Assignment 1B Checklist

Make sure all the following are completed.

Submission Checklist

Student Name: Dang Vi Luan

Student Id: 103802759

Tutorial time: 3/3/2023

Date of submission: 26/02/2023

Submit to Canvas:

A PDF document file as specified in the Submission section of the assignment specification.

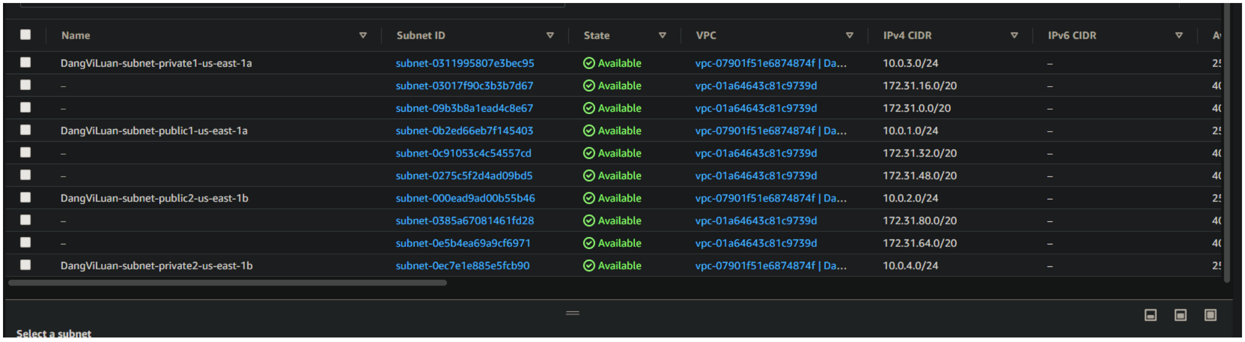
Marking Scheme

Infrastructure Requirements		
VPC with 2 public and 2 private subnets	.5	*Done*
Correct Public and Private Routing tables with correct subnet associations	1	*Done*
Security groups properly configured and attached.	1	*Done*
Network ACL properly configured and attached	1.5	*Done*
Correct Web server and Test instances running in correct subnets	.5	*Done*
Database schema as specified	.5	*Done*
Database running in correct subnets	1	*Done*
S3 objects publicly accessible, using proper access policy	.5	*Done*
Functional Requirements		
album.php page displayed from EC2 Web server	1	*Done*
Provided URL is persistent (Elastic IP Association)	.5	*Done*
Photos loaded from S3 with matching metadata from RDS	1	*Done*
Web server instance reachable from Test instance via ICMP	1	*Done*
Deductions		
Documentation not as specified or poorly presented (up to minus 20)		
Serious misconfigurations of AWS services being used (up to minus 20)		

Marking Scheme

1.1 Infrastructure deployment

The configuration of the subnet and VPC is fairly simple, as it was introduced in previous lab, the following pictures show the set up of 4 subnets and the VPC for this assignment.

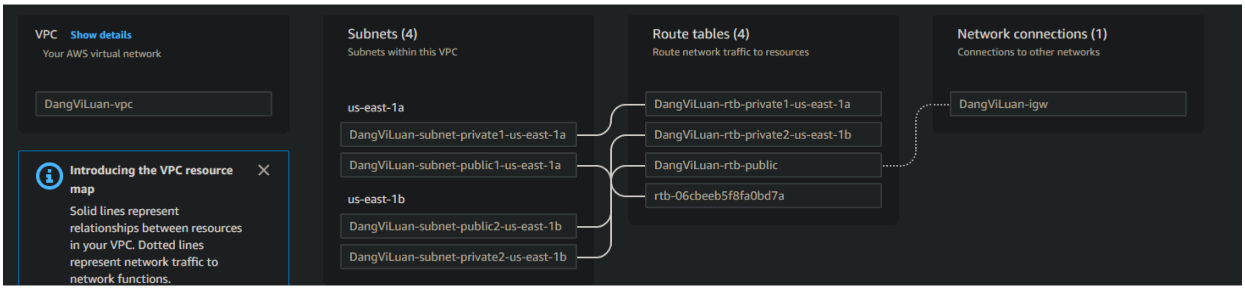


☐	Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR	A
☐	DangViLuan-subnet-private1-us-east-1a	subnet-0311995807e3bec95	✓ Available	vpc-07901f51e6874874f | Da...	10.0.3.0/24	–	25
☐	–	subnet-03017f90c3b3b7d67	✓ Available	vpc-01a64643c81c9739d	172.31.16.0/20	–	40
☐	–	subnet-09b3b8a1ead4c8e67	✓ Available	vpc-01a64643c81c9739d	172.31.0.0/20	–	40
☐	DangViLuan-subnet-public1-us-east-1a	subnet-0b2ed66eb7f145403	✓ Available	vpc-07901f51e6874874f | Da...	10.0.1.0/24	–	25
☐	–	subnet-0c91053c4c54557cd	✓ Available	vpc-01a64643c81c9739d	172.31.32.0/20	–	40
☐	–	subnet-0275c5f2d4ad09bd5	✓ Available	vpc-01a64643c81c9739d	172.31.48.0/20	–	40
☐	DangViLuan-subnet-public2-us-east-1b	subnet-000ead9ad00b55b46	✓ Available	vpc-07901f51e6874874f | Da...	10.0.2.0/24	–	25
☐	–	subnet-0385a67081461fd28	✓ Available	vpc-01a64643c81c9739d	172.31.80.0/20	–	40
☐	–	subnet-0e5b4ea69a9cf6971	✓ Available	vpc-01a64643c81c9739d	172.31.64.0/20	–	40
☐	DangViLuan-subnet-private2-us-east-1b	subnet-0ec7e1e885e5fcb90	✓ Available	vpc-07901f51e6874874f | Da...	10.0.4.0/24	–	25

Select a subnet

Picture 1: 4 subnets for the assignment

The subnets were configured according to the specification, each subnet is in 10.0.0.0/16 VPC and each of them has 251 hosts in its network pool. The routing table and VPC specification are as follows:



VPC Show details
Your AWS virtual network
DangViLuan-vpc

Introducing the VPC resource map
Solid lines represent relationships between resources in your VPC. Dotted lines represent network traffic to network functions.

