

Swinburne University of Technology

COS20019 Cloud Computing Architecture

Assignment 2

Saturday 14th November 2023

*Phan Vu
Student ID: 104222099
104222099@student.swin.edu.au*



I. INTRODUCTION

The Photo Album website project employs EC2 web servers as the hosting infrastructure for the website, using many AWS services including S3, RDS, and Lambda. Users have the ability to upload visual images onto the digital platform, where they may then engage in the exploration of various collections including these photographs. Furthermore, the website provides the functionality of automatically producing and resizing thumbnail photos. The comprehensive source code and accompanying instructions have been supplied to assist the seamless integration of the website with the S3 bucket, RDS database, and Lambda function. This project utilizes the capabilities of Amazon Web Services (AWS) to create a dynamic and user-centric platform for image albums. It incorporates many functionalities, including photo retrieval, uploading, and thumbnail generation.

II. WEBSITE ARCHITECTURE

A. Infrastructure Requirements

- 1) **Virtual Private Cloud (VPC):** VPC configured with 2AZs both with public and private subnets. Public and private route tables route to IGW and NAT, respectively.

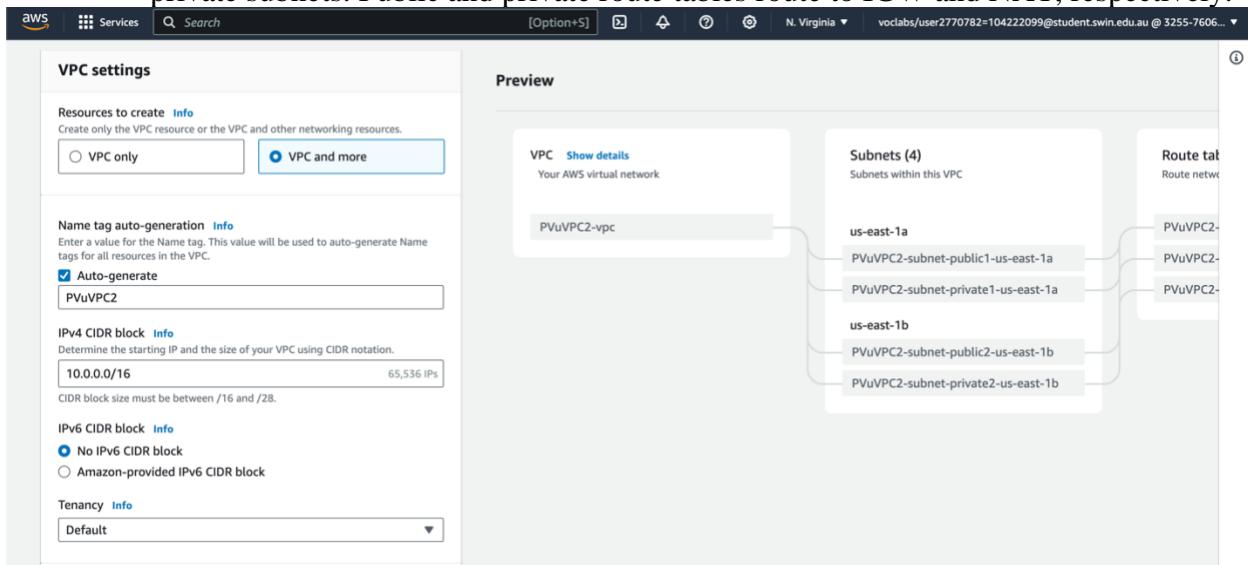


Figure 1. Configuring VPC

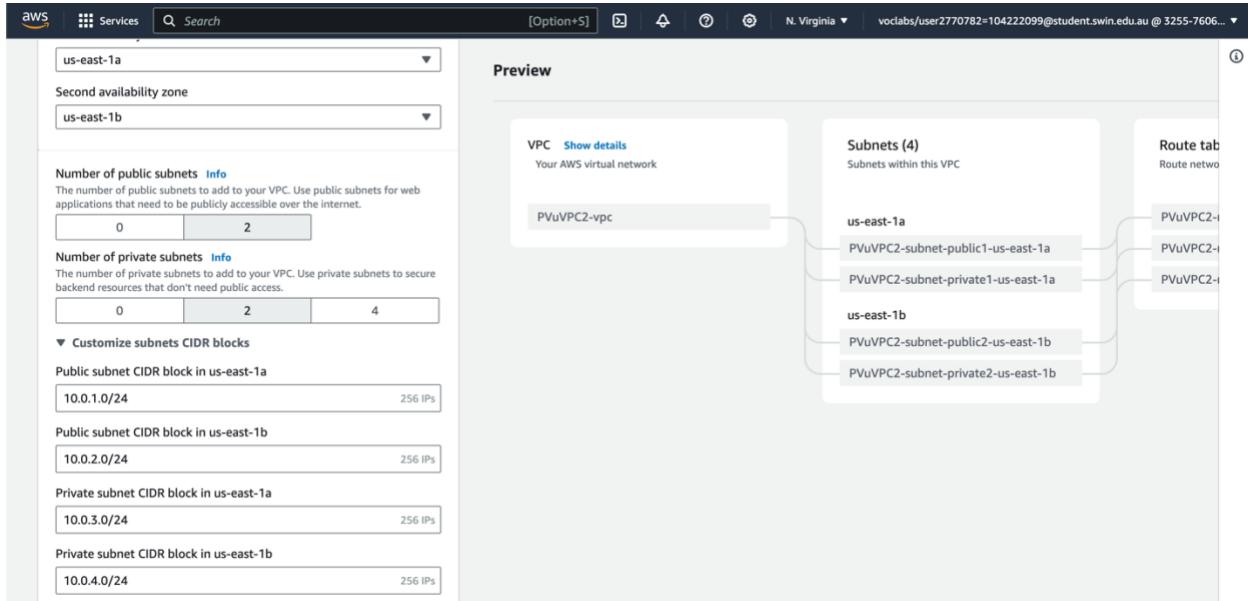


Figure 2. VPC Subnet CIDR Block

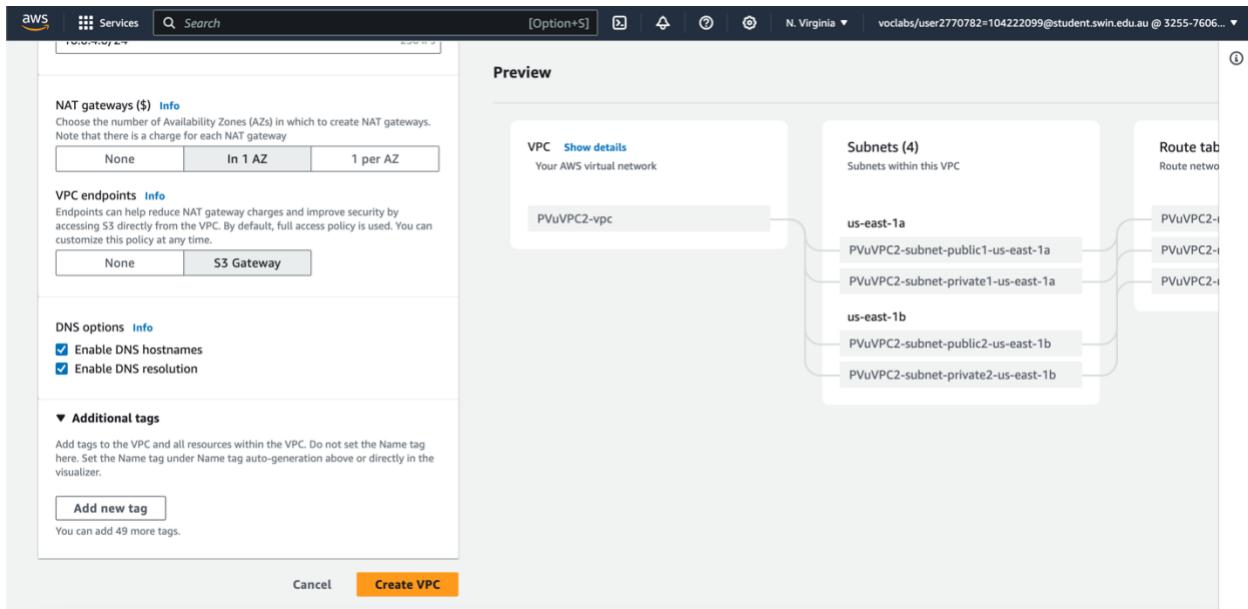


Figure 3. Configuring VPC

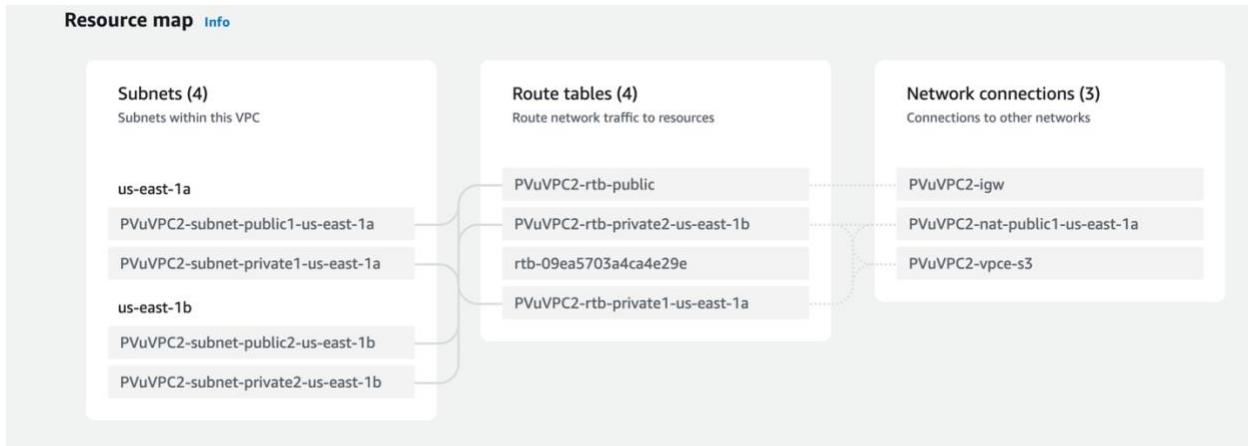


Figure 4. Resource Map

The VPC is ready for the photo album web server to be launched.

- 2) **Security Group:** The architecture has five security groups: ELBSG, WebServerSG, DBServerSG, DevServerSG, and NATServerSG. Since NAT gateway is utilized instead of NAT instance, NATServerSG is unnecessary. All security group outbound rule defaults to IPv4 traffic everywhere.

The screenshot shows the AWS Security Groups interface for the ELBSG. The 'Details' section displays the security group name (ELBSG), ID (sg-058f495c53bc58bbc), description ('for the ELB created'), and VPC ID (vpc-0e1e07996ff5477c). The 'Inbound rules' tab is selected, showing two rules: one for port 443 (HTTPS) and another for port 80 (HTTP).

Name	Security group rule...	IP version	Type	Protocol	Port range	Source
-	sgr-0c8910ddc255bc1c0	IPv4	HTTPS	TCP	443	0.0.0.0/0
-	sgr-03636a24c85a8f0b5	IPv4	HTTP	TCP	80	0.0.0.0/0

Figure 5. ELBSG for the ELB created above.

Details

Security group name WebServerSG	Security group ID sg-0fad8a97cfcbbed8f7	Description for all the web servers in private subnets	VPC ID vpc-0e1e07996ffb5477c
Owner 325576069040	Inbound rules count 4 Permission entries	Outbound rules count 1 Permission entry	

Inbound rules | Outbound rules | Tags

Inbound rules (4)

Name	Type	Protocol	Port range	Source
sgr-085a174a668456c...	HTTPS	TCP	443	sg-058f49
sgr-0ae8eb18ae58accf7	SSH	TCP	22	sg-056b01
sgr-0e4bd1cc2ffee7298	MySQL/Aurora	TCP	3306	sg-011f50
sgr-07c3cfea7d61bd263	HTTP	TCP	80	sg-058f49

Figure 6. WebServerSG for all the web servers in private subnets.

Details

Security group name DevServerSG	Security group ID sg-056b0189bb9aeeef2b	Description for the Dev server	VPC ID vpc-0e1e07996ffb5477c
