AWS re:Start LAB

# Build Your VPC and Launch a Web Server

WEEK 4

aws re/start

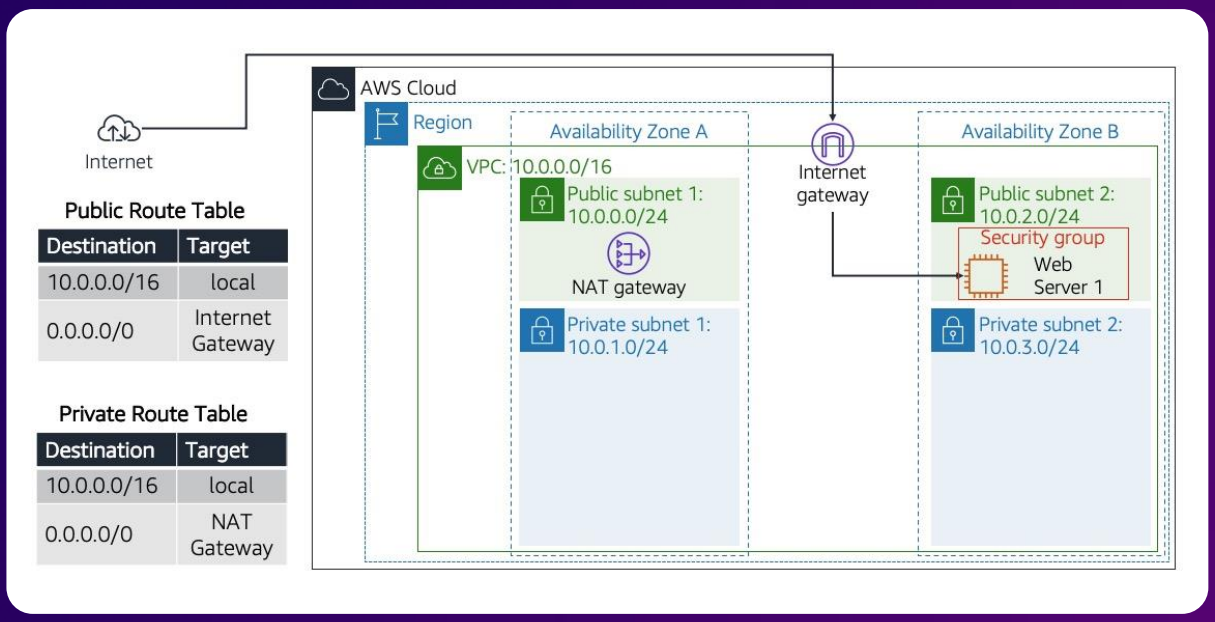
# Overview

## Customer scenario

In this lab, you use Amazon Virtual Private Cloud (VPC) to create your own VPC and add additional components to produce a customized network for a Fortune 100 customer. You also create security groups for your EC2 instance. You then configure and customize an EC2 instance to run a web server and launch it into the VPC that looks like the following customer diagram.

## Customer diagram

The customer is requesting the build of this architecture to launch their web server successfully.



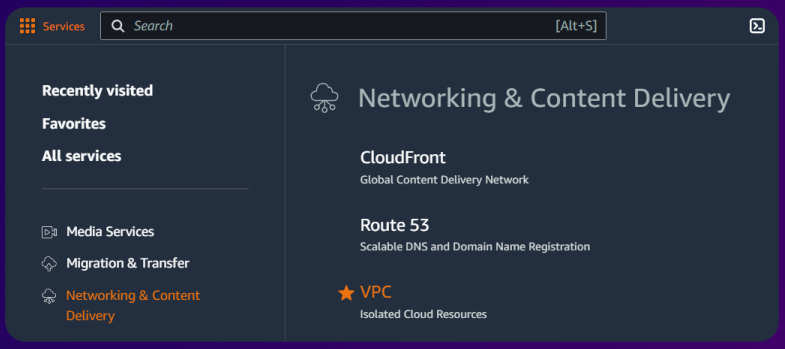

aws re/start

# Task 1

# Create your VPC

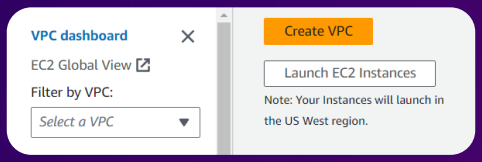
## Step 1: Access the AWS Management Console

Open the AWS Management Console, and select VPC.




