



Access S3 from a VPC



Roshan Thomas

```
usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:

  aws help
  aws <command> help
  aws <command> <subcommand> help

aws: error: argument subcommand: Invalid choice, valid choices are:

ls                               | website
cp                               | mv
rm                               | sync
mb                               | rb
presign

Invalid choice: 'ls', maybe you meant:

  * ls

[ec2-user@ip-10-0-15-168 ~]$ aws s3 ls
2024-07-30 05:47:03 cf-templates-1ntz9rwua5roj-ap-southeast-1
2024-10-21 05:27:24 nextwork-vpc-project-roshan
[ec2-user@ip-10-0-15-168 ~]$ aws s3 ls s3://nextwork-vpc-project-roshan
2024-10-21 05:30:24    1802074 IMG_20220430_195140.jpg
2024-10-21 05:30:23  14578723 IMG_20220430_195209.jpg
[ec2-user@ip-10-0-15-168 ~]$ sudo touch /tmp/test.txt
[ec2-user@ip-10-0-15-168 ~]$ aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-roshan
upload: ../../tmp/test.txt to s3://nextwork-vpc-project-roshan/test.txt
[ec2-user@ip-10-0-15-168 ~]$ aws s3 ls s3://nextwork-vpc-project-roshan
2024-10-21 05:30:24    1802074 IMG_20220430_195140.jpg
2024-10-21 05:30:23  14578723 IMG_20220430_195209.jpg
2024-10-21 05:47:30      0 test.txt
[ec2-user@ip-10-0-15-168 ~]$ ||
```



Roshan Thomas
NextWork Student

NextWork.org

Introducing Today's Project!

What is Amazon VPC?

Amazon VPC (Virtual Private Cloud) allows you to create an isolated network within AWS, providing control over network settings, security, and access. It's useful for securely hosting applications, managing traffic, and integrating on-premise network.

How I used Amazon VPC in this project

I used VPC to create a secure network for my EC2 instance. I set up a public subnet and security groups to control access. This allowed me to access Amazon S3 from my instance.

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

One thing I didn't expect in this project was the complexity involved in setting up access keys and managing credentials for my EC2 instance to interact with Amazon S3.

This project took me...

2 Hours and 53 Minutes.



In the first part of my project...

Step 1 - Architecture set up

I'm going to create a VPC and launch an EC2 instance within it. This will provide a secure environment for my resources and allow me to interact with other AWS services like Amazon S3.

Step 2 - Connect to my EC2 instance

I'm going to connect to my EC2 instance using EC2 Instance Connect to gain direct access and interact with the instance's terminal. This will allow me to perform tasks and access other AWS services.

Step 3 - Set up access keys

I'm going to create access keys for my EC2 instance so it can securely access and manage AWS services, including Amazon S3. This will allow me to interact with these services from within the instance and perform various tasks.

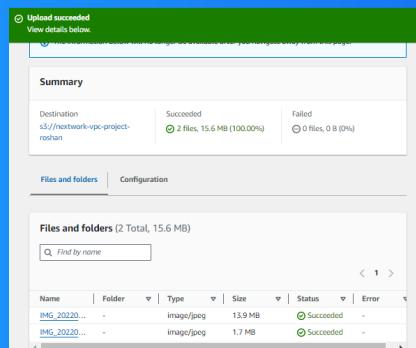
Roshan Thomas
NextWork Student

NextWork.org

Architecture set up

I started my project by launching a VPC with a single public subnet and an EC2 instance in that subnet. I also created a security group for the EC2 instance.

I also set up an S3 bucket named nextwork-vpc-project-roshan and uploaded two files to it. This will allow me to access and manage these files from my EC2 instance.





Running CLI commands

AWS CLI is a command-line interface for managing AWS services. I have access to AWS CLI because it is pre-installed on my EC2 instance. This enables me to perform various tasks and interact with AWS services using commands.

The first command I ran was aws configure. This command is used to configure my AWS credentials, providing my access key ID and secret access key to allow the EC2 instance to access and manage AWS services.

The second command I ran was aws s3 ls. This command is used to list the S3 buckets that I have access to within my AWS account.

```
Amazon Linux 2023
https://www.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-10-0-15-168 ~]$ aws s3 ls
Unable to locate credentials. You can configure credentials by running "aws configure".
[ec2-user@ip-10-0-15-168 ~]$ aws configure
AWS Access Key ID [None]: 
```

i-0b1c8db68c80c0940 (Instance - NextWork VPC Project)
PublicIP: 13.201.88.160 PrivateIP: 10.0.15.168



Access keys

Credentials

To set up my EC2 instance to interact with my AWS environment, I configured its credentials using the aws configure command. This provided the instance with the necessary access keys and secret access key to securely access and manage AWS services.

Access keys are credentials that provide authentication for your AWS account. They consist of an access key ID and a secret access key, which are used to securely access and manage AWS services.

Secret access keys are a crucial part of AWS credentials, providing a secure method of authentication. They are used in conjunction with access key IDs to grant access to AWS services and resources.

Best practice

Although I'm using access keys, a best practice alternative is to use IAM roles for enhanced security and reduced risk.

A circular profile picture of a man with dark hair and a beard, wearing a light-colored striped shirt.

Roshan Thomas
NextWork Student

NextWork.org

In the second part of my project...

