



nextwork.org

VPC Peering



saqibh49@gmail.com

The screenshot shows the AWS VPC Peering Connections console. A prominent green notification bar at the top states: "Your VPC peering connection (pcx-03224254f9d79a066) | VPC 1 <-> VPC 2 has been established." It includes a link to "Modify my route tables now". Below this, the peering connection details are listed:

Details	
Requester owner ID	510482603806
Peering connection ID	pcx-03224254f9d79a066
Status	Active
Expiration time	-
Accepter owner ID	510482603806
Requester VPC	vpc-0c44b3ffaf5bb2aa2 / NextWork-1-vpc
Requester CIDRs	10.1.0.0/16
Requester Region	N. Virginia (us-east-1)
VPC Peering connection ARN	arn:aws:ec2:us-east-1:510482603806:vpc-peering-connection/pcx-03224254f9d79a066
Accepter VPC	vpc-06c019668e41393fa / NextWork-2-vpc
Accepter CIDRs	10.2.0.0/16
Accepter Region	N. Virginia (us-east-1)

Below the details, there are tabs for "DNS", "Route tables", and "Tags". Under the "DNS" tab, there are sections for "Requester VPC" and "Accepter VPC", each with "Edit DNS settings" buttons. At the bottom of the page, there are links for "CloudShell", "Feedback", "Console Mobile App", and copyright information: "© 2016, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences".

saqibh49@gmail.com

NextWork Student

nextwork.org

Introducing Today's Project!

What is Amazon VPC?

Amazon VPC is a sectioned off portion of the AWS cloud and it is useful because it gives a space for separate entities to store resources in a secure place.

How I used Amazon VPC in this project

In today's project, I used Amazon VPC to create all the basic resources found in 2 VPCs and connect the two using a peering connection.

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

One thing I didn't expect in this project was how specific instructions effected the whole process. One line of text changed the entire outcome of the project and that's pretty cool to see in action.

This project took me...

This project took me 45 minutes

saqibh49@gmail.com

NextWork Student

nextwork.org

In the first part of my project...

Step 1 - Set up my VPC

In this step, I will create two VPCs quickly using the set up wizard so I can create a connection between the two and test it out.

Step 2 - Create a Peering Connection

In this step, I will set up a VPC Peering connection because that is how VPCs communicate with each other.

Step 3 - Update Route Tables

In this step, I will set up route tables because without the correct set up, data won't be able to travel between VPC 1 and VPC 2.

Step 4 - Launch EC2 Instances

In this step, I will create an EC2 instance in each of my VPCs because then I'll be able to test the connection between the two.

saqibh49@gmail.com

NextWork Student

nextwork.org

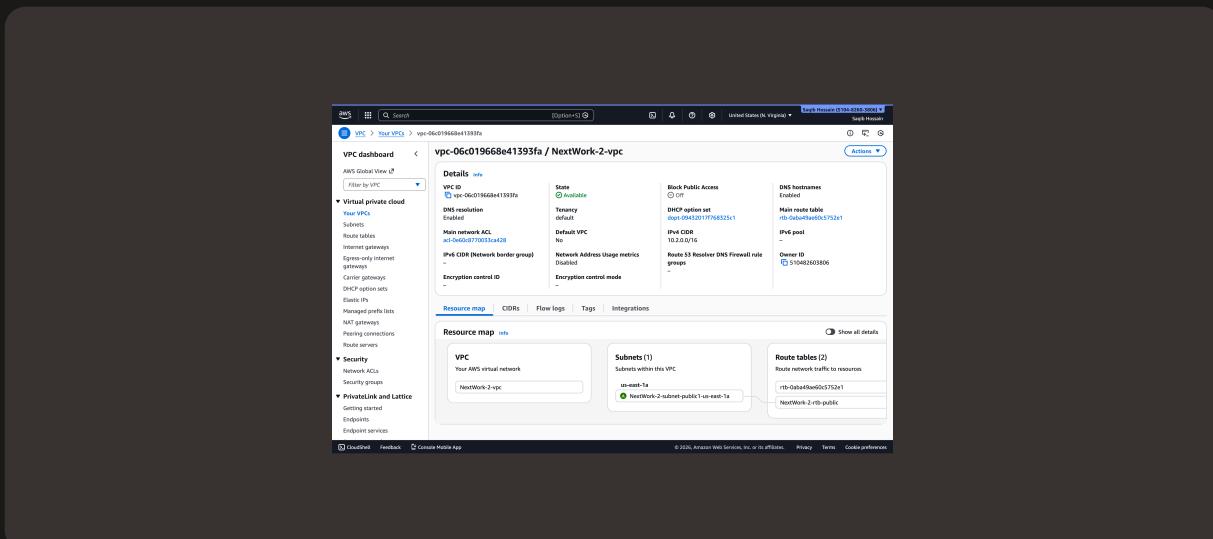
Multi-VPC Architecture

I started my project by launching two VPCs along with their related resources.