## Step 2: Creating the VPC

In the **Amazon VPC** dashboard, choose the Create VPC button to launch the VPC wizard.




aws re/start

# Task 1

# Create your VPC

## Step 3: Set up your VPC

Once in the VPC wizard, use the following parameters to configure the VPC settings.

**VPC settings**

Resources to create Info
Create only the VPC resource or the VPC and other networking resources.
- ○ VPC only
- ◉ VPC and more

Name tag auto-generation Info
Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.
- ☐ Auto-generate

IPv4 CIDR block Info
Determine the starting IP and the size of your VPC using CIDR notation.
10.0.0.0/16 — 65,536 IPs
CIDR block size must be between /16 and /28.

IPv6 CIDR block Info
- ◉ No IPv6 CIDR block
- ○ Amazon-provided IPv6 CIDR block

Tenancy Info
Default

Number of Availability Zones (AZs) Info
Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.
1 | 2 | 3

▶ Customize AZs

Number of public subnets Info
The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.
0 | 1

Number of private subnets Info
The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.
0 | 1 | 2

▼ Customize subnets CIDR blocks

Public subnet CIDR block in us-west-2a
10.0.0.0/24 — 256 IPs

Private subnet CIDR block in us-west-2a
10.0.1.0/24 — 256 IPs

NAT gateways ($) Info
Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway
None | In 1 AZ | 1 per AZ

VPC endpoints Info
Endpoints can help reduce NAT gateway charges and improve security by accessing S3 directly from the VPC. By default, full access policy is used. You can customize this policy at any time.
None | S3 Gateway

DNS options Info
- ☑ Enable DNS hostnames
- ☑ Enable DNS resolution

▶ Additional tags

**Preview**

VPC Show details
Your AWS virtual network
Lab VPC

Subnets (2)
Subnets within this VPC
us-west-2a
- Public Subnet 1
- Private Subnet 1

Route tables (2)
Route network traffic to resources
- Public Route Table
- Private Route Table

Network connections (2)
Connections to other networks
- Internet gateway withou
- NAT gateway without N

