

COS20019 Cloud Computing Architecture Assignment 1B

Name: Lai Yee Fung

ID: 101225312

Tutorial Class: Wednesday 6:30p.m.

BA513

URL: <http://ec2-35-172-168-37.compute-1.amazonaws.com/cos80001/photoalbum/album.php>

Screenshot of database

The screenshot shows the phpMyAdmin interface for a MySQL database named 'assignment1b'. The 'photos' table is selected, and its structure is displayed. The table has five columns: phototitle, description, creationdate, keywords, and reference. The data is as follows:

phototitle	description	creationdate	keywords	reference
img_mountains_wide	image of mountains	2022-09-13	mountain, mountains	https://assignment1b-101225312.s3.amazonaws.com/im...
img_nature_wide.jpg	image of nature	2022-09-13	nature, wide	https://assignment1b-101225312.s3.amazonaws.com/im...
img_snow_wide.jpg	image of snow	2022-09-13	snow, wide	https://assignment1b-101225312.s3.amazonaws.com/im...
Alpaca	picture of an alpaca	2022-09-15	alpaca, animal	https://assignment1b-101225312.s3.amazonaws.com/AL...
Puppy	picture of a puppy	2022-09-15	puppy, dog, animal	https://assignment1b-101225312.s3.amazonaws.com/do...
Swinburne Logo	picture of the Swinburne Logo	2022-09-15	Logo, Swinburne	https://assignment1b-101225312.s3.amazonaws.com/sw...

Screenshot of S3 bucket

The screenshot shows the Amazon S3 console for the bucket 'assignment1b-101225312'. The bucket is publicly accessible. The objects listed are:

Name	Type	Last modified	Size	Storage class
Alpaca.png	png	September 15, 2022, 20:27:37 (UTC+10:00)	14.7 KB	Standard
dog.png	png	September 15, 2022, 20:27:36 (UTC+10:00)	28.3 KB	Standard
img_mountains_wide.jpg	jpg	September 13, 2022, 20:11:22 (UTC+10:00)	59.0 KB	Standard
img_nature_wide.jpg	jpg	September 13, 2022, 20:11:23 (UTC+10:00)	50.6 KB	Standard
img_snow_wide.jpg	jpg	September 13, 2022, 20:11:24 (UTC+10:00)	32.5 KB	Standard
swinburne logo.png	png	September 15, 2022, 20:27:35 (UTC+10:00)	67.7 KB	Standard

Screenshot of terminal ping from Test instance to Web server EC2 instance

The screenshot displays the AWS Management Console interface. On the left, the navigation menu includes sections for EC2 Dashboard, Events, Tags, Limits, Instances, Images, Elastic Block Store, and Network & Security. The main content area shows the 'Instances (1/2)' page with a table of EC2 instances. Two instances are listed: '1b-Bastion/W...' and '1b-TestInstance'. The '1b-TestInstance' is selected, and a terminal window is open, showing the output of a ping command from the '1b-TestInstance' to the '1b-Bastion/W...' instance. The terminal output shows successful ping results with various statistics.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
1b-Bastion/W...	i-01fe98a318f6ebd27	Running	t2.micro	2/2 checks passed	No alarms	us-east-1b	ec2-35-172-168-37.co...	35.172.168.37	35.172.168.37
1b-TestInstance	i-07d6d04afa839cd5e	Running	t2.micro	2/2 checks passed	No alarms	us-east-1b	-	-	-

```
ec2-user@ip-10-0-4-7:~$ ping 10.0.2.10
PING 10.0.2.10 (10.0.2.10) 64(64) bytes of data:
64 bytes from 10.0.2.10: icmp_seq=1 ttl=255 time=0.540 ms
64 bytes from 10.0.2.10: icmp_seq=2 ttl=255 time=0.530 ms
64 bytes from 10.0.2.10: icmp_seq=3 ttl=255 time=0.596 ms
64 bytes from 10.0.2.10: icmp_seq=4 ttl=255 time=0.605 ms
64 bytes from 10.0.2.10: icmp_seq=5 ttl=255 time=0.598 ms
64 bytes from 10.0.2.10: icmp_seq=6 ttl=255 time=0.556 ms
64 bytes from 10.0.2.10: icmp_seq=7 ttl=255 time=0.575 ms
64 bytes from 10.0.2.10: icmp_seq=8 ttl=255 time=0.520 ms
64 bytes from 10.0.2.10: icmp_seq=9 ttl=255 time=0.610 ms
64 bytes from 10.0.2.10: icmp_seq=10 ttl=255 time=0.550 ms
64 bytes from 10.0.2.10: icmp_seq=11 ttl=255 time=0.600 ms
Host 10.0.2.10 ping statistics:
 11 packets transmitted, 11 received, 0% packet loss, time 10239ms
rtt min/avg/max/mdev = 0.520/0.571/0.610/0.091 ms
^C
ec2-user@ip-10-0-4-7:~$
```

This is done by accessing the private Test instance from the public Bastion/Web Server instance with Pageant agent forwarding.

