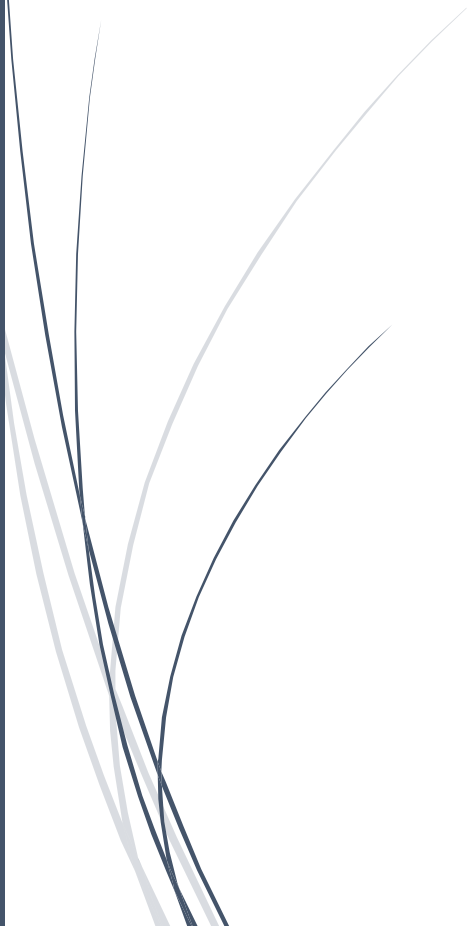


# USER MANUAL

9/13/2023



# I. Introduction

This assignment requires creating and deploying a photo album website on a simple AWS architecture.

## II . Create a secure Virtual Private Cloud (VPC)

Create a secure Virtual Private Cloud (VPC) .The VPC is named ‘AKaggdas’ because we need to give it a name using the initial of our first name and Last name. The VPC has two availability zones each with a private and public subnet with suitable CIDR as specified in the diagram of assignment 1b pdf.

