nextwork.org

VPC Monitoring with Flow Logs

GA Mohamed Galole

#	srcAddr	dstAddr	bytesTransferred
1	18.206.107.…	10.1.9.137	19608
2	10.1.9.137	18.206.107.27	10032
3	13.217.78.1…	10.1.9.137	7506
4	13.220.37.7	10.1.13.107	7454
5	44.216.202.…	10.1.9.137	7402
6	44.216.199.…	10.1.13.107	7402
7	10.1.9.137	44.216.202.52	3918
8	10.1.13.107	13.220.37.7	3918
9	10.1.13.107	44.216.199.76	3918
10	10.1.9.137	13.217.78.180	3918

Mohamed Galole
NextWork Student

Introducing Today's Project!

What is Amazon VPC?

Amazon VPC (Virtual Private Cloud) is a service that lets you create a private, isolated network within the AWS cloud, where you can launch and manage AWS resources like EC2 instances.

How I used Amazon VPC in this project

In today’s project, I created two isolated VPCs, launched EC2 instances in each, set up a VPC peering connection, configured route tables and security groups to allow traffic, and enabled VPC Flow Logs to monitor and analyze network activity.

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

One thing I didn't expect in this project was how crucial correctly configuring route tables and security groups is for VPC peering.

This project took me...

This project took me 60 min

Mohamed Galole
NextWork Student

In the first part of my project...

Step 1 - Set up VPCs

I will create two separate Virtual Private Clouds (VPCs) in AWS each with its own CIDR block, subnets, and basic networking components.

Step 2 - Launch EC2 instances

In this step, I will launch one EC2 instance in each VPC because I'll use them later to test connectivity and verify that the VPC peering connection works correctly.

Step 3 - Set up Logs

In this step, I will set up VPC Flow Logs to capture and record all inbound and outbound network traffic within my VPCs because this helps me monitor, analyze, and troubleshoot network activity by storing the flow log data in Amazon CloudWatch Logs.

Step 4 - Set IAM permissions for Logs

In this step, I will create an IAM role and policy for VPC Flow Logs and attach it to my subnet because Flow Logs need permission to write network traffic data to CloudWatch Logs.

GA
Mohamed Galole
NextWork Student

Multi-VPC Architecture

I started my project by launching One Publi Subnet. I chose to create only one public subnet per VPC because the goal of this project is to focus on VPC peering and network monitoring, not high availability across multiple Availability Zones.

The CIDR blocks for VPCs 1 and 2 are VPC-1: 10.1.0.0/16 VPC-2: 10.2.0.0/16 They have to be unique because each VPC represents an isolated virtual network, and overlapping IP address ranges would prevent proper routing between them.

I also launched EC2 instances in each subnet

My EC2 instances' security groups Allow ICMP traffic from ALL IP addresses This is because allowing ICMP traffic from all IP addresses lets me use ping commands to test connectivity between the EC2 instances in both VPCs after setting up the peering.

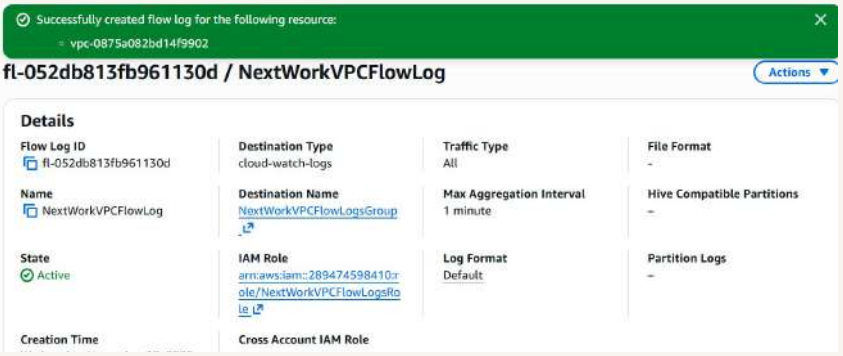
Mohamed Galole
NextWork Student

GA
Mohamed Galole
NextWork Student

Logs

Logs are records of events or activities that happen within a system, application, or network. They capture details about what occurred, when it happened, and sometimes who or what caused it.

Log groups are containers for storing related logs in services like Amazon CloudWatch. Think of a log group as a folder that organizes logs from the same source or purpose. Each log group can contain multiple log streams.



GA
Mohamed Galole
NextWork Student

IAM Policy and Roles

I created an IAM policy because VPC Flow Logs need permission to create log groups and log streams, and to write network traffic data to CloudWatch Logs. Without this policy, Flow Logs would not be able to store or manage the log data.

