Assignment 1 – Part B Creating and deploying Photo Album website onto a simple AWS infrastructure

Cos20019 | Cloud computing architecture

Student name:	Phuthai Hemathulintra
Student ID:	103804205
Lab Section:	Wednesday 6:30 PM
Date of report:	14/09/2023

INTRODUCTION

The project of Photo Album website is implemented by AWS infrastructure. The AWS cloud service variedly provides developer service and existing resource to create from setting up VPC to enable the safe deployment of a web application to linking S3 bucket which store the photo objects. AWS networking has many different aspects implementing Network Access Control List (NACLs) and Security Groups, integrating Amazon RDS, and setting up routing tables. In order to show secure communication inside the VPC, SSH connectivity is also established between instances situated in public and private subnets.

The need to uphold security, scalability, and high availability standards—all of which are crucial in the deployment of contemporary cloud-based applications—is emphasized throughout this scenario. The project's comprehensive nature makes it an exemplary reference for AWS practitioners, providing insights into VPC architecture, network segmentation, and secure data management.Keywords: VPC, VPC, NACL, Security Group, Routing Table, SSH, RDS. By following the following implementation steps, this project can be pictured in a vivid demonstration. The following link is the path to photo album website. Website link: http://ec2-44-218-82-104.compute-1.amazonaws.com/cos20019/photoalbum/album.php

IMPLEMENTATION STEPS

Step1: Create VPC

- Create VPC select VPC only and allocate to 10.0.0.0/16

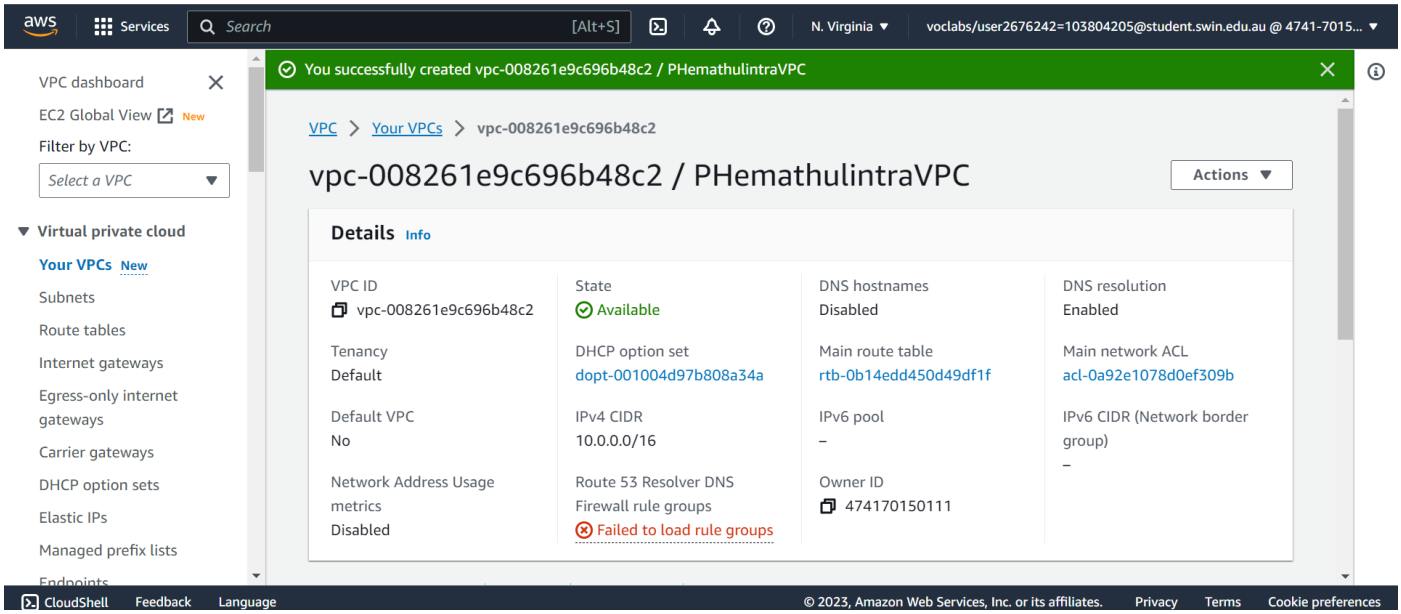


Figure 1 – create VPC

- Setting VPC, enable DNS resolution and hostnames

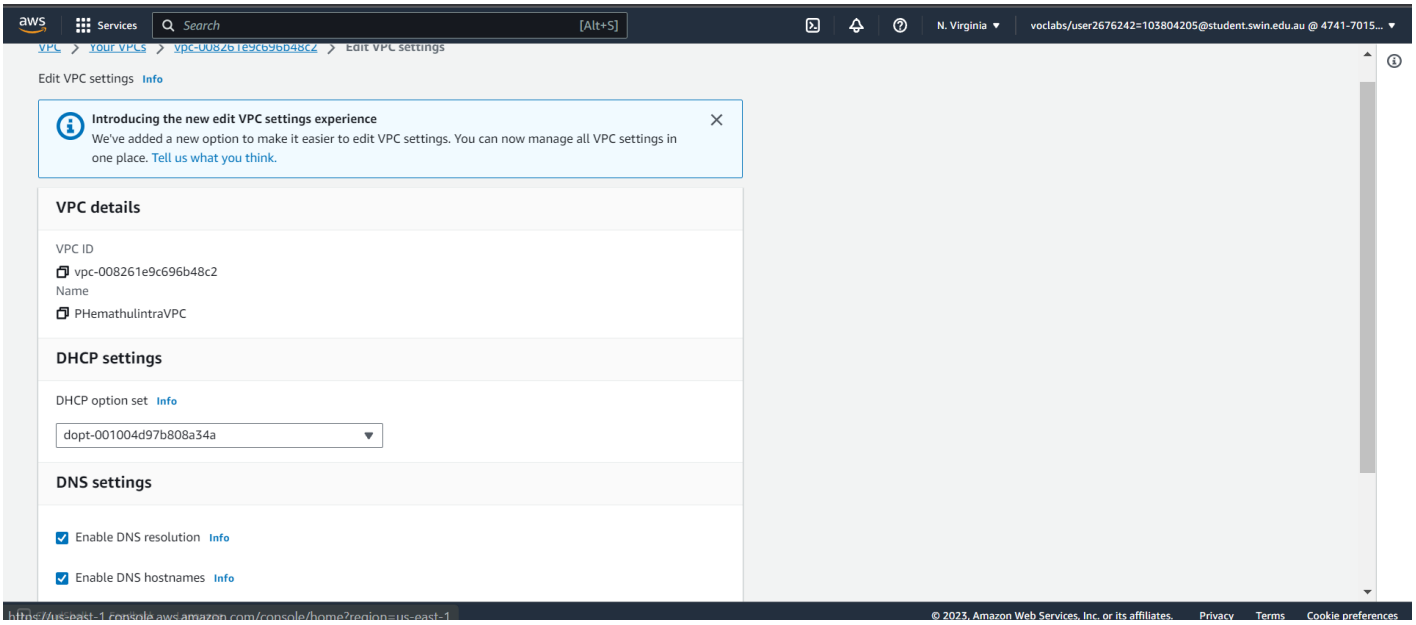


Figure 2 – edit VPC DNS

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

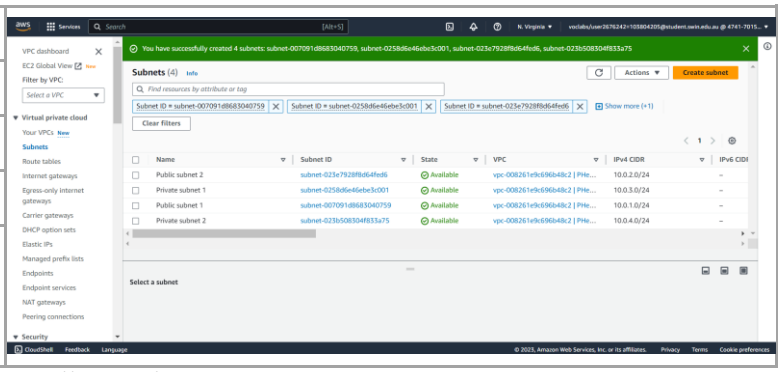


Figure 3 – public-private subnets

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

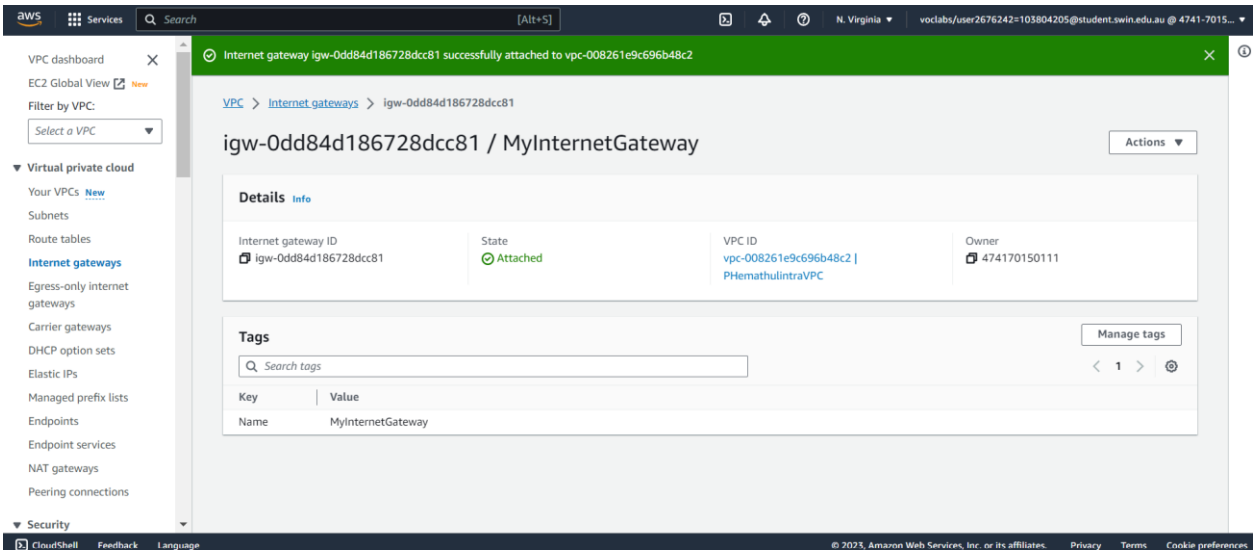


Figure 4– create internet gateway and attach to VPC

Step4: Create 2 routing tables.