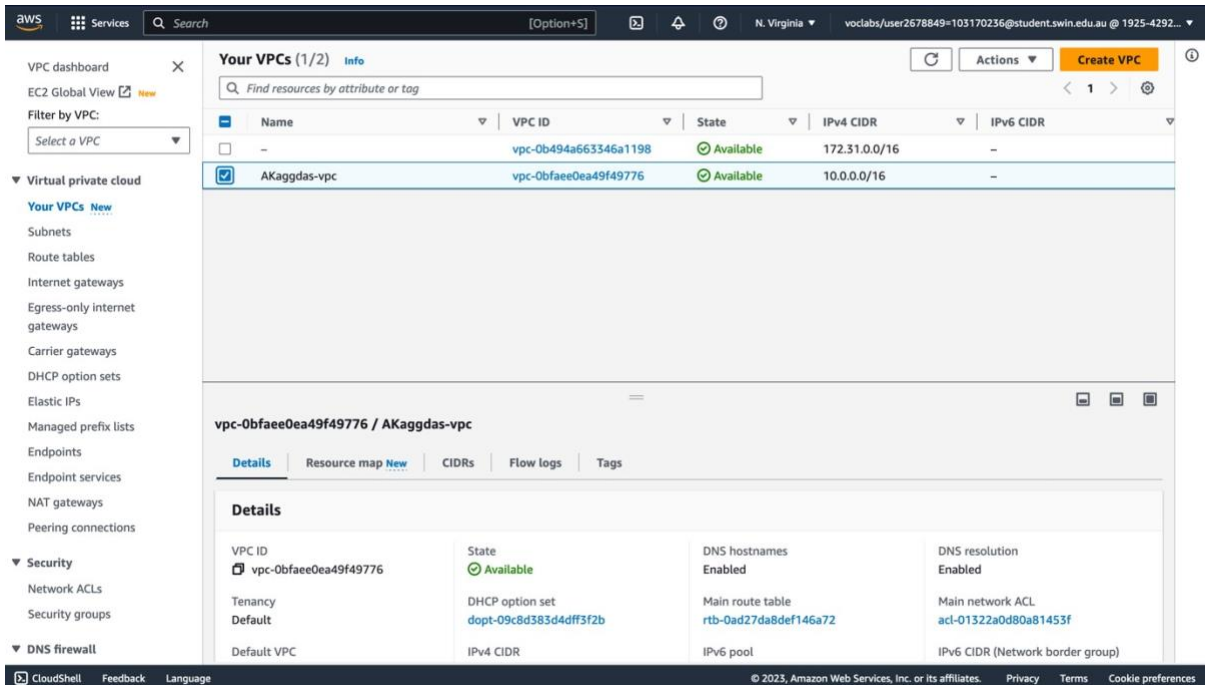


Figure 1 - Virtual Private Cloud

## III. Route Tables

### Public Route Table Configuration :

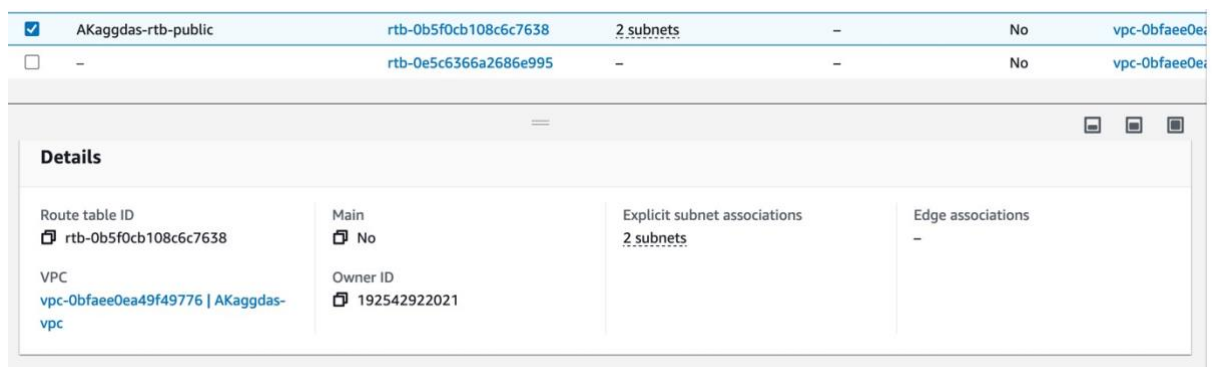


Figure 2 - 'AKaggdas' Public Route table

<input checked="" type="checkbox"/>	AKaggdas-rtb-public	rtb-0b5f0cb108c6c7638	2 subnets	-	No	vpc-0bfaee0
<input type="checkbox"/>	-	rtb-0e5c6366a2686e995	-	-	No	vpc-0bfaee0

Routes (2)

Both

< 1 >

⚙️

Destination	Target	Status	Propagated
0.0.0.0/0	igw-0608f3b4768674ad3	Active	No
10.0.0.0/16	local	Active	No

Figure 3 - 'AKaggdas' Public Routes

<input checked="" type="checkbox"/>	AKaggdas-rtb-public	rtb-0b5f0cb108c6c7638	2 subnets	-	No	vpc-0bfaee0
<input type="checkbox"/>	-	rtb-0e5c6366a2686e995	-	-	No	vpc-0bfaee0

Explicit subnet associations (2)

< 1 >

⚙️

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
AKaggdas-subnet-public1-us-east-1a	subnet-041b29723150d35c0	10.0.1.0/24	-
AKaggdas-subnet-public2-us-east-1b	subnet-00546cfae14df7214	10.0.2.0/24	-

Figure 4 - 'AKaggdas' Public Route table Subnet Associations

## Private Route Table Configuration :

<input checked="" type="checkbox"/>	AKaggdas-rtb-private	rtb-0ad27da8def146a72	2 subnets	-	Yes	vpc-0bfaee0
<input type="checkbox"/>	AKaggdas-rtb-public	rtb-0b5f0cb108c6c7638	2 subnets	-	No	vpc-0bfaee0
<input type="checkbox"/>	-	rtb-0e5c6366a2686e995	-	-	No	vpc-0bfaee0

Details

Route table ID rtb-0ad27da8def146a72  VPC vpc-0bfaee0ea49f49776   AKaggdas-vpc	Main Yes  Owner ID 192542922021	Explicit subnet associations 2 subnets	Edge associations -
--	---	---	------------------------

Figure 5 - 'AKaggdas' Private Route table

<input checked="" type="checkbox"/>	AKaggdas-rtb-private	rtb-0ad27da8def146a72	2 subnets	-	Yes	vpc-0bfaee0e
<input type="checkbox"/>	AKaggdas-rtb-public	rtb-0b5f0cb108c6c7638	2 subnets	-	No	vpc-0bfaee0e
<input type="checkbox"/>	-	rtb-0e5c6366a2686e995	-	-	No	vpc-0bfaee0e

Routes (1)

Both

< 1 >

⚙️

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

Figure 6 - 'AKaggdas' Private Routes

<input checked="" type="checkbox"/>	AKaggdas-rtb-private	rtb-0ad27da8def146a72	2 subnets	-	Yes	vpc-0bfaee0e
<input type="checkbox"/>	AKaggdas-rtb-public	rtb-0b5f0cb108c6c7638	2 subnets	-	No	vpc-0bfaee0e
<input type="checkbox"/>	-	rtb-0e5c6366a2686e995	-	-	No	vpc-0bfaee0e

Find subnet association

< 1 >

⚙️

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
AKaggdas-subnet-private1-us-east-1a	subnet-0a6e4ab5914747030	10.0.3.0/24	-
AKaggdas-subnet-private2-us-east-1b	subnet-0e95d2e0477ec2118	10.0.4.0/24	-

Figure 7 - 'AKaggdas' Private Route table Subnet Associations

## IV. Create Security groups

We create three security groups : 1] TestInstanceSG 2] WebServerSG 3] DBServerSG .We edit the **Inbound rules** for all the three as mentioned in the assignment 1b pdf and leave the outbound rules to default.

**sg-071d1203a3075fbb5 - TestInstanceSG**

**Details**

Security group name TestInstanceSG	Security group ID sg-071d1203a3075fbb5	Description TestInstance Security Group	VPC ID vpc-0bfaee0ea49f49776
Owner 192542922021	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

**Inbound rules (1/1)**

Name	Security group rule...	IP version	Type	Protocol	Port range
-	sgr-02d8959733a7e0...	IPv4	All traffic	All	All

Figure 8 - Test Instance Security Group

**sg-051675608941fba31 - WebServerSG**

**Details**

Security group name WebServerSG	Security group ID sg-051675608941fba31	Description WebServer Security Group	VPC ID vpc-0bfaee0ea49f49776
Owner 192542922021	Inbound rules count 3 Permission entries	Outbound rules count 1 Permission entry	

**Inbound rules (3)**

Name	Security group rule...	IP version	Type	Protocol	Port range
-	sgr-00ad61764d1186...	IPv4	HTTP	TCP	80
-	sgr-03551871743bd9...	-	All ICMP - IPv4	ICMP	All
-	sgr-033514d731d3d8...	IPv4	SSH	TCP	22

