Assignment 1 (Part B) - Creating and deploying Photo Album website on a basic AWS infrastructure

Name : Md Nahid Tanjum
Student ID: 103807068
*Tutorial Class: Class 1-11 Friday,
2:30 PM – 4:30 PM at BA407*

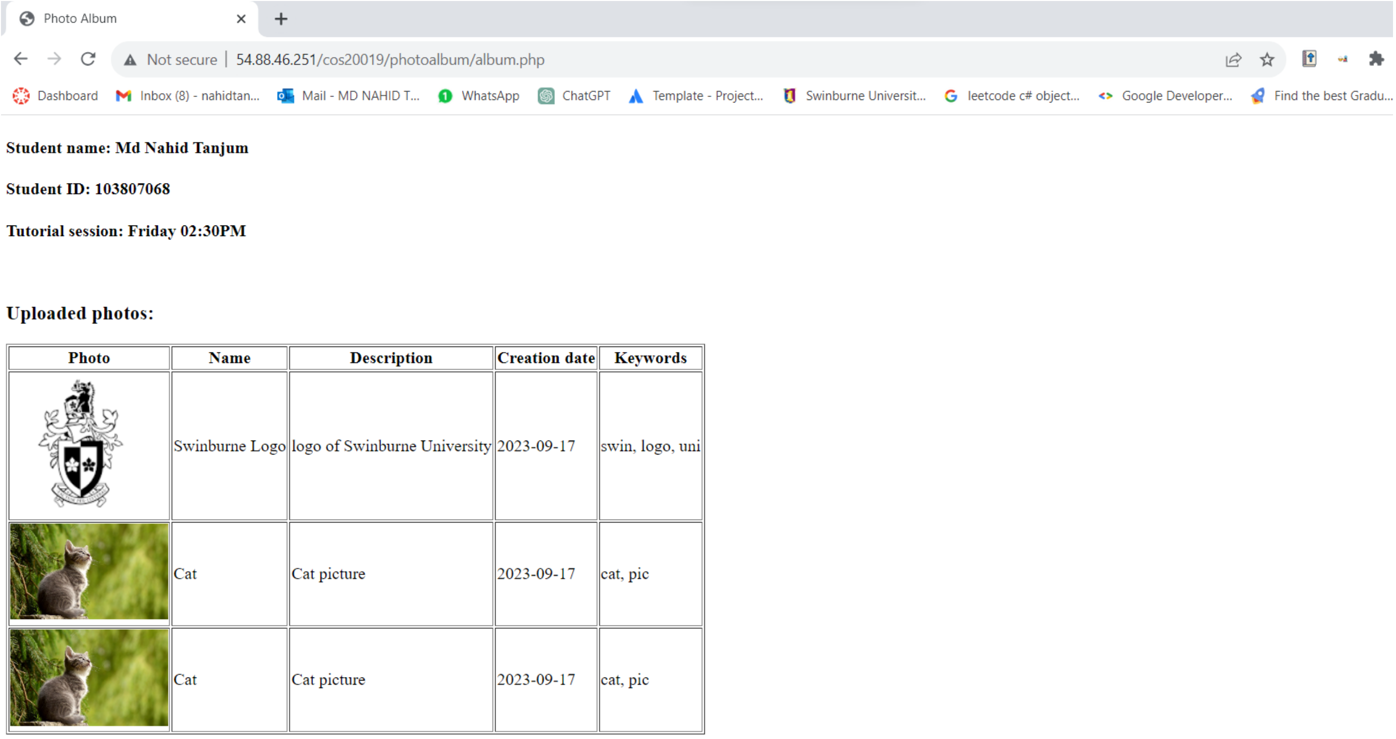
I. Introduction

In this initiative, fundamental AWS frameworks and resources were utilized to launch the Photo Album Website. All essential recommendations from the AWS Academy were incorporated during the rollout, leading to a successful deployment.

II. Photo Album

The primary objective was to make the Photo Album website live, hosted on the "WebServer" EC2 instance, and accessible to the public through an elastic IP address. During this phase, the image file was stored in the "nahids3bucket" S3 bucket, while textual data was saved in the "rds-nahid" AWS RDS. The site was displayed using PHP, drawing from the data stored in the backend AWS services.

URL: http://54.88.46.251/cos20019/photoalbum/album.php



Student name: Md Nahid Tanjum

Student ID: 103807068

Tutorial session: Friday 02:30PM

Uploaded photos:

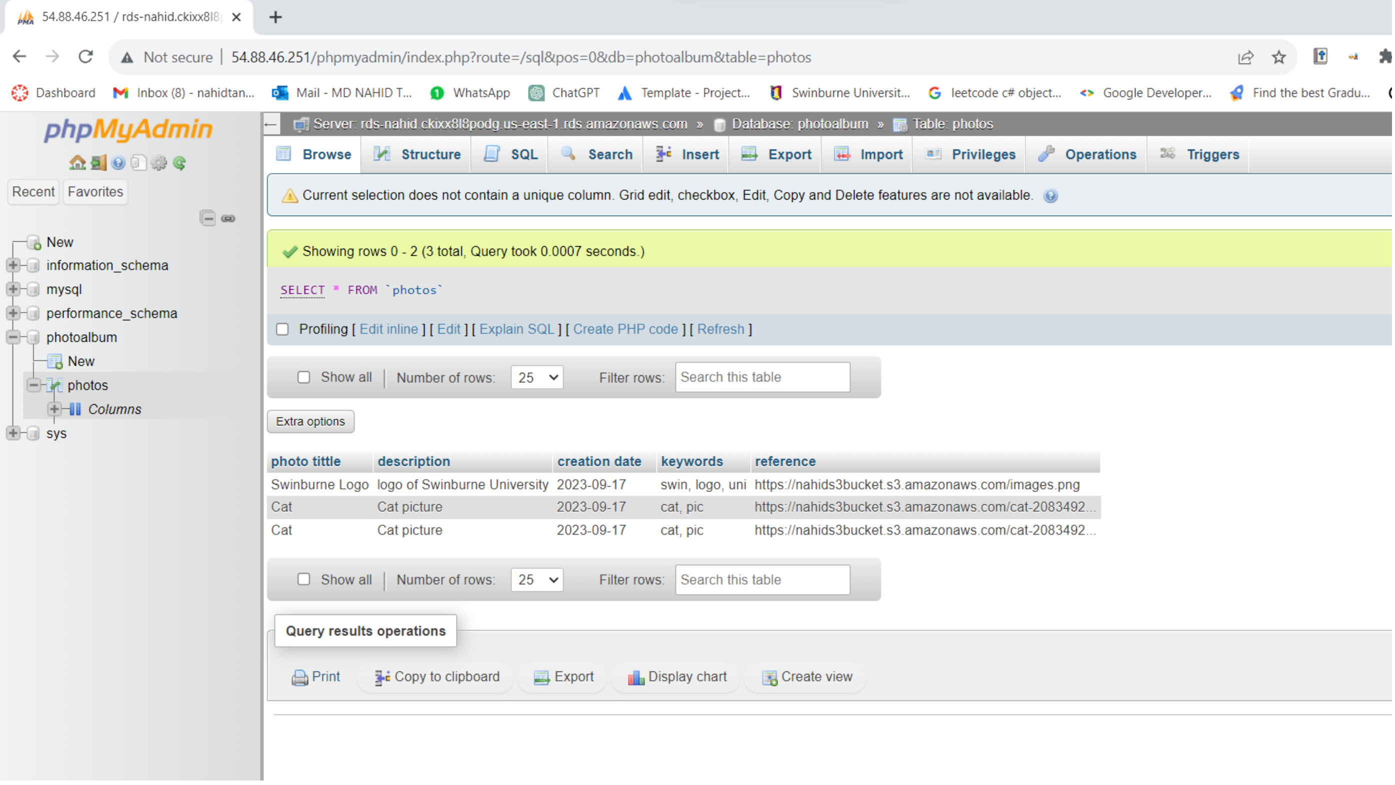
Photo	Name	Description	Creation date	Keywords
	Swinburne Logo	logo of Swinburne University	2023-09-17	swin, logo, uni
	Cat	Cat picture	2023-09-17	cat, pic
	Cat	Cat picture	2023-09-17	cat, pic

