Kezia Somerville-Petch Student ID: 8092794

COSC349 Assignment 2 - Final Report:

Development Process Explanation:

Initial Development Process:

When working to develop a multi VM application deployed to the AWS cloud, I initially focused on the AWS COSC349 lab work as a starting point to allow me to understand the Amazon AWS Educate platform.

I initially focused on the content present within laboratory 8. This laboratory focused on introducing students to the S3 services of AWS through creating a static website and integrating our AWS credentials locally through command line instructions. I progressed through this lab rapidly and understood the concepts presented. I first encountered issues when attempting to install the AWS Command Line Interface (CLI) and access AWS services via the CLI. Regardless of using my current (recently refreshed) AWS access key ID, secret access key and session token, the installation of the AWS CLI never worked properly thus preventing me from interacting with the AWS services via the CLI.

Laboratory 9 involved deploying virtual machines (VMs) into the Amazon EC2 service using Vagrant and further exposed students to ssh connection to the instances created. I initially struggled with ensuring I had the correct subnet ID to include in my Vagrantfile, as my AWS commands were unable to be accessed via the CLI thus the command used to get this ID (aws ec2 describe-subnets --region us-east-1)was unable to be utilised. After completing all the remaining lab exercises a “request expired” error would occur when I attempted to “vagrant up” the created VM, thus preventing me from accessing the VM via the command line. This prevented me from determining if outgoing network access was viable and prevented me from ssh connecting to the created VM. Local testing of the created VM was also not able to be implemented.

Resources used:

Following my initial focus on the COSC349 lab content, I sought after AWS tutorials and videos to help me better understand each AWS service so the development of my Assignment 2 application could progress. The helpful tutorials and videos I attempted to utilise through this development process included:

Tutorial: Connecting to a DB Instance Running the MySQL Database Engine
https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_ConnectToInstance.html

Tutorial: Create an EC2 Instance and Install a Web Server
https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/CHAP_Tutorials.WebServerDB.CreateWebServer.html?fbclid=IwAR2kYSrvUUwFs2xwoLy9V6FUdAAiiwAaP-Itug7GVDrxckXPO9DiNoQGrxo

Tutorial: How to Create a MySQL Database:
https://aws.amazon.com/getting-started/hands-on/create-mysql-db/?fbclid=IwAR350oJ7TgmTrmd0-oot_I4qHIYRqc1Pqgy3LLQBVhc4ZbfvMAlHbLbqXKs

Video: "How to host a website using AWS EC2" - Open Source For Geeks:
https://www.youtube.com/watch?v=7ijHFjX82BI&fbclid=IwAR0wbs1GWutOYryiEuX7tZXOzL6L2Hve677cV0F9GSeMAZs5_8noGKZQQ3c&app=desktop

Video: "How to spin up an AWS EC2 instance" - Open Source For Geeks
https://www.youtube.com/watch?v=6yYoSOhjxhk

Tutorial: Create an Amazon VPC for Use with a DB Instance
https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/CHAP_Tutorials.WebServerDB.CreateVPC.html

Tutorial: Create a Web Server and an Amazon RDS DB Instance
https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/TUT_WebAppWithRDS.html

I found this final tutorial to be extremely beneficial and was something I seriously attempted to utilise in my assignment 2. The final step in this tutorial was to "Create an EC2 Instance and Install a Web Server". I thoroughly struggled with installing the web server on the EC2 instance through using command line instructions and was unsure if these instructions were even required due to the difference in command line actions taken in comparison with lab 4.

Through this tutorial I managed to create a Virtual Private Cloud (VPC), named cosc349a2-vpc, a database, named cosc349a2-correct-vpc and EC2 instance, named cosc349a2-webserver, all of which are still running on my personal AWS Educate Account. This tutorial also provided me with information on how to create separate public and private subnets that I attempted to utilise in specific instances.

Attempts were made, through the aforementioned tutorial, to connect the database with the EC2 instance, while making both of these instances be present in the VPC, though a functional web server that used the information within the created database was not achieved.

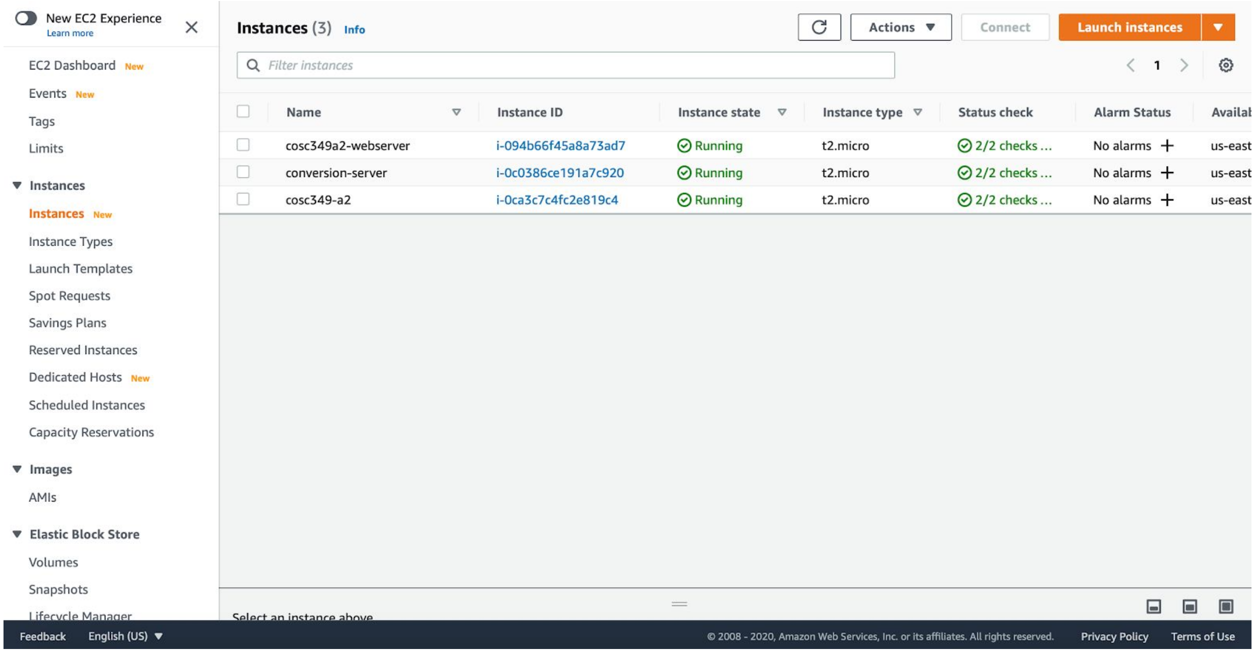
Further Development:

After communication with my lecturer and explaining the struggles I faced, he recommended for me to revert back to an earlier laboratory, lab 4, which taught students to create a simple VM and connect to that VM via ssh.

This laboratory helped me create a functioning web server application, though I struggled with finding where to place my created index.php file in order to allow that to be displayed on the created website, even when I ensured that the required PHP software packages were downloaded in the required place. The current web page, displayed by the cosc349-a2 EC2 instance, displays the default Apache test page, named the “Amazon Linux AMI Test Page”.

Current AWS State - Instances Running and Services used:

EC2 Instances:



Instances (3) Info

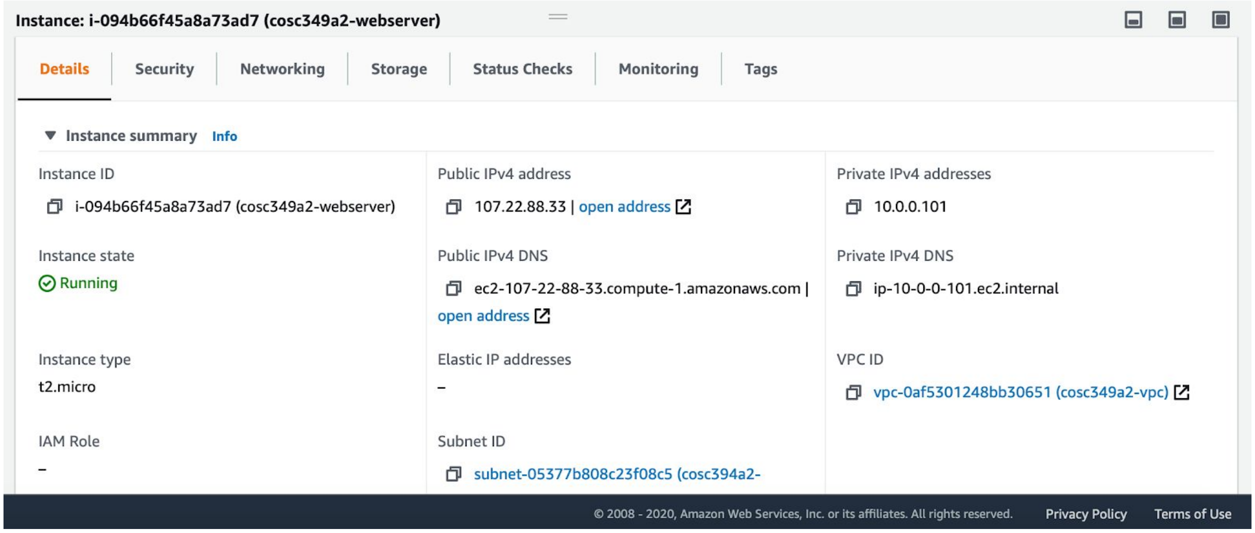
Name	Instance ID	Instance state	Instance type	Status check	Alarm Status	Availal
cosc349a2-webserver	i-094b66f45a8a73ad7	✓ Running	t2.micro	✓ 2/2 checks ...	No alarms +	us-east
conversion-server	i-0c0386ce191a7c920	✓ Running	t2.micro	✓ 2/2 checks ...	No alarms +	us-east
cosc349-a2	i-0ca3c7c4fc2e819c4	✓ Running	t2.micro	✓ 2/2 checks ...	No alarms +	us-east

Select an instance above

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Cosc349a2-webserver Instance:

- This instance was created during the **“Create a Web Server and an Amazon RDS DB Instance”** tutorial. This instance was created to connect with the cosc349a2-correct-VPC database (mentioned below), in order to pull information from the database so it could be displayed on a webpage, powered by this instance.



Instance: i-094b66f45a8a73ad7 (cosc349a2-webserver)

Instance summary Info

Instance ID: i-094b66f45a8a73ad7 (cosc349a2-webserver)

Public IPv4 address: 107.22.88.33 | open address

Private IPv4 addresses: 10.0.0.101

Instance state: ✓ Running

Public IPv4 DNS: ec2-107-22-88-33.compute-1.amazonaws.com | open address

Private IPv4 DNS: ip-10-0-0-101.ec2.internal

Instance type: t2.micro

Elastic IP addresses: –

VPC ID: vpc-0af5301248bb30651 (cosc349a2-vpc)