Figure 9 - Web Server Security Group

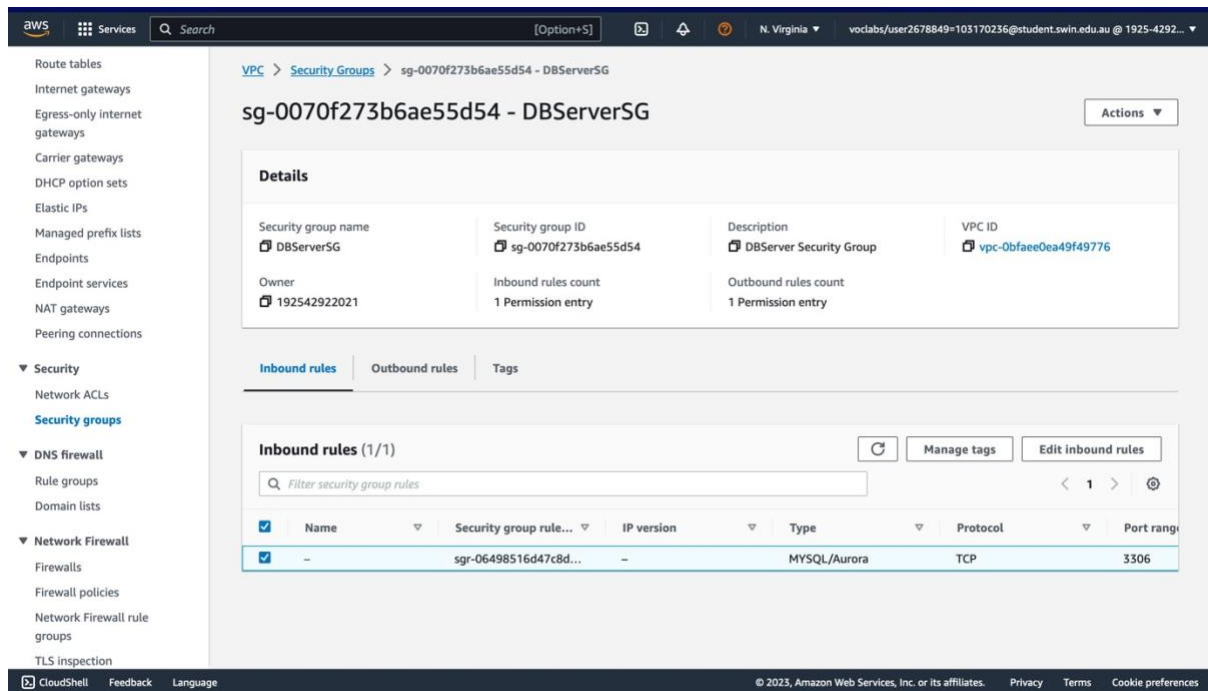


Figure 10 – Data base Server Security Group

## V. Create Key Pair

We create a key pair to associate with the EC2 instances .We name the key pair ‘assignment1b’.

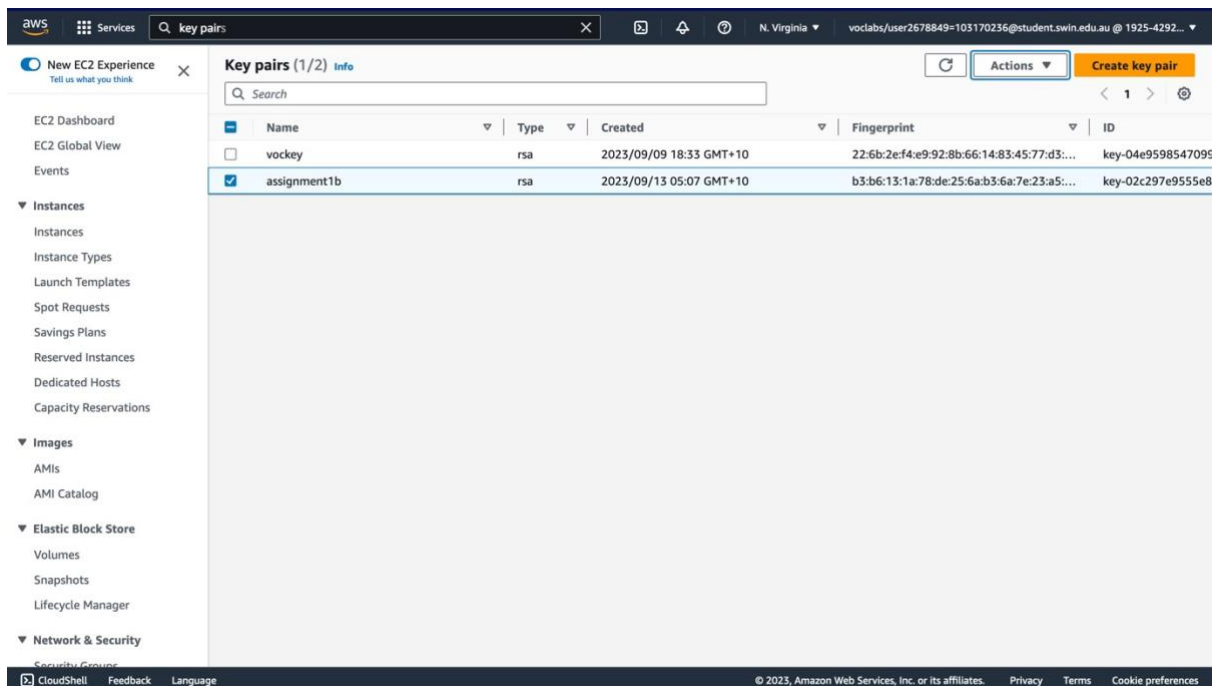


Figure 11 - Key Pair

## VI. Create EC2 Instances

Web server Instance:

We create two EC2 instances, a 'test' instance and a 'web server' instance. We first create the Web Server EC2 instance. The instance type for Web Server Instance is *t2.micro* and Amazon Machine Image is *Amazon Linux 2 AMI (HVM), SSD Volume Type*. For Key pair name we choose assignment1b. When we edit the network settings we select 'AKaggdas-vpc'. Now we expand Advanced settings copy the content provided to us in the 'Install\_PHP\_AWS.rtf' from assignment 1a and paste it in the user data box. We launch the instance. We then add an Elastic IP Address to this instance by allocating an Elastic IP address in the same region. The web server instance is in the public subnet.

