Ministry of Communications and Information Technology

Building a Highly Available, Scalable Web Application Project

Name: Mazen Magdy Ahmed Abdelsalam

Group: GIZ1_SWD8_M1e

Email: mazen_magdy@hotmail.com

Cell Phone: 01090305555

Abstract:

This project aims to develop a highly scalable and available web application using AWS services. The solution leverages Amazon EC2 for web hosting and Amazon RDS for data storage. It incorporates load balancing with an Application Load Balancer (ALB), automatic scaling with an Auto Scaling Group (ASG), and high availability through multi-AZ deployments. Security measures include the use of Amazon VPC, Security Groups, and AWS Secrets Manager to ensure robust, secure credential management.

I. Introduction

1. Overview of the project

The goal of this project is to build a highly available, scalable, and secure web application using Amazon Web Services (AWS). The application is designed to handle high volumes of traffic while ensuring minimal downtime, robust performance, and optimal cost management.

To achieve this, the project utilizes Amazon EC2 for hosting the web application and Amazon RDS (MySQL) for managing the database. Traffic management is handled by an Application Load Balancer (ALB), ensuring even distribution of user requests across multiple instances. An Auto Scaling Group (ASG) is implemented to dynamically adjust the number of EC2 instances based on traffic demand, allowing the application to scale seamlessly during peak usage periods.

Additionally, the architecture is designed with security in mind, leveraging Amazon VPC for network isolation, Security Groups for access control, and AWS Secrets Manager for secure management of sensitive data such as database credentials. The solution also incorporates high availability by deploying resources across multiple Availability Zones (AZs), ensuring uptime and resilience in case of failures.

This project ensures that the web application is highly available, scalable, and cost-efficient while providing a secure environment for managing sensitive data and user interactions.

2. Objectives of the solution

a) High Availability: Ensure that the web application is accessible with minimal downtime by deploying resources across multiple Availability Zones (AZs). This design allows the application to remain operational even in the event of server or zone failures.

b) Scalability: Implement dynamic scaling mechanisms that automatically adjust the number of running instances based on user demand. This ensures that the application can handle varying levels of traffic, especially during peak periods, without over-provisioning or underutilizing resources.

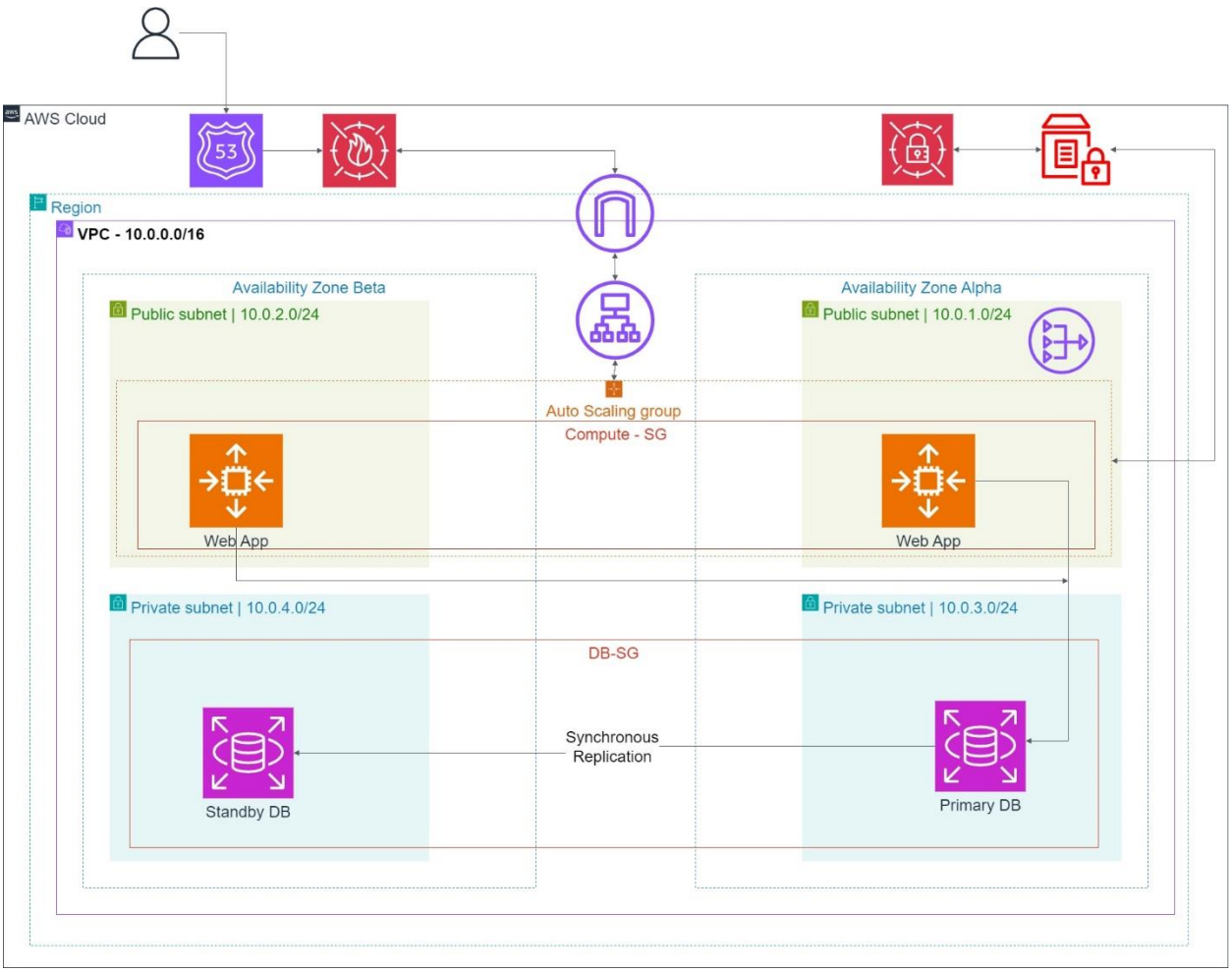
c) Load Balancing: Use an Application Load Balancer (ALB) to evenly distribute incoming traffic across multiple EC2 instances. This helps prevent resource overload, optimizing performance and fault tolerance.

