

COS20019 Cloud Computing Architecture – Assignment 1b

Lorraine Becker

Student ID: 104584773

Tutorial: 08:30 Thursday

Swinburne University of Technology

Melbourne, Australia

104584773@student.swin.edu.au

Abstract— Creating and deploying a Photo Album website onto a simple AWS infrastructure.

Links:

- Album.php:

<http://54.227.85.132/cos20019/photoalbum/album.php>

- phpMyAdmin:

<http://ec2-54-227-85-132.compute-1.amazonaws.com/phpmyadmin/index.php>

Keywords — cloud architecture, deployment, technology

I. INTRODUCTION

Assignment 1b of COS20019 Cloud Computing Architecture entails the development and deployment of a photo album website on simple Amazon Web Services (AWS) infrastructure. The infrastructure deployment is illustrated in the following diagram:

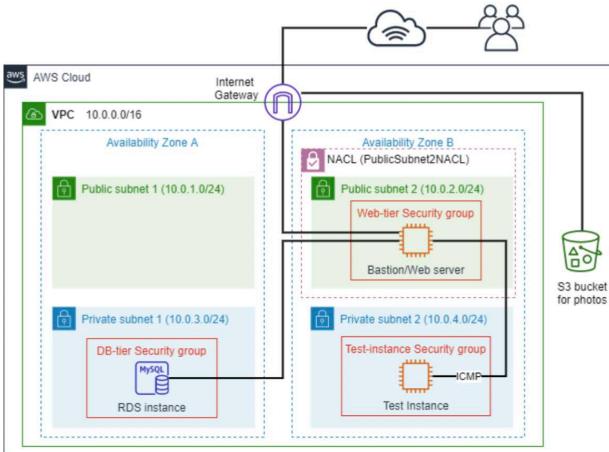


Fig. 1 System Architecture Diagram

INFRASTRUCTURE DEPLOYMENT

A secure Virtual Private Cloud (VPC) has been created, and the services required for this assignment were allocated within this VPC.

I. VPC

The VPC has been named LBeckerVPC. It was configured in the us-east-1 region and consists of two availability zones, each with a private and public subnet which follow the appropriate CIDR blocks specified in Figure 1.

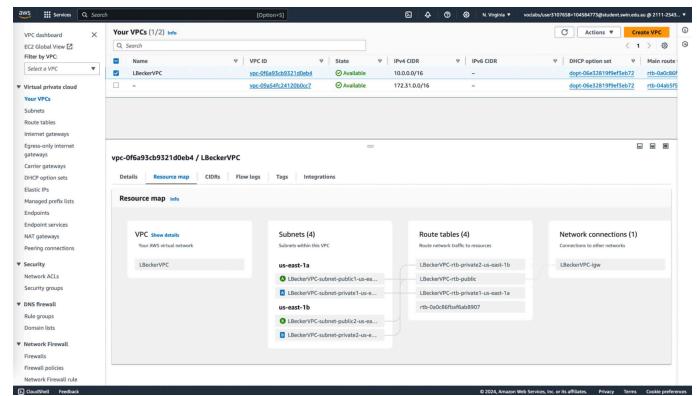


Fig. 2 Created VPC

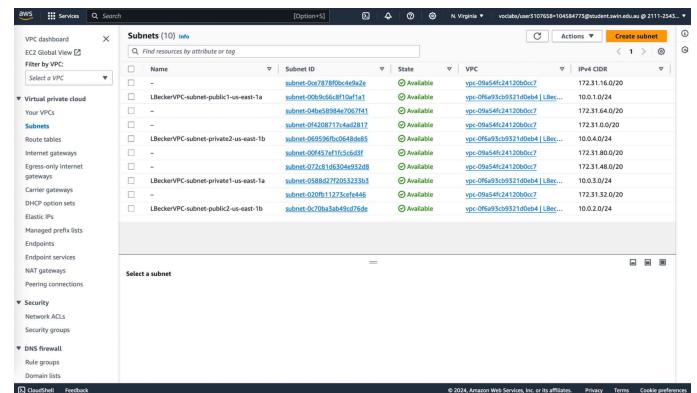


Fig. 3 Created subnets for VPC

The public subnets have been associated with a public route table configured to route traffic to the Internet Gateway. Conversely, the private route tables do not have an Internet gateway configuration. (Fig. 4).