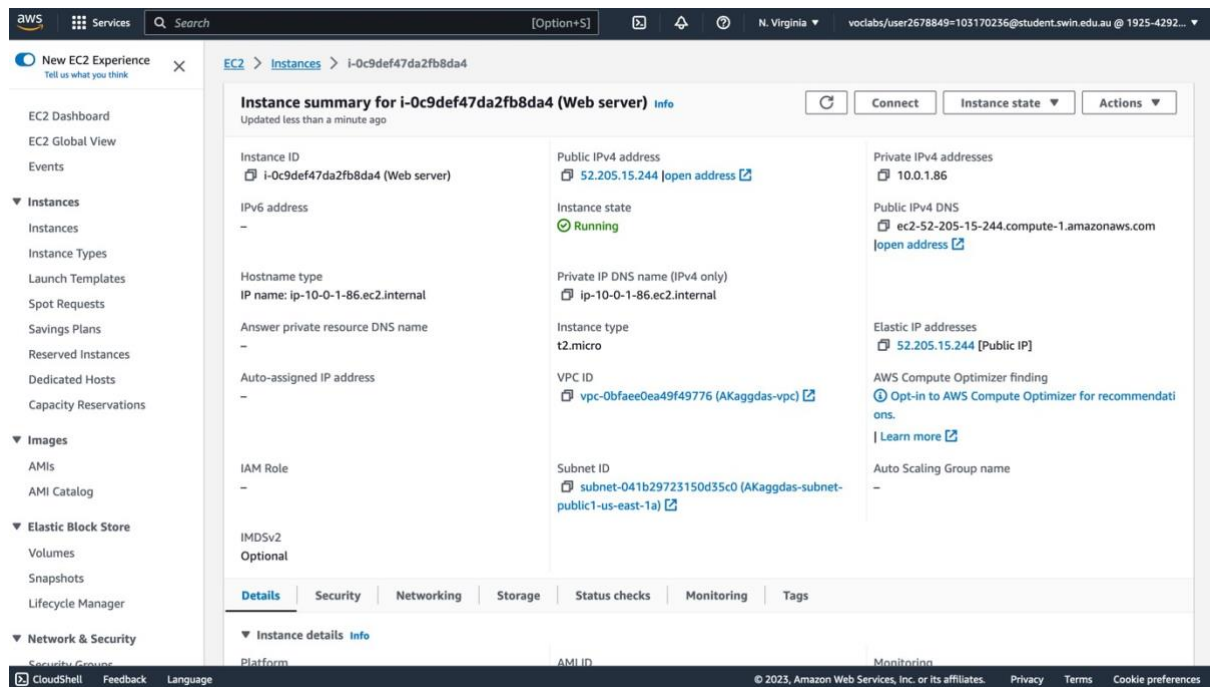


Figure 12 – Summary of Web Server Instance

## Test instance :

The configuration for the test instance is the same as for the web server instance, except that the test instance does not have a public IP address and is located in a private subnet. This instance is used for demonstration purposes only. It does not contribute to the functionality of Photo Album website.