1) Public routing table

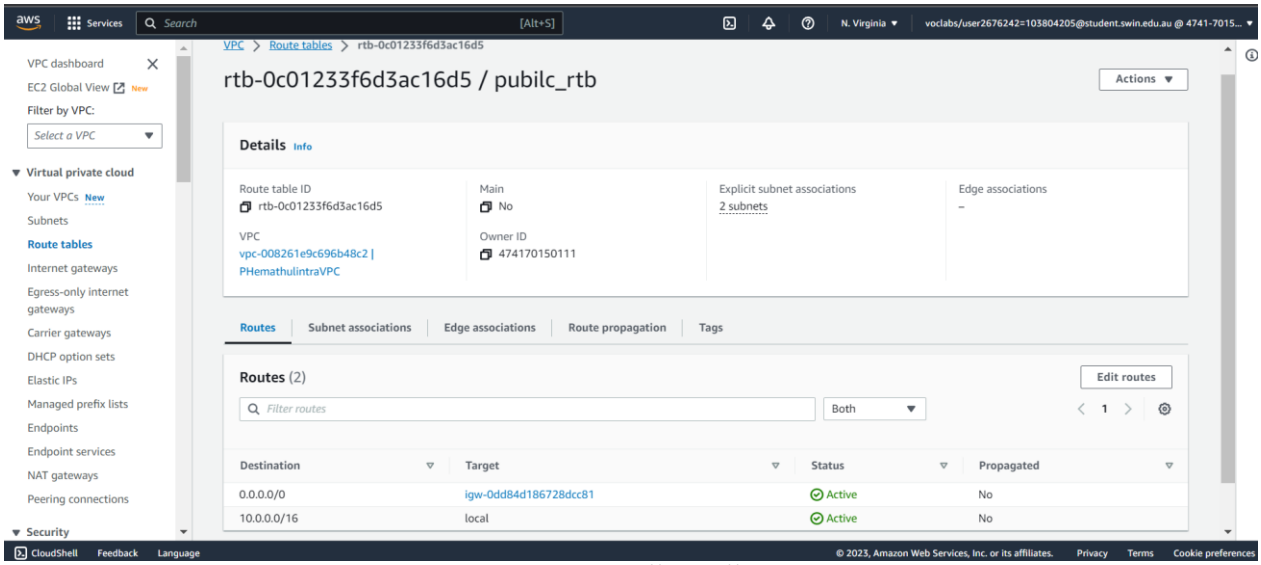


Figure 5 – create public route table

2) Private route table

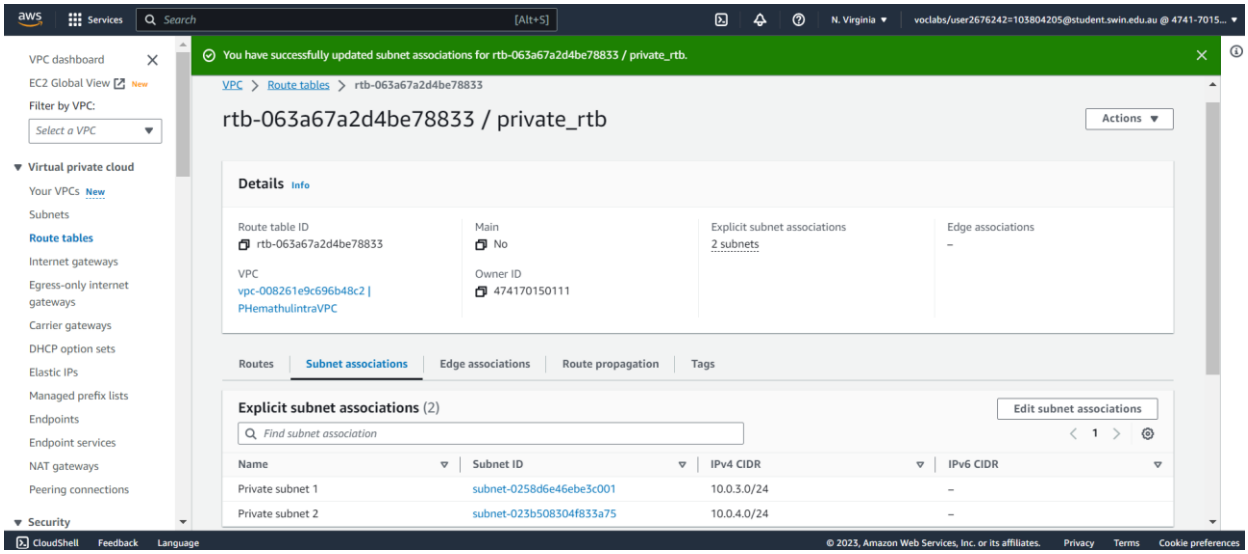


Figure 6 – create private route table

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

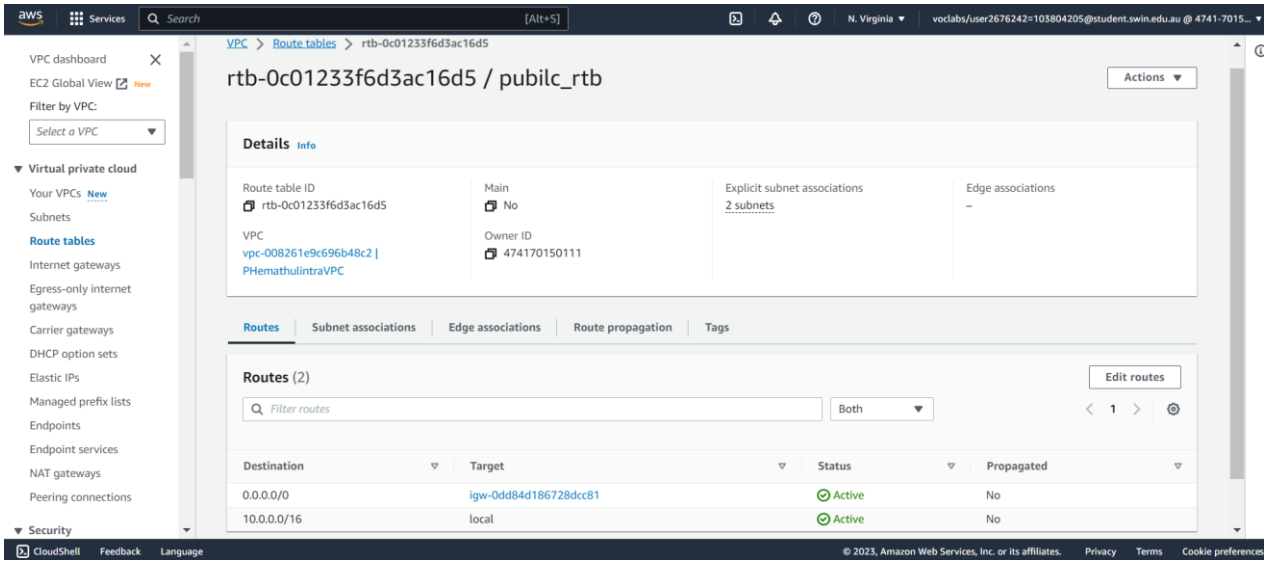


Figure 7 – edit route on public route table

- Associated appropriate subnets to the route tables

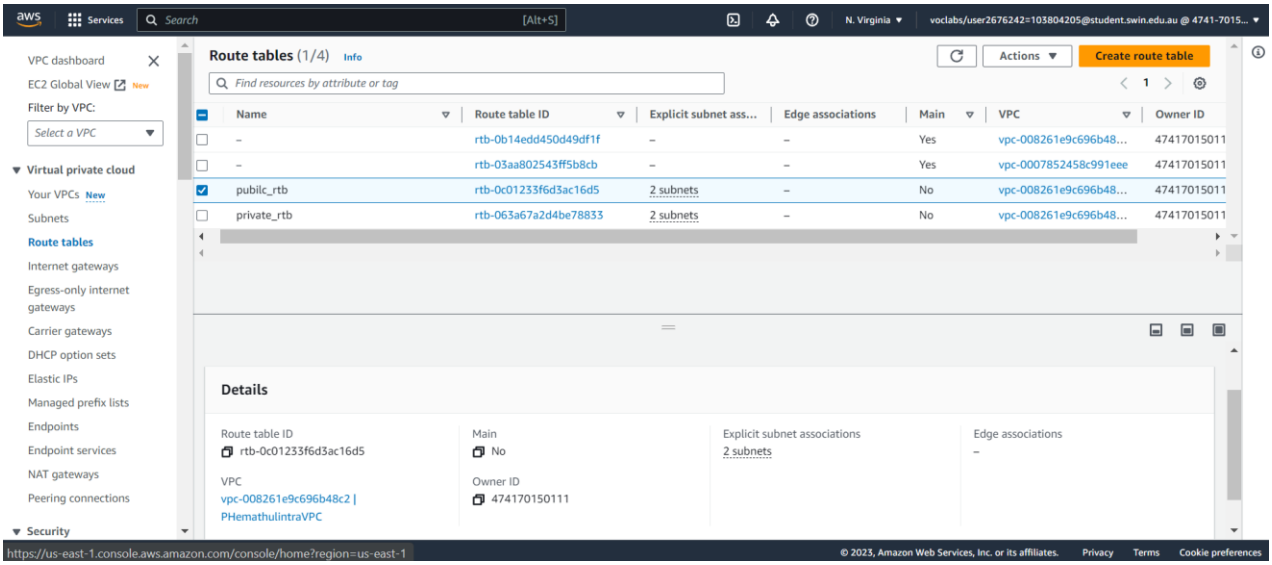


Figure 8 – two route tables

- Check the VPC resource map

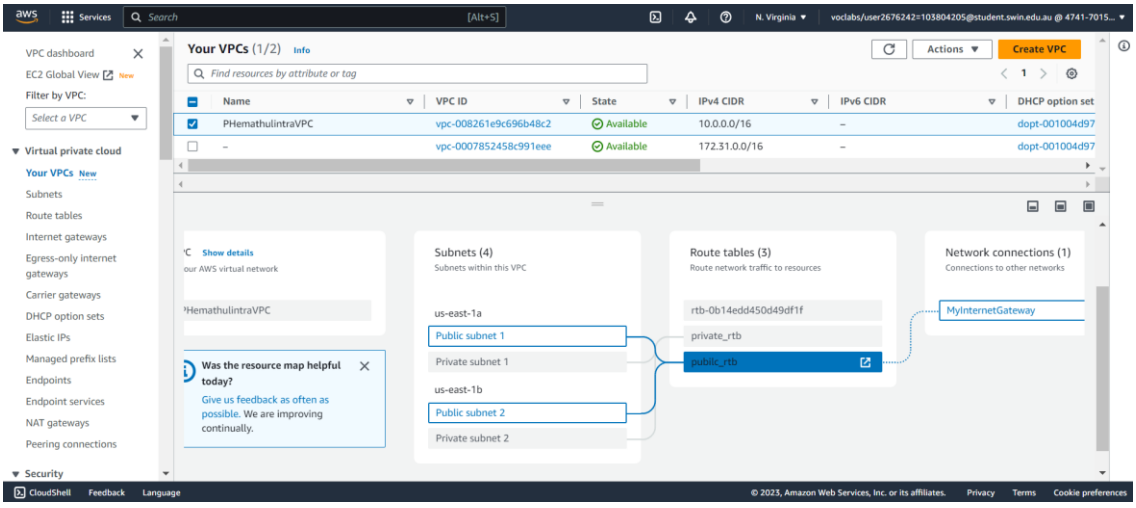


