

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 utilizes EC2 web servers to host the website, with a range of AWS services such as S3, RDS, and Lambda. Individuals can upload visual images onto the online platform, where they may subsequently browse through collections of these images. Additionally, the website offers the feature of automatically generating and adjusting the size of thumbnail images. The complete source code and instructions have been provided to facilitate the integration of the website with the S3 bucket, RDS database, and Lambda function. This project leverages the capabilities of AWS services to develop a dynamic and user-centric picture album platform, incorporating functionalities such as 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.

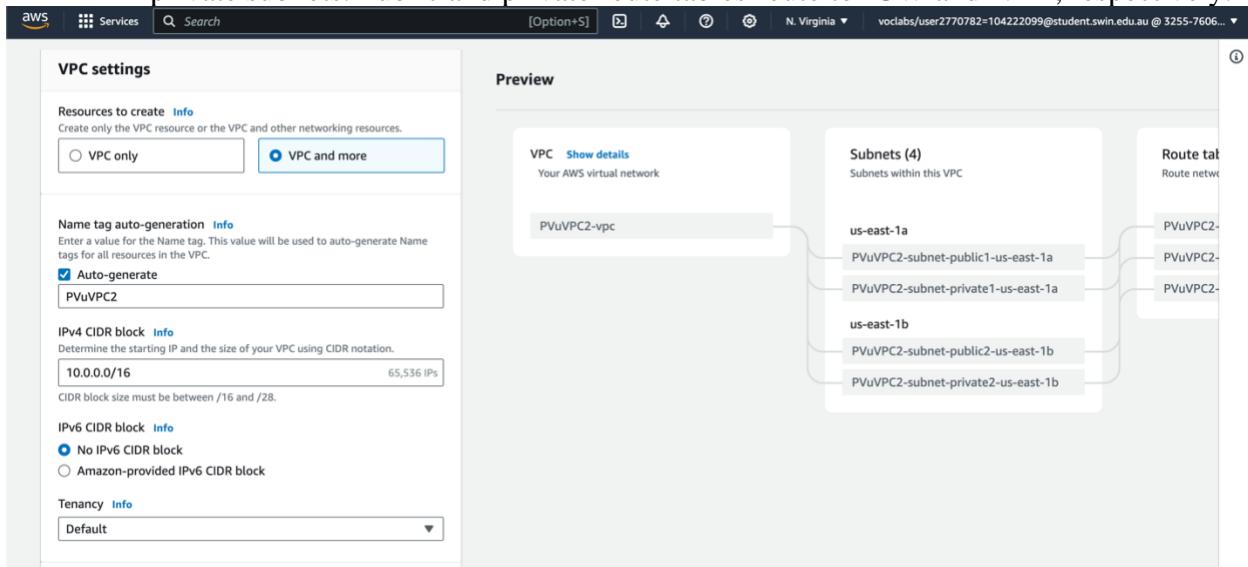


Figure 1. Configuring VPC

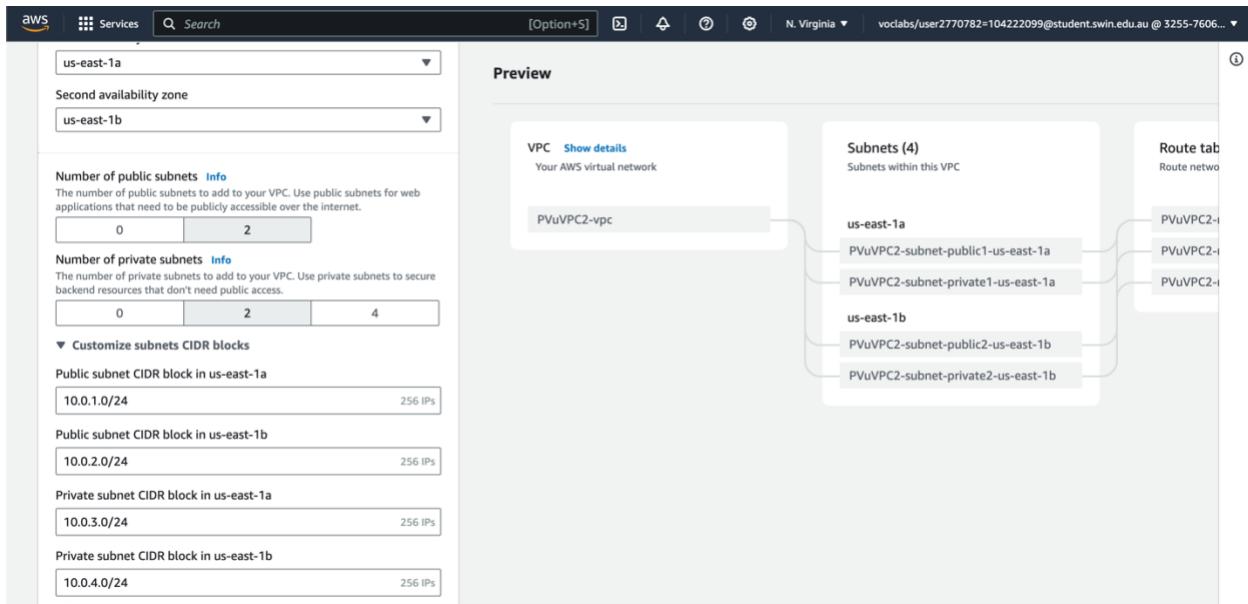


Figure 2. VPC Subnet CIDR Block

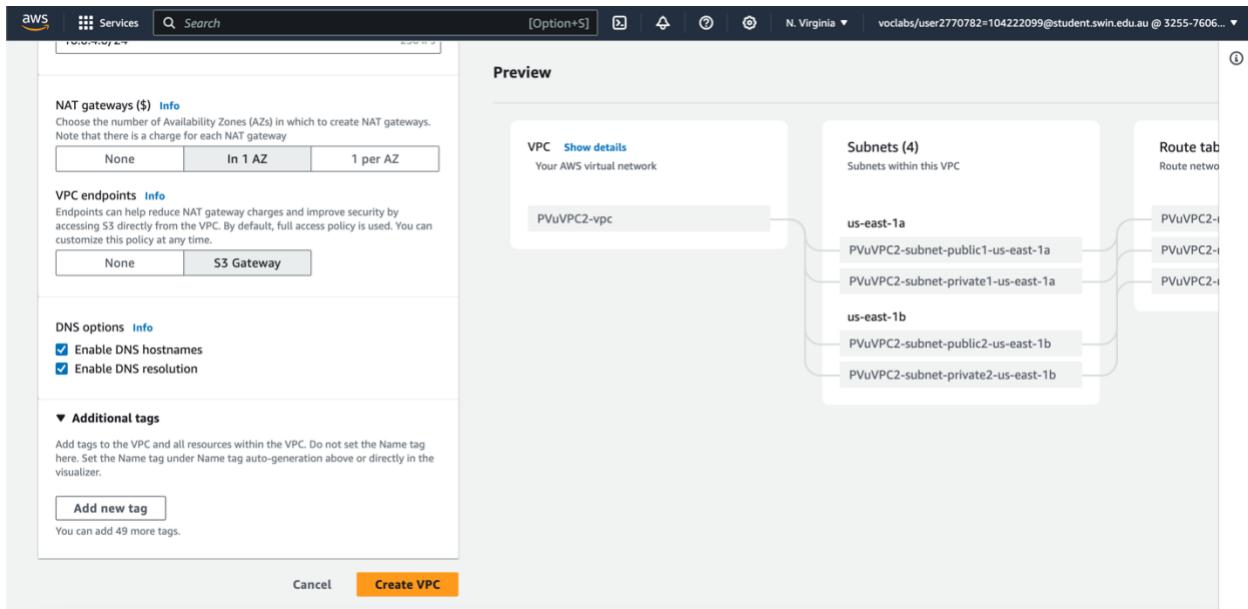


Figure 3. Configuring VPC

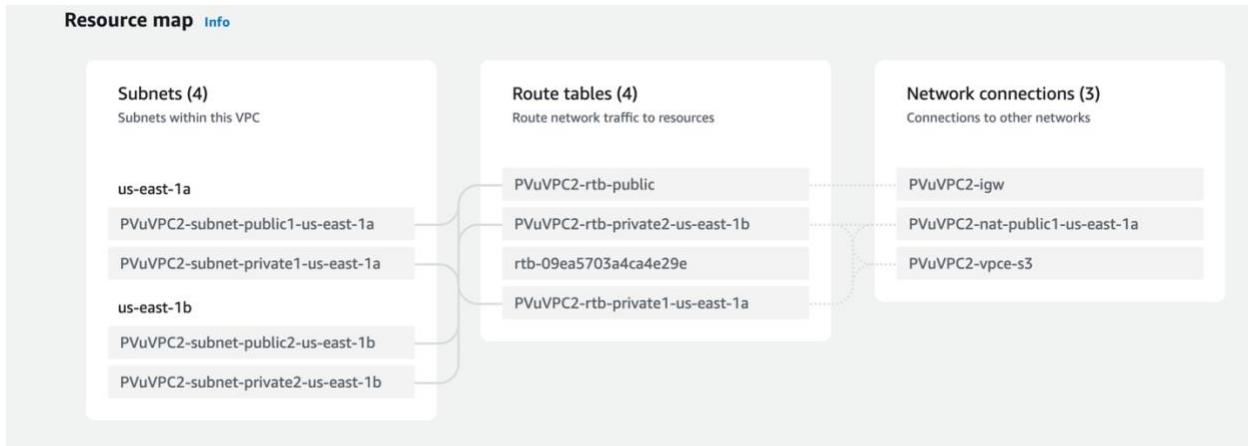


Figure 4. Resource Map

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 security group 'sg-058f495c53bc58bbc - ELBSG'. The 'Details' tab is selected, showing information like Security group ID (sg-058f495c53bc58bbc), Description ('for the ELB created'), and VPC ID (vpc-0e1e07996ffb5477c). The 'Owner' is listed as 325576069040. The 'Inbound rules' tab is selected, showing two rules: one for port 443 (TCP) and another for port 80 (TCP). Both rules allow traffic from 0.0.0.0/0. There are tabs for 'Outbound rules' and 'Tags'.

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

**Inbound rules (4)**

| Name                   | IP version | 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<br>DevServerSG | Security group ID<br>sg-056b0189bb9aeeef2b | Description<br>for the Dev server          | VPC ID<br>vpc-0e1e07996ffb5477c |