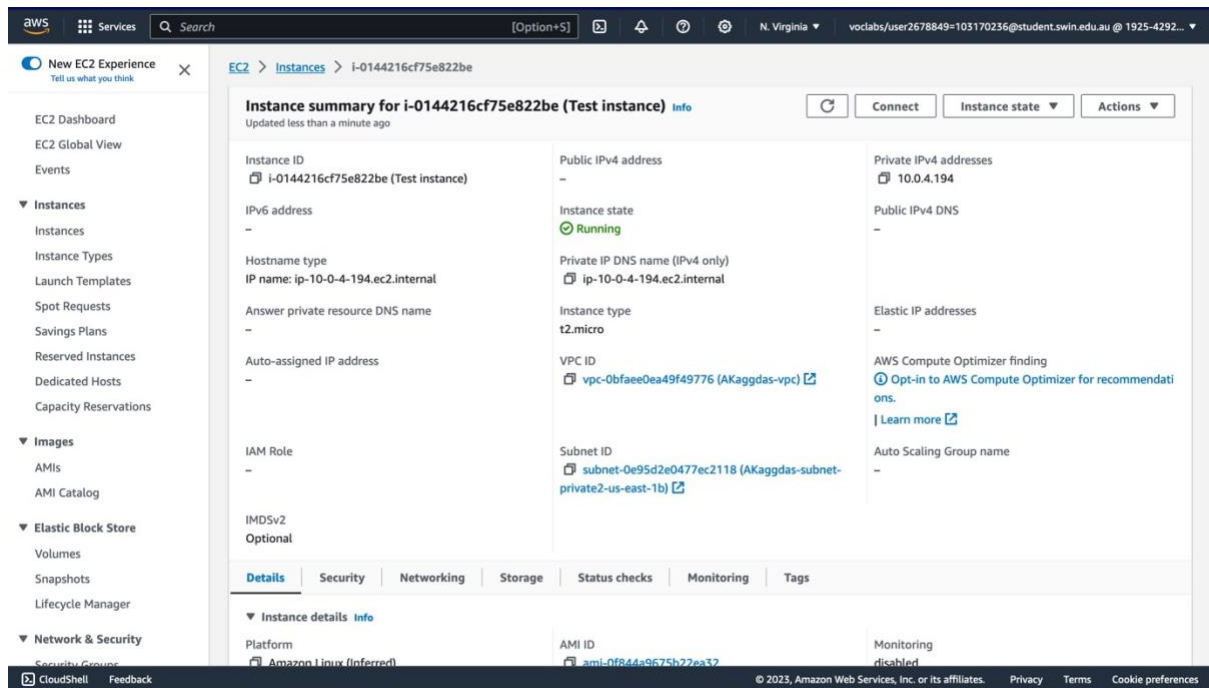


Figure 13 – Summary of Test Instance

We successfully access the test instance in the private subnet by using the commands displayed in the figure below.

```
aishwaryakaggdas@Aishwaryas-MacBook-Air Desktop % ssh-agent bash

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
bash-3.2$ ssh-add -L
-L: No such file or directory
bash-3.2$ ssh-add -K assignment1b.pem
Identity added: assignment1b.pem (assignment1b.pem)
bash-3.2$ ssh-add -L
-L: No such file or directory
bash-3.2$ ssh-add -L assignment1b.pem
-L: No such file or directory
Identity added: assignment1b.pem (assignment1b.pem)
bash-3.2$ ssh -A ec2-user@52.205.15.244
Last login: Fri Sep 15 12:38:27 2023 from 106.216.202.4

    __|  __|  )
    _| (___ /   Amazon Linux 2 AMI
    ___|___|___|

https://aws.amazon.com/amazon-linux-2/
5 package(s) needed for security, out of 36 available
Run "sudo yum update" to apply all updates.
-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory
[ec2-user@ip-10-0-1-86 ~]$ ssh ec2-user@10.0.4.194

    __|  __|  )
    _| (___ /   Amazon Linux 2 AMI
    ___|___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-4-194 ~]$
```

Figure 14 - SSH into an instance in a private subnet



We are able to establish a connection (ICMP ping) between this instance and the Bastion/Web server instance by pinging the public IP address of the web server.

```
https://aws.amazon.com/amazon-linux-2/
5 package(s) needed for security, out of 36 available
Run "sudo yum update" to apply all updates.
-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory
[ec2-user@ip-10-0-1-86 ~]$ ping 10.0.4.194
PING 10.0.4.194 (10.0.4.194) 56(84) bytes of data:
64 bytes from 10.0.4.194: icmp_seq=1 ttl=255 time=0.799 ms
64 bytes from 10.0.4.194: icmp_seq=2 ttl=255 time=0.713 ms
64 bytes from 10.0.4.194: icmp_seq=3 ttl=255 time=0.770 ms
64 bytes from 10.0.4.194: icmp_seq=4 ttl=255 time=0.752 ms
64 bytes from 10.0.4.194: icmp_seq=5 ttl=255 time=0.790 ms
64 bytes from 10.0.4.194: icmp_seq=6 ttl=255 time=0.742 ms
64 bytes from 10.0.4.194: icmp_seq=7 ttl=255 time=0.709 ms
64 bytes from 10.0.4.194: icmp_seq=8 ttl=255 time=0.807 ms
64 bytes from 10.0.4.194: icmp_seq=9 ttl=255 time=0.706 ms
64 bytes from 10.0.4.194: icmp_seq=10 ttl=255 time=0.717 ms
64 bytes from 10.0.4.194: icmp_seq=11 ttl=255 time=0.773 ms
64 bytes from 10.0.4.194: icmp_seq=12 ttl=255 time=0.777 ms
```

Figure 15 – Ping the web server

## VII. Create an Amazon RDS DB Instance

We name the RDS assignment 1b .This RDS instance has the following configs:

- DB engine version: *MySQL 8.0.34*
- Template: *Free tier*
- Public access: *No*

The screenshot displays the Amazon RDS console interface. On the left is a navigation sidebar with options like Dashboard, Databases, Query Editor, and Performance insights. The main panel shows the configuration for the RDS instance 'assignment1b'. A 'Summary' section provides a quick overview: DB identifier 'assignment1b', CPU '-', Status 'Available', Class 'db.t3.micro', Role 'Instance', Current activity, Engine 'MySQL Community', and Region & AZ 'us-east-1b'. Below this, the 'Connectivity & security' tab is selected, showing details for Endpoint & port, Networking, and Security. The Endpoint & port section lists the endpoint 'assignment1b.cuegnziesx.us-east-1.rds.amazonaws.com' and port '3306'. The Networking section shows the Availability Zone 'us-east-1b', VPC 'AKaggdas-vpc (vpc-0bfaee0ea49f49776)', and Subnet group 'default-vpc-0bfaee0ea49f49776'. The Security section shows VPC security groups 'DBServerSG (sg-0070f273b6ae55d54)' as 'Active', and 'Publicly accessible' set to 'No'. The bottom of the console shows a footer with copyright information and links to Privacy, Terms, and Cookie preferences.

