Assignment-1BCreating and deploying Photo Album website onto a simple AWS infrastructure
Cos20019 |Cloud computing architecture

Name: Nur E Siam

Student ID: 103842784
Tutorial Time: Thursday 6.30PM
Date of Submission: 17-09-2023

Introduction

The Photo Album website leverages AWS's vast infrastructure, offering developers an array of resources from establishing a VPC for secure web application deployment to integrating an S3 bucket for photo storage. AWS's networking capabilities encompass diverse elements, such as implementing Network Access Control Lists (NACLs) and Security Groups, assimilating Amazon RDS, and orchestrating routing tables. To ensure encrypted communication within the VPC, there's an SSH linkage between instances in both public and private subnets.

This framework underlines the significance of security, scalability, and high availability, all paramount for modern cloud applications. The extensive nature of this project serves as an invaluable resource for AWS enthusiasts, shedding light on the intricacies of VPC design, strategic network division, and safeguarded data handling.

Keywords: VPC, NACL, Security Group, Routing Table, SSH, RDS.

For a hands-on understanding and visualization of this setup, the Photo Album website can be accessed via the provided link.

Website Link:

http://ec2-44-220-13-146.compute-1.amazonaws.com/cos20019/photoalbum/photoalbum/album.php

IMPLEMENTATION STEPS

Step-1: Create VPC

1. Create VPC select VPC only and allocate to 10.0.0.0/16.

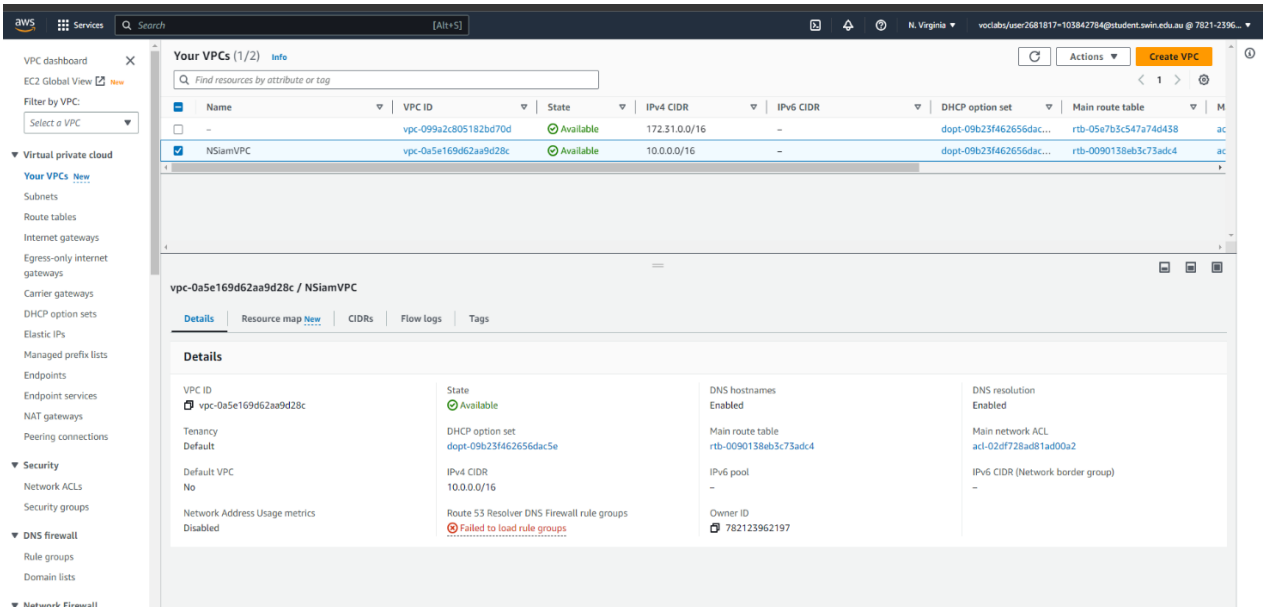


Figure 1-create VPC.

2. Setting VPC, enable DNS resolution and hostnames.

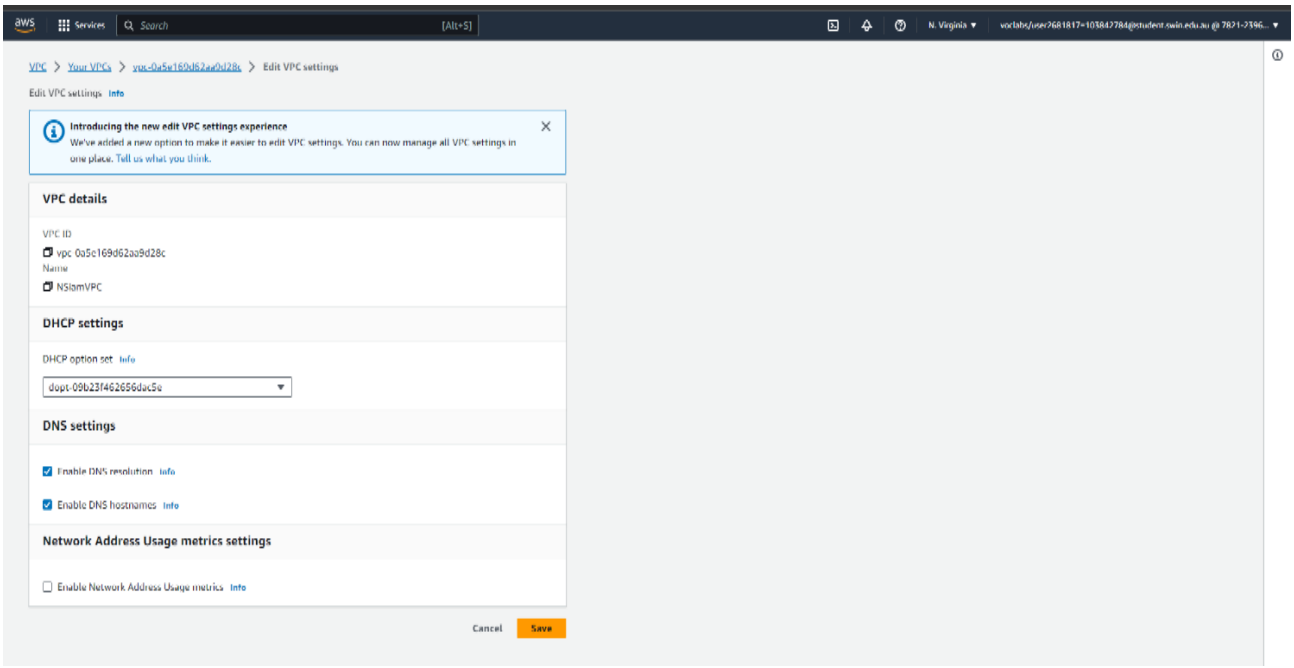


Figure 2-Adding VPC DNS

Step2: Create subnets.

