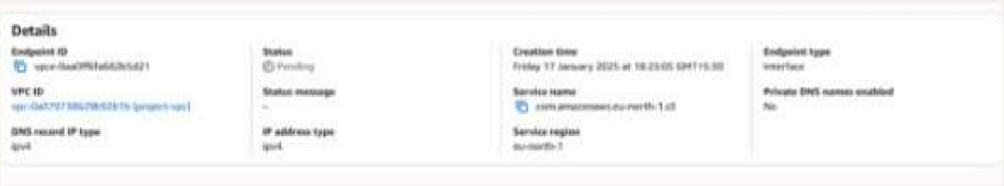
VPC Endpoints

SA samahae1530@gmail.com



Details

Endpoint ID
vpce-0aa0f6fa6b2b5d21

VPC ID
vpc-0a57973db29b92b1b (project-vpc)

DNS record IP type
ipv4

Status
Pending

Status message
–

IP address type
ipv4

Creation time
Friday 17 January 2025 at 18:23:05 GMT+5:30

Service name
com.amazonaws.eu-north-1.s3

Service region
eu-north-1

Endpoint type
Interface

Private DNS names enabled
No

In the first part of my project...

Step 1 - Architecture set up

in this project, setting up the foundations of this project like launching VPC, EC2 instance and S3 bucket so that i can setup endpoint and test the setup in the last step of this project.

Step 2 - Connect to EC2 instance

In this step connecting directly to EC2 instance will help with accessing S3 and running commands later in this project.

Step 3 - Set up access keys

In this step will set up an a access key so EC2 instance will have access to AWS environment

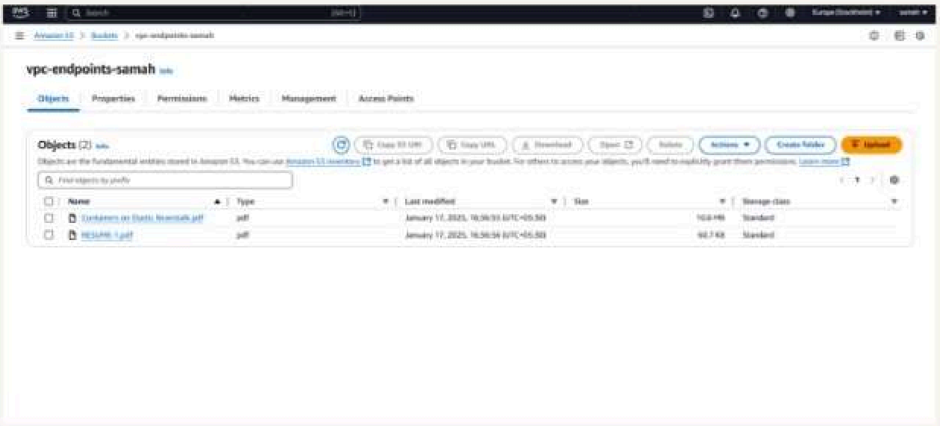
Step 4 - Interact with S3 bucket

In this step i am applying access key credentials to EC2 instance and then using AWs CLI and EC2 instance to access Amazon s3.

Architecture set up

Started project by launching three key resorces - a VPC, EC2 instance and S3 bucket.

I also setup an S3 bucket with two files inside,



vpc-endpoints-samah

Objects (2)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name	Type	Last modified	Size	Storage class
□	Containers on Elastic Beanstalk.pdf	pdf	January 17, 2025, 16:56:55 (UTC+05:30)	10.6 MB	Standard
□	RESUME-1.pdf	pdf	January 17, 2025, 16:56:56 (UTC+05:30)	60.7 KB	Standard

Access keys

Credentials

To set up my EC2 instance to intract with my AWS environment ,i configured the AWS access key ID, secret access key that matches the key ID and default region type.

Access Keys are a set of security credentials used to authenticate programmatic access to AWS services via the AWS Command Line Interface (CLI) or APIs.

A Secret Access Key is the private part of AWS credentials, used with the Access Key ID to securely sign requests to AWS services. It acts like a password, so keep it safe and never share it to avoid unauthorized access.