d) Security: Ensure the application and database are secured using Amazon VPC for network isolation, Security Groups for controlling access, and AWS Secrets Manager for securely managing sensitive credentials like database login information.

e) Cost Optimization: Use AWS services such as Auto Scaling and Amazon RDS to ensure cost efficiency by scaling resources only when necessary and reducing infrastructure costs during periods of low demand.

f) Data Management: Deploy Amazon RDS (MySQL) to handle student or user records efficiently, providing reliable, managed database services that ensure data integrity, backup automation, and performance optimization.

3. **Architecture Diagram**



4. **Functional Requirements**

   4.1 **Web Hosting:**

   - **Amazon EC2:** Instances used to host the web application. The application is deployed on Linux (Ubuntu) servers within a Virtual Private Cloud (VPC).

   4.2 **Database Management:**

   - **Amazon RDS (MySQL):** Used to store and manage data. The database is deployed in a private subnet for enhanced security.

5. **Load Balancing**

   - **Application Load Balancer (ALB):** Distributes incoming traffic evenly across multiple EC2 instances, ensuring fault tolerance and better scalability.

6. **Scalability**

   - **Auto Scaling Group (ASG):** Automatically adjusts the number of EC2 instances based on real-time user traffic, scaling out during high traffic and scaling in during low traffic periods.

7. **High Availability**

   - The application ensures high availability by using both ALB and ASG across multiple Availability Zones (AZs) to handle server or AZ failures without downtime.

8. **Security**
    - **Amazon VPC:** Provides network isolation for secure hosting.
    - **Security Groups:** Regulate inbound/outbound traffic for EC2 and RDS instances.
    - **AWS Secrets Manager:** Manages database credentials, ensuring they are securely accessed by the application without hardcoding sensitive information.

II. Cost Analysis

A cost estimate for running the architecture on AWS for 12 months is calculated using the AWS Pricing Calculator.

Contact your AWS representative: Contact Sales

Export date: **10/23/2024** Language: **English**

Estimate URL: **https://calculator.aws/#/estimate?id=a5efbed5dfa30698750b6845d07c4cc2e6239fe9**

Estimate summary

Upfront cost	Monthly cost	Total 12 months cost
0.00 USD	611.78 USD	7,341.36 USD Includes upfront cost

Detailed Estimate

Name	Group	Region	Upfront cost	Monthly cost
Amazon Virtual Private Cloud (VPC)	No group applied	US East (N. Virginia)	0.00 USD	36.68 USD
Status: - **Description:** **Config summary:** Number of NAT Gateways (1) Number of In-use public IPv4 addresses (1)				
Amazon EC2	No group applied	US East (N. Virginia)	0.00 USD	8.47 USD
Status: - **Description:** **Config summary:** Tenancy (Shared Instances), Operating system (Linux), Workload (Consistent, Number of instances: 2), Advance EC2 instance (t2.micro), Pricing strategy (Compute Savings Plans 3yr No Upfront), Enable monitoring (disabled), DT Inbound: Not selected (0 TB per month), DT Outbound: Not selected (0 TB per month), DT Intra-Region: (0 TB per month)				
Amazon RDS for MySQL	No group applied	US East (N. Virginia)	0.00 USD	358.80 USD
Status: - **Description:** **Config summary:** Storage amount (100 GB), Storage for each RDS instance (General Purpose SSD (gp2)), Nodes (1), Instance type (db.m1.large), Utilization (On-Demand only) (100 %Utilized/Month), Deployment option (Multi-AZ), Pricing strategy (OnDemand)				

