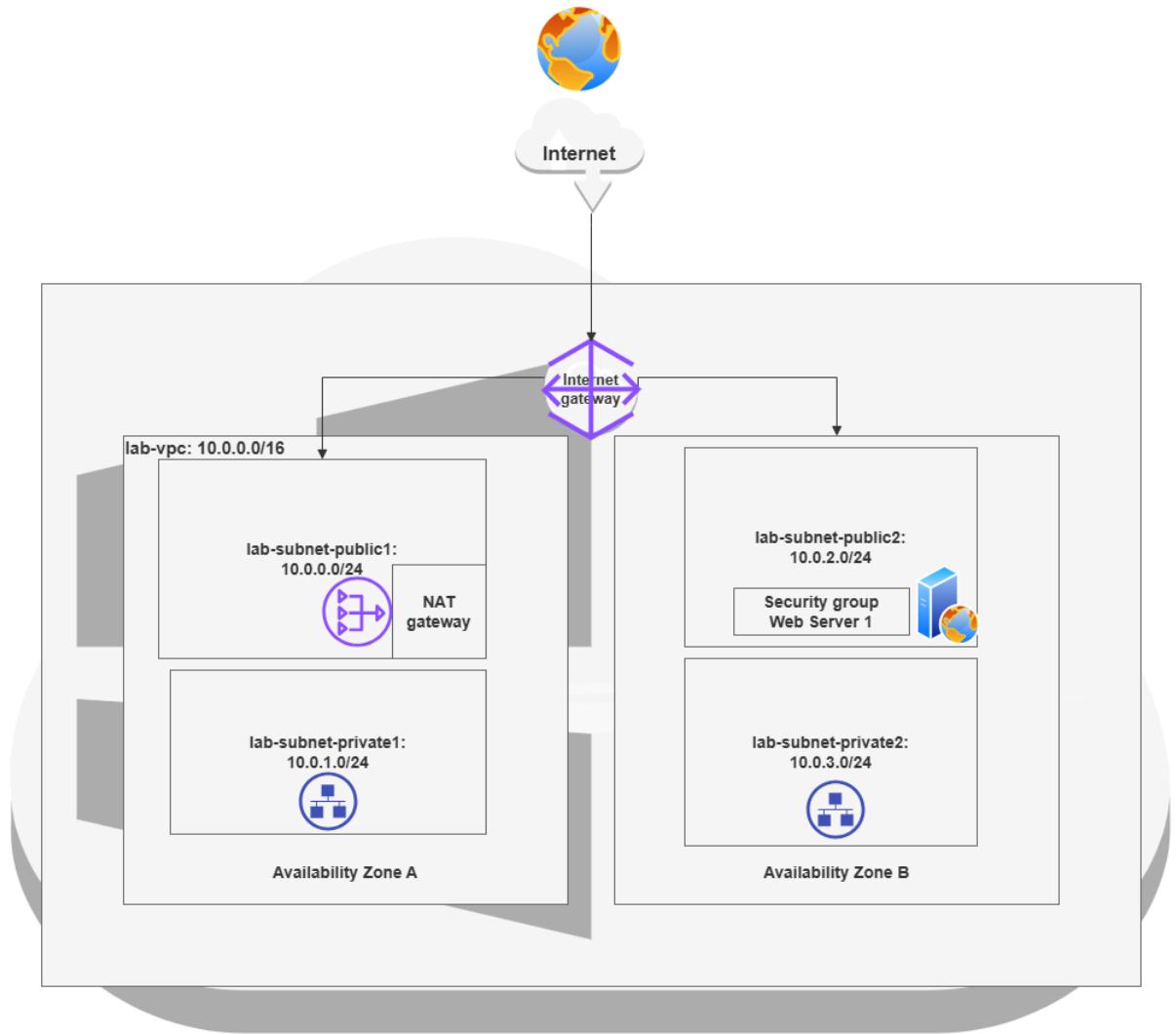


(Module 5 Lab - Build VPC and Launch a Web Server)

