

Launching VPC Resources

Project Overview

This project marks a major milestone in my AWS networking journey - bringing my VPC to life by launching actual EC2 instances! After building the networking foundation in previous projects, I successfully deployed a **public EC2 instance** accessible from the internet and a **private EC2 instance** isolated for backend operations. I also discovered the VPC Wizard, a powerful tool that can create complete VPC architectures in minutes with automatic resource naming and configuration.

Project Duration: Approximately 60 minutes

Difficulty Level: Mildly Spicy

AWS Region Used: eu-west-3 (Paris)

Project Series: Part 4 of NextWork VPC Challenge

Table of Contents

- [What I Built](#)
 - [Technologies & Concepts](#)
 - [Step-by-Step Implementation](#)
 - [Step 1: Create Key Pair](#)
 - [Step 2: Launch Public EC2 Instance](#)
 - [Step 3: Launch Private EC2 Instance](#)
 - [Step 4: Use VPC Wizard](#)
 - [Cleanup](#)
 - [Conclusion](#)
-

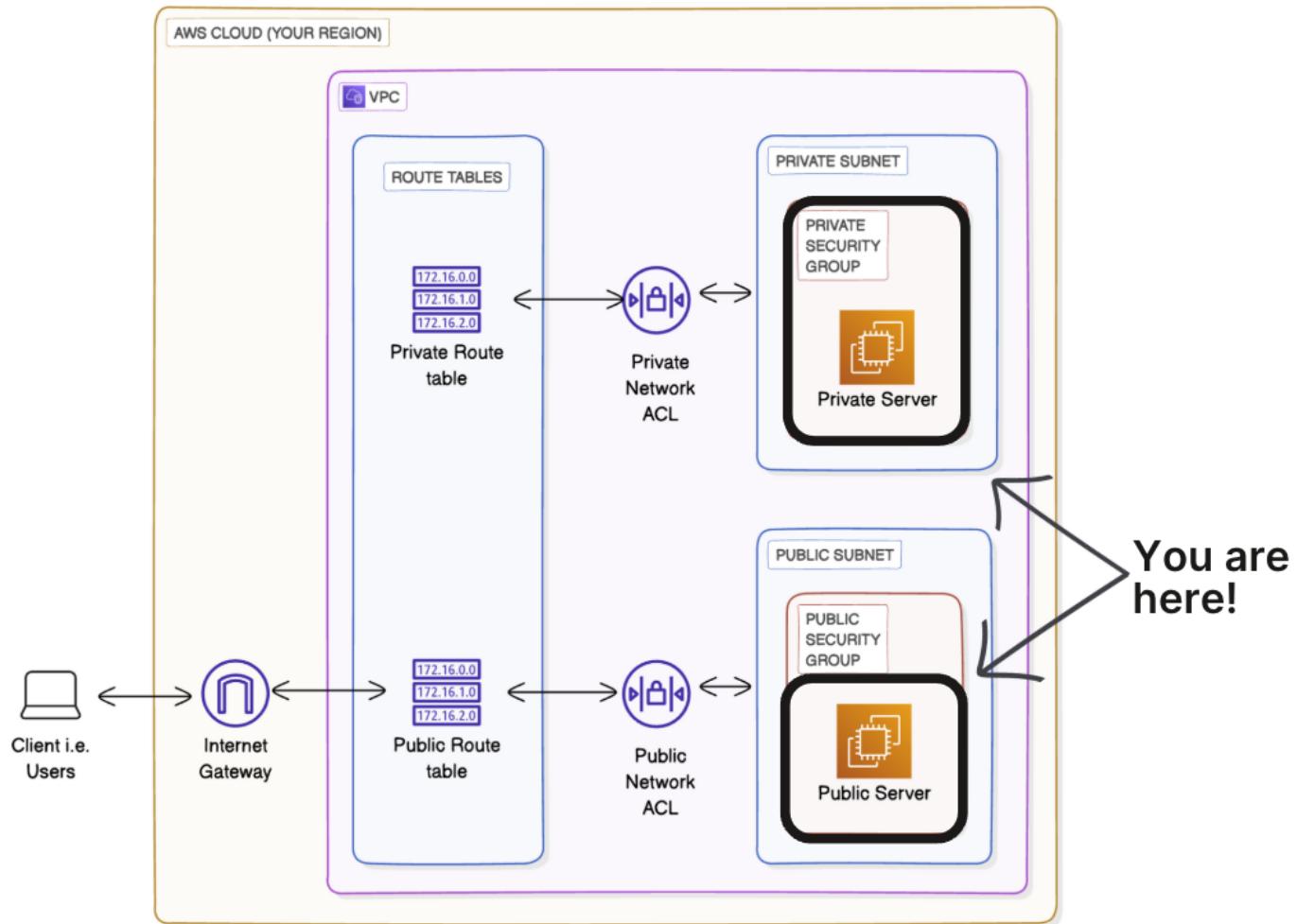
What I Built

A complete VPC with running EC2 instances and explored the VPC Wizard:

Manual VPC Infrastructure (NextWork VPC):

- **Public EC2 Instance:** NextWork Public Server (t2.micro) in eu-west-3a
 - Public IPv4: 51.44.82.224
 - Private IPv4: 10.0.0.172
 - Amazon Linux 2023 AMI
 - Accessible from the internet via HTTP
- **Private EC2 Instance:** NextWork Private Server (t2.micro) in eu-west-3b
 - No Public IP (secure!)
 - Private IPv4: 10.0.1.237
 - Amazon Linux 2023 AMI
 - Only accessible from Public Server via SSH

- **Key Pair:** NextWork Key Pair (RSA, .pem format)
 - Used for secure SSH access to both instances
- **Security Groups:**
 - NextWork Security Group (Public) - Allows HTTP from anywhere
 - NextWork Private Security Group - Allows SSH only from Public Security Group