Step 4 - Set up an S3 bucket

I'm going to create an S3 bucket and upload two files to it. This will allow me to demonstrate how to access and manage S3 resources from my EC2 instance.

Step 5 - Connecting to my S3 bucket

I'm going to use my EC2 instance to interact with the S3 bucket I created. This will demonstrate how I can access and manage S3 resources from within my VPC.



Roshan Thomas
NextWork Student

NextWork.org

Connecting to my S3 bucket

The first command I ran was aws configure. This command is used to configure my AWS credentials, providing my access key ID and secret access key to allow the EC2 instance to access and manage AWS services.

When I ran aws s3 ls, the terminal responded with two bucket names. One was the bucket I created, nextwork-vpc-project-roshan. The other was a default bucket.

```
Last login: Mon Oct 21 05:06:00 2024 from 13.233.177.3
[ec2-user@ip-10-0-16-168 ~]$ aws configure
AWS Access Key ID [None]: AKIAKGUDUIMKNTARM24K
AWS Secret Access Key [None]: o1pRkWfQgP10F0gIftB09R41bVWF/mN0qrVvGFH
Default region name [None]: ap-southeast-1
Default output format [None]:
[ec2-user@ip-10-0-16-168 ~]$ aws s3 ls
usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:
  aws help
  aws <command> help
  aws <command> <subcommand> help
aws: error: argument subcommand: Invalid choice, valid choices are:
  ls          | website
  cp          | mv
  rm          | sync
  mb          | rs
  presign
Invalid choice: "ls~", maybe you meant:
  * ls
[ec2-user@ip-10-0-16-168 ~]$ aws s3 ls
2024-07-30 05:47:03 cf-templates-lntz9rvua5roj-ap-southeast-1
2024-10-21 05:27:24 Network-vpc-project-roshan
[ec2-user@ip-10-0-16-168 ~]$
```



Roshan Thomas
NextWork Student

NextWork.org

Connecting to my S3 bucket

Another CLI command I ran was aws s3 ls s3://nextwork-vpc-project-roshan, which returned a list of objects within my S3 bucket. This command allowed me to view the contents of the bucket and verify that the files I uploaded were successfully stored.

```
[ec2-user@ip-10-0-15-168 ~]$ aws configure
AWS Access Key ID [None]: AMIAK6QDUTDKWQ7A9M24X
AWS Secret Access Key [None]: dpgkLwfdp1OPgffB09R4lbWff/mnN0qtvWGFH
Default region name [None]: ap-south-1
Default output format [None]:
[ec2-user@ip-10-0-168 ~]$ aws s3 ls

usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:
  aws help
  aws <command> help
  aws <command> <subcommand> help
aws: error: argument subcommand: Invalid choice, valid choices are:
ls          | website
sp          | sv
st          | sync
mb          | rb
presign

Invalid choice: 'ls-', maybe you meant:
  * ls

[ec2-user@ip-10-0-15-168 ~]$ aws s3 ls
2024-10-21 05:47:03 cf-templates-lntz5us5ro.ap-southeast-1
2024-10-21 05:47:03 default
[ec2-user@ip-10-0-15-168 ~]$ aws s3 ls s3://nextwork-vpc-project-roshan
2024-10-21 05:30:24 1802074 IMG_20220430_195140.jpg
2024-10-21 05:30:23 14578723 IMG_20220430_195209.jpg
[ec2-user@ip-10-0-15-168 ~]$
```



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 on my EC2 instance.

The second command I ran was aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-roshan. This command will upload the test.txt file to my S3 bucket named nextwork-vpc-project-roshan.

The third command I ran was aws s3 ls s3://nextwork-vpc-project-roshan, which validated that the test.txt file was successfully uploaded to my S3 bucket.

```
usage: aws [options] <command> [<subcommand> [<subcommand> ...]] [<parameters>]
To see help text, you can run:
  aws help
  aws <command> help
  aws <command> <subcommand> help
aws: error: argument subcommand: Invalid choice, valid choices are:
ls          | website
cp          | mv
rm          | sync
mb          | rb
presign

Invalid choice: 'ls-', maybe you meant:
  * ls

[ec2-user@ip-10-0-15-168 ~]$ aws s3 ls
2024-07-30 05:47:03 cf-templates-lnt2s3vut0cgv-sp-southeast-1
[ec2-user@ip-10-0-15-168 ~]$ aws s3 ls s3://nextwork-vpc-project-roshan
[ec2-user@ip-10-0-15-168 ~]$ aws s3 ls s3://nextwork-vpc-project-roshan
2024-10-21 05:30:24 1802074 IMG_20220430_195140.jpg
2024-10-21 05:30:23 14578723 IMG_20220430_195209.jpg
2024-10-21 05:30:23 0 test.txt
[ec2-user@ip-10-0-15-168 ~]$ aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-roshan
upload: ././tmp/test.txt to s3://nextwork-vpc-project-roshan/test.txt
[ec2-user@ip-10-0-15-168 ~]$ aws s3 ls s3://nextwork-vpc-project-roshan
2024-10-21 05:30:24 1802074 IMG_20220430_195140.jpg
2024-10-21 05:30:23 14578723 IMG_20220430_195209.jpg
2024-10-21 05:47:30 0 test.txt
[ec2-user@ip-10-0-15-168 ~]$
```



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