- Screenshot of network infrastructure topology



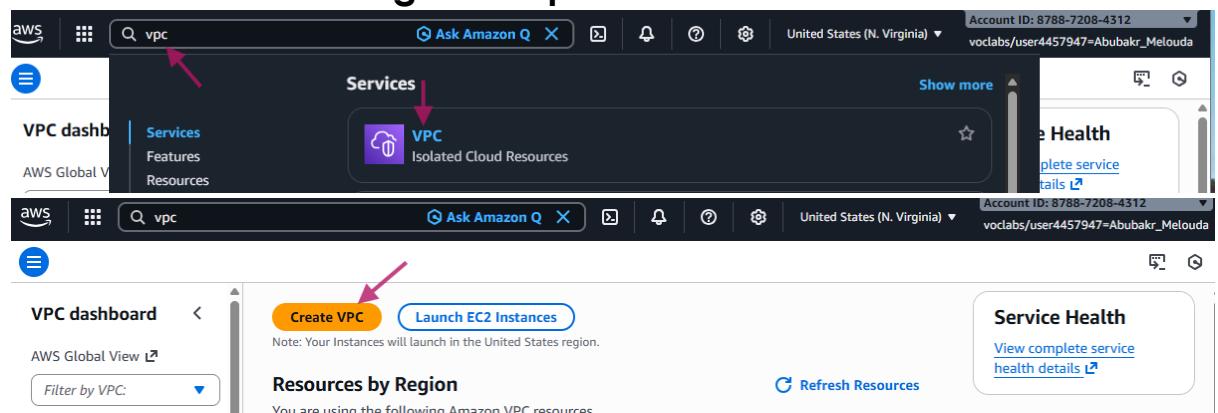
- The network was designed to achieve the following objectives:
 1. Security through segmentation: Public subnets (e.g. `10.0.0.0/24`) host resources like web servers that must be accessible from the internet using an Internet Gateway for external connectivity. Private subnets (e.g. `10.0.1.0/24`) host

internal resources such as application servers, backend systems and databases shielding them from direct internet exposure while still allowing controlled outbound access when needed.

2. High availability across availability zones: By deploying resources across multiple availability zones (AZs), services can continue running in another AZ if one becomes unavailable. This ensures improved fault tolerance and system resilience.

3. Improvement opportunity and enhanced resilience: To further strengthen this design, it would be ideal to deploy NAT Gateways in each availability zone. This would ensure private subnet resources maintain outbound internet connectivity even if availability zone-level failure impacts the NAT Gateway in a single zone.

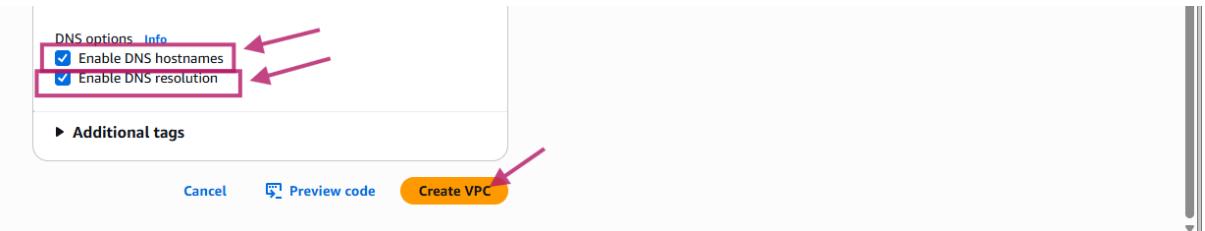
- **Screenshot of creating virtual private cloud network.**



This screenshot shows the 'Create VPC' wizard in the AWS VPC console. The left panel contains the 'VPC settings' section, which includes fields for 'Name tag auto-generation' (set to 'Auto-generate' with 'lab'), 'IPv4 CIDR block' (set to '10.0.0.0/16'), and 'Number of Availability Zones (AZs)' (set to '1'). The right panel shows a preview of the VPC structure, which includes a single VPC named 'lab-vpc' and two subnets in the 'us-east-1a' availability zone: 'lab-subnet-pub' (public) and 'lab-subnet-priv' (private). A blue arrow points to the 'VPC and more' button in the 'Resources to create' dropdown.

