

COS20019 Cloud Computing Architecture

Assignment 1B

Creating and deploying photo album website onto a simple AWS infrastructure

Student Name: Harry

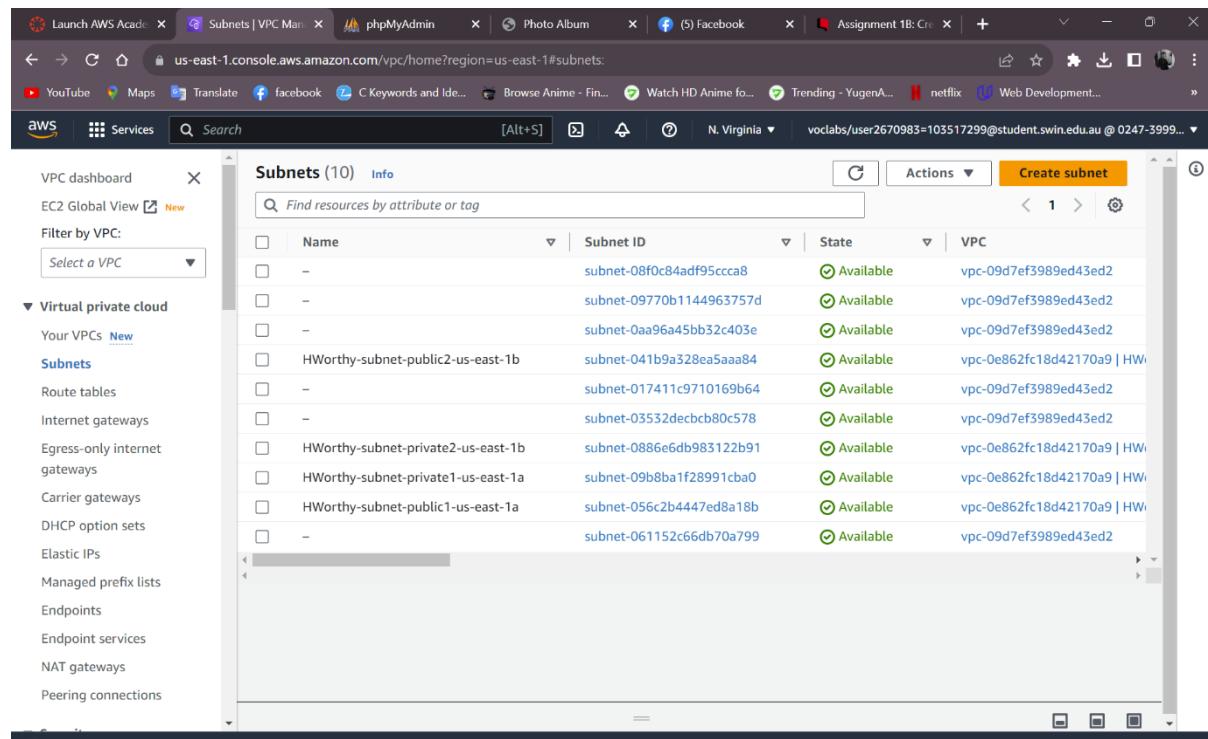
Student ID: 103517299

URL of the album.php page on EC2:

<http://ec2-35-172-50-182.compute-1.amazonaws.com/cos20019/harrysalbum/album.php>

1. VPC:

I've successfully created a new Amazon Virtual Private Cloud (VPC) named "HWorthyVPC" in the us-east-1 region. The VPC is designed to have two availability zones, each with its own private and public subnet. Public Subnet 1 uses the CIDR block 10.0.1.0/24, and Public Subnet 2 uses the CIDR block 10.0.2.0/24. Private Subnet 1 uses the CIDR block 10.0.3.0/24, and Private Subnet 2 uses the CIDR block 10.0.4.0/24. I've associated public subnets with a public route table that routes to the Internet Gateway. As a result, any traffic destined for the internet from these public subnets will be directed through the IGW. This setup enables efficient network management and allows resources in the public subnets to communicate with the internet while maintaining security and control over the private subnets.



