

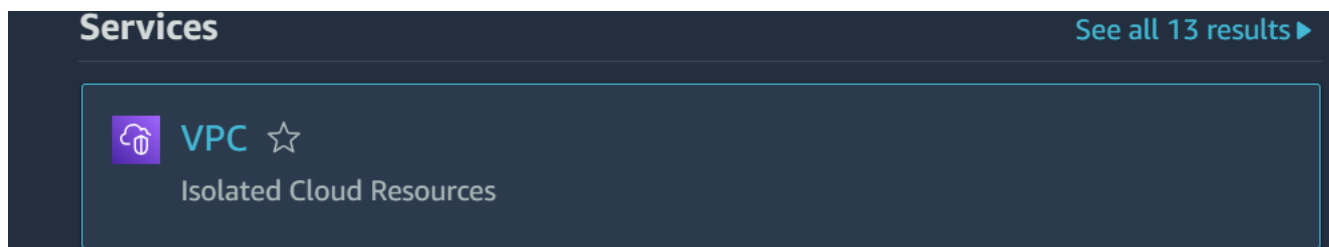
ASSIGNMENT-14

Cloud Computing Practical Assignment No: 14

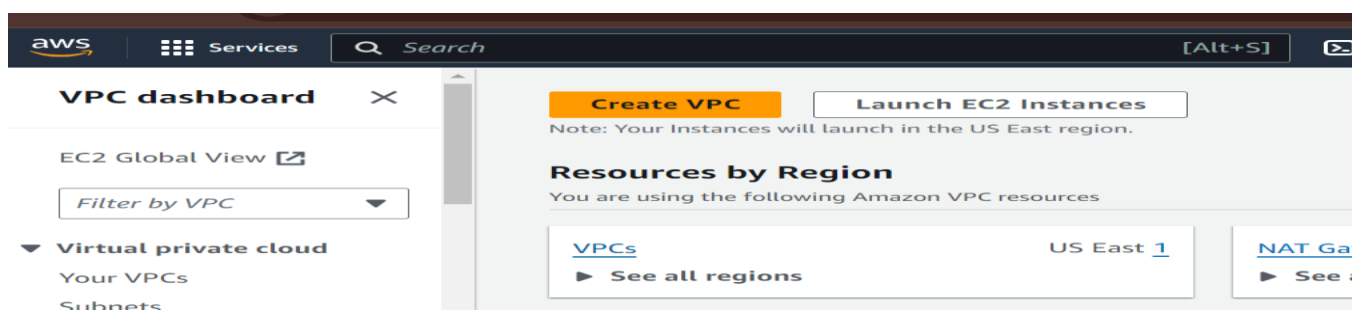
Implementation of VPC using AWS.

Task 1: Create Your VPC

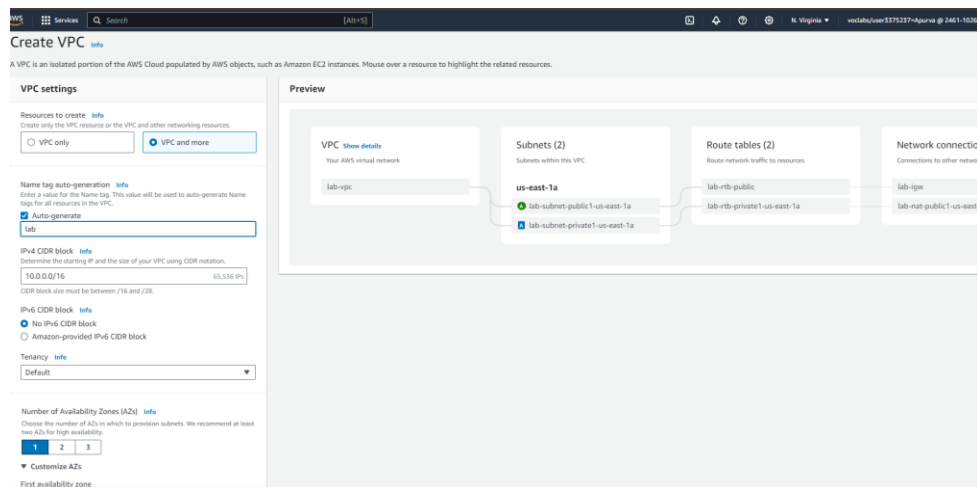
1. Go to the VPC console (search for "VPC" in the search bar).