Subnet	Ip	Availability Zone
Public Subnet 1	10.0.1.0/24	Us-east-1a
Private Subnet 1	10.0.3.0/24	Us-east- 1a
Public Subnet 2	10.0.2.0/24	Us-east-1b
Private Subnet 2	10.0.4.0/24	Us-east-1b

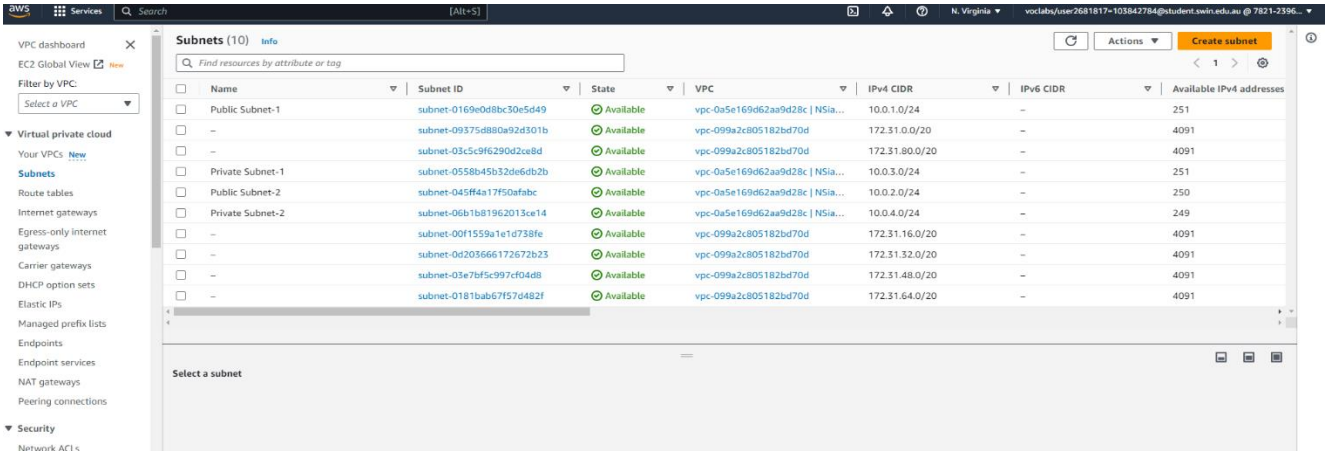


Figure 3-Public & Private Subnets

Step3: Create an internet gateway and attach it to the VPC.

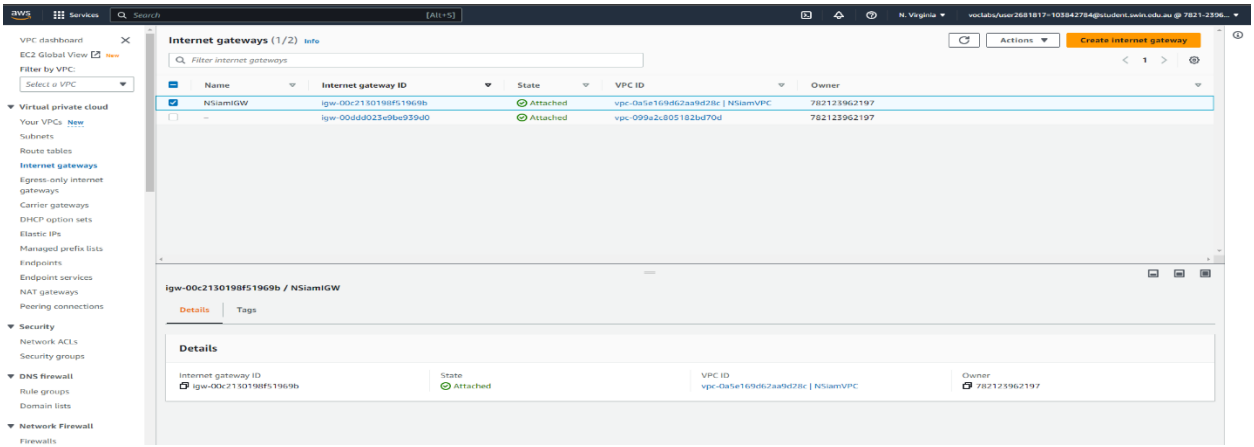


Figure 4-Internet Gateway attached to VPC.

Step4: Create 2 routing tables.

1. Public routing table

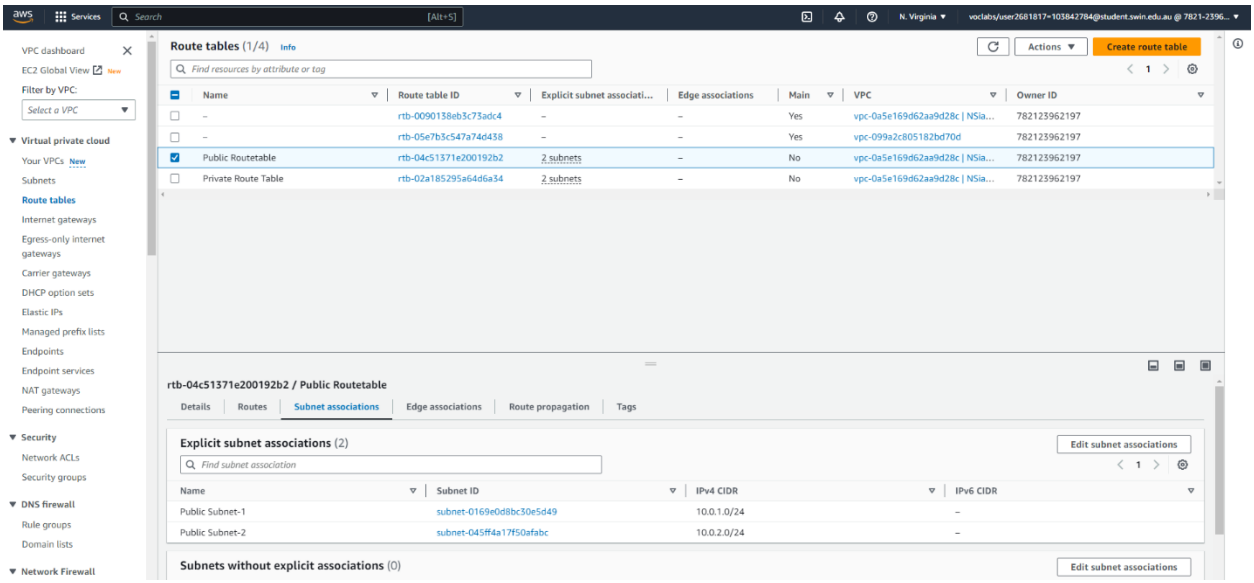
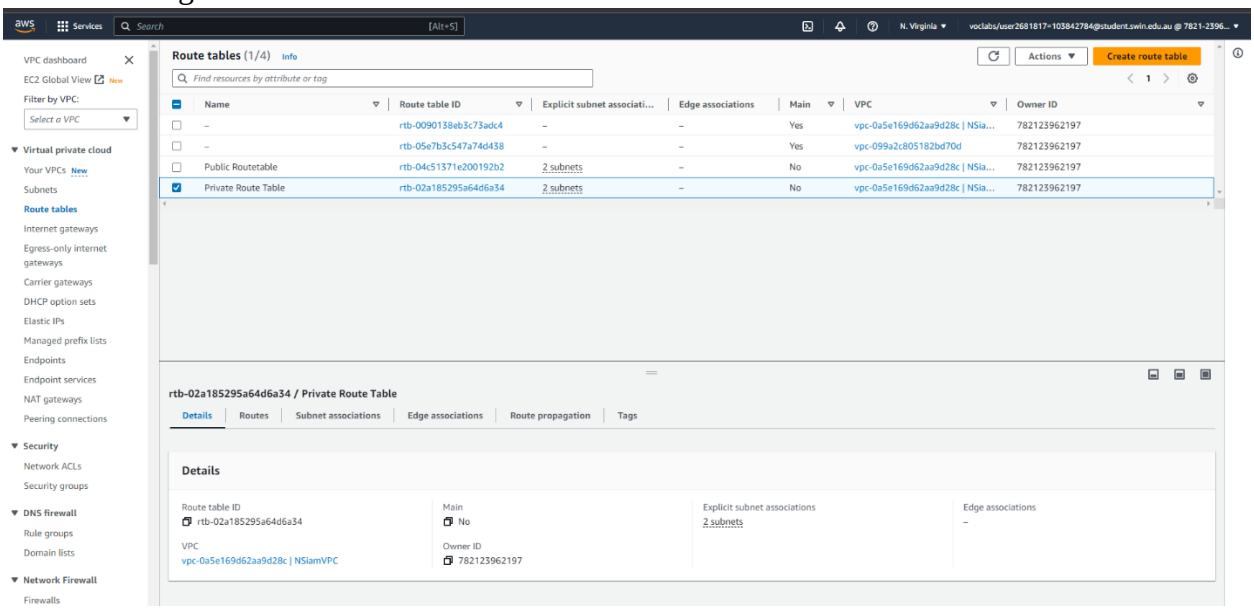