Figure 1 – Photo Album Web Page

III. DATA RECORDS IN THE DATABASE

The RDS data has been managed using phpMyAdmin. According to the deployment's requirements, the data has been recorded.

Student ID: 103807068 ©Md Nahid Tanjum



Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 2 (3 total, Query took 0.0007 seconds.)

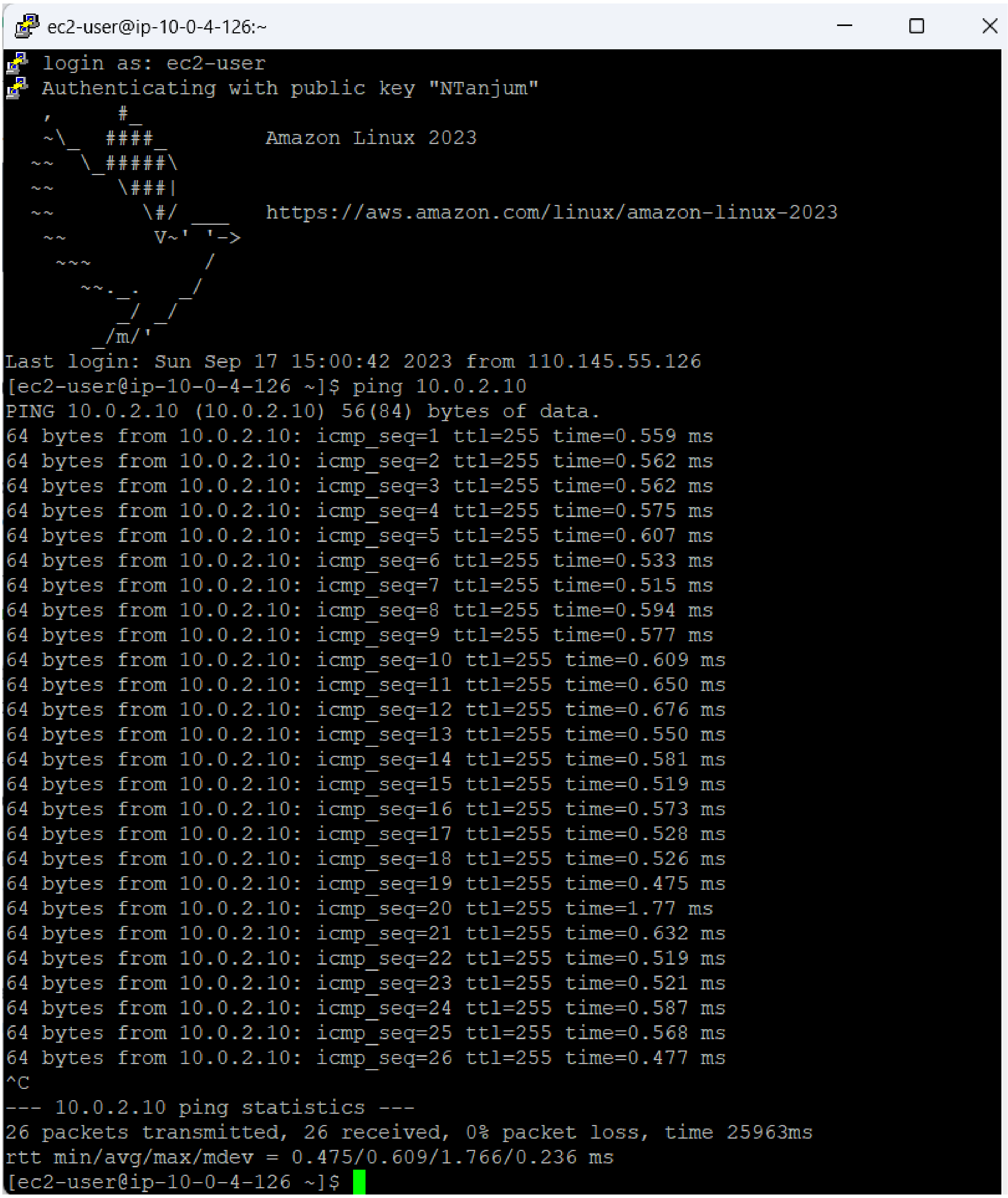
```
SELECT * FROM `photos`
```

photo tittle	description	creation date	keywords	reference
Swinburne Logo	logo of Swinburne University	2023-09-17	swin, logo, uni	https://nahids3bucket.s3.amazonaws.com/images.png
Cat	Cat picture	2023-09-17	cat, pic	https://nahids3bucket.s3.amazonaws.com/cat-2083492...
Cat	Cat picture	2023-09-17	cat, pic	https://nahids3bucket.s3.amazonaws.com/cat-2083492...