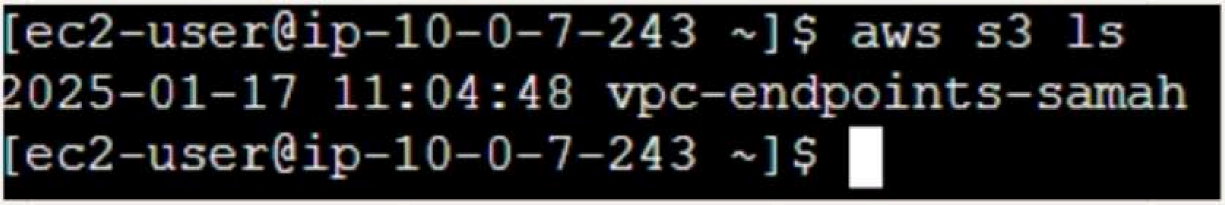
Best practice

Although I'm using access keys in this project, a best practice alternative is to use IAM admin role instead. This means any necessary permissions will be attached to an IAM role will be associated with the relevant resources.

Connecting to my S3 bucket

The command I ran was 'aws s3 ls. this command is used to list all buckets in an aws account.

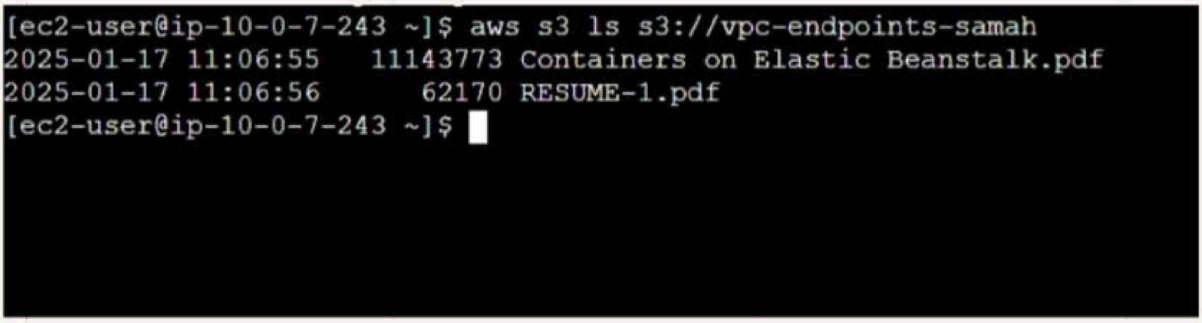
The terminal responded with list of my accounts S3 buckets. This indicated that the access keys I set up correctly give my EC2 instance to my AWS account and environment.



```
[ec2-user@ip-10-0-7-243 ~]$ aws s3 ls
2025-01-17 11:04:48 vpc-endpoints-samah
[ec2-user@ip-10-0-7-243 ~]$ 
```

Connecting to my S3 bucket

I also tested the command aws s3 ls s3://vpc-endpoint-samah. which returned a list of all the objects inside that S3 bucket.



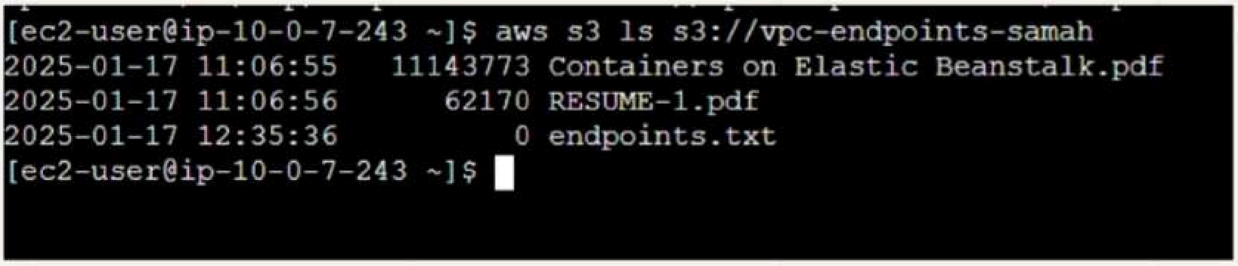
```
[ec2-user@ip-10-0-7-243 ~]$ aws s3 ls s3://vpc-endpoints-samah
2025-01-17 11:06:55   11143773 Containers on Elastic Beanstalk.pdf
2025-01-17 11:06:56      62170 RESUME-1.pdf
[ec2-user@ip-10-0-7-243 ~]$ 
```