Step by step:

1. Creating VPC

The screenshot displays the AWS Management Console 'Create VPC' page. The interface includes a top navigation bar with the AWS logo, 'Services' link, a search bar, and user information. The main content area is titled 'Create VPC' and includes a notification about the new create VPC experience. The 'VPC settings' section on the left contains fields for 'Resources to create' (VPC only, VPC and more), 'Name tag auto-generation' (Auto-generate, project), 'IPv4 CIDR block' (10.0.0.0/16, 65,536 IPs), and 'IPv6 CIDR block' (No IPv6 CIDR block, Amazon-provided IPv6 CIDR block). The 'Preview' section on the right shows a diagram of the VPC architecture, including subnets (us-east-1a, us-east-1b), route tables (project-rtb-public, project-rtb-private1-us-east-1a, project-rtb-private2-us-east-1b), and network connections (project-igw, project-vpc-s3). Below the preview, the 'Your VPCs (2)' table lists existing VPCs.

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP option set	Main route table
-	vpc-001f7ec8d9984b85	Available	172.31.0.0/16	-	dopt-00dffa42692397...	rtb-05240ef2fffe035eb
YLaiVPC-vpc	vpc-02e0848850c18f1cc	Available	10.0.0.0/16	-	dopt-00dffa42692397...	rtb-0227da05add8c1464

I first created the VPC for the assignment. In the creation page, I also created the 4 subnets (2 public and 2 private) required for the assignment with 2 in each availability zone. The CIDR blocks was also configured in this page.

2. Creating Security Groups

The screenshot displays the AWS Management Console interface for creating and managing security groups. The top section shows the 'Create security group' wizard with the following details:

- Security group name:** Assignment Security Group
- Description:** for report use
- VPC:** vpc-02e0848850c18f1cc

The 'Inbound rules' section shows two rules configured:

Type	Protocol	Port range	Source	Description - optional
Custom TCP	TCP	0	Custom	
Custom TCP	TCP	0	Custom	

The bottom section shows the 'Security Groups (1/5)' list with the following data:

Name	Security group ID	Security group name	VPC ID	Description	Owner	Inbound rules count	Outbound rules count
-	sg-0b45c858c1f568e9d	DBServerSG	vpc-02e0848850c18f1cc	Assignment-1B	406290611783	1 Permission entry	1 Permission entry
-	sg-0646caad2ec2c742c	TestInstanceSG	vpc-02e0848850c18f1cc	Assignment-1B	406290611783	1 Permission entry	1 Permission entry
-	sg-0048bd460b3846641	WebServerSG	vpc-02e0848850c18f1cc	Assignment-1B	406290611783	3 Permission entries	1 Permission entry
-	sg-08a97e3e2bcc7119f	default	vpc-001f7ec88d9984b85	default VPC security gr...	406290611783	1 Permission entry	1 Permission entry
-	sg-05ed3c509b012ebff	default	vpc-02e0848850c18f1cc	default VPC security gr...	406290611783	1 Permission entry	1 Permission entry

I created 3 security groups to be used in the VPC. The inbound rules are set in the Inbound rules section by choosing the type of connection and the source to match the requirements of the assignment. Security groups limit the connections allowed to go into the object it is assigned to.

3. Creating EC2 Instances

This screenshot shows the 'Network settings' step of the AWS 'Create Instance' wizard. The 'Key pair name' is set to 'Select'. Under 'Network settings', the 'VPC' is 'vpc-02e048850c18f1cc (VLa/VPC-vpc)' and the 'Subnet' is 'subnet-0edfeda3ecc019b07'. The 'Auto-assign public IP' is set to 'Enable'. Under 'Firewall (security groups)', the 'Create security group' option is selected. The 'Common security groups' list includes 'WebServerSG sg-0048bd460b3846641'. A 'Free tier' notification box is visible on the right, stating that the first year includes 750 hours of t2.micro or t3.micro instances. The 'Launch instance' button is highlighted in orange.

I created 2 EC2 instances for this assignment (Bastion/WebServer and Test Instance). The VPC, subnet, and security groups are configured according to the requirements of the assignment in the network settings section.

This screenshot shows the 'Advanced details' step of the AWS 'Create Instance' wizard. The 'Specify CPU options' checkbox is unchecked. Under 'User data', a script is entered in the text area:

```
#!/bin/bash
yum update -y
amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2
service httpd start
yum install -y httpd mariadb-server php-mbstring php-xml
systemctl start httpd
systemctl enable httpd
usermod -s /bin/bash ec2-user
chown -R ec2-user:apache /var/www
chmod 2775 /var/www
find /var/www -type d -exec sudo chmod 2775 {} \;
find /var/www -type f -exec sudo chmod 0664 {} \;
echo "<?php echo <h2>Welcome to COS80001. Installed PHP version: '
phpversion() . '</h2>';?>" > /var/www/html/phpinfo.php
```

 The 'Launch instance' button is highlighted in orange.