VPC Wizard-Created Infrastructure (nextwork-vpc):

- Complete VPC with auto-generated resources
- 2 Subnets (1 public, 1 private) across 1 Availability Zone
- 3 Route Tables (1 public, 1 private, 1 main)
- Internet Gateway
- Automatic resource naming with "nextwork" prefix

Key Achievement: I now understand the difference between manual VPC creation (full control, more time) vs VPC Wizard (quick setup, best practices baked in).

Technologies & Concepts

AWS Services Used

- **Amazon EC2** - Elastic Compute Cloud for virtual servers

- **Amazon VPC** - Virtual Private Cloud for isolated networking
- **Key Pairs** - RSA cryptographic keys for secure instance access
- **Security Groups** - Resource-level firewalls
- **AMI** - Amazon Machine Image (operating system template)

Key Concepts Learned

1. EC2 Instances

Virtual servers running in the cloud. Think of them as computers you can access remotely over the internet.

Key Components:

- **AMI (Amazon Machine Image)**: The operating system and pre-installed software (like buying a computer with Windows or macOS)
- **Instance Type**: The hardware specifications (CPU, RAM, storage) - like choosing between a laptop vs desktop
- **Key Pair**: Your secure key to access the server via SSH

2. Key Pairs and SSH

Key Pairs consist of two cryptographic keys:

- **Public Key**: Installed on the EC2 instance
- **Private Key**: Stored on your computer (.pem file)

SSH (Secure Shell) is the protocol used to connect:

- Encrypts all communication between you and the server
- Prevents unauthorized access
- Standard method for managing remote servers

Real-world analogy: Think of it like a hotel room - the public key is the lock on the door, and your private key is the key card only you have.