Subnets (4)
Subnets within this VPC
us-east-1a
DangViLuan-subnet-private1-us-east-1a
DangViLuan-subnet-public1-us-east-1a
us-east-1b
DangViLuan-subnet-public2-us-east-1b
DangViLuan-subnet-private2-us-east-1b

Route tables (4)
Route network traffic to resources
DangViLuan-rtb-private1-us-east-1a
DangViLuan-rtb-private2-us-east-1b
DangViLuan-rtb-public
rtb-06cbeeb5f8fa0bd7a

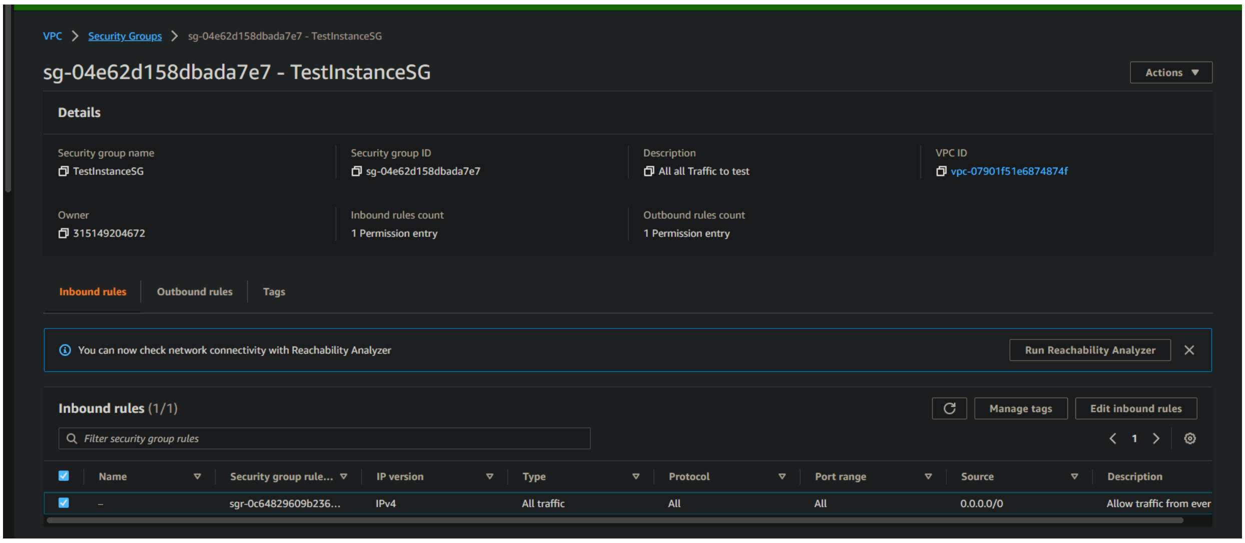
Network connections (1)
Connections to other networks
DangViLuan-igw

Picture 2: VPC specification for the assignment

There are 4 subnet (2 private and 2 public), as we will not be using Subnet Public 1 in this assignment, it is not connected to the internet gateway or any other network. Private 1 and Private 2 will be used as the back-end and testing network for our website. Therefore, they do not need to be connected to the internet gateway as well. Only subnet public 2, which hosts our photo album website, will be allocated with a routing table that connected to the Internet gateway for internet user to reach our website.

1.2 Security groups

The following are the security groups that will be used in this assigenemnt:


VPC > Security Groups > sg-04e62d158dbada7e7 - TestInstanceSG
Actions ▼
Inbound rules
Outbound rules
Tags
Run Reachability Analyzer
Manage tags
Edit inbound rules
Filter security group rules
< 1 >

sg-04e62d158dbada7e7 - TestInstanceSG

Details

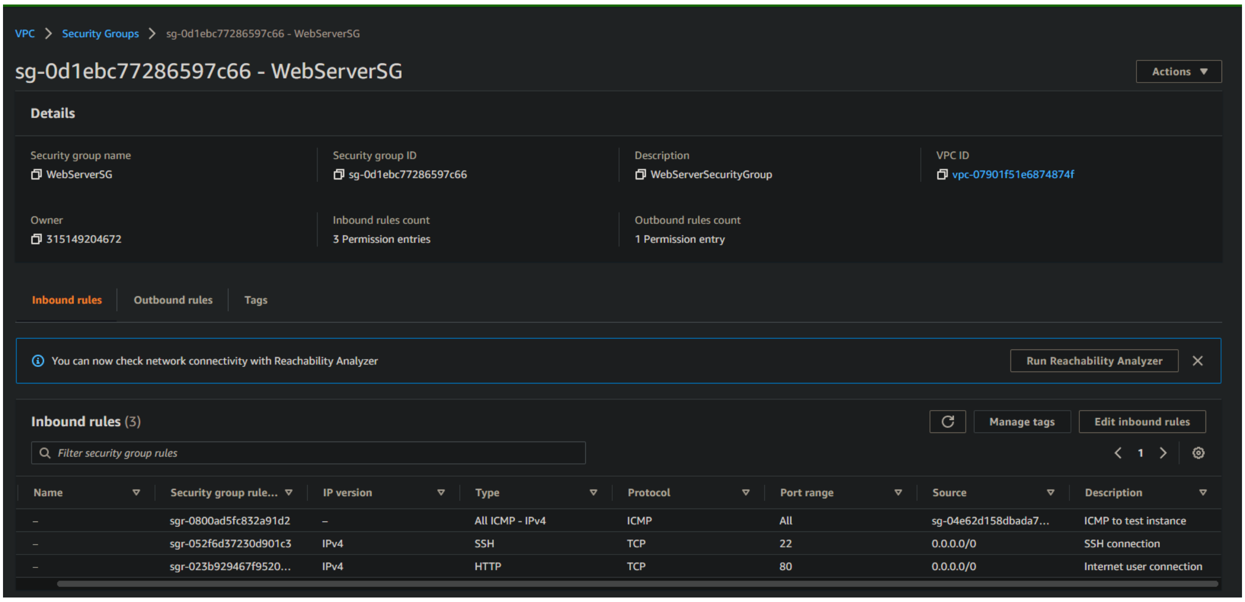
Security group name TestInstanceSG	Security group ID sg-04e62d158dbada7e7	Description All all Traffic to test	VPC ID vpc-07901f51e6874874f
Owner 315149204672	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

You can now check network connectivity with Reachability Analyzer

Inbound rules (1/1)

☑	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
☑	–	sgr-0c64829609b236...	IPv4	All traffic	All	All	0.0.0.0/0	Allow traffic from ever