For the Bastion/Web Server instance, Apache http server and php is installed with a script in the User Data located under Advanced Details section.

for services, features, blogs, docs, and more [Alt+S] N. Virginia voclabs/user2066898=101225312@student.swin.edu.au @ 4062-9061

EC2 > Elastic IP addresses > 35.172.168.37

35.172.168.37 Actions Associate Elastic IP address

Summary			
Allocated IPv4 address 35.172.168.37	Type Public IP	Allocation ID eipalloc-02fb443bb55a68188	Reverse DNS record -
Association ID eipassoc-08fd82b1d72ad8885	Scope VPC	Associated instance ID i-01fe98a318f6ebd27	Private IP address 10.0.2.10
Network interface ID eni-0b337c196131c9c67	Network interface owner account ID 406290611783	Public DNS ec2-35-172-168-37.compute-1.amazonaws.com	NAT Gateway ID -
Address pool Amazon	Network Border Group us-east-1		

Elastic IP was also created and assigned to the Bastion/Web Server instance to make sure that the public IPv4 address, and DNS remains the same even after stopping the instance.

4. Creating Database Subnet and Database

es, features, blogs, docs, and more [Alt+S]

To create a new subnet group, give it a name and a description, and choose an existing VPC. You will then be able to add subnets related to that VPC.

Subnet group details

Name
You won't be able to modify the name after your subnet group has been created.
Assignment1b
Must contain from 1 to 255 characters. Alphanumeric characters, spaces, hyphens, underscores, and periods are allowed.

Description
for report purposes

VPC
Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created.
YLaIVPC-vpc (vpc-02e0848850c18f1cc)

Add subnets

Availability Zones
Choose the Availability Zones that include the subnets you want to add.
Choose an availability zone
us-east-1a X us-east-1b X

Subnets
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.
Select subnets
subnet-0ab53da5f96c5a414 (10.0.4.0/24) X
subnet-05e4ae9a34d48643e (10.0.3.0/24) X

First the subnet group for the database is created for the VPC and subnets.

Services Search for services, features, blogs, docs, and more [Alt+S]

Network type [Info](#)
To use dual-stack mode, make sure that you associate an IPv6 CIDR block with a subnet in the VPC you specify.
☒ IPv4
Your resources can communicate only over the IPv4 addressing protocol.
☐ Dual-stack mode
Your resources can communicate over IPv4, IPv6, or both.

Virtual private cloud (VPC) [Info](#)
Choose the VPC. The VPC defines the virtual networking environment for this DB instance.
YLaIVPC-vpc (vpc-02e0848850c18f1cc)
Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change its VPC.

DB Subnet group [Info](#)
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.
assignment1b

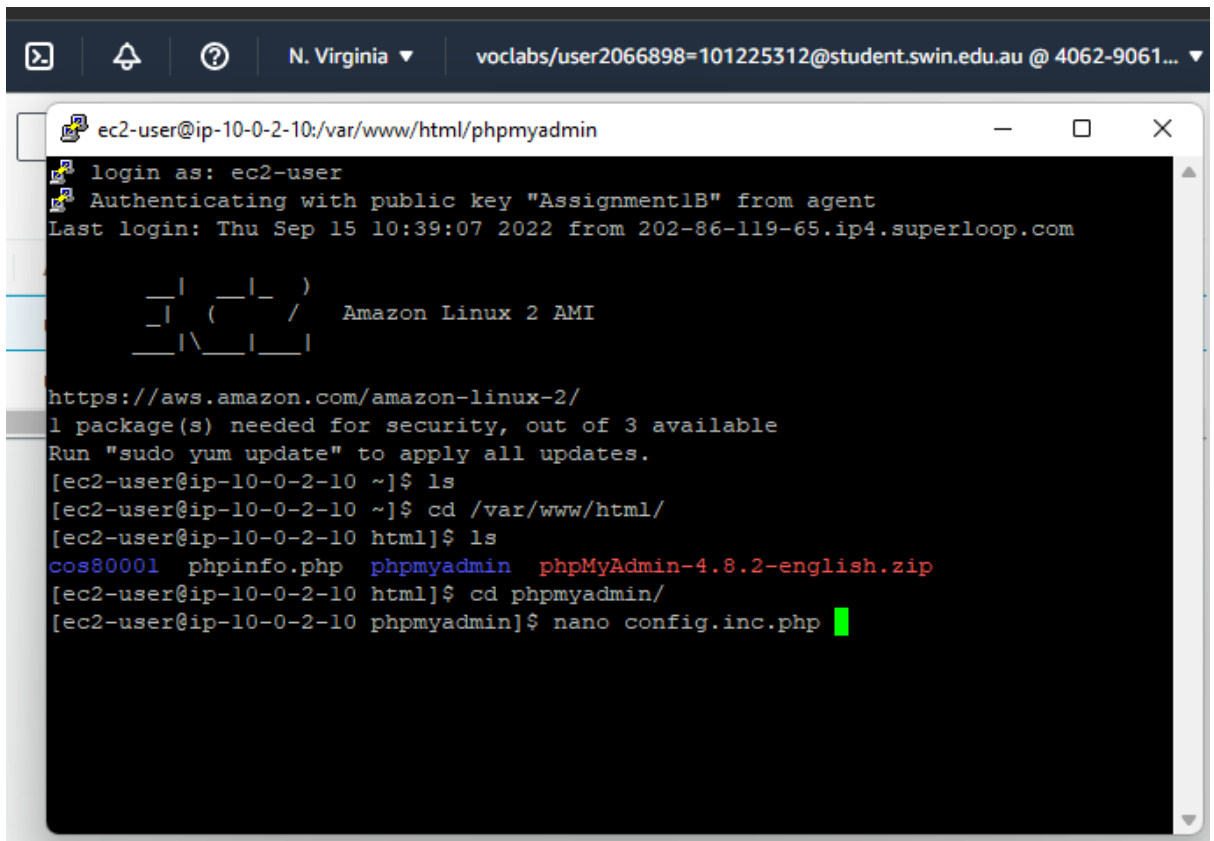
Public access [Info](#)
☐ Yes
RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.
☒ No
RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