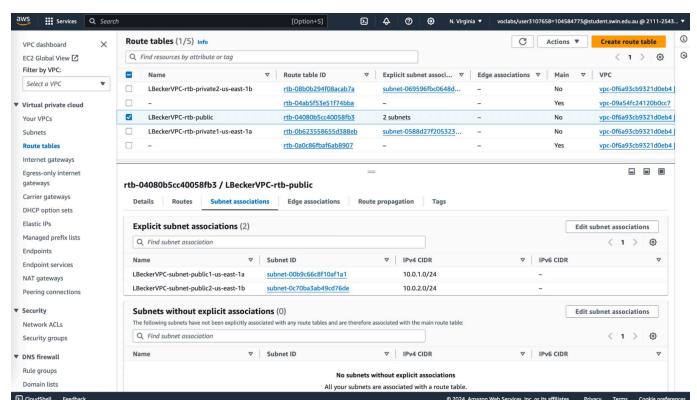


Fig. 4 Subnet associations of the public route table

The screenshot shows the AWS VPC Route Tables page. A specific route table, rtb-04080b5cc4005fb3, is selected. It contains two routes: one to a local destination (gw-0b0e2031f425e1f) and another to a target (10.0.0.0/16). The second route is propagated.

Fig. 5 Routes of public route table

The screenshot shows the AWS Security Groups page. The TestInstanceSG security group is selected. It has three inbound rules: one for HTTP (port 80), one for SSH (port 22), and one for ICMP (All traffic).

Fig. 8 TestInstanceSG Security Group

The screenshot shows the AWS Subnet Associations page for a private subnet. It lists subnet associations for the rtb-0b623558655d588eb route table, which covers the entire subnet range (10.0.3.0/24).

Fig. 6 Subnet associations of Private subnet 1

WebServerSG has three inbound rules:

- Permits HTTP (port 80) traffic from any source.
- Permits SSH (port 22) traffic from any source.
- Permits all ICMP traffic from the security group TestInstanceSG.

In a home environment, the IP address can change at any moment, therefore it is not reliable to depend on a single source. However, it is advisable to restrict SSH access to specific IP address ranges, such as those within an admin subnet of a business network.

The screenshot shows the AWS Subnet Associations page for another private subnet. It lists subnet associations for the rtb-0b8b0b294f08acab7a route table, which covers the range 10.0.4.0/24.

Fig. 7 Subnet associations of Private subnet 2

The screenshot shows the AWS Security Groups page. The WebServerSG security group is selected. It has two inbound rules: one for HTTP (port 80) and one for SSH (port 22). It also has an outbound rule allowing all ICMP traffic.

Fig. 9 WebServerSG Security Group

DBServerSG is configured with an inbound rule that allows MySQL access from the WebServerSG security group. Access from any other sources is restricted.

The screenshot shows the AWS Security Groups page. The DBServerSG security group is selected. It has one inbound rule for MySQL (TCP port 3306) from the WebServerSG security group.

Fig. 10 DBServerSG Security Group

II. SECURITY GROUPS

Security groups were created as depicted in Figure 1, with each one being configured with a specific set of rules.

TestInstanceSG has an inbound rule which permits all traffic attempting to access the resources associated with it.

Two EC2 instances were created. The Web Server Instance was placed in the public subnet in order to make it publicly accessible. The Web Server is assigned an elastic IP address, providing it with a static IP address. This allows for easy access to the Web Server, such as establishing an SSH session, through the static IP address, saving time. As the IP address does not require any changes, this implies that the IP address can be used for an A record on the domain.

