

COS20019

Cloud Computing Architecture

Week 3 – ACF Lab 2:
Build a VPC and launch a Web Server

Truong Ngoc Gia Hieu
105565520

Lab 2: Build your VPC and Launch a Web Server

A. Lab overview and objectives

In this lab, you will use Amazon Virtual Private Cloud (VPC) to create your own VPC and add additional components to produce a customized network. You will also create a security group. You will then configure and customize an EC2 instance to run a web server and you will launch the EC2 instance to run in a subnet in the VPC.

Amazon Virtual Private Cloud (Amazon VPC) enables you to launch Amazon Web Services (AWS) resources into a virtual network that you defined. This virtual network closely resembles a traditional network that you would operate in your own data center, with the benefits of using the scalable infrastructure of AWS. You can create a VPC that spans multiple Availability Zones.

After completing this lab, you should be able to do the following:

- Create a VPC.
- Create subnets.
- Configure a security group.
- Launch an EC2 instance into a VPC.

Duration

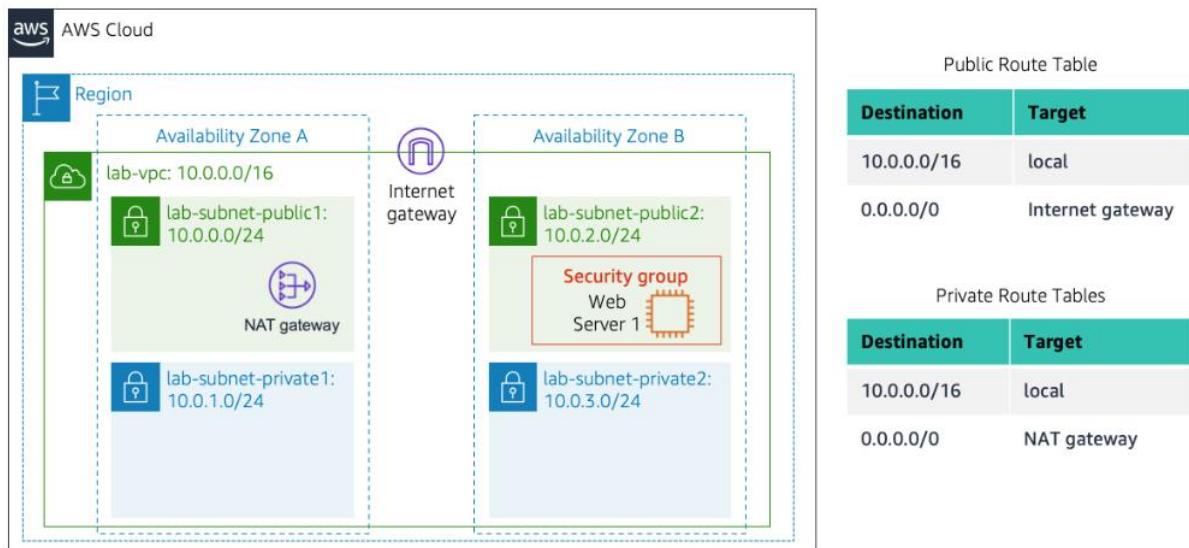
This lab takes approximately **30 minutes** to complete.

AWS service restrictions

In this lab environment, access to AWS services and service actions might be restricted to the ones that are needed to complete the lab instructions. You might encounter errors if you attempt to access other services or perform actions beyond the ones that are described in this lab.

Scenario

In this lab you build the following infrastructure:



Accessing the AWS Management Console

- At the top of these instructions, choose **Start Lab**.
 - The lab session starts.
 - A timer displays at the top of the page and shows the time remaining in the session.

Tip: To refresh the session length at any time, choose **Start Lab** again before the timer reaches 0:00.

 - Before you continue, wait until the circle icon to the right of the [AWS](#) link in the upper-left corner turns green.

[AWS](#)

Figure 1: AWS Console Home Activated

- To connect to the AWS Management Console, choose the [AWS](#) link in the upper-left corner.
 - A new browser tab opens and connects you to the console.

Tip: If a new browser tab does not open, a banner or icon is usually at the top of your browser with the message that your browser is preventing the site from opening pop-up windows. Choose the banner or icon, and then choose **Allow pop-ups**.
- Arrange the AWS Management Console tab so that it displays along side these instructions. Ideally, you will be able to see both browser tabs at the same time, to make it easier to follow the lab steps.

Getting Credit for your work

At the end of this lab you will be instructed to submit the lab to receive a score based on your progress.

Tip: The script that checks your work may only award points if you name resources and set configurations as specified. In particular, values in these instructions that appear in **This Format** should be entered exactly as documented (case-sensitive).

B. Task 1: Create Your VPC

In this task, you will use the **VPC and more** option in the VPC console to create multiple resources, including a **VPC**, an **Internet Gateway**, a **public subnet** and a **private subnet** in a single Availability Zone, two **route tables**, and a **NAT Gateway**.