Picture 3: Security group for TestInstace


VPC > Security Groups > sg-0d1ebc77286597c66 - WebServerSG
Actions ▼
Inbound rules
Outbound rules
Tags
Run Reachability Analyzer
Manage tags
Edit inbound rules
Filter security group rules
< 1 >

sg-0d1ebc77286597c66 - WebServerSG

Details

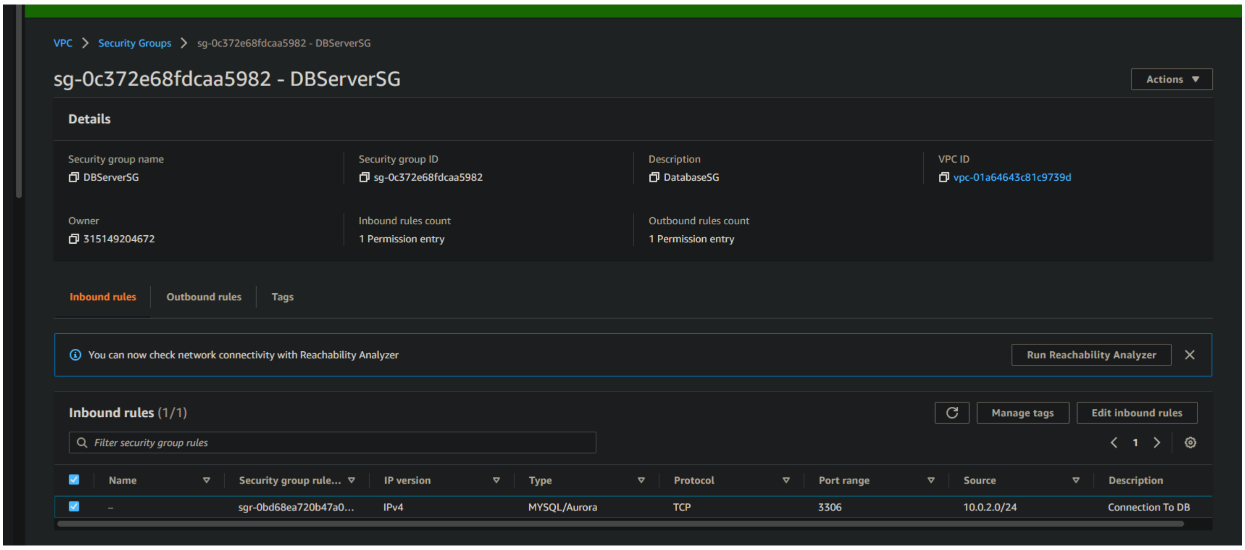
Security group name WebServerSG	Security group ID sg-0d1ebc77286597c66	Description WebServerSecurityGroup	VPC ID vpc-07901f51e6874874f
Owner 315149204672	Inbound rules count 3 Permission entries	Outbound rules count 1 Permission entry	

You can now check network connectivity with Reachability Analyzer

Inbound rules (3)

Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
–	sgr-0800ad5fc832a91d2	–	All ICMP - IPv4	ICMP	All	sg-04e62d158dbada7...	ICMP to test instance
–	sgr-052f6d37230d901c3	IPv4	SSH	TCP	22	0.0.0.0/0	SSH connection
–	sgr-023b929467f9520...	IPv4	HTTP	TCP	80	0.0.0.0/0	Internet user connection

Picture 4: Security group for WebServer


VPC > Security Groups > sg-0c372e68fdcaa5982 - DBServerSG
Actions ▼
Inbound rules
Outbound rules
Tags
Run Reachability Analyzer
Manage tags
Edit inbound rules
Filter security group rules
< 1 >

sg-0c372e68fdcaa5982 - DBServerSG

Details

Security group name DBServerSG	Security group ID sg-0c372e68fdcaa5982	Description DatabaseSG	VPC ID vpc-01a64643c81c9739d
Owner 315149204672	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

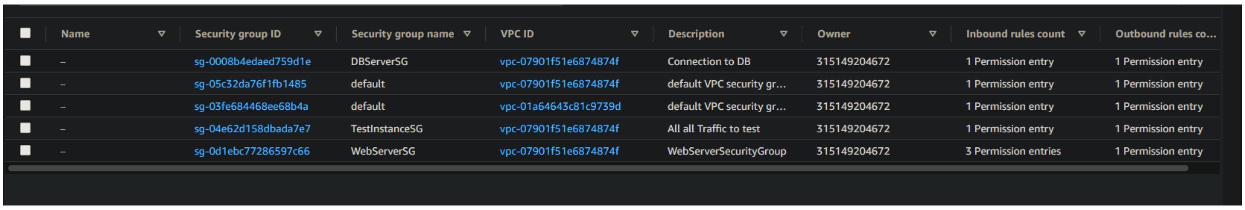
You can now check network connectivity with Reachability Analyzer

Inbound rules (1/1)

☑	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
☑	–	sgr-0bd68ea720b47a0...	IPv4	MYSQL/Aurora	TCP	3306	10.0.2.0/24	Connection To DB

Picture 5: Security group for DBServer

All 3 security groups are then allocated to our previously created VPC:

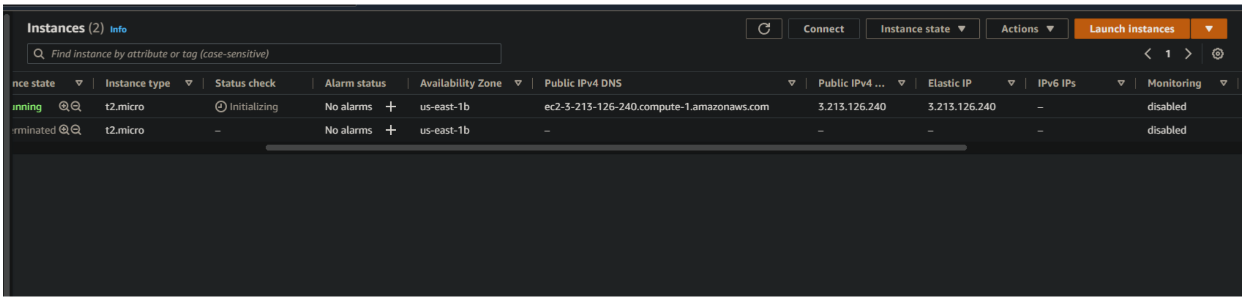


□	Name	Security group ID	Security group name	VPC ID	Description	Owner	Inbound rules count	Outbound rules co...
□	–	sg-0008b4edaed759d1e	DBServerSG	vpc-07901f51e6874874f	Connection to DB	315149204672	1 Permission entry	1 Permission entry
□	–	sg-05c32da76f1fb1485	default	vpc-07901f51e6874874f	default VPC security gr...	315149204672	1 Permission entry	1 Permission entry
□	–	sg-03fe684468ee68b4a	default	vpc-01a64643c81c9739d	default VPC security gr...	315149204672	1 Permission entry	1 Permission entry
□	–	sg-04e62d158dbada7e7	TestInstanceSG	vpc-07901f51e6874874f	All all Traffic to test	315149204672	1 Permission entry	1 Permission entry
□	–	sg-0d1ebc77286597c66	WebServerSG	vpc-07901f51e6874874f	WebServerSecurityGroup	315149204672	3 Permission entries	1 Permission entry