VPC security group (firewall) [Info](#)
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.
☒ Choose existing
Choose existing VPC security groups
☐ Create new
Create new VPC security group

Existing VPC security groups
Choose one or more options
DBServerSG X

A SQL database is created in the assignment VPC and into the DB subnet group. It is also using the security group DBServerSG made in the first step.

5. Installing phpMyAdmin in Bastion/WebServer

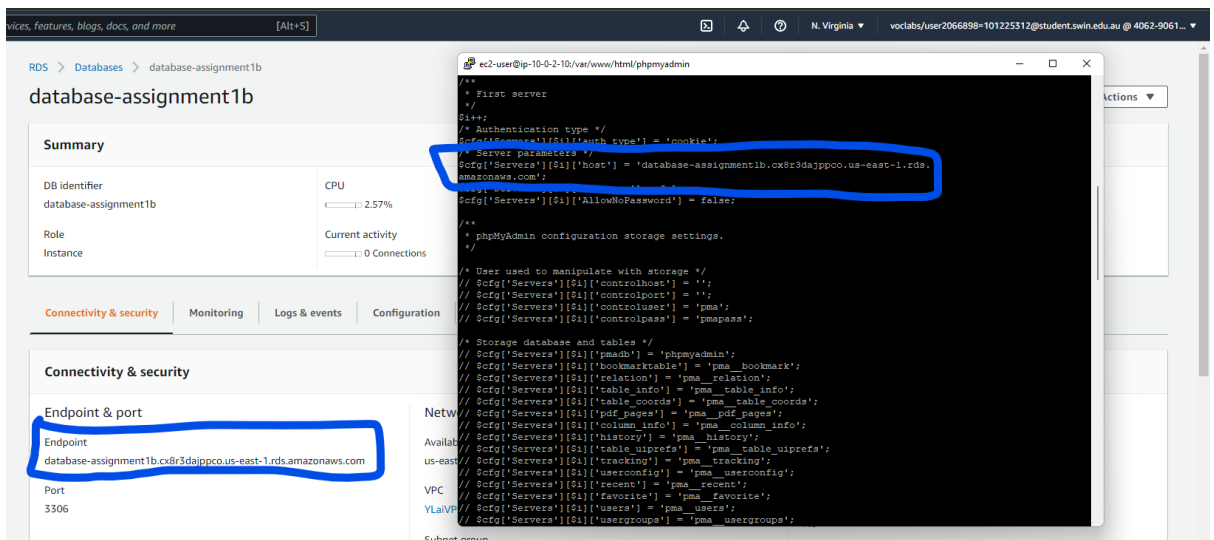


```
ec2-user@ip-10-0-2-10:/var/www/html/phpmyadmin
login as: ec2-user
Authenticating with public key "Assignment1B" from agent
Last login: Thu Sep 15 10:39:07 2022 from 202-86-119-65.ip4.superloop.com

 _ | _ | _ )
 _ | ( _ /   Amazon Linux 2 AMI
 _ | \ _ | _ |

https://aws.amazon.com/amazon-linux-2/
1 package(s) needed for security, out of 3 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-0-2-10 ~]$ ls
[ec2-user@ip-10-0-2-10 ~]$ cd /var/www/html/
[ec2-user@ip-10-0-2-10 html]$ ls
cos80001  phpinfo.php  phpmyadmin  phpMyAdmin-4.8.2-english.zip
[ec2-user@ip-10-0-2-10 html]$ cd phpmyadmin/
[ec2-user@ip-10-0-2-10 phpmyadmin]$ nano config.inc.php
```

I connected to the Bastion/Web Server instance with putty using the public IPv4 DNS and key pair. Using the link provided, I installed phpMyAdmin to /var/www/html/ directory. Then I had to reconfigure the localhost field with my database endpoint using the text editor “nano”.