Figure 9 – view VPC resource map to ensure all subnet configurations

Step5: Create 3 security groups

1) TestInstanceSG

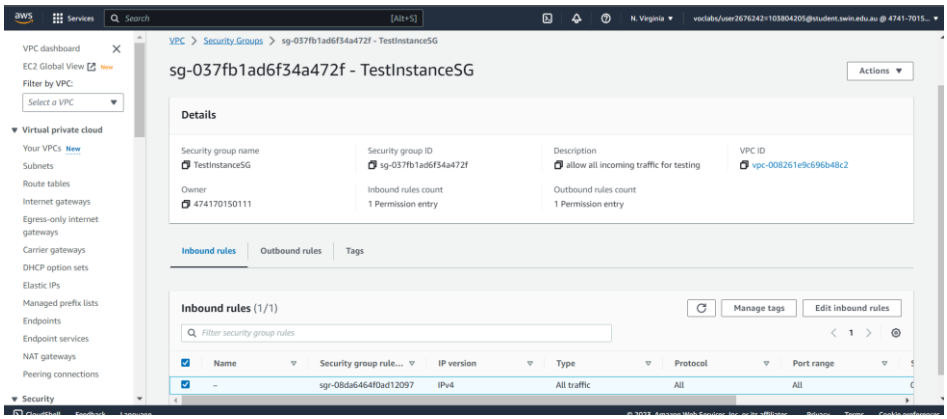


Figure 10 – create security group1

2) WebServerSG

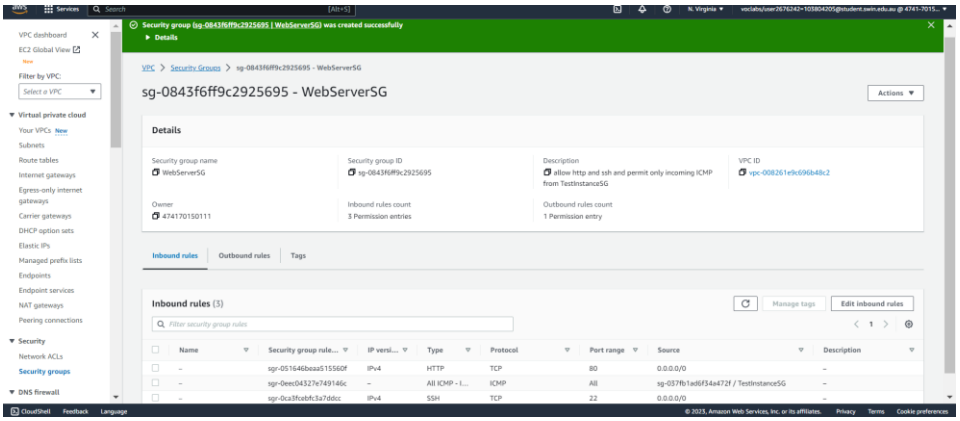


Figure 11– create security group2

3) DBServerSG

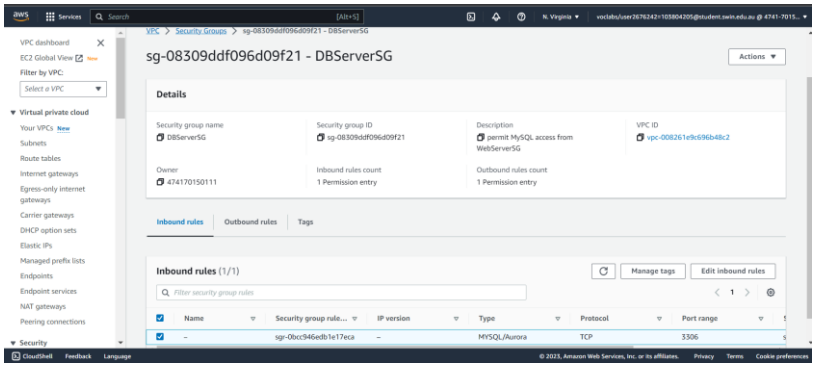


Figure 12 – create security group3

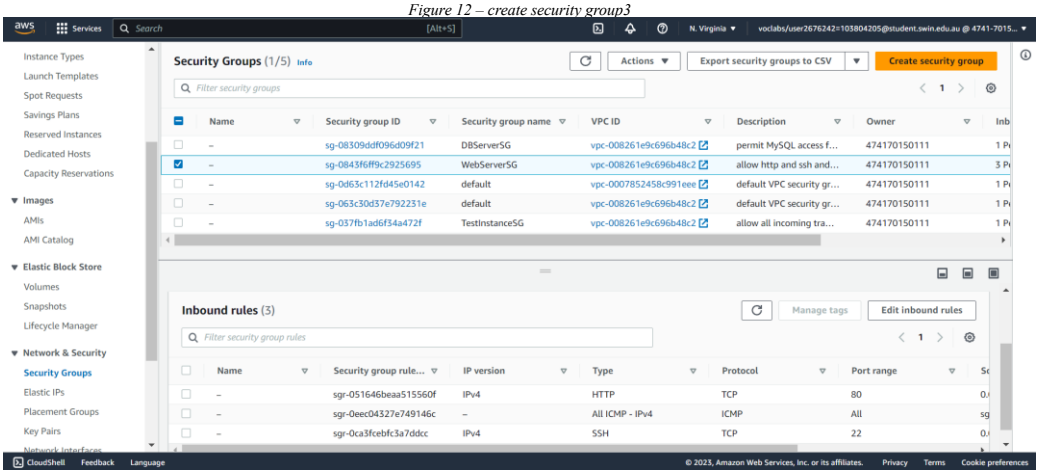


Figure 13 – three security groups

Step 6: create resource for use when launching web server

1) a key pair (.pem file)