Picture 6: All the security groups for VPC

1.3 EC2 Virtual Machine

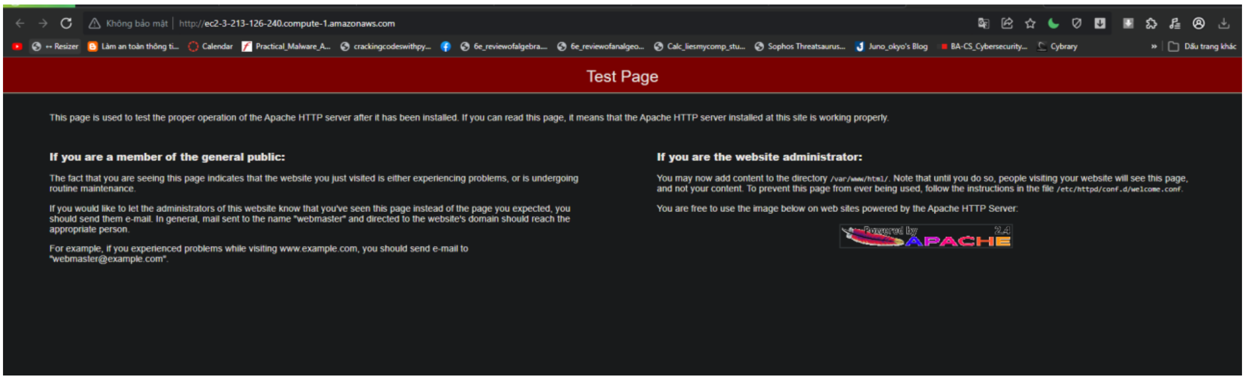
After all the preparation, we can now launch our instance and create our Bastion Web Server.



nce state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs	Monitoring
ınning	t2.micro	Initializing	No alarms +	us-east-1b	ec2-3-213-126-240.compute-1.amazonaws.com	3.213.126.240	3.213.126.240	–	disabled
:rminated	t2.micro	–	No alarms +	us-east-1b	–	–	–	–	disabled

Picture 7: EC2 Instance for the assignment

It is important to assign our VPC with an Elastic IP so that its DNS public IP will not be released after each time we reset. After launching the instance, we can browse our website using the public DNS address:



Test Page

This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".

If you are the website administrator:

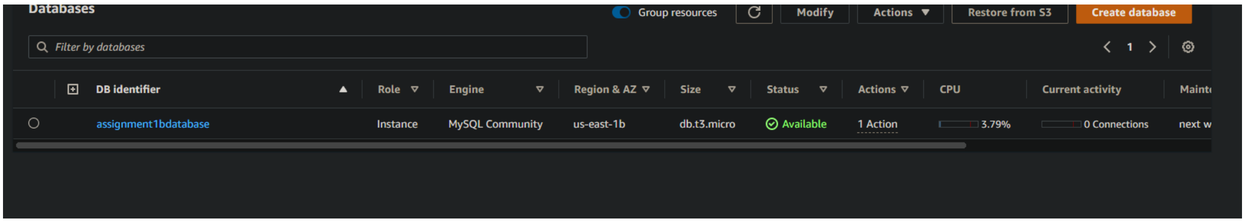
You may now add content to the directory `/var/www/html/`. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

You are free to use the image below on web sites powered by the Apache HTTP Server:

Picture 8: Test the instance

1.4 RDS database instance

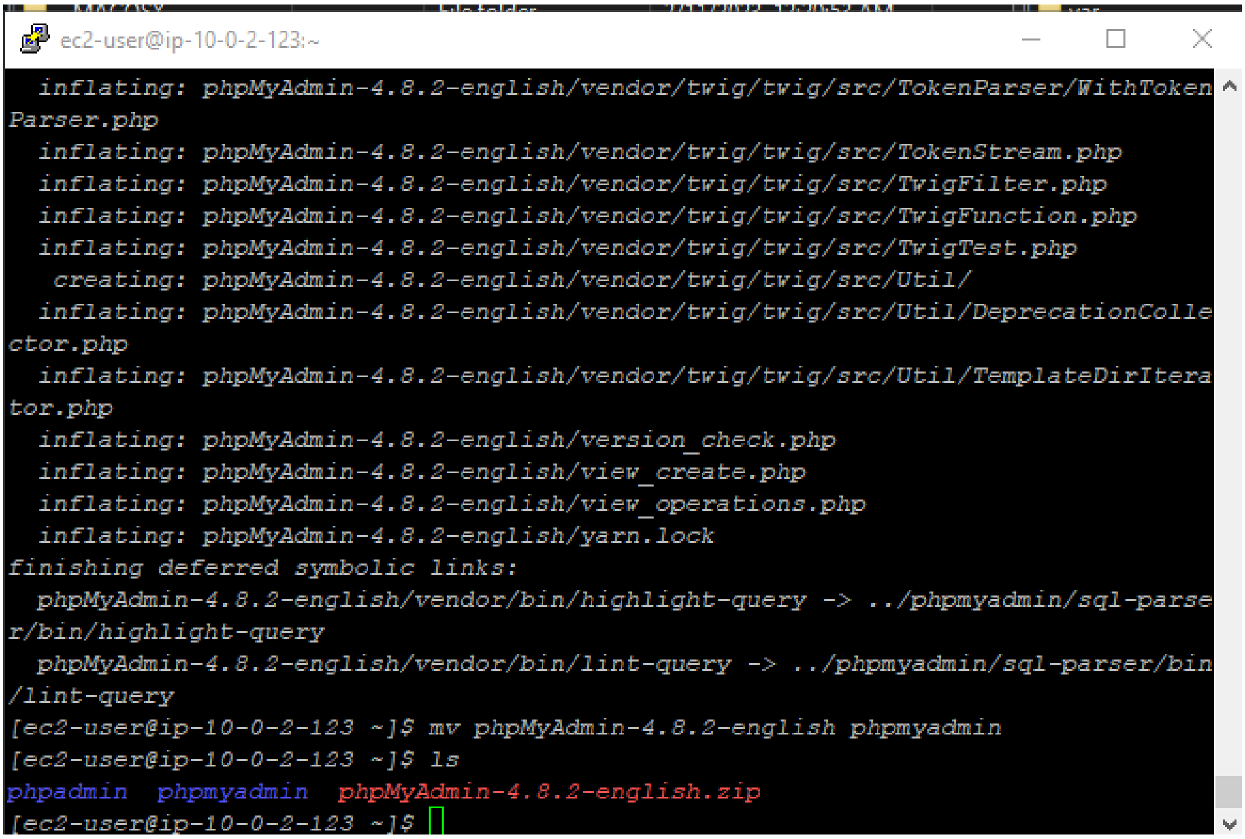
We will also need to create an RDS instance as an database platform for our website.