Uploading objects to S3

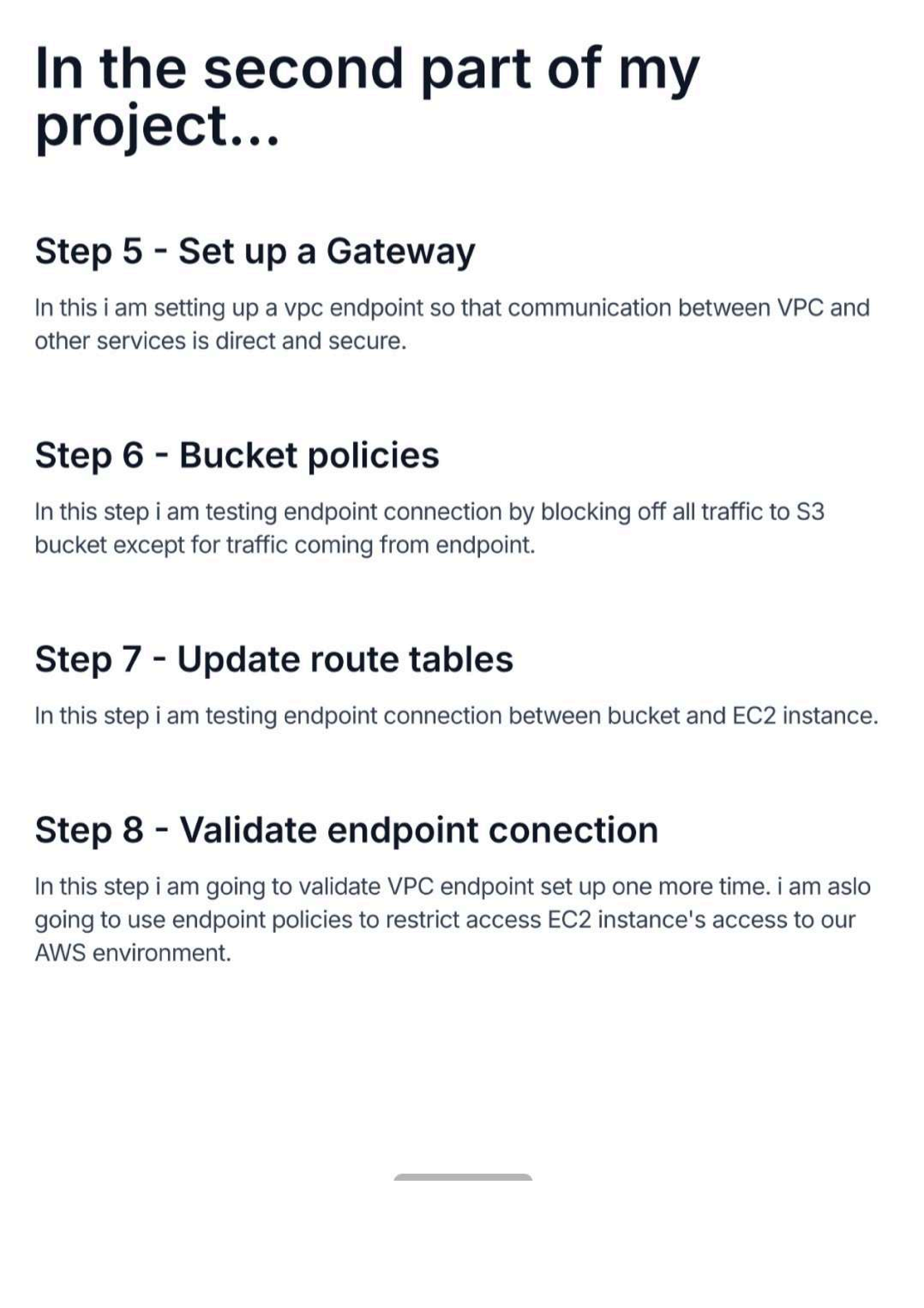
To upload a new file to my bucket, I first ran the command sudo touch /tmp/endpoints.txt. This command creates an empty file named endpoints.txt and saves locally in the EC2 instance.

