



NextWork.org

Launching VPC Resources



Nikhil Bhan

Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances. Mouse over a resource to highlight the related resources.

VPC settings

Resources to create Info
Create only the VPC resources 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 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

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

Preview

VPC Show details
Your AWS virtual network
NextWork-vpc

Subnets (6)
Subnets within this VPC

- ca-central-1a
 - NextWork-subnet-public1-ca
 - NextWork-subnet-private1-ca
 - NextWork-subnet-private2-ca
- ca-central-1b
 - NextWork-subnet-public2-ca
 - NextWork-subnet-private3-ca
 - NextWork-subnet-private4-ca

Route Tables (5)
Route network traffic to resources

- NextWork-rtb-public
- NextWork-rtb-private1-ca-central-1a
- NextWork-rtb-private2-ca-central-1b
- NextWork-rtb-private3-ca-central-1a
- NextWork-rtb-private4-ca-central-1b

Network connections
Connections to other networks

- NextWork-igw
- NextWork-vpc-e-13

Introducing Today's Project!

What is Amazon VPC?

Amazon VPC (Virtual Private Cloud) allows users to create a private and isolated network in their AWS account. They can manage and organize resources as well as configure permissions and access to those resources.

How I used Amazon VPC in this project

In this project I used my NextWork VPC to provision two EC2 instances. I put one instance in the public subnet and the other one in the private subnet. I also used the VPC Resource Map to create a new VPC with components that displayed as a map.

One thing I didn't expect in this project was...

I wasn't expecting to learn about the VPC Resource Map; I found this feature to be useful and I like having that option if I need to customize my VPC instead of doing everything manually in the AWS Console.

This project took me...

This project took me 1 hour and 30 minutes, which also included the 15 minutes I used to write my documentation.

Setting Up Direct VM Access

Directly accessing a virtual machine means being able to log in and connect to it through the Internet, making it seem like you're sitting in front of it. By connecting to the EC2 instance, you are able to manage the operating system and software.

SSH is a key method for directly accessing a VM

SSH traffic means that when a user communicates with an EC2 instance they'll be using a connection where the traffic, commands and responses are all encrypted. This is an ideal method to secure and exchange data, like login information.

To enable direct access, I set up key pairs

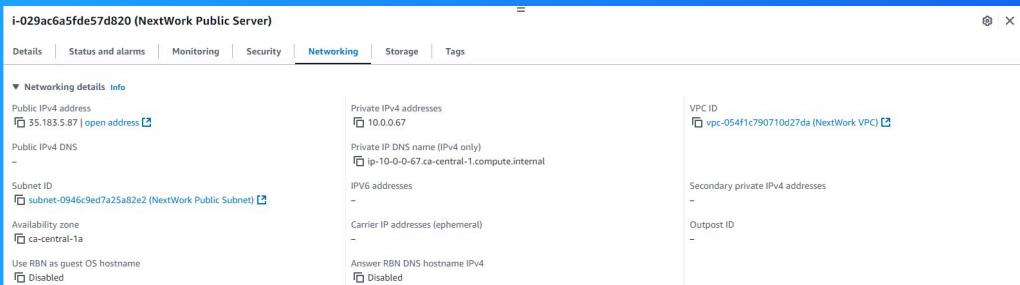
Key pairs are a combination of public keys and private keys. A public key is used to create an encryption on a virtual machine and only the user with the private key can decrypt that, allowing them to access the virtual machine and any data.

A private key's file format means how the private key is saved and what systems and applications will support the format that's being used. My private key's file format was .pem, which is Privacy Enhanced Mail.



Launching a public server

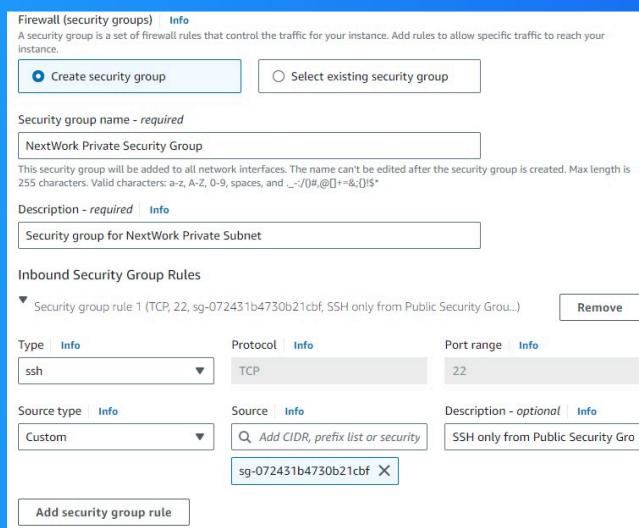
I had to change my EC2 instance's networking settings by clicking Edit in the setup process of my instance. From there I selected the NextWork VPC and chose the public subnet as well as allowing it to use the existing NextWork security group.



Launching a private server

My private server has its own dedicated security group because I've set different criteria to ensure the server's security and management. I'm only allowing SSH connections from resources in the public subnet to access the private server.