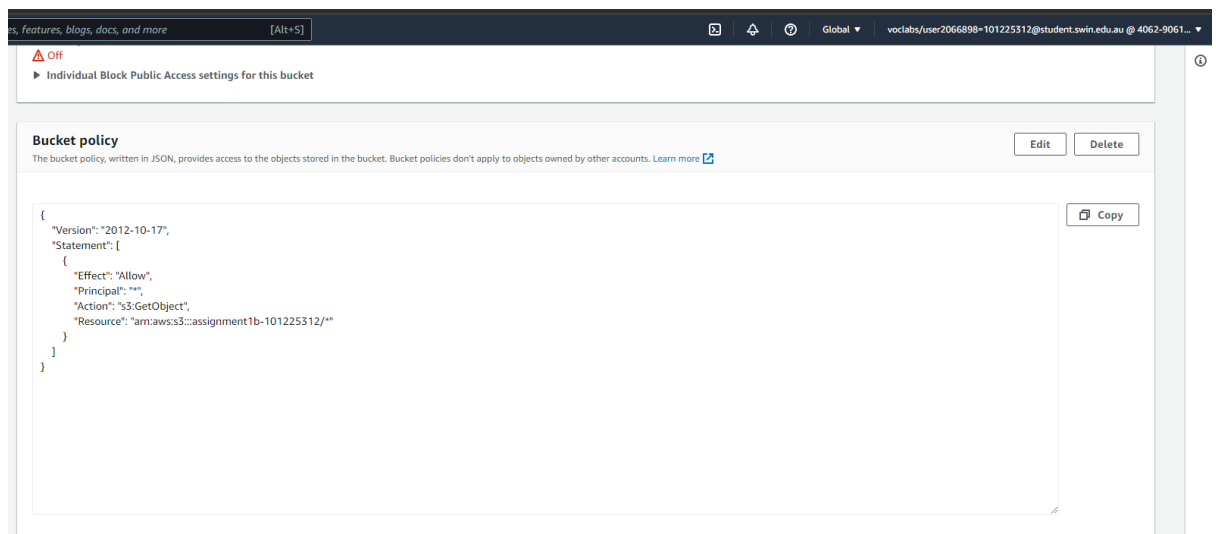


The screenshot shows the AWS RDS console for the instance 'database-assignment1b'. The 'Connectivity & security' tab is selected, and the 'Endpoint and port' section is highlighted. The endpoint is 'database-assignment1b.cx8r3dajppco.us-east-1.rds.amazonaws.com' and the port is '3306'. Overlaid on this is a terminal window showing the configuration of 'config.inc.php' for phpMyAdmin. The 'Servers' array is configured with the following values:

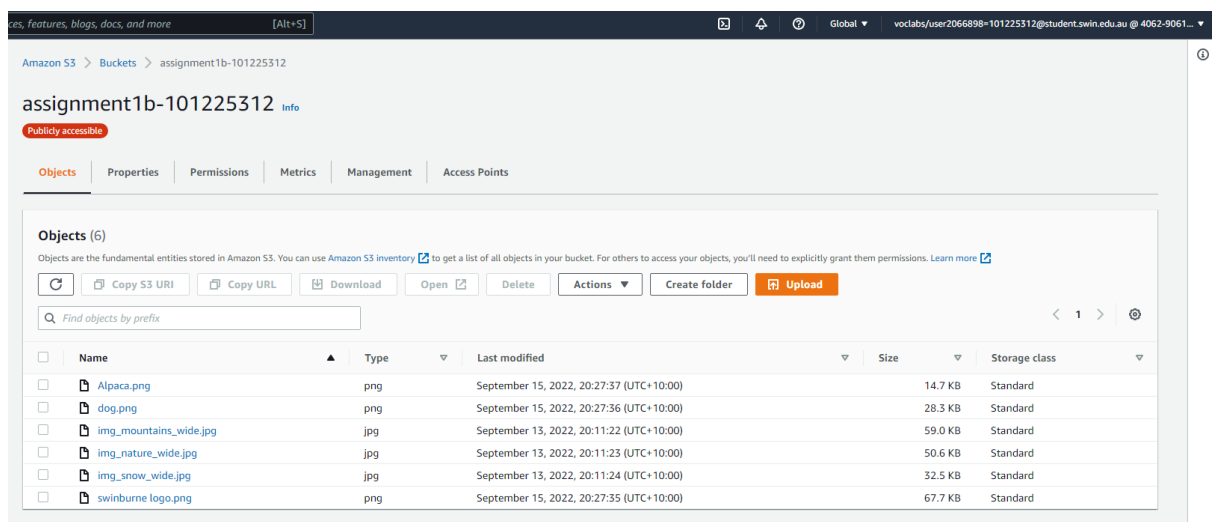
```
$cfg['Servers'][$i]['host'] = 'database-assignment1b.cx8r3dajppco.us-east-1.rds.amazonaws.com';
$cfg['Servers'][$i]['AllowNoPassword'] = false;
```