Figure 2 – Database Records

IV. Successful Ping to the Web Server Instance from the Test Server

Through the Linux console, the ICMP connectivity has been tested from the "Test instance" to the "Webserver." The keypair connected to the "Test instance" has been used to access the Linux terminal through the SSH(22) protocol.



ec2-user@ip-10-0-4-126:~

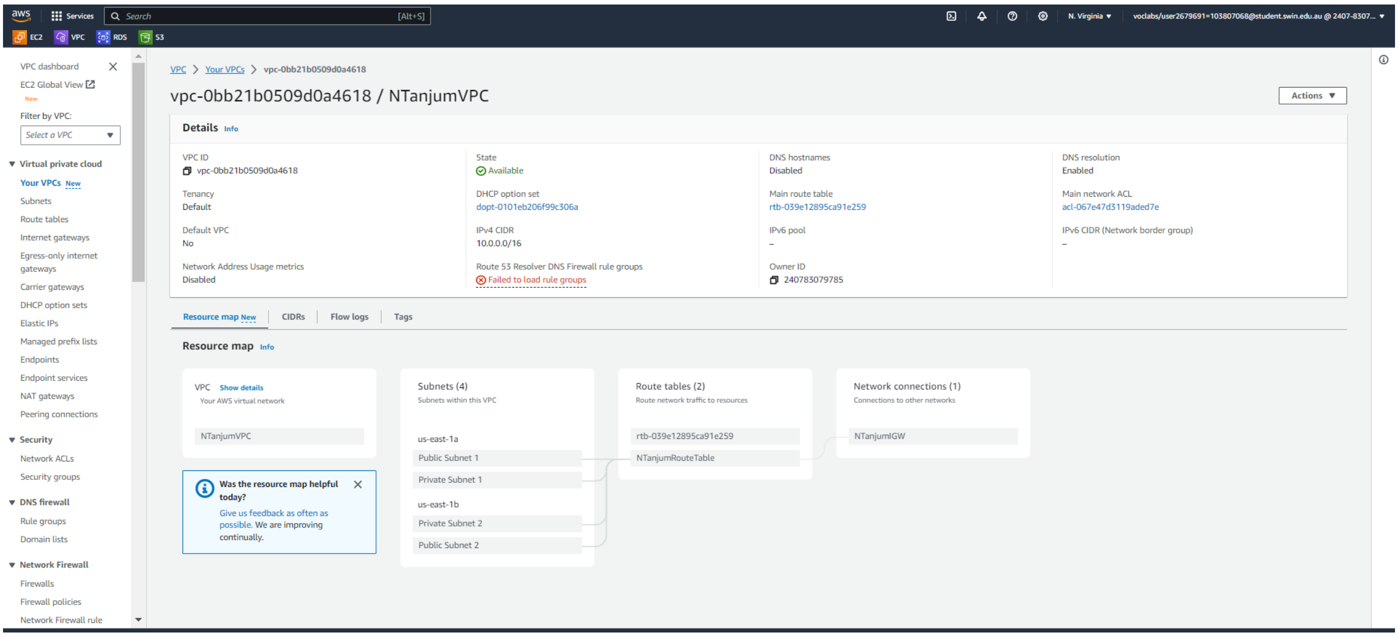
```
 login as: ec2-user
 Authenticating with public key "NTanjum"
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Sun Sep 17 15:00:42 2023 from 110.145.55.126
[ec2-user@ip-10-0-4-126 ~]$ ping 10.0.2.10
PING 10.0.2.10 (10.0.2.10) 56(84) bytes of data.
64 bytes from 10.0.2.10: icmp_seq=1 ttl=255 time=0.559 ms
64 bytes from 10.0.2.10: icmp_seq=2 ttl=255 time=0.562 ms
64 bytes from 10.0.2.10: icmp_seq=3 ttl=255 time=0.562 ms
64 bytes from 10.0.2.10: icmp_seq=4 ttl=255 time=0.575 ms
64 bytes from 10.0.2.10: icmp_seq=5 ttl=255 time=0.607 ms
64 bytes from 10.0.2.10: icmp_seq=6 ttl=255 time=0.533 ms
64 bytes from 10.0.2.10: icmp_seq=7 ttl=255 time=0.515 ms
64 bytes from 10.0.2.10: icmp_seq=8 ttl=255 time=0.594 ms
64 bytes from 10.0.2.10: icmp_seq=9 ttl=255 time=0.577 ms
64 bytes from 10.0.2.10: icmp_seq=10 ttl=255 time=0.609 ms
64 bytes from 10.0.2.10: icmp_seq=11 ttl=255 time=0.650 ms
64 bytes from 10.0.2.10: icmp_seq=12 ttl=255 time=0.676 ms
64 bytes from 10.0.2.10: icmp_seq=13 ttl=255 time=0.550 ms
64 bytes from 10.0.2.10: icmp_seq=14 ttl=255 time=0.581 ms
64 bytes from 10.0.2.10: icmp_seq=15 ttl=255 time=0.519 ms
64 bytes from 10.0.2.10: icmp_seq=16 ttl=255 time=0.573 ms
64 bytes from 10.0.2.10: icmp_seq=17 ttl=255 time=0.528 ms
64 bytes from 10.0.2.10: icmp_seq=18 ttl=255 time=0.526 ms
64 bytes from 10.0.2.10: icmp_seq=19 ttl=255 time=0.475 ms
64 bytes from 10.0.2.10: icmp_seq=20 ttl=255 time=1.77 ms
64 bytes from 10.0.2.10: icmp_seq=21 ttl=255 time=0.632 ms
64 bytes from 10.0.2.10: icmp_seq=22 ttl=255 time=0.519 ms
64 bytes from 10.0.2.10: icmp_seq=23 ttl=255 time=0.521 ms
64 bytes from 10.0.2.10: icmp_seq=24 ttl=255 time=0.587 ms
64 bytes from 10.0.2.10: icmp_seq=25 ttl=255 time=0.568 ms
64 bytes from 10.0.2.10: icmp_seq=26 ttl=255 time=0.477 ms
^C
--- 10.0.2.10 ping statistics ---
26 packets transmitted, 26 received, 0% packet loss, time 25963ms
rtt min/avg/max/mdev = 0.475/0.609/1.766/0.236 ms
[ec2-user@ip-10-0-4-126 ~]$
```