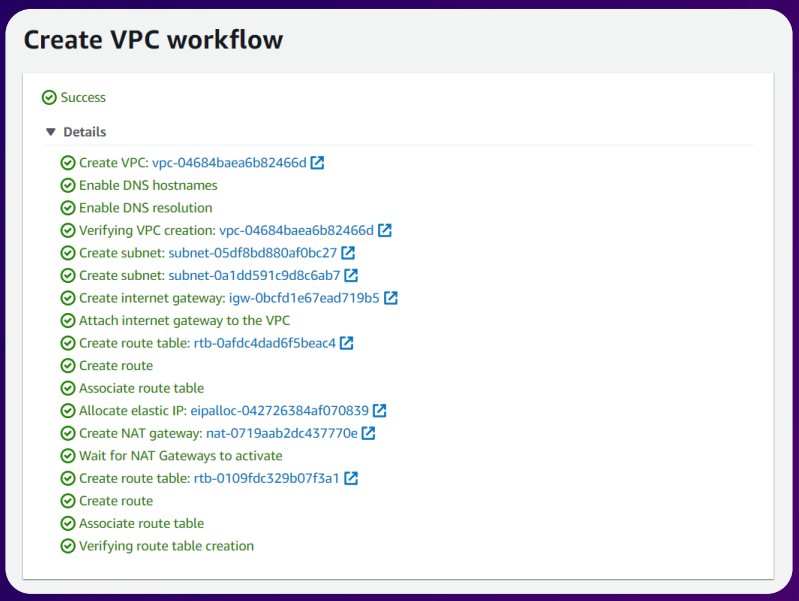
aws re/start

# Task 1

# Create your VPC

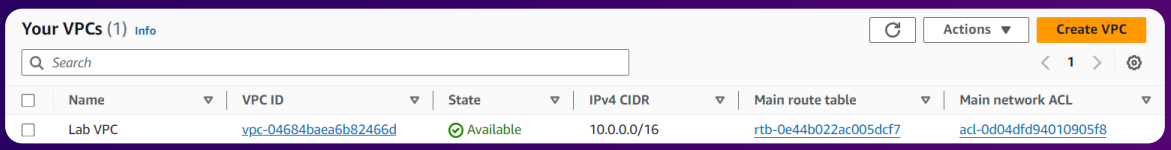
## Step 4: Check the Create VPC workflow

Once you have successfully created the VPC, you should see a Success message in the Create VPC workflow.




## Step 5: Review your VPC

Navigate to the Amazon VPC dashboard and select **Your VPCs** to verify that your VPC is available. You should see your VPC listed.



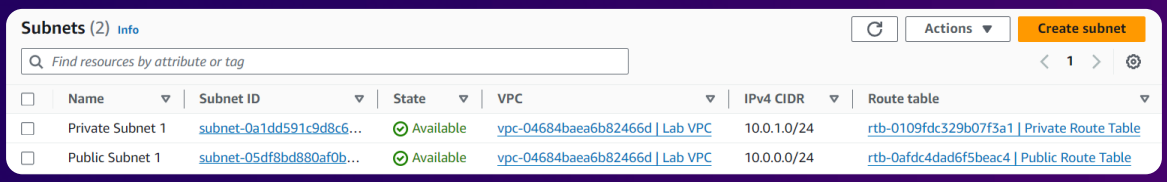

aws re/start

# Task 2

# Create additional subnets

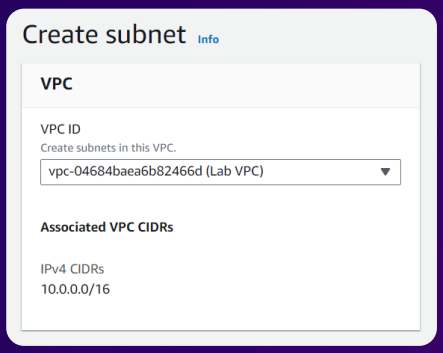
## Step 1: Creating more subnets

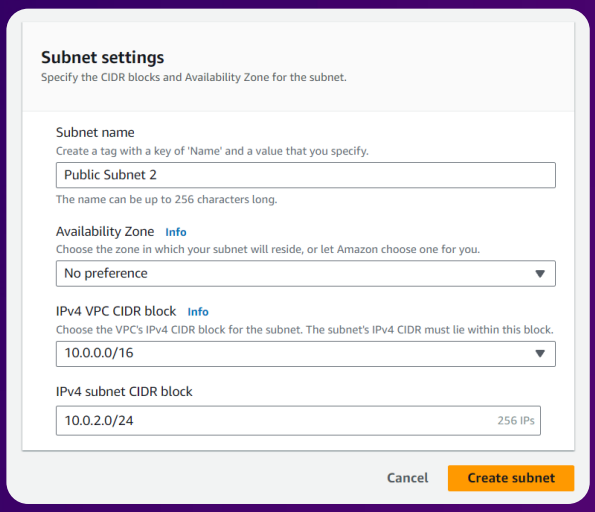
Navigate to the **Subnets** section and select Create subnet.




## Step 2: Create a second public subnet

Use the following parameters to configure the subnet settings.