	DB identifier	Role	Engine	Region & AZ	Size	Status	Actions	CPU	Current activity	Maint
○	assignment1bdatabase	Instance	MySQL Community	us-east-1b	db.t3.micro	✓ Available	1 Action	3.79%	0 Connections	next w

Picture 9: RDS instance for the assignment

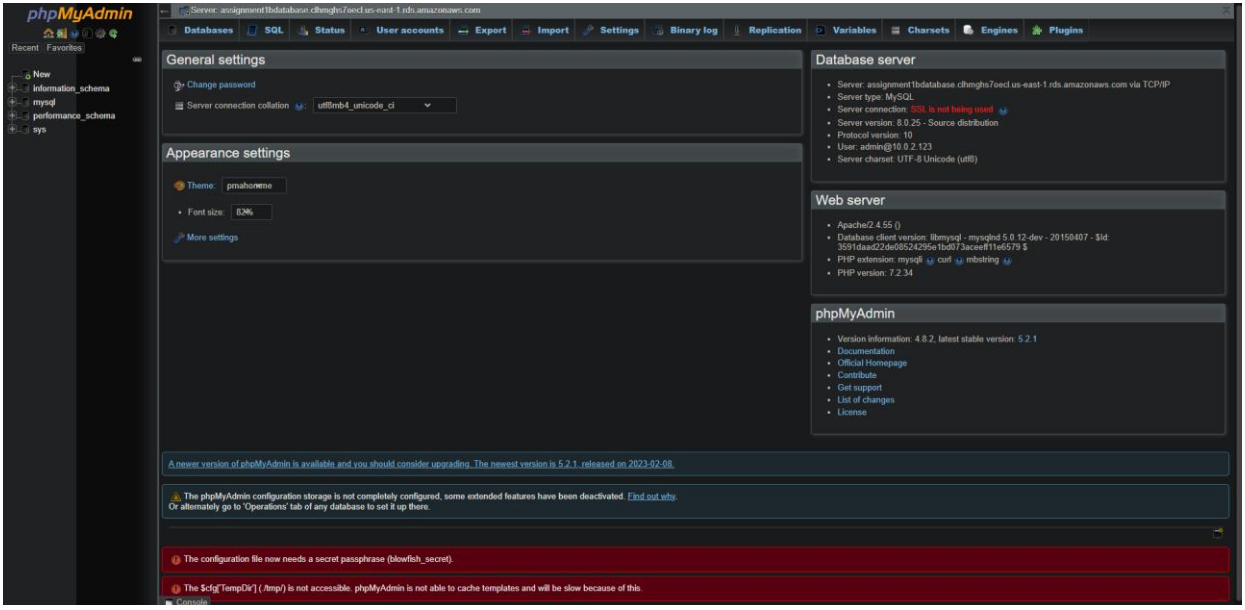
The RDS instance need to be access over the internet so that we can set it up and maintain it, we can do this by installing phpMyAdmin on our EC2 website.



```
  inflating: phpMyAdmin-4.8.2-english/vendor/twig/twig/src/TokenParser/WithToken
Parser.php
  inflating: phpMyAdmin-4.8.2-english/vendor/twig/twig/src/TokenStream.php
  inflating: phpMyAdmin-4.8.2-english/vendor/twig/twig/src/TwigFilter.php
  inflating: phpMyAdmin-4.8.2-english/vendor/twig/twig/src/TwigFunction.php
  inflating: phpMyAdmin-4.8.2-english/vendor/twig/twig/src/TwigTest.php
   creating: phpMyAdmin-4.8.2-english/vendor/twig/twig/src/Util/
  inflating: phpMyAdmin-4.8.2-english/vendor/twig/twig/src/Util/DeprecationColle
ctor.php
  inflating: phpMyAdmin-4.8.2-english/vendor/twig/twig/src/Util/TemplateDirItera
tor.php
  inflating: phpMyAdmin-4.8.2-english/version_check.php
  inflating: phpMyAdmin-4.8.2-english/view_create.php
  inflating: phpMyAdmin-4.8.2-english/view_operations.php
  inflating: phpMyAdmin-4.8.2-english/yarn.lock
finishing deferred symbolic links:
  phpMyAdmin-4.8.2-english/vendor/bin/highlight-query -> ../phpmyadmin/sql-parse
r/bin/highlight-query
  phpMyAdmin-4.8.2-english/vendor/bin/lint-query -> ../phpmyadmin/sql-parser/bin
/lint-query
[ec2-user@ip-10-0-2-123 ~]$ mv phpMyAdmin-4.8.2-english phpmyadmin
[ec2-user@ip-10-0-2-123 ~]$ ls
phpadmin  phpmyadmin  phpMyAdmin-4.8.2-english.zip
[ec2-user@ip-10-0-2-123 ~]$
```

Picture 10: phpMyAdmin installed on EC2 Instance

After that, we can log onto our phpMyAdmin console via our public DNS address and proceed to create our database.



Server: assignment1bdatabase.clhmghs7oecl.us-east-1.rds.amazonaws.com

General settings

- Change password
- Server connection collation: utf8mb4_unicode_ci

Appearance settings

- Theme: pmahomme
- Font size: 82%
- More settings

Database server

- Server: assignment1bdatabase.clhmghs7oecl.us-east-1.rds.amazonaws.com via TCP/IP
- Server type: MySQL
- Server connection: SSL is not being used
- Server version: 8.0.25 - Source distribution
- Protocol version: 10
- User: admin@10.0.2.123
- Server charset: UTF-8 Unicode (utf8)

Web server

- Apache/2.4.55 ()
- Database client version: libmysql - mysqlnd 5.0.12-dev - 20150407 - $Id: 3591daad22de08524295e1bd073aceeff11e6579 $
- PHP extension: mysqli curl mbstring
- PHP version: 7.2.34

phpMyAdmin

- Version information: 4.8.2, latest stable version: 5.2.1
- Documentation
- Official Homepage
- Contribute
- Get support
- List of changes
- License

A newer version of phpMyAdmin is available and you should consider upgrading. The newest version is 5.2.1, released on 2023-02-08.