Figure 3 – Linux Terminal of the Test instance

V. Deployment Steps

A. VPC, Subnets, IGW, NACL

VPC: To accommodate all of the Public and Private Subnets, a VPC with the subnet 10.0.0.0/16 has been constructed in two separate availability zones (AZ-A and AZ-B).



vpc-0bb21b0509d0a4618 / NTanjumVPC

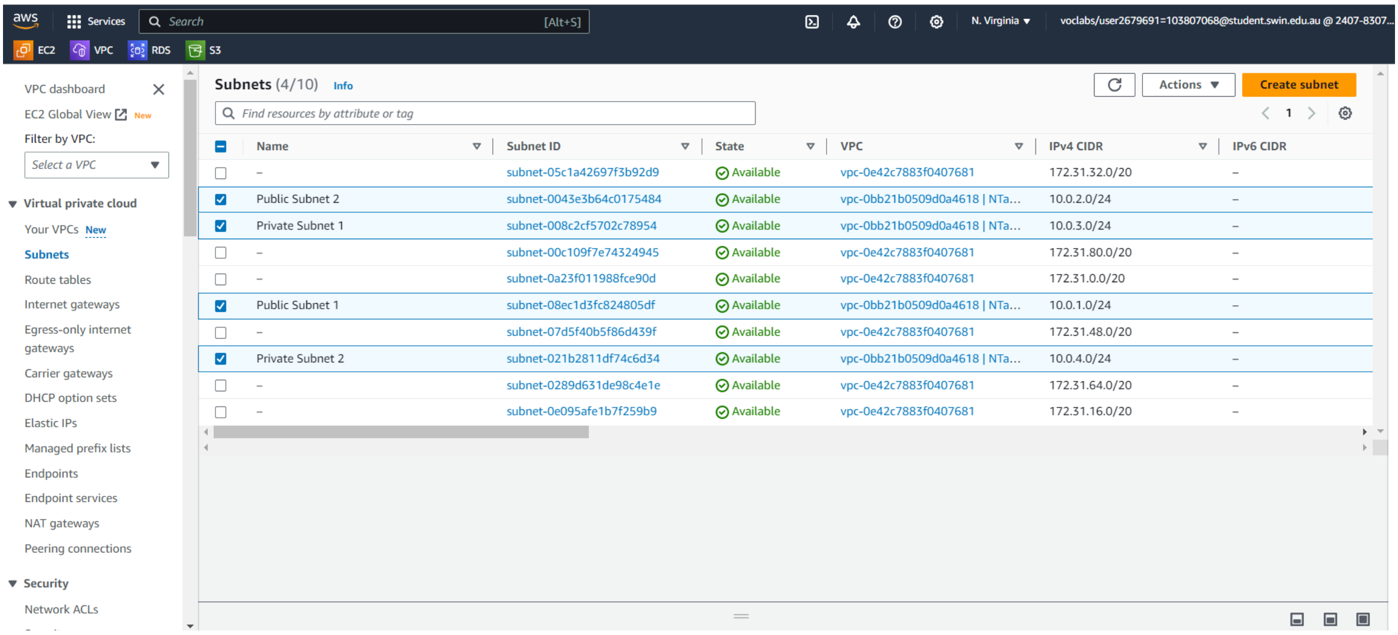
Details Info

VPC ID: vpc-0bb21b0509d0a4618
State: Available
DNS hostnames: Disabled
DNS resolution: Enabled
Tenancy: Default
DHCP option set: dopt-0101eb206f99c306a
Main route table: rtb-039e12895ca91e259
Main network ACL: acl-067e47d3119aded7e
Default VPC: No
IPv4 CIDR: 10.0.0.0/16
IPv6 pool: –
IPv6 CIDR (Network border group): –
Network Address Usage metrics: Disabled
Route 53 Resolver DNS Firewall rule groups: Failed to load rule groups
Owner ID: 240783079785

Resource map Info

Figure 4 – VPC

Subnets: Four subnets has been created within the VPC for separation of the network.