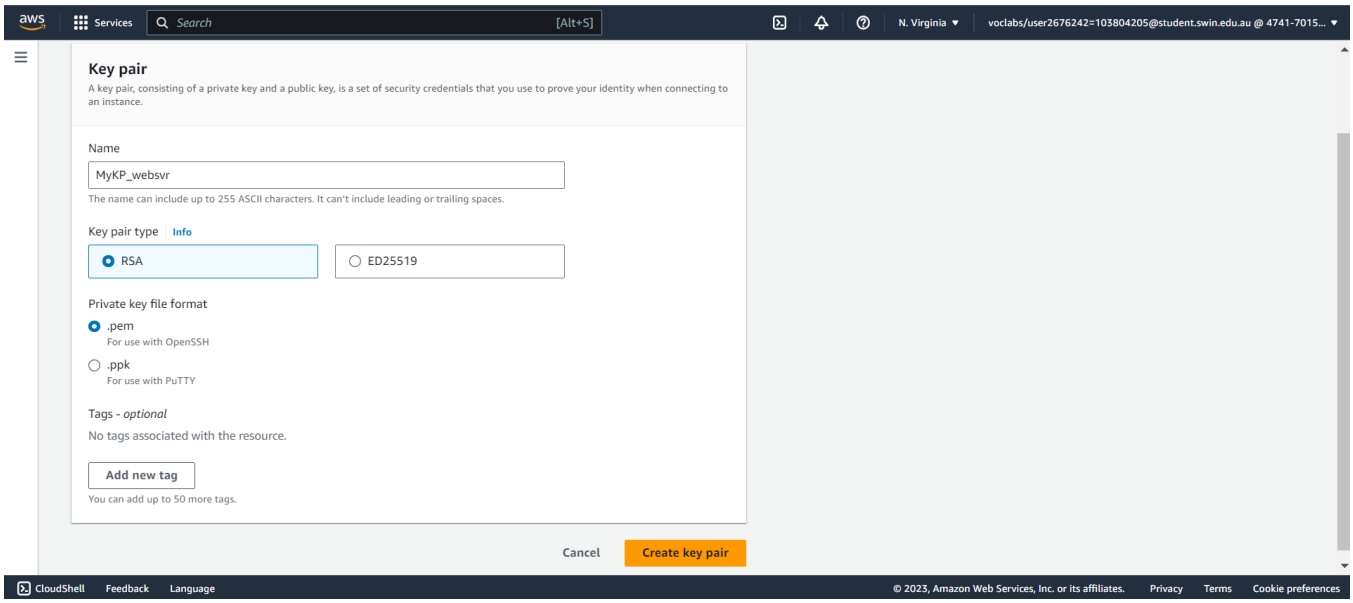


Figure 14 – create a key pair for web server

2) public Elastic IP

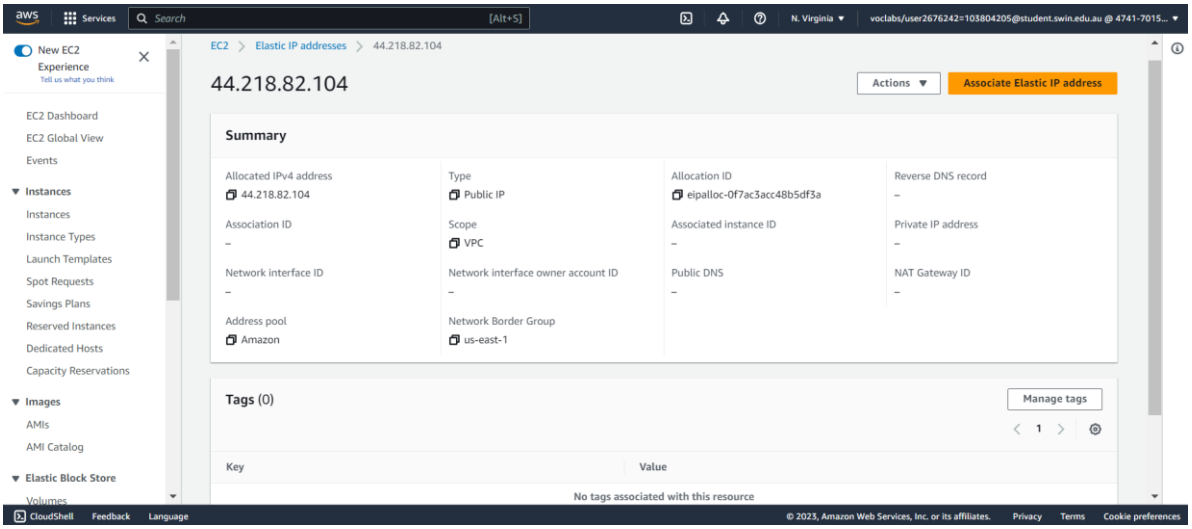


Figure 15 – create static IP

Step 7: launch the instances

1) Bastion/Web server

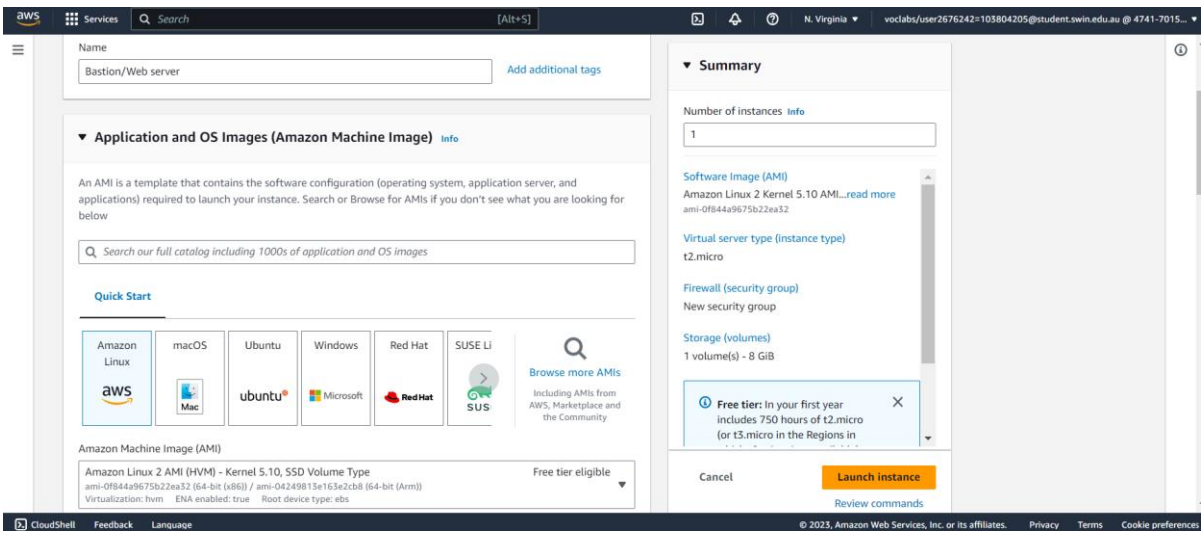


Figure 14– launch web server: select image

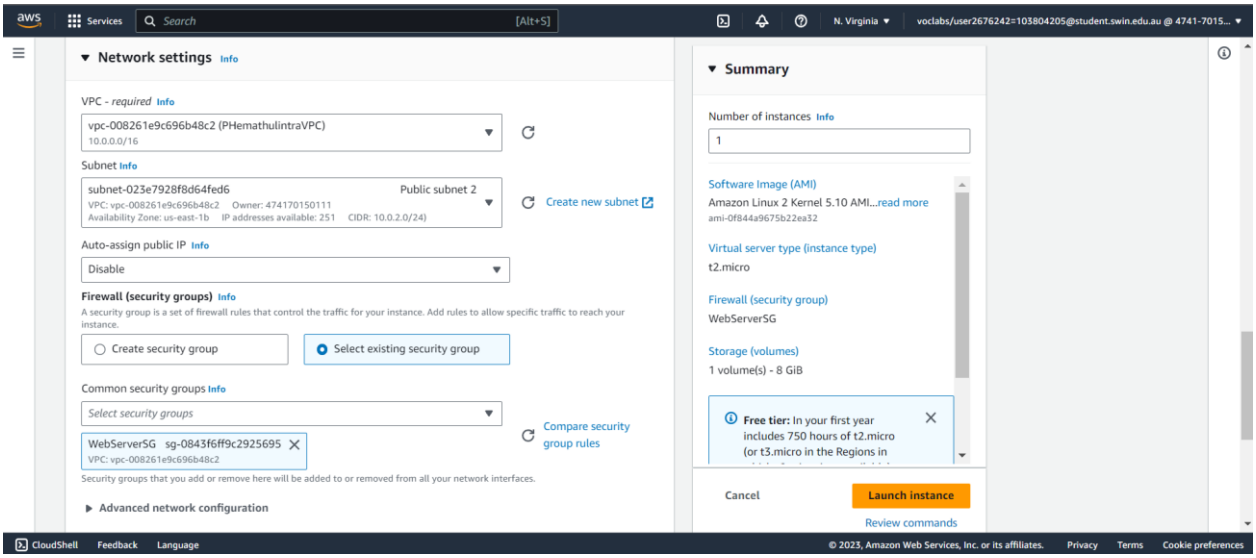


Figure 15 – launch web server: edit network settings

- attach Elastic IP to Bastion/Web instance

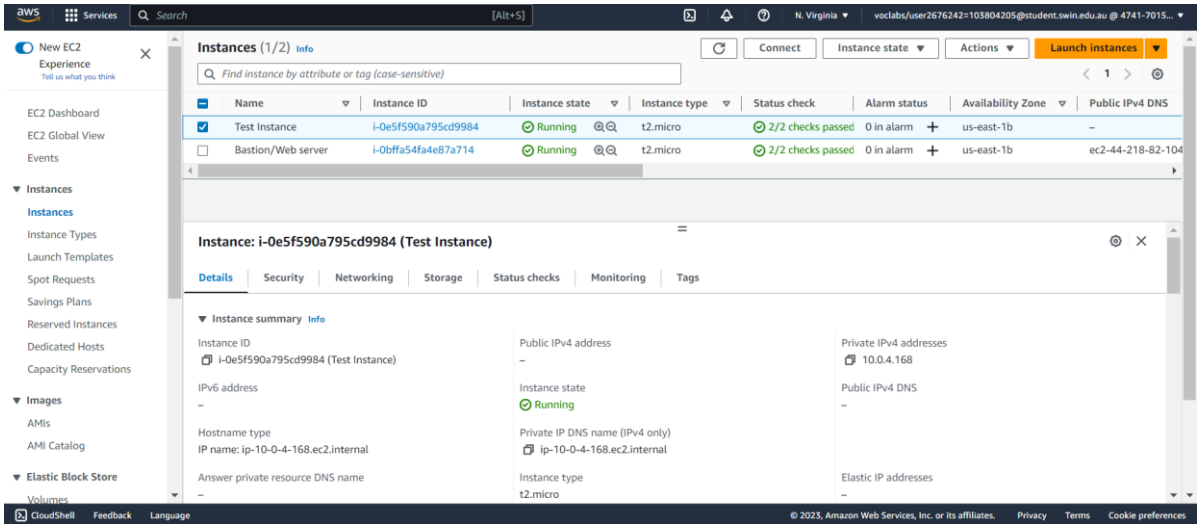


Figure 16 – attach static IP to web server

- check that PHP is installed

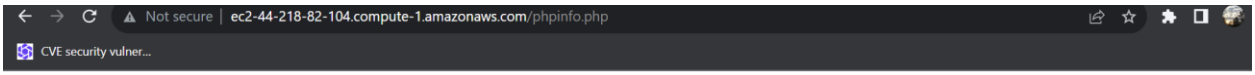


Figure 17 – access phpinfo to test php functionality

2) Test Instance

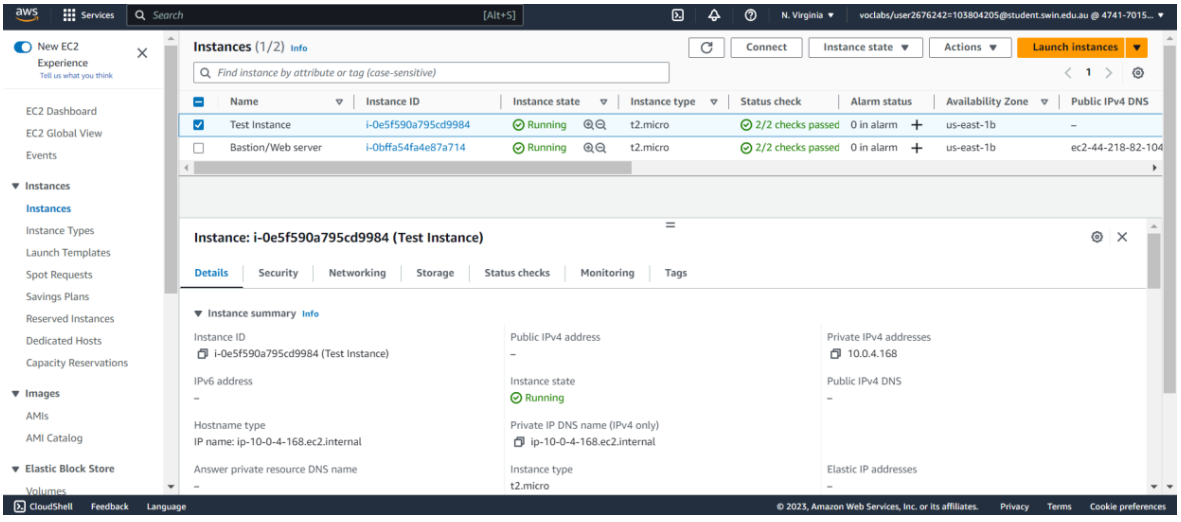


Figure 18 – test instance is in private subnet 2

Step 8: initiate SSH via variant of putty components

1) Open puttygen. Upload pem file. Create passphrase. Select SSH-1(RSA) and save private key.