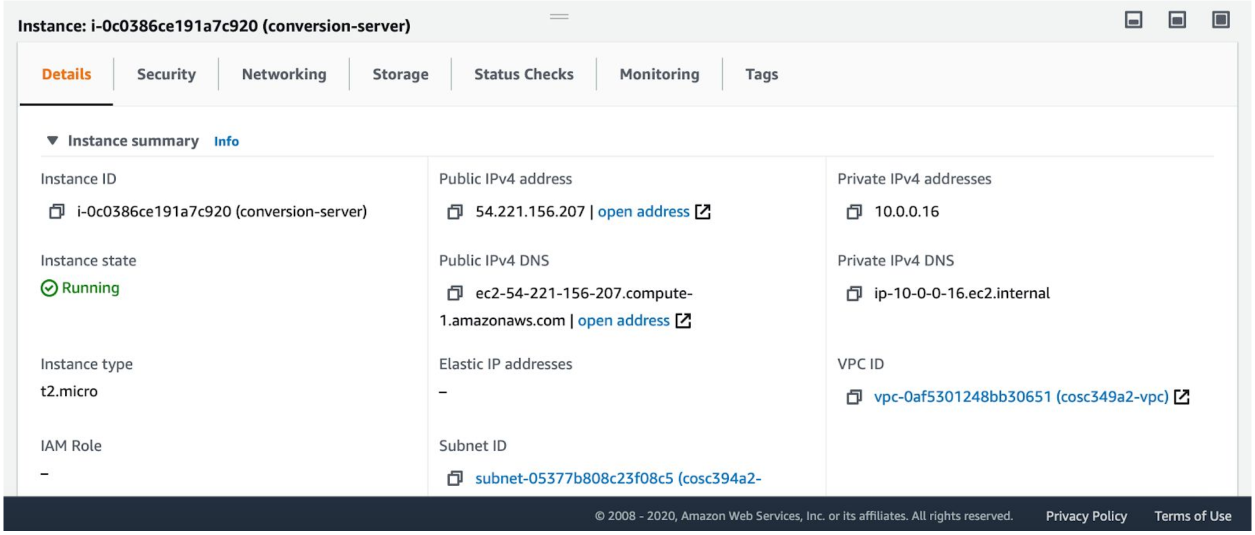
IAM Role: –

Subnet ID: subnet-05377b808c23f08c5 (cosc394a2-

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Conversion-server Instance:

- This instance was initially created in order to compute and display the timezone or currency conversion chosen by the user from the initial web server page. Once the “convert” button was pressed by the user this conversion-server should allow the computation of the chosen conversion method and should display the result of this conversion on a website associated with the initial web server.



Instance: i-0c0386ce191a7c920 (conversion-server)

Instance ID
i-0c0386ce191a7c920 (conversion-server)

Public IPv4 address
54.221.156.207 | open address

Private IPv4 addresses
10.0.0.16

Instance state
Running

Public IPv4 DNS
ec2-54-221-156-207.compute-1.amazonaws.com | open address

Private IPv4 DNS
ip-10-0-0-16.ec2.internal

Instance type
t2.micro

Elastic IP addresses
–

VPC ID
vpc-0af5301248bb30651 (cosc349a2-vpc)

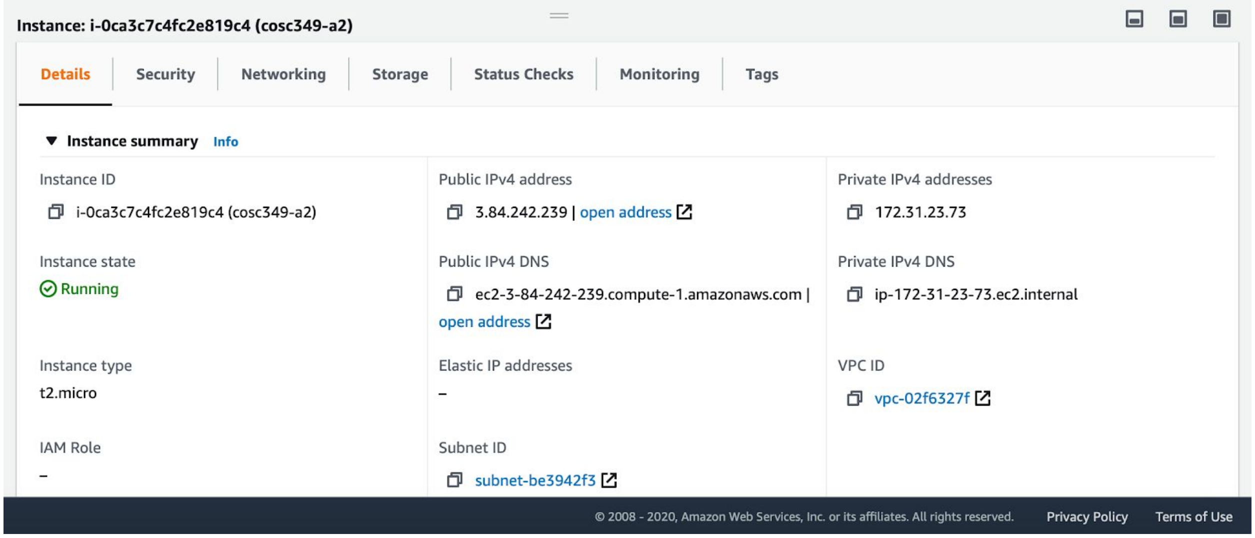
IAM Role
–

Subnet ID
subnet-05377b808c23f08c5 (cosc394a2-

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Cosc349-a2 Instance:

- This instance was created when following the steps in laboratory 4. This instance also functions as a single web server instance and displays the Apache test page when accessed via http://3.84.242.239 in any web browser.



Instance: i-0ca3c7c4fc2e819c4 (cosc349-a2)

Instance ID
i-0ca3c7c4fc2e819c4 (cosc349-a2)