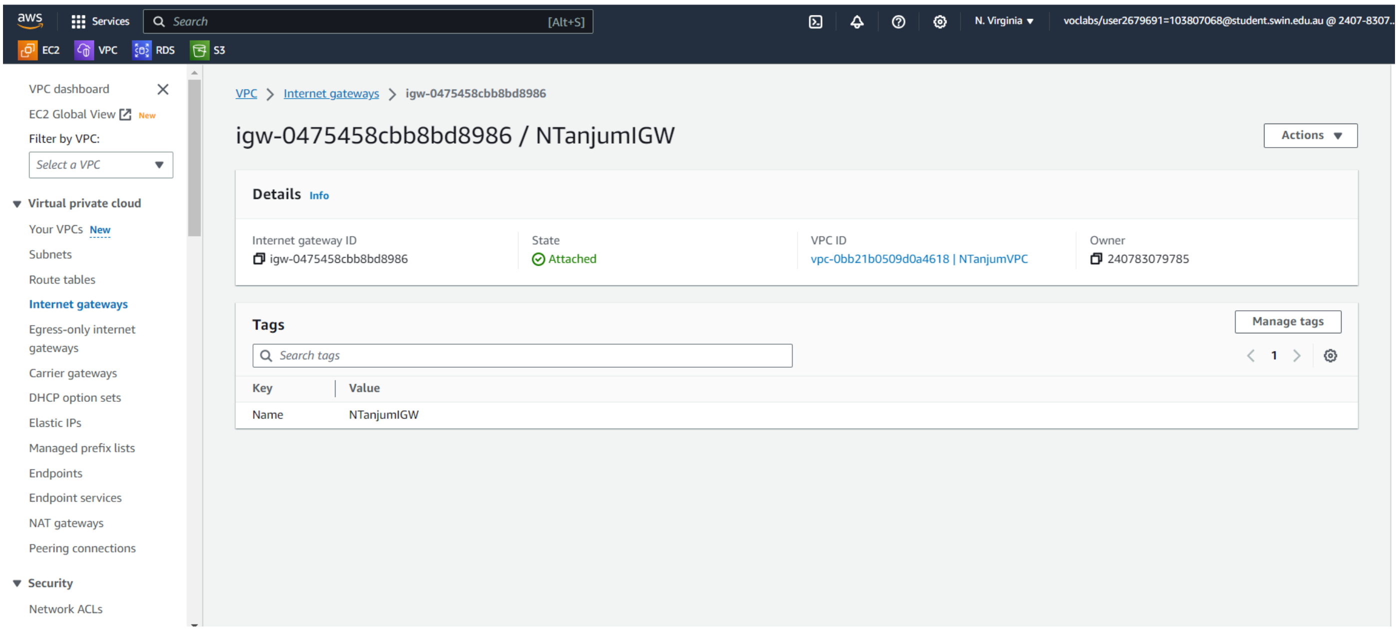


Subnets (4/10) Info

	Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
□	–	subnet-05c1a42697f3b92d9	Available	vpc-0e42c7883f0407681	172.31.32.0/20	–
☑	Public Subnet 2	subnet-0043e3b64c0175484	Available	vpc-0bb21b0509d0a4618 | NTa...	10.0.2.0/24	–
☑	Private Subnet 1	subnet-008c2cf5702c78954	Available	vpc-0bb21b0509d0a4618 | NTa...	10.0.3.0/24	–
□	–	subnet-00c109f7e74324945	Available	vpc-0e42c7883f0407681	172.31.80.0/20	–
□	–	subnet-0a23f011988fce90d	Available	vpc-0e42c7883f0407681	172.31.0.0/20	–
☑	Public Subnet 1	subnet-08ec1d3fc824805df	Available	vpc-0bb21b0509d0a4618 | NTa...	10.0.1.0/24	–
□	–	subnet-07d5f40b5f86d439f	Available	vpc-0e42c7883f0407681	172.31.48.0/20	–
☑	Private Subnet 2	subnet-021b2811df74c6d34	Available	vpc-0bb21b0509d0a4618 | NTa...	10.0.4.0/24	–
□	–	subnet-0289d631de98c4e1e	Available	vpc-0e42c7883f0407681	172.31.64.0/20	–
□	–	subnet-0e095afe1b7f259b9	Available	vpc-0e42c7883f0407681	172.31.16.0/20	–

Figure 5 - Subnets

Internet Gateway: In order to implement routeing within the subnets and the internet, an Internet Gateway has been constructed.



VPC > Internet gateways > igw-0475458cbb8bd8986

igw-0475458cbb8bd8986 / NTanjumIGW

Details

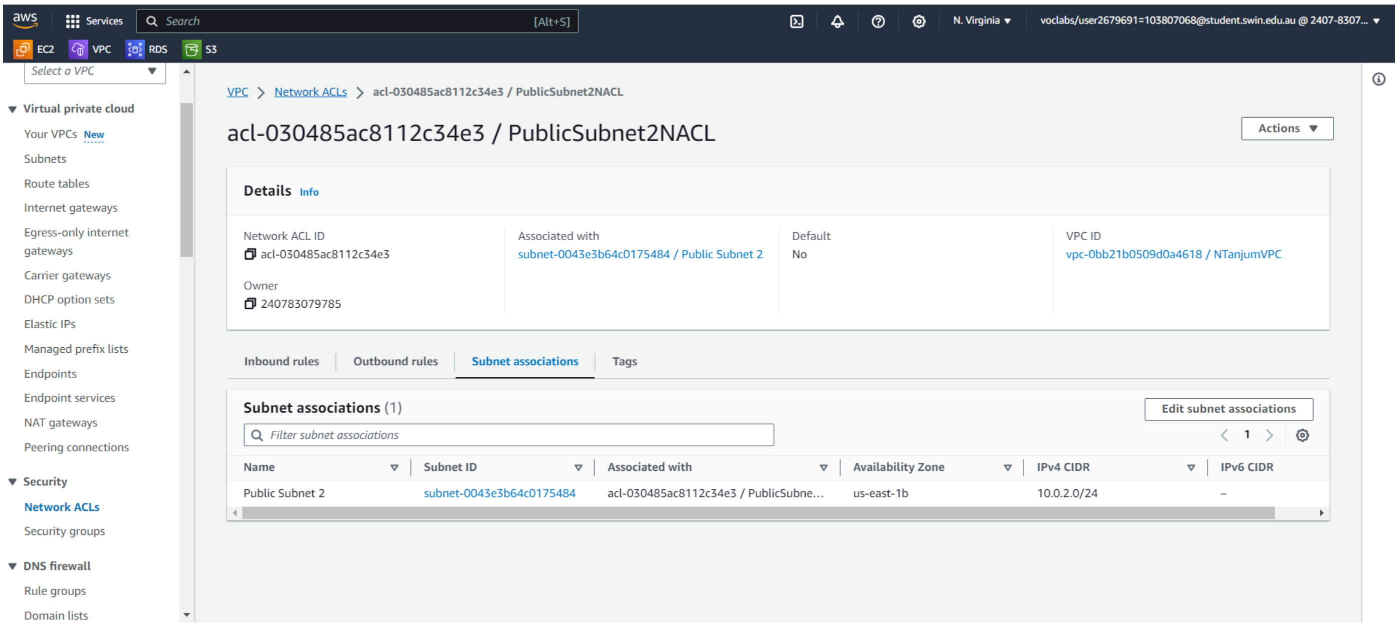
Internet gateway ID	State	VPC ID	Owner
igw-0475458cbb8bd8986	Attached	vpc-0bb21b0509d0a4618 | NTanjumVPC	240783079785

Tags

Key	Value
Name	NTanjumIGW

Figure 6 – Internet Gateway

Network ACL: A NACL has been set up with the necessary inbound and outbound rules to limit access to and from the Public Subnet 2 (10.0.2.0/24) in Availability Zone B.



VPC > Network ACLs > acl-030485ac8112c34e3 / PublicSubnet2NACL

acl-030485ac8112c34e3 / PublicSubnet2NACL

Details

Network ACL ID	Associated with	Default	VPC ID
acl-030485ac8112c34e3	subnet-0043e3b64c0175484 / Public Subnet 2	No	vpc-0bb21b0509d0a4618 / NTanjumVPC