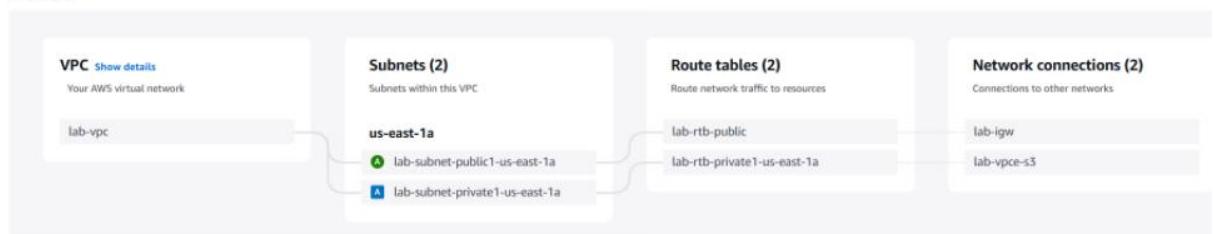
Screenshot of creating subnet 1 within the VPC.

This screenshot continues the 'Create VPC' wizard. The left panel shows the configuration for subnet creation, including 'Number of public subnets' (set to '1') and 'Number of private subnets' (set to '1'). Under 'Customize subnets CIDR blocks', the 'Public subnet CIDR block in us-east-1a' is set to '10.0.0.0/24' and the 'Private subnet CIDR block in us-east-1a' is set to '10.0.1.0/24'. The right panel shows the updated preview of the VPC structure, now including a single subnet named 'lab-subnet-pub' under the 'us-east-1a' availability zone. A blue arrow points to the '1' button in the 'Number of public subnets' dropdown.

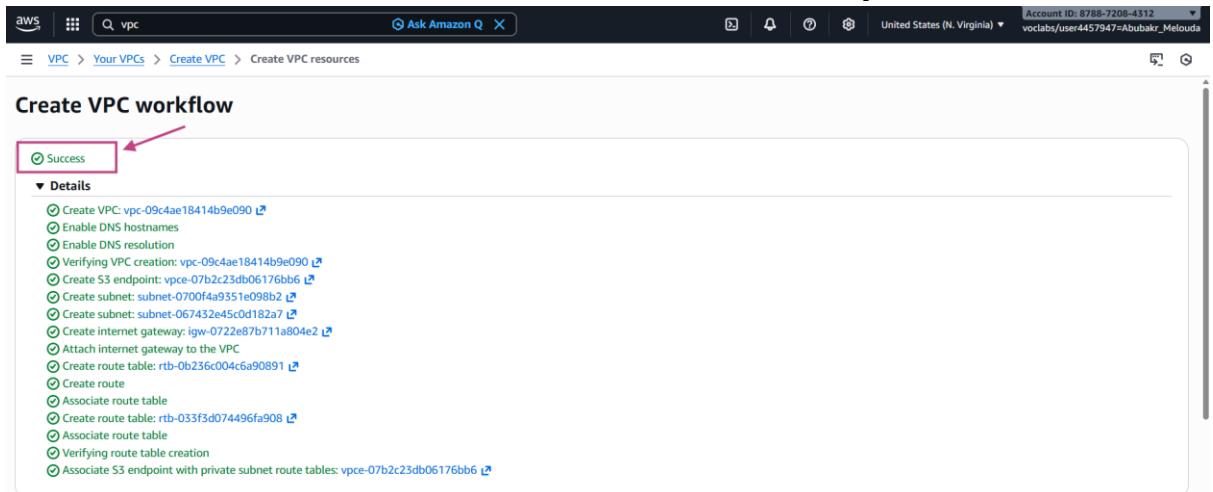


- Screenshot of network infrastructure overview before creation showing private and public subnet, routing table and network connection method.

Preview



Screenshot of VPC subnet 1 created successfully:



- Screenshot step-by-step creating additional public subnet.