Figure 5-Public Routing table

2. Private routing table



3.

Figure 6-Private Routing table

3. Edit route of the Public route table to internet gateway and assign destination to 0.0.0.0/0

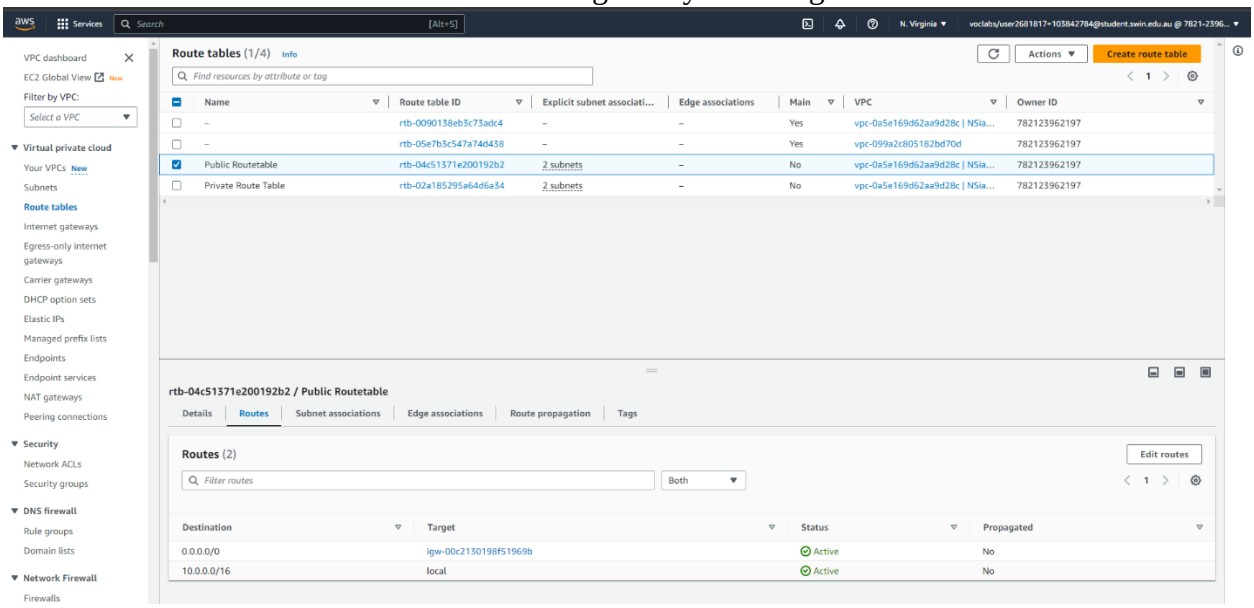


Figure 7-Edit routes on Public Route table.

Step5: Create 3 security groups.

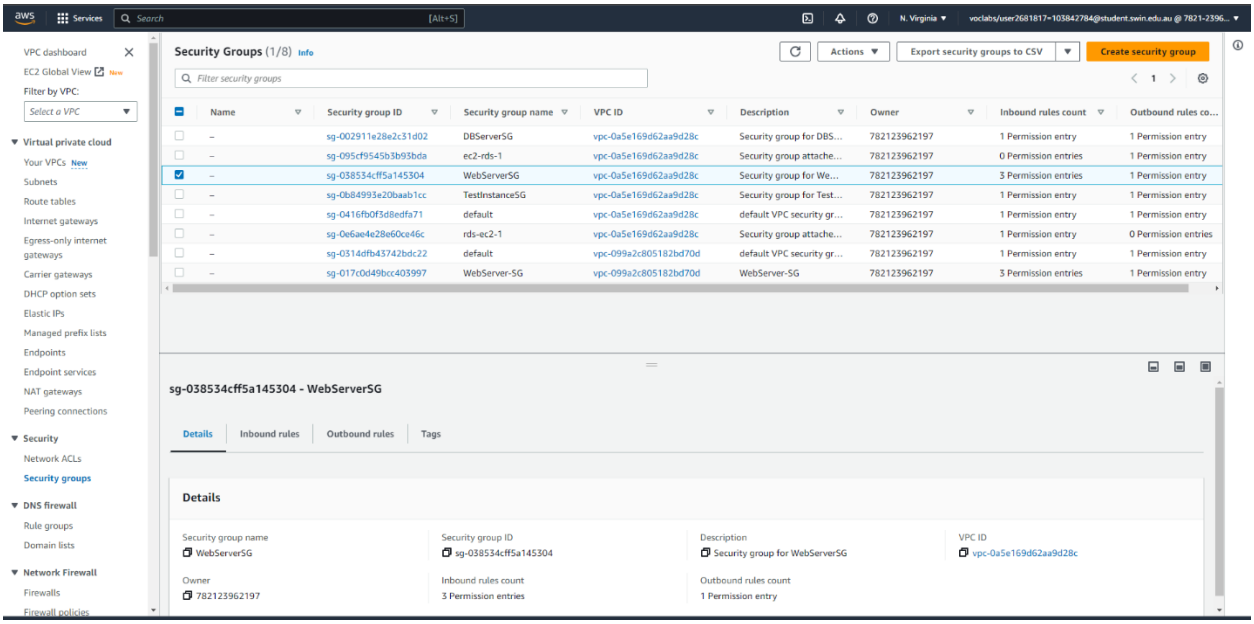


Figure 8-creating 3 security group.

Step 6: create resources for use when launching web server.