The screenshot shows the AWS VPC Subnets page. The left sidebar is collapsed, showing options like VPC dashboard, EC2 Global View, Filter by VPC (Select a VPC), and a list of VPC components including Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, and Peering connections. The main content area is titled 'Subnets (10)' and contains a table with 10 rows. The columns are: Name, Subnet ID, State, and VPC. The table shows the following data:

Name	Subnet ID	State	VPC
-	subnet-08f0c84adf95ccca8	Available	vpc-09d7ef3989ed43ed2
-	subnet-09770b1144963757d	Available	vpc-09d7ef3989ed43ed2
-	subnet-0aa96a45bb32c403e	Available	vpc-09d7ef3989ed43ed2
HWorthy-subnet-public2-us-east-1b	subnet-041b9a328ea5aaa84	Available	vpc-0e862fc18d42170a9 HW
-	subnet-017411c9710169b64	Available	vpc-09d7ef3989ed43ed2
-	subnet-03532decbb80c578	Available	vpc-09d7ef3989ed43ed2
HWorthy-subnet-private2-us-east-1b	subnet-0886e6db983122b91	Available	vpc-0e862fc18d42170a9 HW
HWorthy-subnet-private1-us-east-1a	subnet-09b8ba1f28991cba0	Available	vpc-0e862fc18d42170a9 HW
HWorthy-subnet-public1-us-east-1a	subnet-056c2b4447ed8a18b	Available	vpc-0e862fc18d42170a9 HW
-	subnet-061152c66db70a799	Available	vpc-09d7ef3989ed43ed2

fig1: VPC subnets

Screenshot of the AWS VPC Manager console showing the details of a VPC and its resources.

VPC Details:

VPC ID	vpc-0e862fc18d42170a9	State	Available
Tenancy	Default	DHCP option set	dopt-0de1aa77a6caaf642
Default VPC	No	IPv4 CIDR	10.0.0.0/16
Network Address Usage metrics	Disabled	Route 53 Resolver DNS Firewall rule groups	Failed to load rule groups
		DNS hostnames	Enabled
		Main route table	rtb-0f6208af9b28d0fe3
		IPv6 pool	-
		Owner ID	024739990749
		DNS resolution	Enabled
		Main network ACL	acl-062b45a38b46f5071
		IPv6 CIDR (Network border group)	-

Resource map:

```

graph LR
    VPC[VPC] --- Subnets[Subnets (4)]
    VPC --- RTTables[Route tables (4)]
    VPC --- NCConnections[Network connections (3)]
    
    Subnets --- usEast1a[us-east-1a]
    Subnets --- usEast1b[us-east-1b]
    usEast1a --- HWorthySubnetPublic1[HWorthy-subnet-public1-us-east-1a]
    usEast1a --- HWorthySubnetPrivate1[HWorthy-subnet-private1-us-east-1a]
    usEast1b --- HWorthySubnetPublic2[HWorthy-subnet-public2-us-east-1b]
    usEast1b --- HWorthySubnetPrivate2[HWorthy-subnet-private2-us-east-1b]
    
    RTTables --- HWorthyRTBPrivate1[HWorthy-rtb-private1-us-east-1a]
    RTTables --- HWorthyRTBPublic[HWorthy-rtb-public]
    RTTables --- HWorthyRTBNat[rtb-0f6208af9b28d0fe3]
    RTTables --- HWorthyRTBPrivate2[HWorthy-rtb-private2-us-east-1b]
    
    NCConnections --- HWorthyIGW[HWorthy-igw]
    NCConnections --- HWorthyNat[HWorthy-nat-public1-us-east-1a]
    NCConnections --- HWorthyVPC[HWorthy-vpce-s3]
  
```

The diagram illustrates the architecture of the VPC. It starts with a central VPC node, which branches into four main components: Subnets, Route tables, and Network connections. The Subnets component contains two subnets under 'us-east-1a' and two under 'us-east-1b'. The Route tables component contains four route tables: 'HWorthy-rtb-private1-us-east-1a', 'HWorthy-rtb-public', 'rtb-0f6208af9b28d0fe3', and 'HWorthy-rtb-private2-us-east-1b'. The Network connections component shows three connections to external networks: 'HWorthy-igw', 'HWorthy-nat-public1-us-east-1a', and 'HWorthy-vpce-s3'.

fig2: Route table

2. Security Groups:

I've created 3 security groups respectively TestInstanceSG, WebServerSG, And DBServerSG where one is for Test Instance, one is for Bastion server (Harrys Server) and the other is for RDS Database Instance. During the process of creating security groups, I initially configured outbound rules instead of inbound rules, which required some time to identify and rectify the issue.