| Owner<br>325576069040              | Inbound rules count<br>1 Permission entry  | Outbound rules count<br>1 Permission entry |                                 |

**Inbound rules (1/1)**

| Name                  | IP version | Type | Protocol | Port range | Source    |
|-----------------------|------------|------|----------|------------|-----------|
| sgr-07d30170a51e18... | IPv4       | SSH  | TCP      | 22         | 0.0.0.0/0 |

Figure 7. DevServerSG for the Dev server.

**Details**

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

**Inbound rules (1/1)**

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

Figure 8. DBServerSG for the RDS instance.

Since NAT gateway is used instead of NAT instance, NATServerSG is not necessary. Outbound rule of all security group set to default (All traffic, anywhere IPv4).

| 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". A sub-header says "Change which subnets are associated with this network ACL." Below is a table titled "Available subnets (2/4)". It lists four subnets: PVuVPC2-subnet-public2..., PVuVPC2-subnet-public1..., PVuVPC2-subnet-private..., and PVuVPC2-subnet-private... (with the last two checked). The "Selected subnets" section shows two subnets: subnet-06d816e998233b2e9 / PVuVPC2-subnet-private1-us-east-1a and subnet-057a01944778361f6 / PVuVPC2-subnet-private2-us-east-1b. At the bottom 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". A sub-header says "Inbound rules control the incoming traffic that's allowed to reach the VPC." Below is a table titled "Rule number". It lists three rules: Rule 1 (Deny All ICMP - IPv4), Rule 2 (Allow All traffic), and a wildcard rule (\* Deny All traffic). At the bottom are "Add new rule", "Sort by rule number", "Cancel", "Preview changes", and "Save changes" buttons.