Abubakr Melouda

The screenshot shows the AWS VPC Subnets page. On the left, there's a sidebar with sections like 'Virtual private cloud' (Subnets, highlighted with a red arrow), 'Security', and 'Network ACLs'. The main area displays a table of subnets:

Name	Subnet ID	State	VPC
lab-subnet-public1-us-east-1a	subnet-0700f4a9351e098b2	Available	vpc-09c4ae18414b9e09c
-	subnet-05c836aa0580f0513	Available	vpc-0346a1a418e9b5c59
-	subnet-0f48a86ae3af024f0	Available	vpc-0346a1a418e9b5c59
-	subnet-078b9ba8daf1f8c50	Available	vpc-0346a1a418e9b5c59
Work Public Subnet	subnet-00f374ed01a41532f	Available	vpc-0331d7551335792eb
-	subnet-0f63dc1fff8409319	Available	vpc-0346a1a418e9b5c59
lab-subnet-private1-us-east-1a	subnet-067432e45c0d182a7	Available	vpc-09c4ae18414b9e09c
-	subnet-056bb1ae74964807c	Available	vpc-0346a1a418e9b5c59
-	subnet-00ae804e97b61ae95	Available	vpc-0346a1a418e9b5c59

A red arrow points to the 'Create subnet' button at the top right of the table.

The screenshot shows the 'Create subnet' wizard. Step 1 is 'VPC'. It has a 'VPC ID' section with a dropdown labeled 'Select a VPC'. A search bar above it shows 'vpc-09c4ae18414b9e090 (lab-vpc)'. Below the dropdown, there are two options listed:

- vpc-09c4ae18414b9e090 (lab-vpc) 10.0.0.0/16
- vpc-0331d7551335792eb (Work VPC) 10.0.0.0/16

A red arrow points to the first option in the dropdown.

Subnet settings

Specify the CIDR blocks and Availability Zone for the subnet.

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

Tags - optional

Key	Value - optional
<input type="text" value="Name"/> X	<input type="text" value="lab-subnet-public2"/> X Remove

Add new tag
You can add 49 more tags.
Remove

Add new subnet

Cancel Create subnet ↓

VPC dashboard <

Subnets (1) Info

You have successfully created 1 subnet: subnet-0a59cb866d2a4aff7

Last updated less than a minute ago

Actions Create subnet

Subnets (1) Info

Find subnets by attribute or tag

Subnet ID : subnet-0a59cb866d2a4aff7 X | Clear filters

<input type="checkbox"/> Name	Subnet ID	State	VPC
<input type="checkbox"/> lab-subnet-public2	subnet-0a59cb866d2a4aff7	Available	vpc-09c4ae18414b9e09c

- Screenshot step-by-step creating additional private subnet.

Abubakr Melouda

The screenshot shows the 'Create subnet' wizard in the AWS VPC console. In the 'VPC' section, the VPC ID is selected as 'vpc-09c4ae18414b9e090 (lab-vpc)'. Under 'Associated VPC CIDRs', the IPv4 CIDR is set to '10.0.0.0/16'. In the 'Subnet settings' section, the subnet name is 'lab-subnet-private2', the availability zone is 'United States (N. Virginia) / us-east-1a (us-east-1b)', and the IPv4 subnet CIDR block is '10.0.3.0/24'. A note indicates that 256 IPs are available.

The screenshot shows the 'Subnets' page in the AWS VPC console. A green success message at the top states 'You have successfully created 1 subnet: subnet-0bc3ae0f8346cd729'. The table lists one subnet: 'lab-subnet-private2' with Subnet ID 'subnet-0bc3ae0f8346cd729', State 'Available', and VPC 'vpc-09c4ae18414b9e090'. The table has columns for Name, Subnet ID, State, and VPC.

Name	Subnet ID	State	VPC
lab-subnet-private2	subnet-0bc3ae0f8346cd729	Available	vpc-09c4ae18414b9e090

- Screenshot navigating to the routing table:

The screenshot shows the 'NAT gateways' page in the AWS VPC console. A red arrow points to the 'NAT gateways' link in the left sidebar under 'Virtual private cloud'. The main table displays a message: 'No NAT gateways found'. The table has columns for Name, NAT gateway ID, Connectivity..., State, and State message.