Figure 16- Summary of assignment 1b RDS



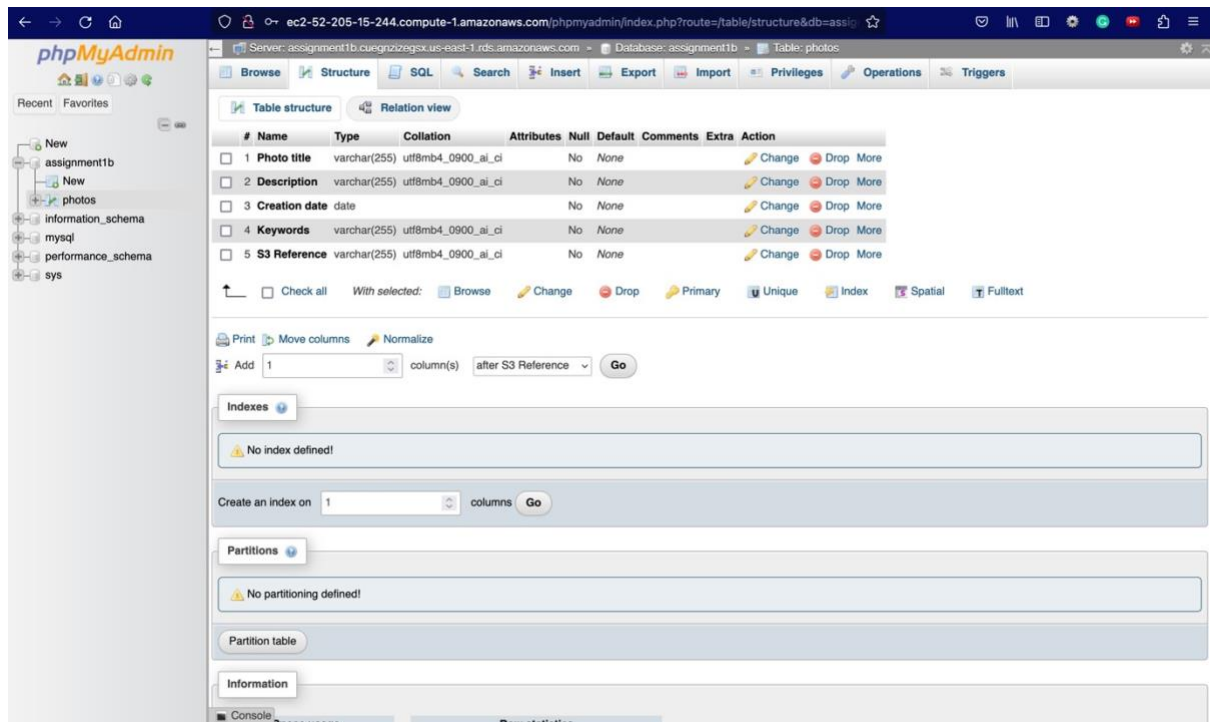


Figure 19- Access phpMyAdmin from your local machine and create Appropriate table on your RDS Database:

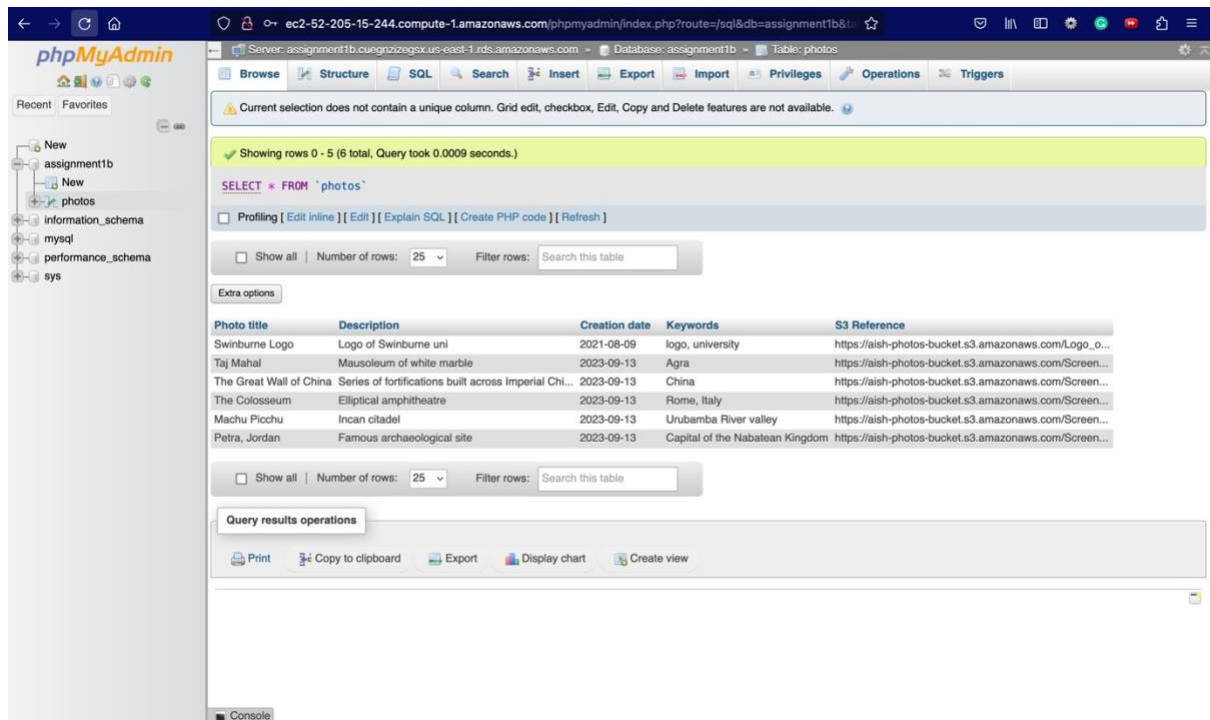


Figure 20- Populate 'photos' table with a few records

## IX. Reconfigure phpMyAdmin

We Open Filezilla and navigate to phpmyadmin directory (var/www/html/phpmyadmin). We change the name of config.sample.inc.php file to config.inc.php. We open config.inc.php file and look for this line:  
`$cfg['Servers'][$i]['host'] = 'localhost';`  
 Replace 'localhost' with the endpoint of your RDS instance.  
`$cfg['Servers'][$i]['host'] = 'assignment1b.cuegnzix.us-east-1.rds.amazonaws.com';`