Public IPv4 address
3.84.242.239 | open address

Private IPv4 addresses
172.31.23.73

Instance state
Running

Public IPv4 DNS
ec2-3-84-242-239.compute-1.amazonaws.com | open address

Private IPv4 DNS
ip-172-31-23-73.ec2.internal

Instance type
t2.micro

Elastic IP addresses
–

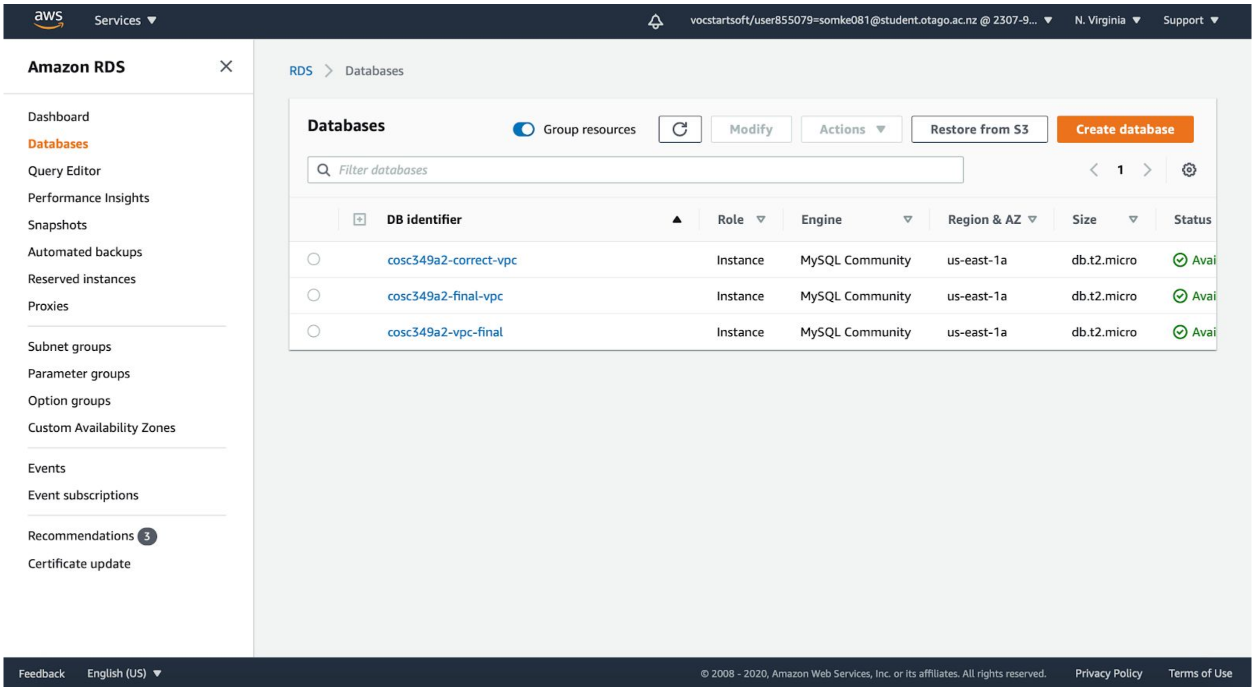
VPC ID
vpc-02f6327f

IAM Role
–

Subnet ID
subnet-be3942f3

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

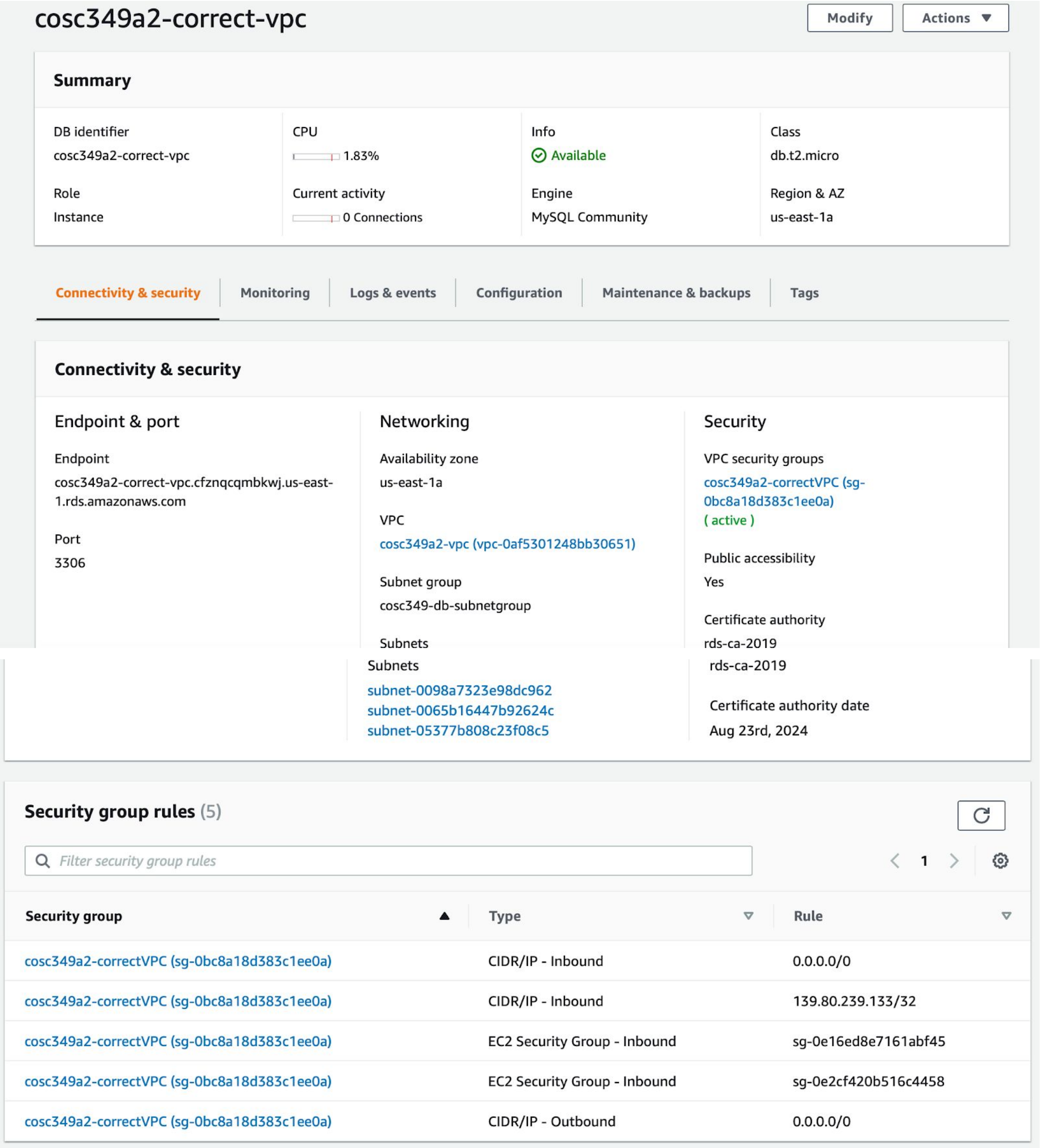
RDS Database Instances:



DB identifier	Role	Engine	Region & AZ	Size	Status
cosc349a2-correct-vpc	Instance	MySQL Community	us-east-1a	db.t2.micro	✓ Avai
cosc349a2-final-vpc	Instance	MySQL Community	us-east-1a	db.t2.micro	✓ Avai
cosc349a2-vpc-final	Instance	MySQL Community	us-east-1a	db.t2.micro	✓ Avai

Cosc349a2-correct-vpc:

- This instance was created during the **“Create a Web Server and an Amazon RDS DB Instance”** tutorial and currently contains MySQL code allowing the creation of a database that contains all of the countries and country information found within the first assignment. I had issues attempting to connect this database to the aforementioned EC2 instance that also was created due to this tutorial thus the MySQL code that this database contains could not be utilised.



cosc349a2-correct-vpc

Summary

DB identifier cosc349a2-correct-vpc	CPU 1.83%	Info ✓ Available	Class db.t2.micro
Role Instance	Current activity 0 Connections	Engine MySQL Community	Region & AZ us-east-1a

Connectivity & security

Endpoint & port

Endpoint
cosc349a2-correct-vpc.cfznqcqmbkwj.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability zone
us-east-1a

VPC
cosc349a2-vpc (vpc-0af5301248bb30651)

Subnet group
cosc349-db-subnetgroup

Subnets

Subnets

subnet-0098a7323e98dc962
subnet-0065b16447b92624c
subnet-05377b808c23f08c5

Security

VPC security groups
cosc349a2-correctVPC (sg-0bc8a18d383c1ee0a)
(active)

Public accessibility
Yes

Certificate authority
rds-ca-2019

rds-ca-2019

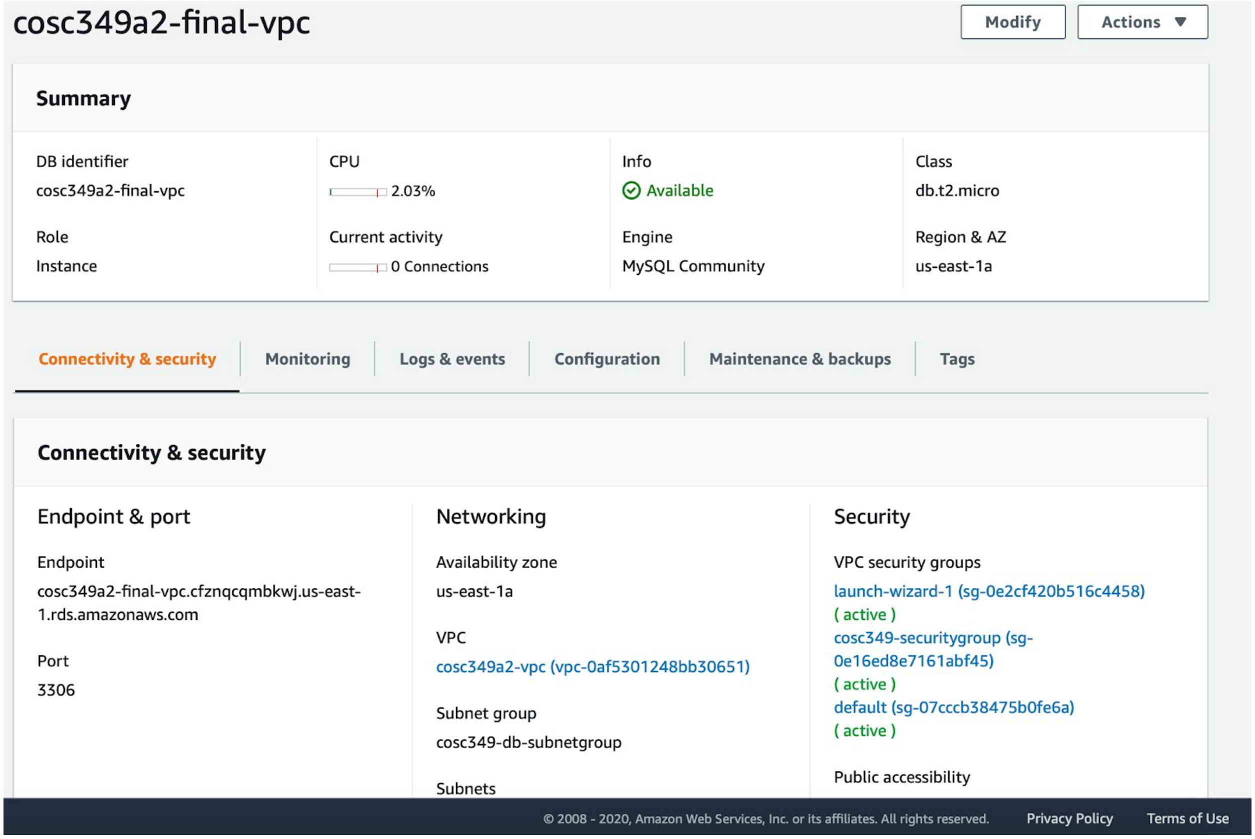
Certificate authority date
Aug 23rd, 2024

Security group rules (5)

Security group ▲	Type	Rule
cosc349a2-correctVPC (sg-0bc8a18d383c1ee0a)	CIDR/IP - Inbound	0.0.0.0/0
cosc349a2-correctVPC (sg-0bc8a18d383c1ee0a)	CIDR/IP - Inbound	139.80.239.133/32
cosc349a2-correctVPC (sg-0bc8a18d383c1ee0a)	EC2 Security Group - Inbound	sg-0e16ed8e7161abf45
cosc349a2-correctVPC (sg-0bc8a18d383c1ee0a)	EC2 Security Group - Inbound	sg-0e2cf420b516c4458
cosc349a2-correctVPC (sg-0bc8a18d383c1ee0a)	CIDR/IP - Outbound	0.0.0.0/0