Figure 11. Inbound rules of PrivateSubnetsNACL

The screenshot shows the AWS VPC Network ACLs page. The URL is [VPC > Network ACLs > acl-082a9ded6a1c67345 / PrivateSubnetsNACL > Edit outbound rules](#). The title is "Edit outbound rules". It displays three rules:

| Rule number | Type            | Protocol | Port range | Destination | Allow/Deny |
|-------------|-----------------|----------|------------|-------------|------------|
| 1           | All ICMP - IPv4 | ICMP (1) | All        | 10.0.2.0/24 | Deny       |
| 2           | All traffic     | All      | All        | 0.0.0.0/0   | Allow      |
| *           | All traffic     | All      | All        | 0.0.0.0/0   | Deny       |

Buttons at the bottom include "Add new rule", "Sort by rule number", "Cancel", "Preview changes", and "Save changes".

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.

The screenshot shows the AWS IAM Roles page. The URL is [IAM > Roles > LabRole](#). The title is "LabRole". The "Summary" section includes:

| Creation date                         | ARN                                                    | Link to switch roles in console                                                                                                                                                     | Instance profile ARN                                                          |
|---------------------------------------|--------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------|
| September 27, 2023, 16:18 (UTC+07:00) | <a href="#">arn:aws:iam::325576069040:role/LabRole</a> | <a href="https://signin.aws.amazon.com/switchrole?roleName=LabRole&amp;account=325576069040">https://signin.aws.amazon.com/switchrole?roleName=LabRole&amp;account=325576069040</a> | <a href="#">arn:aws:iam::325576069040:instance-profile/LabInstanceProfile</a> |
| Last activity                         | Maximum session duration                               |                                                                                                                                                                                     |                                                                               |
| <a href="#">1 hour ago</a>            | 1 hour                                                 |                                                                                                                                                                                     |                                                                               |

Buttons include "Delete" and "Edit".

Figure 13. IAM role

- 5) CreateThumbnail Lambda function:** The Lambda function was permitted with an IAM execution role to ensure the proper security and permissions. The Lambda function is granted access to and control over the objects in the specified S3 bucket by the IAM role named LabRole, which was already created with the least privilege principle.

