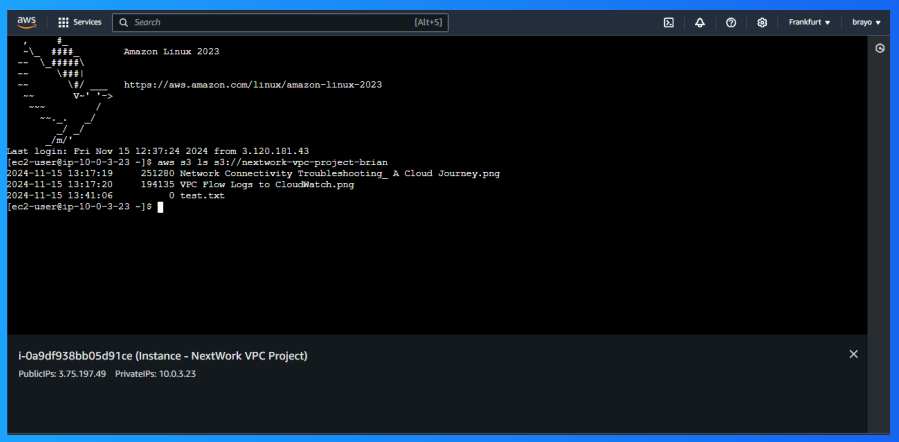
NextWork.org

Access S3 from a VPC

Brian Kimemia N



Brian Kimemia N
NextWork Student

Introducing Today's Project!

What is Amazon VPC?

Amazon VPC Virtual Private Cloud is a private, secure network within AWS. It allows you to control networking, isolate resources, and securely connect to on-premises systems. It’s useful for enhancing security, flexibility, and scalability in cloud.

How I used Amazon VPC in this project

I used Amazon VPC in today's project to create an isolated and secure network environment for my resources, including an EC2 instance and S3 bucket. This ensured controlled access and secure communication between the resources within the same region

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

One thing I didn't expect was how seamlessly the AWS CLI integrated with the EC2 instance for managing resources like S3 buckets. The ease of configuring access and interacting with AWS services directly from the terminal was a pleasant surprise.

This project took me...

This project took me approximately a 1 hour to complete. It involved setting up the EC2 instance, configuring the AWS CLI, creating an S3 bucket, uploading files, and running various commands to ensure proper interaction with the AWS environment.

Brian Kimemia N
NextWork Student

In the first part of my project...

Step 1 - Architecture set up

Create a VPC from scratch!. Launch an EC2 instance into the VPC.

Step 2 - Connect to my EC2 instance

Connecting to the EC2 instance and try access an AWS service!

Step 3 - Set up access keys

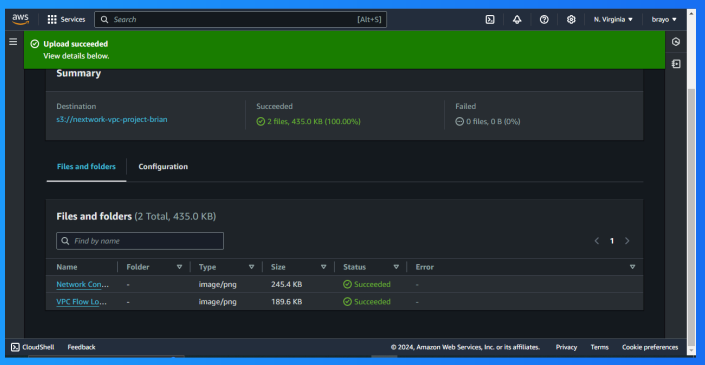
Creating Access Keys, for EC2 instance needs credentials to access your AWS services

Brian Kimemia N
NextWork Student

Architecture set up

I started my project by launching a VPC named NextWork with one public subnet and no private subnets. I also launched an EC2 instance with a public IP and SSH access enabled.

I also set up an S3 bucket named nextwork-vpc-project-brian to store and manage objects. After creating the bucket, I uploaded two files from my local computer to demonstrate file storage and access in S3.