4. Public Elastic IP

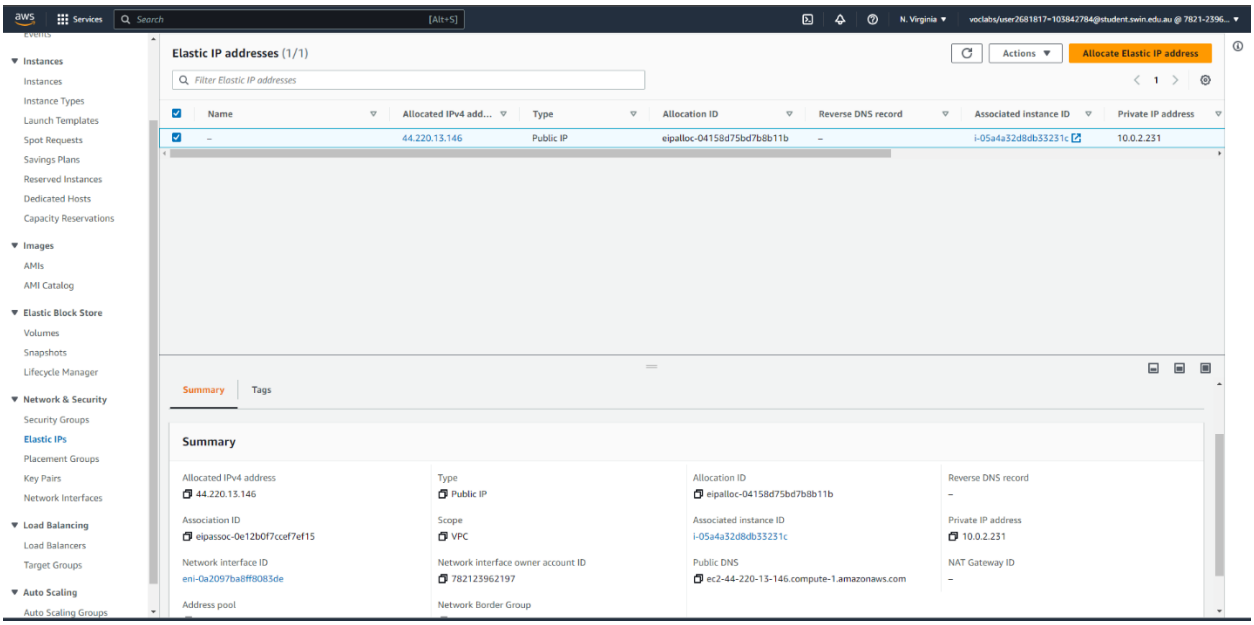


Figure 9-Elastic Ip

Step 7: launch the instances.

1. Create the Web Server instance in Public Subnet 2 and associate the Elastic IP.

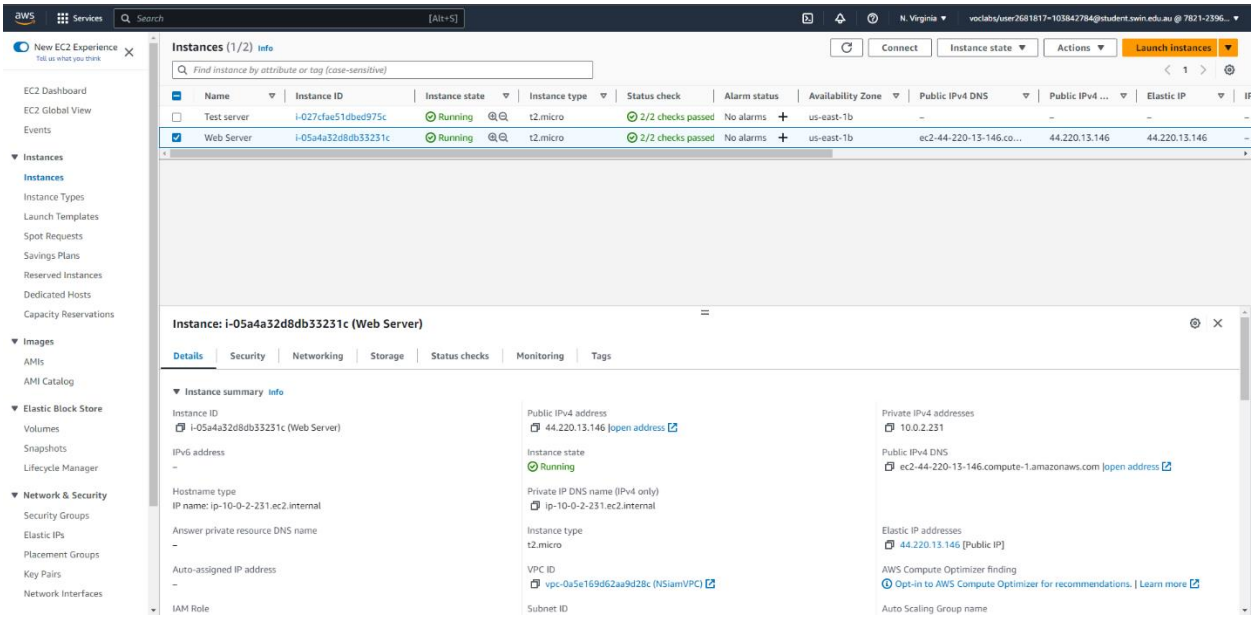


Figure 10-Web Server Instance & Elastic IP.

2. Create the Test server instance in Private Subnet 2

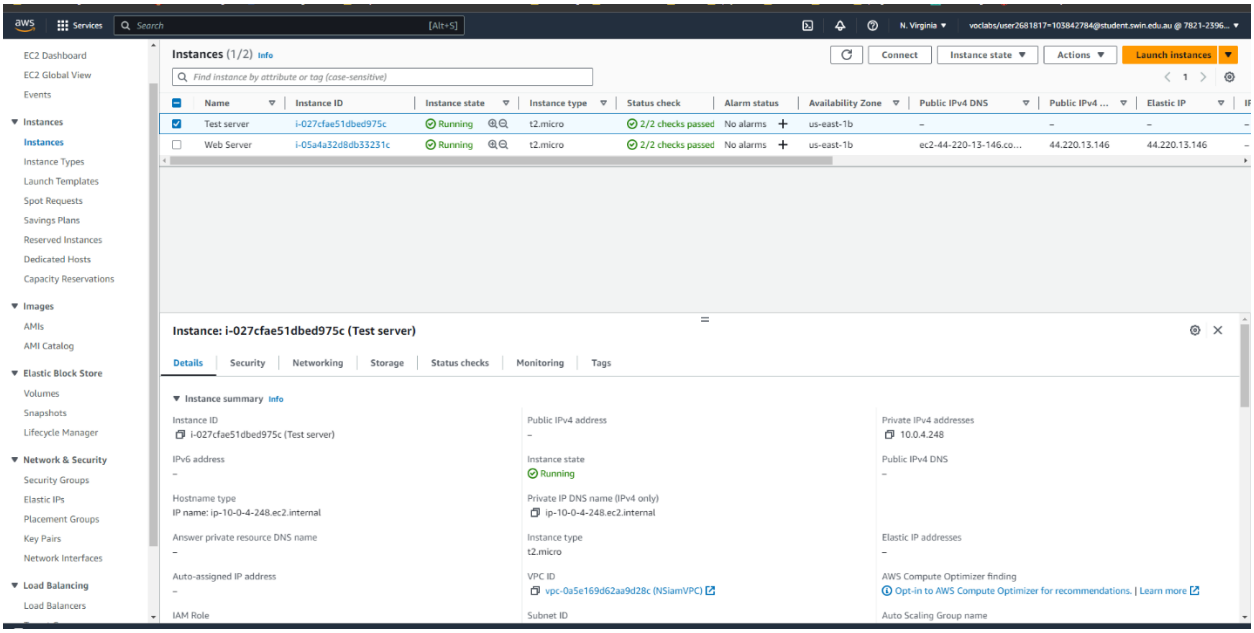


Figure 11-Test instance in Private subnet 2

3. Check that PHP is installed check that PHP is installed

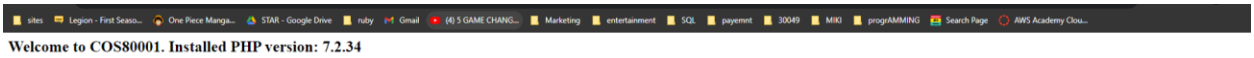


Figure 12- access phpinfo.php to test php functionality

Step 8: Login as ec2-user and SSH to Test Instance using its private IP address, and ping from Test instance to private IP address of Web server.

