



# Creating a Private Subnet



Leon Williams

The screenshot shows the AWS VPC 'Create subnet' wizard. The 'Subnet settings' step is displayed, which allows users to specify the CIDR block and Availability Zone for the subnet.

**Subnet 1 of 1**

**Subnet name:** My Private Subnet

**Availability Zone:** United States (N. Virginia) / us-east-1b

**IPv4 VPC CIDR block:** 10.0.0.0/16

**IPv4 subnet CIDR block:** 10.0.0.0/24 (256 IPs)

**Tags - optional:**

Key	Value - optional
Name	My Private Subnet

**Actions:** Cancel, Create subnet



**Leon Williams**  
NextWork Student

[nextwork.org](http://nextwork.org)

# Introducing Today's Project!

## What is Amazon VPC?

Amazon VPC lets you create a private and isolated network in the cloud. It is useful because it gives you full control over networking, such as IP ranges, subnets, route tables, and gateways, to securely run AWS resources.

## How I used Amazon VPC in this project

I used Amazon VPC to build a secure network environment by creating public and private subnets, setting up route tables to control traffic flow, and configuring network ACLs to manage access. This setup keeps sensitive resources protected.

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

One thing I didn't expect in this project was that the default network ACL allows all traffic, even to private subnets. I thought removing internet access in the route table was enough, but the ACL had to be secured too.



**Leon Williams**  
NextWork Student

[nextwork.org](http://nextwork.org)

---

## This project took me...

It took me just over 33 minutes to complete. The estimate was 60 minutes, but reusing my AWS setup from the last project definitely cut my time in half.



# Private vs Public Subnets

The difference between public and private subnets is that public subnets can access the internet through an internet gateway, while private subnets are isolated and used for sensitive resources like databases.

Having private subnets is useful because they let you separate resources, like databases, from the public internet. This prevents unauthorized access and keeps your network organized using unique IP ranges that avoid overlap with public subnets.

My private and public subnets cannot have the same CIDR block, because each subnet in a VPC must have a unique IP range to avoid conflicts and ensure traffic is routed correctly.

**Leon Williams**  
NextWork Student

[nextwork.org](http://nextwork.org)

The screenshot shows the 'Create subnet' wizard in the AWS VPC console. The current step is 'Subnet settings'. The subnet is named 'My Private Subnet' and is located in the 'United States (N. Virginia) / us-east-1b' availability zone. The IPv4 CIDR block is set to '10.0.0.0/16'. The IPv4 subnet CIDR block is '10.0.1.0/24', which provides 256 IPs. There are no tags added yet. The 'Create subnet' button is highlighted in orange at the bottom right.



## A dedicated route table

By default, my private subnet is associated with the already created Route Table, which is the default route table AWS created with my VPC. This happens because subnets not linked to another route table automatically use the default one.

I had to set up a new route table because the default route table has a route to an internet gateway, which would make my private subnet public. To keep it isolated, I created a separate route table without internet access.

My private subnet's route table only allows local traffic within the VPC. It has no route to the internet gateway, so it keeps the subnet isolated by restricting traffic to internal resources and blocking any direct internet access.

**Leon Williams**  
NextWork Student

[nextwork.org](http://nextwork.org)

The screenshot shows the AWS VPC dashboard with the 'Route tables' section selected. A success message at the top states: "You have successfully updated subnet associations for rtb-083fab435d493d237 / My Private Route Table." The 'Route tables (1/3) info' table lists one route table:

Name	Route table ID	Explicit subnet associ...	Main
rtb-0527292528c3da2b9	rtb-09d449c55e6283bc2	subnet-04fc03637b5d5...	Yes
<b>My Public Route Table</b>	<b>rtb-09d449c55e6283bc2</b>	<b>subnet-04fc03637b5d5...</b>	<b>Yes</b>
My Private Route Table	rtb-083fab435d493d237	subnet-0b86ffab97ddff77...	No

Below the table, the details for the selected route table (rtb-09d449c55e6283bc2) are shown:

Details	
Route table ID	rtb-09d449c55e6283bc2
Main	Yes
Owner ID	117163562320
VPC	vpc-05d570dbe6a25ce63   My VPC
Explicit subnet associations	subnet-04fc03637b5d3279e / My Public Subnet
Edge associations	-



# A new network ACL

By default, my private subnet is associated with the VPC's default network ACL, which allows all traffic. This happens because I haven't explicitly linked the private subnet to a custom-made private Network ACL yet.

I set up a dedicated private network ACL for my private subnet because the default ACL allows all traffic, which is insecure. A custom ACL lets me control inbound and outbound traffic, adding an extra layer of protection for sensitive resources.

My new network ACL has two simple rules: it denies all inbound and all outbound traffic by default. Custom NACLs start this way to ensure nothing gets through until specific rules are added to allow trusted traffic.

**Leon Williams**  
NextWork Student

[nextwork.org](http://nextwork.org)

The screenshot shows the AWS VPC dashboard with the 'Network ACLs' section selected. A green success message at the top right states: "You have successfully updated subnet associations for acl-03d263b45e4f0077a / My Private NACL." Below this, the 'Network ACLs (1/4) info' table lists one item:

Name	Network ACL ID	Associated with	Default	VPC ID
My Public NACL	acl-02bbbe9aafe269c240	subnet-04fc03657b5d5279e / My Public Subnet	No	vpc-05d570d

On the right, there's a 'Create network ACL' button. The main content area shows the details for 'acl-03d263b45e4f0077a / My Private NACL'. It has tabs for 'Details', 'Inbound rules', 'Outbound rules', 'Subnet associations', and 'Tags'. The 'Inbound rules (1)' table contains one rule:

Rule number	Type	Protocol	Port range	Source	Allow/Deny
*	All traffic	All	All	0.0.0.0/0	Deny



[nextwork.org](https://nextwork.org)

# The place to learn & showcase your skills

Check out [nextwork.org](https://nextwork.org) for more projects