Fig. 11 Elastic IP address

Fig. 12 Web Server EC2 Instance

```
Last login: Sun Apr 14 14:59:56 on ttys000
1b0000228Lorraine-Becker ~% ssh-add -K /Users/lb000022/Downloads/TestInstance.pem
WARNING: The -K and -A flags are deprecated and have been replaced
by the --apple-use-keychain and --apple-load-keychain
flags, respectively. To suppress this warning, set the
environment variable APPLE_SSH_ADD_BEHAVIOR as described in
the ssh-add(1) manual page.
Identity added: /Users/lb000022/Downloads/TestInstance.pem (/Users/lb000022/Downloads/TestInstance.pem)
1b0000228Lorraine-Becker ~% ssh -A -i /Users/lb000022/Downloads/WebServer.pem ec2-user@ec2-54-227-85-1
32.compute-1.amazonaws.com
Last login: Sat Apr 13 11:32:15 2024 from 46.82.70.115.static.exetel.com.au
#
# Amazon Linux 2
## #####
## \#####
## \### AL2 End of Life is 2025-06-30.
## \### Y-+-->
## / A newer version of Amazon Linux is available!
## / \
## / Amazon Linux 2023, GA and supported until 2028-03-15.
## / https://aws.amazon.com/linux/amazon-linux-2023/
## / \
## /m/ \
## /m/ \
[ec2-user@ip-10-0-2-198 ~]$ ssh ec2-user@10.0.2.72
Last login: Sat Apr 13 11:33:10 2024 from ip-10-0-2-190.ec2.internal
#
# Amazon Linux 2
## #####
## \#####
## \### AL2 End of Life is 2025-06-30.
## \### Y-+-->
## / A newer version of Amazon Linux is available!
## / \
## / Amazon Linux 2023, GA and supported until 2028-03-15.
## / https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-10-0-2-198 ~]$ ping 10.0.2.190
PING 10.0.2.190 (10.0.2.190) 56(84) bytes of data.
64 bytes from 10.0.2.190: icmp_seq=1 ttl=255 time=0.269 ms
64 bytes from 10.0.2.190: icmp_seq=2 ttl=255 time=0.491 ms
^C
-- 10.0.2.190 ping statistics --
2 packets transmitted, 2 received, 0% packet loss, time 1026ms
rtt min/avg/max/mdev = 0.269/0.388/0.491/0.111 ms
[ec2-user@ip-10-0-2-198 ~]$
```

Fig. 13 Establish an SSH connection to both the Web Server and the Test Instance, then ping the Web Server from the Test Instance.

An SSH connection was successfully created with the Web Server. Using agent forwarding through the Mac terminal, another SSH connection was made from the Web Server to the Test Instance. The Server was tested by pinging the Web Server from the Test Instance. Ultimately, the Web Server's ICMP rule was verified through the server's connectivity.

IV. RDS Database Instance

An instance of Amazon Relational Database Service (RDS) named "photoalbum-db" has been established to serve as the host for the photo album database, which includes a table dedicated to storing photo metadata. Following this setup, the PHP application accesses the "photos" table. Security measures include associating the RDS instance with the "DBServerSG" security group, which permits MySQL traffic from the "WebServerSG" security group. This setup ensures secure communication between the Web Server instance and the database. Additionally, the RDS instance's subnet group is configured to use the VPC's two private subnets.

Fig. 14 Photo Album RDS Instance

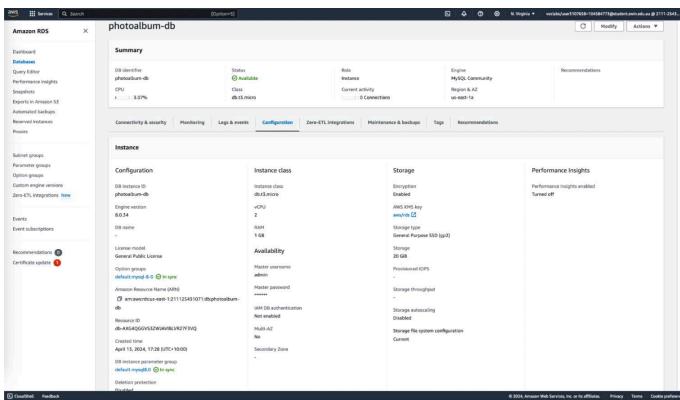


Fig. 15 Configuration of the Photo Album RDS Instance.