The second command I ran was aws s3 cp /tmp/endpoints.txt s3://vpc-endpoints-samah. This command will copy the file i created i.e. endpoints.txt and upload to my S3 bucket.

The third command I ran was aws s3 ls s3://vpc-endpoints-samah which validated that a new file was created and uploaded into my S3 bucket.



```
[ec2-user@ip-10-0-7-243 ~]$ aws s3 ls s3://vpc-endpoints-samah
2025-01-17 11:06:55   11143773 Containers on Elastic Beanstalk.pdf
2025-01-17 11:06:56      62170 RESUME-1.pdf
2025-01-17 12:35:36          0 endpoints.txt
[ec2-user@ip-10-0-7-243 ~]$ 
```



In the second part of my project...

Step 5 - Set up a Gateway

In this i am setting up a vpc endpoint so that communication between VPC and other services is direct and secure.

Step 6 - Bucket policies

In this step i am testing endpoint connection by blocking off all traffic to S3 bucket except for traffic coming from endpoint.

Step 7 - Update route tables

In this step i am testing endpoint connection between bucket and EC2 instance.

Step 8 - Validate endpoint conection

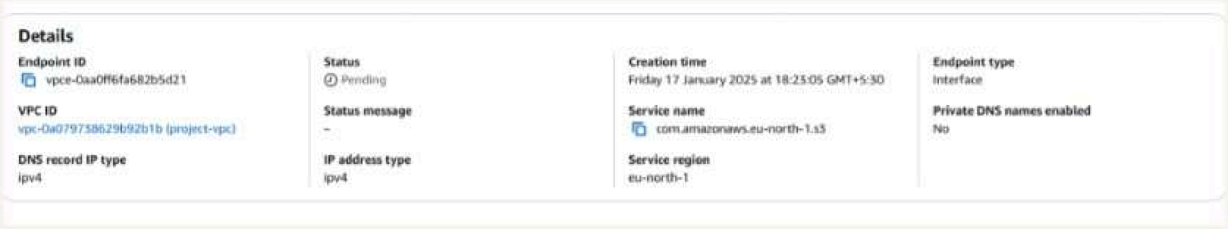
In this step i am going to validate VPC endpoint set up one more time. i am aslo going to use endpoint policies to restrict access EC2 instance's access to our AWS environment.

Setting up a Gateway

I set up an S3 Gateway, which is a type of endpoint specifically designed for Amazon S3. Gateways work by updating the route table of associated subnets, so that S3 bound traffic goes through the Gateway instead of the internet.

What are endpoints?

An endpoint is a connection point that allows your resources (like EC2 instances or VPCs) to securely connect to AWS services, such as S3 or DynamoDB.



Details

Endpoint ID
vpce-0aa0ff6fa682b5d21

VPC ID
vpc-0a079738629b92b1b (project-vpc)