2. Verify the region is N. Virginia (us-east-1).
3. Choose "Create VPC".

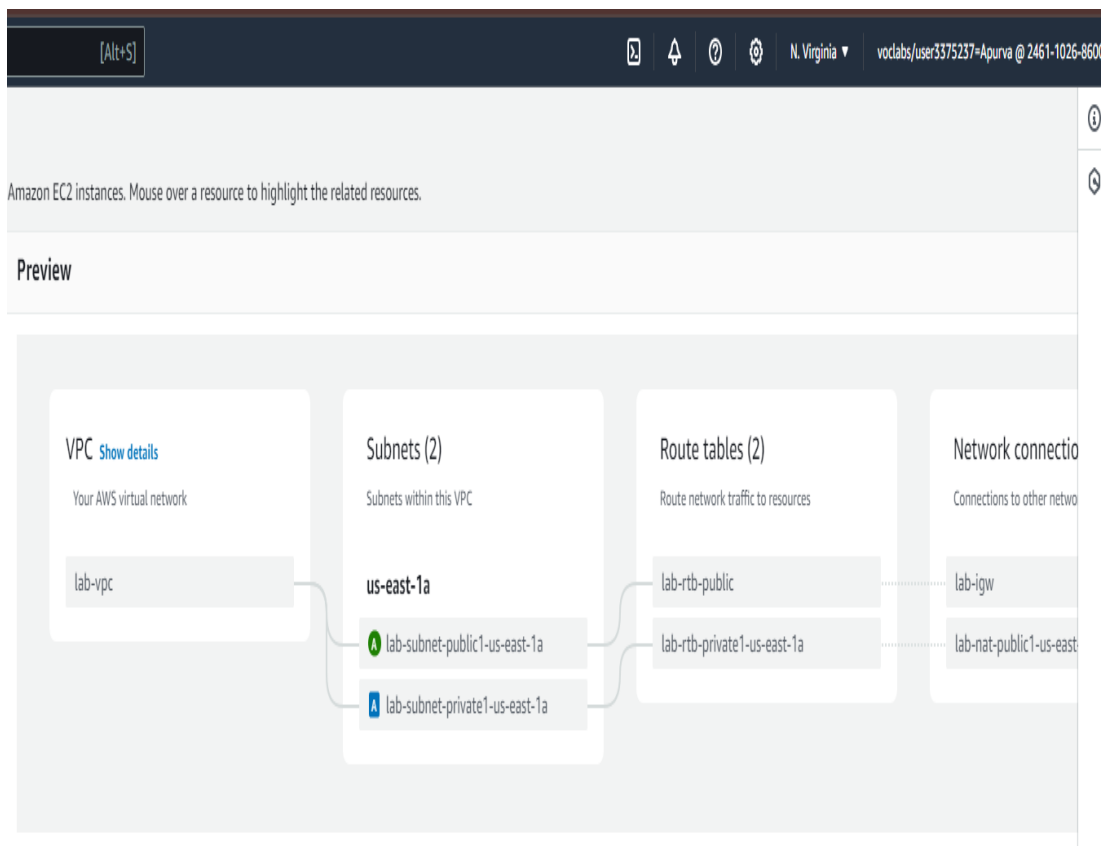


4. Under VPC settings:
- Keep "Auto-generate" selected for Name tag, but change the value to "lab".
 - Keep IPv4 CIDR block set to 10.0.0.0/16.
 - Choose 1 for Number of Availability Zones.



- Keep settings for Number of public subnets (1) and Number of private subnets (1).
- Expand "Customize subnets CIDR blocks" and change:

- Public subnet CIDR block to 10.0.0.0/24
 - Private subnet CIDR block to 10.0.1.0/24
 - Set NAT gateways to "In 1 AZ".
 - Keep VPC endpoints set to "None".
5. Review the settings in the Preview panel.