3. Public vs Private Instances

Public EC2 Instance:

- Has a public IPv4 address (accessible from internet)
- Lives in a public subnet with route to Internet Gateway
- Use case: Web servers, load balancers, bastion hosts
- Example: 51.44.82.224 can be reached from anywhere

Private EC2 Instance:

- Has NO public IP address (only private IP)
- Lives in a private subnet with no internet route
- Use case: Databases, application servers, sensitive workloads
- Example: 10.0.1.237 can only be reached from within the VPC

4. Security Group Source Types

For Public Instance:

- Source: 0.0.0.0/0 (Anywhere) - Anyone can access
- Acceptable for HTTP traffic (port 80) - public websites need this!

For Private Instance:

- Source: Security Group ID (e.g., sg-01d9df52af4817bbb)
- Only resources with that security group can connect
- Provides defense in depth - even if someone breaches the public server, they can't directly access the private server from the internet

5. VPC Wizard Benefits

Manual Creation (Parts 1-3):

- Full control over every component
- Learn deeply by doing each step
- Time-consuming (multiple projects)
- Must manually name each resource

VPC Wizard:

- Creates complete VPC in minutes
- Follows AWS best practices automatically
- Auto-generates consistent resource names
- Perfect for quick testing environments
- Configures route tables and associations automatically

6. Instance State and Public IPs

Important behavior:

- When you STOP an instance, it loses its public IP
- When you START it again, it gets a NEW public IP
- Private IPs remain the same
- Solution: Use Elastic IPs for persistent public addresses (costs apply when not attached to running instance)

Step-by-Step Implementation

Step 1: Create Key Pair

What I did:

Before launching any EC2 instances, I needed to create a key pair for secure SSH access.

Understanding Key Pairs

Why do we need key pairs?

Imagine if anyone could access your server just by knowing its IP address - that would be a security nightmare! Key pairs ensure that only authorized people (with the private key) can access your instances.

How they work:

1. AWS installs the **public key** on your EC2 instance
2. You download and keep the **private key** (.pem file) secure on your computer
3. When connecting via SSH, your private key proves you're authorized
4. The connection is encrypted - no one can intercept your session

Creating the Key Pair

Key Pair Configuration:

Create key pair

Key pair name
Key pairs allow you to connect to your instance securely.

NextWork Key Pair

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

RSA
RSA encrypted private and public key pair

ED25519
ED25519 encrypted private and public key pair

Private key file format

.pem
For use with OpenSSH

.ppk
For use with PuTTY

⚠️ When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** [Learn more ↗](#)

[Cancel](#) [Create key pair](#)

⚠️ Critical Warning:

"When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.**"

Why .pem format?

- .pem (Privacy Enhanced Mail) works with OpenSSH on Mac/Linux
- .ppk format is for PuTTY on Windows
- I chose .pem for maximum compatibility

Security Best Practice:

After downloading, I stored the key in a secure location and set proper permissions:

```
chmod 400 NextWork-Key-Pair.pem
```

This ensures only I can read the file, preventing unauthorized access.

Step 2: Launch Public EC2 Instance

What I did:

I launched my first EC2 instance in the public subnet, making it accessible from the internet.

Instance Configuration

Name and AMI Selection:

Name and tags info

Name

 [Add additional tags](#)

AMI Selection:

▼ 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  | Ubuntu  | Windows  | Red Hat  | SUSE Linux  | Debian  |
|---|---|--|--|---|---|

 [Browse more AMIs](#)
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI
ami-078abd88811000d7e (64-bit (x86), uefi-preferred) / ami-0387f15c965e9e817 (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs Free tier eligible ▾

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

| Architecture | Boot mode | AMI ID | Publish Date | Username |
|----------------|----------------|-----------------------|--------------|---|
| 64-bit (x86) ▾ | uefi-preferred | ami-078abd88811000d7e | 2025-12-03 | ec2-user Verified provider |

Why Amazon Linux 2023?