4. In the search box to the right of **Services**, search for and choose **VPC** to open the VPC console.

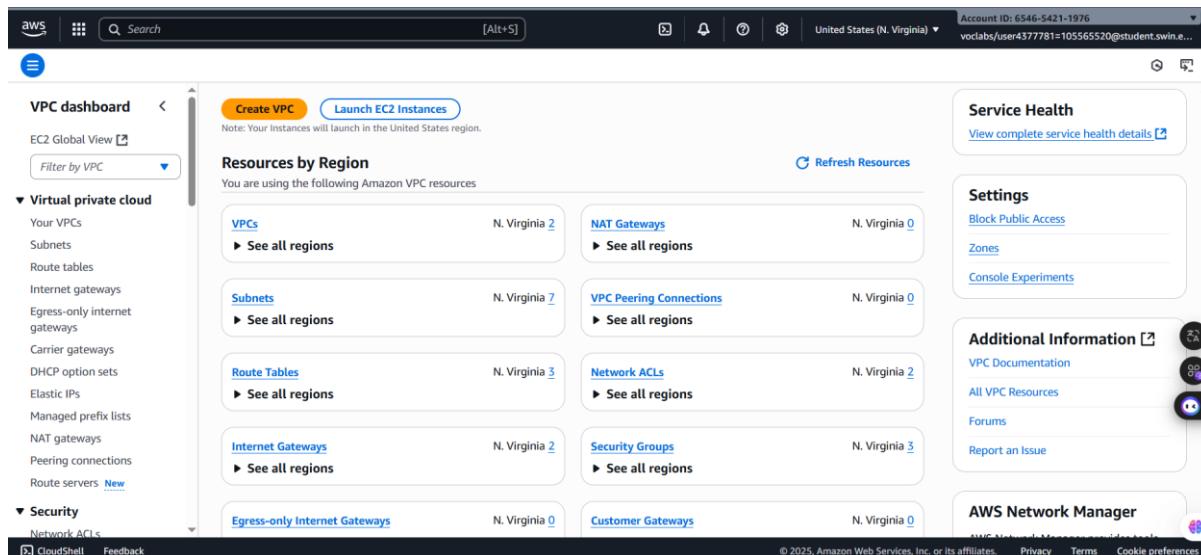
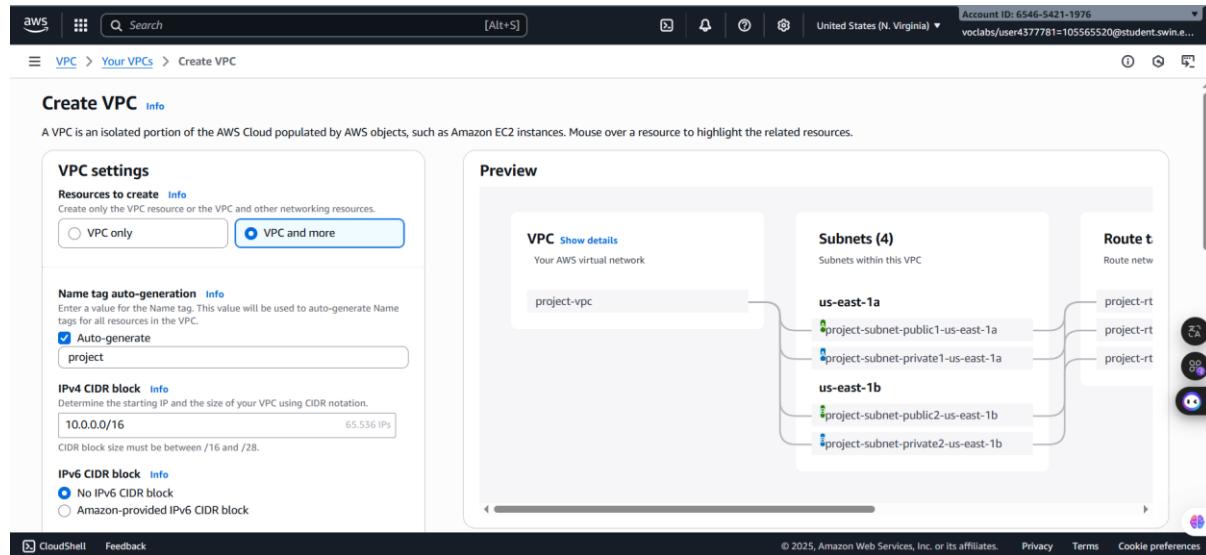


Figure 2: VPC Homepage

5. Begin creating a VPC.

- In the top right of the screen, verify that **N. Virginia (us-east-1)** is the region.
- Choose the **VPC dashboard** link which is towards the top left of the console.
- Next, choose **Create VPC**.

Note: If you do not see a button with that name, choose the Launch VPC Wizard button instead.

**Figure 3: VPC Configurations page**

6. Configure the VPC details in the *VPC settings* panel on the left:

- Choose **VPC and more**.
- Under **Name tag auto-generation**, keep *Auto-generate* selected, however change the value from project to lab.
- Keep the **IPv4 CIDR block** set to 10.0.0.0/16
- For **Number of Availability Zones**, choose **1**.
- For **Number of *public* subnets**, keep the **1** setting.
- For **Number of *private* subnets**, keep the **1** setting.
- Expand the **Customize subnets CIDR blocks** section
 - Change **Public subnet CIDR block** in **us-east-1a** to 10.0.0.0/24
 - Change **Private subnet CIDR block** in **us-east-1a** to 10.0.1.0/24
- Set **NAT gateways** to **In 1 AZ**.
- Set **VPC endpoints** to **None**.
- Keep both **DNS hostnames** and **DNS resolution enabled**