6. Click "Create VPC".

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-east-1a

10.0.0.0/24 256 IPs

Private subnet CIDR block in us-east-1a

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**

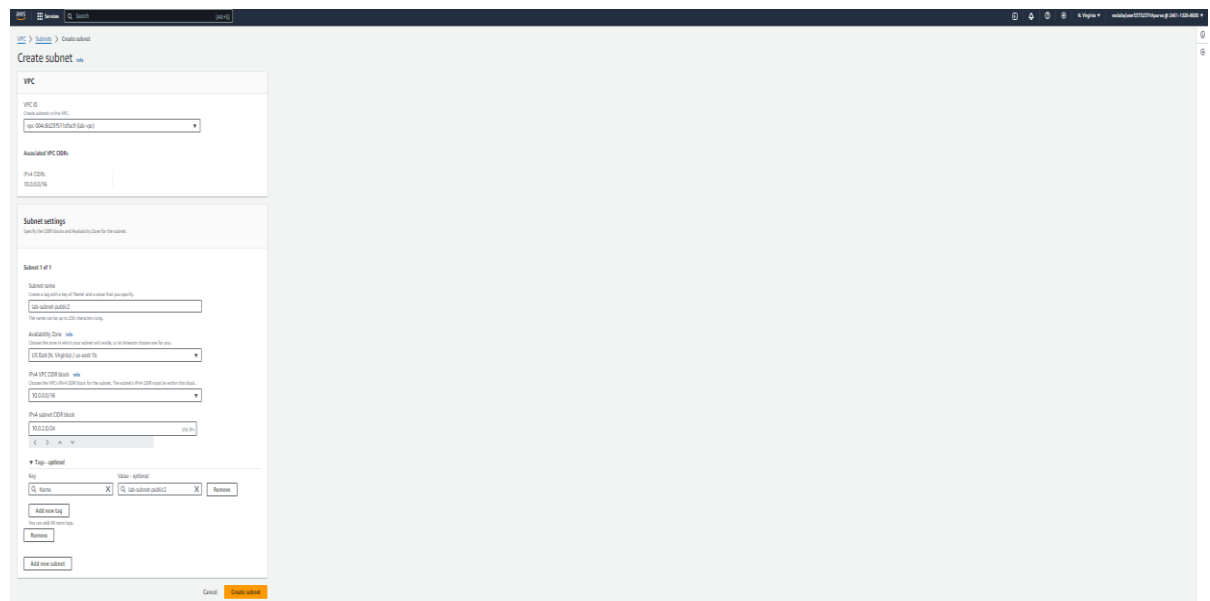
Cancel Create VPC

7. Wait for the VPC resources to be created (including the NAT Gateway).

8. Once complete, choose "View VPC".

Task 2: Create Additional Subnets

1. Go to the Subnets section in the VPC console.
2. Create a second public subnet with:
 - VPC ID: lab-vpc
 - Subnet name: lab-subnet-public2
 - Availability Zone: Select the second zone (e.g., us-east-1b)
 - IPv4 CIDR block: 10.0.2.0/24



The screenshot shows the 'Create subnet' form in the AWS VPC console. The form is titled 'Create subnet' and has a 'VPC' section and a 'Subnet settings' section. In the 'VPC' section, the 'VPC ID' is set to 'vpc-08a4e29f71e2c704e-us-east-1'. In the 'Subnet settings' section, the 'Subnet name' is 'lab-subnet-public2', the 'Availability Zone' is 'us-east-1b', and the 'IPv4 CIDR block' is '10.0.2.0/24'. The 'Subnet type' is set to 'Public'. There are also fields for 'Subnet tags' and 'Add new tag'.

3. Create a second private subnet with:
 - VPC ID: lab-vpc

- Subnet name: lab-subnet-private2
 - Availability Zone: Select the second zone (e.g., us-east-1b)
 - IPv4 CIDR block: 10.0.3.0/24
4. Configure the new private subnet to route internet traffic to the NAT Gateway (explained in the full instructions).

The screenshot shows the 'Subnet 2 of 2' configuration page in the AWS console. It includes fields for 'Subnet name' (lab-subnet-private2), 'Availability Zone' (US East (N. Virginia) / us-east-1b), 'IPv4 VPC CIDR block' (10.0.0.0/16), and 'IPv4 subnet CIDR block' (10.0.3.0/24). There is also a 'Tags' section with a key 'Name' and value 'lab-subnet-private2'. At the bottom are 'Cancel' and 'Create subnet' buttons.