- Designed specifically for AWS (best performance and integration)
- Free tier eligible (no charges!)
- 5 years of long-term support
- Comes with AWS CLI and tools pre-installed
- Regular security updates

Instance Type

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

Instance type

| | |
|--|--|
| t2.micro | Free tier eligible |
| Family: t2 1 vCPU 1 GiB Memory Current generation: true On-Demand RHEL base pricing: 0.0276 USD per Hour | On-Demand SUSE base pricing: 0.0132 USD per Hour On-Demand Linux base pricing: 0.0132 USD per Hour |
| On-Demand Ubuntu Pro base pricing: 0.015 USD per Hour | On-Demand Windows base pricing: 0.0178 USD per Hour |

All generations [Compare instance types](#)

Additional costs apply for AMIs with pre-installed software

What does "burstable" mean?

t2.micro instances can "burst" above baseline CPU performance when needed, using CPU credits. Perfect for workloads with occasional spikes (like web servers with variable traffic).

Key Pair Selection

▼ 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

NextWork Key Pair [▼](#) [Create new key pair](#)

- **Key pair name:** NextWork Key Pair (the one I just created!)

Important: Without selecting a key pair, I wouldn't be able to SSH into my instance later.

Network Settings

▼ Network settings [Info](#)

VPC - required | [Info](#)

vpc-0fed86c62b1c0a9d6 (NextWork VPC)
10.0.0.0/16 [▼](#) [Create new VPC](#)

Subnet | [Info](#)

subnet-04daee107878ed50e NextWork Public Subnet
VPC: vpc-0fed86c62b1c0a9d6 Owner: 056481036163 Availability Zone: eu-west-3a (euw3-az1)
Zone type: Availability Zone IP addresses available: 251 CIDR: 10.0.0.0/24 [▼](#) [Create new subnet](#)

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 [▼](#)

NextWork Security Group sg-01d9df52af4817bbb [X](#) [▼](#) [Compare security group rules](#)
VPC: vpc-0fed86c62b1c0a9d6

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

► Advanced network configuration

Why enable auto-assign public IP?

Without a public IP, my instance would be unreachable from the internet, even though it's in a public subnet!

Firewall (Security Groups):

- **Option selected:** Select existing security group
- **Security group:** NextWork Security Group (sg-01d9df52af4817bbb)

This security group already has HTTP (port 80) configured from my previous project, allowing web traffic from anywhere (0.0.0.0/0).

Storage Configuration

Configure storage [Info](#) [Advanced](#)

1x 8 GiB gp3 Root volume, 3000 IOPS, Not encrypted

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage [X](#)

[Add new volume](#)

① Click refresh to view backup information [Edit](#)

The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

0 x File systems [Edit](#)

Advanced details [Info](#)

Networking Details

i-0a7bdb810574f2de0 (NextWork Public Server)

| Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags |
|--|-------------------|------------|----------|--|---------------------|---|
| VPC ID vpc-0fed86c62b1c0a9d6 (NextWork VPC) | | | | Subnet ID subnet-04daee107878ed50e (NextWork Public Subnet) | | Availability zone eu-west-3a |
| Availability zone ID euw3-az1 | | | | Outpost ID - | | |
| IP addresses Info | | | | Private IPv4 addresses 10.0.0.172 | IPv6 addresses - | |
| Public IPv4 address 51.44.82.224 open address | | | | Carrier IP addresses (ephemeral) - | | |
| Secondary private IPv4 addresses - | | | | | | |

Success! My public server is now running and accessible from the internet! 🎉

Step 3: Launch Private EC2 Instance

What I did:

I launched a second EC2 instance in the private subnet with restricted SSH access - only allowing connections from the public security group.

Instance Configuration

Name and Basic Settings:

Name and tags [Info](#)

Name
 [Add additional tags](#)

- **Name:** NextWork Private Server

▼ 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](#).