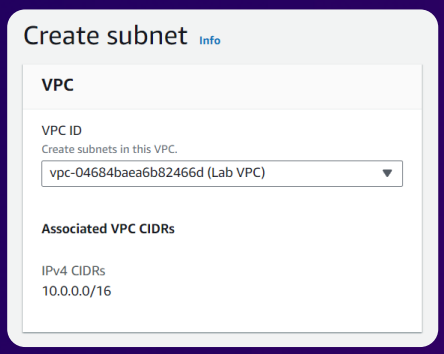

aws re/start

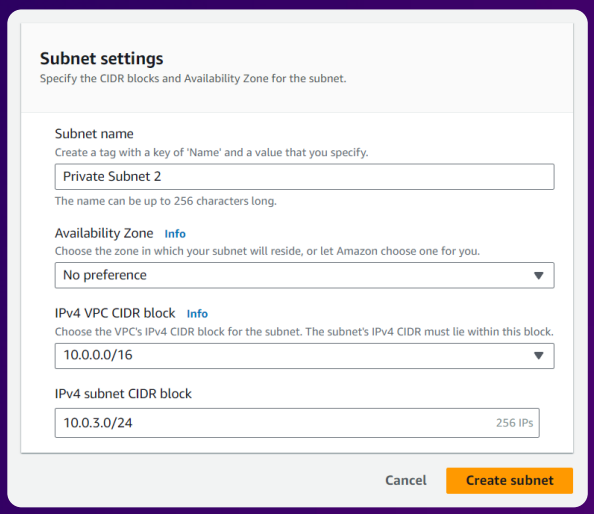
# Task 2

# Create additional subnets

## Step 3: Create a second private subnet

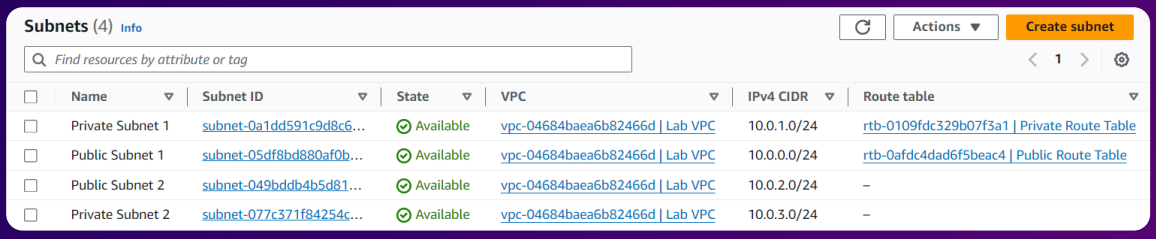
Use the following parameters to configure the subnet settings.




## Step 4: Review the new subnets

Once you have created the subnets, navigate to the **Subnets** section to verify that your new subnets are available.



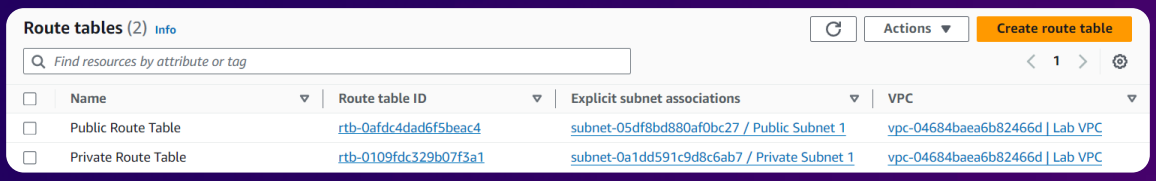

| Name | Subnet ID | State | VPC | IPv4 CIDR | Route table |
|---|---|---|---|---|---|
| Private Subnet 1 | subnet-0a1dd591c9d8c6... | Available | vpc-04684baea6b82466d \| Lab VPC | 10.0.1.0/24 | rtb-0109fdc329b07f3a1 \| Private Route Table |
| Public Subnet 1 | subnet-05df8bd880af0b... | Available | vpc-04684baea6b82466d \| Lab VPC | 10.0.0.0/24 | rtb-0afdc4dad6f5beac4 \| Public Route Table |
| Public Subnet 2 | subnet-049bddb4b5d81... | Available | vpc-04684baea6b82466d \| Lab VPC | 10.0.2.0/24 | – |
| Private Subnet 2 | subnet-077c371f84254c... | Available | vpc-04684baea6b82466d \| Lab VPC | 10.0.3.0/24 | – |