Go to the Route tables section in the VPC console.

The screenshot shows the 'Route tables (1)' section in the AWS console. It includes a search bar, a table with columns for Name, Route table ID, Explicit subnet associations, Edge associations, Main, and VP, and a 'Create route table' button.

| | Name | Route table ID | Explicit subnet associations | Edge associations | Main | VP |
|--------------------------|------|---------------------------------------|------------------------------|-------------------|------|--------------------|
| <input type="checkbox"/> | - | rtb-0f3a17470e37e7a19 | - | - | Yes | vp |

Select the lab-rtb-private1-us-east-1a route table.

In the Routes tab, verify that the destination 0.0.0.0/0 is set to Target nat-xxxxxxx. This means that traffic destined for the internet will be sent to the NAT Gateway.

Go to the Subnet associations tab.

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

Select lab-subnet-private2 in addition to lab-subnet-private1-us-east-1a.

The screenshot shows the AWS Management Console interface for editing subnet associations. The breadcrumb trail is: VPC > Route tables > rtb-032c0a03fa75e9b6f > Edit subnet associations. The page title is "Edit subnet associations" with a subtitle "Change which subnets are associated with this route table." Below this, there is a section for "Available subnets (1/4)" with a search filter "Filter subnet associations". A table lists the available subnets:

| | Name | Subnet ID | IPv4 CIDR | IPv6 CIDR | Route table ID |
|-------------------------------------|--------------------------------|--------------------------|-------------|-----------|---|
| <input type="checkbox"/> | lab-subnet-private1-us-east-1a | subnet-06d45dce633446f5a | 10.0.1.0/24 | - | rtb-032c0a03fa75e9b6f / lab-rtb-priv... |
| <input type="checkbox"/> | lab-subnet-public2 | subnet-09b521115570b0aee | 10.0.2.0/24 | - | Main (rtb-04c6bd7f70bc15f7b) |
| <input type="checkbox"/> | lab-subnet-public1-us-east-1a | subnet-0e150b84481cf4353 | 10.0.0.0/24 | - | rtb-0fd83fbfb48952330 / lab-rtb-public |
| <input checked="" type="checkbox"/> | lab-subnet-private2 | subnet-0de6bf12e361efb7e | 10.0.3.0/24 | - | Main (rtb-04c6bd7f70bc15f7b) |

Below the table, the "Selected subnets" section shows a tag: "subnet-0de6bf12e361efb7e / lab-subnet-private2" with a close button. At the bottom right, there are "Cancel" and "Save associations" buttons.

Click Save associations.

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

The screenshot shows the AWS Management Console interface for Route Tables. At the top, a green notification bar states: "You have successfully updated subnet associations for rtb-032c0a03fa75e9b6f / lab-rtb-private1-us-east-1a." Below this, the "Route tables (1/4)" section is active. A table lists the route tables:

| Name | Route table ID | Explicit subnet associations | Edge associations | Main | VP |
|--|---------------------------------------|--|-------------------|------|----|
| - | rtb-0f3a17470e37e7a19 | - | - | Yes | vp |
| lab-rtb-private1-us-east-1a | rtb-032c0a03fa75e9b6f | subnet-0de6bf12e361efb7e / lab-subnet-private2 | - | No | vp |
| - | rtb-04c6bd7f70bc15f7b | - | - | Yes | vp |
| <input checked="" type="checkbox"/> lab-rtb-public | rtb-0fd83fbfb48952330 | subnet-0e150b84481cf4353 / lab-subnet-public1-us-east-1a | - | No | vp |

Below the table, the "Explicit subnet associations (1)" section is expanded, showing a table with one entry:

| Name | Subnet ID | IPv4 CIDR | IPv6 CIDR |
|-------------------------------|--|-------------|-----------|
| lab-subnet-public1-us-east-1a | subnet-0e150b84481cf4353 | 10.0.0.0/24 | - |

In the Routes tab, verify that the destination 0.0.0.0/0 is set to Target igw-xxxxxxx, which is an Internet Gateway.

Go to the Subnet associations tab.

In the Explicit subnet associations area, choose Edit subnet associations.

Select lab-subnet-public2 in addition to lab-subnet-public1-us-east-1a.