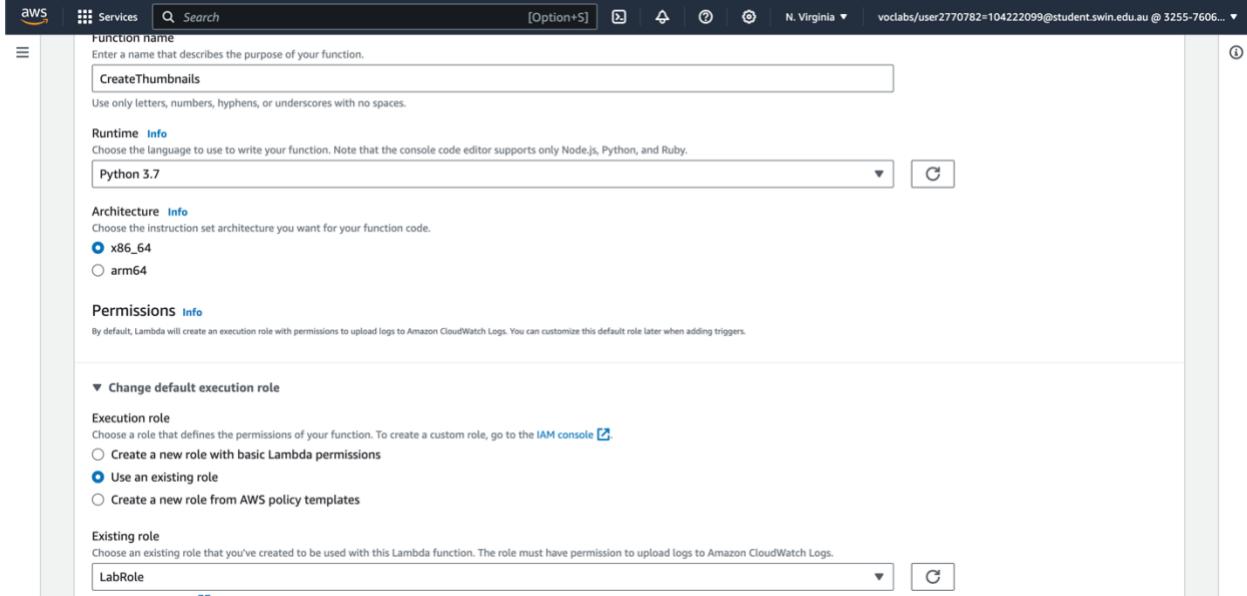


Figure 14. CreateThumbnail Lambda function configuration

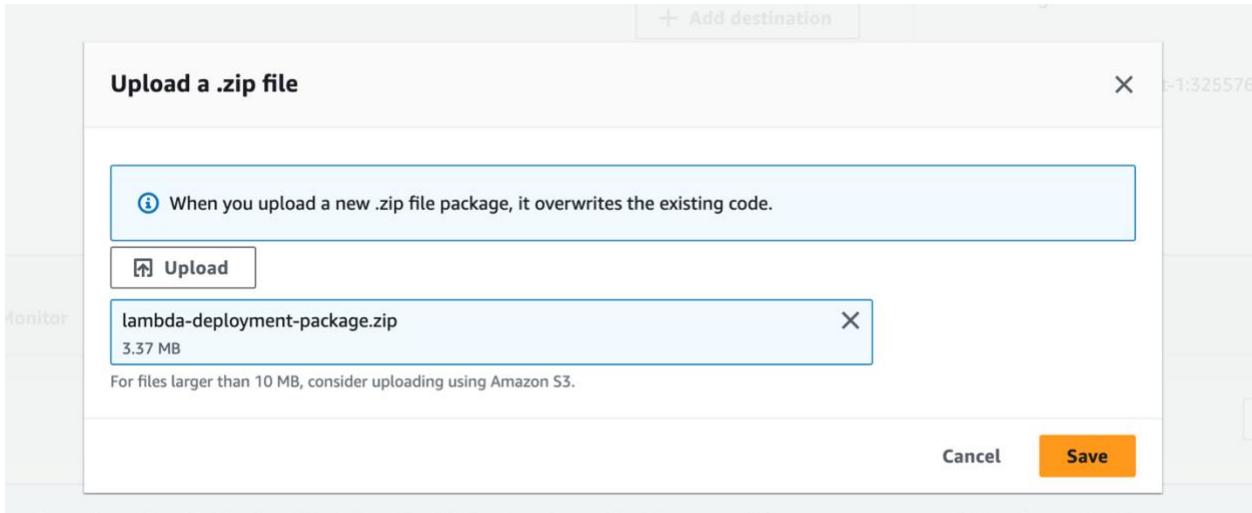


Figure 15. CreateThumbnail Lambda function configuration

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

*Dev Server Instance:* DevServer is not receiving traffic from ELB, as it serves as the platform solely for developing the custom AMI required to run the PhotoAlbum website. The custom AMI encompasses all the necessary components, such as the AWS PHP SDK, Apache web server, and website source code. Additionally, the DevServer can manage MySQL RDS instance using phpMyAdmin.

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

Updated less than a minute ago

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

Figure 16. DevServer instance resides in Public Subnet 2 (CIDR: 10.0.2.0/24) with an Elastic IP associated

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

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

Modify IAM role [Info](#)

Attach an IAM role to your instance.

|             |                                                                                                                                                                                     |
|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Instance ID | <a href="#">i-0477063fd41ea4d71 (DevServer)</a>                                                                                                                                     |
| 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. |
|             | <a href="#">LabInstanceProfile</a> <a href="#">Create new IAM role</a>                                                                                                              |

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

Figure 17. 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](#) Unregistered

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

Open Connection Search

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 18. DevServer Directory Structure with necessary SDKs and components included

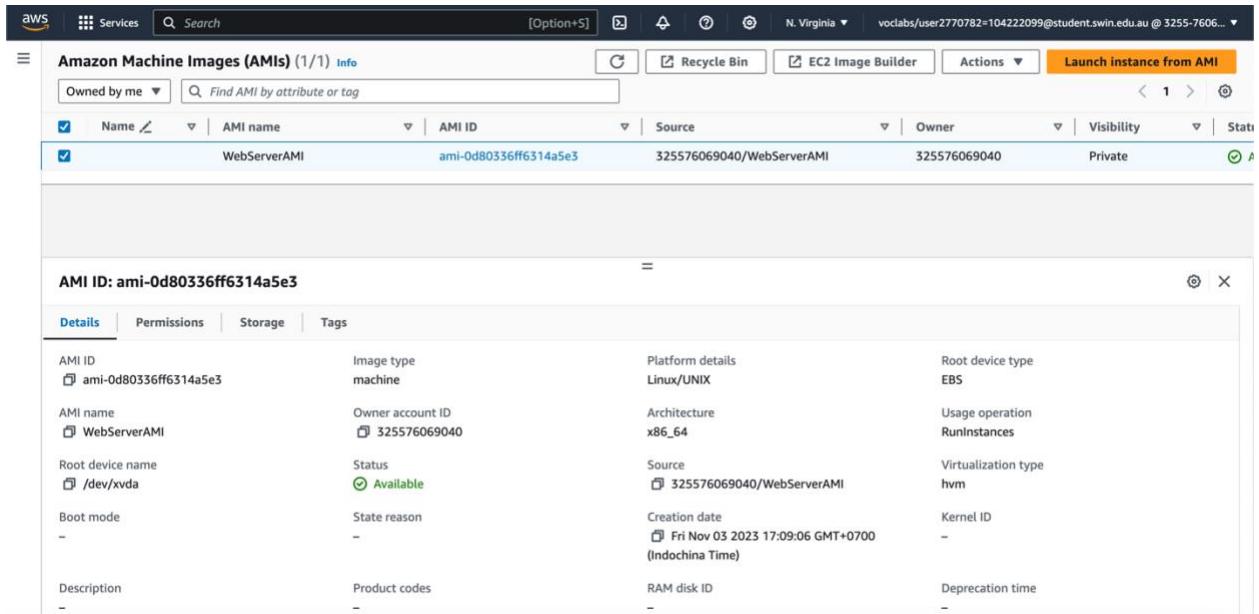


Figure 19. DevServer image created, ready to use

Launch template: Create a launch template with instance type t2.micro and IAM role LabInstanceProfile using the Figure 20 AMI.