Owner 325576069040	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

Inbound rules | Outbound rules | Tags

Inbound rules (1/1)

Name	Type	Protocol	Port range	Source
sgr-07d30170a51e18...	SSH	TCP	22	0.0.0.0/0

Figure 7. DevServerSG for the Dev server.

Details

Security group name DBServerSG	Security group ID sg-011f50f31f0802325	Description for the RDS instance	VPC ID vpc-0e1e07996ffb5477c
Owner 325576069040	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

Inbound rules | Outbound rules | Tags

Inbound rules (1/1)

Name	Type	Protocol	Port range	Source
sgr-03b1e8e9129ddf784	MySQL/Aurora	TCP	3306	sg-0fad8a97cfcbbed8f7

Figure 8. DBServerSG for the RDS instance.

Security group name	Protocols	Source
WebServerSG	HTTP (80), HTTPS (443)	ELBSG
	SSH (22)	DevServerSG
	MYSQL/Aurora (3306)	DBServerSG
DBServerSG	MYSQL/Aurora (3306)	WebServerSG
DevServerSG	SSH (22)	Anywhere-IPv4
ELBSG	HTTP (80), HTTPS (443)	Anywhere-IPv4

Figure 9. Inbound security rules summary

- 3) **Network ACLs (NACL):** The “PrivateSubnetsNACL” NACL was established to enhance web server security on private subnets. The NACL limits ICMP communication to and from the DevServer in both directions.

The screenshot shows the AWS VPC Network ACLs interface. The URL is [VPC > Network ACLs > acl-082a9ded6a1c67345 / PrivateSubnetsNACL > Edit subnet associations](#). The page title is "Edit subnet associations". It displays two sections: "Available subnets (2/4)" and "Selected subnets". Under "Available subnets", there are four subnets listed with checkboxes. The third and fourth subnets have checkboxes checked. Under "Selected subnets", two subnets are listed: "subnet-06d816e998233b2e9 / PVuVPC2-subnet-private1-us-east-1a" and "subnet-057a01944778361f6 / PVuVPC2-subnet-private2-us-east-1b". At the bottom right are "Cancel" and "Save changes" buttons.

Figure 10. NACL associated with private subnets in the VPC

The screenshot shows the AWS VPC Network ACLs interface. The URL is [VPC > Network ACLs > acl-082a9ded6a1c67345 / PrivateSubnetsNACL > Edit inbound rules](#). The page title is "Edit inbound rules". It displays a table of inbound rules with columns: Rule number, Type, Protocol, Port range, Source, and Allow/Deny. There are three rules: rule 1 (All ICMP - IPv4, ICMP (1), All, 10.0.2.0/24, Deny), rule 2 (All traffic, All, All, 0.0.0.0/0, Allow), and a wildcard rule (*, All traffic, All, 0.0.0.0/0, Deny). At the bottom right are "Cancel", "Preview changes", and "Save changes" buttons.

Figure 11. Inbound rules of PrivateSubnetsNACL

The screenshot shows the AWS VPC Network ACLs interface. The URL is [VPC > Network ACLs > acl-082a9ded6a1c67345 / PrivateSubnetsNACL > Edit outbound rules](#). The page title is "Edit outbound rules". It displays a table of outbound rules with columns: Rule number, Type, Protocol, Port range, Destination, and Allow/Deny. There are three rules: rule 1 (All ICMP - IPv4, ICMP (1), All, 10.0.2.0/24, Deny), rule 2 (All traffic, All, All, 0.0.0.0/0, Allow), and a wildcard rule (*, All traffic, All, 0.0.0.0/0, Deny). At the bottom right are "Cancel", "Preview changes", and "Save changes" buttons.