The screenshot shows the AWS VPC Management Console with the URL <https://us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#SecurityGroup:groupId=sg-0a084311645320006>. The page displays the details for the security group sg-0a084311645320006 - TestInstanceSG. The details include:

Security group name	sg-0a084311645320006	Security group ID	sg-0a084311645320006	Description	All traffic	VPC ID	vpc-0e862fc18d42170a9
Owner	024739990749	Inbound rules count	1 Permission entry	Outbound rules count	1 Permission entry		

The Inbound rules section shows one rule:

Name	Security group rule...	IP version	Type	Protocol
sgr-0030a26c78151a...	IPv4	All traffic	All	

fig3: Test Instance SG

The screenshot shows the AWS VPC Management Console with the URL <https://us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#SecurityGroup:groupId=sg-09e90a9baba16e940>. The page displays the details for the security group sg-09e90a9baba16e940 - WebServerSG. The details include:

Security group name	sg-09e90a9baba16e940	Security group ID	sg-09e90a9baba16e940	Description	httpsshimcp	VPC ID	vpc-0e862fc18d42170a9
Owner	024739990749	Inbound rules count	3 Permission entries	Outbound rules count	1 Permission entry		

The Inbound rules section shows three rules:

Name	Security group rule...	IP version	Type	Protocol
sgr-0aad7d1e97a244e...	IPv4	HTTP	TCP	
sgr-031b138816b715...	IPv4	SSH	TCP	
sgr-0eb2ee833fc418b40	-	All ICMP - IPv4	ICMP	

fig4: WebServer SG

The screenshot shows the AWS VPC Management Console with the URL <https://us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#SecurityGroup:groupId=sg-0962e073f8803eb05>. The page displays the details of a security group named "sg-0962e073f8803eb05 - DBServerSG".

Details:

Security group name	sg-0962e073f8803eb05	Security group ID	sg-0962e073f8803eb05	Description	sql	VPC ID	vpc-0e862fc18d42170a9
Owner	024739990749	Inbound rules count	1 Permission entry	Outbound rules count	1 Permission entry		

Inbound rules (1/1):

Name	Security group rule...	IP version	Type	Protocol
-	sgr-0f085175e44c7236c	-	MySQL/Aurora	TCP

fig5: DBServerSG

3.EC2 Instances:

I have successfully deployed an EC2 instance named it Harrys Server in Public Subnet 2 to serve as both the web server for the Photo Album web application and a bastion host for SSH access to the Test instance in the private subnet. The instance is configured with Apache web server, PHP packages, and uses an Elastic IP address for consistent accessibility, ensuring the stability of the web application and facilitating secure remote access.

The screenshot shows the AWS EC2 Instance Details page for an instance named "Harrys Server" (i-0323436f0e3b8b51e). The instance is currently running. Key details include:

Attribute	Value
Public IPv4 address	3.81.70.70
Instance state	Running
Private IP address	10.0.2.141
Public IPv4 DNS	ec2-3-81-70-70.compute-1.amazonaws.com
Instance type	t2.micro
VPC ID	vpc-0e862fc18d42170a9 (HWorthy-vpc)
Subnet ID	subnet-041b9a328ea5aaa84 (HWorthy-subnet-public2-us-east-1b)

fig6: HarrysServer

Instance summary for i-0323436f0e3b8b51e (Harrys Server) [Info](#)
Updated less than a minute ago

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0323436f0e3b8b51e (Harrys Server)	35.172.50.182 open address Details	10.0.2.141
IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-35-172-50-182.compute-1.amazonaws.com open address Details
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-10-0-2-141.ec2.internal	ip-10-0-2-141.ec2.internal	35.172.50.182 [Public IP] Details
Answer private resource DNS name	Instance type	AWS Compute Optimizer finding
-	t2.micro	Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address	VPC ID	Auto Scaling Group name
-	vpc-0e862fc18d42170a9 (HWorthy-vpc) Details	-
IAM Role	Subnet ID	
-	subnet-041b9a328ea5aaa84 (HWorthy-subnet-public2-us-east-1b) Details	

fig7: Elastic IP

Once connected to the test instance via PuTTY, use the ping 10.0.4.165 command to ping.

Instance summary for i-038634258170a1cf7 (TestInstance) [Info](#)
Updated less than a minute ago

Instance ID	Public IPv4 address	Private IPv4 addresses
i-038634258170a1cf7	35.172.50.182 open address Details	10.0.4.165
IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-35-153-127-184.compute-1.amazonaws.com open address Details
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-10-0-2-141.ec2.internal	ip-10-0-2-141.ec2.internal	-
Answer private resource DNS name	Instance type	AWS Compute Optimizer finding
-	t2.micro	Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address	VPC ID	Auto Scaling Group name
-	vpc-0e862fc18d42170a9 (HWorthy-vpc) Details	-
IAM Role	Subnet ID	
-	subnet-041b9a328ea5aaa84 (HWorthy-subnet-public2-us-east-1b) Details	

fig8: ping

4.RDS Database Instance:

I configured an RDS instance in a private subnet, ensuring security with access limited to the WebServerSG security group. While residing in a private subnet, I've established internet access for setup and maintenance. Furthermore, I've created a photos table within the database to store essential metadata about the images stored in the S3 bucket, including photo title, description, creation date, keywords, and references to the S3 objects.