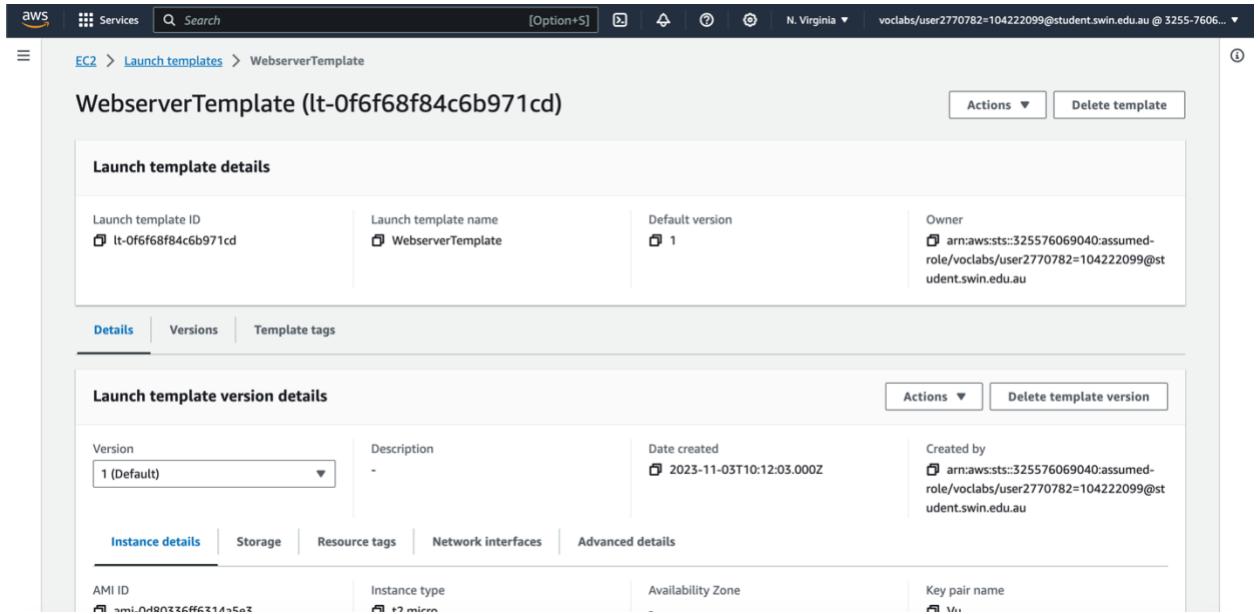


Figure 20. Launch template for ASG

Auto Scaling Group: The ASG is specifically configured to launch instances into the private subnets, maintaining a minimum of 2 instances and a maximum of 3 instances, with 2 instances as the desired number. This ensures the application has a minimum number of instances available while preventing the infrastructure from scaling beyond the defined maximum.

**Group details**

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

**Launch template**

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

Figure 21. 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<br><a href="#">WebServerTG</a> | Classic Load Balancers<br>- |
|------------------------------------------------------------|-----------------------------|

**VPC Lattice integration options**

|                                |
|--------------------------------|
| VPC Lattice target groups<br>- |
|--------------------------------|

**Health checks**

|                          |                                 |
|--------------------------|---------------------------------|
| Health check type<br>EC2 | Health check grace period<br>90 |
|--------------------------|---------------------------------|

Figure 22. ASG health check configuration with target group attached

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

|                                                       |                          |
|-------------------------------------------------------|--------------------------|
| <b>Target Tracking Policy</b>                         | <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 23. Target tracking policy based on application load balancer request

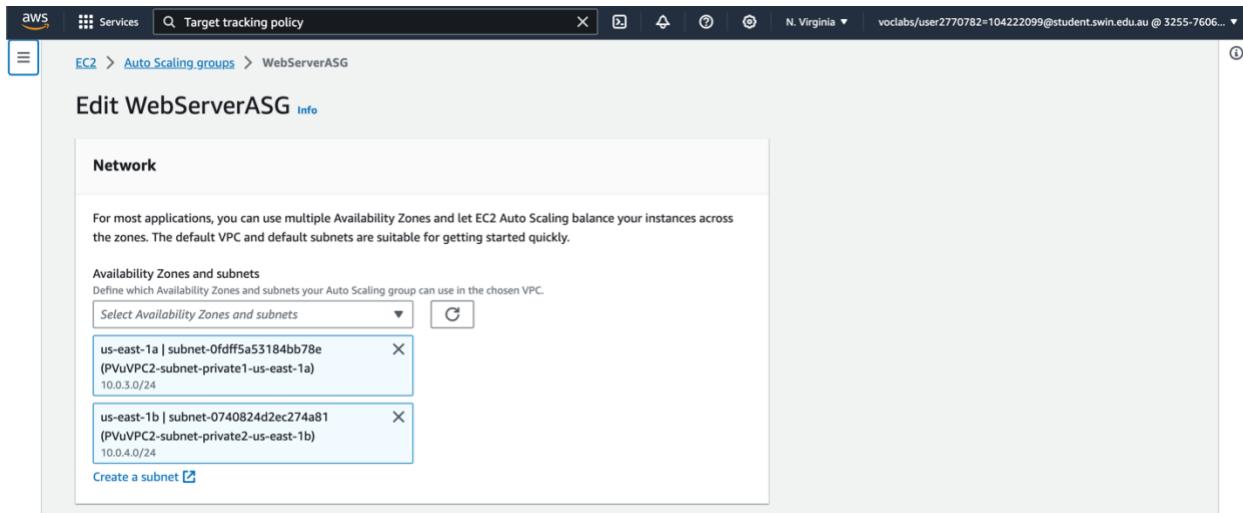


Figure 24. ASG network mapped to private subnets

Target tracking scaling was used to limit instances based on ELB target group requests (Figure 24). 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.