Figure 12. Outbound rules of PrivateSubnetsNACL

```
[ec2-user@ip-10-0-2-99 ~]$ ping 10.0.3.78
PING 10.0.3.78 (10.0.3.78) 56(84) bytes of data.
^C
--- 10.0.3.78 ping statistics ---
162 packets transmitted, 0 received, 100% packet loss, time 164868ms

[ec2-user@ip-10-0-2-99 ~]$ ping 10.0.4.26
PING 10.0.4.26 (10.0.4.26) 56(84) bytes of data.
^C
--- 10.0.4.26 ping statistics ---
6 packets transmitted, 0 received, 100% packet loss, time 5126ms
```

Figure 13. Testing the NACL (ping from DevServer to WebInstance ASG)

```
[ec2-user@ip-10-0-4-26 ~]$ ping 10.0.2.99
PING 10.0.2.99 (10.0.2.99) 56(84) bytes of data.
^C
--- 10.0.2.99 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2051ms
```

Figure 14. Testing NACL (ping from WebInstance ASG to DevServer)

```
[ec2-user@ip-10-0-3-78 ~]$ ping 10.0.2.99
PING 10.0.2.99 (10.0.2.99) 56(84) bytes of data.
^C
--- 10.0.2.99 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4075ms
```

Figure 15. Testing NACL (ping from WebInstance ASG to DevServer)

- 4) **IAM Role:** The management console contains IAM roles such as “LabRole” and “Labinstancerole” with essential permissions for this assignment.

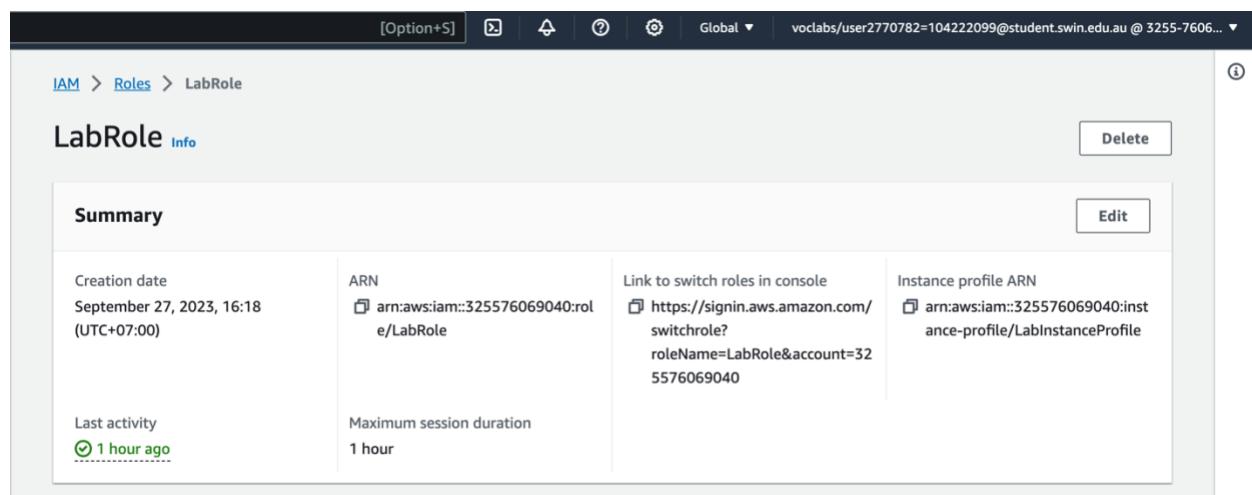


Figure 16. IAM role

- 5) **CreateThumbnail Lambda function:** The IAM execution role was assigned to the Lambda function to ensure appropriate security measures and permissions. The IAM role named LabRole, which adheres to the concept of least privilege, grants the Lambda function the necessary access and control over the items within the designated S3 bucket.

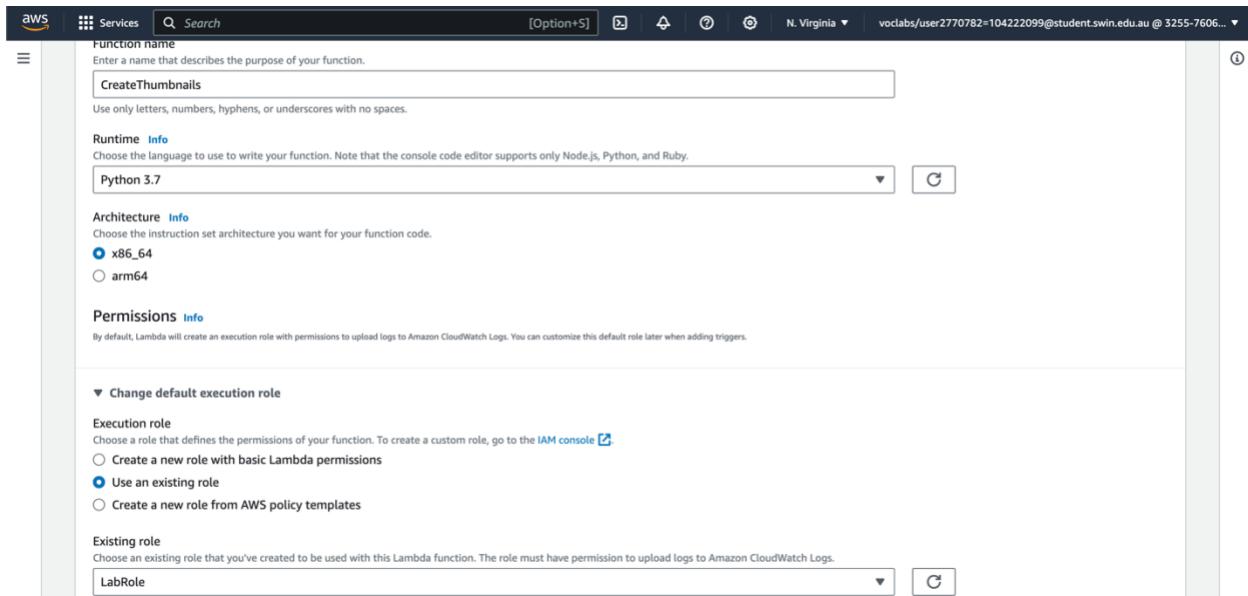


Figure 17. CreateThumbnail Lambda function configuration

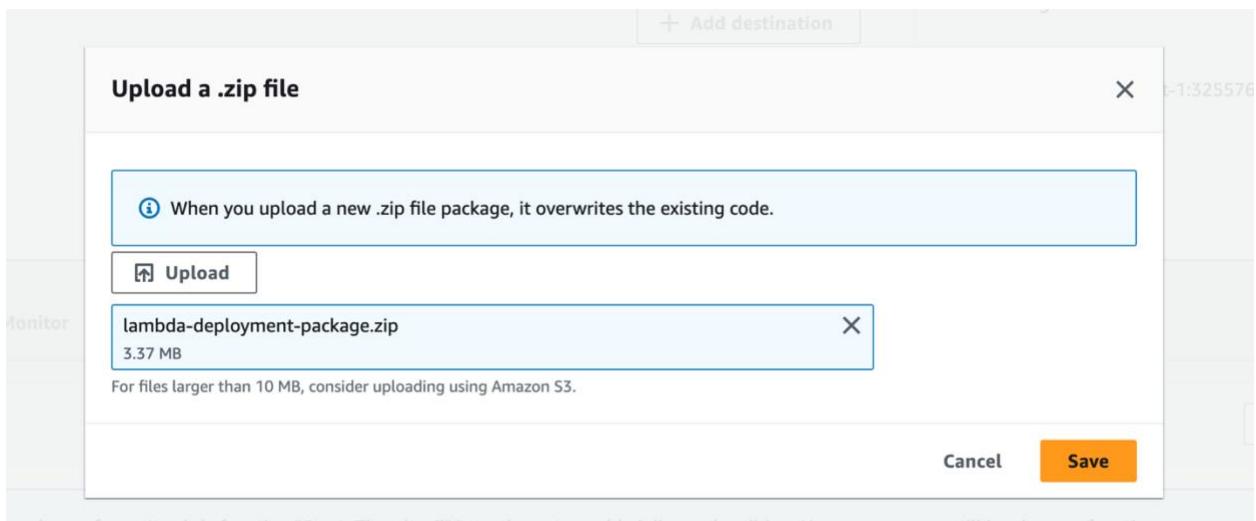


Figure 18. CreateThumbnail Lambda function configuration

To grant the web server the necessary permissions, an IAM role entitled LabInstanceProfile was created using the principle of least privilege. The role was created to grant the selected S3 bucket Web Server rights for adding objects and calling the CreateThumbnail Lambda function.