Recents **Quick Start**



Amazon Linux

aws



Ubuntu

ubuntu®



Windows

Microsoft



Red Hat

Red Hat



SUSE Linux

SUSE



Debian

debian

🔍 [Browse more AMIs](#)
 Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI
 ami-078abd88811000d7e (64-bit (x86), uefi-preferred) / ami-0387f15c965e9e817 (64-bit (Arm), uefi)
 Virtualization: hvm ENA 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

| Architecture | Boot mode | AMI ID | Publish Date | Username |
|----------------|----------------|-----------------------|--------------|----------|
| 64-bit (x86) ▾ | uefi-preferred | ami-078abd88811000d7e | 2025-12-03 | ec2-user |

Verified provider

- **AMI:** Amazon Linux 2023 AMI (same as public instance)

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

Instance type

t2.micro Free tier eligible
 Family: t2 1 vCPU 1 GiB Memory Current generation: true On-Demand RHEL base pricing: 0.0276 USD per Hour
 On-Demand SUSE base pricing: 0.0132 USD per Hour On-Demand Linux base pricing: 0.0132 USD per Hour
 On-Demand Ubuntu Pro base pricing: 0.015 USD per Hour On-Demand Windows base pricing: 0.0178 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

NextWork Key Pair

Create new key pair

Instance Type and Key Pair:

- **Instance type:** t2.micro
- **Key pair:** NextWork Key Pair (reusing the same key pair!)

Can I use the same key pair for multiple instances?

Yes! This makes management easier - one key to access both instances. However, security consideration: anyone with this key can access ALL instances using it.

Network Settings & Creating the Private Security Group - The Critical Part!

▼ Network settings [Info](#)

VPC - required | [Info](#)

vpc-0fed86c62b1c0a9d6 (NextWork VPC)
10.0.0.0/16

Subnet | [Info](#)

subnet-0e277487d5cb52651 NextWork Private Subnet
VPC: vpc-0fed86c62b1c0a9d6 Owner: 056481036163 Availability Zone: eu-west-3b (euw3-az2)
Zone type: Availability Zone IP addresses available: 251 CIDR: 10.0.1.0/24

Create new subnet [↗](#)

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

Security group name - required

Nextwork Private Security Group

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-:/()#,@!+=;&,\$^*

Description - required | [Info](#)

A Security Group For Nextwork Private Subnet

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, sg-01d9df52af4817bbb) [Remove](#)

| Type Info | Protocol Info | Port range Info |
|------------------------------------|---|---|
| ssh | TCP | 22 |
| Source type Info | Source Info | Description - optional Info |
| Custom | Add CIDR, prefix list or security group sg-01d9df52af4817bbb X | e.g. SSH for admin desktop |

[Add security group rule](#)

► Advanced network configuration

What does this mean?

Instead of allowing SSH from "Anywhere" (0.0.0.0/0), I'm restricting SSH access to ONLY resources that have the **NextWork Public Security Group** attached.

⚠ Yellow Warning Explained:

AWS initially warns: "Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance."

By changing the source to a security group ID, I eliminated this warning and implemented proper security!

Security Architecture:



Defense in Depth:

1. Private subnet has no internet route (architectural)
2. Network ACL controls subnet-level traffic
3. Security group allows SSH only from public security group
4. Even if public server is compromised, attacker can't SSH from their own computer directly to private server

Storage Configuration

The screenshot shows the 'Configure storage' section for a Lambda function. It displays a single volume configuration: 1x 8 GiB gp3. A note indicates that free-tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. There is also an 'Add new volume' button and a note about refreshable backup information.

Private Instance Networking Details

The screenshot shows the networking details for instance i-0f9cd610c24bc347a. It includes sections for VPC ID, Availability zone ID, IP addresses (with a note about auto-assigned public IP), Subnet ID, Outpost ID, and Carrier IP addresses (ephemeral). There are tabs for Details, Status and alarms, Monitoring, Security, Networking (selected), Storage, and Tags.

⚠ Important Discovery:

Even though I launched in a private subnet, the instance got a public IP because I left "Auto-assign public IP" enabled!

However, because the private subnet's route table has NO route to the Internet Gateway, this public IP is effectively useless - the instance still can't communicate with the internet.

Best practice: For private instances, disable auto-assign public IP to avoid confusion.

Success! My private server is running and secured with proper SSH restrictions! 🎉

Step 4: Use VPC Wizard

What I did:

After manually creating VPCs in previous projects (which took hours!), I discovered the VPC Wizard - AWS's automated tool that creates complete VPC architectures in minutes.