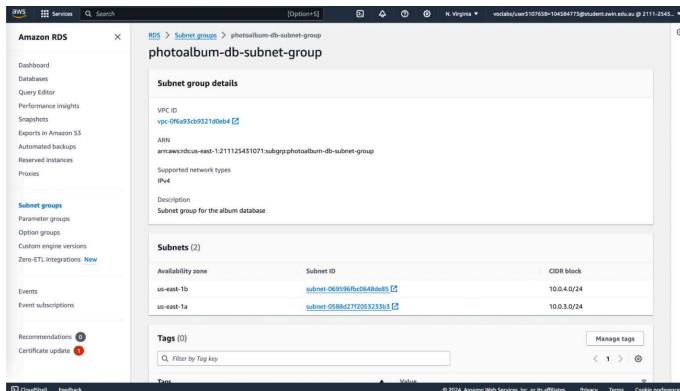


Fig. 16 Subnet group configuration for the Photo Album RDS Instance.

I created a photos table in my photoalbum database on the RDS instance, then added some data to it using phpMyAdmin on the Web Server instance. I connected phpMyAdmin to the RDS instance using its endpoint address.

```
config.inc.php

1 <?php
2 /**
3  * phpMyAdmin sample configuration, you can use it as base for
4  * manual configuration. For easier setup you can use setup/
5  *
6  * All directives are explained in documentation in the doc/ folder
7  * or at <https://docs.phpmyadmin.net/>.
8  */
9
10 declare(strict_types=1);
11
12 /**
13  * This is needed for cookie based authentication to encrypt the cookie.
14  * Needs to be a 32-bytes long string of random bytes. See FAQ 2.10.
15  */
16 $cfg['blowfish_secret'] = ''; /* YOU MUST FILL IN THIS FOR COOKIE AUTH! */
17
18 /**
19  * Servers configuration
20  */
21 $i = 0;
22
23 /**
24  * First server
25  */
26 $i++;
27 /**
28  * Authentication type */
29 $cfg['Servers'][$i]['auth_type'] = 'cookie';
30 /* Server parameters */
31 $cfg['Servers'][$i]['host'] = 'photoalbum-db.cj4kokhgig77.us-east-1.rds.amazonaws.com';
32 $cfg['Servers'][$i]['compress'] = false;
33 $cfg['Servers'][$i]['AllowNoPassword'] = false;
34
35 /**
36  * phpMyAdmin configuration storage settings.
37 */
38
39 /* User used to manipulate with storage */
40 // $cfg['Servers'][$i]['controlhost'] = '';
41 // $cfg['Servers'][$i]['controlport'] = '';
42 // $cfg['Servers'][$i]['controluser'] = 'pma';
43 // $cfg['Servers'][$i]['controlpass'] = 'pmapass';
44
45 /* Storage database and tables */
46 // $cfg['Servers'][$i]['pmadb'] = 'phpmyadmin';
47 // $cfg['Servers'][$i]['bookmarktable'] = 'pma_bookmark';
48 // $cfg['Servers'][$i]['relation'] = 'pma_relation';
49 // $cfg['Servers'][$i]['table_info'] = 'pma_table_info';
50 // $cfg['Servers'][$i]['table_coords'] = 'pma_table_coords';
51 // $cfg['Servers'][$i]['pdf_pages'] = 'pma_pdf_pages';
52 // $cfg['Servers'][$i]['column_info'] = 'pma_column_info';
53 // $cfg['Servers'][$i]['history'] = 'pma_history';

Line 1 Column 1
```