# Task 3

# Associate the subnets and add routes

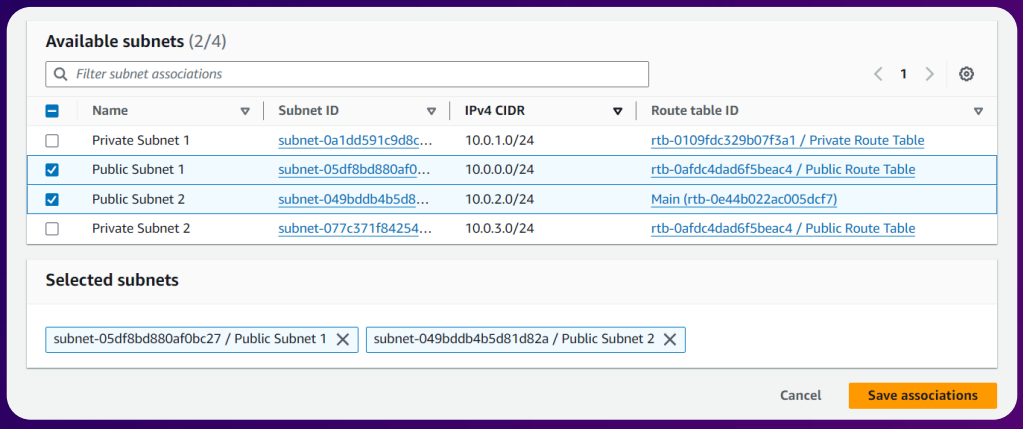
## Step 1: Associate the new subnets

Navigate to the **Route Tables** section. You should see that each route table is currently associated with one subnet.




## Step 2: Associate Public Subnet 2

In the **Subnet associations** tab, associate the Public subnet 2 to the Public route table and click Save associations.



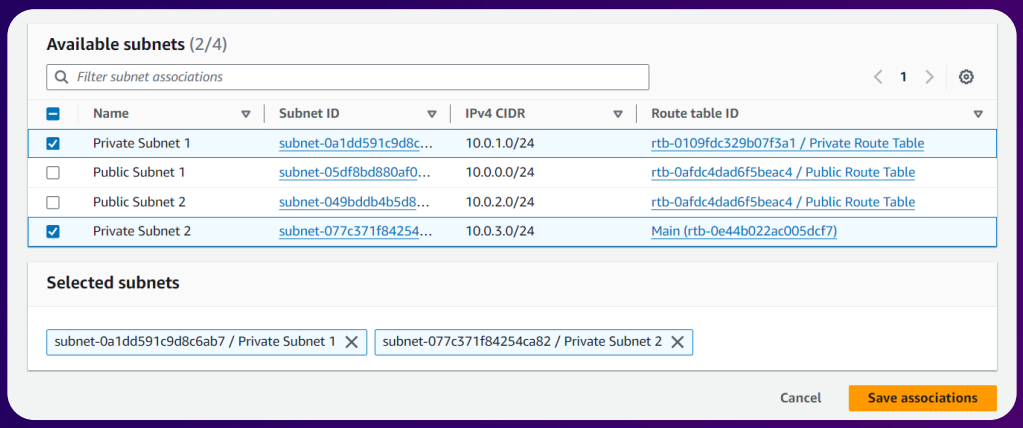

aws re/start

# Task 3

# Associate the subnets and add routes

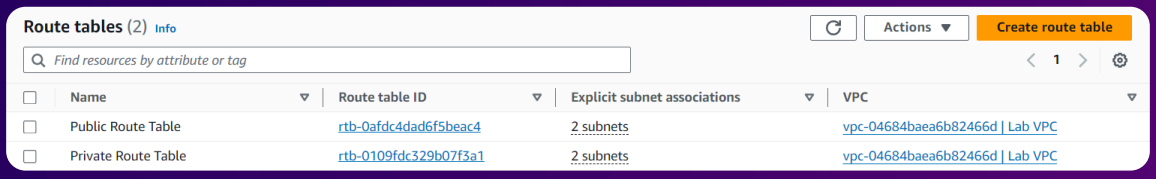
## Step 3: Associate Private Subnet 2

In the **Subnet associations** tab, associate the Private subnet 2 to the Private route table and click Save associations.