- 6) **Auto Scaling Group (ASG):** To create ASG, we need to create a DevServer instance first, which is used to develop the custom AMI for the web server and make a launch template later

DevServer Instance: Since it only develops the specific AMI needed to run PhotoAlbum, DevServer does not get traffic from ELB. The custom AMI includes the AWS PHP SDK, Apache web server, and website source code. The DevServer also manages MySQL RDS using phpMyAdmin.

Instance summary for i-0bdc6974319258b5a (DevServer) [Info](#)

Updated less than a minute ago

Instance ID	i-0bdc6974319258b5a (DevServer)	Public IPv4 address	3.223.2.187 [open address]	Private IPv4 addresses	10.0.2.99
IPv6 address	-	Instance state	Running	Public IPv4 DNS	ec2-3-223-2-187.compute-1.amazonaws.com [open address]
Hostname type	IP name: ip-10-0-2-99.ec2.internal	Private IP DNS name (IPv4 only)	ip-10-0-2-99.ec2.internal	Elastic IP addresses	3.223.2.187 [Public IP]
Answer private resource DNS name	-	Instance type	t2.micro	AWS Compute Optimizer finding	Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address	-	VPC ID	vpc-0e1e07996ffb5477c (PVuVPC2-vpc)	Auto Scaling Group name	-
IAM Role	LabRole	Subnet ID	subnet-080982f479d334033 (PVuVPC2-subnet-public2-us-east-1b)		
IMDSv2	Optional				

Figure 19. DevServer instance resides in Public Subnet 2 (CIDR: 10.0.2.0/24) with an Elastic IP associated

DevServer has t2-micro instance type, Amazon Linux 2 AMI (HVM), SSD Volume Type, and bash script-installed Apache Web Server in assignment 1a. AWS Learner Lab already has LabRole IAM Role.

EC2 > Instances > i-0477063fd41ea4d71 > Modify IAM role

Modify IAM role [Info](#)

Attach an IAM role to your instance.

Instance ID	i-0477063fd41ea4d71 (DevServer)
IAM role	Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to your instance.
	LabInstanceProfile Create new IAM role

[Cancel](#) [Update IAM role](#)

Figure 20. DevServer IAM Role

The DevServer is associated with an Elastic IP to allow SSH connection to manage the DevServer directory.

ec2-3-223-2-187.co... [Open Connection](#) [Search](#) Unregistered

ec2-user@ec2-3-223-2-1...

Filemanager

Filename Size Modified

- var -- Today, 10:53
- yp -- 10/04/2019, 02:57
- www -- Today, 10:53
 - html -- Today, 14:25
 - phpmyadmin -- Today, 13:16
 - phpinfo.php 91 B Today, 10:53
 - photoalbum -- Today, 11:20
 - aws -- Today, 14:15
 - cgi-bin -- 27/10/2023, 03:09
 - tmp -- Today, 12:07

Figure 21. DevServer Directory Structure with necessary SDKs and components included

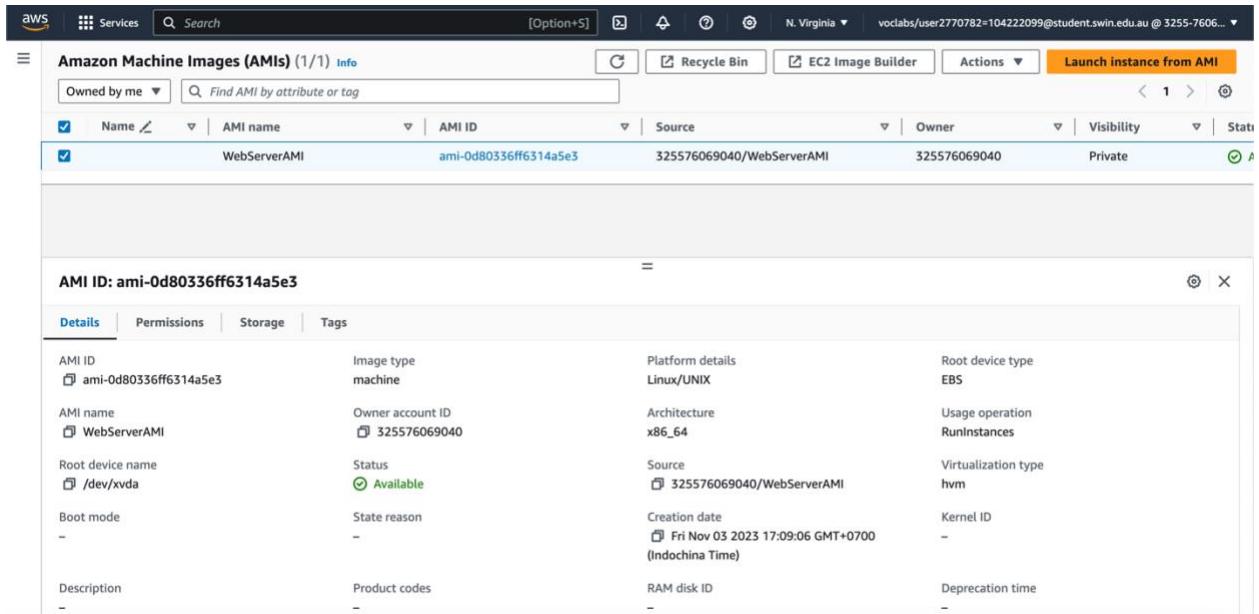


Figure 22. DevServer image created, ready to use

Launch template: Create a launch template with instance type t2.micro and IAM role LabInstanceProfile using the AMI created.