AWS Secrets Manager	No group applied	US East (N. Virginia)	0.00 USD	5.40 USD
Status: - **Description:** **Config summary:** Number of secrets (1), Average duration of each secret (30 days), Number of API calls (1000000 per month)				
Amazon Route 53	No group applied	US East (N. Virginia)	0.00 USD	40.00 USD
Status: - **Description:**				
Elastic Load Balancing	No group applied	US East (N. Virginia)	0.00 USD	162.43 USD
Status: - **Description:** **Config summary:** Number of Application Load Balancers (1)				

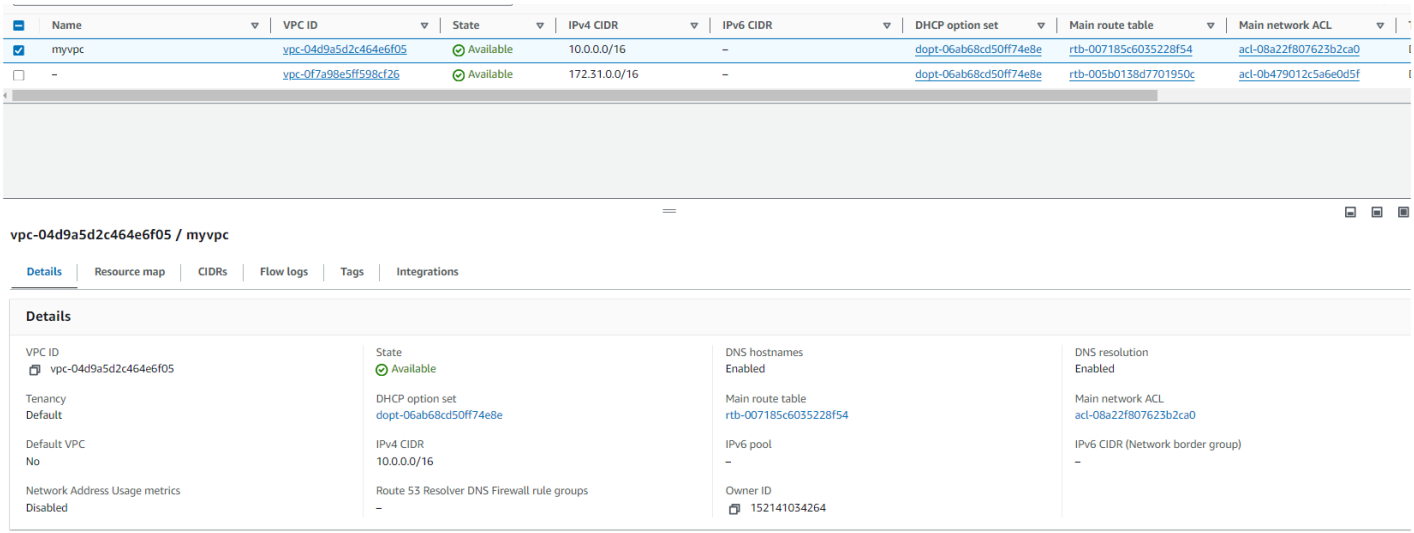
Acknowledgement

AWS Pricing Calculator provides only an estimate of your AWS fees and doesn't include any taxes that might apply. Your actual fees depend on a variety of factors, including your actual usage of AWS services. Learn more

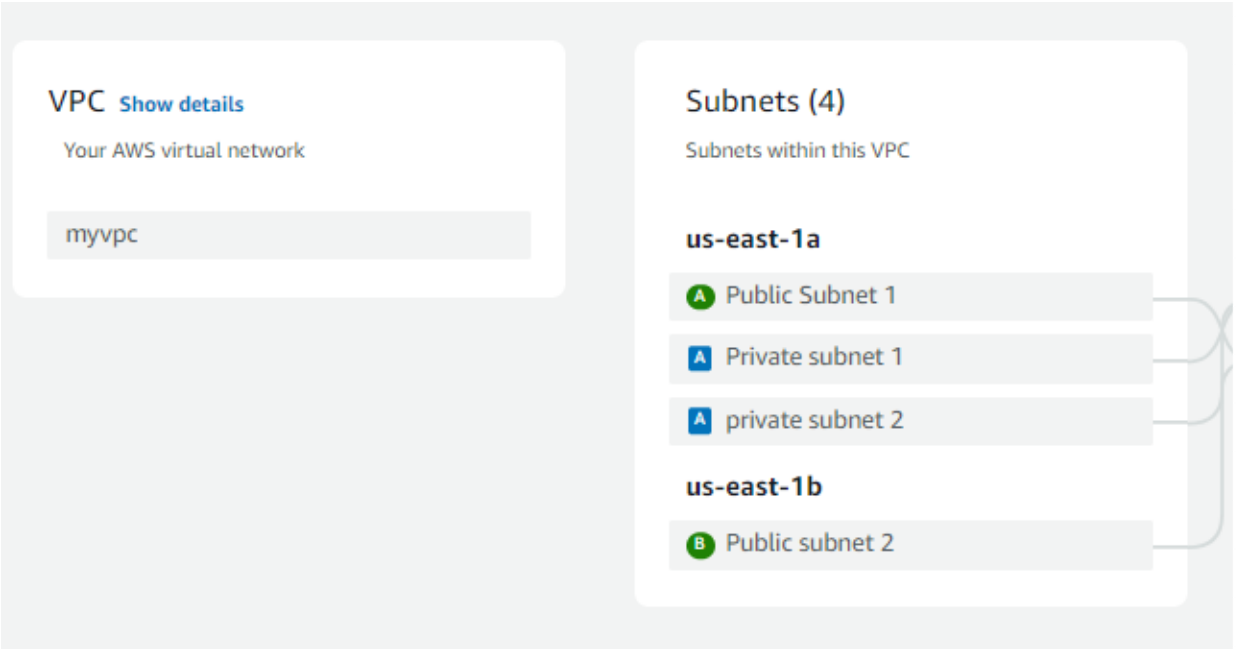
This includes the costs for EC2, RDS, ALB, and other related services.