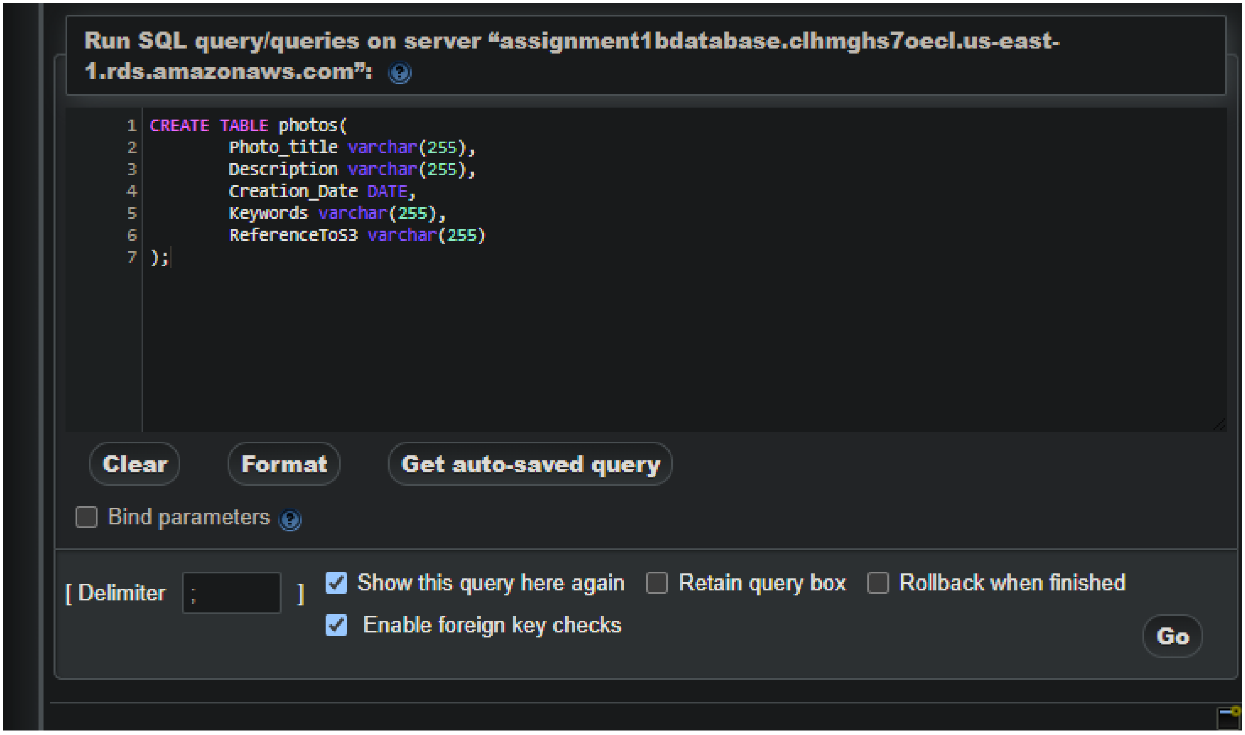
The phpMyAdmin configuration storage is not completely configured, some extended features have been deactivated. Find out why.
Or alternately go to 'Operations' tab of any database to set it up there.

The configuration file now needs a secret passphrase (blowfish_secret).

The $cfg['TempDir'] (./tmp/) is not accessible. phpMyAdmin is not able to cache templates and will be slow because of this.

Picture 11: phpMyAdmin console

Using some simple SQL command we now have a database for our website.



Run SQL query/queries on server "assignment1bdatabase.clhmghs7oecl.us-east-1.rds.amazonaws.com":

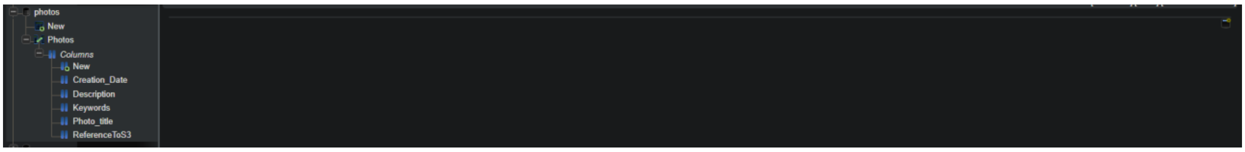
```sql
CREATE TABLE photos(
        Photo_title varchar(255),
        Description varchar(255),
        Creation_Date DATE,
        Keywords varchar(255),
        ReferenceToS3 varchar(255)
);
```

□ Bind parameters

[Delimiter ;] ☑ Show this query here again □ Retain query box □ Rollback when finished

☑ Enable foreign key checks

Picture 12: Create table for photos

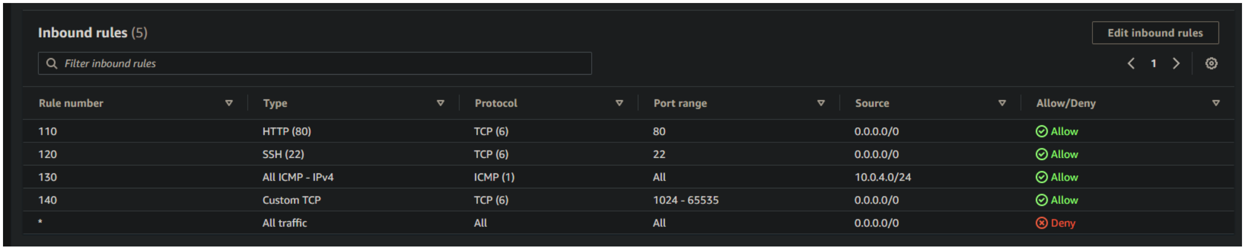


```
photos
  New
  Photos
    Columns
      New
      Creation_Date
      Description
      Keywords
      Photo_title
      ReferenceToS3
```

Picture 13: Table Photos

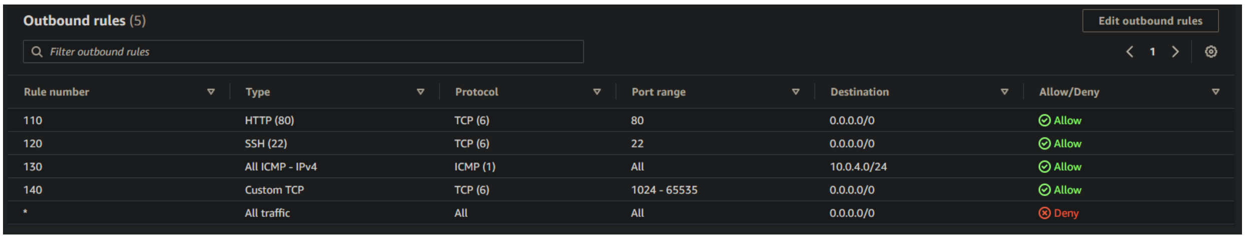
1.5 Network ACL

Network Access Control List is crucial for controlling specific inbound or outbound traffic at the subnet level of our VPC. We will set up the Network ACL in this step.