DNS record IP type
ipv4

Status
Pending

Status message
-

IP address type
ipv4

Creation time
Friday 17 January 2025 at 18:23:05 GMT+5:30

Service name
com.amazonaws.eu-north-1.s3

Service region
eu-north-1

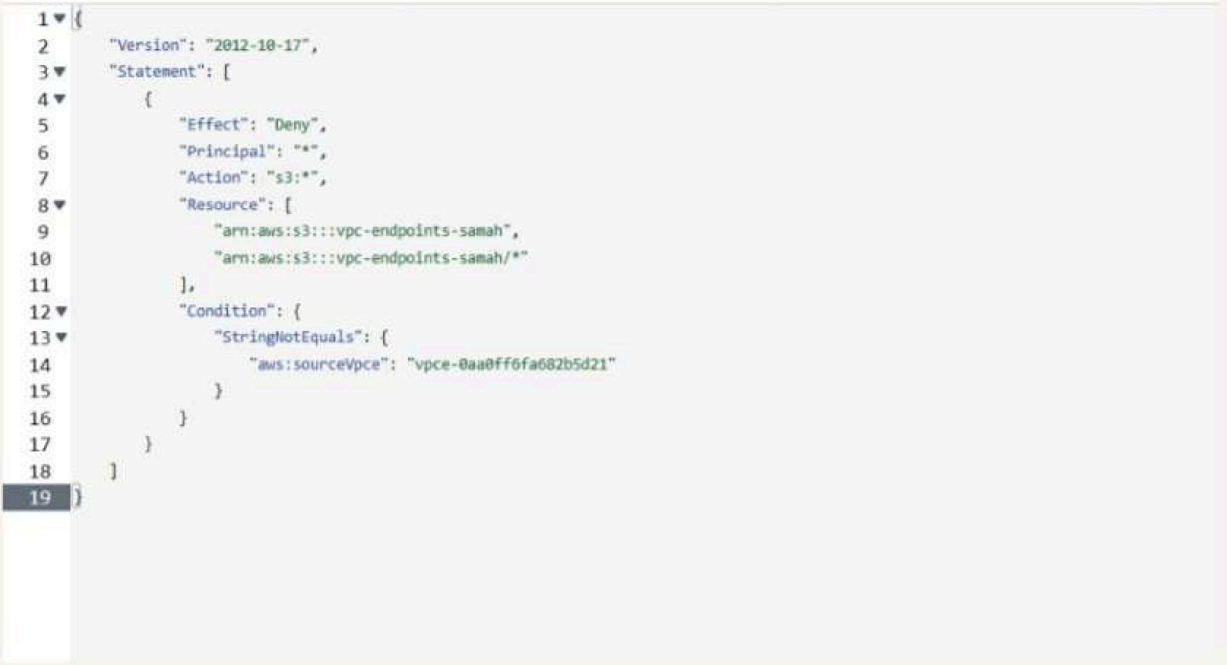
Endpoint type
Interface

Private DNS names enabled
No

Bucket policies

A bucket policy in AWS is a set of permissions attached to an S3 bucket. It defines who can access the bucket's data and what actions they can perform (like read, write, delete), helping control access to the stored files.

My bucket policy will deny traffic from all sources except for traffic coming from Vpc endpoint.

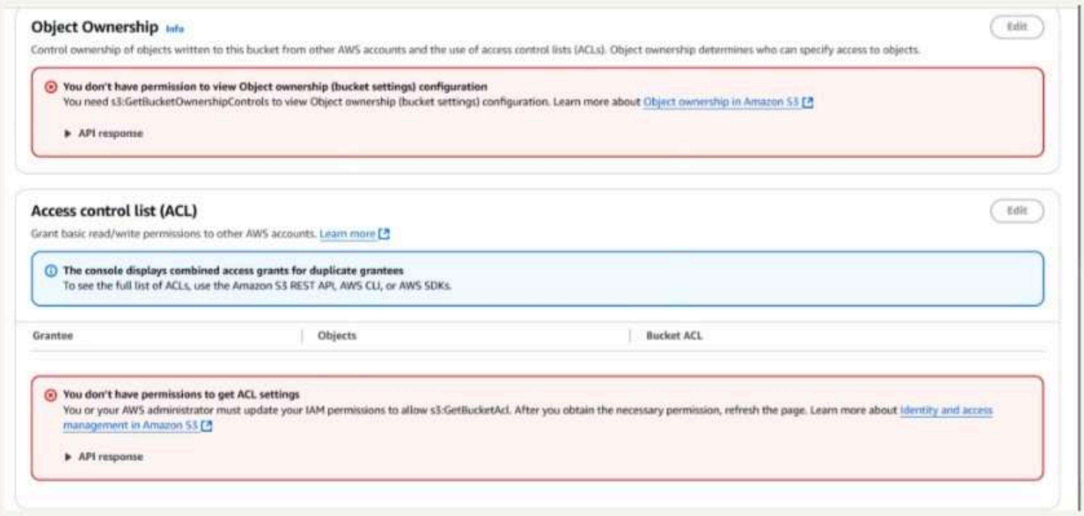


```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:::vpc-endpoints-samah",
                "arn:aws:s3:::vpc-endpoints-samah/*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:sourceVpce": "vpce-0aa0ff6fa682b5d21"
                }
            }
        }
    ]
}
```

Bucket policies

Right after saving my bucket policy, my S3 bucket page showed 'denied access' warnings. This was because my bucket is denying all traffic that doesn't come from my endpoint i.e the policy denies traffic from AWS management console.