Why Use the VPC Wizard?

Manual VPC Creation (Parts 1-3):

- Learn every component deeply
- Full control over configuration
- Time-consuming (multiple steps across projects)
- Easy to make mistakes
- Must manually name and tag everything

VPC Wizard:

- Creates VPC in minutes
- Follows AWS best practices
- Auto-generates consistent resource names
- Perfect for testing/development environments
- Less control over individual components

Wizard Configuration

Main 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
nextwork

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 ▾

► **Encryption settings - optional**

Why auto-generate names?

AWS will automatically prefix all resources with "nextwork-" (e.g., nextwork-vpc, nextwork-subnet-public1, nextwork-rtb-public), making it easy to identify which resources belong together!

Availability Zones and Subnets

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

First availability zone
euw3-az1 (eu-west-3a) ▾

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 eu-west-3a
10.0.0.0/24 256 IPs

Private subnet CIDR block in eu-west-3a
10.0.1.0/24 256 IPs

AWS enforces best practices! When you select 2 AZs, AWS requires at least 1 public subnet per AZ for high availability. You can't create just 1 public subnet across 2 AZs.

Why /24 instead of /20?

Default wizard uses /20 (4,096 IPs per subnet), but I changed to /24 (256 IPs) to match my manual VPC configuration and conserve IP space.

NAT Gateways and VPC Endpoints

NAT gateways (\$) - updated [Info](#)

NAT gateway allows private resources to access the internet from any availability zone within a VPC, providing a single managed internet exit point for the entire region. Additional charges apply.

[None](#) | [Regional - new](#) | [Zonal](#)**Introducing regional NAT gateway**

AWS now offers a multi-AZ NAT Gateway, eliminating the need for separate NAT Gateways across availability zones.

X

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**What are NAT Gateways?**

NAT (Network Address Translation) Gateways allow private subnet resources to access the internet for updates/patches while blocking inbound traffic from the internet.

Cost consideration: NAT Gateways cost ~\$0.045/hour + data transfer charges. For learning, I skipped this.

What are VPC Endpoints?

VPC Endpoints allow private connectivity to AWS services (like S3) without going through the internet. The wizard offers S3 Gateway endpoint by default.

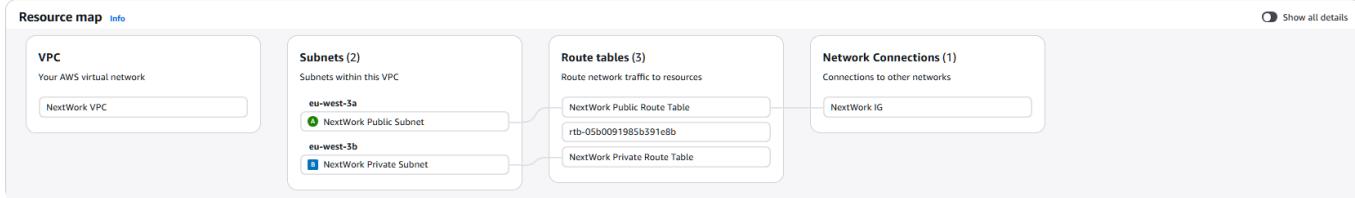
Why skip for now?

We'll explore VPC endpoints in detail in a later project of this series!