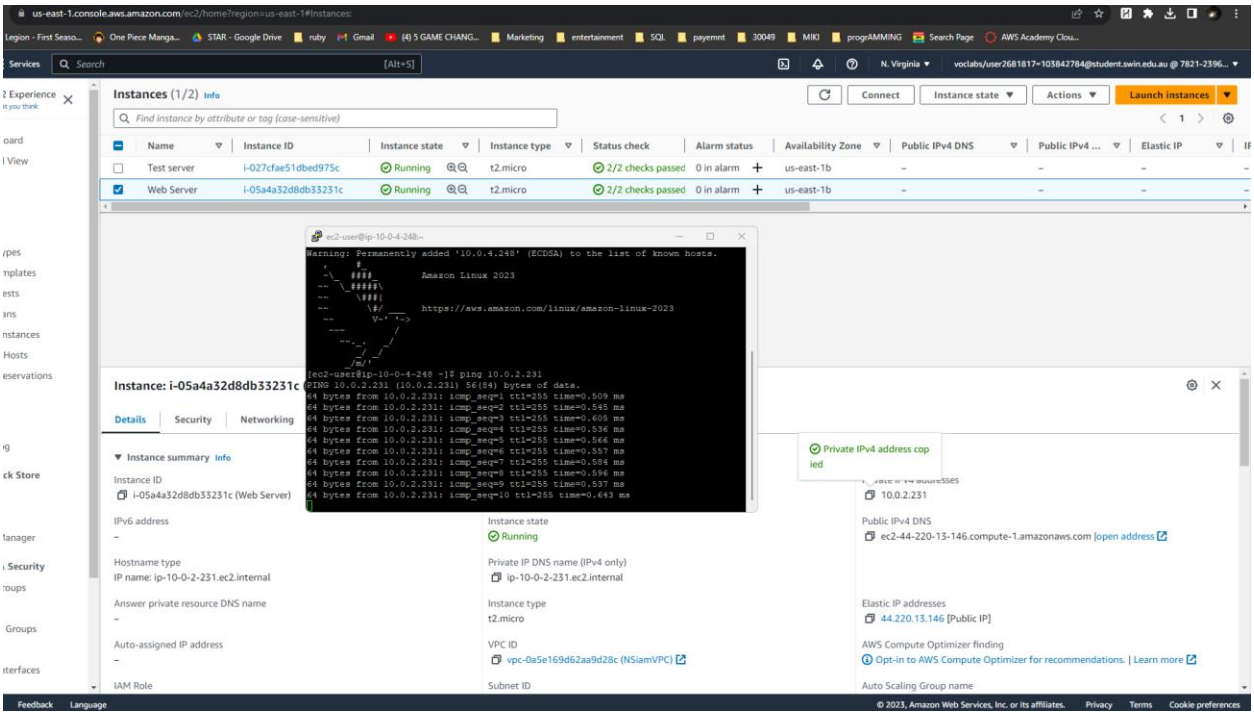


Figure 13- login as EC2 user and Ping

Step 9: Configuring RDS

- In RDS, create a subnet group and allocate two private subnet and create the RDS data base.
- The username = admin & password = admin1234
- RDS END Point = rds-instance.cfg1su8spy97.us-east-1.rds.amazonaws.com
- Install PhpMyAdmin on EC2 and change the file name from config.sample.inc.php to config.inc.php and then from inside the file change the localhost to the RDS end point.

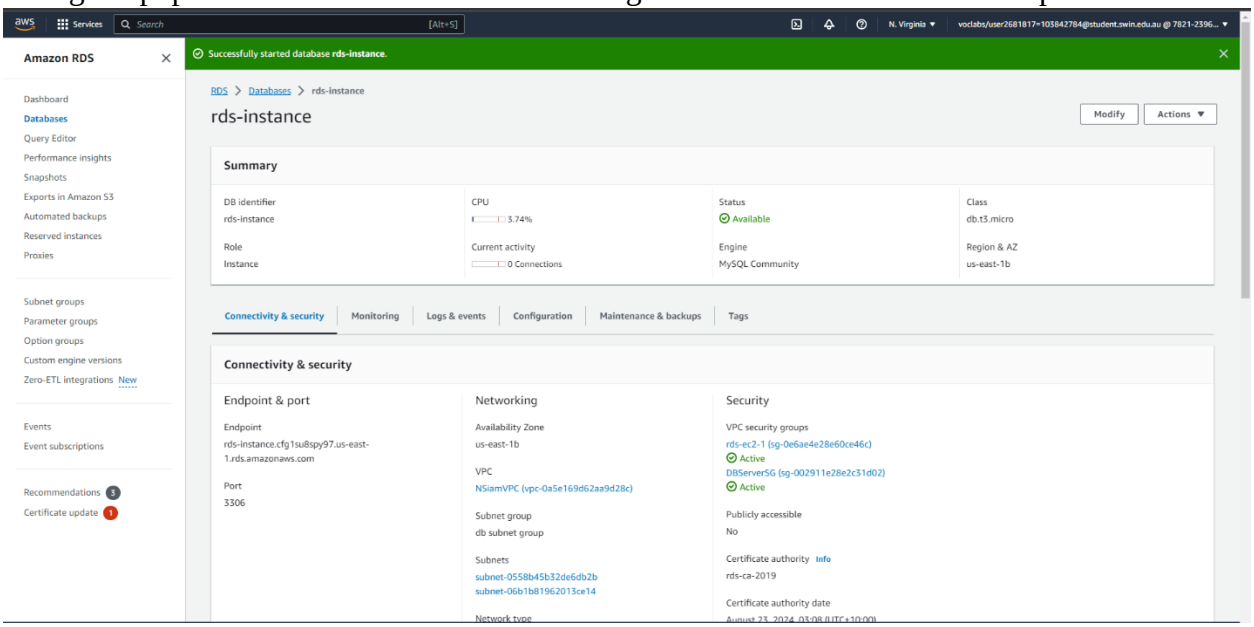


Figure 14- Database RDS

```
/*
 * Servers configuration
 */
$i = 0;

/**
 * First server
 */
$i++;
/* Authentication type */
$cfg['Servers'][$i]['auth_type'] = 'cookie';
/* Server parameters */
$cfg['Servers'][$i]['host'] = 'rds-instance.cfg1su8spy97.us-east-1.rds.amazonaws.com';
$cfg['Servers'][$i]['compress'] = false;
$cfg['Servers'][$i]['AllowNoPassword'] = false;

/**
 * phpMyAdmin configuration storage settings.
 */

/* User used to manipulate with storage */
// $cfg['Servers'][$i]['controlhost'] = '';
// $cfg['Servers'][$i]['controlport'] = '';
// $cfg['Servers'][$i]['controluser'] = 'pma';
// $cfg['Servers'][$i]['controlpass'] = 'pmapass';

/* Storage database and tables */
// $cfg['Servers'][$i]['pmadb'] = 'phpmyadmin';
// $cfg['Servers'][$i]['bookmarktable'] = 'pma__bookmark';
```

Figure 15-replacing the local host to RDS end point

Step 10: Go to http://ec2-44-220-13-146.compute-1.amazonaws.com/phpmyadmin.