Name	NAT gateway ID	Connectivity...	State	State message
No NAT gateways found				

- Screenshot creating routing table:

Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.

VPC
The VPC to use for this route table.

Tags
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.

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="lab-rtb-private1-us-east-1a"/>

[Add new tag](#)

You can add 49 more tags.

[Cancel](#) [Create route table](#)

VPC dashboard

Route tables

rtb-074d3761ce10cf8b6 / lab-rtb-private1-us-east-1a

Route table rtb-074d3761ce10cf8b6 | lab-rtb-private1-us-east-1a was created successfully.

[Actions](#)

- Screenshot associating private subnet rules to “lab-subnet-private1-us-east-1a” and “lab-subnet-private2”:

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (2/4)

<input type="checkbox"/>	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input type="checkbox"/>	lab-subnet-public2	subnet-0fce3683c43cf4fea	10.0.2.0/24	-	Main (rtb-06ad0d9)
<input checked="" type="checkbox"/>	lab-subnet-private1-us-east-1a	subnet-0e6bd8025c58ef...	10.0.1.0/24	-	rtb-08dc13a8afee1
<input type="checkbox"/>	lab-subnet-public1-us-east-1a	subnet-0770765c36f4e7f...	10.0.0.0/24	-	rtb-017a8b447e80
<input checked="" type="checkbox"/>	lab-subnet-private2	subnet-0904294fd5dcb88...	10.0.3.0/24	-	rtb-074d3761ce10

Selected subnets

[Cancel](#) [Save associations](#)

The screenshot shows the AWS VPC Route Tables page. At the top, there's a success message: "You have successfully updated subnet associations for rtb-08dc13a8afee13e77 / lab-rtb-private1-us-east-1a." Below the message, it says "Route tables (1/7) Info" and "Last updated 9 minutes ago". There are "Actions" and "Create route table" buttons.

- Screenshot associating public subnet rules to “lab-subnet-public1-us-east-1a” and “lab-subnet-public2”:

The screenshot shows the AWS VPC Route Tables page with the "lab-rtb-public" route table selected (indicated by a red arrow). In the "Subnet associations" tab, there is one entry: "lab-subnet-public1-us-eas..." associated with "subnet-0770765c36f4e..." and "10.0.0.0/24". A red arrow points to the "Edit subnet associations" button.

Name	Route table ID	Explicit subnet associations
lab-rtb-public	rtb-017a8b447e803cc2e	subnet-0770765c36f4e7fa4 / lab-subnet-p...
Work Public Route Table	rtb-0e8ef610d5cafc307	subnet-050c40f7df14f756f / Work Public S...
-	rtb-06fdf715377108a9c	-
-	rtb-018121eaf23edb9bc	-
lab-rtb-private1-us-east-1a	rtb-074d3761ce10cf8b6	-
-	rtb-06ad0d9964319da34	-

rtb-017a8b447e803cc2e / lab-rtb-public

Details | Routes | **Subnet associations** | Edge associations | Route propagation | Tags

Explicit subnet associations (1)

Edit subnet associations

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
lab-subnet-public1-us-eas...	subnet-0770765c36f4e...	10.0.0.0/24	-

The screenshot shows the 'Edit subnet associations' page for a specific route table. In the 'Available subnets' table, three subnets are selected: 'lab-subnet-public2', 'lab-subnet-public1-us-east-1a', and 'lab-subnet-public1-us-east-1a'. These selected subnets are highlighted with blue outlines. Arrows point from these subnets to the 'Selected subnets' section below. The 'Selected subnets' section contains two entries: 'subnet-0770765c36f4e7fa4 / lab-subnet-public1-us-east-1a' and 'subnet-0dce3683c43cf4fea / lab-subnet-public2'. A large red arrow points from the 'Selected subnets' section down to the 'Save associations' button at the bottom right.