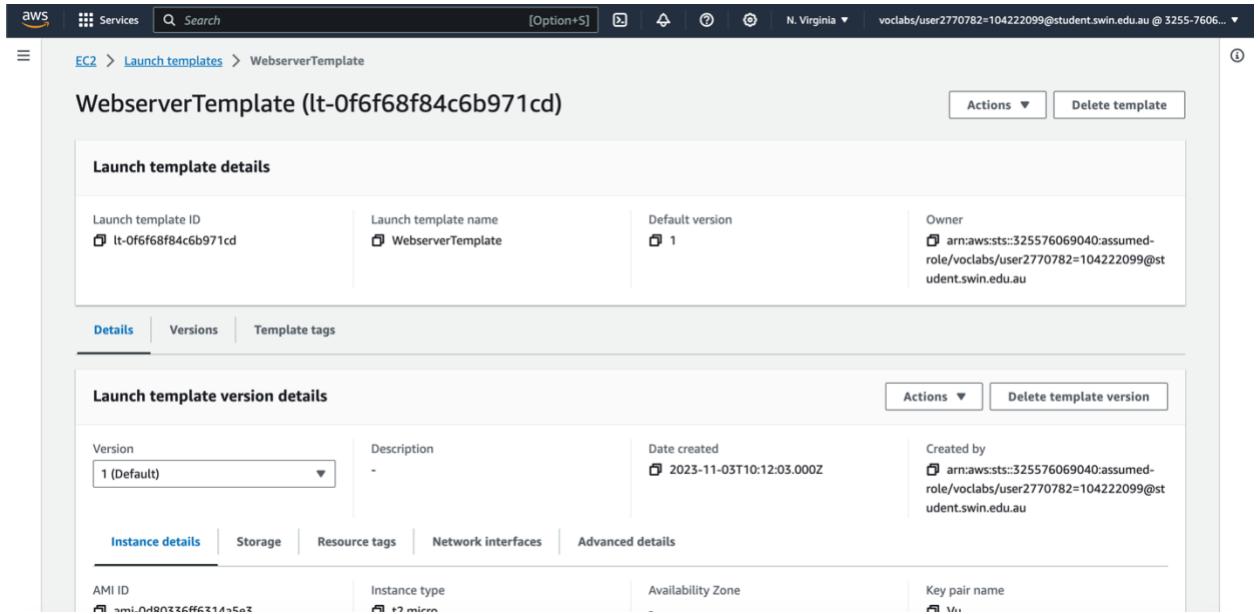


Figure 23. Launch template for ASG

Auto Scaling Group: The ASG is configured to launch instances exclusively into the private subnets, with a minimum of two and a maximum of three instances maintained, with two instances being the optimal number. This guarantees that the application maintains a minimum number of available instances while restricting the infrastructure's scalability to the specified maximum.

Group details

Auto Scaling group name WebServerASG	Desired capacity 2	Status -	Amazon Resource Name (ARN) arn:aws:autoscaling:us-east-1:325576069040:autoScalingGroup:6753e746-5d44-426d-8cb8-4b744b2251fb:autoScalingGroupName/WebServerASG
Date created Fri Nov 03 2023 17:13:44 GMT+0700 (Indochina Time)	Minimum capacity 2	Maximum capacity 3	

Launch template

Launch template lt-0f6f68f84c6b971cd WebserverTemplate	AMI ID ami-0d80336ff6314a5e3	Instance type t2.micro	Owner arn:aws:sts::325576069040:assumed-role/voclabs/user2770782=104222099@student.swin.edu.au
Version Default	Security groups -	Security group IDs sg-0fad8a97cfcbbed87	Create time Fri Nov 03 2023 17:12:03 GMT+0700

Figure 24. ASG basic configuration

Instance type requirements

Your Auto Scaling group adheres to the launch template for purchase option and instance type.

Load balancing

Load balancer target groups WebServerTG	Classic Load Balancers -
--	-----------------------------

VPC Lattice integration options

VPC Lattice target groups -

Health checks

Health check type EC2	Health check grace period 90
--------------------------	---------------------------------

Figure 25. ASG health check configuration with target group attached

Dynamic scaling policies (1) [Info](#)

Target Tracking Policy	<input type="checkbox"/>
Target tracking scaling	
Enabled	
As required to maintain Average CPU utilization at 30	
Add or remove capacity units as required	
300 seconds to warm up before including in metric	
Enabled	

Figure 26. Target tracking policy based on application load balancer request

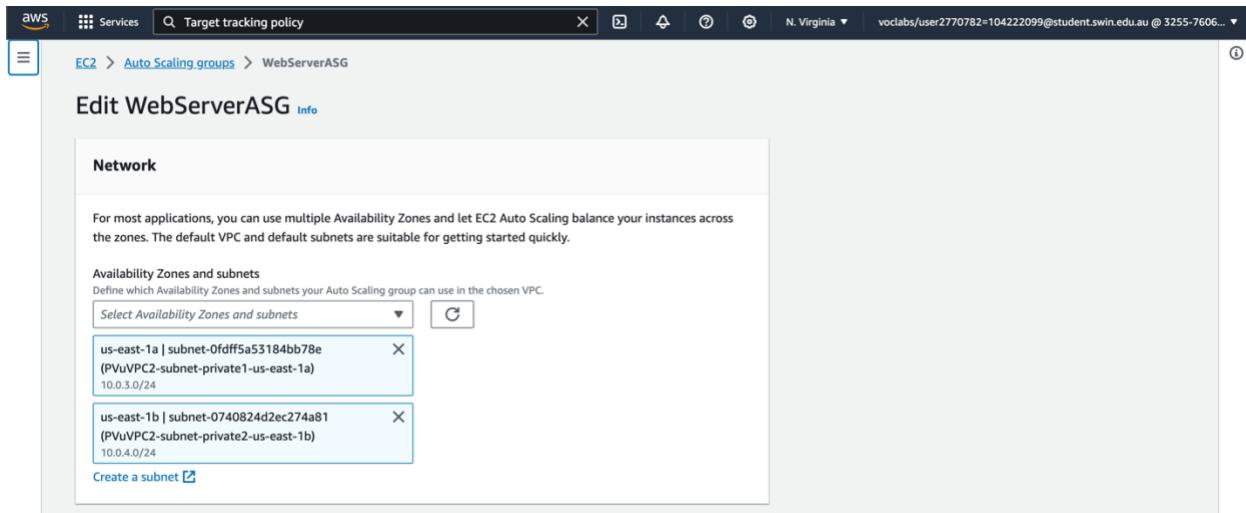


Figure 27. ASG network mapped to private subnets

Target tracking scaling was used to limit instances based on ELB target group requests. The policy sets a target request count of 30. The auto-scaling group automatically scales instances up or down to maintain the required request count per target, optimizing performance and resource use.

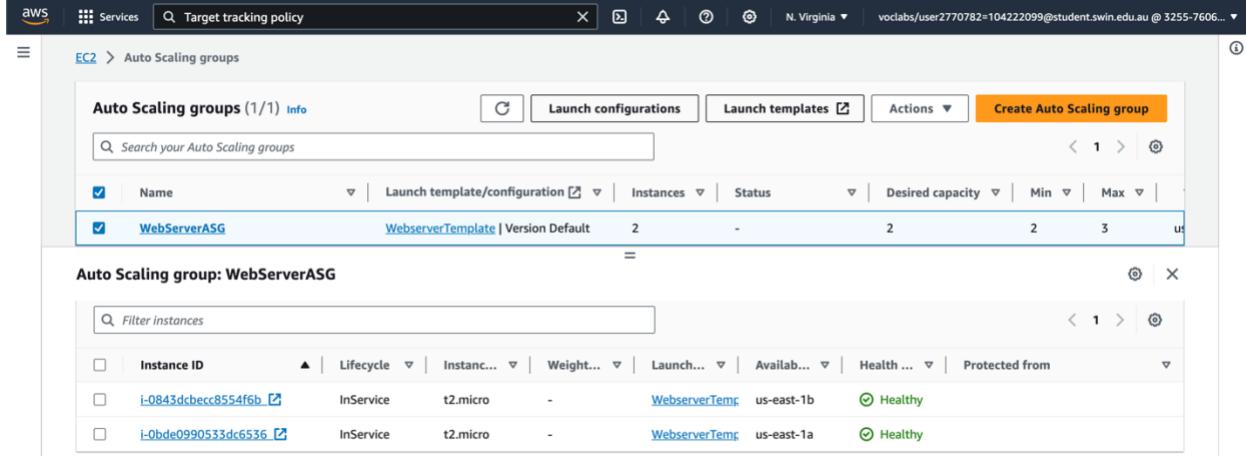


Figure 28. EC2 instances are properly distributed across two private subnets with healthy states

The web server may now adjust its capacity based on request load, maintaining a consistent number of requests per target and optimizing resource use.

- 7) **Elastic Load Balancing (ELB):** First, create a new target group as the load balancer needs to route requests to the targets in a target group and perform health checks on the targets.