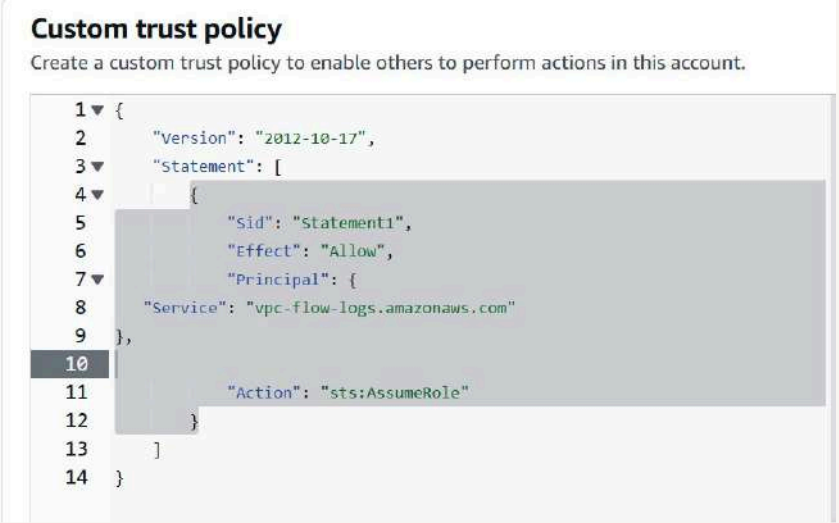
I also created an IAM role because VPC Flow Logs needs a role it can assume to use the permissions defined in the IAM policy. This role allows the flow logs service to write network traffic data to CloudWatch Logs securely.

A custom trust policy is a JSON document attached to an IAM role that defines which entities (users, services, or accounts) are allowed to assume that role.

GA

Mohamed Galole

NextWork Student



Custom trust policy

Create a custom trust policy to enable others to perform actions in this account.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Statement1",
            "Effect": "Allow",
            "Principal": {
   "Service": "vpc-flow-logs.amazonaws.com"
},