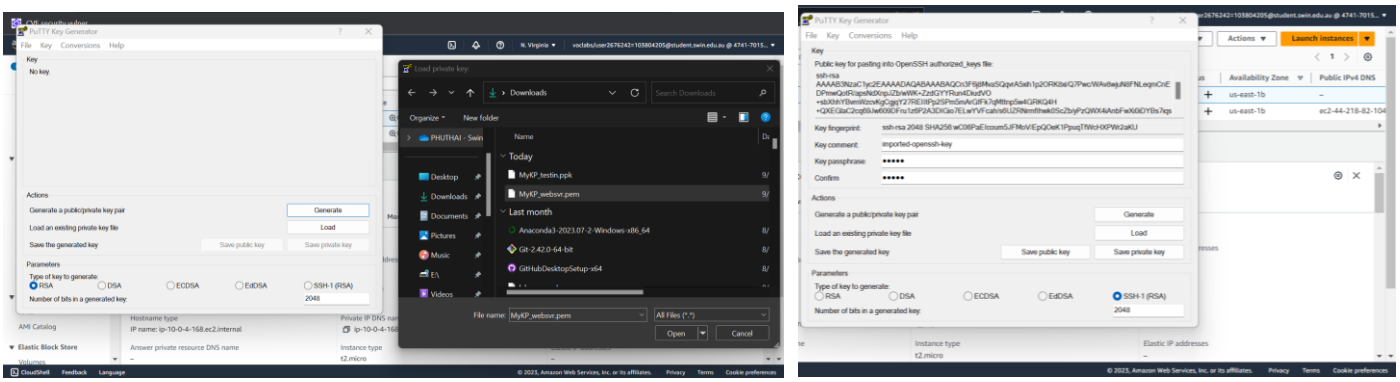


Figure 19-20 – puttygen key convertion

2) Open pageant and add key. Use the one that is converted to ppk. Enter passphrase and close pegeant.

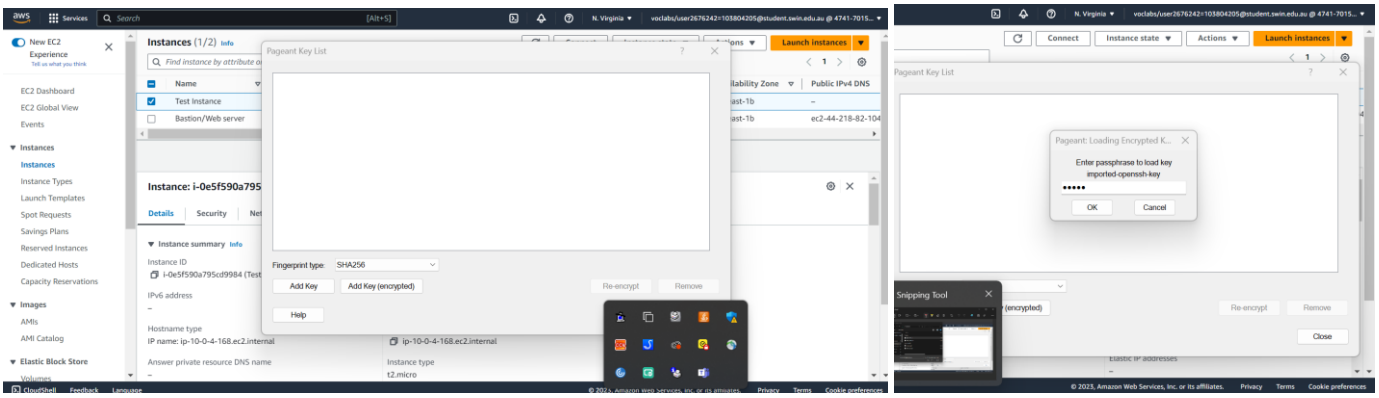


Figure 21-22 – pegeant configurations

3) Open putty. In host field, enter public IPv4 of Bastion. Select Allow agent forwarding. Click open.

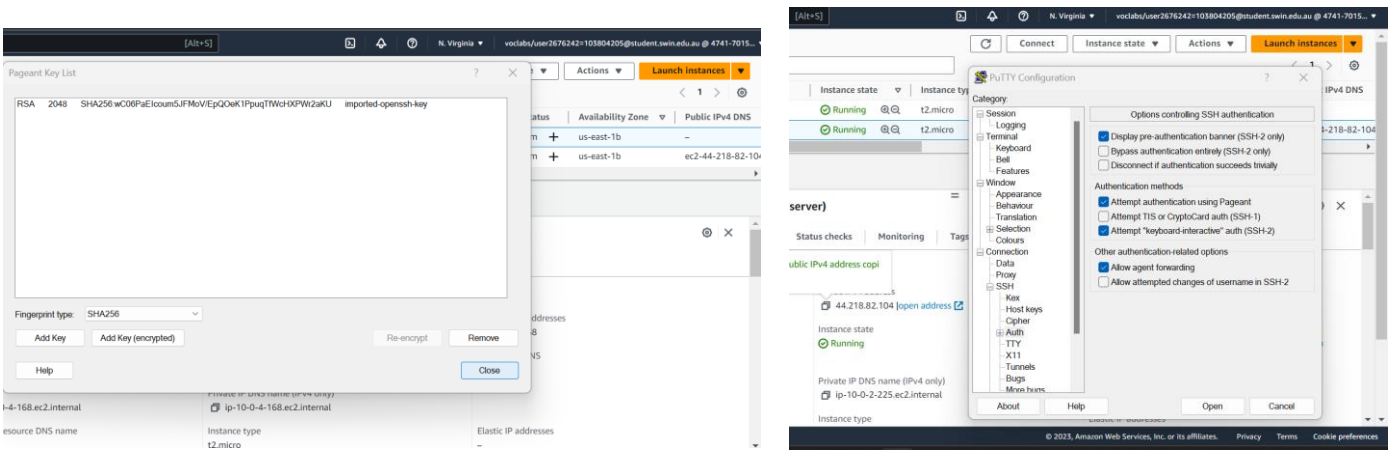


Figure 23-24 – putty configuration

4) Login as ec2-user and SSH to Test Instance using its private IP address

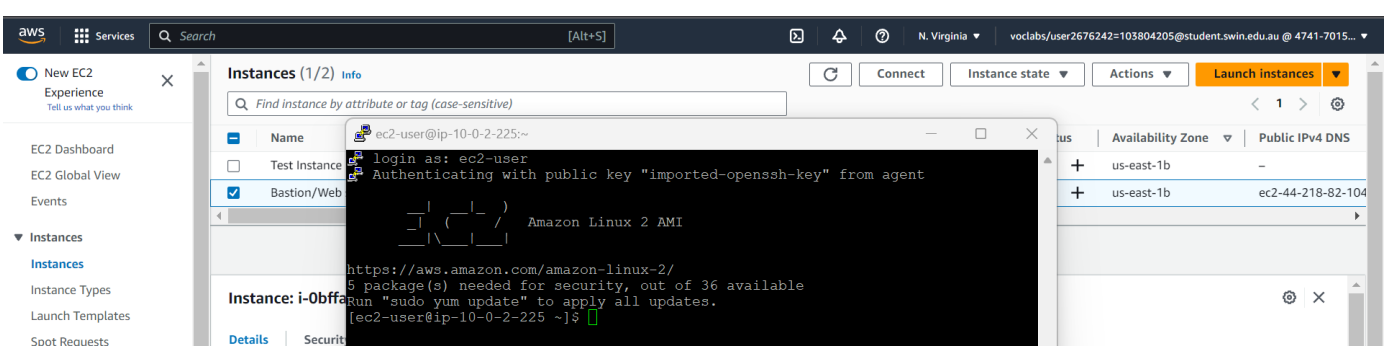


Figure 25 – connect to ec2 web server and login

5) Ping from Test instance to private IP address of Bastion.

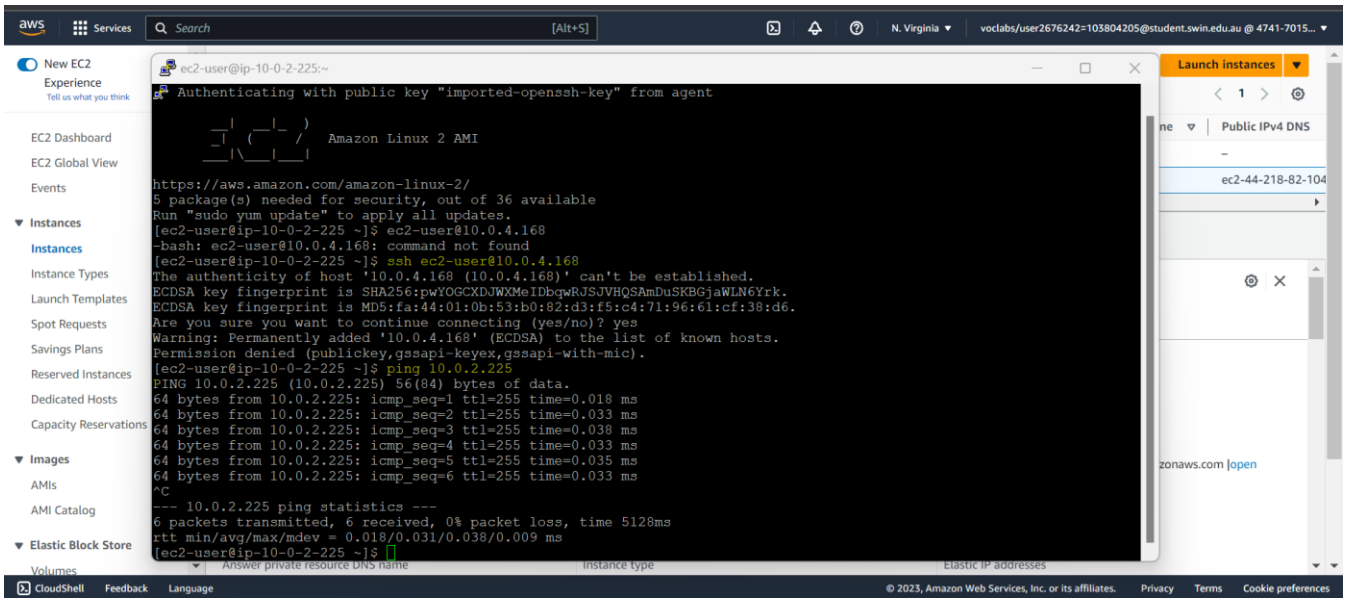
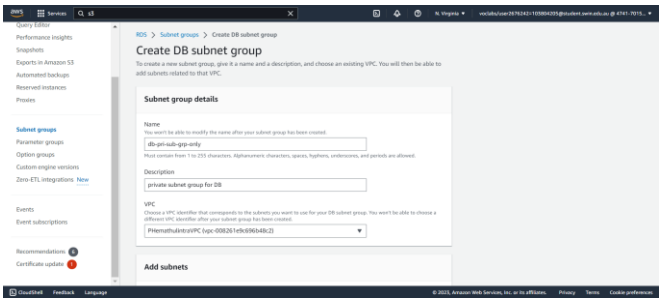