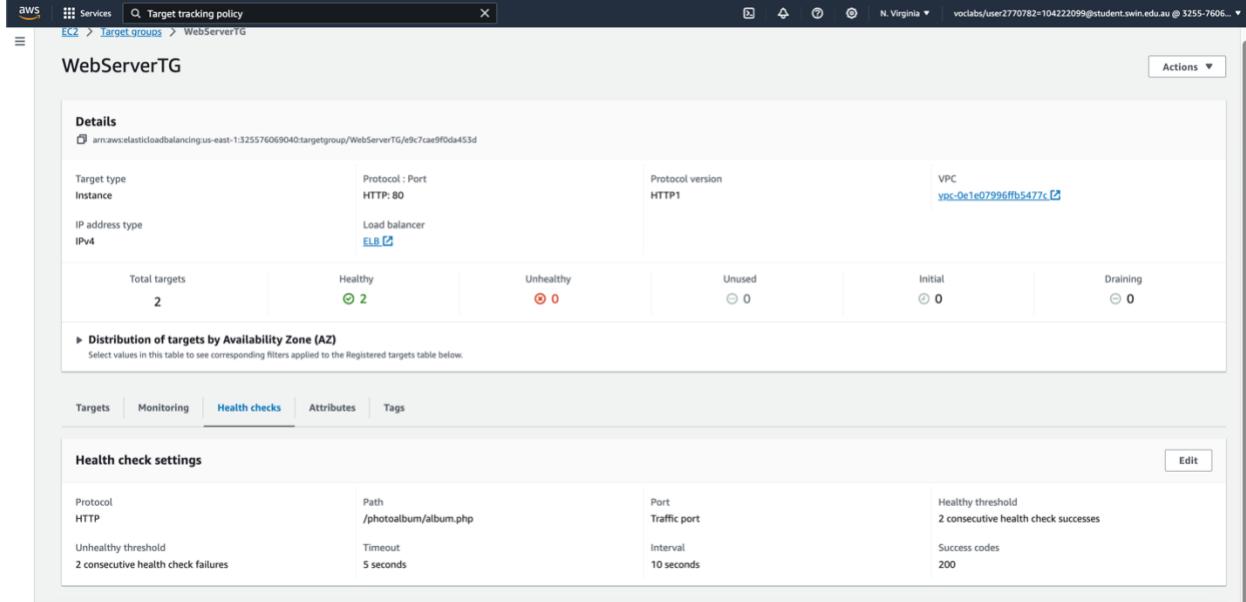


Figure 29. Target group configuration, with health check path set to /photoalbum/album.php

Create a new load balancer and attach it to the target group.

The screenshot shows the AWS CloudFormation console with a search bar at the top. Below the search bar, there are tabs for 'Metrics' and 'Logs'. A large orange button labeled 'Create CloudWatch Metrics Insights' is prominently displayed. To its right, there is a 'Next Step' button. The main area shows a table with columns for 'Name', 'Type', 'Status', and 'Last updated'. There is one item listed: 'CloudWatch Metrics Insights' (Type: Metrics Insights, Status: Pending creation, Last updated: 2023-10-03 10:20:24 UTC).

Figure 30. Application Load Balancer mapped to Public Subnet 1 and Public Subnet 2

The screenshot shows the AWS CloudFormation console with a search bar at the top. Below the search bar, there are tabs for 'Metrics' and 'Logs'. A large orange button labeled 'Create CloudWatch Metrics Insights' is prominently displayed. To its right, there is a 'Next Step' button. The main area shows a table with columns for 'Name', 'Type', and 'Status'. There is one item listed: 'CloudWatch Metrics Insights' (Type: Metrics Insights, Status: Pending creation, Last updated: 2023-10-03 10:20:24 UTC).

Figure 31. ELB Listener check rule which forwarded to the target group created in Figure 26

Now, The ELB can distribute incoming HTTP and HTTPS traffic across multiple EC2 targets.

- 8) **Simple Storage Service (S3):** Assignment 1b's S3 bucket for photos is similar. This S3 bucket has permissions and policies to make objects accessible. These steps set photo access permissions correctly.

The screenshot shows the AWS CloudFormation console with a search bar at the top. Below the search bar, there are tabs for 'Metrics' and 'Logs'. A large orange button labeled 'Create CloudWatch Metrics Insights' is prominently displayed. To its right, there is a 'Next Step' button. The main area shows a table with columns for 'Name', 'Type', and 'Status'. There is one item listed: 'CloudWatch Metrics Insights' (Type: Metrics Insights, Status: Pending creation, Last updated: 2023-10-03 10:20:24 UTC).

Figure 32. Properties of S3 bucket

```

1▼ {
2  "Version": "2012-10-17",
3  "Id": "assignmentbucket2",
4  "Statement": [
5    {
6      "Sid": "PublicRead",
7      "Effect": "Allow",
8      "Principal": "*",
9      "Action": [
10        "s3:GetObject",
11        "s3:GetBucketLocation",
12        "s3>ListBucket",
13        "s3:PutObject"
14      ],
15      "Resource": [
16        "arn:aws:s3:::pvuassignment2/*",
17        "arn:aws:s3:::pvuassignment2"
18      ],
19      "Condition": {
20        "StringLike": {
21          "aws:Referer": [
22            "http://elb-1943824096.us-east-1.elb.amazonaws.com/*",
23            "http://ec2-3-223-2-187.compute-1.amazonaws.com/*"
24          ]
25        }
26      }
27    }
28  ]
29 }

```

Figure 33. S3 policy to restrict access to a specific HTTP referer from Dev Server and Elastic Load Balancer

This policy limits access to the S3 bucket to GET requests from specific domains, ensuring restricted and secure access to the bucket's objects.

- 9) **Lambda Function:** A Lambda function named “CreateThumbnail” was created using Python 3.7 as the runtime environment.

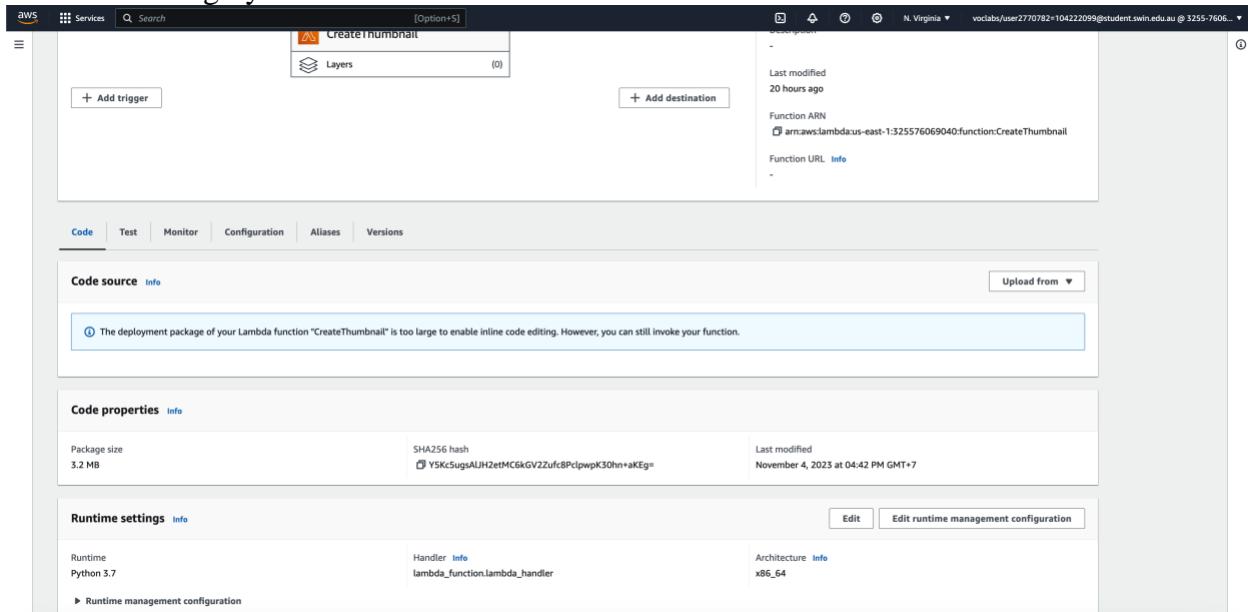


Figure 34. CreateThumbnail Lambda function configuration

Uploading “lambda-deployment-package.zip”. This package includes picture resizing and S3 bucket download/upload libraries and source code.