III. Implementation Steps

1. **VPC Creation:** Set up a Virtual Private Cloud with public and private subnets.



Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP option set	Main route table	Main network ACL
myvpc	vpc-04d9a5d2c464e6f05	Available	10.0.0.0/16	–	dopt-06ab68cd50ff74e8e	rtb-007185c6035228f54	acl-08a22f807623b2ca0
–	vpc-0f7a98e5ff598cf26	Available	172.31.0.0/16	–	dopt-06ab68cd50ff74e8e	rtb-005b0138d7701950c	acl-0b479012c5a6e0d5f



2. **Security Group Configuration:** Define Security Groups to control access to the EC2 and RDS instances.

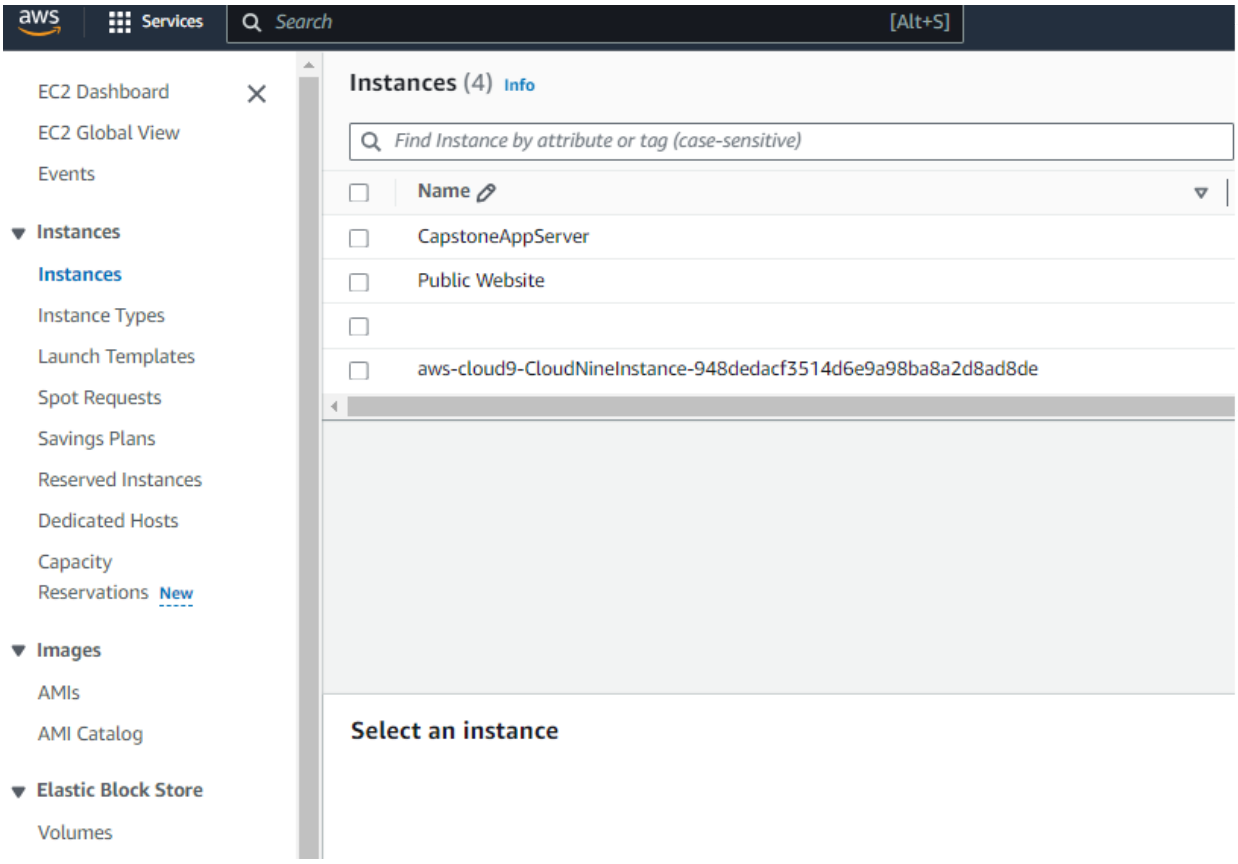
Security Groups (8) Info

Find resources by attribute or tag

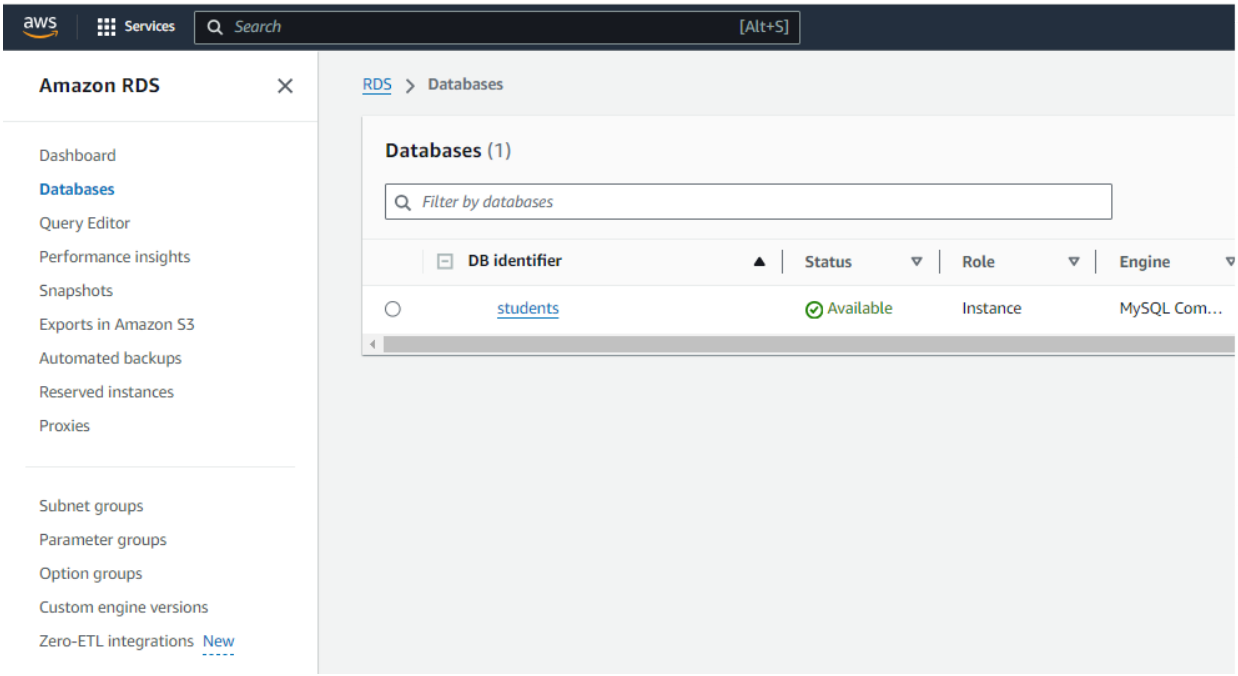
Name	Security group ID	Security group name
–	sg-0665b85f37db31c84	ALB-SG
–	sg-02c032ba2067f4424	launch-wizard-1
–	sg-02510beca681ab57b	ec2-rds-1
–	sg-0781496be4b398163	DBSG
aws-cloud9-CloudN…	sg-03e0ec4e247a5a85c	aws-cloud9-CloudNineInstance-948de…
–	sg-0effbd33d38b42d77	rds-ec2-1
–	sg-03afed581e3a4c9a2	default
–	sg-029c1a2246c67397b	default