Click Save associations.

[Alt+S] N. Virginia voclabs/user3375237=Apurva @ 2461-1026-8600

You have successfully updated subnet associations for rtb-032c0a03fa75e9b6f / lab-rtb-private1-us-east-1a.

Route tables (1/4) Info Last updated 1 minute ago Actions Create route table

Find resources by attribute or tag

| Name | Route table ID | Explicit subnet associations | Edge associations | Main | VP |
|-----------------------------|---------------------------------------|--|-------------------|------|--------------------|
| - | rtb-0f3a17470e37e7a19 | - | - | Yes | vp |
| lab-rtb-private1-us-east-1a | rtb-032c0a03fa75e9b6f | subnet-0de6bf12e361efb7e / lab-subnet-private2 | - | No | vp |
| - | rtb-04c6bd7f70bc15f7b | - | - | Yes | vp |
| lab-rtb-public | rtb-0fd83fbfb48952330 | subnet-0e150b84481cf4353 / lab-subnet-public1-us-east-1a | - | No | vp |

Explicit subnet associations (1) Edit subnet associations

Find subnet association

| Name | Subnet ID | IPv4 CIDR | IPv6 CIDR |
|-------------------------------|--|-------------|-----------|
| lab-subnet-public1-us-east-1a | subnet-0e150b84481cf4353 | 10.0.0.0/24 | - |

Subnets without explicit associations (2) Edit subnet associations

The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table.

These steps will configure the route tables for the newly created subnets, ensuring that private subnet traffic is routed through the NAT Gateway and public subnet traffic is routed directly to the Internet Gateway.

Task 3: Create a VPC Security Group

1. Go to the Security groups section in the VPC console.

| | Name | Security group ID | Security group name | VPC ID | Description |
|--------------------------|------|--------------------------------------|---------------------|---------------------------------------|------------------------|
| <input type="checkbox"/> | - | sg-0b701bd8817b5b734 | default | vpc-0ca9c2c02fd652798 | default VPC security g |
| <input type="checkbox"/> | - | sg-004c636ad516bbae | launch-wizard-1 | vpc-0ca9c2c02fd652798 | launch-wizard-1 creat |

2. Create a security group with:
 - Security group name: Web Security Group
 - Description: Enable HTTP access
 - VPC: Choose lab-vpc

VPC > Security Groups > Create security group

Create security group [Info](#)

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name [Info](#)

Web Security Group

Name cannot be edited after creation.

Description [Info](#)

Enable HTTP access

VPC [Info](#)

vpc-004c8d23f511dfac9 (lab-vpc)

3. Add an inbound rule:
 - Type: HTTP
 - Source: Anywhere-IPv4
 - Description: Permit web requests

Inbound rules [Info](#)

| Type Info | Protocol Info | Port range Info | Source Info | Description - optional Info | |
|---------------------------|-------------------------------|---------------------------------|-----------------------------|---|---------------------|
| HTTP ▼ | TCP | 80 | Anywh... ▼ | 0.0.0.0/0 X | Permit web requests |

[Add rule](#) [Delete](#)

4. Save the security group.

Tags - optional

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

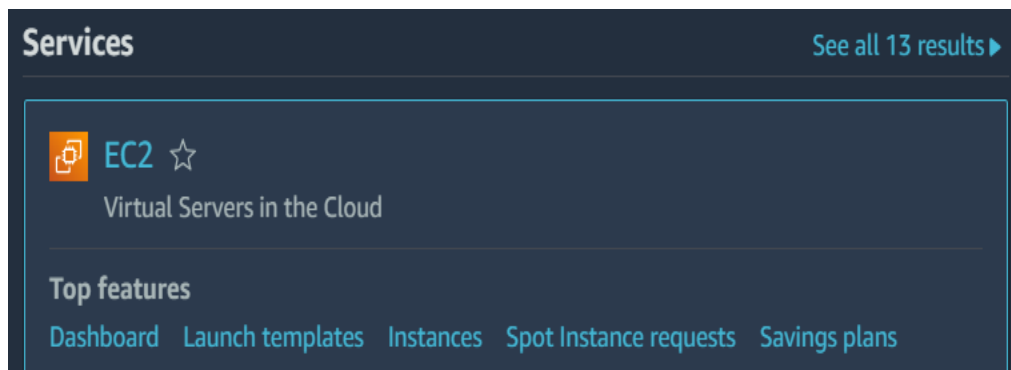
[Add new tag](#)

You can add up to 50 more tags.