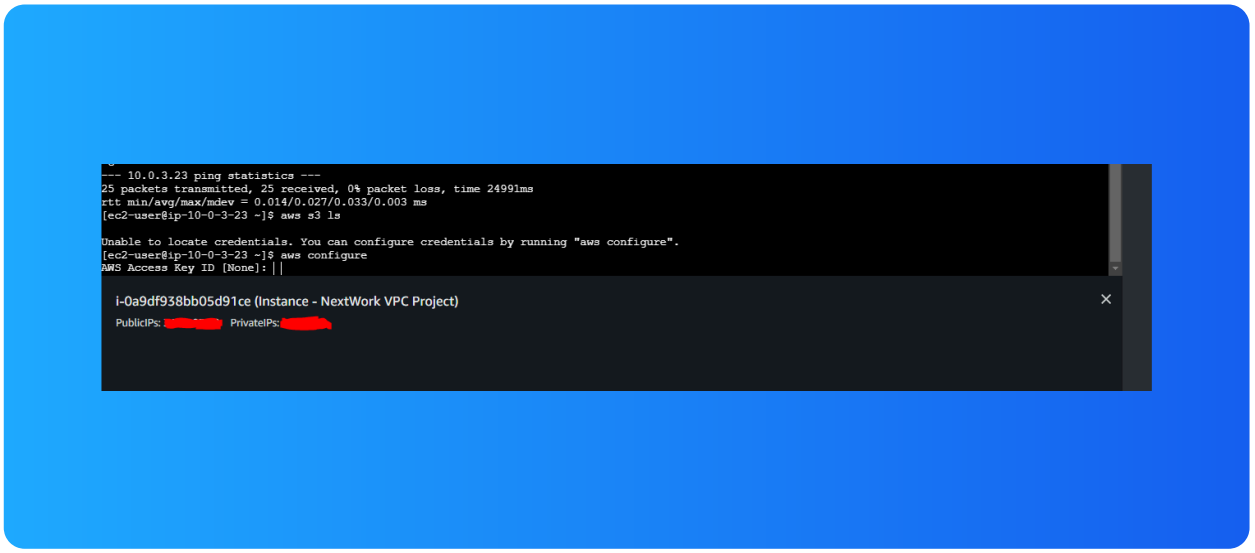
Brian Kimemia N
NextWork Student

Running CLI commands

AWS CLI is a tool for managing AWS services from the command line. I have access to it because I can install it and authenticate using my AWS credentials.

The first command I ran was aws s3 ls. This command is used to list all the S3 buckets in my AWS account.

The second command I ran was aws configure. This command is used to set up AWS credentials, including the access key ID and secret access key, to allow the CLI to interact with AWS services securely.



Brian Kimemia N
NextWork Student

Access keys

Credentials

To set up my EC2 instance to interact with my AWS environment, I configured the AWS CLI using the aws configure command. This allowed me to provide credentials and region settings needed for secure and seamless communication with AWS services.

Access keys are a pair of security credentials, consisting of an access key ID and a secret access key, used to authenticate and securely access AWS services via the AWS CLI, SDKs, or APIs.

Secret access keys are part of AWS credentials, paired with an access key ID, and are used to securely sign requests made to AWS services via the CLI, SDKs, or APIs.

Best practice

Although I'm using access keys in this project, a best practice alternative is to use IAM roles. IAM roles provide temporary credentials to applications running on AWS services, eliminating the need to hard-code access keys and improving security.

Brian Kimemia N
NextWork Student

In the second part of my project...

Step 4 - Set up an S3 bucket

Creating a bucket in Amazon S3 . After creating this bucket, we'll learn how to access it from our EC2 instance and do things like checking what objects are in the bucket.