## Step 4: Review associations

In the **Route Tables** section, you should see that each route table is now associated with two subnets.



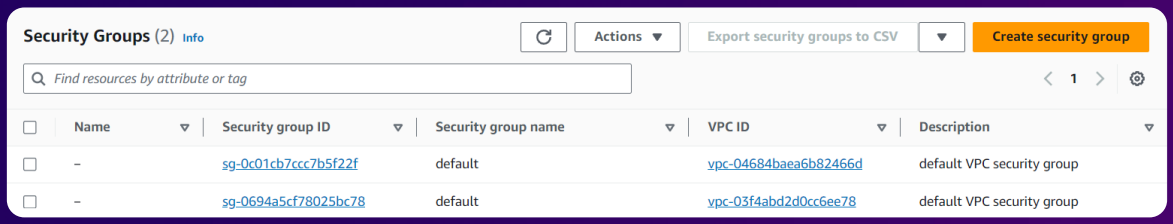

aws re/start

# Task 4

# Create a VPC security group

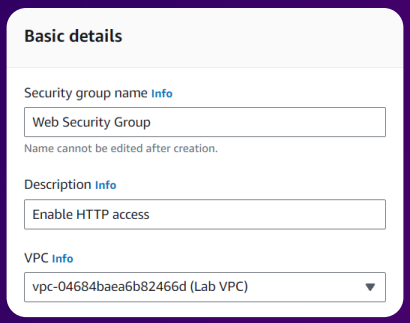
## Step 1: Creating a security group

Navigate to the **Security Groups** section and select Create security group.




## Step 2: Set up the security group

Use the following parameters to configure the security group basic details.



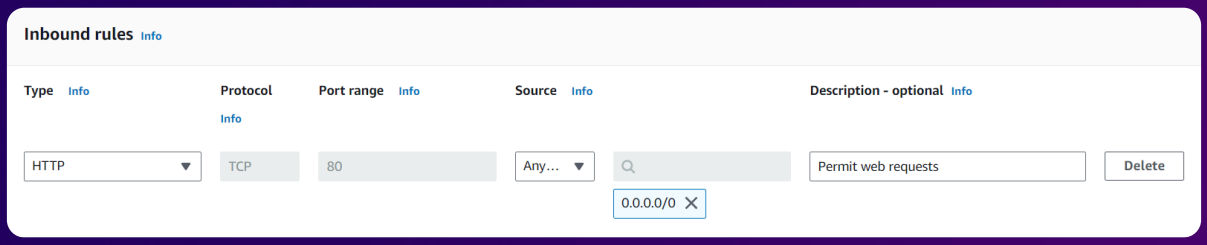

aws re/start

# Task 4

# Create a VPC security group

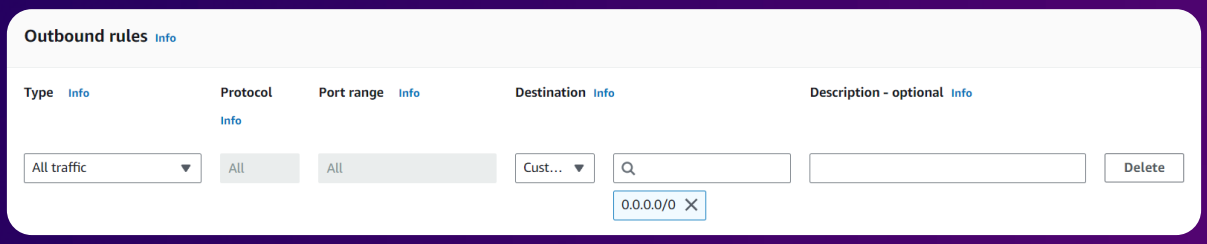
## Step 3: Set up security group inbound rules

Configure the following inbound rules for the security group to permit incoming HTTP requests.