DNS Options**What do these do?**

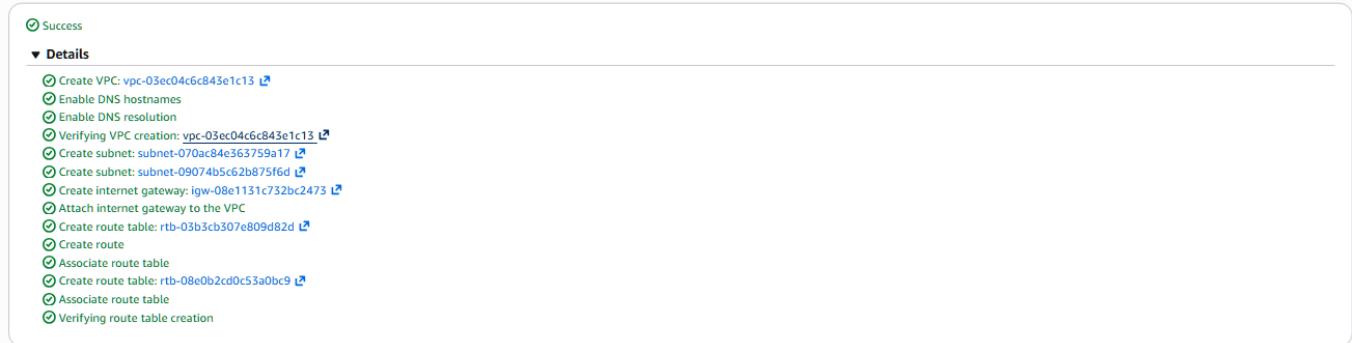
- **DNS hostnames:** Instances get human-readable names like `ec2-13-38-100-157.eu-west-3.compute.amazonaws.com`
- **DNS resolution:** AWS translates these hostnames to IP addresses automatically

Resource Map Preview**(Single AZ):**



Wizard Execution

Create VPC workflow



Both VPCs Running

| Instances (2/2) Info | | | | | | | | | | |
|---|-------------------------|---------------------|---|---------------|-----------------------------------|-------------------------------|-------------------|-----------------|-----------------|------------|
| Find Instance by attribute or tag (case-sensitive) All states | | | | | | | | | | |
| <input checked="" type="checkbox"/> | Name O | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 DNS | Public IPv4 ... | Elastic IP |
| <input checked="" type="checkbox"/> | NextWork Public Server | i-0a7bdb810574f2de0 | Running Q Q | t2.micro | 2/2 checks passed | View alarms + | eu-west-3a | – | 51.44.82.224 | – |
| <input checked="" type="checkbox"/> | NextWork Private Server | i-0f9cd610c24bc347a | Running Q Q | t2.micro | 2/2 checks passed | View alarms + | eu-west-3b | – | 13.38.100.157 | – |

Last updated 2 minutes ago [C](#) [Connect](#) [Instance state](#) [Actions](#) [Launch instances](#)

Stop instance Start instance Reboot instance Hibernate instance Terminate (delete) instance

Monitoring Security group

Sabiled NextWork Sec

Gabled Nextwork Priv

I now have EC2 instances running in both VPCs:

- **NextWork Public Server** (i-0a7bdb810574f2de0) - Running in NextWork VPC, eu-west-3a
- **NextWork Private Server** (i-0f9cd610c24bc347a) - Running in NextWork VPC, eu-west-3b

Both instances are healthy with 2/2 status checks passed!

Cleanup

Deletion Order

Important: Delete resources in the correct order to avoid dependency issues!

1. **Terminate EC2 Instances** (they're attached to VPCs)
2. **Delete NextWork VPC** (cascade deletes subnets, route tables, etc.)
3. **Delete nextwork-vpc** (wizard-created VPC)

Step 1: Terminate EC2 Instances

Terminate (delete) instances

⚠️ On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated. Storage on any local drives will be lost.

Are you sure you want to terminate these instances?

| Instance ID | Termination protection |
|--|--|
| <input type="checkbox"/> i-0a7bdb810574f2de0 (NextWork Public Server) | <input checked="" type="checkbox"/> Disabled |
| <input type="checkbox"/> i-0f9cd610c24bc347a (NextWork Private Server) | <input checked="" type="checkbox"/> Disabled |

To confirm that you want to delete the instances, choose the terminate button below. Instances with termination protection enabled will not be terminated. Terminating the instance cannot be undone.

Skip OS shutdown
This option skips the graceful OS shutdown process. Use only when your instance must be stopped immediately, such as during an emergency or failover.