Figure 26 – shh to ec2 test instance

Step 9: configure RDS

- In RDS, create a subnet group



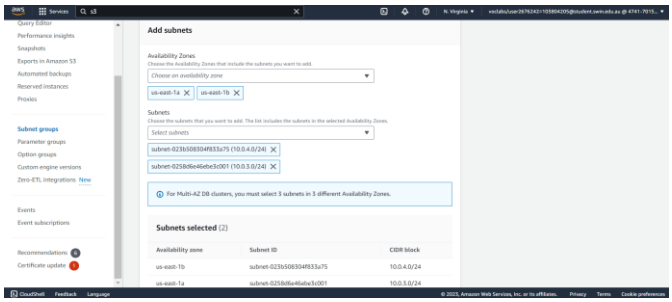
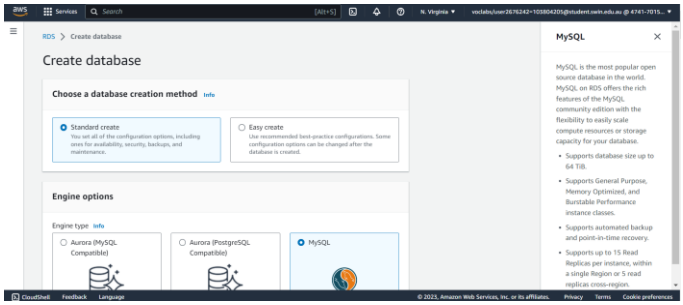
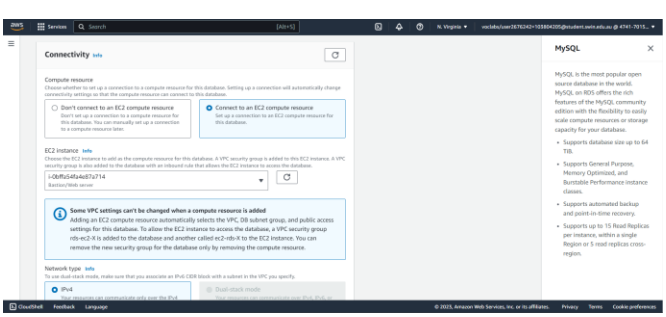


Figure 27 – create subnet group for DB, Figure 28 –subnet group for DB: select both zones and subnets



-Username = admin, Password = Pa55w.rd

- Create RDS



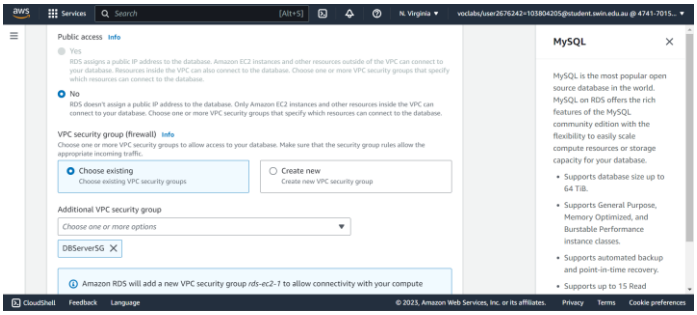


Figure 28 – create RDS: connect to web server, Figure 29 – create RDS: use the RDS security group

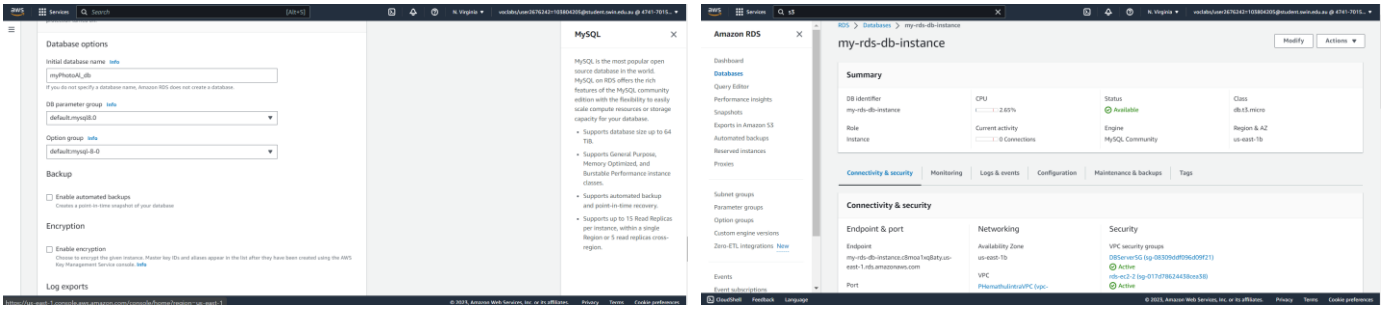


Figure 30 – create RDS: mark out automation and encryption, Figure 31– find RDS endpoint

- My endpoint = my-rds-db-instance.c8moa1xq8aty.us-east-1.rds.amazonaws.com

Step 10: Install phpMyAdmin on EC2

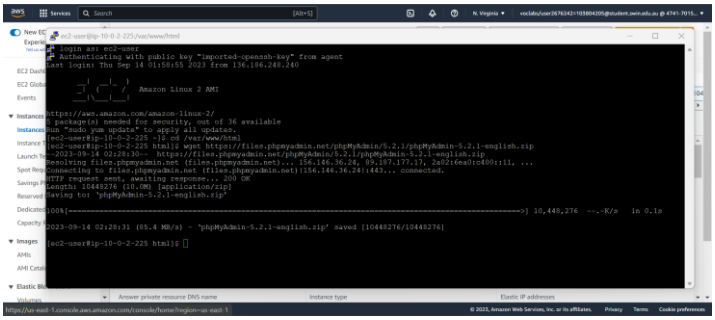


Figure 32 – Install phpMyAdmin: download

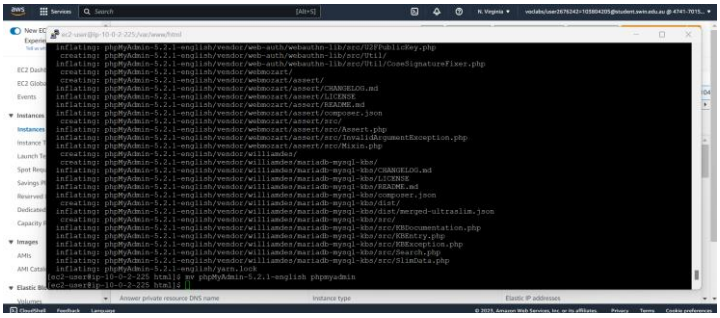


Figure 33 – Install phpMyAdmin: extract and change file name

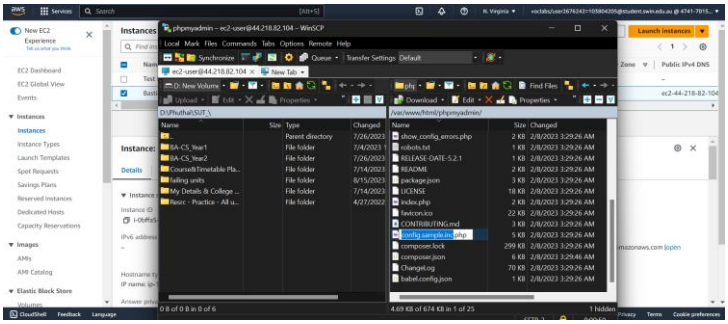


Figure 34 – WINSCP: change file name

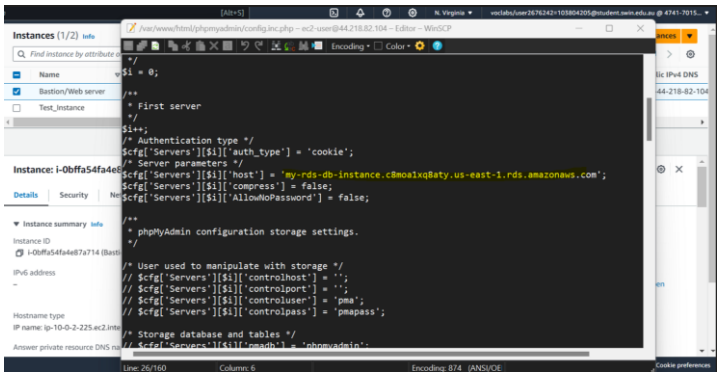


Figure 35 – WINSCP: insert RDS endpoint

Step 11: go to http://[publicDNSipv4]/phpmyadmin

- Create a table ‘photos’. Add 5 columns to the table.