The CIDR blocks for VPCs 1 and 2 are 10.1.0.0/16 and 10.2.0.0/16. They have to be unique because if the IP addresses overlap, that could cause problems when trying to access resources within each VPC.

I also launched 2 EC2 instances

I didn't set up key pairs for these EC2 instances as we'll only be accessing these instances using EC2 Instance Connect, and Instance Connect automatically manages key pairs so we don't need to worry about it in this case.



saqibh49@gmail.com

NextWork Student

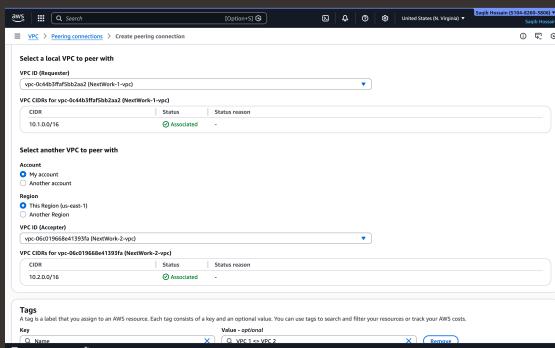
nextwork.org

VPC Peering

A VPC peering connection is a direct connection between two VPCs.

VPCs would use peering connections to connect without using the open internet, which is more secure and cost effective.

The difference between a Requester and an Acceptor in a peering connection is essentially the direction data will flow in a peering connection. The requester asks to connect and the accepter either accepts or denies the request.



saqibh49@gmail.com

NextWork Student

nextwork.org

Updating route tables

After accepting a peering connection, my VPCs' route tables need to be updated because with the new routes in place, data from each VPC will be able to find the other VPC that it needs to go to.

My VPCs' new routes have a destination of the the other VPC's IP address. The routes' target was the peering connection I created earlier.

The screenshot shows the AWS VPC Route Tables page. The route table ID is rtb-0d2c57a40096e7f74, associated with the VPC vpc-06c019668e41393fa. It has three routes:

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	igw-0c26496a52c8221e1	Active	No	Create Route
10.1.0.0/16	pex-03224254fd79a066	Active	No	Create Route
10.2.0.0/16	local	Active	No	Create Route Table

saqibh49@gmail.com

NextWork Student

nextwork.org

In the second part of my project...

Step 5 - Use EC2 Instance Connect

In this step, I will attempt to connect to Instance 1 to test if everything is set up correctly.

Step 6 - Connect to EC2 Instance 1

In this step, I will connect to Instance 1 for real this time because now it has a public IP address.

Step 7 - Test VPC Peering

In this step, I will attempt to send data from instance 1 to instance 2 and back to test out the connection between the two and troubleshoot any errors.

saqibh49@gmail.com

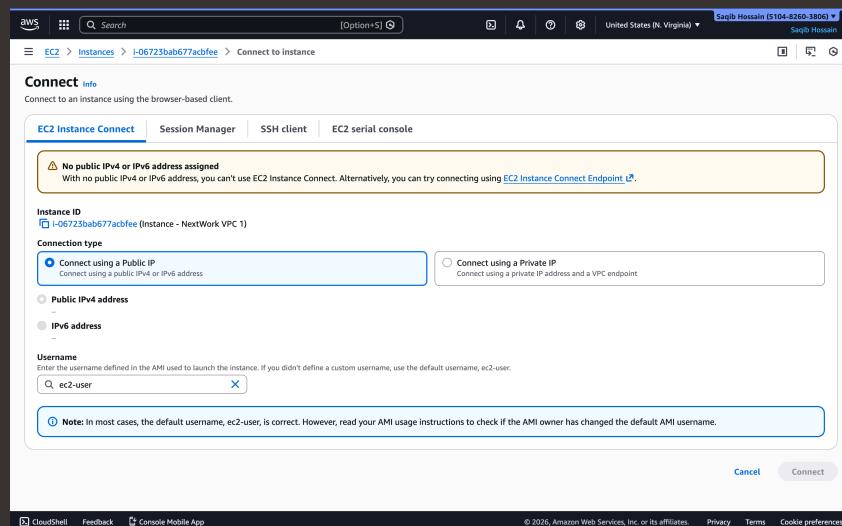
NextWork Student

nextwork.org

Troubleshooting Instance Connect

I'm using EC2 Instance Connect to test the connection between my two Instances.