Owner
240783079785

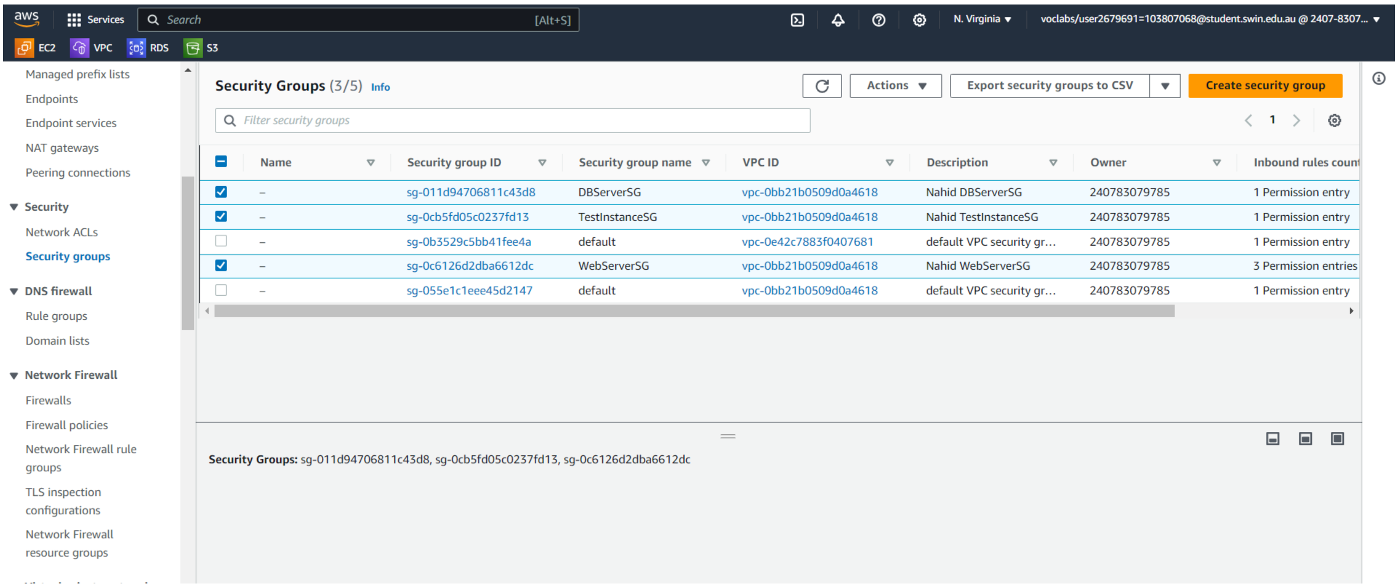
Subnet associations (1)

Name	Subnet ID	Associated with	Availability Zone	IPv4 CIDR	IPv6 CIDR
Public Subnet 2	subnet-0043e3b64c0175484	acl-030485ac8112c34e3 / PublicSubne...	us-east-1b	10.0.2.0/24	–

Figure 7 – Network ACL

B. Security Groups

With the stated access restrictions and associated to the appropriate AWS services, Security Groups have been built to function as a firewall.



Security Groups (3/5) Info

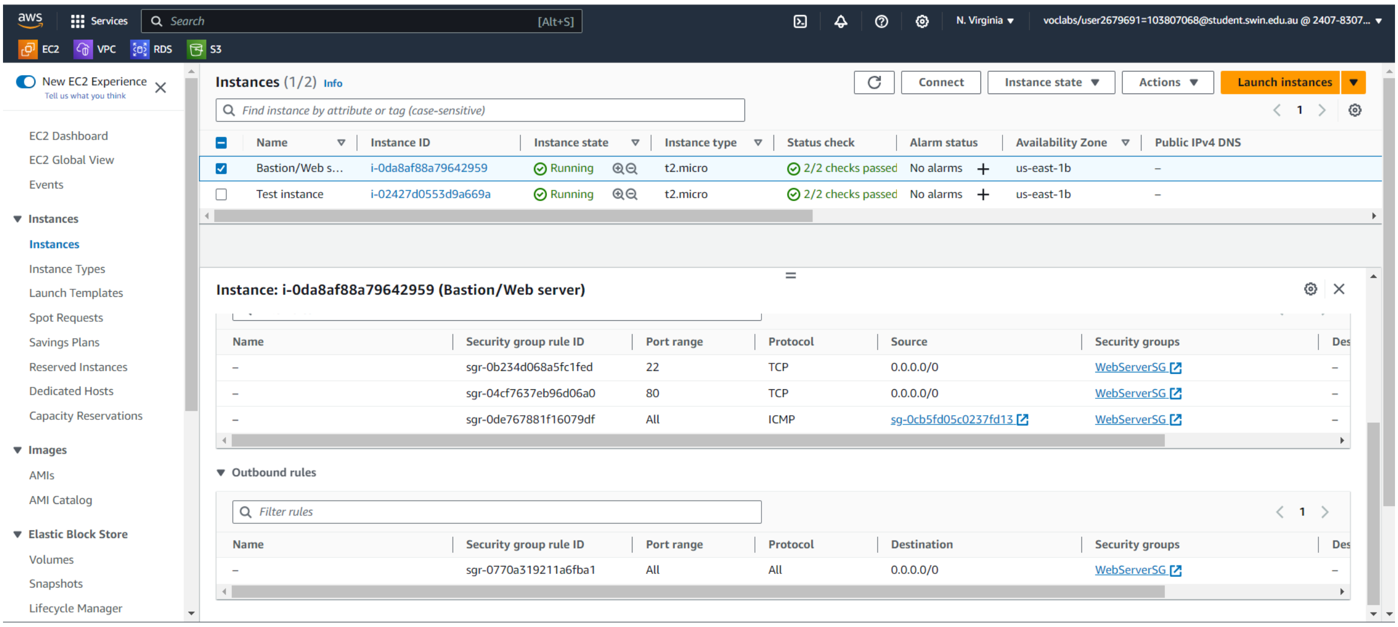
☐	Name	Security group ID	Security group name	VPC ID	Description	Owner	Inbound rules count
☑	–	sg-011d94706811c43d8	DBServerSG	vpc-0bb21b0509d0a4618	Nahid DBServerSG	240783079785	1 Permission entry
☑	–	sg-0cb5fd05c0237fd13	TestInstanceSG	vpc-0bb21b0509d0a4618	Nahid TestInstanceSG	240783079785	1 Permission entry
☐	–	sg-0b3529c5bb41fee4a	default	vpc-0e42c7883f0407681	default VPC security gr...	240783079785	1 Permission entry
☑	–	sg-0c6126d2dba6612dc	WebServerSG	vpc-0bb21b0509d0a4618	Nahid WebServerSG	240783079785	3 Permission entries
☐	–	sg-055e1c1eee45d2147	default	vpc-0bb21b0509d0a4618	default VPC security gr...	240783079785	1 Permission entry

Security Groups: sg-011d94706811c43d8, sg-0cb5fd05c0237fd13, sg-0c6126d2dba6612dc

Figure 8 – Security Groups

C. EC2 Instances

The web application "Photo Album" is being hosted by the "Bastion/Webserver" webserver instance. Additionally, it enables SSH access to the "Test instance" that is located in the public subnet when used as a Bastion host.