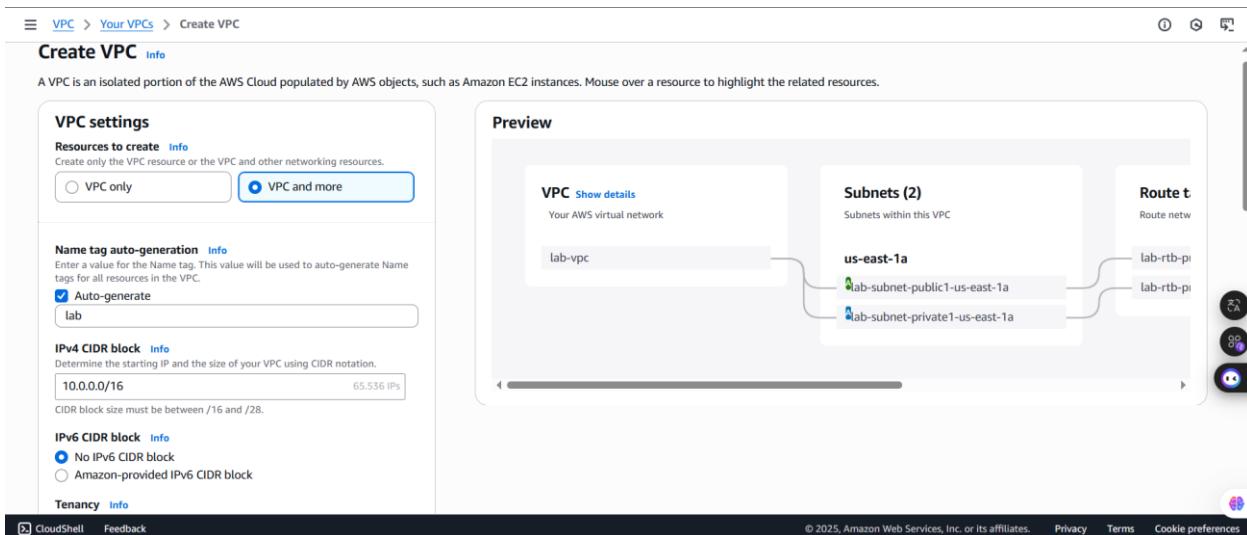


Figure 4: VPC settings

7. In the *Preview* panel on the right, confirm the settings you have configured.

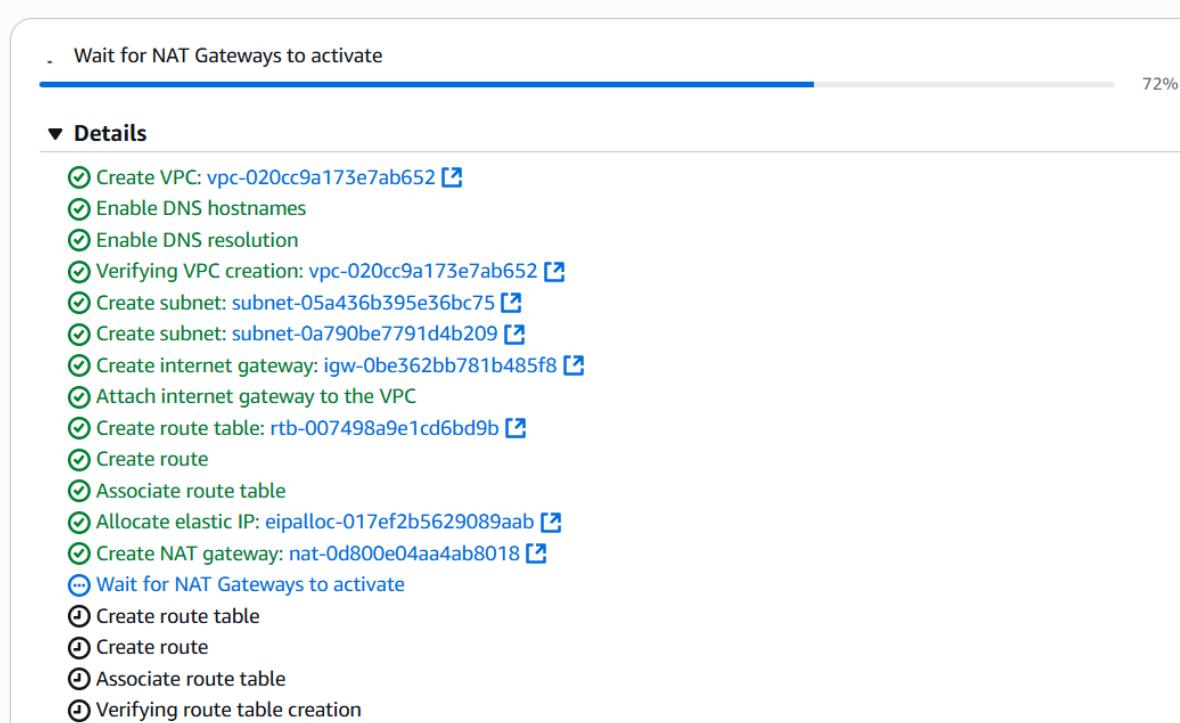
- **VPC:** lab-vpc
- **Subnets:**
 - us-east-1a
 - **Public subnet name:** lab-subnet-public1-us-east-1a
 - **Private subnet name:** lab-subnet-private1-us-east-1a
- **Route tables**
 - lab-rtb-public
 - lab-rtb-private1-us-east-1a
- **Network connections**
 - lab-igw
 - lab-nat-public1-us-east-1a

8. At the bottom of the screen, choose **Create VPC**

The VPC resources are created. The NAT Gateway will take a few minutes to activate.

Please wait until *all* the resources are created before proceeding to the next step.

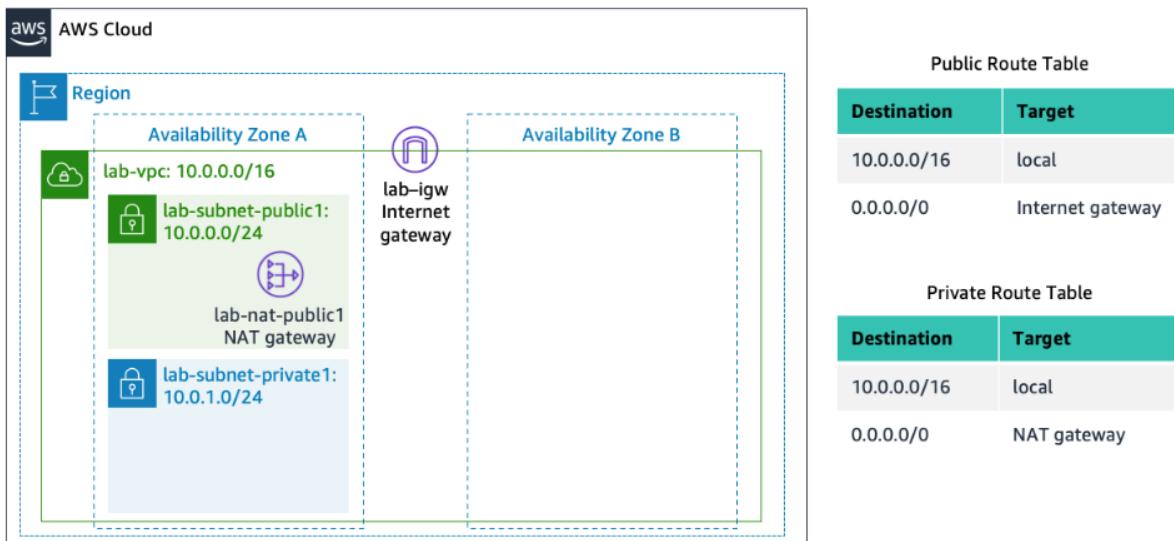
Create VPC workflow

**Figure 5: Setting**

9. Once it is complete, choose **View VPC**

The wizard has provisioned a VPC with a public subnet and a private subnet in one Availability Zone with route tables for each subnet. It also created an Internet Gateway and a NAT Gateway.

To view the settings of these resources, browse through the VPC console links that display the resource details. For example, choose **Subnets** to view the subnet details and choose **Route tables** to view the route table details. The diagram below summarizes the VPC resources you have just created and how they are configured.



An *Internet gateway* is a VPC resource that allows communication between EC2 instances in your VPC and the Internet.

The lab-subnet-public1-us-east-1a public subnet has a CIDR of **10.0.0.0/24**, which means that it contains all IP addresses starting with **10.0.0.x**. The fact the route table associated with this public subnet routes 0.0.0.0/0 network traffic to the internet gateway is what makes it a public subnet.