- 10) **Relational Database Service (RDS):** The RDS instance used in this assignment is configured like the previous one.

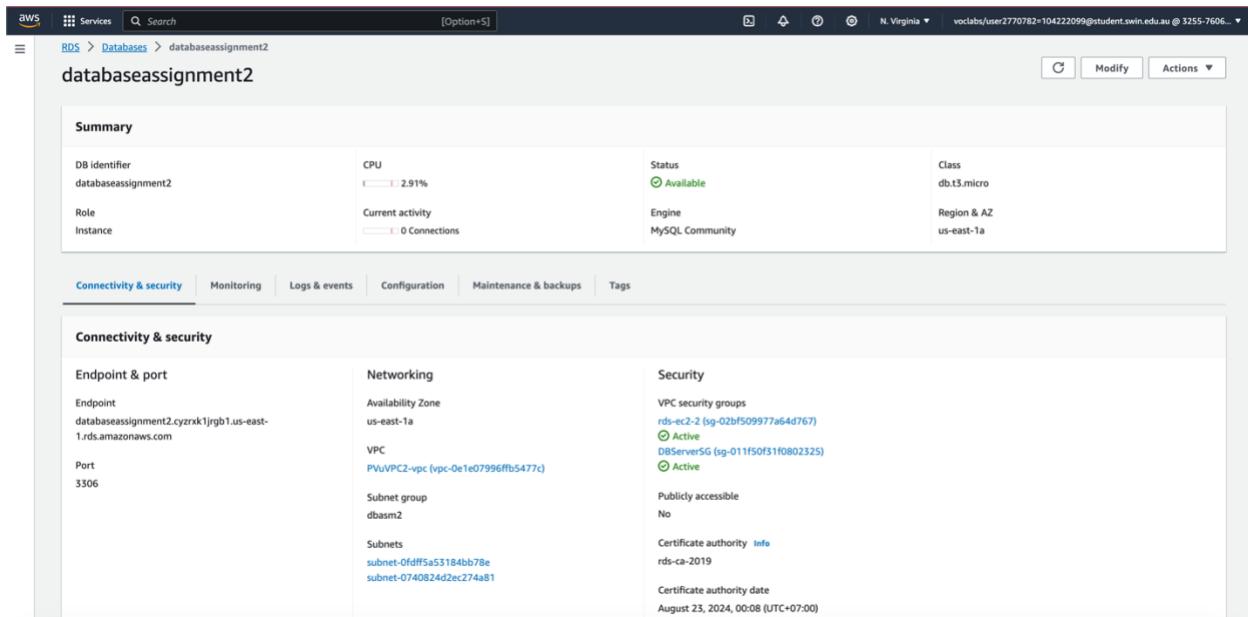


Figure 35. RDS instance

Configuration:

- Template: Free-tier
- Database engine: MySQL Community 8.0.28
- Public access set to No.
- Use DBServerSG for VPC security group
- AZ set to us-east-1a (According to the provided diagram).
- The RDS instance is associated with a subnet group “dbasm2” comprising private subnets in both AZs.

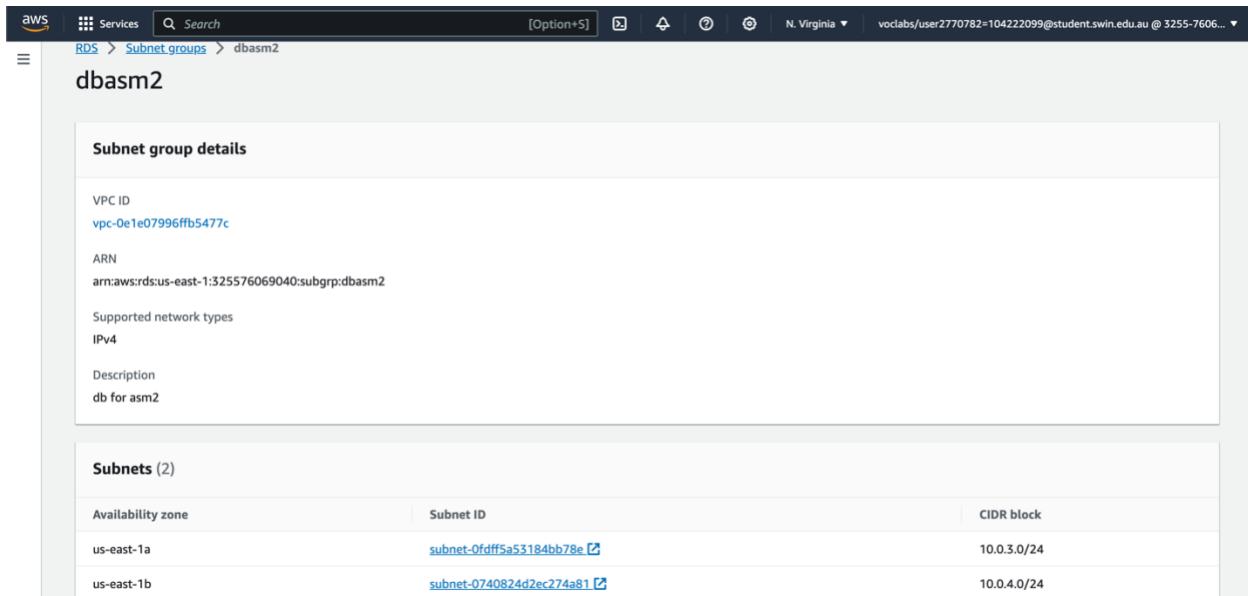


Figure 36. Subnet dbasm2 with Private subnet 3 and Private subnet 4

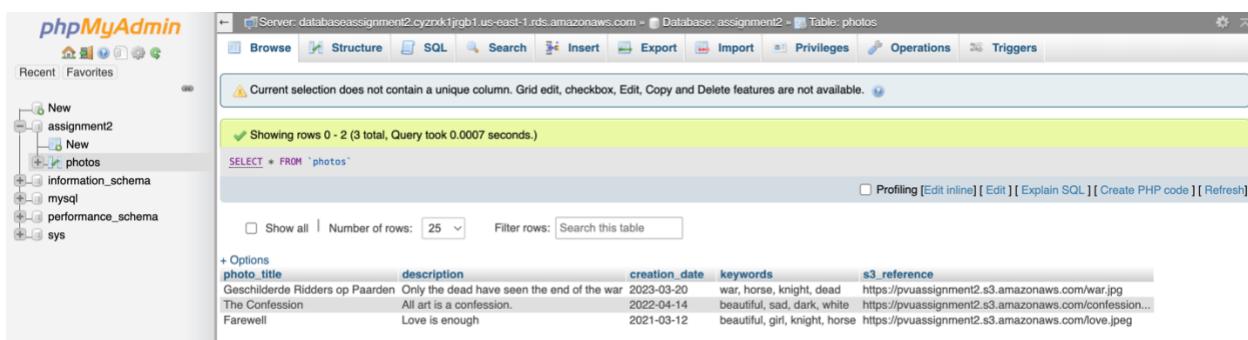


Figure 37. Data records of the database

B. Functional requirements

- 1) **Website accessibility:** To access the PhotoAlbum website, use the URL: <http://elb-1943824096.us-east-1.elb.amazonaws.com/photoalbum/album.php>. It allows you to view and interact with the PhotoAlbum web application. Additionally, to upload photos and their associated metadata, utilize the PhotoUploader web page at <http://elb-1943824096.us-east-1.elb.amazonaws.com/photoalbum/photouploader.php>. Using this page, multiple photos and their corresponding metadata can be easily uploaded to enhance the functionality of the PhotoAlbum website.