Fig. 17 phpMyAdmin configuration

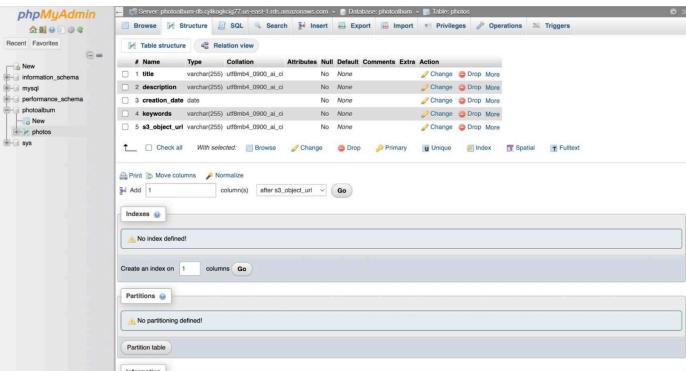
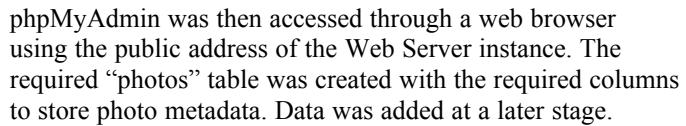


Fig. 18 Creation of the table structure using phpMyAdmin.

- Virtual Network ACLs:

Network Access Control List Creation:

- A Network Access Control List (NACL) named "PublicSubnet2NACL" was created with the following rules:
 - Allow SSH traffic from anywhere.
 - Allow ICMP traffic only from the Test Instance's subnet (10.0.4.0/24).
 - Allow other traffic to ensure the complete functionality of the Photo Album website for users accessing it from anywhere.

Inbound Rules for NACL:

- **Rule 10:** Authorize HTTP (port 80) access from any location, guaranteeing global accessibility to the photo album site hosted on the Web Server.
 - **Rules 20 and 30:** Allow the ephemeral port range [2] to support MySQL connections within the subnets of the database subnet group. Additionally, Rule 30 facilitates SSH outgoing connections from the Web Server to the Test Instance via this port range.
 - **Rule 40:** Explicitly permit SSH from any source. Although not ideal, it's necessary for operational purposes, as specifying specific sources for SSH access isn't feasible.
 - **Rule 50:** Permit ICMP from private subnet 2, which accommodates the Test Instance, enabling it to ping the Web Server.
 - **Rule 60:** Restrict ICMP traffic from any source outside the designated private subnet, ensuring security by denying access from unauthorized locations.

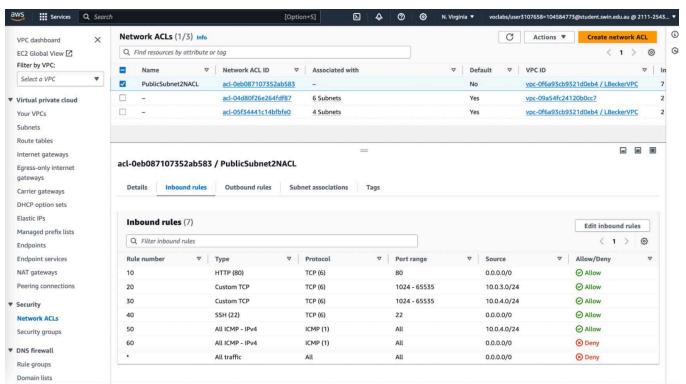


Fig. 19 PublicSubnet2NACL inbound rules

In the outbound rules, a rule was added to allow traffic from any source. This rule overrides the default rule that denies all traffic. Without this change, only incoming traffic would be permitted, while outbound communication would be restricted, resulting in one-way communication.