Rule number	Type	Protocol	Port range	Source	Allow/Deny
110	HTTP (80)	TCP (6)	80	0.0.0.0/0	✓ Allow
120	SSH (22)	TCP (6)	22	0.0.0.0/0	✓ Allow
130	All ICMP - IPv4	ICMP (1)	All	10.0.4.0/24	✓ Allow
140	Custom TCP	TCP (6)	1024 - 65535	0.0.0.0/0	✓ Allow
*	All traffic	All	All	0.0.0.0/0	⊗ Deny

Picture 14: Inbound rules for Network ACL

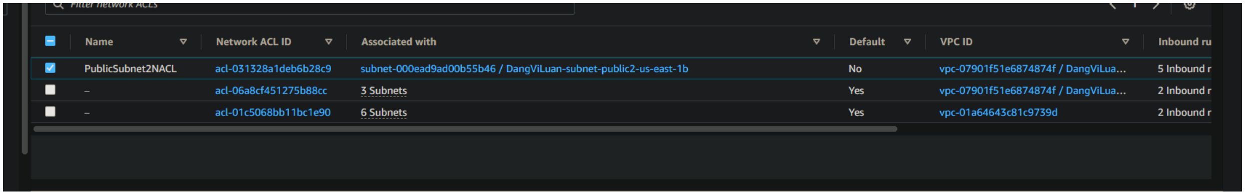


Rule number	Type	Protocol	Port range	Destination	Allow/Deny
110	HTTP (80)	TCP (6)	80	0.0.0.0/0	✓ Allow
120	SSH (22)	TCP (6)	22	0.0.0.0/0	✓ Allow
130	All ICMP - IPv4	ICMP (1)	All	10.0.4.0/24	✓ Allow
140	Custom TCP	TCP (6)	1024 - 65535	0.0.0.0/0	✓ Allow
*	All traffic	All	All	0.0.0.0/0	⊗ Deny

Picture 15: Outbound rules for Network ACL

It is noteworthy that we need to set up an ephemeral ports so that the service running on an instacne is accessible for internet user.

After that we can assign Network ACL to the designated subnet.

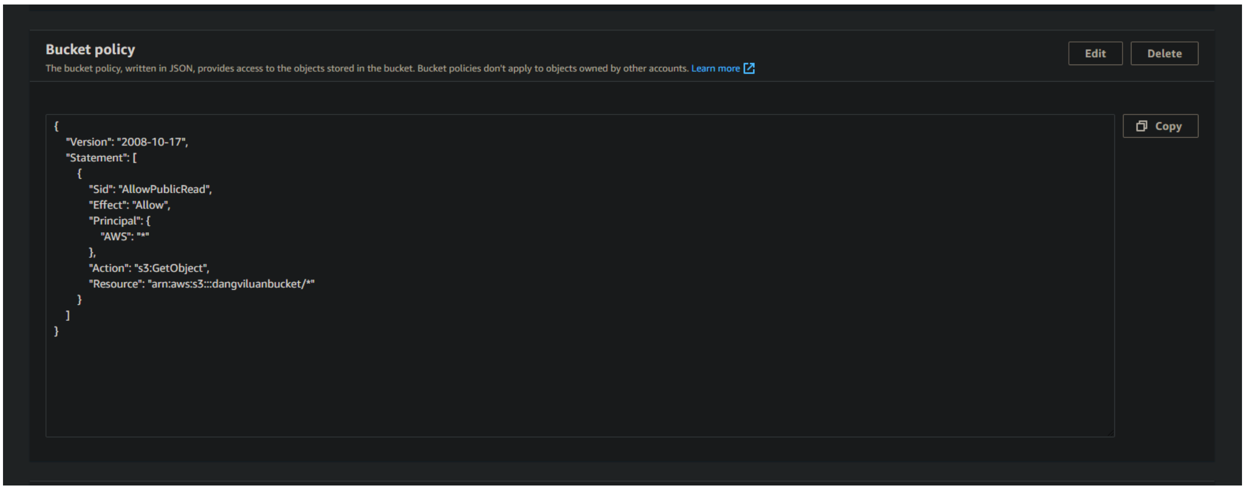


	Name	Network ACL ID	Associated with	Default	VPC ID	Inbound ru
☑	PublicSubnet2NACL	acl-031328a1deb6b28c9	subnet-000ead9ad00b55b46 / DangViLuan-subnet-public2-us-east-1b	No	vpc-07901f51e6874874f / DangViLua...	5 Inbound r
□	–	acl-06a8cf451275b88cc	3 Subnets	Yes	vpc-07901f51e6874874f / DangViLua...	2 Inbound r
□	–	acl-01c5068bb11bc1e90	6 Subnets	Yes	vpc-01a64643c81c9739d	2 Inbound r

Picture 16: Network ACL is assigned to Public subnet 2

2.1 Infrastructure deployment

An S3 instance will be needed to host the picture, we will also need to provide a bucket policy for our picture to be accessed by everyone



Bucket policy

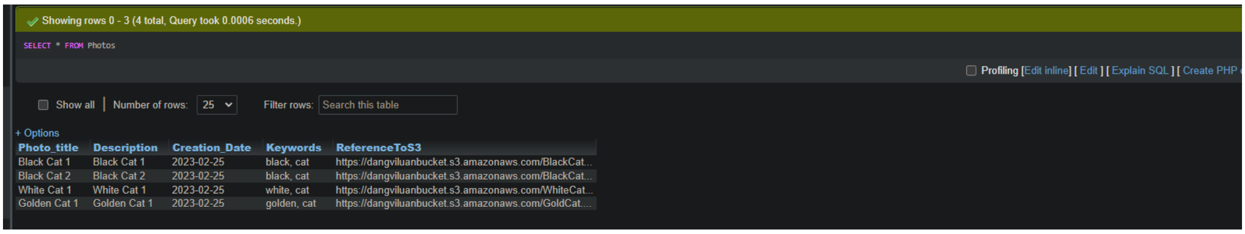
The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

```
{
    "Version": "2008-10-17",
    "Statement": [
        {
            "Sid": "AllowPublicRead",
            "Effect": "Allow",
            "Principal": {
                "AWS": "*"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::dangviluanbucket/*"
        }
    ]
}
```

Picture 17: Bucket policy for S3 Instance

2.2 Photo meta-data in RDS database

We will then populate some meta data in myphpadmin console



✓ Showing rows 0 - 3 (4 total, Query took 0.0006 seconds.)

```
SELECT * FROM Photos
```