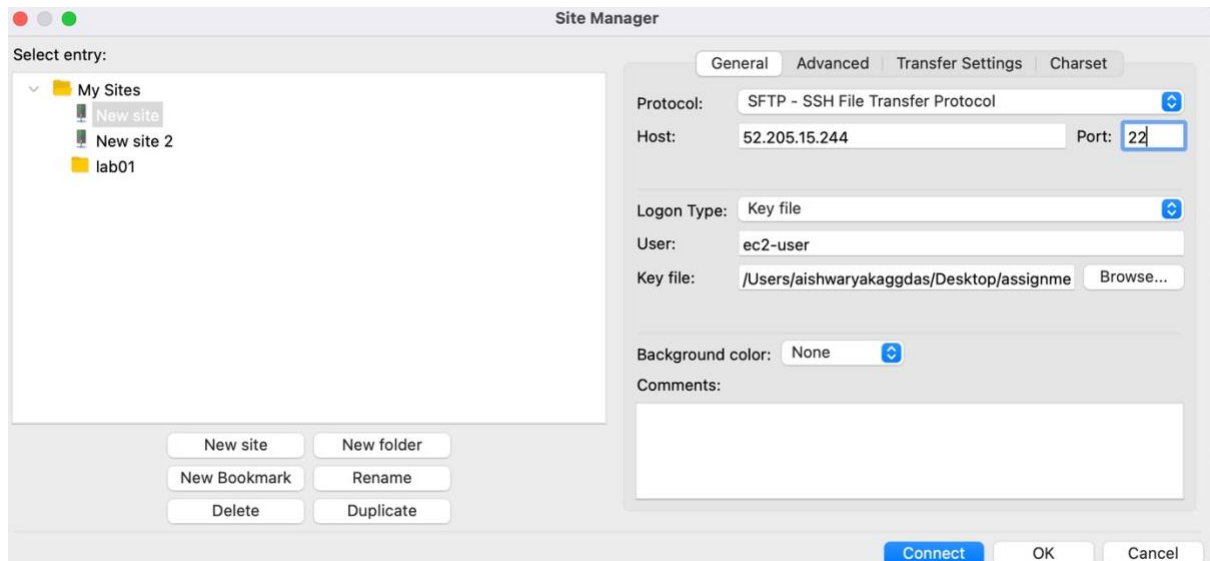


Figure 21 – Connect to FileZilla

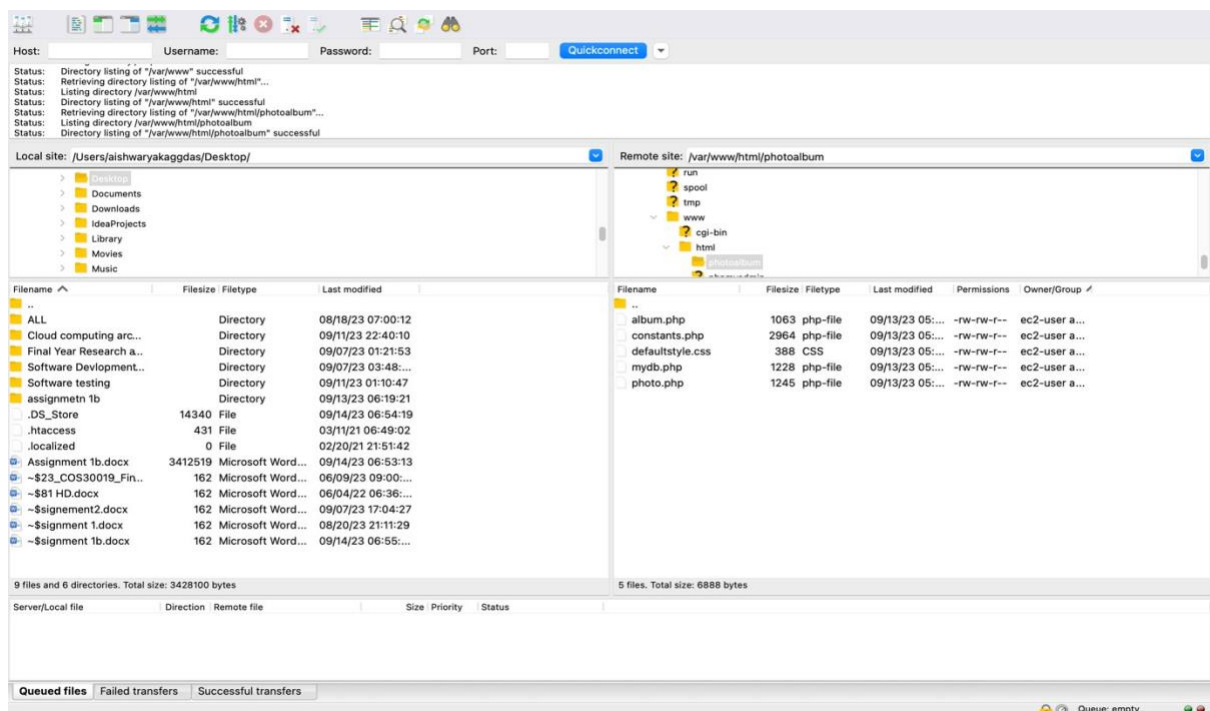


Figure 22- Navigate to phpmyadmin directory

## X . Functional requirements of Photo Album website

The Photo Album website has the following functional requirements.

### 2.1 – Photo storage

We create an S3 bucket and name it 'aish-photos-bucket' to store our photos. We manually upload some photos onto S3 bucket and ensure they have been successfully uploaded.