Available subnets (2/4)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/> lab-subnet-public2	subnet-0dce3683c43cf4fea	10.0.2.0/24	-	Main (rtb-06ad0d9964319da34)
<input type="checkbox"/> lab-subnet-private1-us-e...	subnet-0e6bd8025c58ef...	10.0.1.0/24	-	rtb-08dc13a8afee13e77 / lab-rtb-priv...
<input checked="" type="checkbox"/> lab-subnet-public1-us-ea...	subnet-0770765c36f4e7f...	10.0.0.0/24	-	rtb-017a8b447e803cc2e / lab-rtb-public
<input type="checkbox"/> lab-subnet-private2	subnet-0904294fd5dc8...	10.0.3.0/24	-	rtb-08dc13a8afee13e77 / lab-rtb-priv...

Selected subnets

subnet-0770765c36f4e7fa4 / lab-subnet-public1-us-east-1a subnet-0dce3683c43cf4fea / lab-subnet-public2

Save associations

- Screenshots of creating Security Group for Web Server access:

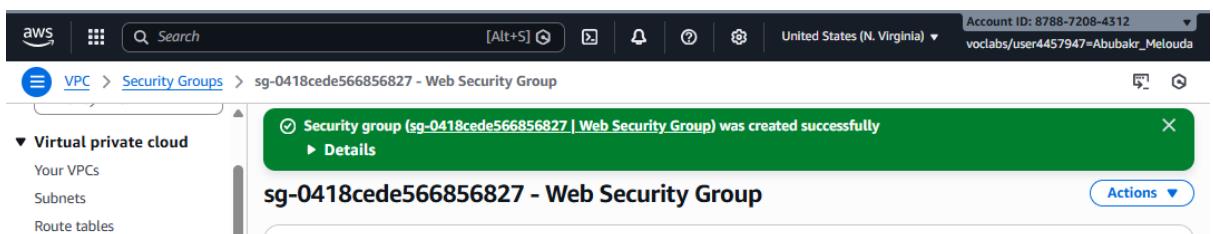
The screenshot shows the 'Security Groups' page. On the left sidebar, under the 'Security' section, 'Security groups' is selected. A red arrow points to this selection. In the main content area, there is a table of existing security groups. At the top right of the table, there is a yellow 'Create security group' button, which also has a red arrow pointing to it. The table columns include Name, Security group ID, Security group name, and VPC ID.

Security Groups (4) Info

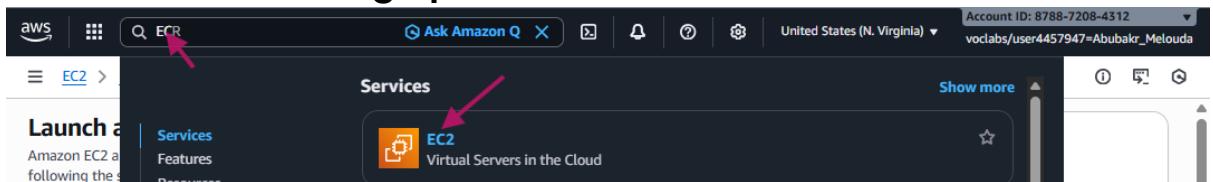
Name	Security group ID	Security group name	VPC ID
-	sg-054b1a5905e603b6f	Ec2SecurityGroup	vpc-018dad5
-	sg-0e7c1abbd556e5288	default	vpc-049a4df9
-	sg-0d64ec95ccb0cb54a	default	vpc-0e63f491
-	sg-06010a0f73aebd223	default	vpc-018dad5

Create security group

The screenshot shows the 'Create security group' page in the AWS VPC console. In the 'Basic details' section, the security group name is 'Web Security Group' and the description is 'Enable HTTP access'. The VPC is set to 'vpc-0e63f491381660ff9 (lab-vpc)'. In the 'Inbound rules' section, there is one rule: 'HTTP' (Protocol: TCP, Port range: 80) from '0.0.0.0/0' with the description 'Permit web requests'. The 'Add rule' button is visible.



- Screenshots of setting up a webserver:



Abubakr Melouda

Screenshot of the AWS EC2 console showing the main dashboard.