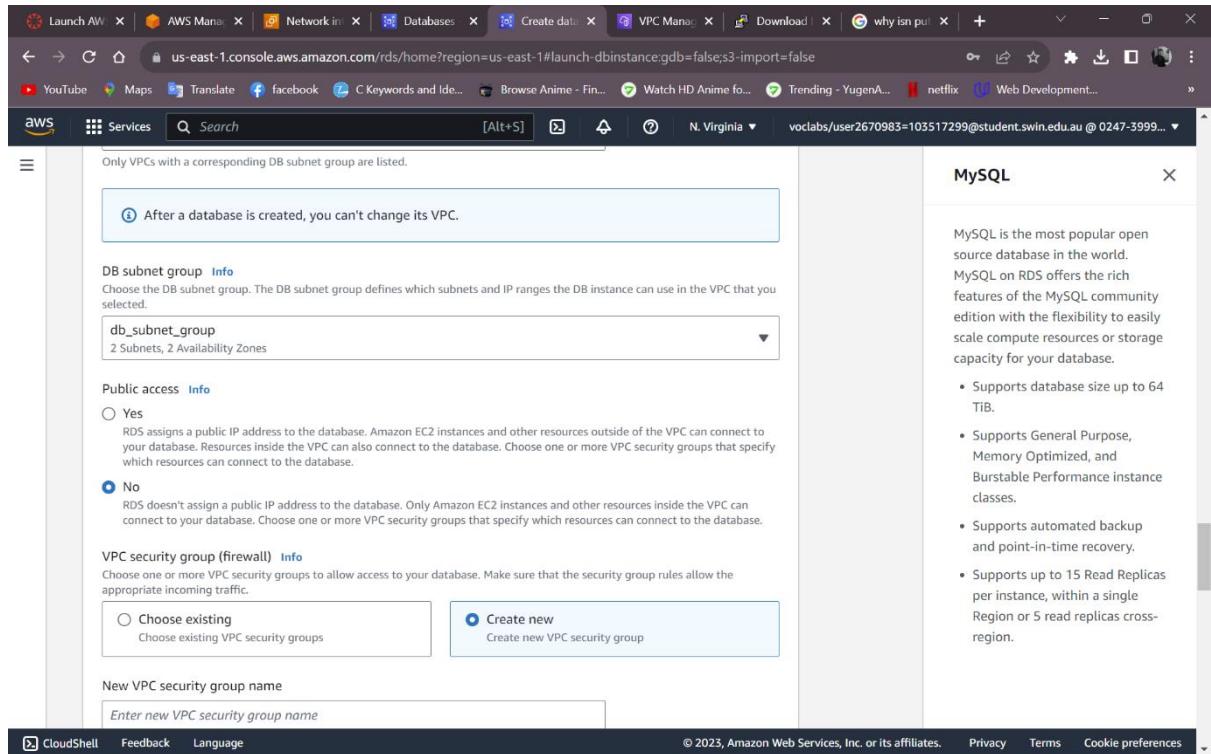


fig8: creating db instance

Summary

DB identifier database-assignment1b	CPU <div style="width: 2.63%;">2.63%</div>	Status Available	Class db.t3.micro
Role Instance	Current activity <div style="width: 0%;">0 Connections</div>	Engine MySQL Community	Region & AZ us-east-1b

Connectivity & security

Endpoint & port	Networking	Security
Endpoint database-assignment1b.cf563ujs8yt3.us-east-1.rds.amazonaws.com	Availability Zone us-east-1b	VPC security groups DBServerSG (sg-0fce21bce6de336a8) Active
Port 3306	VPC HWorthy-vpc (vpc-0e862fc18d42170a9)	Publicly accessible No
	Subnet group	Certificate authority Info

fig9:rds database instance

VPC dashboard

Virtual private cloud

Subnets

Route tables

Internet gateways

Egress-only internet gateways

Carrier gateways

DHCP option sets

Elastic IPs

Managed prefix lists

Endpoints

Endpoint services

NAT gateways

Peering connections

CloudWatch Metrics

Feedback

Language

© 2023, Amazon Web Services, Inc. or its affiliates.