A *NAT Gateway*, is a VPC resource used to provide internet connectivity to any EC2 instances running in *private* subnets in the VPC without those EC2 instances needing to have a direct connection to the internet gateway.

The lab-subnet-private1-us-east-1a private subnet has a CIDR of **10.0.1.0/24**, which means that it contains all IP addresses starting with **10.0.1.x**.

C. Task 2: Create Additional Subnets

In this task, you will create two additional subnets for the VPC in a second Availability Zone. Having subnets in multiple Availability Zones within a VPC is useful for deploying solutions that provide *High Availability*.

After creating a VPC as you have already done, you can still configure it further, for example, by adding more subnets. Each subnet you create resides entirely within one Availability Zone.

10. In the left navigation pane, choose Subnets.

First, you will create a second *public* subnet.

The screenshot shows the AWS VPC Subnets mainpage. On the left, there's a navigation pane with 'VPC dashboard' selected. The main area displays a table titled 'Subnets (7) Info' with columns for Name, Subnet ID, State, VPC, Block Public..., and IPv4 CIDR. The table lists seven subnets, each associated with a specific VPC ID and IP range. The last updated time is 25 minutes ago.

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
-	subnet-0976aa95ffa13a33e	Available	vpc-0871038bb6152e8eb	Off	172.31.32.0
-	subnet-08e1cbd966bfc43a6	Available	vpc-0871038bb6152e8eb	Off	172.31.16.0
-	subnet-062539069bdfea499	Available	vpc-0871038bb6152e8eb	Off	172.31.80.0
-	subnet-0ff3effe1ad02109c	Available	vpc-0871038bb6152e8eb	Off	172.31.64.0
-	subnet-055decfafb1d5c408	Available	vpc-052e896005b3e898d1 Wor...	Off	10.0.0.0/24
Work Public Subnet					

Figure 6: Subnets mainpage

11. Choose **Create subnet then configure:**

- **VPC ID: lab-vpc** (select from the menu).
- **Subnet name: lab-subnet-public2**
- **Availability Zone: Select the second Availability Zone (for example, us-east-1b)**
- **IPv4 CIDR block: 10.0.2.0/24**

The subnet will have all IP addresses starting with **10.0.2.x**.

The screenshot shows the 'Create subnet' configuration page. It includes fields for Subnet name (lab-subnet-public2), Availability Zone (United States (N. Virginia) / use1-az4 (us-east-1b)), IPv4 VPC CIDR block (10.0.0.0/16), and IPv4 subnet CIDR block (10.0.2.0/24). A note indicates that the subnet's IPv4 CIDR must lie within the VPC's CIDR block.

Figure 7: Public subnet configuration

Create subnet

12. Choose

The second *public* subnet was created. You will now create a second *private* subnet.

13. Choose **Create subnet** then configure:

- **VPC ID:** lab-vpc
- **Subnet name:** lab-subnet-private2
- **Availability Zone:** Select the second Availability Zone (for example, us-east-1b)
- **IPv4 CIDR block:** 10.0.3.0/24

The subnet will have all IP addresses starting with **10.0.3.x**

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 256 IPs
< > ^ v

Figure 8: Private subnet configuration

14. Choose **Create subnet**

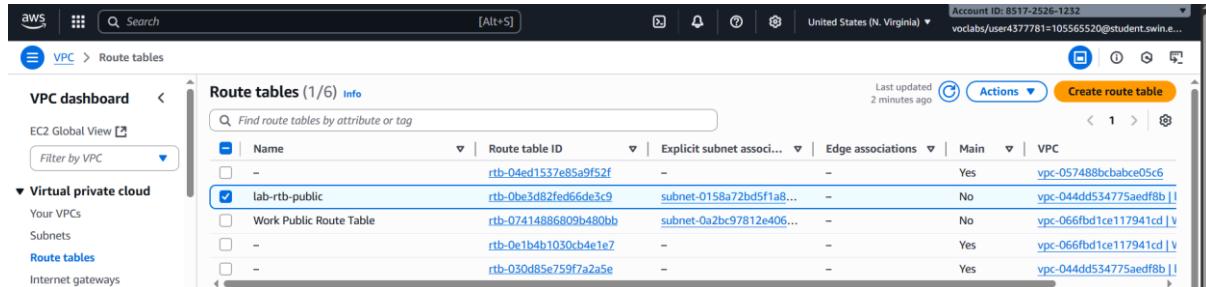
The second private subnet was created.

You will now configure this new private subnet to route internet-bound traffic to the NAT Gateway so that resources in the second private subnet are able to connect to the Internet, while still keeping the resources private. This is done by configuring a Route Table.

A route table contains a set of rules, called routes, that are used to determine where network traffic is directed. Each subnet in a VPC must be associated with a route table; the route table controls routing for the subnet

15. In the left navigation pane, choose **Route tables**.16. Select the **lab-rtb-private1-us-east-1a** route table.

Note: If the newly created routes are not visible, choose  refresh button at the top to update the list of routes.



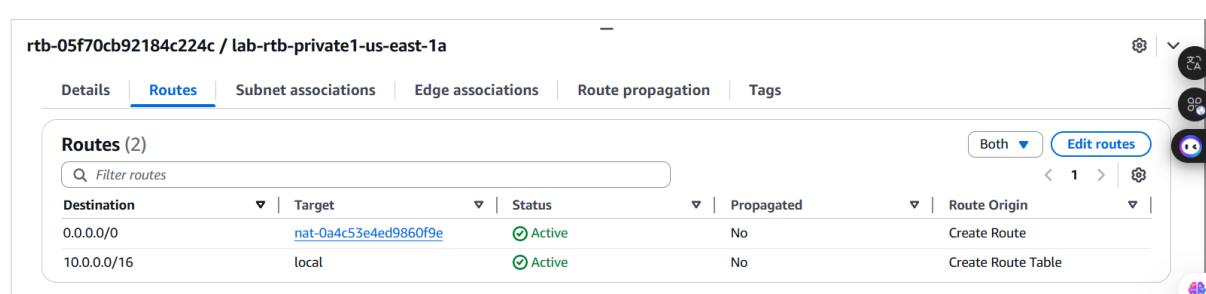
Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
rtb-04ed1537e85a9f52f	-	-	-	Yes	vpc-057488bbcbae05c6
lab-rtb-public	rtb-0be3d82fed66de3c9	subnet-0158a72bd5f1a8...	-	No	vpc-044dd534775aedff8b 
Work Public Route Table	rtb-07414886809b480bb	subnet-0a2bc97812e406...	-	No	vpc-066fbdb1ce117941cd 
-	rtb-0e1b4b1030cb4e1e7	-	-	Yes	vpc-066fbdb1ce117941cd 
-	rtb-030d85e759f7a2a5e	-	-	Yes	vpc-044dd534775aedff8b 

Figure 9: lab-rtb-private1-us-east-1a selected

17. In the lower pane, choose the **Routes** tab.

Note that **Destination 0.0.0.0/0** is set to **Target nat-xxxxxxxx**. This means that traffic destined for the internet (0.0.0.0/0) will be sent to the NAT Gateway. The NAT Gateway will then forward the traffic to the internet.