## Step 4: Set up security group outbound rules

Configure the following outbound rules for the security group to allow all types of outgoing traffic.



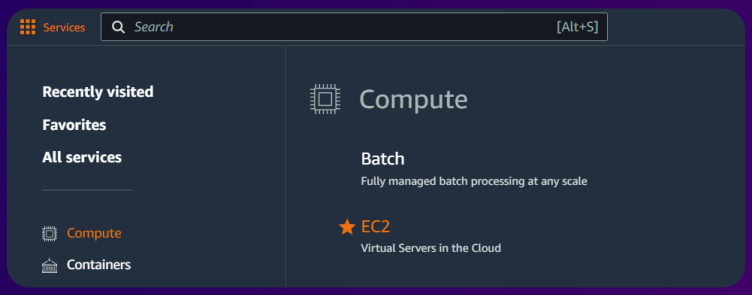

aws re/start

# Task 5

# Launch a web server instance

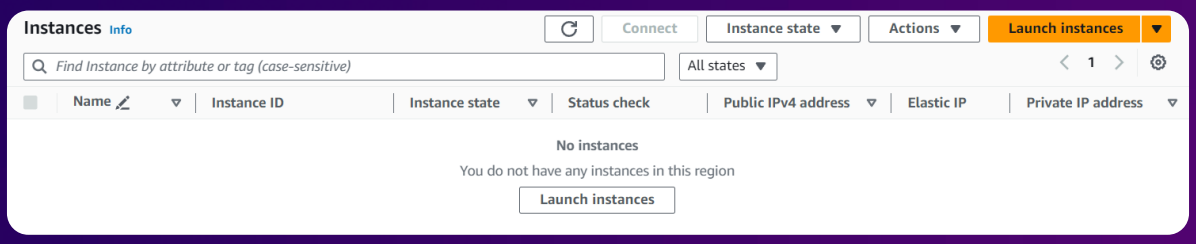
## Step 1: Access the EC2 Management Console

Open the AWS Management Console, and select EC2.




## Step 2: Launch instance

Navigate to the **Instances** section and select Launch instances.




aws re/start

# Task 5

# Launch a web server instance

## Step 3: Set up the instance

Use the following parameters to configure the instance settings.

**Name and tags** Info

Name

Web Server 1

**Instance type** Info | Get advice

Instance type

t3.micro
Family: t3 2 vCPU 1 GiB Memory
Current generation: true
On-Demand SUSE base pricing: 0.0104 USD per Hour
On-Demand Windows base pricing: 0.0196 USD per Hour
On-Demand RHEL base pricing: 0.0704 USD per Hour
On-Demand Linux base pricing: 0.0104 USD per Hour

**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*

vockey

Create new key pair

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*

Web Security Group sg-02f2a00804bb40668
VPC: vpc-04684baea6b82466d

Compare security group rules

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

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type Free tier eligible
ami-089313d40efd067a9 (64-bit (x86)) / ami-08d93ffc25e0a3513 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2 Kernel 5.10 AMI 2.0.20240329.0 x86_64 HVM gp2

Architecture

64-bit (x86)

AMI ID

ami-089313d40efd067a9 Verified provider

**Network settings** Info

VPC - *required* Info

vpc-04684baea6b82466d (Lab VPC)
10.0.0.0/16

Subnet Info

subnet-049bddb4b5d81d82a Public Subnet 2
VPC: vpc-04684baea6b82466d Owner: 471112886328
Availability Zone: us-west-2a IP addresses available: 251 CIDR: 10.0.2.0/24)