With this, phpMyAdmin webpage accessed with the Bastion/Webserver will be able to use the database created.

6. Creating and configuring S3 bucket

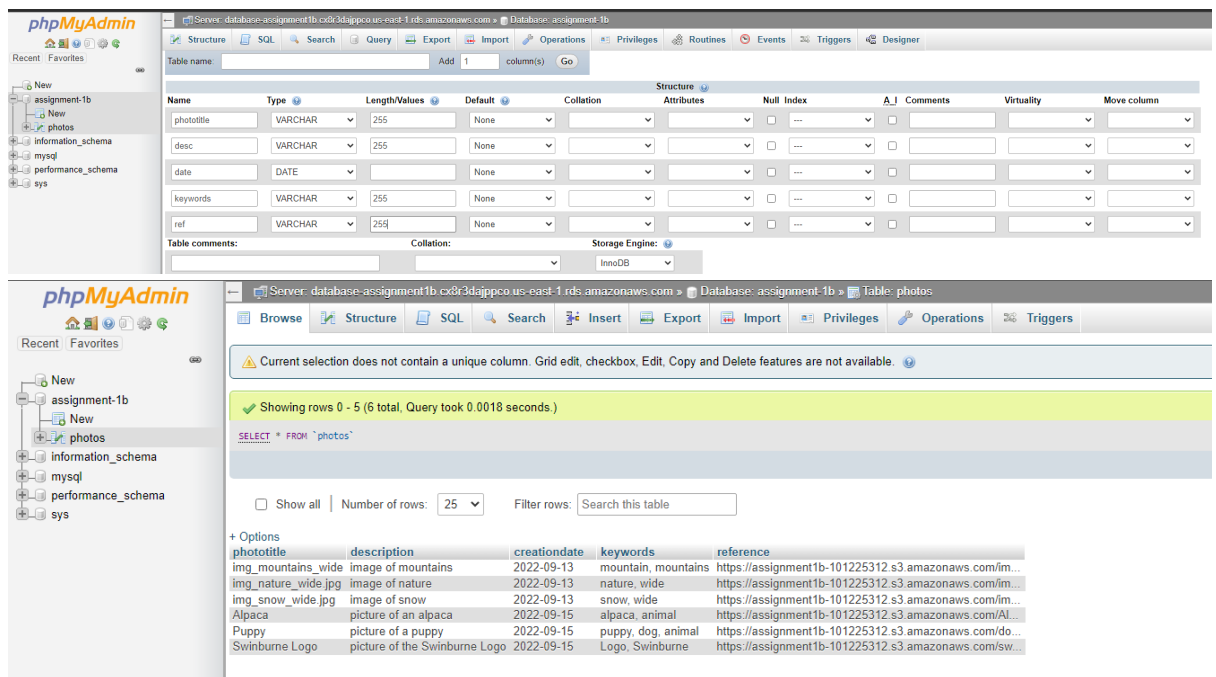


An S3 bucket was created to store the images. I looked up online and placed this statement in the bucket policy to allow public access to the files of the bucket as seen in the red bar that says “Publicly accessible” under the bucket name.



I also uploaded several photos to be used in the database.

7. Creating and populating database



The first screenshot shows the phpMyAdmin interface for a database named 'assignment-1b'. The 'Structure' tab is selected, displaying the table structure for 'photos'. The table has five columns: 'phototitle' (VARCHAR, 255), 'desc' (VARCHAR, 255), 'date' (DATE), 'keywords' (VARCHAR, 255), and 'ref' (VARCHAR, 255). The storage engine is InnoDB.