Cosc349a2-final-vpc & Cosc349a2-vpc-final:

- Both of these database instances were created in an attempt to connect to a created EC2 instance as well as connect to the MySQL Workbench Software. I had issues connecting to both services using either database, thus these databases contain no MySQL code and are empty.



cosc349a2-final-vpc

Summary

DB identifier
cosc349a2-final-vpc

Role
Instance

CPU
2.03%

Current activity
0 Connections

Info
✓ Available

Engine
MySQL Community

Class
db.t2.micro

Region & AZ
us-east-1a

Connectivity & security

Endpoint & port

Endpoint
cosc349a2-final-vpc.cfznqcqmbkwj.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability zone
us-east-1a

VPC
cosc349a2-vpc (vpc-0af5301248bb30651)

Subnet group
cosc349-db-subnetgroup

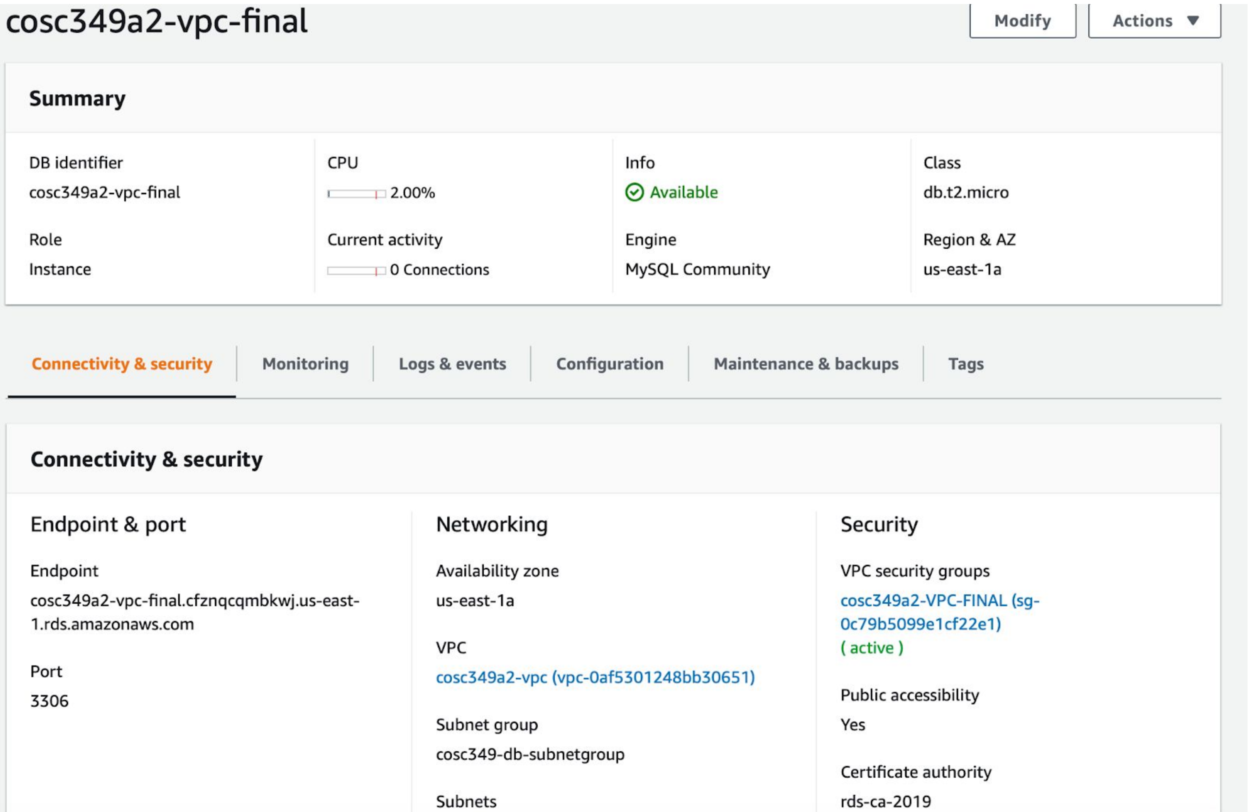
Subnets

Security

VPC security groups
launch-wizard-1 (sg-0e2cf420b516c4458)
(active)
cosc349-securitygroup (sg-0e16ed8e7161abf45)
(active)
default (sg-07cccb38475b0fe6a)
(active)

Public accessibility

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use



cosc349a2-vpc-final

Summary

DB identifier
cosc349a2-vpc-final

Role
Instance

CPU
2.00%

Current activity
0 Connections

Info
✓ Available

Engine
MySQL Community

Class
db.t2.micro

Region & AZ
us-east-1a

Connectivity & security

Endpoint & port

Endpoint
cosc349a2-vpc-final.cfznqcqmbkwj.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability zone
us-east-1a

VPC
cosc349a2-vpc (vpc-0af5301248bb30651)