fig10: downloading phpadmin

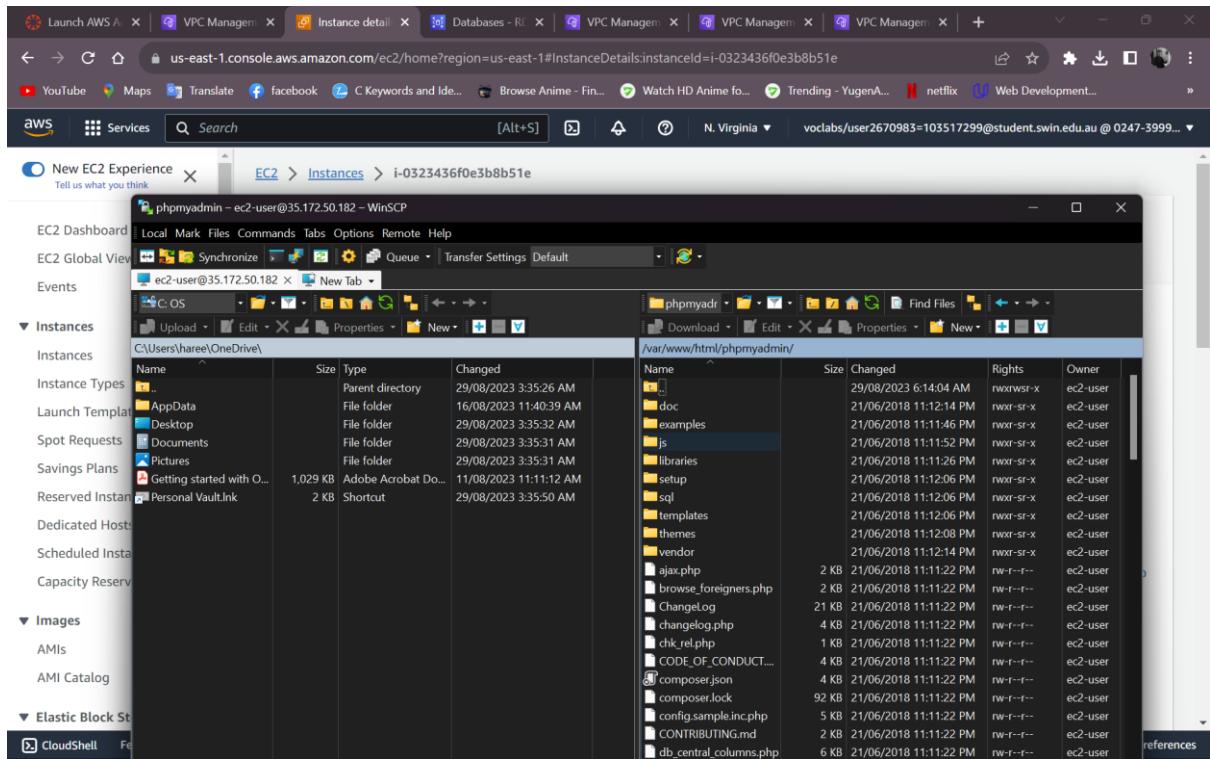
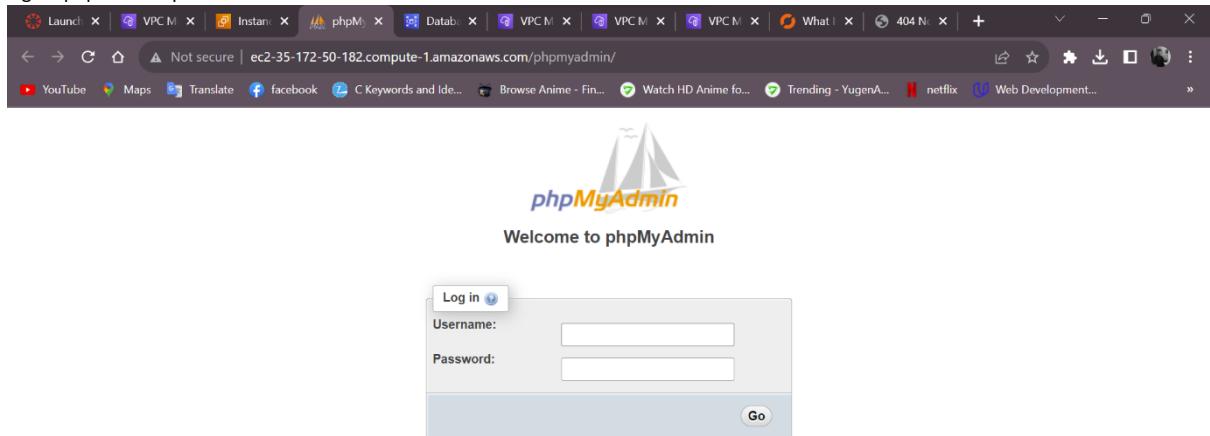