Step 5 - Connecting to my S3 bucket

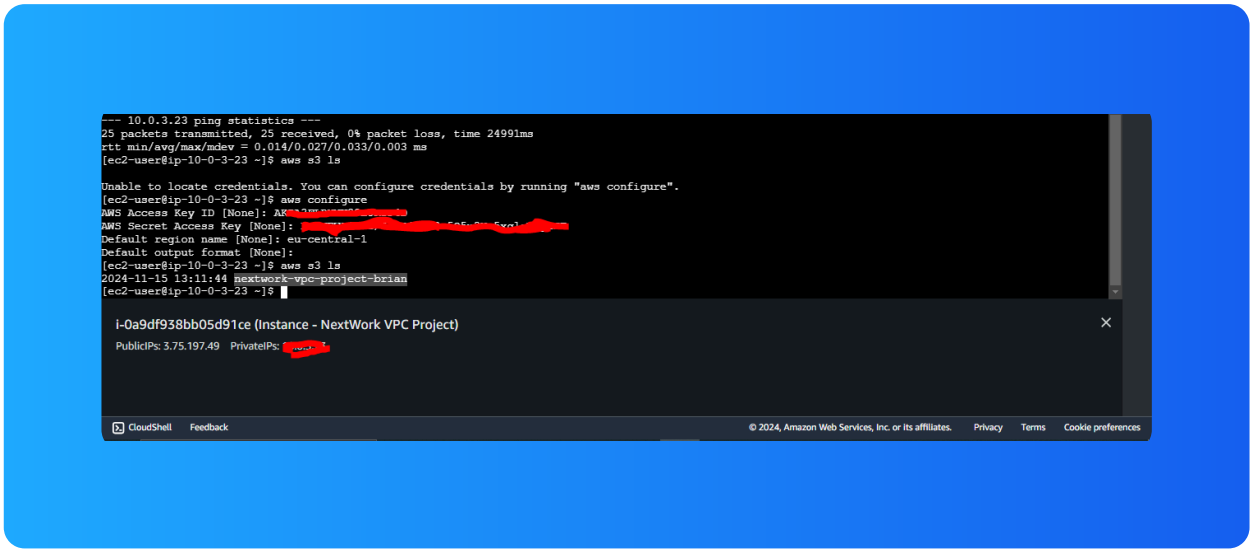
Heading back to the EC2 instance.. Get the EC2 instance to interact with my S3 bucket.

Brian Kimemia N
NextWork Student

Connecting to my S3 bucket

The first command I ran was aws s3 ls. This command is used to list all the S3 buckets in my AWS account.

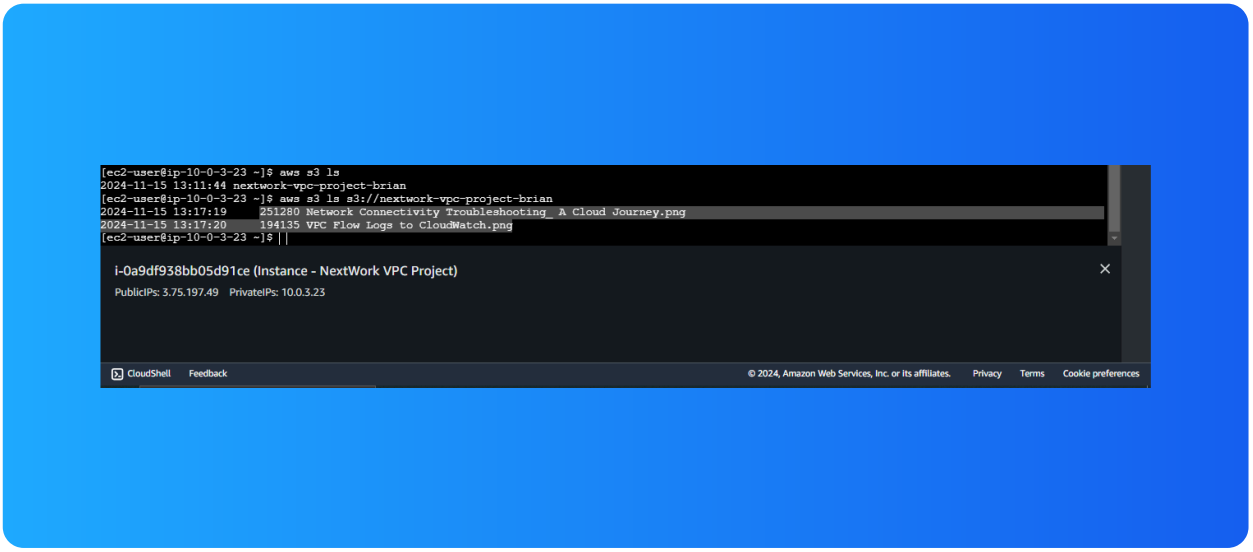
When I ran the command aws s3 ls, the terminal responded with a list of S3 buckets in my account. This indicated that the AWS CLI was successfully configured and connected to my AWS environment.



