



Access S3 from a VPC



alexten3517@gmail.com

```
# AWS CloudShell - N. Virginia - newproject10 @ 3904-0253

aws services search [Option+S]
VPC IAM EC2 CloudWatch Amazon Lex S3

# AWS CloudShell - N. Virginia - newproject10 @ 3904-0253

Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Sat Nov  9 21:40:52 2024 from 18.206.107.28
[ec2-user@ip-10-0-14-112 ~]$ aws s3 ls s3://nextwork-vpc-project-alex
2024-11-09 20:40:33      256472 Screenshot 2024-11-08 at 5.23.03 PM.png
2024-11-09 20:40:33      198972 access key to instance.png
[ec2-user@ip-10-0-14-112 ~]$ aws s3 ls s3://nextwork-vpc-project-alex

2024-11-09 20:40:33      256472 Screenshot 2024-11-08 at 5.23.03 PM.png
2024-11-09 20:40:33      198972 access key to instance.png
[ec2-user@ip-10-0-14-112 ~]$ sudo touch /tmp/test.txt
[ec2-user@ip-10-0-14-112 ~]$ aws s3 ls s3://nextwork-vpc-project-alex
2024-11-09 20:40:33      256472 Screenshot 2024-11-08 at 5.23.03 PM.png
2024-11-09 20:40:33      198972 access key to instance.png
[ec2-user@ip-10-0-14-112 ~]$ aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-alex
upload failed: ./tmp/test.txt to s3://nextwork-vpc-project-alex/test.txt An error occurred (NoSuchBucket)
when calling the PutObject operation: The specified bucket does not exist
[ec2-user@ip-10-0-14-112 ~]$ aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-alex
upload failed: ./tmp/test.txt to s3://nextwork-vpc-project-alex/test.txt An error occurred (NoSuchBucket)
when calling the PutObject operation: The specified bucket does not exist
[ec2-user@ip-10-0-14-112 ~]$ aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-alex
upload: ./tmp/test.txt to s3://nextwork-vpc-project-alex/test.txt
[ec2-user@ip-10-0-14-112 ~]$
```

CloudShell Feedback Privacy Terms Cookie pre © 2024, Amazon Web Services, Inc. or its affiliates.

Introducing Today's Project!

What is Amazon VPC?

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

How I used Amazon VPC in this project

I used it to learn how we are able to control our S3 with our CLI in a very proficient and fast manner.

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

In todays project I launched a VPC with a public subnet and an EC2 instance directly accessed/managed Amazon S3 bucket through the EC2 instance using AWS CLI.

This project took me...

This project took me about 50 min.

In the first part of my project...

Step 1 - Architecture set up

In this step I will be creating an EC2 instance and a VPC which will be the first initial steps.

Step 2 - Connect to my EC2 instance

In this step I directly accessed an ec2 instance using ec2 instance connect.

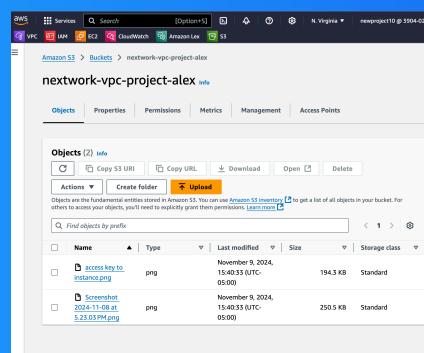
Step 3 - Set up access keys

In this step I create access keys so my EC2 instance can access our AWS enviroment. Specifically the ability to interact with our S3 bucket.

Architecture set up

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

I set up an S3 bucket with two files inside.

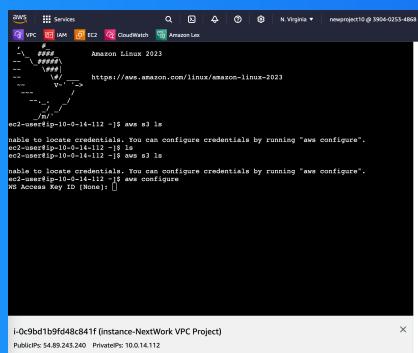


Running CLI commands

AWS CLi is a software we can download into a local computers terminal so we can have access to our aws account without needing to use AWS management console.I have access to AWS CLi because it comes pre installed in our EC2.

The first command we ran was AWS s3 ls. Its used to list all s3 buckets in the AWS account.It is the s3 bucket the instance has access to.

The second command I ran was... This command is used to AWS configure. This command is used to set up my EC2 instances credentials in order to access my AWS enviorment.



A screenshot of a terminal window titled "Amazon Linux 2023" running on an EC2 instance. The terminal shows the following AWS CLI commands being run:

```
ec2-user@ip-10-0-14-112:~$ aws s3 ls
unable to locate credentials. You can configure credentials by running "aws configure".
ec2-user@ip-10-0-14-112:~$ aws s3 ls
unable to locate credentials. You can configure credentials by running "aws configure".
ec2-user@ip-10-0-14-112:~$ aws configure
[No input] : [No input]
```

The terminal window also displays the instance ID, public IP, and private IP at the bottom.

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 default output format.

Access keys are Id's are credentials that my EC2 instance plus applications need in order to access my AWS enviroment.

Secret access keys are a specialized / passwords which allow an EC2 instance the access to communicate with our AWS enviorment.

Best practice

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

In the second part of my project...