My private server's security group's source is the NextWork Public Security Group, which means that only resources and traffic from the public subnet will be allowed to enter the private subnet. This provides a higher level of security.

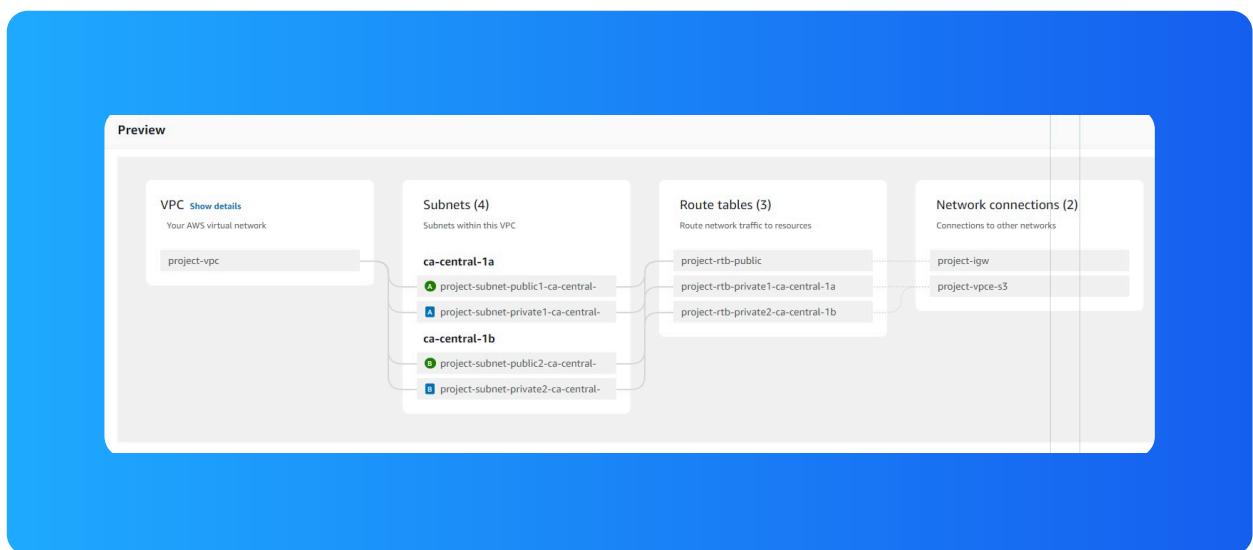


Speeding up VPC creation

I used an alternative way to set up an Amazon VPC! This time, I clicked on "Create VPC" from the VPC Dashboard and selected the "VPC and More" option. By doing this, it allowed me to add subnets, Availability Zones and other components to my VPC.

A VPC resource map is a visual representation of the VPC architecture, which makes it convenient and intuitive for the user. The user can add components and make changes to the architecture and view those changes live on the resource map.

My new VPC has a CIDR block of 10.0.0.0/16. It is possible for my new VPC to have the same IPv4 CIDR block as my existing VPC because these two don't automatically connect; as long as they are separate entities they can have the same CIDR.





Speeding up VPC creation

Tips for using the VPC resource map

When determining the number of public subnets in my VPC, I only had two options: - 0 - 2 This was because AWS provides redundancy and high availability for users. If an AZ goes down, the other one can work and resources are still accessible.

The set up page also offered to create NAT gateways, which allow instances in private subnets to access the Internet for patches and updates; the NAT gateways can also protect these instances from the incoming traffic and external threats.

The screenshot shows the 'Create VPC' setup page on the AWS CloudFormation console. On the left, the 'VPC settings' section includes fields for 'Name tag prefix-generation' (set to 'Auto-generate' with 'NextWork'), 'IPv4 CIDR block' (set to '10.0.0.0/16'), and 'Tenancy' (set to 'Default'). Below these are options for 'Number of Availability Zones (AZs)' (set to '2') and 'Customize AZs'. On the right, the 'Preview' section displays a hierarchical resource map. It starts with a 'VPC' node labeled 'NextWork-vpc'. This node connects to 'Subnets (6)' under 'ca-central-1a' and 'ca-central-1b'. Each subnet node lists specific subnets: 'ca-central-1a' has 'NextWork-subnet-public1-ca-central-1a', 'NextWork-subnet-private1-ca-central-1a', and 'NextWork-subnet-private3-ca-central-1a'; 'ca-central-1b' has 'NextWork-subnet-public2-ca-central-1b', 'NextWork-subnet-private2-ca-central-1b', and 'NextWork-subnet-private4-ca-central-1b'. These subnet nodes further connect to 'Route tables (5)'. A 'Network connections' section shows 'NextWork-rtb-public', 'NextWork-rtb-private1-ca-central-1a', 'NextWork-rtb-private2-ca-central-1b', 'NextWork-rtb-private3-ca-central-1a', 'NextWork-rtb-private4-ca-central-1b', 'NextWork-igw', and 'NextWork-vpc-e-s3'.



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