Instances (1/2) Info

☐	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
☑	Bastion/Web s...	i-0da8af88a79642959	Running	t2.micro	2/2 checks passed	No alarms +	us-east-1b	–
☐	Test instance	i-02427d0553d9a669a	Running	t2.micro	2/2 checks passed	No alarms +	us-east-1b	–

Instance: i-0da8af88a79642959 (Bastion/Web server)

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Des
–	sgr-0b234d068a5fc1fed	22	TCP	0.0.0.0/0	WebServerSG	–
–	sgr-04cf7637eb96d06a0	80	TCP	0.0.0.0/0	WebServerSG	–
–	sgr-0de767881f16079df	All	ICMP	sg-0cb5fd05c0237fd13	WebServerSG	–

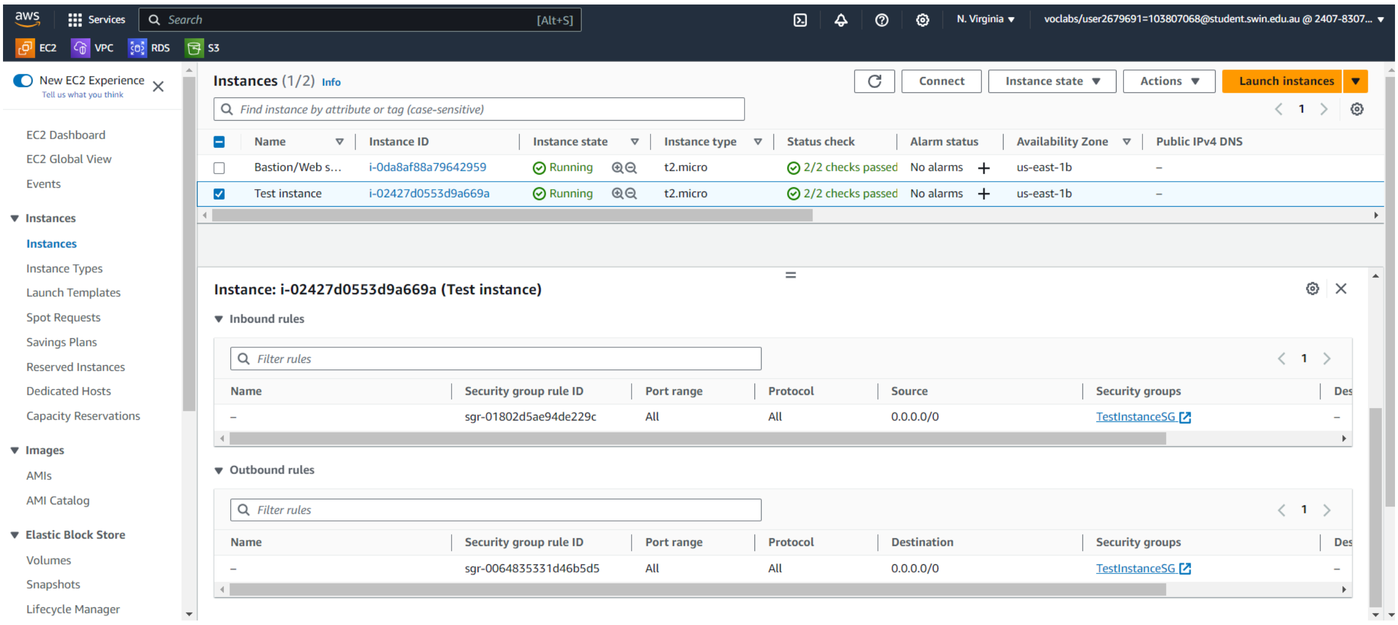
Outbound rules

Filter rules

Name	Security group rule ID	Port range	Protocol	Destination	Security groups	Des
–	sgr-0770a319211a6fba1	All	All	0.0.0.0/0	WebServerSG	–

Figure 9 – Webserver Instance

"test instance" has been set up to check the Network ACl policies' ability to connect the Public Subnet 2 (10.0.2.0/24) and Private Subnet 2 (10.0.4.0/24) networks.



	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
□	Bastion/Web s...	i-0da8af88a79642959	Running	t2.micro	2/2 checks passed	No alarms +	us-east-1b	–
☑	Test instance	i-02427d0553d9a669a	Running	t2.micro	2/2 checks passed	No alarms +	us-east-1b	–

Instance: i-02427d0553d9a669a (Test instance)

Inbound rules

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Des
–	sgr-01802d5ae94de229c	All	All	0.0.0.0/0	TestInstanceSG	–