            "Action": "sts:AssumeRole"
        }
    ]
}
```

GA
Mohamed Galole
NextWork Student

In the second part of my project...

Step 5 - Ping testing and troubleshooting

In this step, I will send test messages (like ping commands) from Instance 1 to Instance 2 because I want to verify that the VPC peering connection is working and that traffic can successfully flow between the two VPCs.

Step 6 - Set up a peering connection

In this step, I will set up a connection link between my VPCs. because this link will allow private network traffic to flow securely between them without using the public internet.

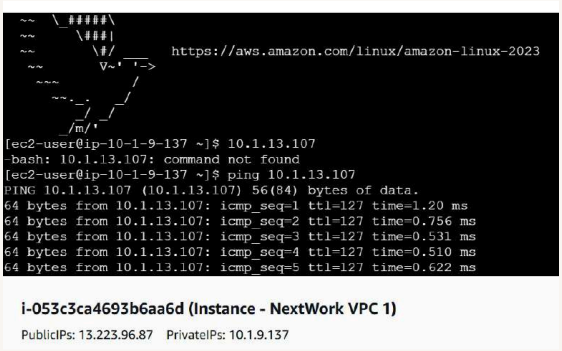
Step 7 - Analyze flow logs

In this step, I will review and analyze the flow logs from VPC 1’s public subnet because this will help me understand the network traffic patterns, verify connectivity between resources, and identify any rejected or unusual traffic.

Mohamed Galole
NextWork Student

Connectivity troubleshooting

My first ping test between my EC2 instances had no replies, which means the instances couldn’t communicate with each other, likely because the route tables or security group rules were not configured correctly to allow traffic between the two VPCs.



I could receive ping replies if I ran the ping test using the other instance's public IP address, which means instances can communicate over the internet, but not through their private IPs yet indicating that the VPC peering connection not configured

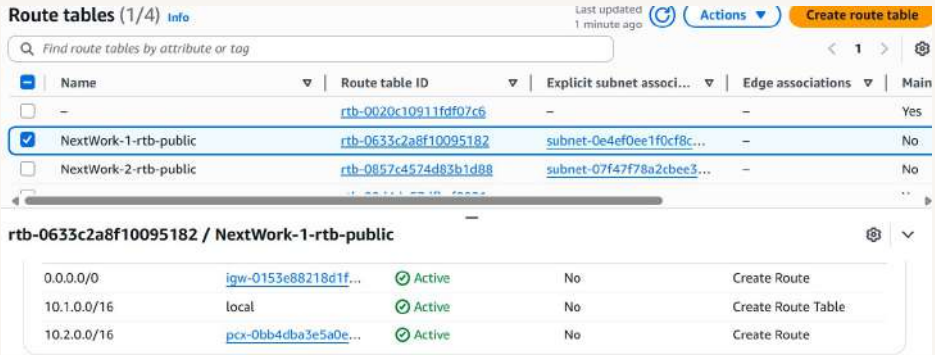
GA
Mohamed Galole
NextWork Student

Connectivity troubleshooting

Looking at VPC 1's route table, I identified that the ping test with Instance 2's private address failed because VPC 1’s route table did not have a route pointing to VPC 2’s CIDR block through the VPC peering connection.

To solve this, I set up a peering connection between my VPCs

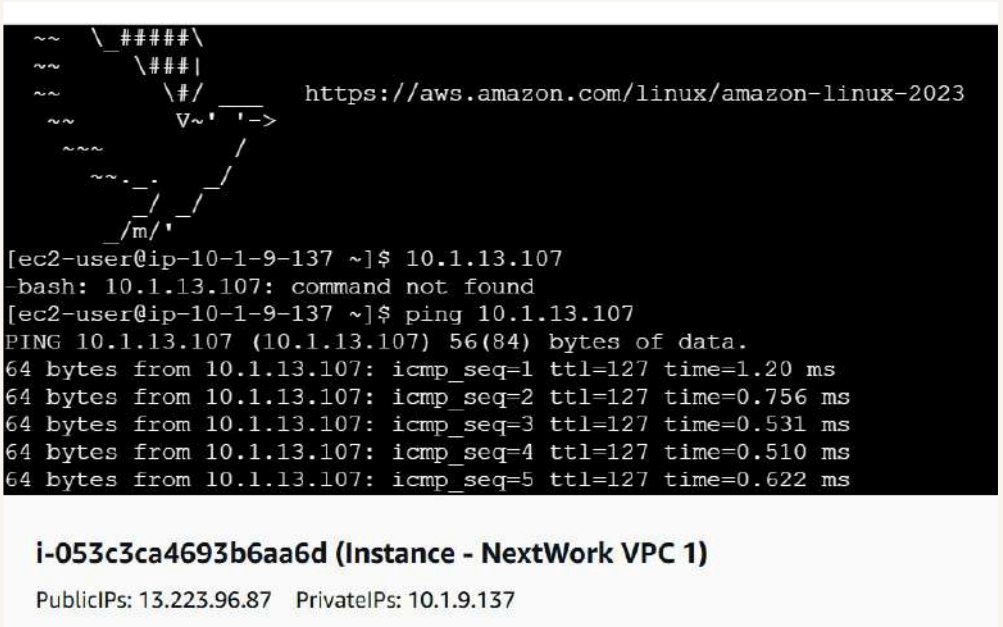
I also updated both VPCs' route tables so that each VPC needs a route that knows how to reach the other VPC’s network through the peering connection.



Mohamed Galole
NextWork Student

Connectivity troubleshooting

I received ping replies from Instance 2's private IP address! This means the VPC peering connection and route tables are correctly configured, and private network traffic can now flow securely between the two VPCs without using the public internet.

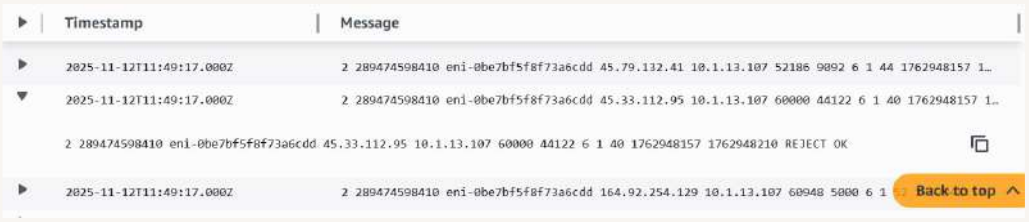


GA
Mohamed Galole
NextWork Student

Analyzing flow logs

Flow logs tell us about the details of network traffic going in and out of your VPC, subnet, or network interface. Each flow log record is made up of several parts that describe what happened in the traffic flow.

For example, the flow log I've captured tells us a TCP connection attempt from 45.33.112.95 to 10.1.13.107 on ports 60000 → 44122 was blocked, and the attempt only sent 1 packet (40 bytes). It shows both who tried to connect, where it was going.



GA
Mohamed Galole
NextWork Student

Logs Insights

Logs Insights is a tool in AWS that lets you search, analyze, and visualize log data from services like VPC Flow Logs, Lambda, or EC2.

I ran the query which analyzes the top 10 source and destination IP address pairs by total byte transfer in my VPC Flow Logs.

#	srcAddr	dstAddr	bytesTransferred
1	18.206.107.…	10.1.9.137	19608
2	10.1.9.137	18.206.107.27	10032
3	13.217.78.1…	10.1.9.137	7506
4	13.220.37.7	10.1.13.107	7454
5	44.216.202.…	10.1.9.137	7402
6	44.216.199.…	10.1.13.107	7402
7	10.1.9.137	44.216.202.52	3918
8	10.1.13.107	13.220.37.7	3918
9	10.1.13.107	44.216.199.76	3918
10	10.1.9.137	13.217.78.180	3918

The place to learn & showcase your skills

Check out nextwork.org for more projects