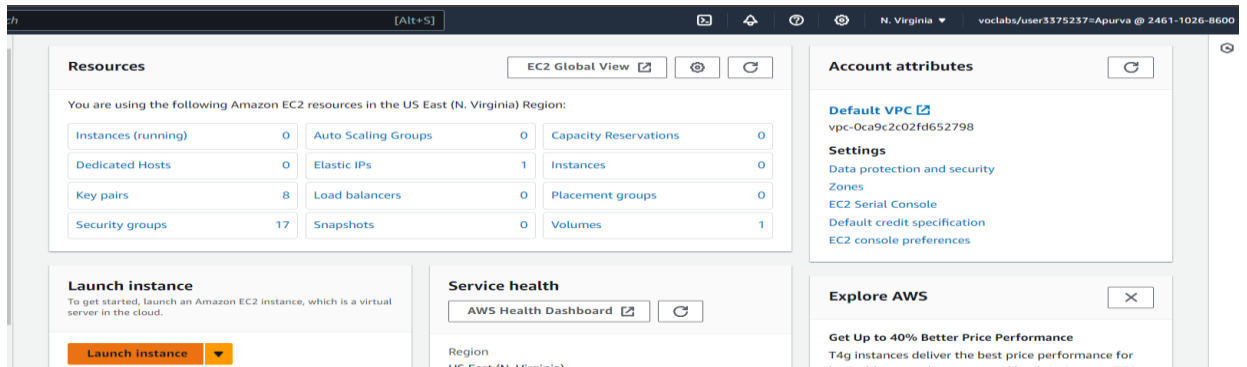
[Cancel](#) [Create security group](#)

Task 4: Launch a Web Server Instance

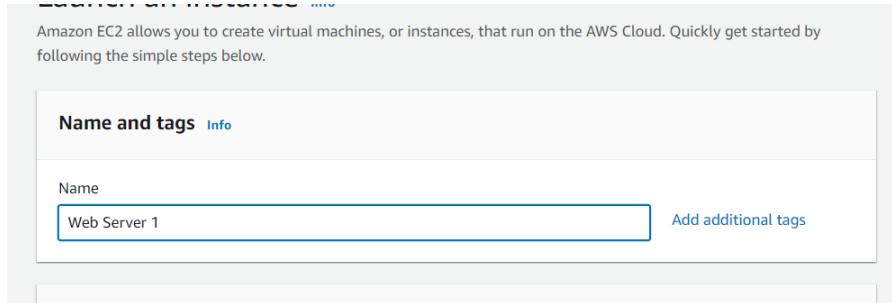
1. Go to the EC2 console (search for "EC2" in the search bar).



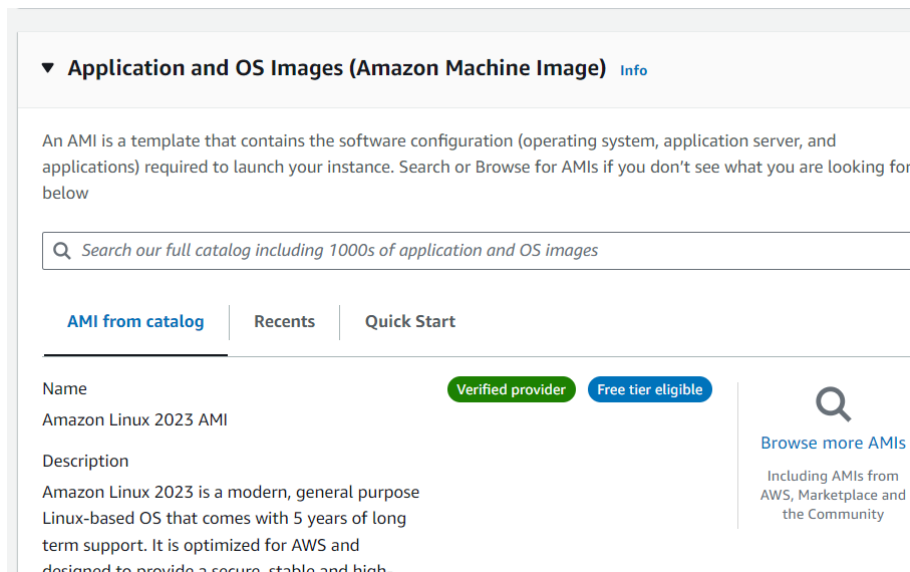
2. Launch a new instance.