Skip OS shutdown

[Cancel](#) [Terminate \(delete\)](#)

| Instances (2) Info | | Last updated less than a minute ago | Connect | Instance state ▾ | Actions ▾ | Launch instances | ▼ |
|--|---------------------|--------------------------------------|-------------------------|----------------------------------|-------------------------------|----------------------------------|-------------|
| | | All states | | | | | |
| Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 |
| <input type="checkbox"/> NextWork Public Server | i-0a7bdb810574f2de0 | Terminated | t2.micro | - | View alarms + | eu-west-3a | - |
| <input type="checkbox"/> NextWork Private Server | i-0f9cd610c24bc347a | Terminated | t2.micro | - | View alarms + | eu-west-3b | - |

Step 2: Delete NextWork VPC

Delete VPC

Will be deleted
This VPC will be deleted permanently and cannot be recovered later:

| Name | VPC ID | State |
|---------------------------------------|--|---|
| <input type="checkbox"/> NextWork VPC | <input type="checkbox"/> vpc-0fed86c62b1c0a9d6 | <input checked="" type="checkbox"/> Available |

Will also be deleted
The following 9 resources will also be deleted permanently and cannot be recovered later:

| Name | Resource ID | State |
|------------------------------|---------------------------------------|---|
| NextWork IG | igw-0f9c8a9e8964a441b | <input checked="" type="checkbox"/> Available |
| NextWork Private NACL | acl-0aae34fac068e401a | - |
| NextWork Public NACL | acl-0937b780d62ae88b2 | - |
| NextWork Public Route Table | rtb-0a34d888417c99873 | - |
| NextWork Private Route Table | rtb-05eb8a6d99d87cd33 | - |

To confirm deletion, type **delete** in the field:

[Cancel](#) [Delete](#)

Why so many resources?

Deleting a VPC cascades to all dependent resources. This is why we manually created these in Projects 1-3 - to understand what the VPC actually contains!

Step 3: Delete nextwork-vpc

The screenshot shows the 'Delete VPC' dialog box. At the top, it says 'Delete VPC' and has a close button (X). Below that, a section titled 'Will be deleted' indicates that the VPC will be deleted permanently and cannot be recovered later. It shows details for the VPC: Name (nextwork-vpc), VPC ID (vpc-03ec04c6c843e1c13), and State (Available). A large table below lists 5 resources that will also be deleted: nextwork-igw, nextwork-rtb-public, nextwork-rtb-private1-eu-west-3a, nextwork-subnet-public1-eu-west-3a, and nextwork-subnet-private1-eu-west-3a. Each resource has a Resource ID link and a State column showing they are available. At the bottom, there is a field to type 'delete' to confirm, and buttons for 'Cancel' and 'Delete'.

| Name | Resource ID | State |
|-------------------------------------|--|---|
| nextwork-igw | igw-08e1131c732bc2473 | <input checked="" type="checkbox"/> Available |
| nextwork-rtb-public | rtb-03b3cb307e809d82d | - |
| nextwork-rtb-private1-eu-west-3a | rtb-08e0b2cd0c53a0bc9 | - |
| nextwork-subnet-public1-eu-west-3a | subnet-070ac84e363759a17 | <input checked="" type="checkbox"/> Available |
| nextwork-subnet-private1-eu-west-3a | subnet-09074b5c62b875f6d | <input checked="" type="checkbox"/> Available |

Conclusion

This project marked a significant milestone in my AWS learning journey. After spending Parts 1-3 building the networking foundation (VPC, subnets, route tables, internet gateway, security groups, and NACLs), I finally got to **deploy actual resources** into that infrastructure. In **Part 5: Testing VPC Connectivity**, I'll test the network architecture by connecting to instances and verifying that the security configuration works as designed.

Project Completed: January 2026

Author: YOUHAD AYOUB

Region: eu-west-3 (Paris)

NextWork Challenge: AWS Beginners Challenge - Project 7