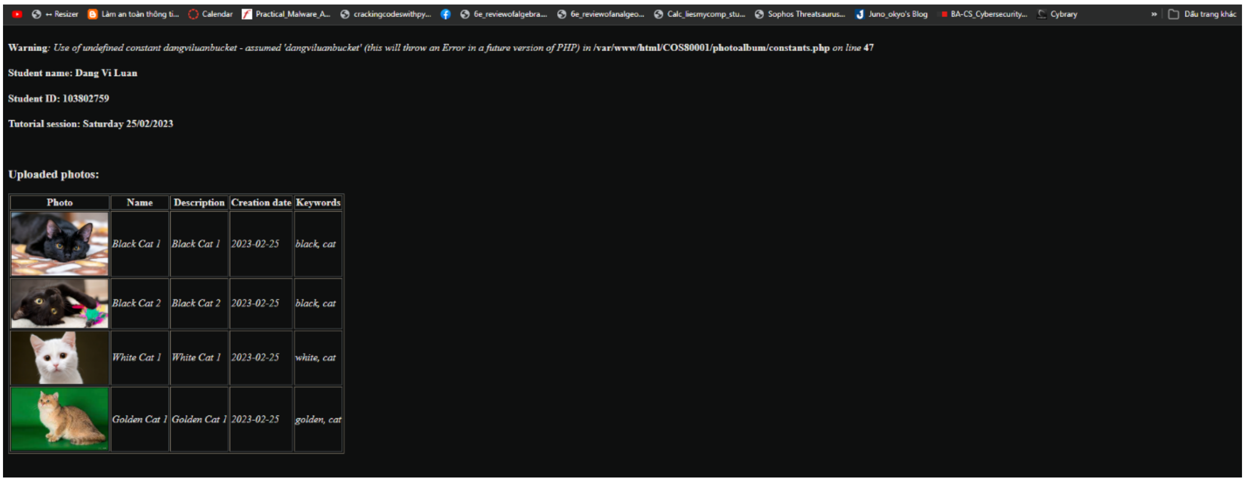
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP
□ Show all | Number of rows: 25
Filter rows: Search this table
+ Options

Photo_title	Description	Creation_Date	Keywords	ReferenceToS3
Black Cat 1	Black Cat 1	2023-02-25	black, cat	https://dangviluanbucket.s3.amazonaws.com/BlackCat...
Black Cat 2	Black Cat 2	2023-02-25	black, cat	https://dangviluanbucket.s3.amazonaws.com/BlackCat...
White Cat 1	White Cat 1	2023-02-25	white, cat	https://dangviluanbucket.s3.amazonaws.com/WhiteCat...
Golden Cat 1	Golden Cat 1	2023-02-25	golden, cat	https://dangviluanbucket.s3.amazonaws.com/GoldCat....

Picture 18: Meta-data for the website

2.3 Photo Album website functionality

After that we can check the functionality of our website



Warning: *Use of undefined constant dangviluanbucket - assumed 'dangviluanbucket' (this will throw an Error in a future version of PHP) in* **/var/www/html/COS80001/photoalbum/constants.php** *on line* **47**

Student name: Dang Vi Luan

Student ID: 103802759

Tutorial session: Saturday 25/02/2023

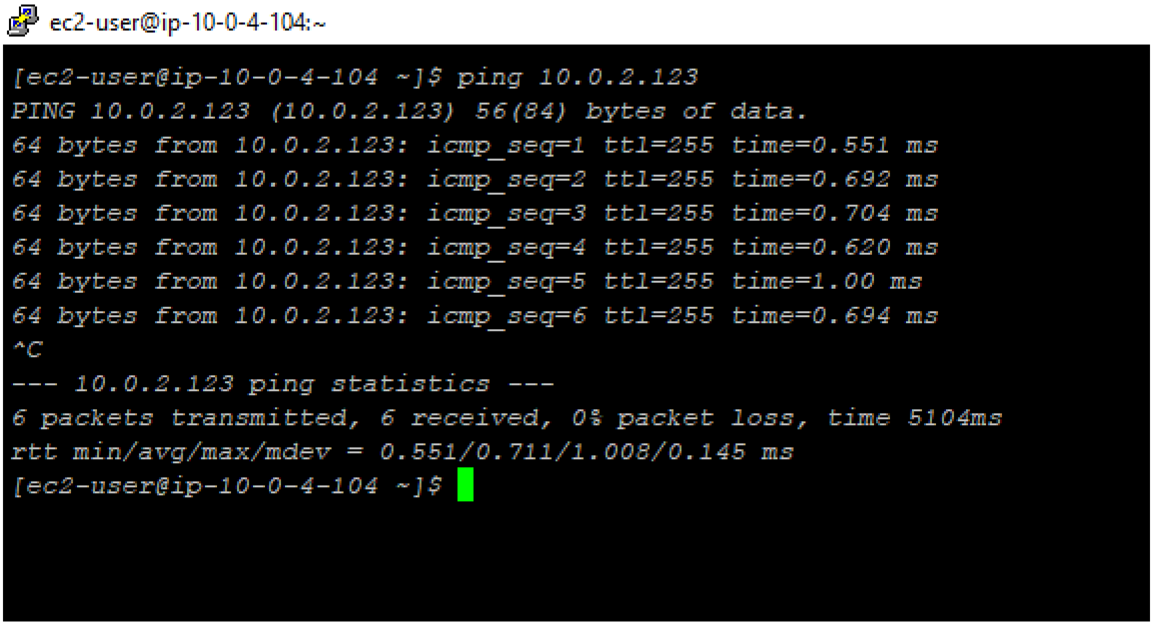
Uploaded photos:

Photo	Name	Description	Creation date	Keywords
	Black Cat 1	*Black Cat 1*	*2023-02-25*	*black, cat*
	Black Cat 2	*Black Cat 2*	*2023-02-25*	*black, cat*
	White Cat 1	*White Cat 1*	*2023-02-25*	*white, cat*
	Golden Cat 1	*Golden Cat 1*	*2023-02-25*	*golden, cat*

Picture 19: Functionality of the website

2.4 Testing

To check our website, we can go to our private subnet 2 and send and ICMP packet to our website



```
[ec2-user@ip-10-0-4-104 ~]$ ping 10.0.2.123
PING 10.0.2.123 (10.0.2.123) 56(84) bytes of data.
64 bytes from 10.0.2.123: icmp_seq=1 ttl=255 time=0.551 ms
64 bytes from 10.0.2.123: icmp_seq=2 ttl=255 time=0.692 ms
64 bytes from 10.0.2.123: icmp_seq=3 ttl=255 time=0.704 ms
64 bytes from 10.0.2.123: icmp_seq=4 ttl=255 time=0.620 ms
64 bytes from 10.0.2.123: icmp_seq=5 ttl=255 time=1.00 ms
64 bytes from 10.0.2.123: icmp_seq=6 ttl=255 time=0.694 ms
^C
--- 10.0.2.123 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5104ms
rtt min/avg/max/mdev = 0.551/0.711/1.008/0.145 ms
[ec2-user@ip-10-0-4-104 ~]$ 
```

Picture 20: ICMP sent successfully

3. Additional information for marking

EC2 link to album.php: http://ec2-3-213-126-240.compute1.amazonaws.com/COS80001/photoalbum/album.php

EC2 link to phpmyadmin: http://ec2-54-224-224-191.compute-1.amazonaws.com/phpmyadmin/

Comments