Fig11: php in winscp



ec2-35-172-50-182.compute-1.amazonaws.com/phpmyadmin/url.php?url=https%3A%2F%2Fwww.phpmyadmin.net%2F

fig12: phpmyadminlogin

Table name: Photos

Structure

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index
PhotoTitle	VARCHAR	255	None	utf8mb4_unicode_ci		NO	---
Description	VARCHAR	255	None	utf8mb4_unicode_ci		NO	---
CreationDate	DATE		None	utf8mb4_unicode_ci		NO	---
Keywords	VARCHAR	255	None	utf8mb4_unicode_ci		NO	---
ReferenceS3	VARCHAR	255	None	utf8mb4_unicode_ci		NO	---

Table comments:

Storage Engine: InnoDB

Preview SQL Save

fig:tableinphpmyadmin

5. Network ACL

I've successfully designed and deployed a Network ACL named "PublicSubnet2NACL" for Public Subnet 2. The ACL allows SSH (port 22) traffic from anywhere, permits ICMP traffic exclusively from the Test instance's subnet, and allows necessary traffic for full functionality of the Photo Album website for instance TCP (Was suggested by ChatGPT).

Details

Network ACL ID acl-0ab8dff7e24c0d4be	Associated with subnet-0886e6db98312b91 / HWorthy-subnet-private2-us-east-1b	Default No	VPC ID vpc-0e862fc18d42170a9 / HWorthy-vpc
---	---	---------------	---

Inbound rules (4)

Rule number	Type	Protocol	Port range	Source	Allow/Deny
2	SSH (22)	TCP (6)	22	0.0.0.0/0	Allow
3	All ICMP - IPv4	ICMP (1)	All	10.0.4.0/24	Allow
5	All TCP	TCP (6)	All	0.0.0.0/0	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

fig: Network ACL

6.S3 Bucket Photo Upload and Public Access:

I've created an S3 bucket(harrysalbum) to store photos, manually uploaded images, and confirmed their successful upload. I've applied an access policy to ensure all objects in the bucket are publicly accessible, achieving the desired level of public visibility for the photos.