Brian Kimemia N
NextWork Student

Connecting to my S3 bucket

Another CLI command I ran was aws s3 ls s3://nextwork-vpc-project-brian, which returned a list of objects stored in the specified S3 bucket. This confirmed that the bucket was accessible and the files were successfully uploaded.



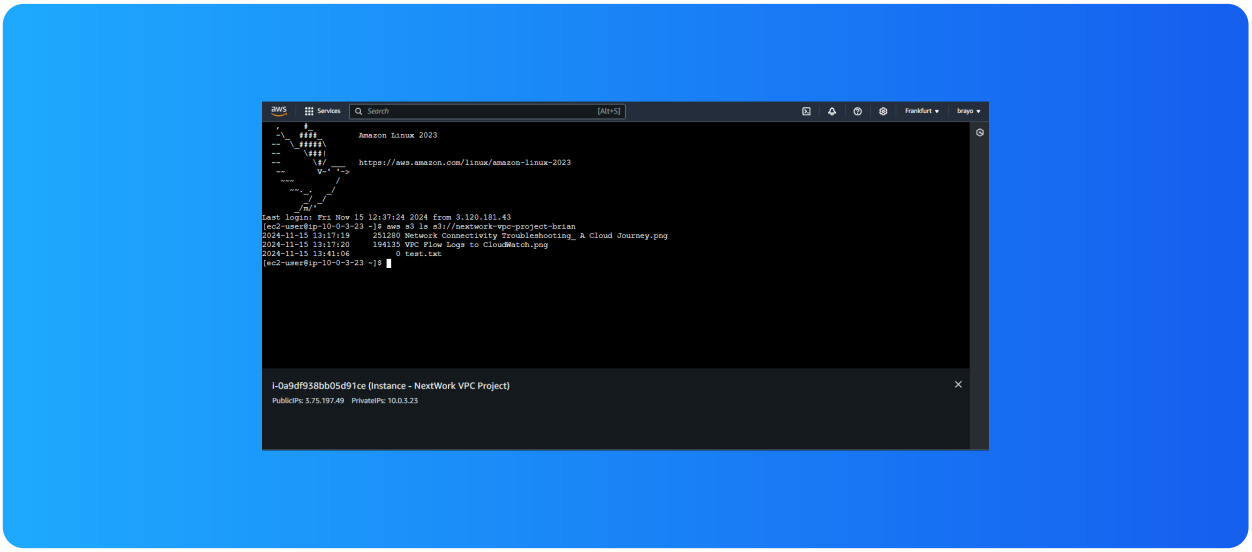
Brian Kimemia N
NextWork Student

Uploading objects to S3

To upload a new file to my bucket, I first ran the command sudo touch /tmp/test.txt. This command creates an empty text file named test.txt in the /tmp directory of my EC2 instance.

The second command I ran was aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-brian. This command will copy the test.txt file from the /tmp directory of my EC2 instance and upload it to the specified S3 bucket (nextwork-vpc-project-brian).

The third command I ran was aws s3 ls s3://nextwork-vpc-project-brian, which validated that the test.txt file was successfully uploaded to the S3 bucket. This command displayed the list of objects in the bucket, confirming the presence of new file.



Everyone should be in a job they love.

Check out nextwork.org for more projects