The screenshot shows the AWS Network ACLs (1/3) page. It lists two Network ACLs: 'PublicSubnet2NACL' and 'PublicSubnet2NACL'. Under 'PublicSubnet2NACL', there are 6 Subnets. The 'Outbound rules' section shows two rules:

Rule number	Type	Protocol	Port range	Destination	Action
4	All traffic	All	All	0.0.0.0/0	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

Fig. 20 Outbound rules for PublicSubnet2NACL

FUNCTIONAL REQUIREMENTS FOR THE PHOTO ALBUM WEBSITE

I. S3 bucket

In the us-east-1 region, a bucket named 'lbecker-photoalbum-bucket' was created. The bucket is dedicated to storing photos for the photo album website. Additionally, a policy was implemented to enable public access within the bucket.

The screenshot shows the 'Permissions' tab of the 'lbecker-photoalbum-bucket' bucket. It displays a 'Block public access (Bucket settings)' section with 'Block all public access' selected. Below it is a 'Bucket policy' section containing the following JSON policy:

```

{
    "Version": "2008-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::lbecker-photoalbum-bucket/*"
        }
    ]
}

```

Fig. 21 S3 bucket policy

II. Added rows to the photo table

For testing, more rows were added to the photos table within the RDS instance's photoalbum database. Each entry included URLs for each photo, which point to the bucket, then added to the table.

The screenshot shows the phpMyAdmin interface for the 'photos' table in the 'photoalbum' database. The table has columns: title, description, creation_date, keywords, and a2_object_url. There are four records:

title	description	creation_date	keywords	a2_object_url
Swinburne Logo	Logo of Swinburne uni	2021-08-09	logo, university	https://lbecker-photoalbum-bucket.s3.amazonaws.com/
Bunny	Image of a bunny	0000-00-00	bunny, animal	https://lbecker-photoalbum-bucket.s3.amazonaws.com/
Motorcycle	Picture of a motorcycle	2022-04-08	motorcycle, purple	https://lbecker-photoalbum-bucket.s3.amazonaws.com/
Switzerland	The Alps in Switzerland	2020-06-18	Switzerland, mountains, Alps	https://lbecker-photoalbum-bucket.s3.amazonaws.com/

Fig. 22 Photos table with several records of data

The source files of the photo album were transferred to the Web Server instance via SFTP using Cyberduck on a Mac system.

The screenshot shows the Cyberduck interface with the path 'ec2-54-227-85-132... /'. The 'Filename' list shows various files transferred from the S3 bucket to the EC2 instance, including 'cos2019', 'photoalbum', 'constants.php', 'defaultstyle.css', 'mydb.php', and 'photo.php'. The 'Size' and 'Modified' columns provide details for each file.

Fig. 23 Transferred source files to Web Server

The constants.php file was altered to incorporate details associated with the database such as the RDS's endpoint address, database name, and column names. These adjustments specify the location of the photos and dictate which columns to use for presenting information. Upon accessing album.php from a web browser, images are shown as expected by retrieving each image through its corresponding URL from the S3 bucket.

The screenshot shows a web browser window with the URL 'Not Secure | 54.227.85.132/cos2019/photoalbum/album.php'. The page displays a table titled 'Uploaded photos:' with four rows, each showing a thumbnail image, the photo name, description, creation date, and keywords. The student information at the top of the page is as follows:

- Student name: Lorraine Becker
- Student ID: 104584773
- Tutorial session: Thursday 08:30AM

Fig. 24 Accessing album.php via the web browser over the Internet

CONCLUSION

Amazon Web Services offers robust and flexible infrastructure and tools for creating cloud-based systems. The Photo Album website was launched on AWS by using AWS EC2 for the Web Server, RDS for the photo metadata, S3 for storing the image files, and the implementation of Network ACLs to enhance the web server's security.

REFERENCES

- [1] Swinburne University, “<https://swinburne.instructure.com/courses/56945/modules>” 2024. [Online].
- [2] “Ephemeral port selection,” [www.ibm.com](https://www.ibm.com/docs/en/zos/3.1.0?topic=profiletcpip-ephemeral-port-selection). “<https://www.ibm.com/docs/en/zos/3.1.0?topic=profiletcpip-ephemeral-port-selection>” 2021. [Online]