- Create a table named “photos”. Add 5 columns to the table.

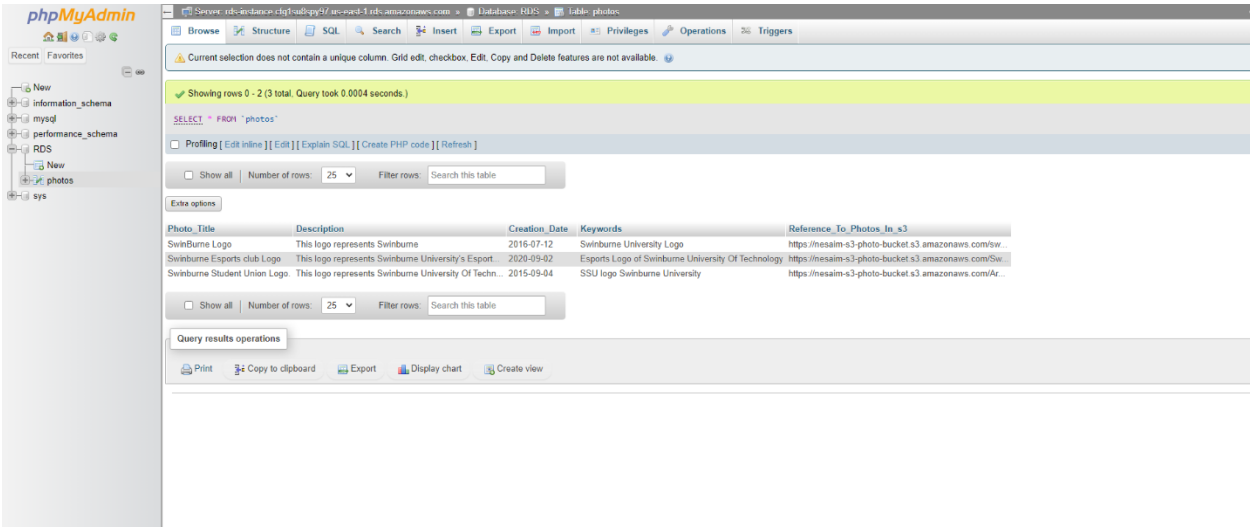


Figure 16-Table creation

Step 11: Creating Network ACL

- In Network ACL Inbound rules and Outbound rules should be the same and attach it to public subnet 2.

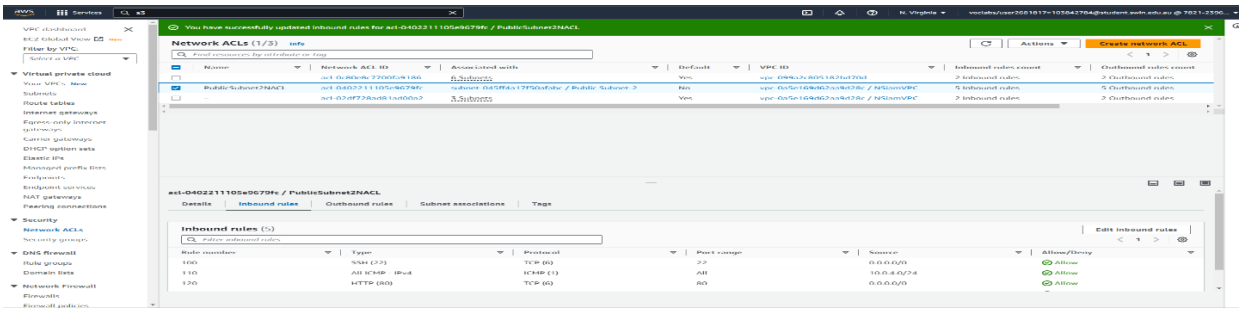


Figure 17-Network ACL with Inbound rules

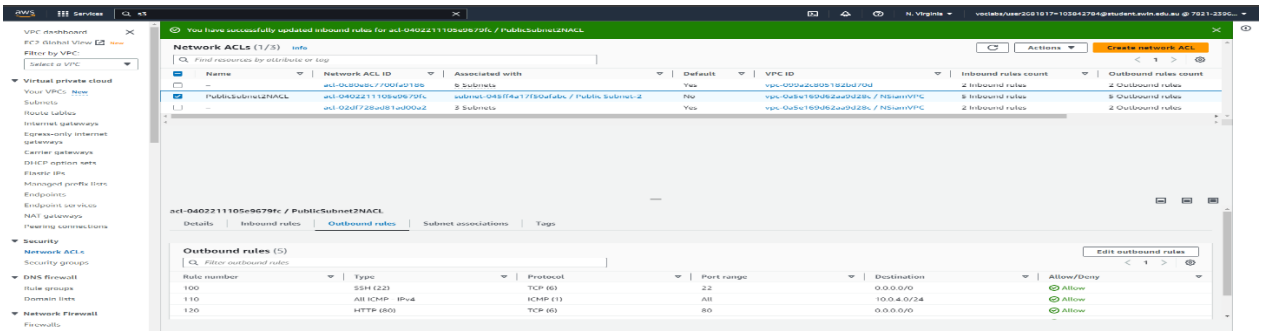


Figure 18-Newtwork ACL with Outbound rules

Step 12: Create S3 Bucket

- Specify Public access and configure the bucket policy.
- Upload the images and copy the URL

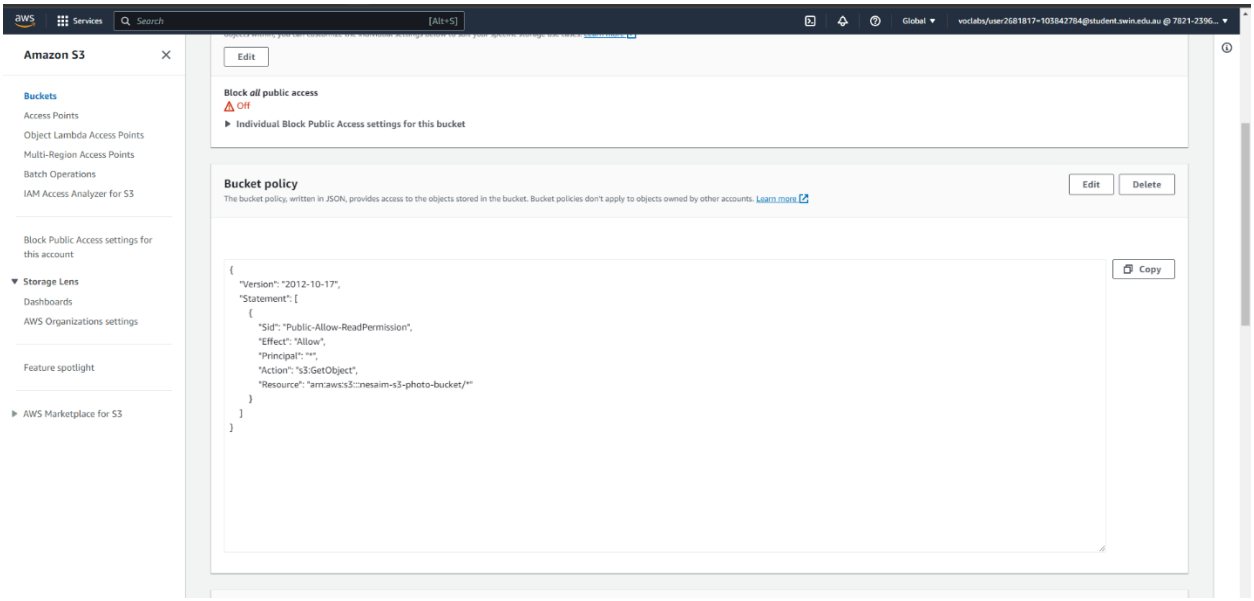


Figure 19-S3 bucket policy and public access enable.