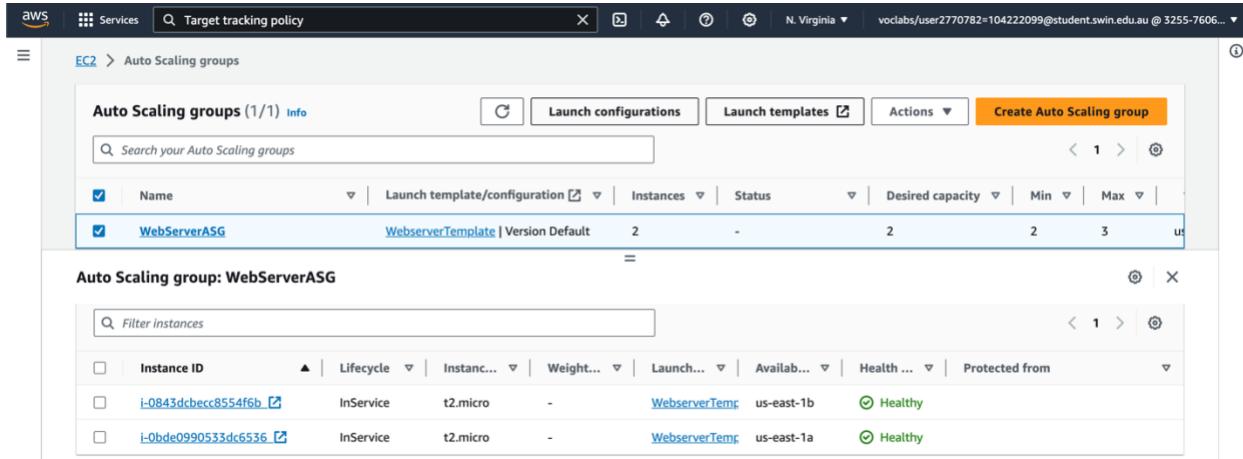
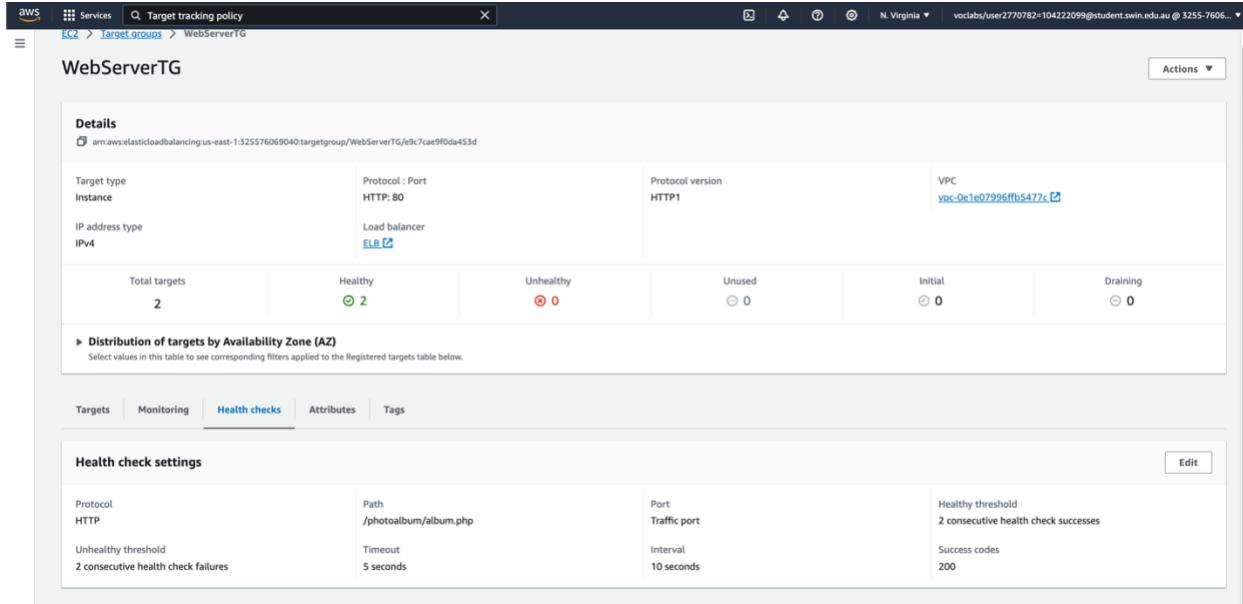


Figure 25. EC2 instances are properly distributed across two private subnets with healthy states

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 26. 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 ELB console. In the top navigation bar, 'Services' is selected, followed by 'Load balancers'. A search bar at the top right contains the query 'Target tracking policy'. Below the navigation, a table lists 'Load balancers (1/1)' with one entry: 'ELB' (Active, vpc-0e1e07996ffb5477c, 2 Availability Zones, application type, created November 3, 2023, 16:38 UTC+07:00). The main content area is titled 'Load balancer: ELB' and shows the 'Network mapping' tab selected. It details the VPC configuration (VPC: vpc-0e1e07996ffb5477c, IPv4: 10.0.0.0/16, IPv6: -) and the subnet mappings:

| Zone                 | Subnet                   | IPv4 address    | Private IPv4 address           | IPv6 address   |
|----------------------|--------------------------|-----------------|--------------------------------|----------------|
| us-east-1a (use1-a2) | subnet-0b9d14972ea0dd0ff | Assigned by AWS | Assigned from CIDR 10.0.1.0/24 | Not applicable |
| us-east-1b (use1-a4) | subnet-080982f479d334033 | Assigned by AWS | Assigned from CIDR 10.0.2.0/24 | Not applicable |

*Figure 27. Application Load Balancer mapped to Public Subnet 1 and Public Subnet 2*

The screenshot shows the AWS ELB console. In the top navigation bar, 'Services' is selected, followed by 'Load balancers'. A search bar at the top right contains the query 'Target tracking policy'. Below the navigation, a table lists 'Load balancers (1/1)' with one entry: 'ELB' (Active, vpc-0e1e07996ffb5477c, 2 Availability Zones, application type, created November 3, 2023, 16:38 UTC+07:00). The main content area is titled 'Load balancer: ELB' and shows the 'Listeners and rules' tab selected. It displays a single rule for port 80:

| Protocol:Port | Default action                                                                   | Rules  | ARN | Security policy | Default SSL/TLS certificate | Tags   |
|---------------|----------------------------------------------------------------------------------|--------|-----|-----------------|-----------------------------|--------|
| HTTP:80       | Forward to target group<br>• WebServerTg (100%)<br>• Group-level stickiness: Off | 1 rule | ARN | Not applicable  | Not applicable              | 0 tags |

*Figure 28. ELB Listener check rule which forwarded to the target group created in Figure 26*