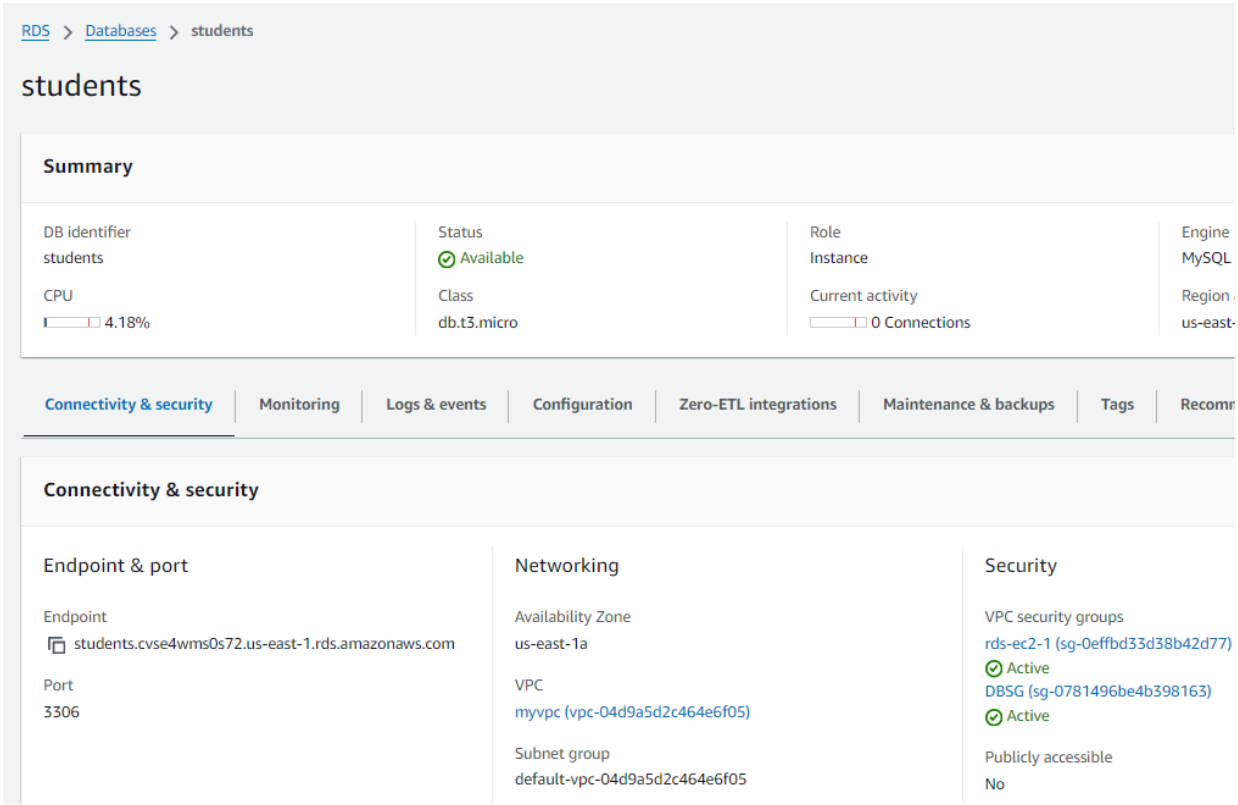
Description	Owner	Inbound rules count
Allow internet traffic via ALB	152141034264	1 Permission entry
launch-wizard-1 created 2024-10-23T…	152141034264	5 Permission entries
Security group attached to instances t…	152141034264	0 Permission entries
DatabaseSecurityGroup	152141034264	1 Permission entry
Security group for AWS Cloud9 enviro…	152141034264	2 Permission entries
Security group attached to STUDENTS…	152141034264	1 Permission entry
default VPC security group	152141034264	1 Permission entry
default VPC security group	152141034264	1 Permission entry

3. **EC2 Deployment:** Launch EC2 instances for the web application in the public subnet.