I also had to update my route table because my route table by default didn't provide a route for traffic in my public subnet to the VPC endpoint.



Object Ownership

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

You don't have permission to view Object ownership (bucket settings) configuration
You need s3:GetBucketOwnershipControls to view Object ownership (bucket settings) configuration. Learn more about Object ownership in Amazon S3

▶ API response

Access control list (ACL)

Grant basic read/write permissions to other AWS accounts. Learn more

The console displays combined access grants for duplicate grantees
To see the full list of ACLs, use the Amazon S3 REST API, AWS CLI, or AWS SDKs.

Grantee	Objects	Bucket ACL

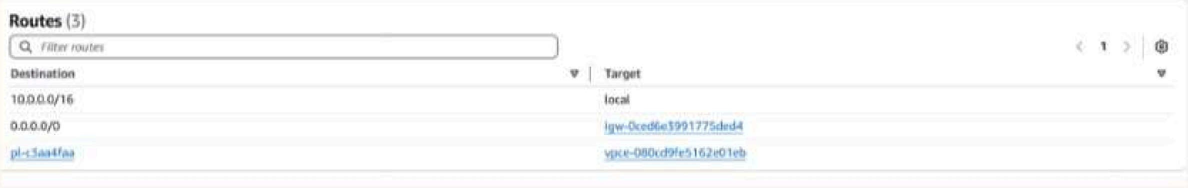
You don't have permissions to get ACL settings
You or your AWS administrator must update your IAM permissions to allow s3:GetBucketAcl. After you obtain the necessary permission, refresh the page. Learn more about Identity and access management in Amazon S3

▶ API response

Route table updates

To update my route table i visited the endpoints page of my VPC console and modified the the routetable from there associate VPC's public subnet.

After updating my public subnet's route table, my terminal could connect with S3 bucket , access was no longer denied.



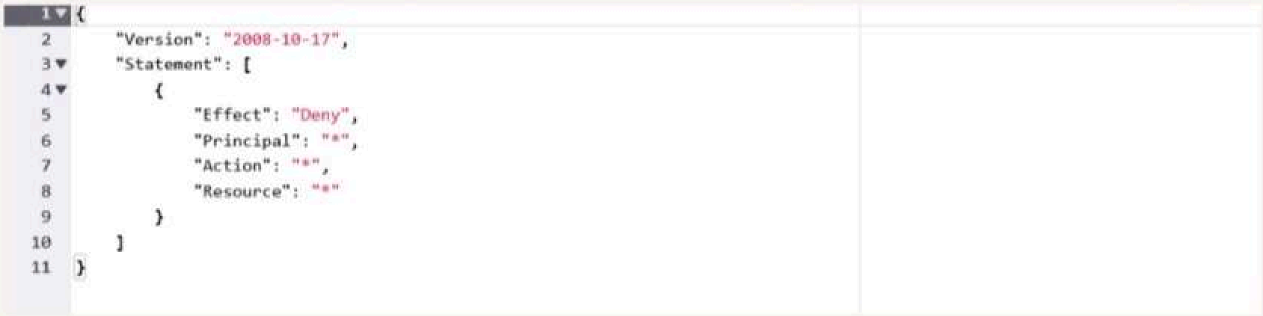
Routes (3)

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-0ced6e3991775ded4
pl-c3aa4faa	vpce-080cd9fe5162e01eb

Endpoint policies

An endpoint policy is type of policy designed for specifying the range of resources and actions permitted by an endpoint.

I updated my endpoint's policy by the changing the effect from 'allow' to 'deny', i could see the effect right away because my EC2 was again denied access to S3 when i tried to run 'aws s3' command.



```
{
    "Version": "2008-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Principal": "*",
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```