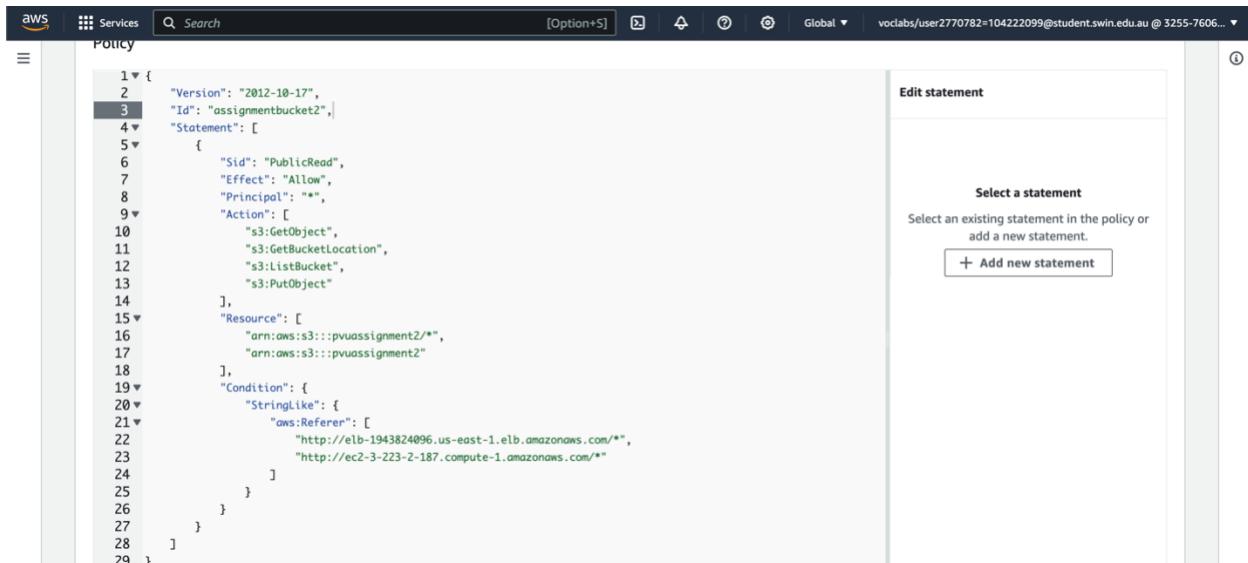
Now, The ELB can distribute incoming HTTP and HTTPS traffic across multiple EC2 targets.

- 8) **Simple Storage Service (S3):** The S3 bucket for images is almost identical to assignment 1b's. Permissions and policies have been added to this S3 bucket to ensure object accessibility. These steps ensure that the correct permissions are set to access the photos.

The screenshot shows the AWS S3 console. In the top navigation bar, 'Services' is selected, followed by 'Amazon S3'. A search bar at the top right contains the query 'Search'. Below the navigation, a table lists 'Buckets' with one entry: 'pvuassignment2' (Publicly accessible, US East (N. Virginia) region, ARN: arn:aws:s3:::pvuassignment2, Creation date: November 3, 2023, 10:20:24 (UTC+07:00)). The main content area is titled 'pvuassignment2' and shows the 'Properties' tab selected. It displays the bucket overview:

| AWS Region                      | Amazon Resource Name (ARN)  | Creation date                          |
|---------------------------------|-----------------------------|----------------------------------------|
| US East (N. Virginia) us-east-1 | arn:aws:s3:::pvuassignment2 | November 3, 2023, 10:20:24 (UTC+07:00) |

*Figure 29. 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 30. 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.

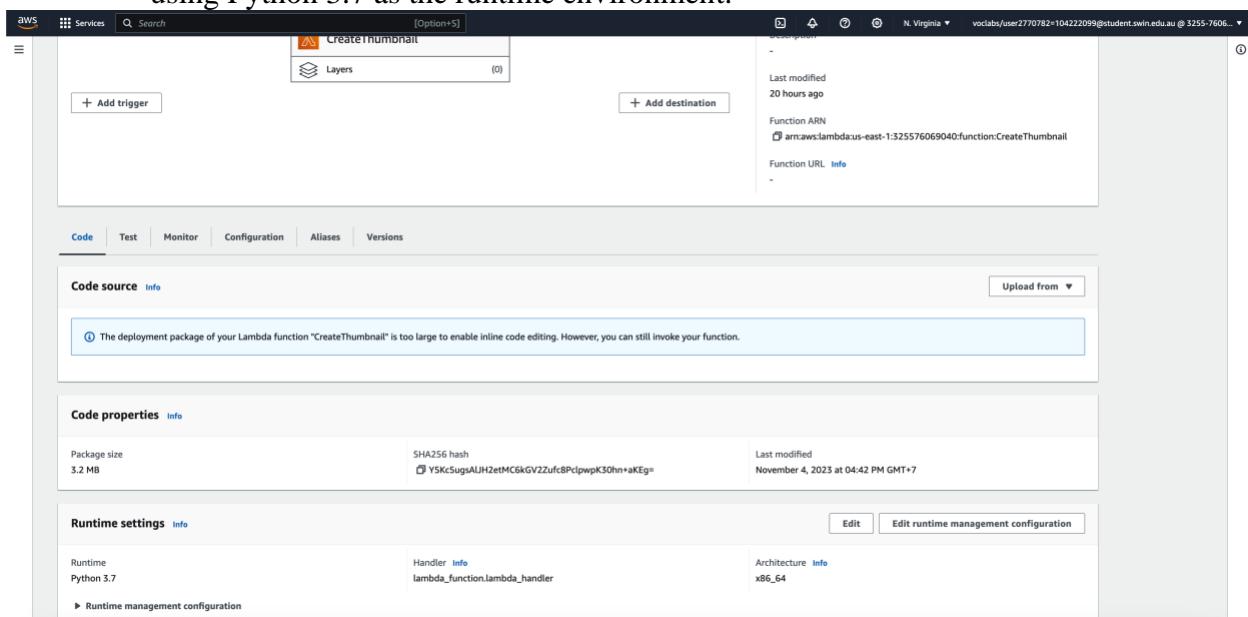


Figure 31. CreateThumbnail Lambda function configuration

The deployment package “lambda-deployment-package.zip” has been uploaded. This package contains the library and source code for resizing images and processing S3 bucket downloads and uploads.