This route table is therefore being used to route traffic from private subnets.



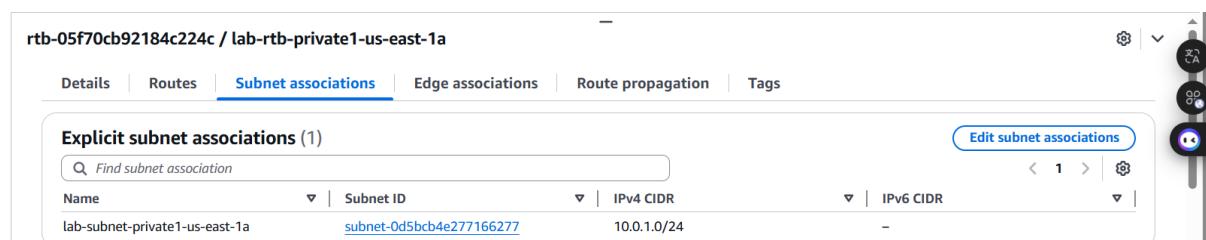
Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	nat-0a4c53e4ed9860f9e	Active	No	Create Route
10.0.0.0/16	local	Active	No	Create Route Table

Figure 10: Routes tab

18. Choose the **Subnet associations** tab.

You created this route table in task 1 when you chose to create a VPC and multiple resources in the VPC. That action also created lab-subnet-private-1 and associated that subnet with this route table.

Now that you have created another private subnet, lab-subnet-private-2, you will associate this route table with that subnet as well.



Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
lab-subnet-private1-us-east-1a	subnet-0d5bcb4e277166277	10.0.1.0/24	-

Figure 11: Subnet associations tab

19. In the Explicit subnet associations panel, choose **Edit subnet associations**

20. Leave **lab-subnet-private1-us-east-1a** selected, but also select **lab-subnet-private2**.

Available subnets (1/4)					
	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input type="checkbox"/>	lab-subnet-private1-us-east-1a	subnet-0d5bcb4e277166277	10.0.1.0/24	-	rtb-05f70cb92184c224c / lab-rtb-priv
<input type="checkbox"/>	lab-subnet-public2	subnet-0d117ba1edc9de7d3	10.0.2.0/24	-	Main (rtb-030d85e759f7a2a5e)
<input checked="" type="checkbox"/>	lab-subnet-private2	subnet-0366302c1c97a4982	10.0.3.0/24	-	Main (rtb-030d85e759f7a2a5e)
<input type="checkbox"/>	lab-subnet-public1-us-east-1a	subnet-0158a72bd5f1a85c6	10.0.0.0/24	-	rtb-0be3d82fed66de3c9 / lab-rtb-public

Selected subnets	
subnet-0366302c1c97a4982 / lab-subnet-private2	X

Figure 12: Configuration changed

21. Choose **Save associations**

You will now configure the Route Table that is used by the Public Subnets.

22. Select the **lab-rtb-public** route table (and deselect any other subnets).

Route tables (1/6) Info						
	Name	Route table ID	Explicit subnet associations	Edge associations	Main	VPC
<input type="checkbox"/>		rtb-04ed1537e85a9f52f	-	-	Yes	vpc-057488bcbabce05c6
<input checked="" type="checkbox"/>	lab-rtb-public	rtb-0be3d82fed66de3c9	subnet-0158a72bd5f1a8...	-	No	vpc-044dd534775aeafdf8b11
<input type="checkbox"/>	Work Public Route Table	rtb-07414886809b480bb	subnet-02bc97812e406...	-	No	vpc-066fbfd1ce117941cd11
<input type="checkbox"/>		rtb-0a11b1103b1b1a1a7	-	-	Yes	vpc-066fbfd1ce117941cd11

Figure 13: lab-rtb-public selected

23. In the lower pane, choose the **Routes** tab.

Note that **Destination 0.0.0.0/0** is set to Target **igw-xxxxxxxx**, which is an Internet Gateway. This means that internet-bound traffic will be sent straight to the internet via this Internet Gateway.

You will now associate this route table to the second public subnet you created.

24. Choose the **Subnet associations** tab.

25. In the Explicit subnet associations panel, choose **Edit subnet associations**

26. Leave **lab-subnet-public1-us-east-1a** selected, but also select **lab-subnet-public2**

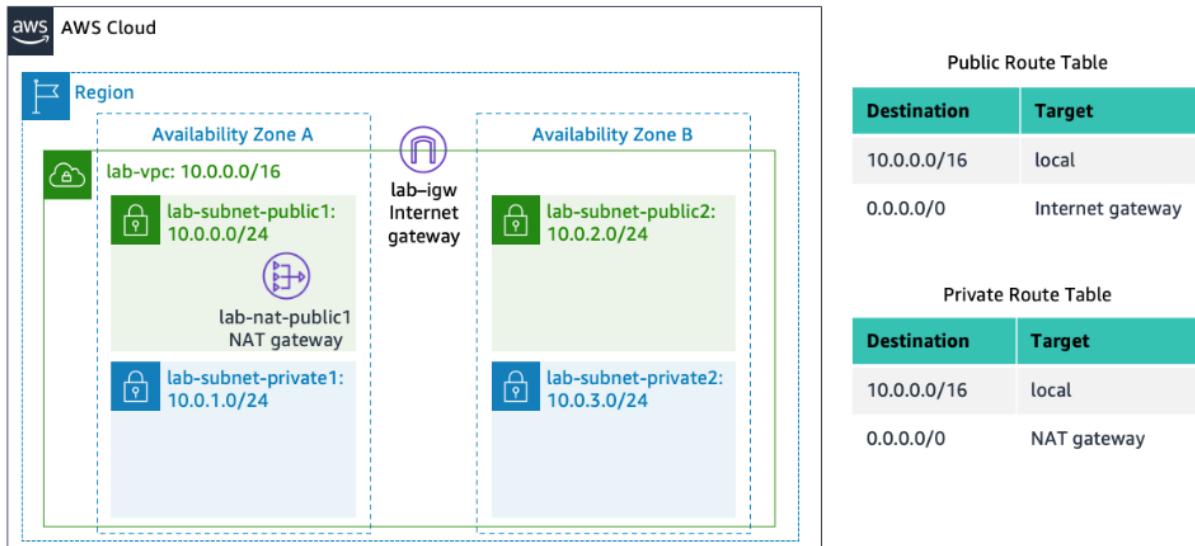
The screenshot shows the 'Edit subnet associations' page in the AWS VPC console. The 'Available subnets' table lists four subnets:

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
lab-subnet-private1-us-east-1a	subnet-0d5bcb4e277166277	10.0.1.0/24	-	Main (rtb-030d85e759f7a2a5e)
<input checked="" type="checkbox"/> lab-subnet-public2	subnet-0d117ba1edc9de7d3	10.0.2.0/24	-	Main (rtb-030d85e759f7a2a5e)
lab-subnet-private2	subnet-0366302c1c97a4982	10.0.3.0/24	-	rtb-05f70cb92184c224c / lab-rtb-priv...
lab-subnet-public1-us-east-1a	subnet-0158a72bd5f1a85c6	10.0.0.0/24	-	rtb-0be3d82fed66de3c9 / lab-rtb-public

The 'Selected subnets' list contains 'subnet-0d117ba1edc9de7d3 / lab-subnet-public2'. At the bottom right are 'Cancel' and 'Save associations' buttons.

Figure 14: lab-subnet-public2 selected

27. Choose **Save associations**
Your VPC now has public and private subnets configured in two Availability Zones. The route tables you created in task 1 have also been updated to route network traffic for the two new subnets.



D. Task 3: Create a VPC Security Group

In this task, you will create a VPC security group, which acts as a virtual firewall. When you launch an instance, you associate one or more security groups with the instance. You can add rules to each security group that allow traffic to or from its associated instances.

28. In the left navigation pane, choose **Security groups**.

29. Choose **Create security group** and then configure:

- **Security group name:** Web Security Group
- **Description:** Enable HTTP access
- **VPC:** choose the X to remove the currently selected VPC, then from the drop down list choose **lab-vpc**

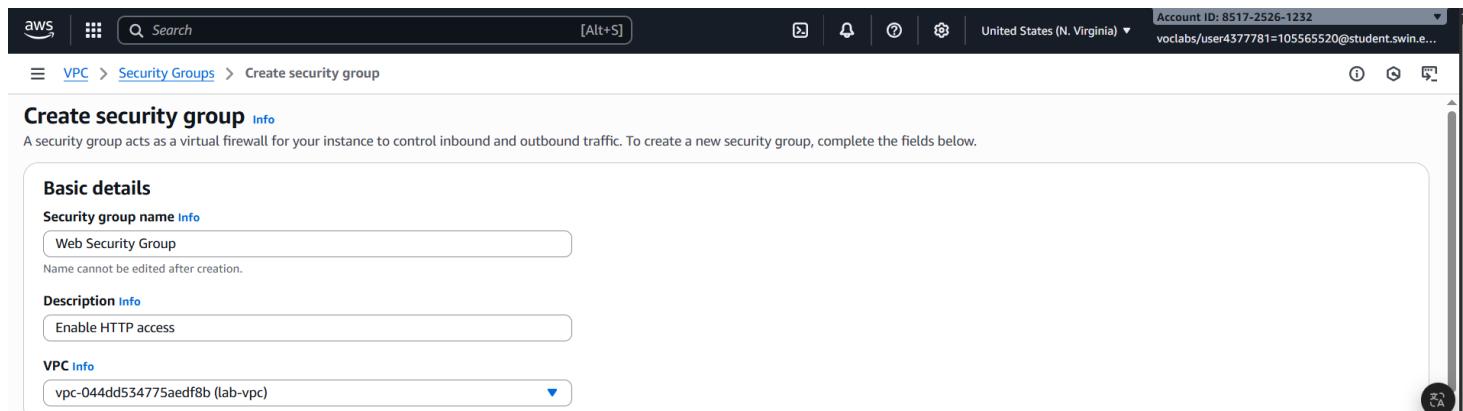


Figure 15: Security groups configuration

30. In the **Inbound rules** pane, choose **Add rule**

31. Configure the following settings:

- **Type:** HTTP
- **Source:** Anywhere-IPv4
- **Description:** Permit web requests