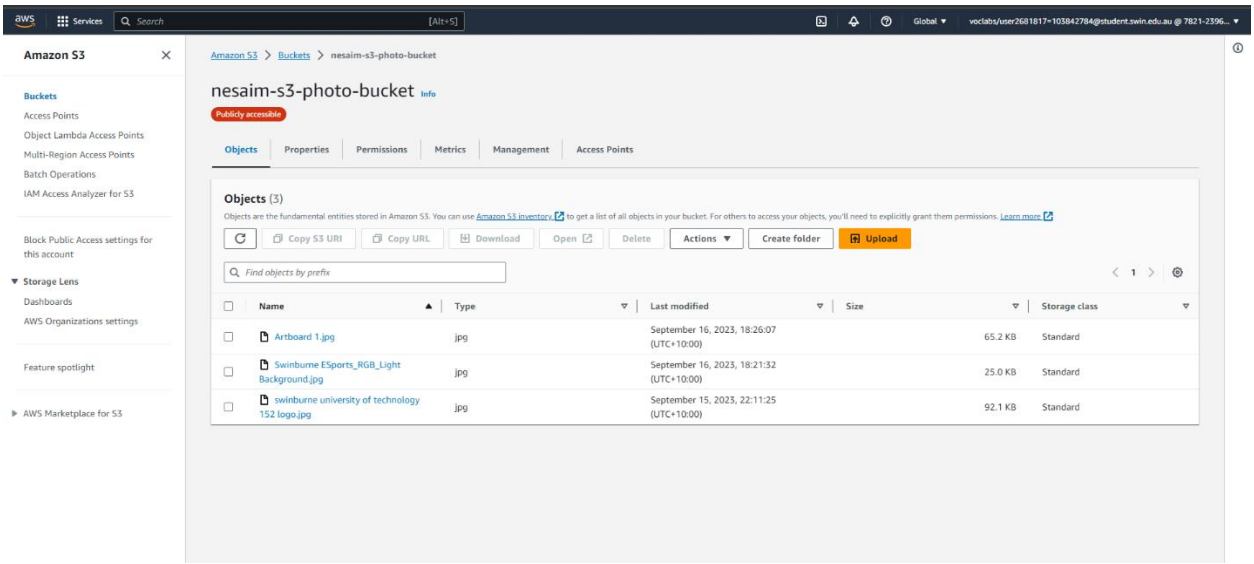


Figure 20-upload the photos

Step 13: In PhpMyAdmin, insert new roles with photos' URLs in photos table.

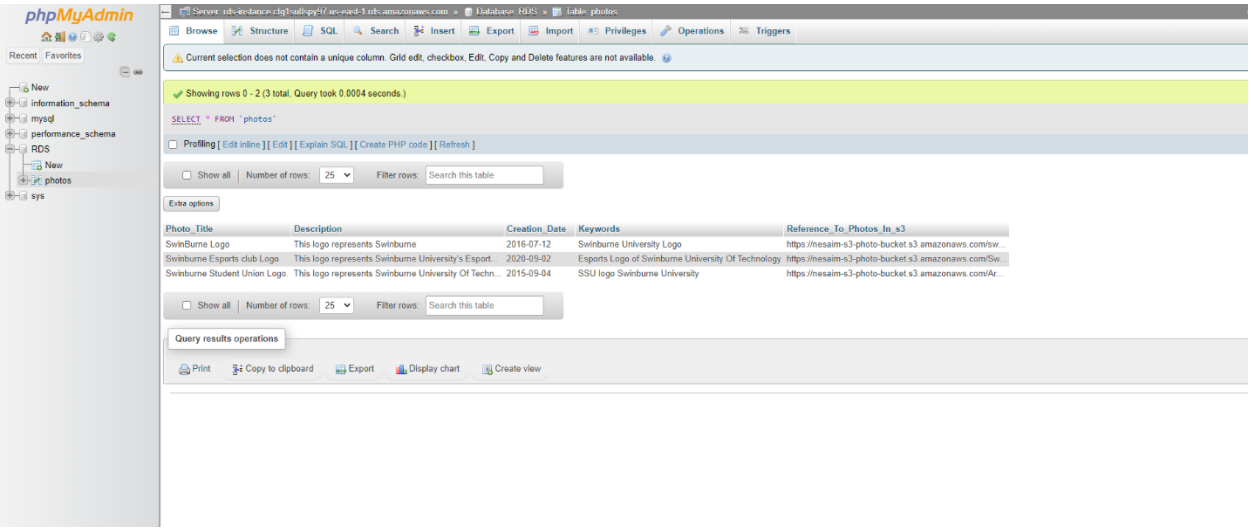


Figure 21-Table with the picture links

Step 14: In photoalbum_v3, enter the correct code based on the resources in AWS.

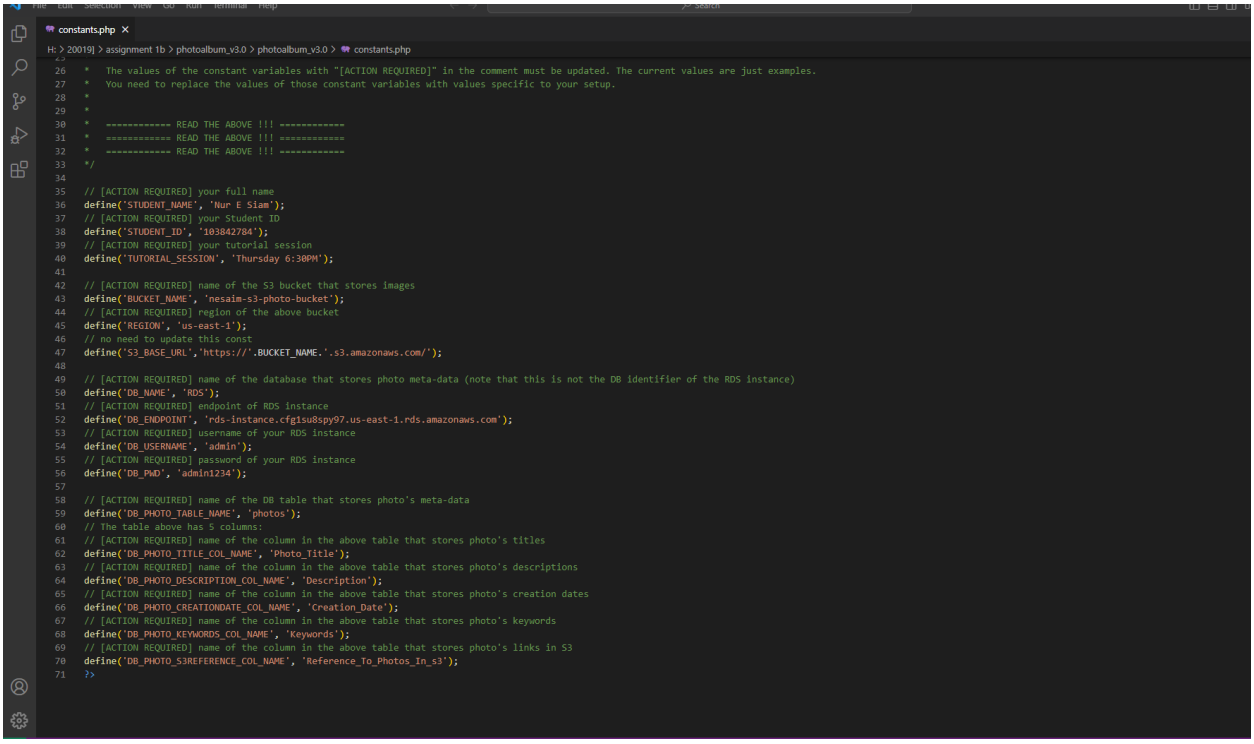


Figure 22-Fixing the codes in constants.php

Step 15: In WINSCP, create a path of directory where photoalbum is hosting the website.