Subnet group
cosc349-db-subnetgroup

Subnets

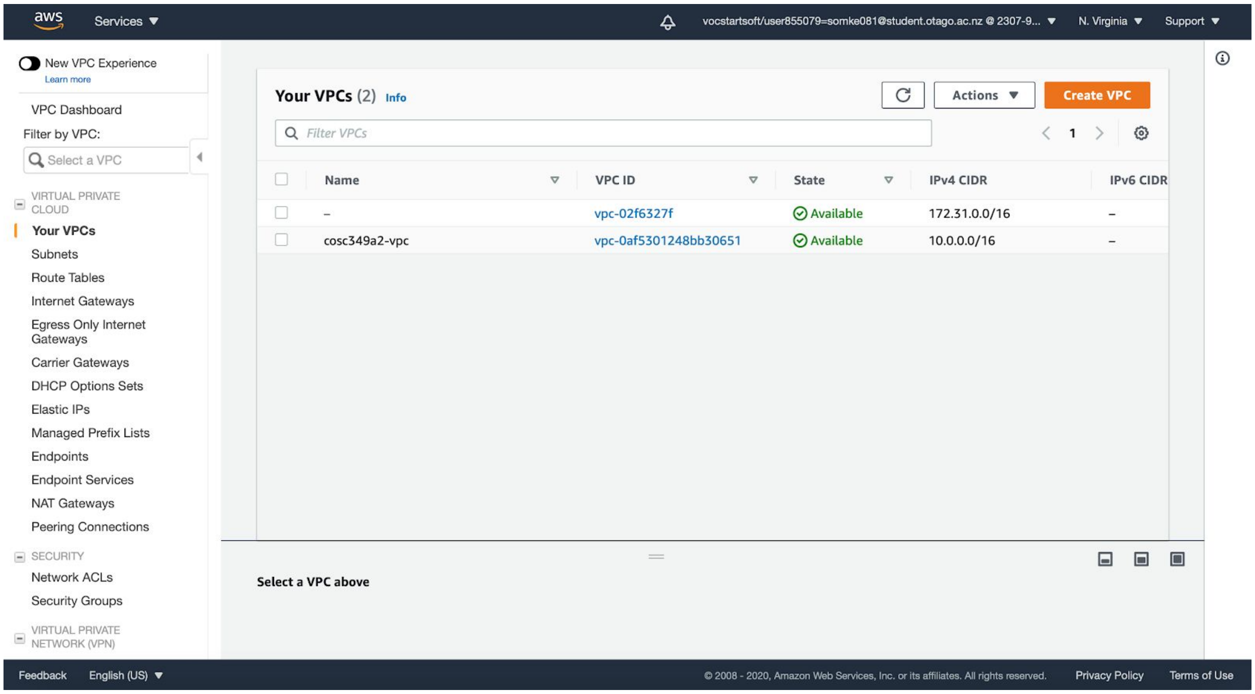
Security

VPC security groups
cosc349a2-VPC-FINAL (sg-0c79b5099e1cf22e1)
(active)

Public accessibility
Yes

Certificate authority
rds-ca-2019

Virtual Private Cloud (VPC):



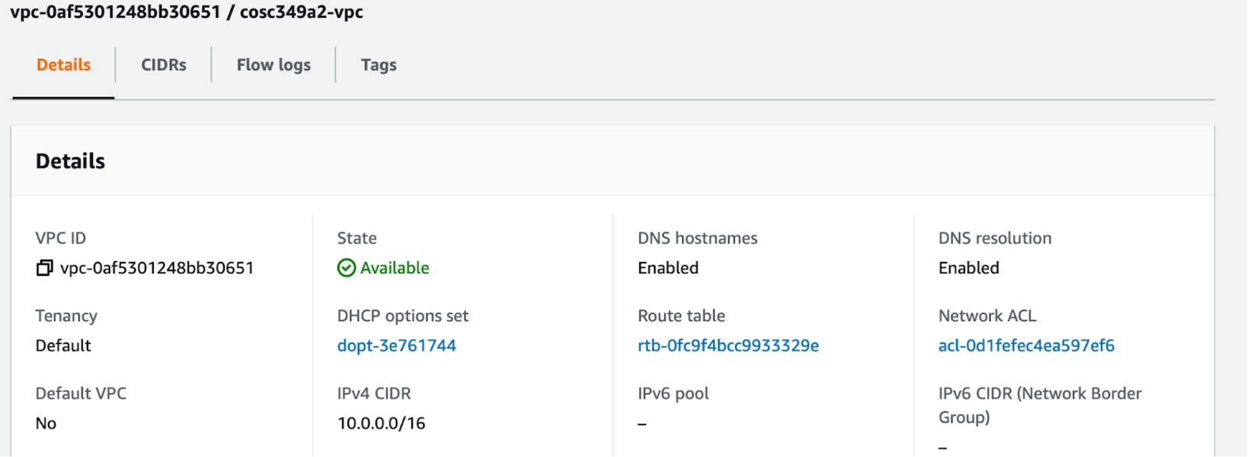
Your VPCs (2) Info

□	Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR
□	–	vpc-02f6327f	✓ Available	172.31.0.0/16	–
□	cosc349a2-vpc	vpc-0af5301248bb30651	✓ Available	10.0.0.0/16	–

Select a VPC above

Cosc349a2-vpc:

- This VPC was also created during the **"Create a Web Server and an Amazon RDS DB Instance"** tutorial in an attempt to contain the created EC2 instance and database instance, though I struggled to create this EC2 and database connection.