Auto-assign public IP Info

Enable

User data - *optional* Info

Upload a file with your user data or enter it in the field.

```
#!/bin/bash
#Install Apache Web Server and PHP
yum install -y httpd mysql php
#Download Lab files
wget https://aws-tc-largeobjects.s3.us-west-
2.amazonaws.com/CUR-TF-100-RESTRT-1/267-lab-NF-
build-vpc-web-server/s3/lab-app.zip
unzip lab-app.zip -d /var/www/html/
#Turn on web server
chkconfig httpd on
service httpd start
```

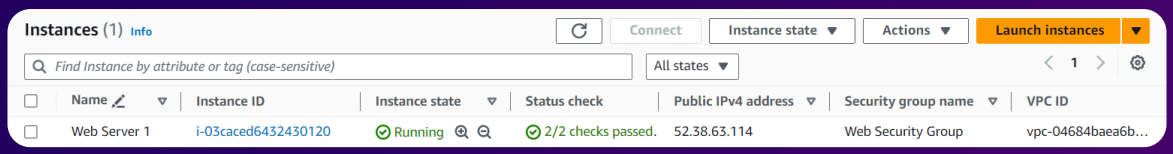
aws re/start

# Task 5

# Launch a web server instance

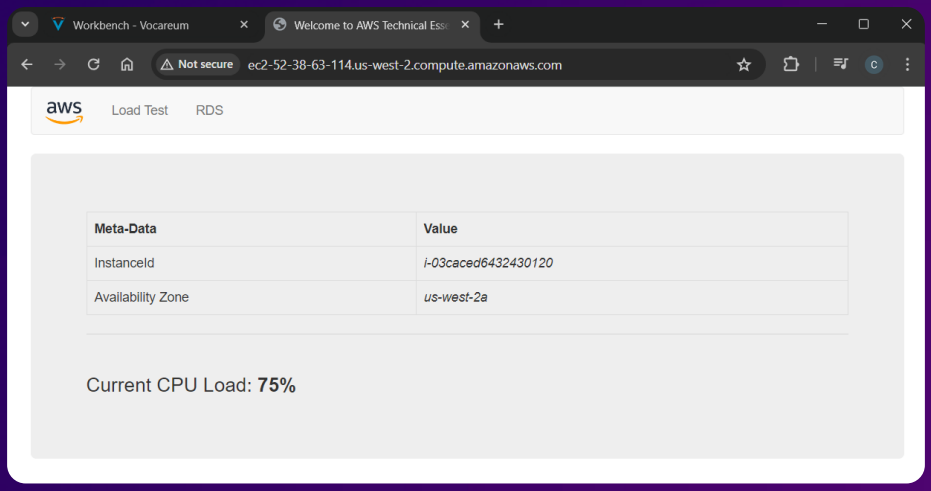
## Step 4: Review the instance

Once you have created the instance, navigate to the **Instances** section to verify that your instance is now running.




## Step 5: Connect to the web server

Open a new browser tab and enter the Public IPv4 DNS of the instance to connect to the web server running on the instance.




aws re/start

# Conclusions

### VPC Launch Wizard
The VPC Launch Wizard simplifies the process of setting up AWS resources within a Virtual Private Cloud, enhancing ease of deployment and configuration.

### Additional Subnets
Creating additional subnets within a VPC allows for better organization of resources, improved network segmentation, and enhanced security through subnet-specific settings.

### Subnets and Route Table Associations
Associating subnets with route tables controls traffic flow between subnets and enables efficient routing based on network requirements.

### Security Groups
Security groups act as virtual firewalls, governing inbound and outbound traffic to instances based on defined rules, enhancing network security and access control.

### Web Server Instances
Deploying web server instances within AWS VPCs provides scalable and secure hosting solutions, leveraging VPC features like subnets, route tables, and security groups for robust web application deployments.

aws re/start

**Cristhian Becerra**

cristhian-becerra-espinoza

+51 951 634 354

cristhianbecerra99@gmail.com

Lima, Peru