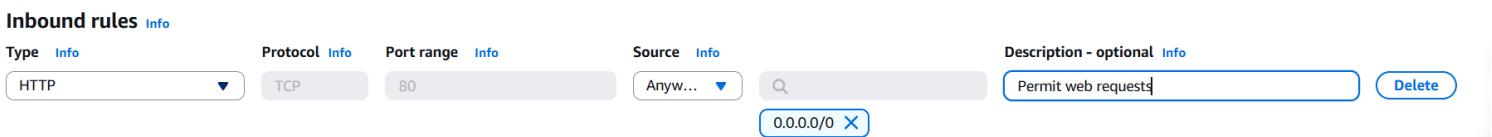


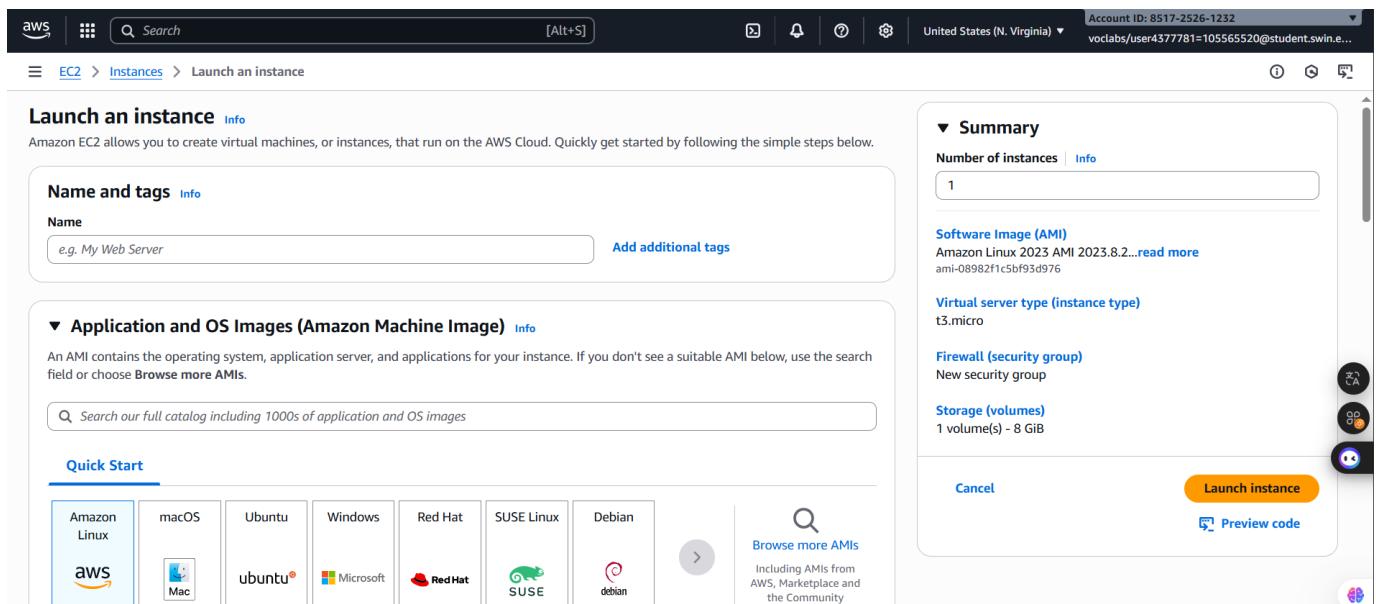
Figure 16: Inbound configuration

32. Scroll to the bottom of the page and choose **Create security group**
You will use this security group in the next task when launching an Amazon EC2 instance.

E. Task 4: Launch a Web Server Instance

33. In the search box to the right of **Services**, search for and choose EC2 to open the EC2 console.

34. From

Launch instancethe menu choose **Launch instance**.**Figure 17: EC2 Homepage**

35. Name the instance:

- Give it the name Web Server 1

When you name your instance, AWS creates a tag and associates it with the instance. A tag is a key value pair. The key for this pair is ***Name***, and the value is the name you enter for your EC2 instance.

Name and tags [Info](#)

Name

Web Server 1

[Add additional tags](#)**Figure 18: Setting name for web server**

36. Choose an AMI from which to create the instance:

- In the list of available Quick Start AMIs, keep the default **Amazon Linux** selected.
- Also keep the default **Amazon Linux 2023 AMI** selected.

The type of Amazon Machine Image (AMI) you choose determines the Operating System that will run on the EC2 instance that you launch.

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose **Browse more AMIs**.

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

Recents Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux Debian

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI
ami-08982f1c5bf93d976 (64-bit (x86), uefi-preferred) / ami-039f81f5ce6752b10 (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Figure 19: AMI Configuration

37. Choose an Instance type:

- In the Instance type panel, keep the default t2.micro selected.
The Instance Type defines the hardware resources assigned to the instance.

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

Instance type

t2.micro

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](#)

[Additional costs apply for AMIs with pre-installed software](#)

Figure 20: Choose an instance type

38. Select the key pair to associate with the instance:

- From the **Key pair name** menu, select **vockey**.

The vockey key pair you selected will allow you to connect to this instance via SSH after it has launched. Although you will not need to do that in this lab, it is still required to identify an existing key pair, or create a new one, or choose to proceed without a key pair, when you launch an instance.

▼ 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

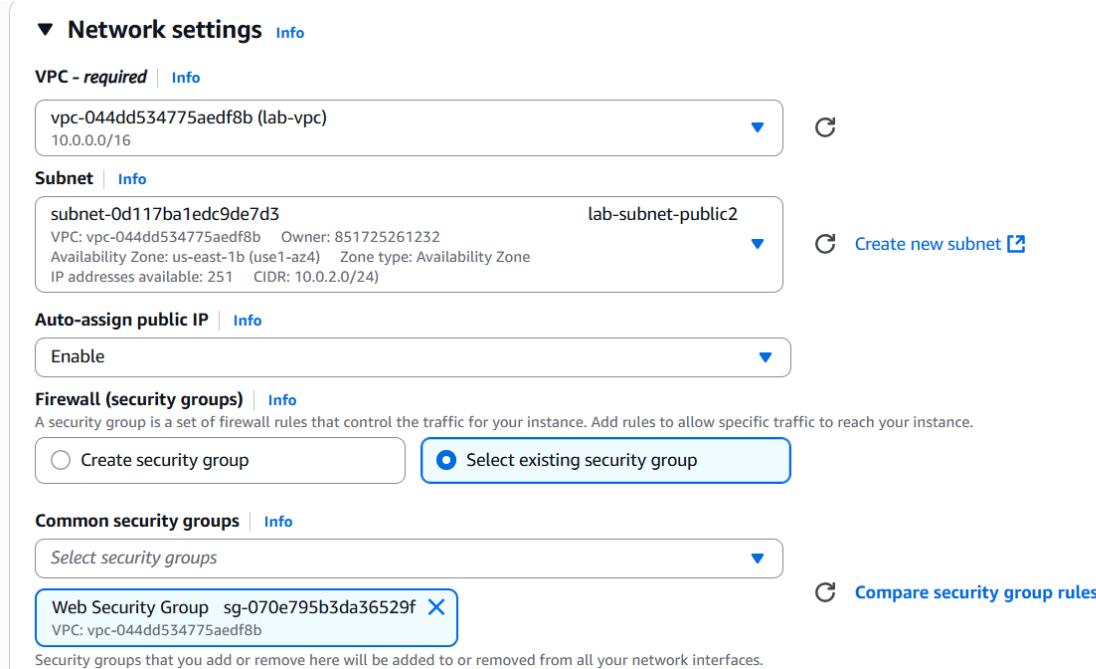
vockey

[Create new key pair](#)

Figure 21: Choose the key pair

39. Configure the Network settings:

- Next to Network settings, choose **Edit**, then configure:
 - **Network:** lab-vpc
 - **Subnet:** lab-subnet-public2 (not Private!)
 - **Auto-assign public IP:** Enable
- Next, you will configure the instance to use the Web Security Group that you created earlier.
 - Under Firewall (security groups), choose **Select existing security group.**
 - For **Common security groups**, select **Web Security Group.**
This security group will permit HTTP access to the instance.