vpc-0af5301248bb30651 / cosc349a2-vpc

Details

VPC ID: vpc-0af5301248bb30651

State: ✓ Available

DNS hostnames: Enabled

DNS resolution: Enabled

Tenancy: Default

DHCP options set: dopt-3e761744

Route table: rtb-0fc9f4bcc9933329e

Network ACL: acl-0d1fefec4ea597ef6

Default VPC: No

IPv4 CIDR: 10.0.0.0/16

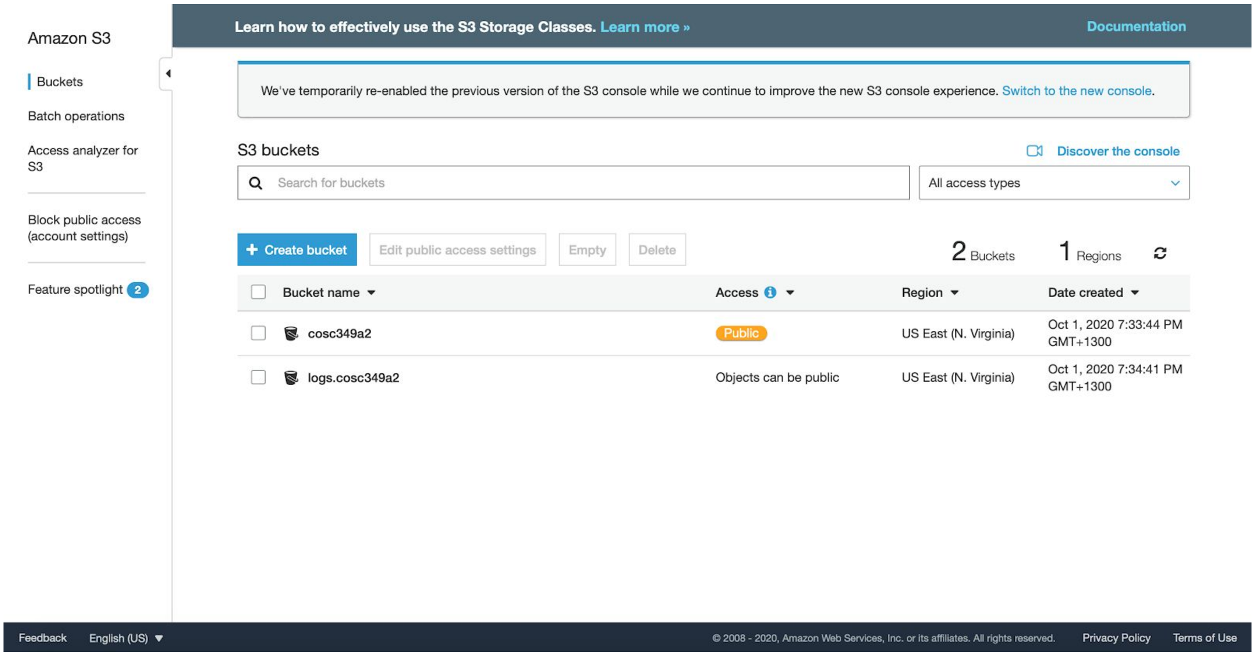
IPv6 pool: –

IPv6 CIDR (Network Border Group): –

S3:

- These two buckets were created when completing laboratory 8. I attempted to use them within my assignment 2 development though struggled to connect either bucket to any other instance and/or use them in any way.

- The cosc349a2 bucket still functions as a web server, running the index.html code and displaying ‘Hello world” on the website page.
  - This can be viewed at the Static Website Hosting Endpoint: http://cosc349a2.s3-website-us-east-1.amazonaws.com



Learn how to effectively use the S3 Storage Classes. Learn more »

We've temporarily re-enabled the previous version of the S3 console while we continue to improve the new S3 console experience. Switch to the new console.

S3 buckets

Bucket name	Access	Region	Date created
cosc349a2	Public	US East (N. Virginia)	Oct 1, 2020 7:33:44 PM GMT+1300
logs.cosc349a2	Objects can be public	US East (N. Virginia)	Oct 1, 2020 7:34:41 PM GMT+1300

How to Connect to the Important Instances:

EC2 - Cosc349a2-webserver:

View this web server online: http://3.84.242.239

Connecting via an ssh Client

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is cosc349a2-webserver.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
   chmod 400 cosc349a2-webserver.pem
4. Connect to your instance using its Public DNS:
   ec2-107-22-88-33.compute-1.amazonaws.com

S3 - Cosc349a2 bucket:

View via the Static Website Hosting Endpoint:
http://cosc349a2.s3-website-us-east-1.amazonaws.com

Why These Services Were Chosen:

S3:
The S3 cloud storage service was chosen because of how easy it is to access files. They are accessible by the url of the object stored in the S3 bucket which means that no permissions are required to run the program. The user does not need to log in to AWS or require any credentials before being able to access this area of the application. S3 was also chosen as the application only requires a GET request thus is able to be publicly accessible.

EC2:
Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the AWS cloud. EC2 means that you don't need to invest in hardware up front, allowing you to develop and deploy applications faster. EC2 also allows you to launch numerous instances at one time. It also configures security and manages storage for you automatically. EC2 enables you to scale up or down automatically to account for increased or decreased traffic flow. This ensures that you only pay for what you use.

When you sign up for AWS, you can get started with Amazon EC2 for free using the AWS Free Tier. This is incredibly beneficial and allows providers to experiment with EC2 and all of the AWS services.

EC2 provides you with rapid deployment speed, allowing you to customise server instances and have them online and ready to accept connections in under 10 minutes. Then you can quickly shut down instances when they are no longer needed.

RDS Database:
Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the AWS Cloud. RDS is available for many different database instance types and provides you with six familiar database engines to choose from. It is also optimised for memory, performance or I/O in all of the different instances and thus adapts for your needs and your environment.

Virtual Private Cloud (VPC):
A virtual private cloud (VPC) is a pool of computing resources that are shared and allocated within a public cloud environment. This provides a certain level of isolation between resources outside of that VPC.

The main benefit that a VPC provides is advanced security features, such as security groups and network access control lists. This allows users to adapt inbound and outbound filtering to every VPC, to account for the different requirements of specific VPC's. You can also store data in S3 and restrict access to its buckets so that it's only able to be accessed for instances present within a specific VPC.

The greatest benefit to this service is that there are no additional charges for creating and using the VPC itself.

Alternative Services:

Using Docker

https://cloud.docker.com/repository/docker/walst388/repository

AWS and Docker

https://aws.amazon.com/docker/

Setting-up MySQL in Docker

https://medium.com/better-programming/setting-up-mysql-database-in-a-docker-d6c69a3e9afe

Ideal Implementation & Future Development:

Ideally developers that work further on this application will be able to understand and utilise the AWS resources that are available to them better than I have, in order to get the application up and running in the AWS cloud. If done correctly this would give users access to the timezone and currency converter application first exposed to the user in assignment one, and would allow users to find the timezone of their chosen country represented in NZ time, as well as the currency of their chosen country represented in NZ currency.

For future development of this application, I would hope that developers could extend this application into a tourist advisory application, which would allow users of this application to get extensive knowledge of their chosen country. This information would include the current season, expected weather and any current events that are occurring in that country, including the crime rates and common scams that are used in their chosen country. Many more countries should also be included and accessible to the user to account for the different possible travel locations as the current country pool is very limited.