Figure 38. Website accessible through ELB DNS

2) Photo display function:

Photo	Name	Description	Creation date	Keywords
	Geschilderde Ridders op Paarden	Only the dead have seen the end of the war	2023-03-20	war, horse, knight, dead
	The Confession	All art is a confession.	2022-04-14	beautiful, sad, dark, white
	Farewell	Love is enough	2021-03-12	beautiful, girl, knight, horse

Figure 39. Photo display function

The photo display function is working correctly.

3) Photo uploader function:

Figure 40. Photo uploader function

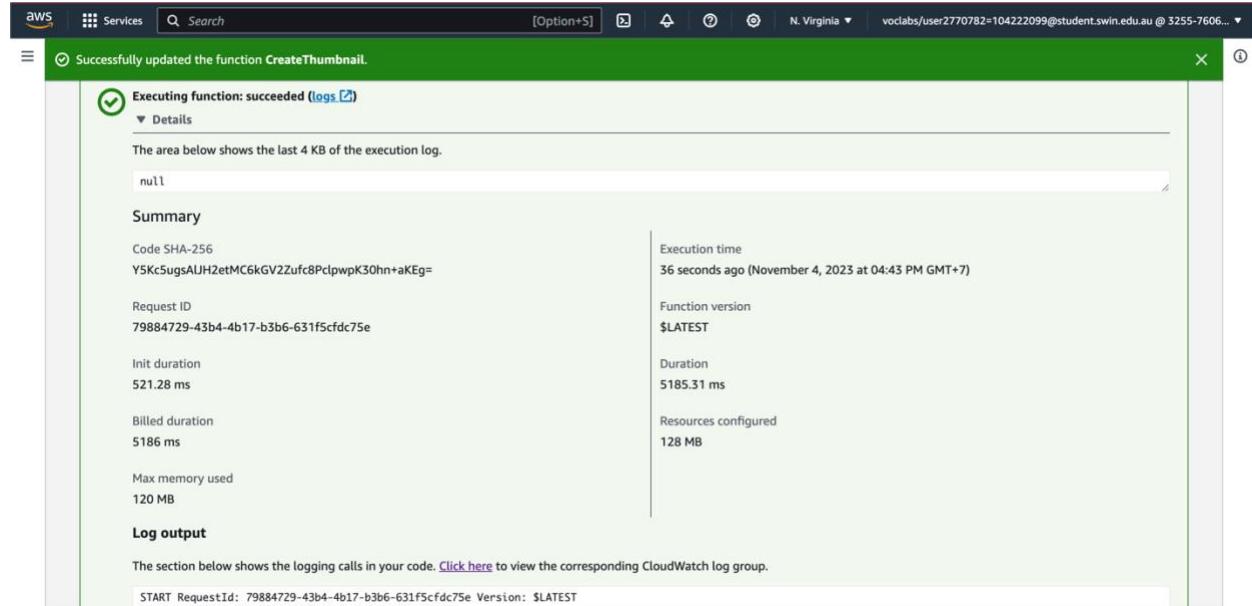
Student name: Phan Vu
 Student ID: 104222099
 Tutorial session: Saturday 01:00PM

Uploaded photos:[Upload more photos](#)

Photo	Name	Description	Creation date	Keywords
	Geschildeerde Ridders op Paarden	Only the dead have seen the end of the war	2023-03-20	war, horse, knight, dead
	The Confession	All art is a confession.	2022-04-14	beautiful, sad, dark, white
	Farewell	Love is enough	2021-03-12	beautiful, girl, knight, horse
	Mary - Mother of Jesus	Mary was a first-century Judean woman of Nazareth,[6] the wife of Joseph and the mother of Jesus.	0001-12-24	Catholic, Church, Maria

Figure 41. Photo uploaded

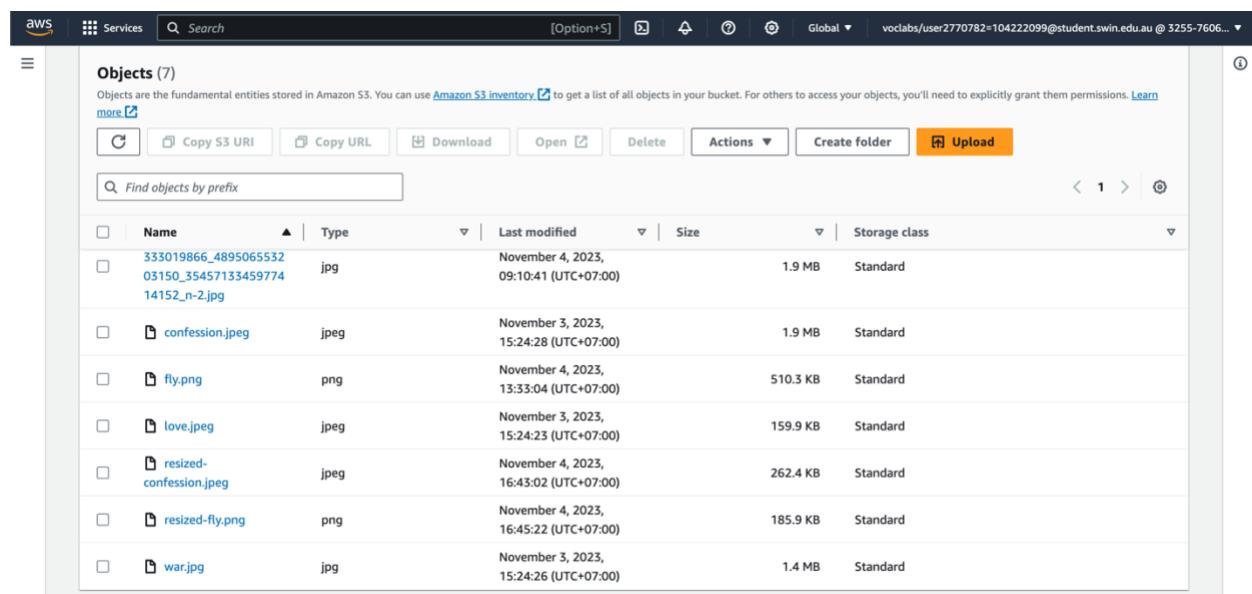
The photo uploader function is working correctly.

4) Resizing Lambda:


The screenshot shows the AWS Lambda function execution details. A green success message at the top states: "Successfully updated the function CreateThumbnail." Below it, a summary table provides execution statistics:

Summary	Value
Code SHA-256	Y5KcSugsAlJH2etMC6kGV2Zufc8PclpwpK30hn+aKEg=
Request ID	79884729-43b4-4b17-b3b6-631f5cfcd75e
Init duration	521.28 ms
Billed duration	5186 ms
Max memory used	120 MB
Execution time	36 seconds ago (November 4, 2023 at 04:43 PM GMT+7)
Function version	\$LATEST
Duration	5185.31 ms
Resources configured	128 MB

Log output section shows the command: START RequestId: 79884729-43b4-4b17-b3b6-631f5cfcd75e Version: \$LATEST

Figure 42. Successfully resize image


The screenshot shows the AWS S3 Objects list. It displays a table of files with their details:

Name	Type	Last modified	Size	Storage class
333019866_4895065532.jpg	jpg	November 4, 2023, 09:10:41 (UTC+07:00)	1.9 MB	Standard
14152_n-2.jpg				
confession.jpeg	jpeg	November 3, 2023, 15:24:28 (UTC+07:00)	1.9 MB	Standard
fly.png	png	November 4, 2023, 13:33:04 (UTC+07:00)	510.3 KB	Standard
love.jpeg	jpeg	November 3, 2023, 15:24:23 (UTC+07:00)	159.9 KB	Standard
resized-confession.jpeg	jpeg	November 4, 2023, 16:43:02 (UTC+07:00)	262.4 KB	Standard
resized-fly.png	png	November 4, 2023, 16:45:22 (UTC+07:00)	185.9 KB	Standard
war.jpg	jpg	November 3, 2023, 15:24:26 (UTC+07:00)	1.4 MB	Standard

Figure 43. Resized images uploaded