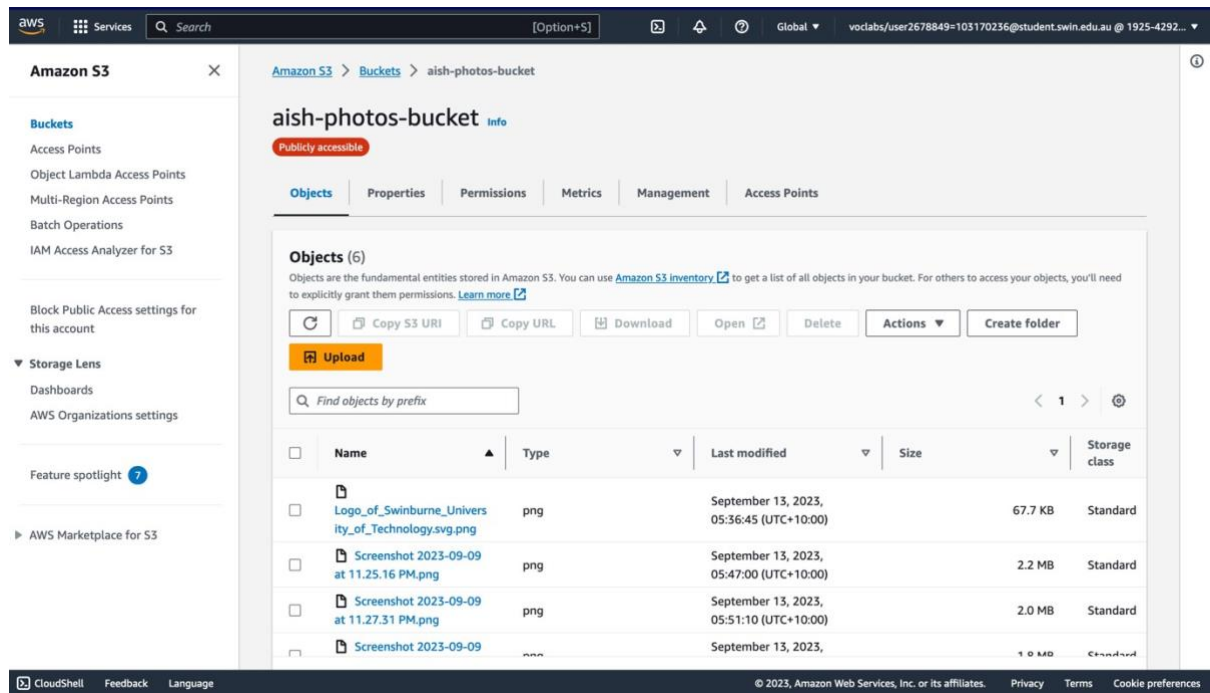


Figure 23 – aish-photos bucket

## 2.2 – Photo meta-data in RDS Database

We populate the table with a few records the first being

- Photo title: *Swinburne Logo*
- Description: *Logo of Swinburne uni*
- Creation date: *2021-08-09*
- Keywords: *logo, university*
- Object URL in S3: [https://aish-photosbucket.s3.amazonaws.com/Logo\\_of\\_Swinburne\\_University\\_of\\_Technology.svg.png](https://aish-photosbucket.s3.amazonaws.com/Logo_of_Swinburne_University_of_Technology.svg.png)

Student name: Aishwarya Kaggdas  
Student ID: 103170236  
Tutorial session: Sunday 09:15AM

Uploaded photos:

Photo	Name	Description	Creation date	Keywords
	Swinburne Logo	Logo of Swinburne uni	2021-08-09	logo, university

Figure 24 – First record of the table

## 2.3 – Photo Album website functionality

The website is able to list all the photos (stored in the S3 bucket) along with their meta-data (stored in the database).

Student name: Aishwarya Kaggdas

Student ID: 103170236

Tutorial session: Sunday 09:15AM

Uploaded photos:






Photo	Name	Description	Creation date	Keywords
	Swinburne Logo	Logo of Swinburne uni	2021-08-09	logo, university
	Taj Mahal	Mausoleum of white marble	2023-09-13	Agra
	The Great Wall of China	Series of fortifications built across Imperial China	2023-09-13	China
	The Colosseum	Elliptical amphitheatre	2023-09-13	Rome, Italy
	Machu Picchu	Incan citadel	2023-09-13	Urubamba River valley
	Petra, Jordan	Famous archaeological site	2023-09-13	Capital of the Nabatean Kingdom

Figure 25 – album.php

URL of the album.php :

<http://ec2-52-205-15-244.compute-1.amazonaws.com/photoalbum/album.php>

## XI. Problems And Achievements

### Problems:

The objects in the S3 bucket were not publicly accessible, which was one of the challenges I faced. To solve the problem, I researched and discovered that we require to attach a public bucket policy to make the objects accessible to everyone. The link that I used to resolve this issue is provided below.

URL : <https://saturncloud.io/blog/how-to-access-images-from-amazon-s3-by-url/>

The web server needed an additional layer of protection, which was my second problem. I was unable to design and deploy a network ACL .I attempted but failed to complete this task. It was the most difficult part of the assignment.

**Achievements :**

- Create VPC with 2 public and 2 private subnets
- Correct Public and Private Routing tables with correct subnet associations
- Security groups properly configured and attached.
- Correct Web server and Test instances running in correct subnets
- Database schema as specified
- Database running in correct subnets
- S3 objects publicly accessible, using proper access policy

**Functional Requirements:**

- album.php page displayed from EC2 Web server
- Provided URL is persistent (Elastic IP Association)
- Photos loaded from S3 with matching metadata from RDS
- Web server instance reachable from Test instance via ICMP