Left sidebar:

- EC2
- Dashboard
- EC2 Global View
- Events
- Instances**
 - Instances
 - Instance Types
 - Launch Templates
 - Spot Requests
 - Savings Plans
 - Reserved Instances
 - Dedicated Hosts
 - Capacity Reservations
 - Capacity Manager
- Images**
 - AMIs
 - AMI Catalog
- Elastic Block Store**
 - Volumes

Center pane:

Resources

You are using the following Amazon EC2 resources in the United States (N. Virginia) Region:

Instances (running)	1	Auto Scaling Groups	0
Capacity Reservations	0	Dedicated Hosts	0
Elastic IPs	1	Instances	1
Key pairs	1	Load balancers	0
Placement groups	0	Security groups	5
Snapshots	0	Volumes	1

Launch instance

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

Service health

AWS Health Dashboard

Region

United States (N. Virginia)

Account attributes

Default VPC: vpc-049a4df9a8eb0d25f

Settings

Data protection and security
Allowed AMIs
Zones
EC2 Serial Console
Default credit specification
EC2 console preferences

Explore AWS

Get Up to 40% Better Price Performance

T4g instances deliver the best price performance for burstable general purpose workloads in Amazon EC2. Learn more

Enable Best Price-Performance with AWS Graviton2

AWS Graviton2-powered EC2

Screenshot of the AWS EC2 "Launch an instance" wizard.

Step 1: Name and tags

Name: Web Server 1

Step 2: Application and OS Images (Amazon Machine Image)

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

Recent AMIs: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SU, > (with a red arrow pointing to it)

Quick Start: Amazon Linux (highlighted with a red arrow), macOS, Ubuntu, Windows, Red Hat, SU, >

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI

Free tier eligible

Virtualization: hvm
ENAv2 enabled: true
Root device type: ebs

Description

Amazon Linux 2023 (kernel-6.1) 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 2023.9.20251208.0 x86_64 HVM kernel-6.1

Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.9.2...read more

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier note: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

Buttons:

- Cancel
- Launch instance
- Preview code

The screenshot shows the AWS EC2 'Launch an instance' wizard. The process is divided into several steps:

- Step 1: Instance type** (Completed)
 - Selected instance type: t2.micro
 - Description: Family: t2 1 vCPU 1 GiB Memory Current generation: true
 - Pricing: 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
- Step 2: Key pair (login)** (Completed)
 - Key pair name: vockey
 - Options: Create new key pair
- Step 3: Network settings** (Completed)
 - VPC - required: lab-vpc (selected)
 - Subnet: lab-subnet-public2 (selected)
 - Auto-assign public IP: Enabled
 - Firewall (security groups):
 - Create security group
 - Select existing security group (selected)
- Step 4: Summary** (Completed)
 - Number of instances: 1
 - Software Image (AMI): Amazon Linux 2023 AMI 2023.9.2...read more
 - Virtual server type (instance type): t2.micro
 - Firewall (security group): Web Security Group
 - Storage (volumes): 1 volume(s) - 8 GiB
- Free tier information** (Info):
 - In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.
- Actions** (Bottom right):
 - Cancel
 - Launch instance
 - Preview code

Abubakr Melouda

Screenshot of the AWS EC2 Launch an instance wizard.

Step 1: Firewall (security groups)

Selected: Select existing security group
Common security groups: Web Security Group sg-0418cede566856827 (VPC: vpc-0e63f491381660f9)

Step 2: Configure storage

Root volume: 8 GiB gp3 (Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage)

Step 3: Metadata IPv6 endpoint

Metadata version: V2 only (token required)
Warning: For V2 requests, you must include a session token in all instance metadata requests. Applications or agents that use V1 for instance metadata access will break.