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

Figure 32. 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 33. Subnet dbasm2 with Private subnet 3 and Private subnet 4

Figure 34. 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 35. 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 36. Photo display function

The photo display function is working correctly.

## 3) Photo uploader function:

Figure 37. Photo uploader function

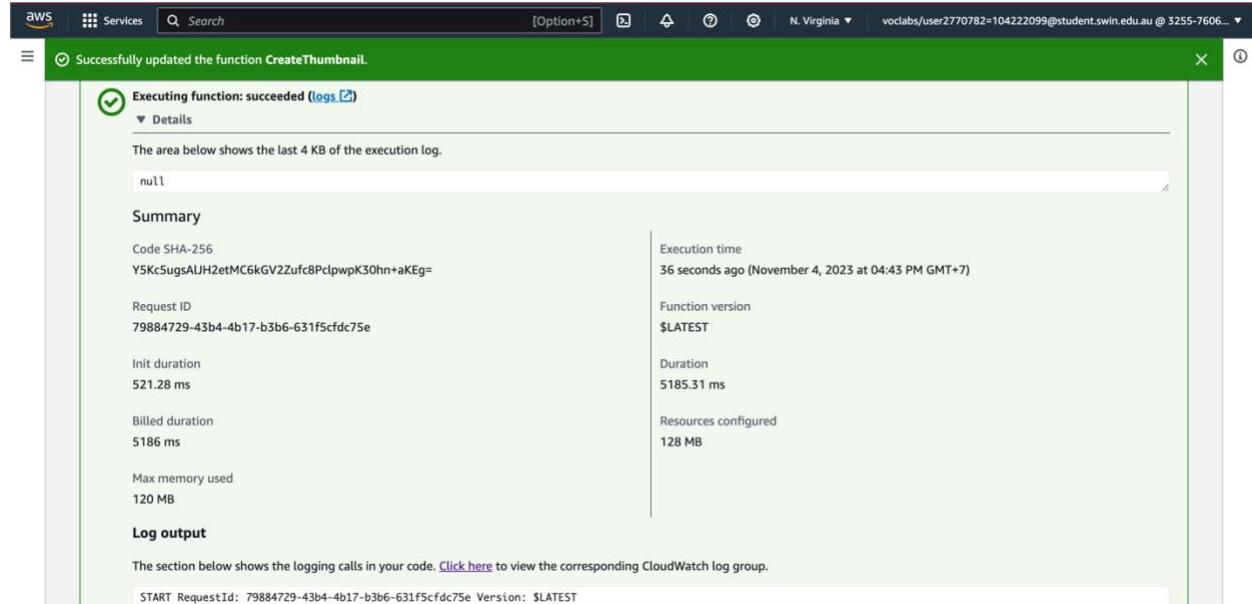
Student name: Phan Vu  
 Student ID: 104222099  
 Tutorial session: Saturday 01:00PM

**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 38. Photo uploaded*

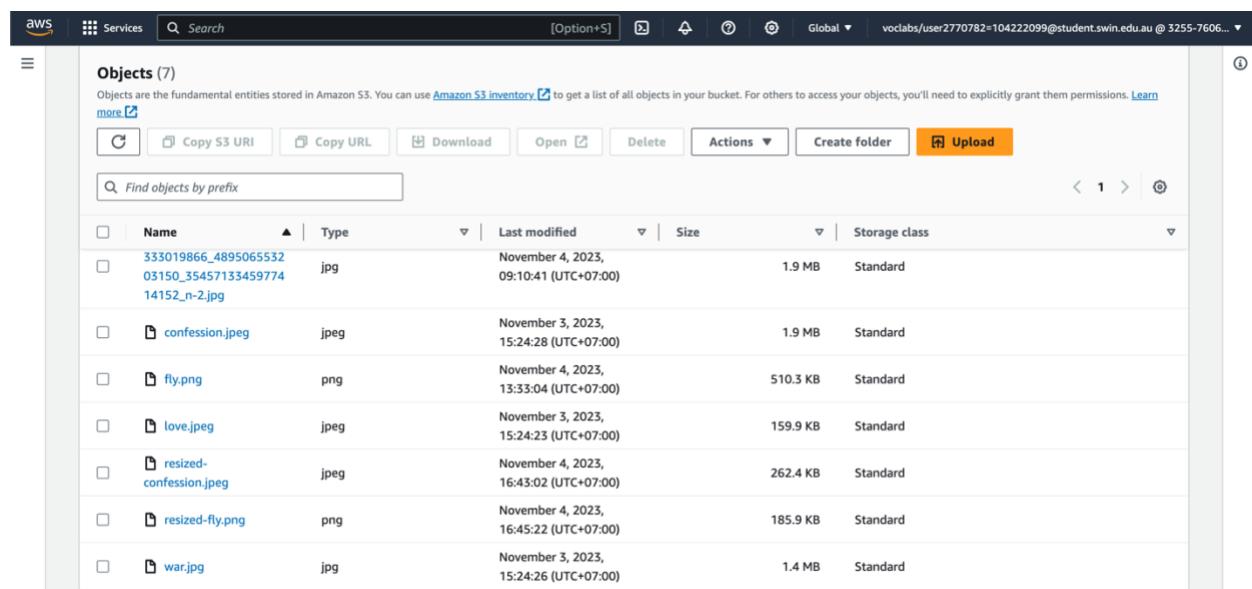
The photo uploader function is working correctly.

**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 39. 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 40. Resized images uploaded*