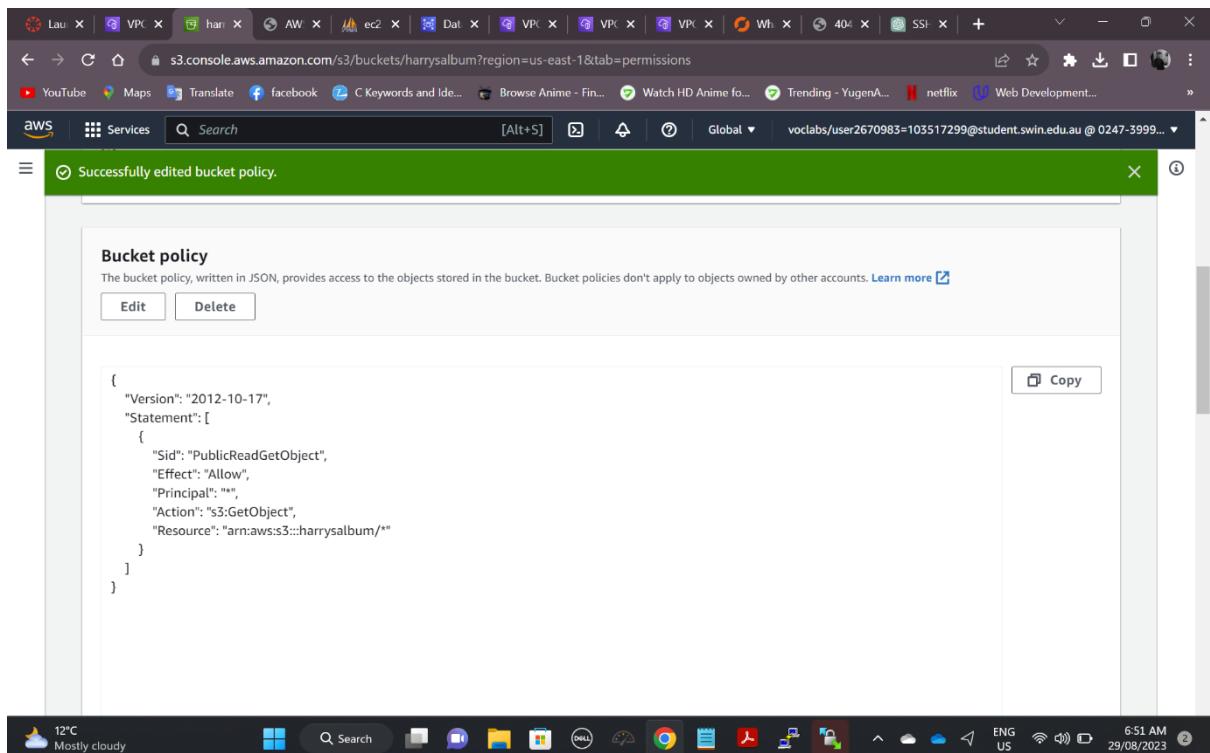


Fig:Bucketpolicy

The screenshot shows the AWS S3 console interface. At the top, there's a navigation bar with various AWS services like Lambda, VPC, S3, AWS CloudWatch, and others. Below that is a toolbar with links to YouTube, Maps, Translate, Facebook, and several trending topics. The main content area has a green header bar indicating "Upload succeeded". It shows a summary table with one row: Destination (s3://harrysalbum) with Status "Succeeded" and 3 files (197.7 KB), and Failed with 0 files (0 B). Below this is a tabbed section with "Files and folders" selected, showing a list of three files: Swinburnelogo.jpg, images.jpg, and Naruto_newshot.jpg, all of which have a status of "Succeeded". The bottom of the screen shows the Windows taskbar with icons for File Explorer, Start, Search, Task View, and other system tools, along with the date and time (29/08/2023, 7:00 AM).

fig: uploading photos in S3

7.Photo album website Functionality:

I've integrated the provided Photo Album website code to display photos and their associated metadata from the S3 bucket and RDS database. I've updated the constants.php file with the appropriate information to ensure seamless functionality.