Step 4: User data - optional

User data script:#!/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

User data has already been base64 encoded

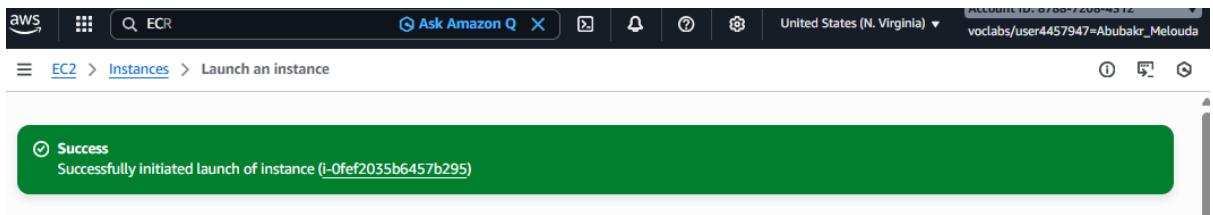
Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.9.2...read more
Virtual server type (instance type): t2.micro
Firewall (security group): Web Security Group
Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

Launch instance



Screenshot of successful test access to the Web server access in Zone B:

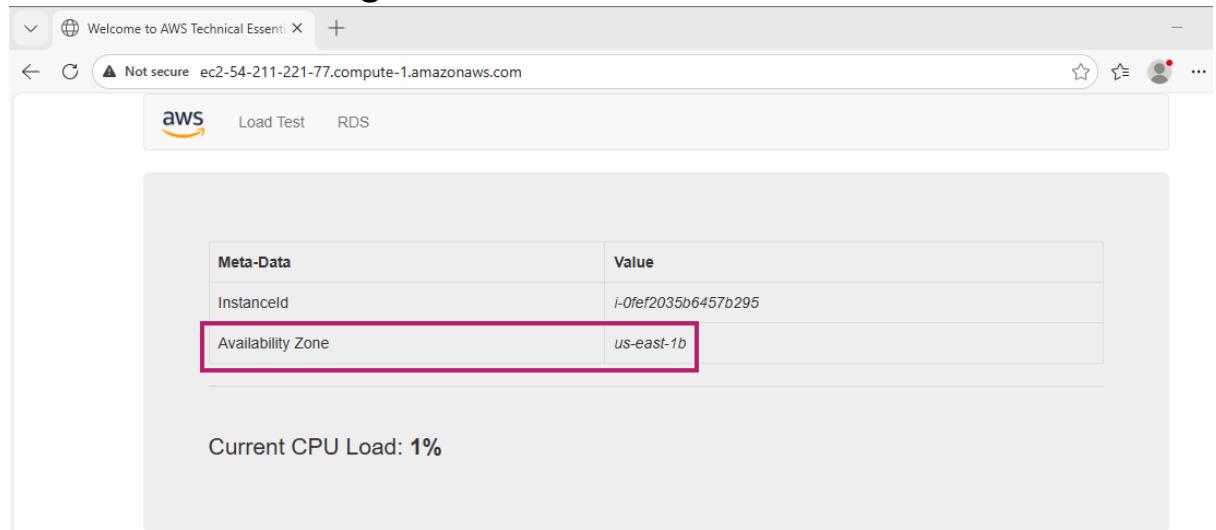
Instance Status	Check Status	Action
Initializing	2/2 checks passed	
Running	2/2 checks passed	
Running	2/2 checks passed	

Testing Web Server access using public IPv4 DNS.

i-0fef2035b6457b295 (Web Server 1)

Details	Status and alarms	Monitoring	Security	Networking	Storage	Tags
Instance summary Instance ID : i-0fef2035b6457b295 IPV6 address : Hostname type : IP name: ip-10-0-2-195.ec2.internal Answer private resource DNS name : Public IPv4 address : 54.211.221.77 open address Instance state : Running Private IP DNS name (IPv4 only) : ip-10-0-2-195.ec2.internal Instance type : t2.micro	Public IPv4 addresses Public DNS copied ec2-54-211-221-77.compute-1.amazonaws.com open address	Elastic IP addresses : -				

Screenshot of testing access to the Web server:



The last step confirms the successful deployment of a web server in the public subnet “lab-subnet-public2”. The Apache server is running with PHP support serving content from the `/var/www/html/` directory as configured in the EC2 instance user data script. The displayed AWS logo and instance metadata verify that all VPC networking components, security groups, route tables and NAT Gateway are successfully configured.