43b0246d (Terraform-Assignment)**

PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

```
ubuntu@ip-172-31-34-196:~$ ls
aws  awscliv2.zip
ubuntu@ip-172-31-34-196:~$ sudo ./aws/install
You can now run: /usr/local/bin/aws --version
ubuntu@ip-172-31-34-196:~$ /usr/local/bin/aws --version
aws-cli/2.15.62 Python/3.11.8 Linux/6.8.0-1008-aws exe/x86_64.ubuntu.24
ubuntu@ip-172-31-34-196:~$
```

**i-0d81c556e43b0246d (Terraform-Assignment)**

PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

## Setting AWS CLI Config

```
ubuntu@ip-172-31-34-196:~$ aws configure
AWS Access Key ID [None]: AKIARPFYXAWSGACDTUNL
AWS Secret Access Key [None]: sxRmeNNRZdhMey2yu0J0zYY+O4ZKtChNXXXxE3Ef
Default region name [None]:
Default output format [None]:
ubuntu@ip-172-31-34-196:~$
```

**i-0d81c556e43b0246d (Terraform-Assignment)**

PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

## Verify AWS Config setup

```
ubuntu@ip-172-31-34-196:~$ aws sts get-caller-identity
{
  "UserId": "101304436132",
  "Account": "101304436132",
  "Arn": "arn:aws:iam::101304436132:root"
}
ubuntu@ip-172-31-34-196:~$
```

**i-0d81c556e43b0246d (Terraform-Assignment)**

PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

## Writing Terraform script to launch an EC2 in Ohio region

```
provider "aws" {
  region = "us-east-2"
}
```

```
resource "aws_instance" "instance1" {
  ami = "ami-09040d770ffe2224f"
```

```
instance_type = "t2.micro"

tags = {
    Name = "terraform-instance"
}
}
```

```
ubuntu@ip-172-31-34-196:~/assignment1$ vi main.tf
ubuntu@ip-172-31-34-196:~/assignment1$ cat main.tf
provider "aws" {
    region = "us-east-2"
}

resource "aws_instance" "instance1" {
    ami = "ami-09040d770ffe2224f"
    instance_type = "t2.micro"
    tags = {
        Name = "terraform-instance"
    }
}
```

ubuntu@ip-172-31-34-196:~/assignment1\$

**i-0d81c556e43b0246d (Terraform-Assignment)**

PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

```
ubuntu@ip-172-31-34-196:~/assignment1$ terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.52.0...

```

**i-0d81c556e43b0246d (Terraform-Assignment)**

PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

```
ubuntu@ip-172-31-34-196:~/assignment1$ terraform plan

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

# aws_instance.instance1 will be created
+ resource "aws_instance" "instance1" {
    + ami = "ami-09040d770ffe2224f"
}
```

**i-0d81c556e43b0246d (Terraform-Assignment)**

PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

```
ubuntu@ip-172-31-34-196:~/assignment1$ terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

i-0d81c556e43b0246d (Terraform-Assignment)
PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

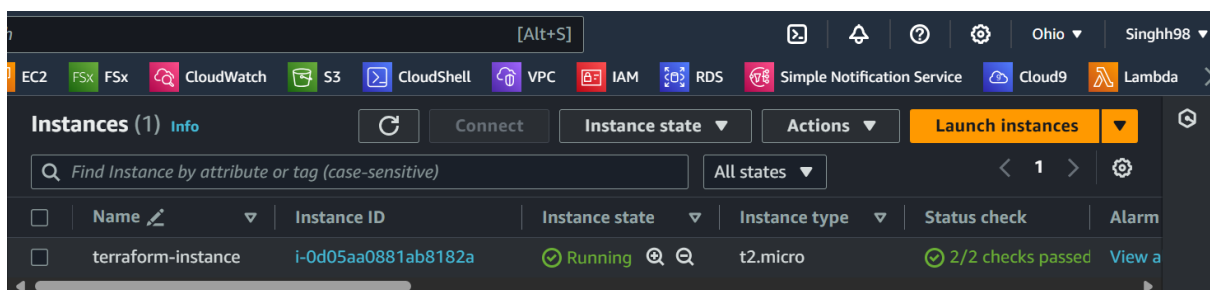
Enter a value: yes

aws_instance.instance1: Creating...
aws_instance.instance1: Still creating... [10s elapsed]
aws_instance.instance1: Still creating... [20s elapsed]
aws_instance.instance1: Still creating... [30s elapsed]
aws_instance.instance1: Creation complete after 35s [id=i-0d05aa0881ab8182a]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-34-196:~/assignment1$
```

i-0d81c556e43b0246d (Terraform-Assignment)  
PublicIPs: 13.126.139.48 PrivateIPs: 172.31.34.196

Now we could see that the instance has been launched in ohio region.



\*\*\*\*\* THE END \*\*\*\*\*