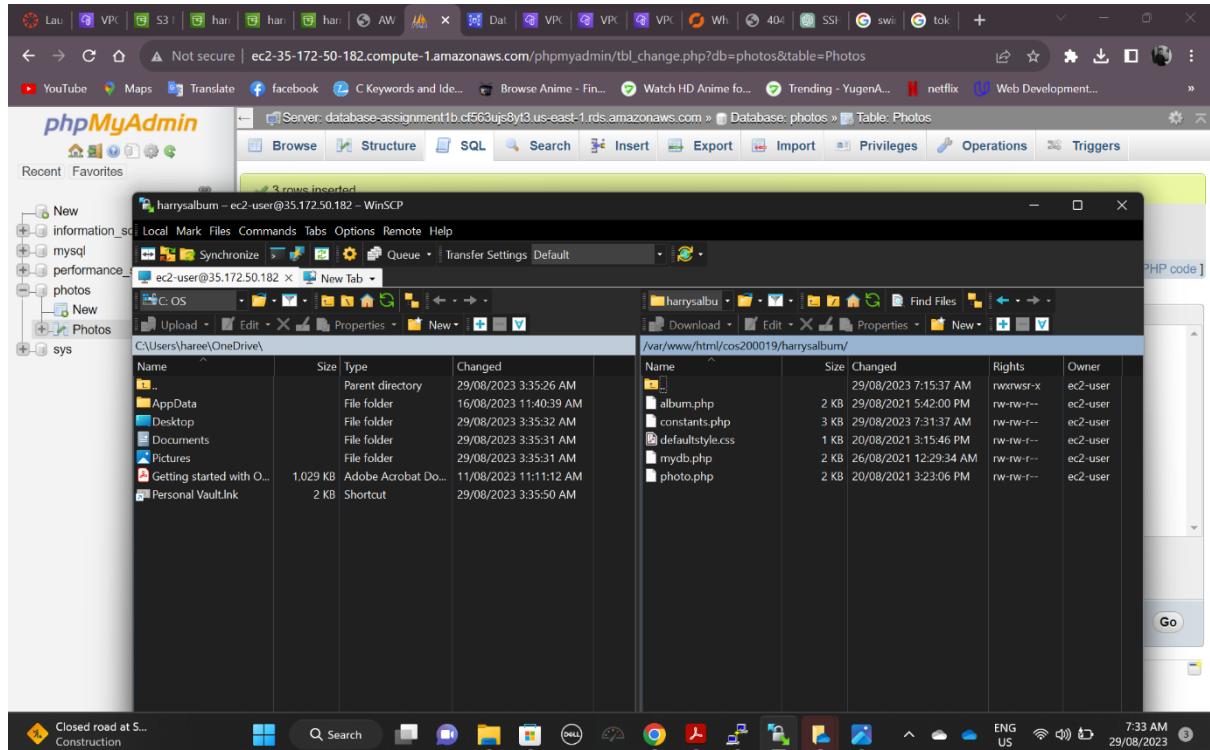


fig:photoalbumzipwinSCP

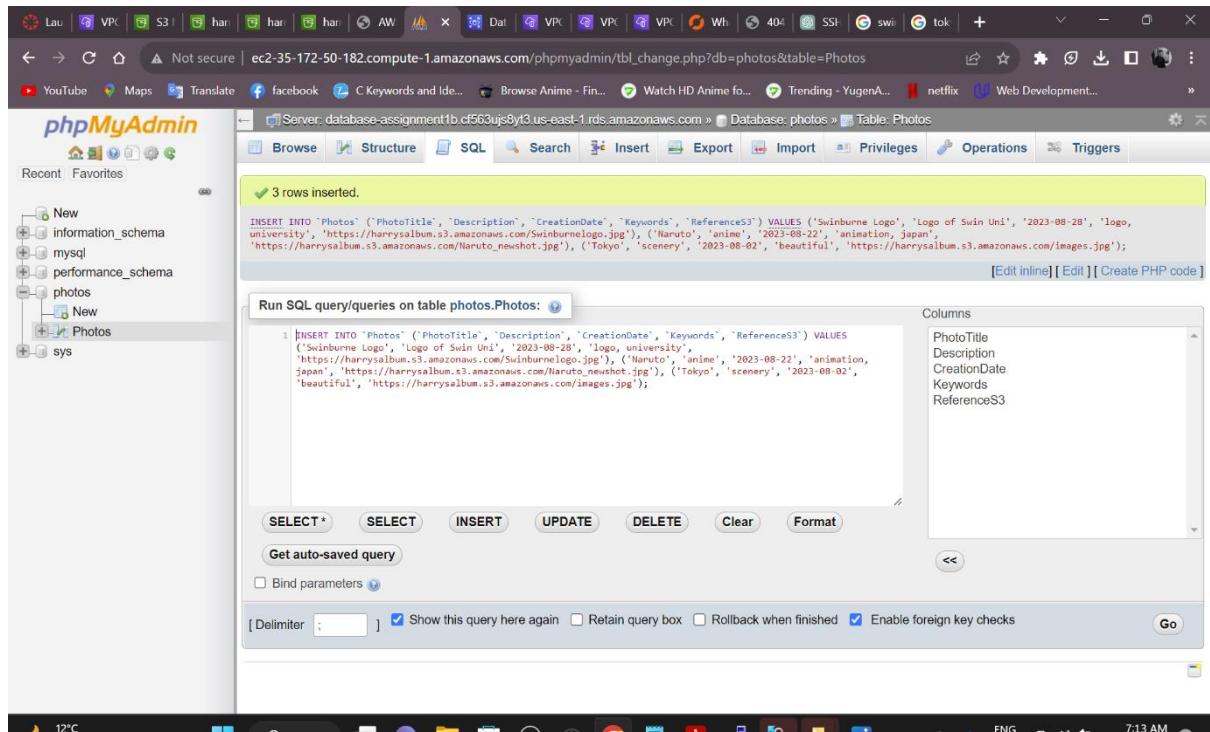
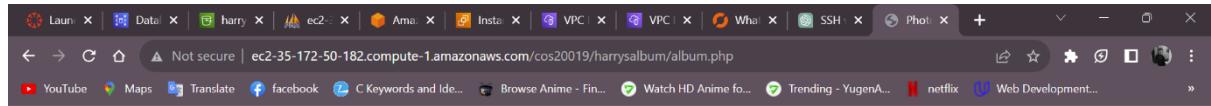


fig:metadata

Testing:



Student name: Harry

Student ID: 103517299

Tutorial session: Thursday 20:00PM

Uploaded photos:

Photo	Name	Description	Creation date	Keywords
	Swinburne Logo	Logo of Swin Uni	2023-08-28	logo, university
	Naruto	anime	2023-08-22	animation, japan
	Tokyo	scenery	2023-08-02	beautiful



fig: yay