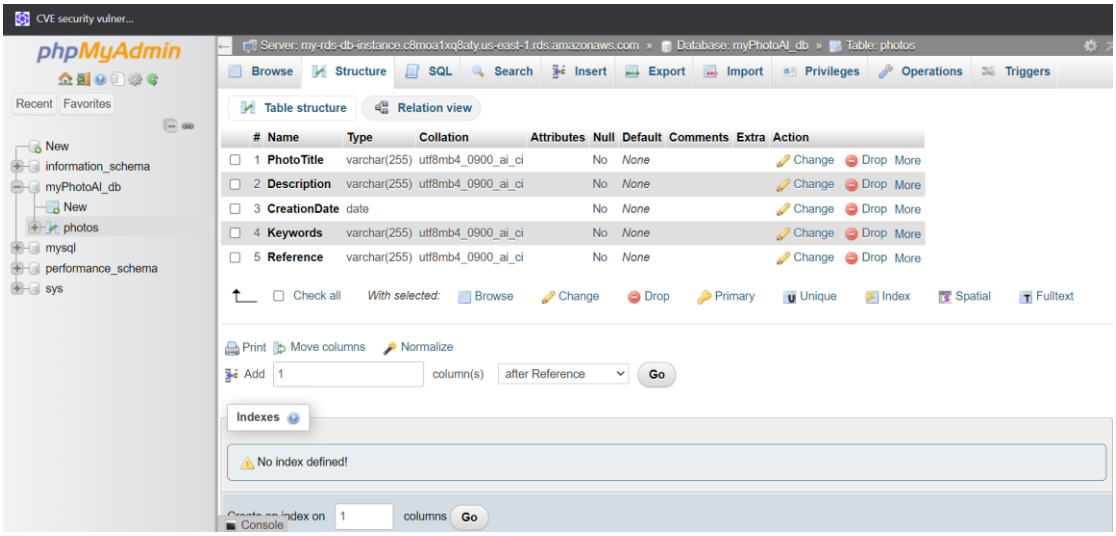


Figure 36 – create a table , 5 columns with input types

Step 12: create NACL

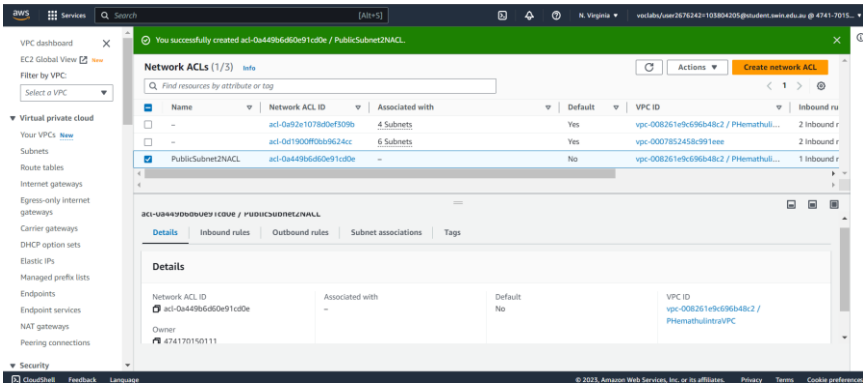


Figure 37 – create Network ACL

- Inbound rules and outbound rules should be the same and attach it to public subnet 2

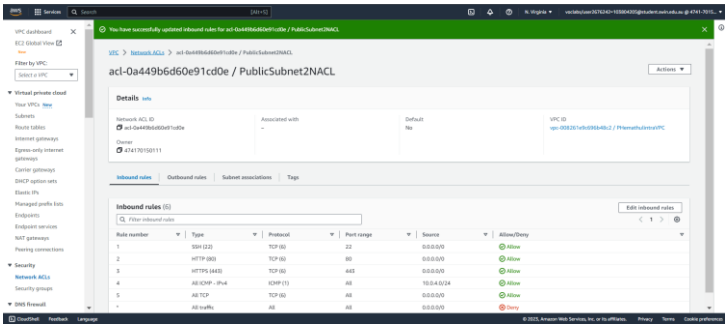


Figure 38 – NACL: edit inbound rules

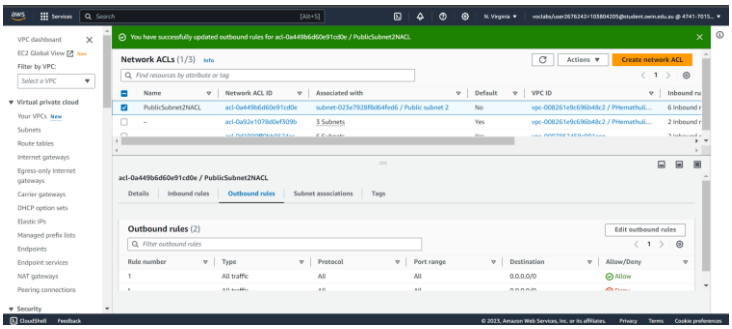
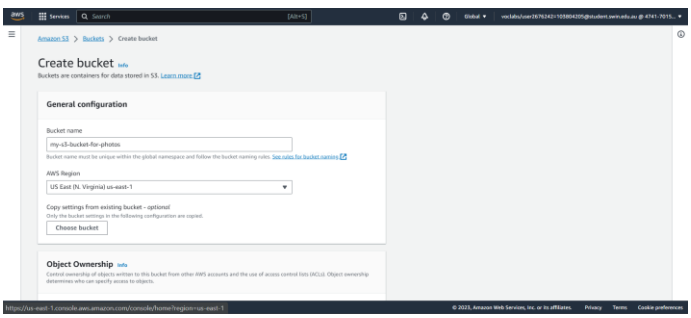