I was stopped from using EC2 Instance Connect as my instance didn't have a public IP address.



saqibh49@gmail.com

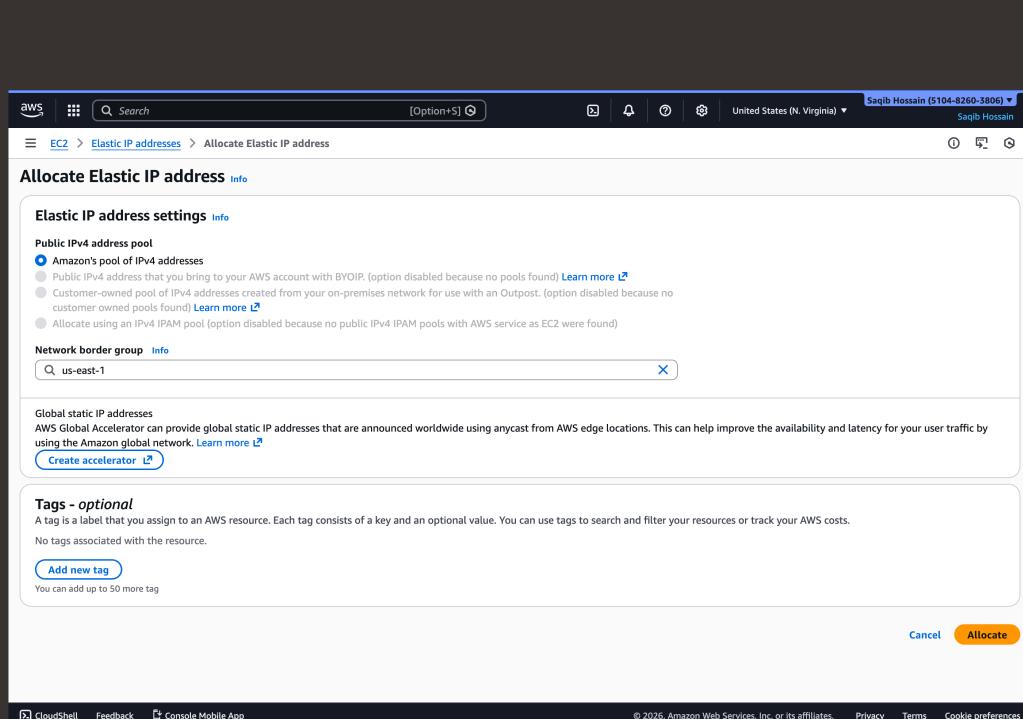
NextWork Student

nextwork.org

Elastic IP addresses

To resolve this error, I set up Elastic IP addresses. Elastic IP addresses are static IP addresses that allow an instance to have a stable place on the internet, without an elastic IP, every time someone wants to access the instance, it receives a new IP, changing how it has to be accessed on the internet.

Associating an Elastic IP address resolved the error because it provides my instance with a public IP that Instance Connect that connect using.



saqibh49@gmail.com

NextWork Student

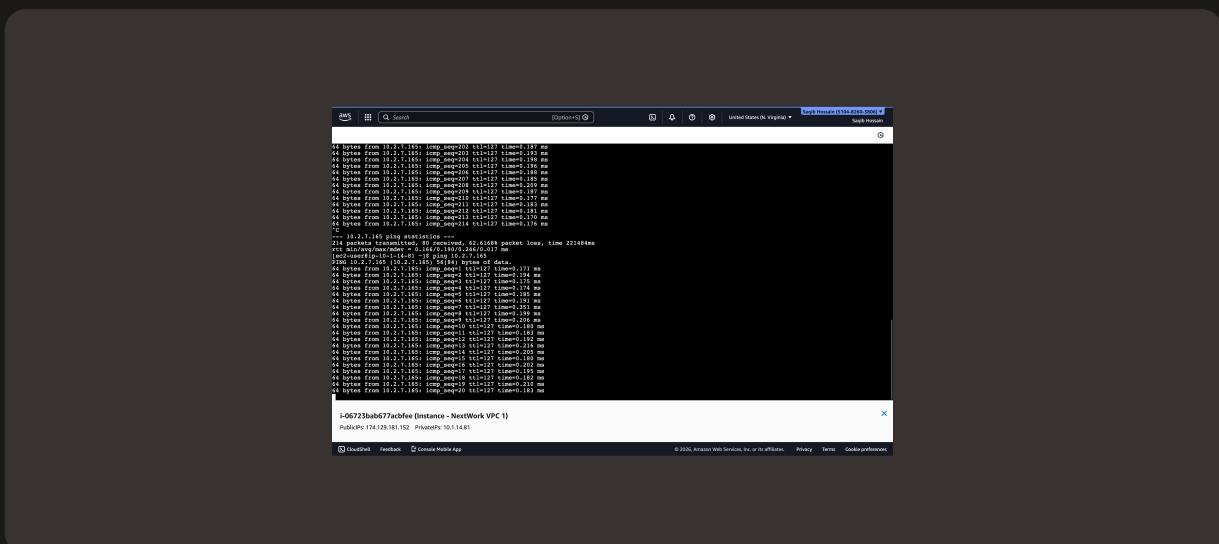
nextwork.org

Troubleshooting ping issues

To test VPC peering, I ran the command ping 10.2.7.165

A successful ping test would validate my VPC peering connection because it means the two instances are able to communicate.

I had to update my second EC2 instance's security group because without the new rule, it wouldn't be able to connect to the other instance. I added a new rule that allows ICMP traffic since that's the traffic it receives from Instance Connect.





nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