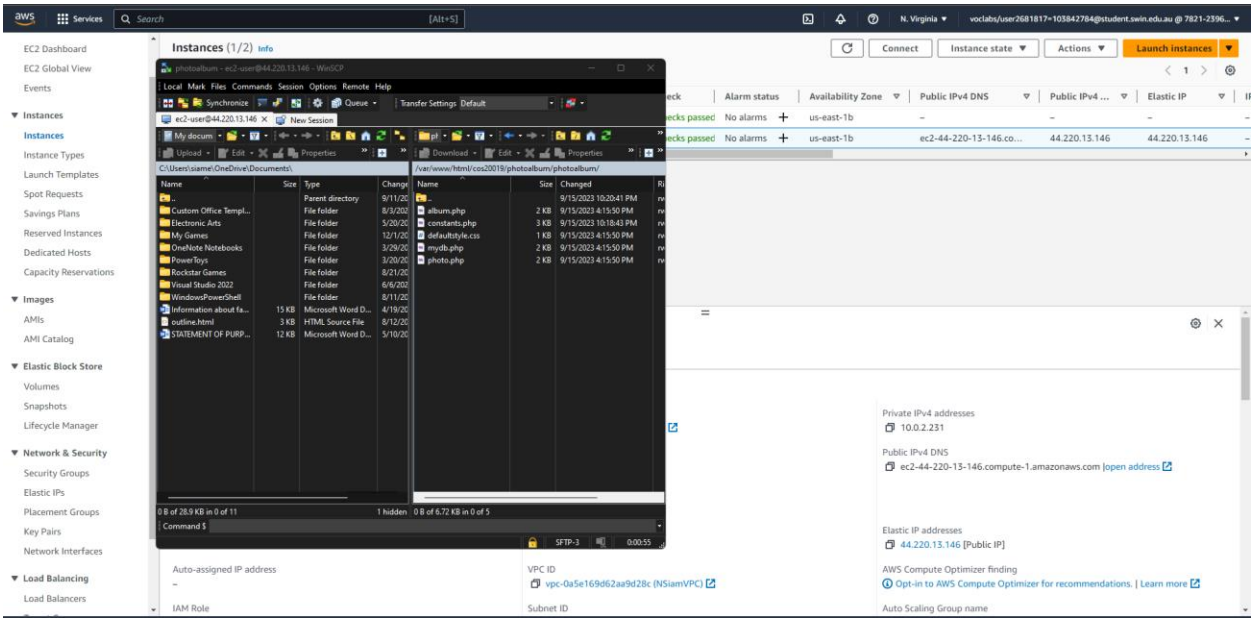


Figure 23-WINSCP uploading the photoalbum file with corrct constants.php code

Step 16: Access my newly created website which is its database in cloud.

Website link:

http://ec2-44-220-13-146.compute-1.amazonaws.com/cos20019/photoalbum/photoalbum/album.php

Student name: Nur E Siam

Student ID: 103842784

Tutorial session: Thursday 6:30PM

Uploaded photos:

Photo	Name	Description	Creation date	Keywords
	SwinBurne Logo	This logo represents Swinburne	2016-07-12	Swinburne University Logo

Figure 24-Final product the Website

Step 17: Testing

- Add more photos.
- And do a ping to web server again to test Network ACL.

Student name: Nur E Siam

Student ID: 103842784

Tutorial session: Thursday 6:30PM

Uploaded photos:

Photo	Name	Description	Creation date	Keywords
	SwinBurne Logo	This logo represents Swinburne	2016-07-12	Swinburne University Logo
	Swinburne Esports club Logo	This logo represents Swinburne University's Esports club	2020-09-02	Esports Logo of Swinburne University Of Technology
	Swinburne Student Union Logo.	This logo represents Swinburne University Of Technology's Student Union.	2015-09-04	SSU logo Swinburne University

Figure 25- Final website with more photos.

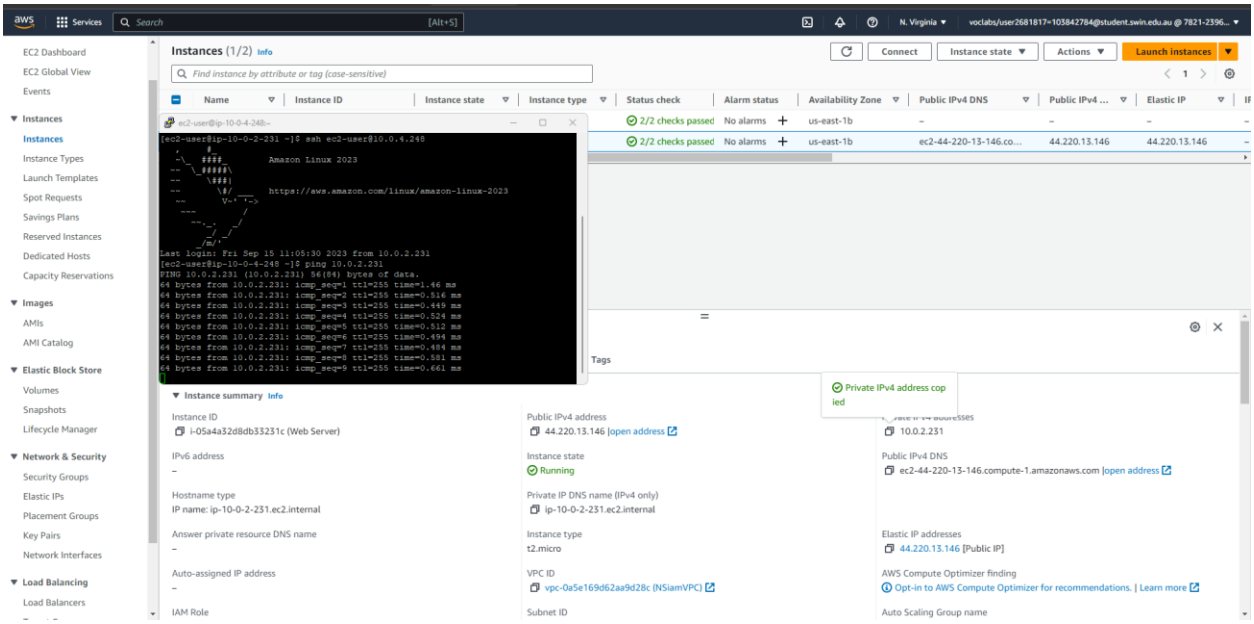


Figure 26- PIng after Network ACL is done with more photos being uploaded.