The second screenshot shows the 'Browse' tab for the 'photos' table. It displays a message: 'Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.' Below this, it shows 'Showing rows 0 - 5 (6 total, Query took 0.0018 seconds.)' and a SQL query: 'SELECT * FROM `photos`'. The table data is as follows:

phototitle	description	creationdate	keywords	reference
img_mountains_wide	image of mountains	2022-09-13	mountain, mountains	https://assignment1b-101225312.s3.amazonaws.com/im...
img_nature_wide.jpg	image of nature	2022-09-13	nature, wide	https://assignment1b-101225312.s3.amazonaws.com/im...
img_snow_wide.jpg	image of snow	2022-09-13	snow, wide	https://assignment1b-101225312.s3.amazonaws.com/im...
Alpaca	picture of an alpaca	2022-09-15	alpaca, animal	https://assignment1b-101225312.s3.amazonaws.com/Al...
Puppy	picture of a puppy	2022-09-15	puppy, dog, animal	https://assignment1b-101225312.s3.amazonaws.com/do...
Swinburne Logo	picture of the Swinburne Logo	2022-09-15	Logo, Swinburne	https://assignment1b-101225312.s3.amazonaws.com/sw...

I created a new relational database and a table with the following attributes. After that, I manually populated the table with data using the images I uploaded to the S3 bucket as reference.

8. Importing php source code

```
35 // [ACTION REQUIRED] your full name
36 define('STUDENT_NAME', 'Lai Yee Fung');
37 // [ACTION REQUIRED] your Student ID
38 define('STUDENT_ID', '101225312');
39 // [ACTION REQUIRED] your tutorial session
40 define('TUTORIAL_SESSION', 'Wednesday 6:30p.m.');
```

```
41
42 // [ACTION REQUIRED] name of the S3 bucket that stores images
43 define('BUCKET_NAME', 'assignment1b-101225312');
44 // [ACTION REQUIRED] region of the above bucket
45 define('REGION', 'us-east-1');
46 // no need to update this const
47 define('S3_BASE_URL', 'https://'.BUCKET_NAME.'.s3.amazonaws.com/');
```

```
48
49 // [ACTION REQUIRED] name of the database that stores photo meta-data (note that this is not the DB identifier of the RDS instance)
50 define('DB_NAME', 'assignment1b');
51 // [ACTION REQUIRED] endpoint of RDS instance
52 define('DB_ENDPOINT', 'database-assignment1b.cx8p3dajppco.us-east-1.rds.amazonaws.com');
53 // [ACTION REQUIRED] username of your RDS instance
54 define('DB_USERNAME', 'admin');
55 // [ACTION REQUIRED] password of your RDS instance
56 define('DB_PWD', 'assignment1b');
```

```
57
58 // [ACTION REQUIRED] name of the DB table that stores photo's meta-data
59 define('DB_PHOTO_TABLE_NAME', 'photos');
```

```
60 // The table above has 5 columns:
61 // [ACTION REQUIRED] name of the column in the above table that stores photo's titles
62 define('DB_PHOTO_TITLE_COL_NAME', 'phototitle');
```

```
63 // [ACTION REQUIRED] name of the column in the above table that stores photo's descriptions
64 define('DB_PHOTO_DESCRIPTION_COL_NAME', 'description');
```

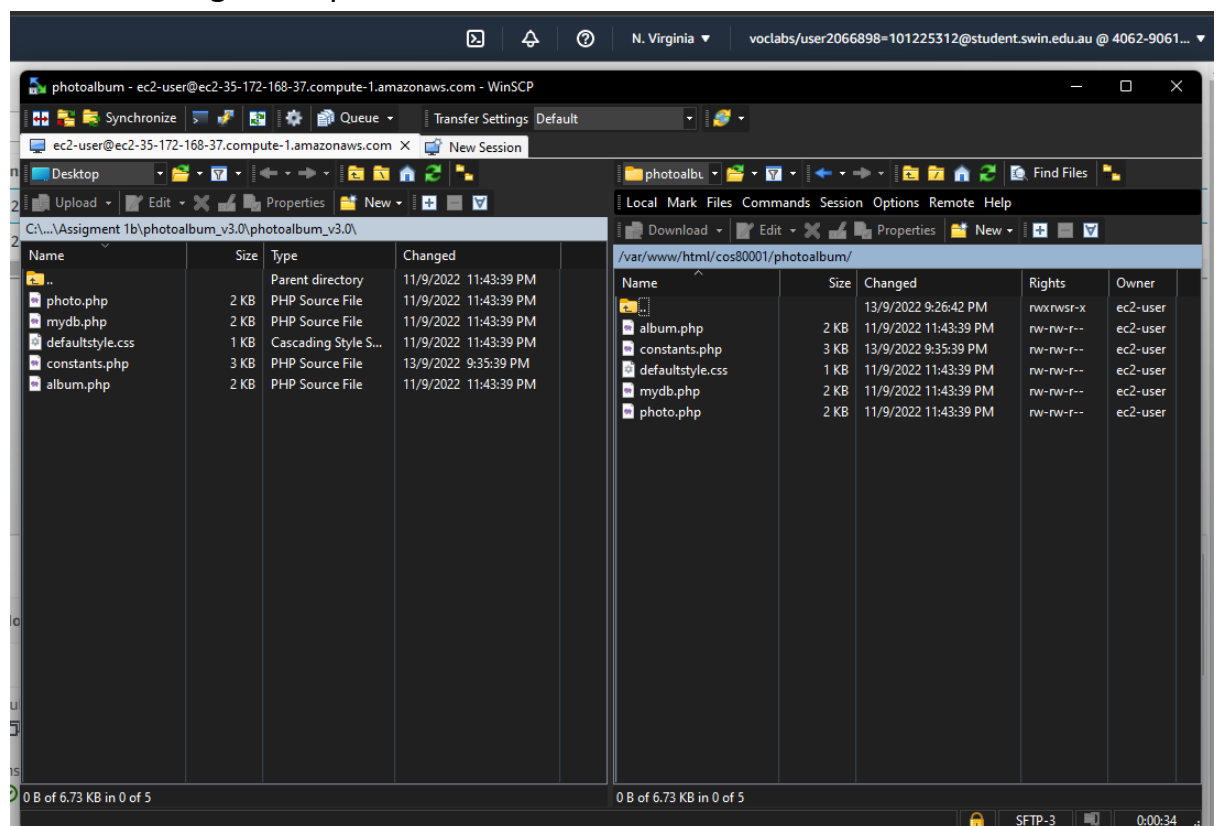
```
65 // [ACTION REQUIRED] name of the column in the above table that stores photo's creation dates
66 define('DB_PHOTO_CREATIONDATE_COL_NAME', 'creationdate');
```

```
67 // [ACTION REQUIRED] name of the column in the above table that stores photo's keywords
68 define('DB_PHOTO_KEYWORDS_COL_NAME', 'keywords');
```

```
69 // [ACTION REQUIRED] name of the column in the above table that stores photo's links in S3
70 define('DB_PHOTO_S3REFERENCE_COL_NAME', 'reference');
```

```
71 ?>
```

