



Access S3 from a VPC



mmoorej24@gmail.com

```
fec2-user@ip-10-0-11-106 ~]$ sudo touch /tmp/test.txt
fec2-user@ip-10-0-11-106 ~]$ aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-mitchell
upload: ../../tmp/test.txt to s3://nextwork-vpc-project-mitchell/test.txt
fec2-user@ip-10-0-11-106 ~]$ aws s3 ls s3://nextwork-vpc-project-mitchell
2024-08-17 19:55:54      2431554 NextWork - Denzel is awesome.png
2024-08-17 19:55:55      2399812 NextWork - Lelo is awesome.png
2024-08-17 20:09:12           0 test.txt
fec2-user@ip-10-0-11-106 ~]$ █
```



mmoorej24@gmail.com
NextWork Student

NextWork.org

Introducing Today's Project!

What is Amazon VPC?

Amazon VPC is AWS's foundational networking service that allows us to create our own isolated network within an AWS region and control network traffic and security, etc.

How I used Amazon VPC in this project

I launched a VPC with a public subnet and EC2 instance, and directly accessed/managed an AWS S3 bucket through the EC2 instance.

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

How challenging this project would be and that some steps I had to research to progress through the project.

This project took me...

This project took me about an hour and thirty minutes.



In the first part of my project...

Step 1 - Architecture set up

In this step, I launch a VPC with a public subnet. I also launch an EC2 instance inside that public subnet.

Step 2 - Connect to my EC2 instance

In this step, I directly access an EC2 instance using EC2 instance using EC2 Instance Connect.

Step 3 - Set up access keys

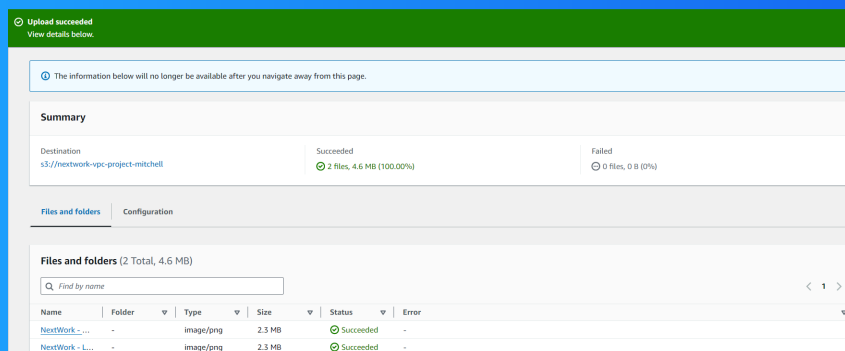
In this step, I create access keys so that our EC2 instance can have access to our AWS environment, specifically, the ability to interact with an S3 bucket.



Architecture set up

I started my project by launching a VPC with a public subnet, an EC2 instance inside the public subnet.

I also set up an S3 bucket with two files inside.





Running CLI commands

AWS CLI is a software we can download into a local computer terminal so that we can have access to our AWS account and different actions without needing AWS Management console. I have access to AWS CLI because it's pre-installed in EC2 instances.

The first command I ran was `aws s3 ls`. This command is used to list all S3 buckets inside the AWS account (that the EC2 instance/application) has access to.

The second command I ran was `AWS configure`. This command is used to set up my EC2 instance's credentials in order to access my AWS environment.

```
#####
#_#         Amazon Linux 2023
#####\
#####!
#/#/       https://aws.amazon.com/linux/amazon-linux-2023
V~'|~>
~
~/m/'
[ec2-user@ip-10-0-11-106 ~]$ aws s3 ls
Unable to locate credentials. You can configure credentials by running "aws configure".
[ec2-user@ip-10-0-11-106 ~]$ aws configure
AWS Access Key ID [None]:
```



Access keys

Credentials

To set up my EC2 instance to interact with my AWS environment, I configured an Access Key ID, Secret Access Key, default region, and a default output format.

Access keys are credentials that my EC2 instance + other applications/servers needs in order to get access to my AWS environment i.e. interact with my AWS resources/services.

Secret access keys are like passwords for our access keys / credentials. Our EC2 instance / other applications would need secret access keys as authentication and log in to our AWS environment.

Best practice

Although I'm using access keys in this project, a best practice alternative is to use IAM roles with permissions attached. This is a more secure way to grant access to an EC2 instance because it is much easier to track, attach, and detach IAM access.



In the second part of my project...

Step 4 - Set up an S3 bucket

In this step, I launch an AWS S3 bucket with two files inside. This S3 bucket will be accessed by my EC2 instance later in the project, so I can test whether my access keys have successfully have given my EC2 instance the access to my EC2 instance.

Step 5 - Connecting to my S3 bucket

In this step, I launch an AWS S3 bucket with two files inside. This S3 bucket will be accessed by my EC2 instance later in the project, so I can test whether my access keys have successfully have given my EC2 instance the access to my EC2 instance.



Connecting to my S3 bucket

The first command I ran was `aws s3 ls`. This command is used to list all S3 buckets inside the AWS account (that the EC2 instance/application) has access to.

When I ran the command "`aws s3 ls`" again, the terminal responded with a list of my S3 buckets. This indicated that my access keys work!

```
ec2-user@ip-10-0-11-106 ~]$ aws s3 ls
2024-08-17 19:54:05 nextwork-vpc-project-mitchell
ec2-user@ip-10-0-11-106 ~]$ █
```



Connecting to my S3 bucket

Another CLI command I ran was "aws s3 ls s3://nextwork-vpc-project-mitchell" which returned a list of objects inside the s3 bucket.

```
[ec2-user@ip-10-0-11-106 ~]$ aws s3 ls s3://nextwork-vpc-project-mitchell
2024-08-17 19:55:54      2431554 NextWork - Denzel is awesome.png
2024-08-17 19:55:55      2399812 NextWork - Lelo is awesome.png
[ec2-user@ip-10-0-11-106 ~]$
```



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 a blank file called test.txt in my EC2 instance's local directory.

The second command I ran was "aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-mitchell". This command will copy i.e. upload the blank file created into my s3 bucket.

The third command I ran was "aws s3 ls s3://nextwork-vpc-project-mitchell" which returned a list of all objects in my s3 bucket - including test.txt.

```
[ec2-user@ip-10-0-11-106 ~]$ sudo touch /tmp/test.txt
[ec2-user@ip-10-0-11-106 ~]$ aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-mitchell
upload: ../../tmp/test.txt to s3://nextwork-vpc-project-mitchell/test.txt
[ec2-user@ip-10-0-11-106 ~]$ aws s3 ls s3://nextwork-vpc-project-mitchell
2024-08-17 19:55:54      2431554 NextWork - Denzel is awesome.png
2024-08-17 19:55:55      2399812 NextWork - Lelo is awesome.png
2024-08-17 20:09:12           0 test.txt
[ec2-user@ip-10-0-11-106 ~]$ █
```



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