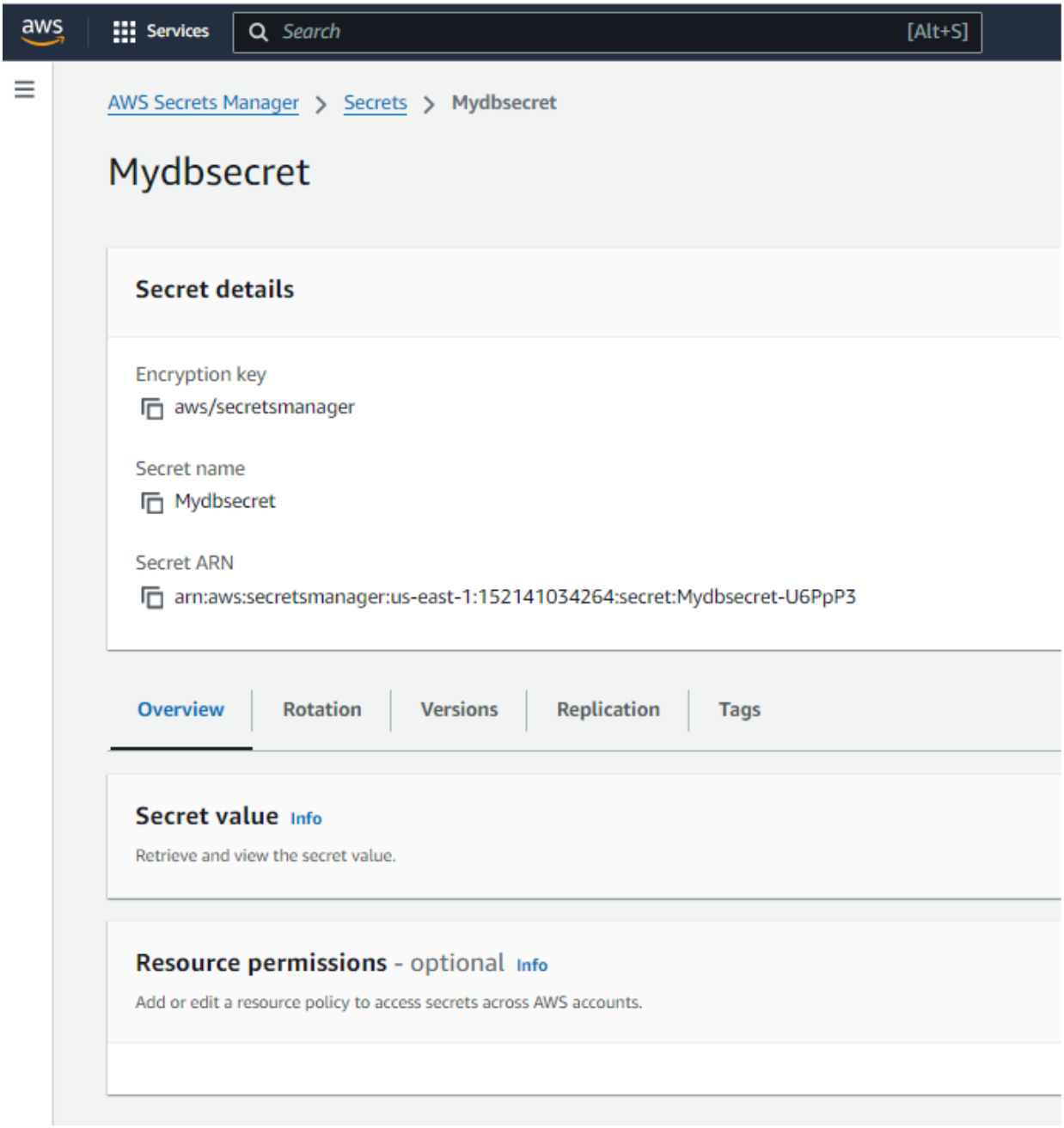


4. **Amazon RDS Setup:** Deploy an RDS instance (MySQL) in the private subnet..