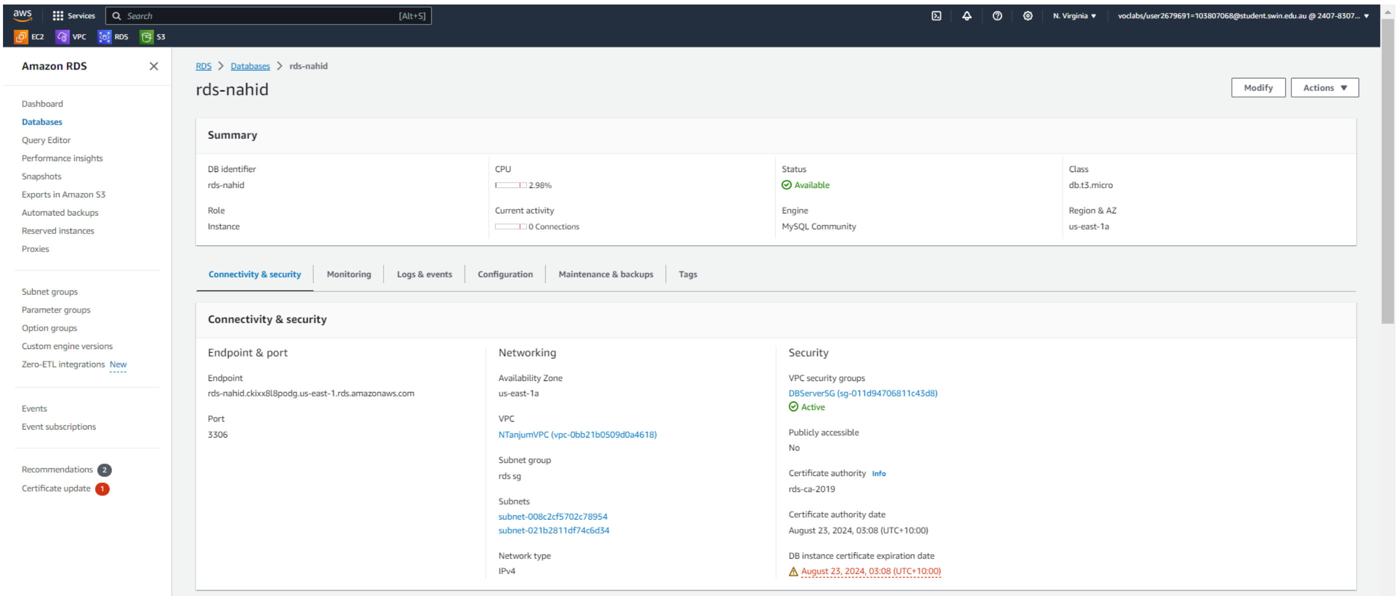
Outbound rules

Name	Security group rule ID	Port range	Protocol	Destination	Security groups	Des
–	sgr-0064835331d46b5d5	All	All	0.0.0.0/0	TestInstanceSG	–

Figure 10 – Test Instance

D. RDS

To store data with the appropriate subnet groups and security groups applied, MySQL 8.3.34 was used to construct the single zone database instance "rds-nahid".



rds-nahid

Summary

DB identifier: rds-nahid
Role: Instance
CPU: 2.98%
Current activity: 0 Connections
Status: Available
Engine: MySQL Community
Class: db.t3.micro
Region & AZ: us-east-1a

Connectivity & security

Endpoint & port
Endpoint: rds-nahid.ckixx8l8podg.us-east-1.rds.amazonaws.com
Port: 3306

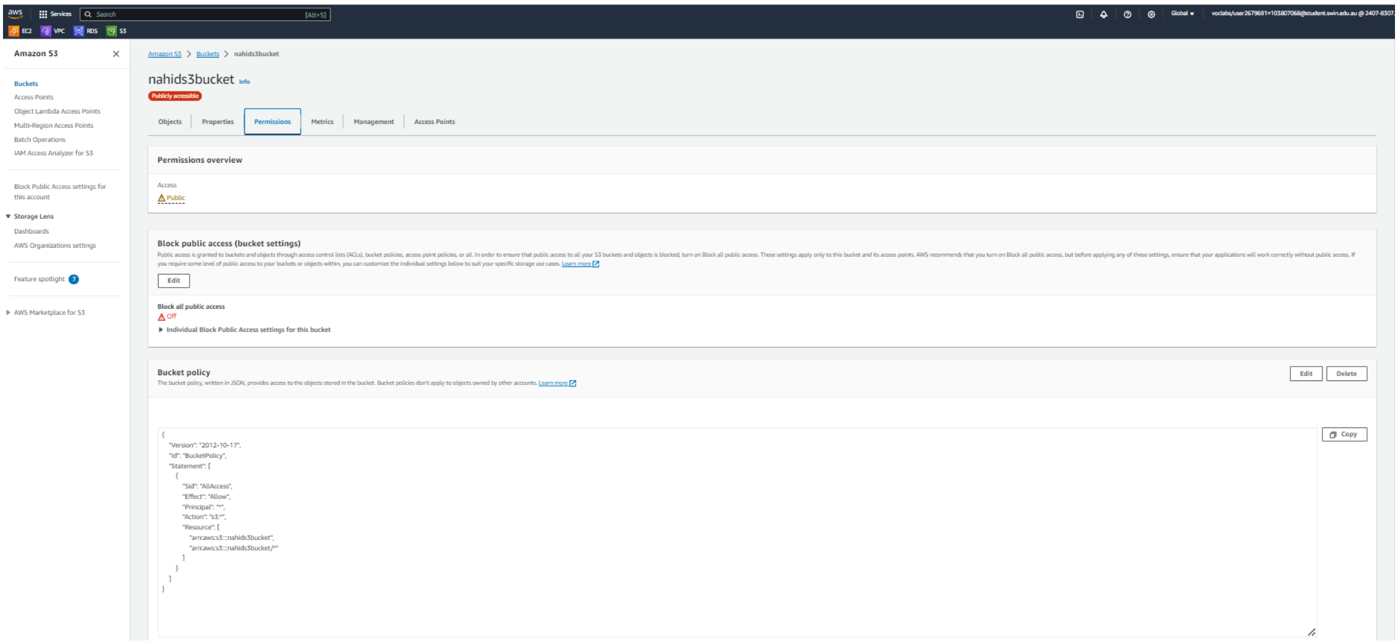
Networking
Availability Zone: us-east-1a
VPC: NTanjumVPC (vpc-0bb21b0509d0a4618)
Subnet group: rds sg
Subnets: subnet-008c2cf5702c78954, subnet-021b2811df74c6d34
Network type: IPv4

Security
VPC security groups: DBServerSG (sg-011d94706811c43d8) Active
Publicly accessible: No
Certificate authority: rds-ca-2019
Certificate authority date: August 23, 2024, 03:08 (UTC+10:00)
DB instance certificate expiration date: August 23, 2024, 03:08 (UTC+10:00)

Figure 11 - Database

E. S3

A S3 bucket storage has been set up to store the picture file with the necessary availability and no limits on public access.


aws
Services
Search
[Alt+S]
Global ▾
voclabs/user2679691=103807068@student.swin.edu.au @ 2407-8307
EC2
VPC
RDS
S3
Amazon S3
Buckets
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
Feature spotlight 7
AWS Marketplace for S3
Amazon S3 > Buckets > nahids3bucket
Info
Publicly accessible
Objects
Properties
Permissions
Metrics
Management
Access Points
Edit
Delete
Copy

nahids3bucket

Permissions overview

Access: ⚠ Public

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

Block *all* public access: ⚠ Off

▶ Individual Block Public Access settings for this bucket

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

```
{
  "Version": "2012-10-17",
  "Id": "BucketPolicy",
  "Statement": [
    {
      "Sid": "AllAccess",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::nahids3bucket",
        "arn:aws:s3:::nahids3bucket/*"
      ]
    }
  ]
}
```

Figure 12 – S3 Bucket Storage