Figure 39 – edit outbound rules

Step 13: create S3 bucket



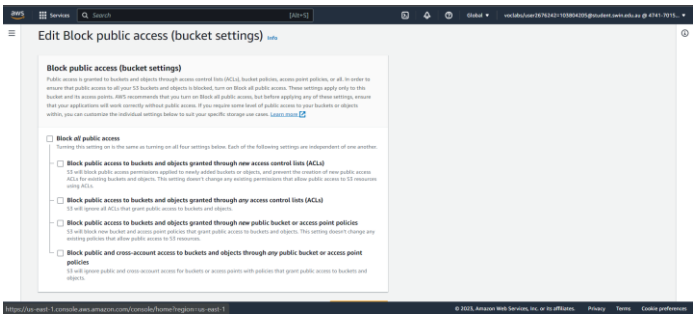


Figure 40 – create S3 bucket, Figure 41 – S3: edit public access

- Specify public access and bucket policy for bucket policy

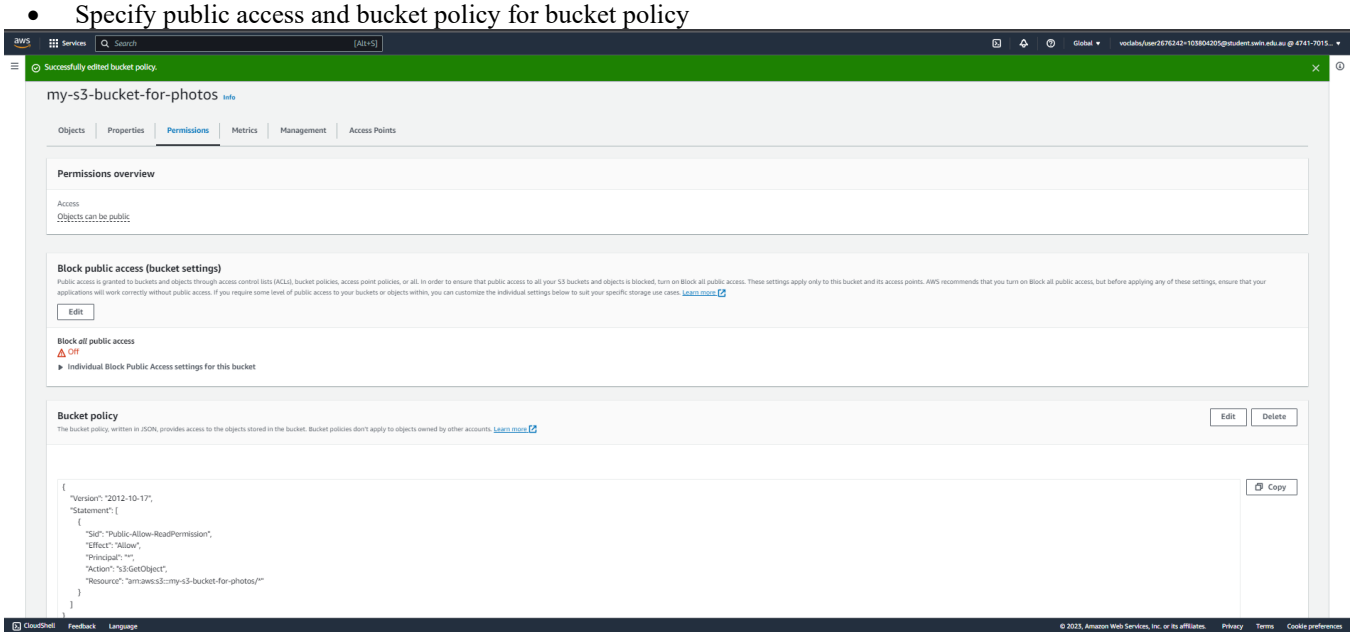


Figure 42 – S3: edit new bucket policy

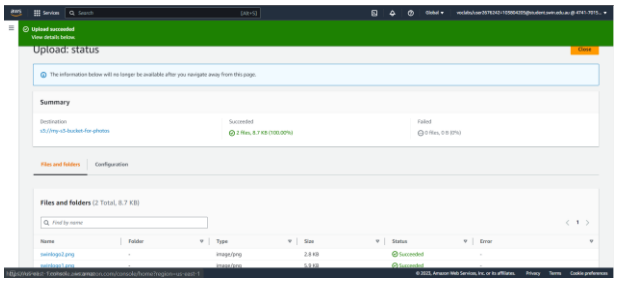


Figure 43 – upload photos successfully

Step 14: In phpMyAdmin, insert new roles with photos' URLs in photos table

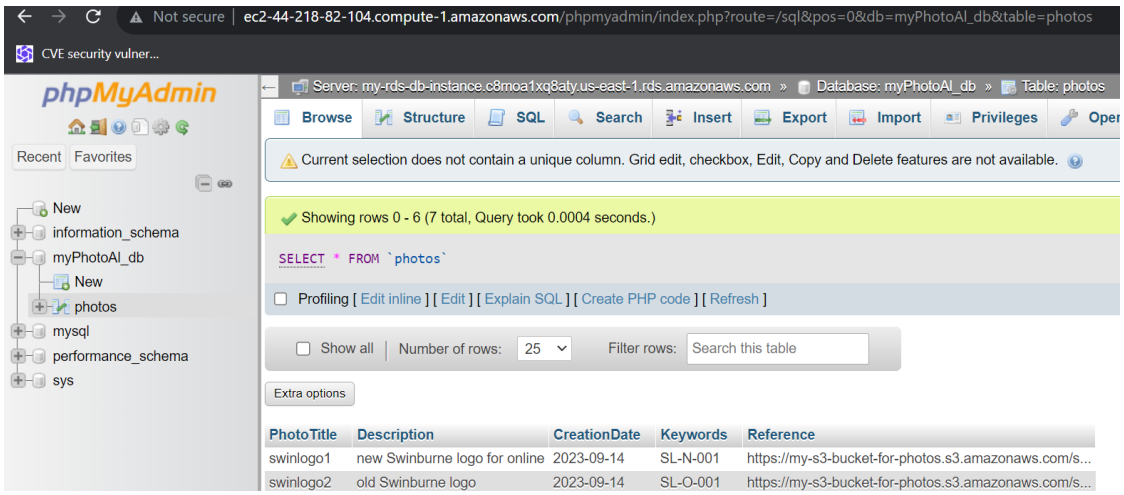


Figure 44 – insert row with URL of each S3 photo

Step 15: In photoalbumv3, enter correct code based on resource in AWS

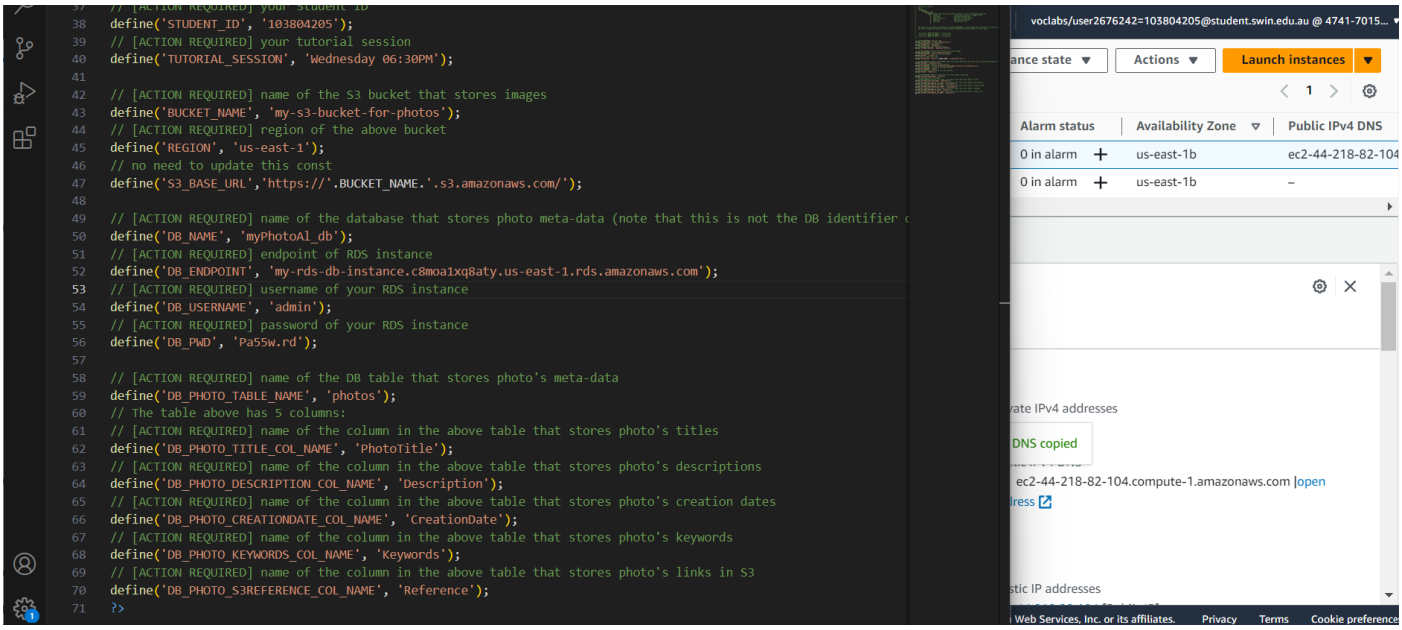


Figure 45 – edit constant.php with identities from previous configuration accordingly

Step 16: In WINSCP, create path of directory where photoalbum is hosting the website

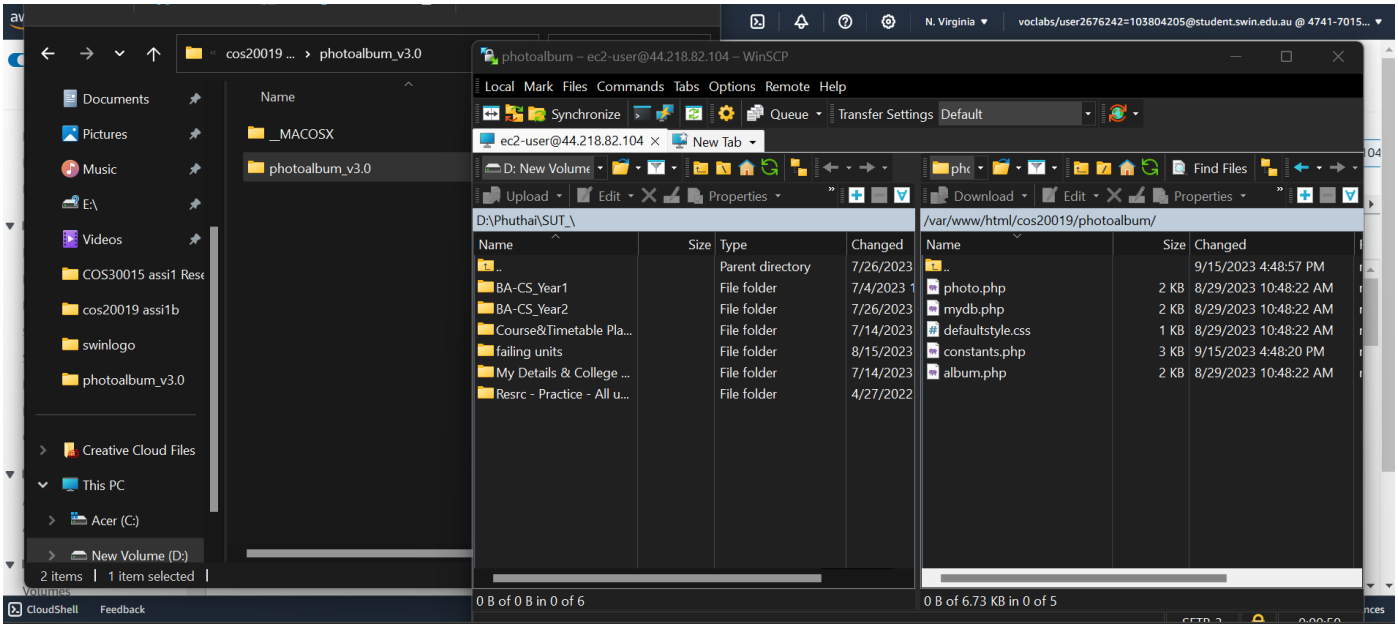
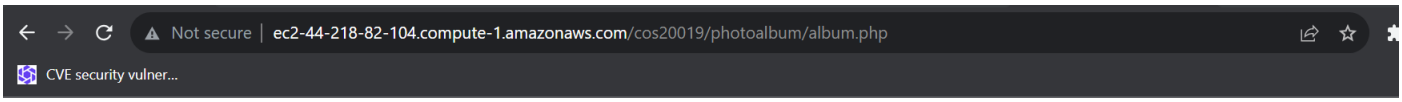


Figure 46 – WINSCP: uploading constant.php with other php files in photoalbum folder in the well-organized directory.

Step17: access to your photo album website which its database is in cloud

Website link: http://ec2-44-218-82-104.compute-1.amazonaws.com/cos20019/photoalbum/album.php



Student name: Phuthai Hemathulintra

Student ID: 103804205

Tutorial session: Wednesday 06:30PM

Uploaded photos:

Photo	Name	Description	Creation date	Keywords
SWINBURNE UNIVERSITY OF TECHNOLOGY	swinlogo1	new Swinburne logo for online	2023-09-14	SL-N-001
	swinlogo2	old Swinburne logo	2023-09-14	SL-O-001

Figure 47 – two photos were uploaded with description

Step 18: Testing

- add more photos

Student name: Phuthai Hemathulintra

Student ID: 103804205

Tutorial session: Wednesday 06:30PM

Uploaded photos:

Photo	Name	Description	Creation date	Keywords
	swinlogo1	new Swinburne logo for online	2023-09-14	SL-N-001
	swinlogo2	old Swinburne logo	2023-09-14	SL-O-001
	swinlogo3	razor	0000-00-00	SL-N-002
	swinlogo4	old logo with red background	0000-00-00	SL-O-002
	swinlogo5	Swinburne student Union logo	2023-09-15	SL-N-003
	swinlogo6	Swinburne online logo	2023-09-15	SL-N-004
	swinlogo7	Swinburne simple logo	2023-09-15	SL-N-005

Figure 48 – more photos were uploaded with description successfully

- ping to web server again to test NACL

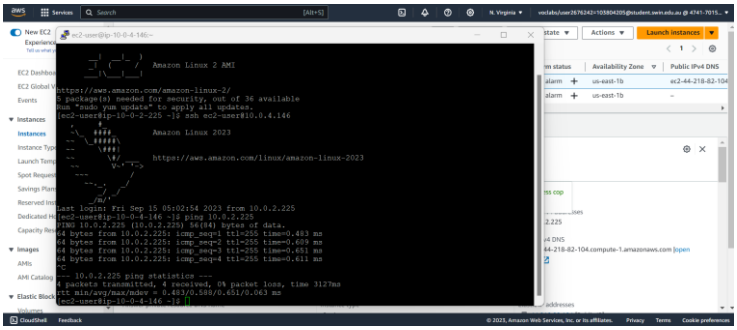


Figure 48 – more photos were uploaded with description successfully