Figure 21: Choose the key pair**Figure 22: Network configuration**

40. In the Configure storage section, keep the default settings.

Note: The default settings specify that the root volume of the instance, which will host the Amazon Linux guest operating system that you specified earlier, will run on a general purpose SSD (gp3) hard drive that is 8 GiB in size. You could alternatively add more storage volumes, however that is not needed in this lab.

41. Configure a script to run on the instance when it launches:

- Expand the **Advanced details** panel.

- Scroll to the bottom of the page and then copy and paste the code shown below into the User data box:

```
#!/bin/bash
# Install Apache Web Server and PHP
dnf install -y httpd wget php mariadb105-server
# Download Lab files
wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-100-ACCLFO-2/2-
lab2-vpc/s3/lab-app.zip
unzip lab-app.zip -d /var/www/html/
# Turn on web server
chkconfig httpd on
service httpd start
```

This script will run with root user permissions on the guest OS of the instance. It will run automatically when the instance launches for the first time. The script installs a web server, a database, and PHP libraries, and then it downloads and installs a PHP web application on the web server.

User data - optional | [Info](#)

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

[Choose file](#)

```
#!/bin/bash
# Install Apache Web Server and PHP
dnf install -y httpd wget php mariadb105-server
# Download Lab files
wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-100-ACCLFO-2/2-
lab2-vpc/s3/lab-app.zip
unzip lab-app.zip -d /var/www/html/
# Turn on web server
chkconfig httpd on
service httpd start
```

Figure 23: User Data Configuration

42. At the bottom of the **Summary** panel on the right side of the screen choose

Launch instance

You will see a Success message.

Success
Successfully initiated launch of instance (i-072526eade2b69bb2)

Figure 24: Launch instance successfully

43. Choose **View all instances**

44. Wait until **Web Server 1** shows 2/2 checks passed in the **Status check** column.

💡 This may take a few minutes. Choose the refresh icon at the top of the page every 30 seconds or so to more quickly become aware of the latest status of the instance.

You will now connect to the web server running on the EC2 instance.

45. Select **Web Server 1**.

46. Copy the **Public IPv4 DNS** value shown in the **Details** tab at the bottom of the page.

47. Open a new web browser tab, paste the **Public DNS** value and press Enter.

You should see a web page displaying the AWS logo and instance meta-data values.

The complete architecture you deployed is:

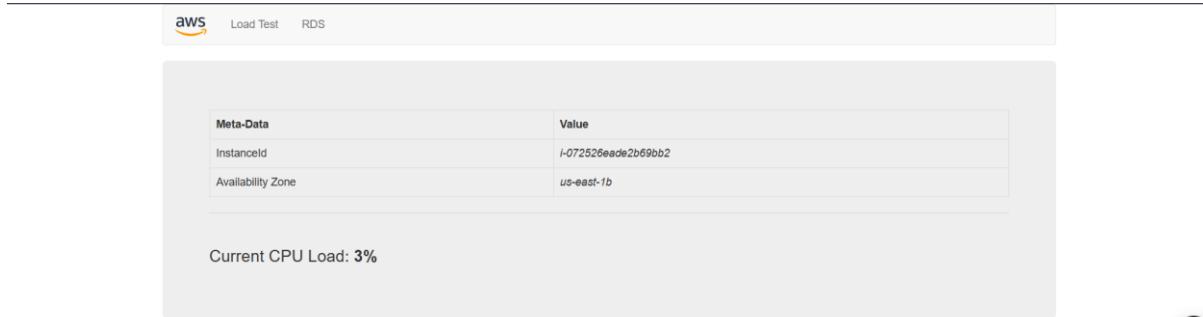
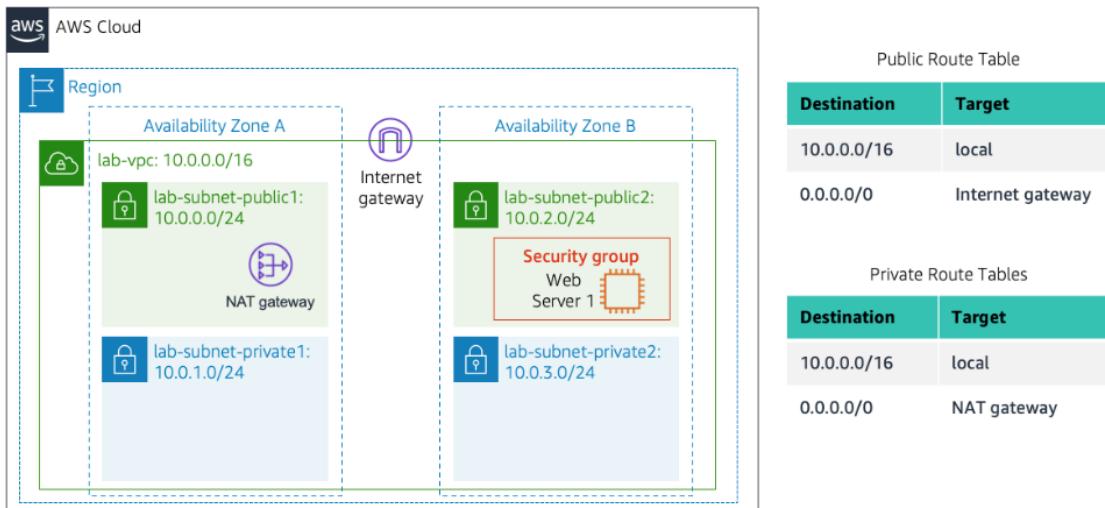


Figure 25: Launch DNS address successfully

F. Submitting your work

48. To record your progress, choose **Submit** at the top of these instructions.



Figure 26: Submit Confirmation

49. When prompted, choose **Yes**.

After a couple of minutes, the grades panel appears and shows you how many points you earned for each task. If the results don't display after a couple of minutes, choose Grades at the top of these instructions.

 Tip: You can submit your work multiple times. After you change your work, choose Submit again. Your last submission is recorded for this lab.

50. To find detailed feedback about your work, choose Submission Report.

! Tip: For any checks where you did not receive full points, there are sometimes helpful details provided in the submission report.

Total score	30/30
Task 1 - VPC created correctly	5/5
Task 2a - New subnets created correctly	5/5
Task 2b - Subnet route table association	5/5
Task 3 - Security group created correctly	5/5
Task 4a - EC2 instance created correctly	5/5
Task 4b - EC2 instance website accessible	5/5

Figure 27: Lab completed