Step 4 - Set up an S3 bucket

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

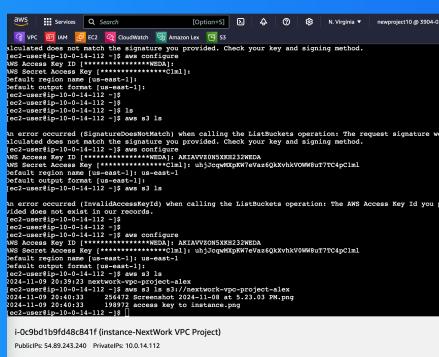
Step 5 - Connecting to my S3 bucket

In this step we are using AWS command to try and control our S3 bucket. This means we are interacting our S3 bucket through our EC2 instance /VPC instead of our AWS management console

Connecting to my S3 bucket

The first command we ran was AWS s3 ls. Its used to list all s3 buckets in the AWS account. It is the s3 bucket the instance has access to.

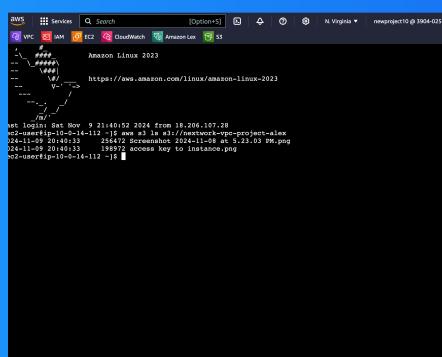
When I ran the command again with aws s3 ls the terminal responded with an output of my S3 bucket. Then after running aws s3 ls s3://nextwork-vpc-project-alex which is the buckets name with ls in I was able to see the files that were in the bucket.



```
aws s3 ls
An error occurred (SignatureDoesNotMatch) when calling the ListBuckets operation: The request signature we
calculated does not match the signature you provided. Check your key and signing method.
aws s3 ls
An error occurred (SignatureDoesNotMatch) when calling the ListBuckets operation: The request signature we
calculated does not match the signature you provided. Check your key and signing method.
aws s3 ls s3://nextwork-vpc-project-alex
2024-11-09 20:49:23 254672 Screenshot 2024-11-09 at 5.23.33 PM.png
```

Connecting to my S3 bucket

aws s3 ls s3://nextwork-vpc-project-alex returned a list of all objects inside that bucket.



The screenshot shows a terminal window titled "Amazon Linux 2023" running on an AWS Lambda instance. The terminal displays the output of the "aws s3 ls" command:

```
aws s3 ls s3://nextwork-vpc-project-alex
2024-11-09 20:40:52 2024 from 18.206.107.28
2024-11-09 20:40:53 256472 Screenlet.msi 2024-11-09 at 5:23:03 9M.png
2024-11-09 20:40:53 199972 access key to instance.png
2024-11-09 20:40:53 117 10
```

Uploading objects to S3

To upload a new file to my bucket I first ran the command "sudo touch /tmp/test.txt" this file creates a blank file called test.txt. in my EC2 instances local directory

The second command I ran "aws s3 cp /tmp/test.txt s3:// nextwork-vpc-project-alex". This command copied up uploaded the blank file created to my S3 bucket.

The third command I ran was "aws s3 cp ls /tmp/test.txt s3://nextwork-vpc-project-alex" which returned a list of all files in my bucket including text..txt.This validated the CLI commands working properly with my EC2 instance.

A screenshot of an AWS CloudShell terminal window. The terminal shows a series of Linux shell commands being run. The user first creates a blank file named 'test.txt' in the '/tmp' directory using the 'touch' command. Then, they use the 'aws s3 cp' command to upload this file to an S3 bucket named 'nextwork-vpc-project-alex'. Finally, they run another 'aws s3 cp' command with the 'ls' option to list the contents of the S3 bucket, which includes the newly uploaded 'test.txt' file. The terminal also displays several screenshots of the Amazon Linux 2023 desktop environment.

```
last login: Sat Mar 9 21:45:52 2024 from 10.246.197.28
[ec2-user@ip-10-0-1-12 ~]$ sudo touch test.txt
[ec2-user@ip-10-0-1-12 ~]$ aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-alex
2024-11-09 20:46:33      256472 Screenshot 2024-11-09 at 5:23.03 PM.png
[ec2-user@ip-10-0-1-12 ~]$ aws s3 ls s3://nextwork-vpc-project-alex
[ec2-user@ip-10-0-1-12 ~]$ aws s3 cp ls s3://nextwork-vpc-project-alex
2024-11-09 20:46:33      256472 Screenshot 2024-11-09 at 5:23.03 PM.png
[ec2-user@ip-10-0-1-12 ~]$ aws s3 cp test.txt s3://nextwork-vpc-project-alex
[ec2-user@ip-10-0-1-12 ~]$ aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-alex
2024-11-09 20:46:33      198972 instance key to instance.png
[ec2-user@ip-10-0-1-12 ~]$ aws s3 cp ls s3://nextwork-vpc-project-alex
2024-11-09 20:46:33      256472 Screenshot 2024-11-09 at 5:23.03 PM.png
[ec2-user@ip-10-0-1-12 ~]$ aws s3 cp ls s3://nextwork-vpc-project-alex
2024-11-09 20:46:33      256472 Screenshot 2024-11-09 at 5:23.03 PM.png
[ec2-user@ip-10-0-1-12 ~]$ aws s3 cp test.txt s3://nextwork-vpc-project-alex
[ec2-user@ip-10-0-1-12 ~]$ aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-alex
[ec2-user@ip-10-0-1-12 ~]$ aws s3 cp ls s3://nextwork-vpc-project-alex
[ec2-user@ip-10-0-1-12 ~]$ aws s3 cp test.txt s3://nextwork-vpc-project-alex
[ec2-user@ip-10-0-1-12 ~]$ aws s3 cp /tmp/test.txt s3://nextwork-vpc-project-alex
[ec2-user@ip-10-0-1-12 ~]$
```



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