I edited the php source code to match the fields used in the database as well as the login and password for the database.




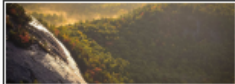

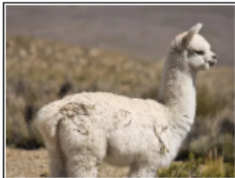

Using WinSCP, I imported all the php source files into the Bastion/Web Server instance. I also had to manually create the cos80001 directory. To check if it worked, I just had to go into the album page.

Student name: Lai Yee Fung

Student ID: 101225312

Tutorial session: Wednesday 6:30p.m.

Uploaded photos:

Photo	Name	Description	Creation date	Keywords
	img_mountains_wide	image of mountains	2022-09-13	mountain, mountains
	img_nature_wide.jpg	image of nature	2022-09-13	nature, wide
	img_snow_wide.jpg	image of snow	2022-09-13	snow, wide
	Alpaca	picture of an alpaca	2022-09-15	alpaca, animal
	Puppy	picture of a puppy	2022-09-15	puppy, dog, animal
	Swinburne Logo	picture of the Swinburne Logo	2022-09-15	Logo, Swinburne

9. Configuring NACL

The screenshot displays the AWS Management Console interface for configuring a Network ACL (NACL). The breadcrumb navigation shows the path: VPC > Network ACLs > acl-0aa7fd304d6418c34 / PublicSubnet2NACL. The title bar indicates the user is logged in as 'voclabs/user2066898-101225312@student.swin.edu.au' in the 'N. Virginia' region.

Network ACL Details:

- Network ACL ID:** acl-0aa7fd304d6418c34
- Associated with:** subnet-0edfeda3ece019b07 / 1b-public2-us-east-1b
- Default:** No
- VPC ID:** vpc-02e084850c18f1cc / YLaiVPC-vpc
- Owner:** 406290611783

Inbound rules (5):

Rule number	Type	Protocol	Port range	Source	Allow/Deny
1	SSH (22)	TCP (6)	22	0.0.0.0/0	Allow
2	All ICMP - IPv4	ICMP (1)	All	10.0.4.0/24	Allow
3	HTTP (80)	TCP (6)	80	0.0.0.0/0	Allow
5	All TCP	TCP (6)	All	0.0.0.0/0	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

Outbound rules (3):

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

NACL inbound and outbound rules are set as another layer of security above security groups. An issue I ran into here was with which rules I needed to set for the album page to work. As stated in the requirements, SSH and ICMP is allowed but the other types are not specified.

I tested with the ICMP inbound and outbound to private subnet 2 which worked after several tries by pinging in linux terminal through Putty. As for the other types, I figured it out through trial and error by testing whether the album page would load if I denied access to a certain connection type.

Overall, most of the parts were not too hard for this assignment as it had been done before in previous lab sessions. As for the new parts I had to do some minor searching from previous lab materials or on the web to find the solution.