3. Name the instance: Web Server 1



4. Choose the Amazon Linux 2023 AMI.



5. Choose the t2.micro instance type.

▼ 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 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

Free tier eligible

☒ All generations

[Compare instance types](#)

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

6. Select your key pair (e.g., radhey).

▼ 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*

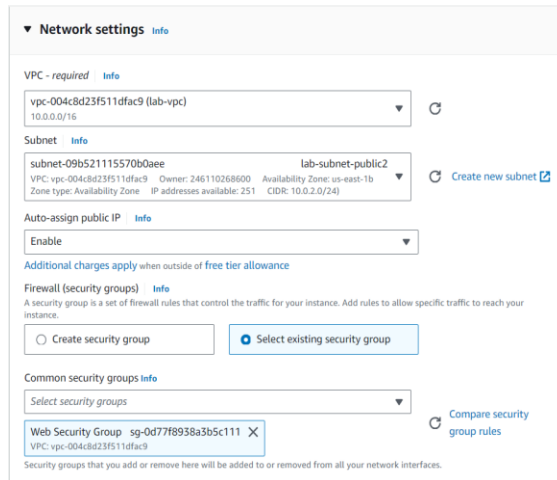
radhey

▼

[Create new key pair](#)

- ## 7. Configure network settings:
- Network: lab-vpc
 - Subnet: lab-subnet-public2 (public subnet)
 - Auto-assign public IP: Enable

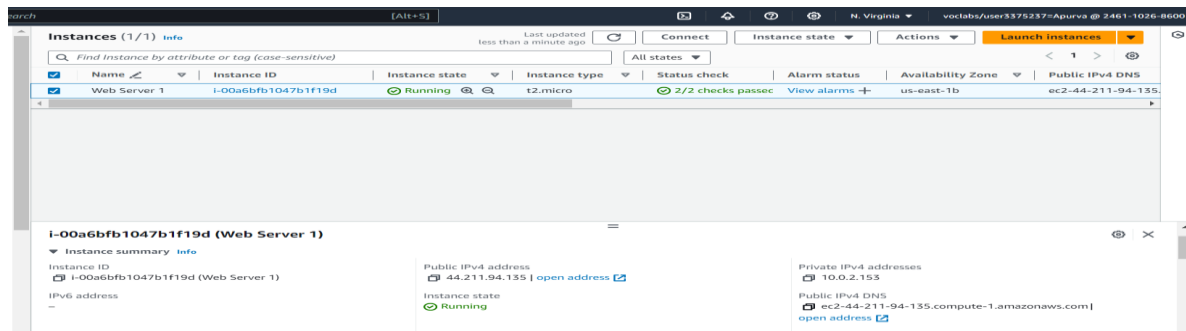
- Security group: Web Security Group (previously created)



The screenshot shows the 'Network settings' section of an AWS EC2 instance configuration. It includes a dropdown for 'VPC - required' set to 'vpc-004c8d23f511dfac9 (lab-vpc)'. Below it, a 'Subnet' dropdown is set to 'subnet-09b521115570b0aee (lab-subnet-public2)'. The 'Auto-assign public IP' option is set to 'Enable'. Under the 'Firewall (security groups)' section, the 'Select existing security group' radio button is chosen. A dropdown menu shows 'Web Security Group sg-0d77f8938a3b5c111' as the selected option. There are links for 'Create new subnet' and 'Compare security group rules'.

8. Keep the default storage settings.
 9. In the User data box, paste the script provided in the instructions. This script installs a web server and configure it.
- ```
#!/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
```
10. Launch the instance.
  11. Wait for the instance to launch (may take a few minutes).

12. Once launched, find the Public IPv4 DNS value in the instance details.



13. Open a web browser and paste the Public IPv4 DNS value. You should see a web page with the AWS logo and instance metadata.