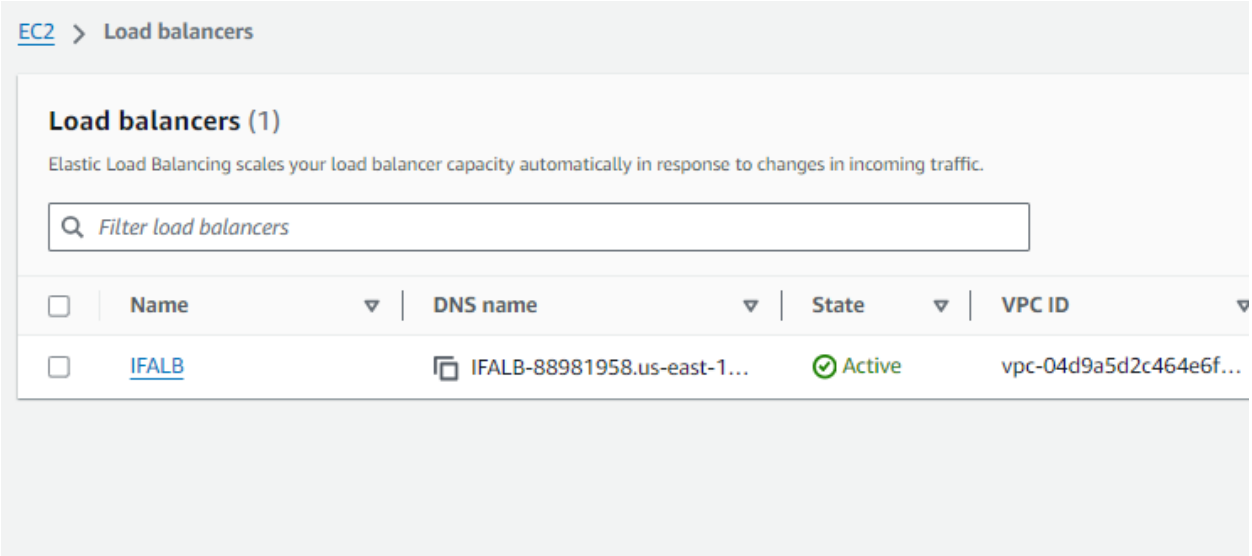


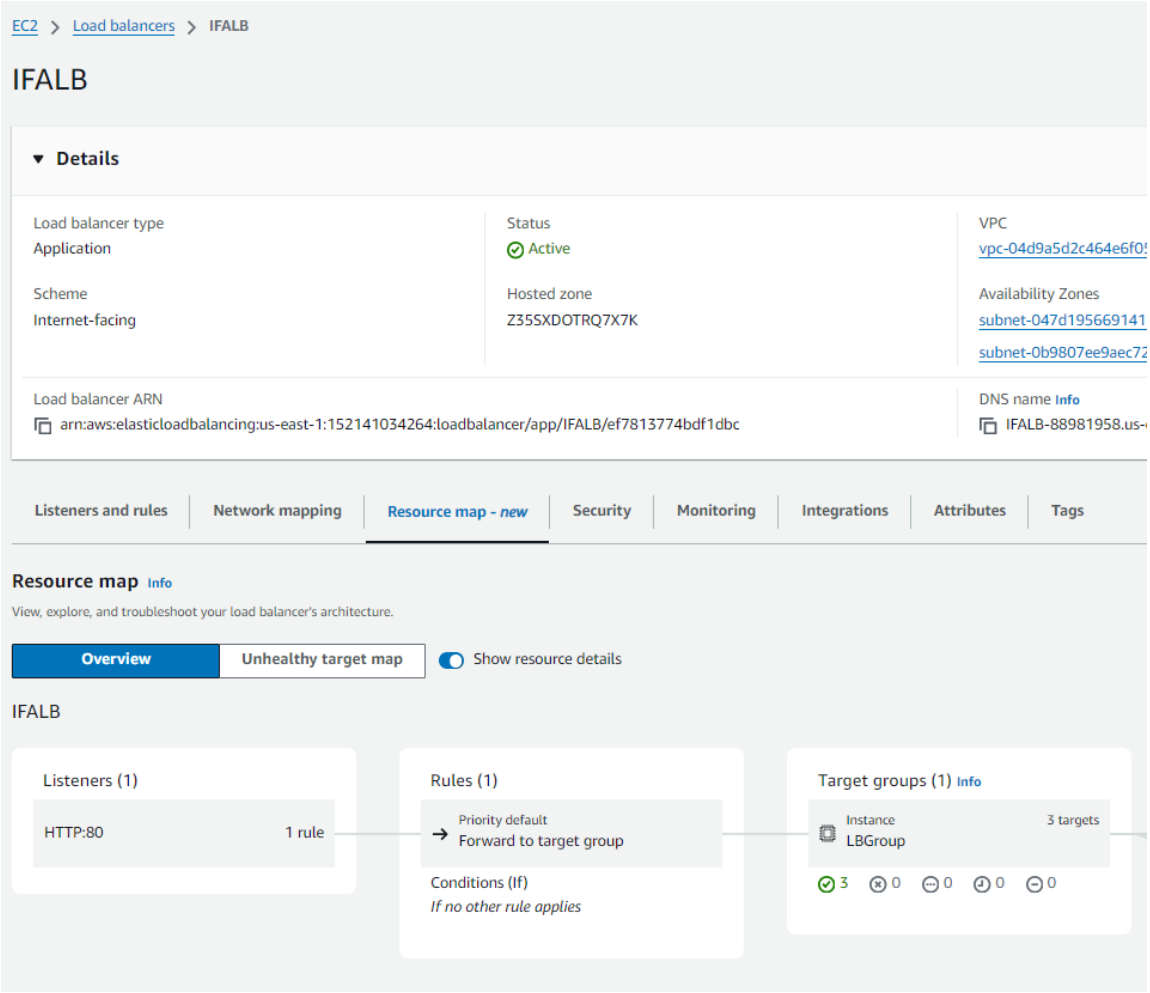


5. **AWS Secrets Manager Setup:** Store and manage database credentials securely.

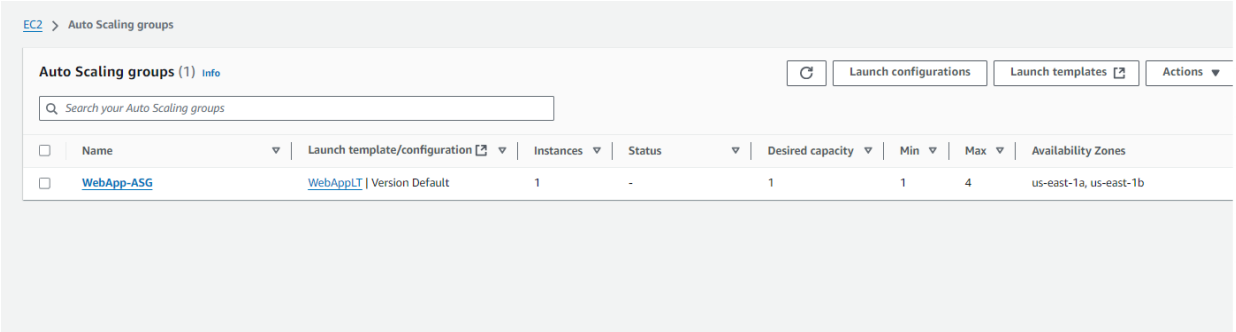


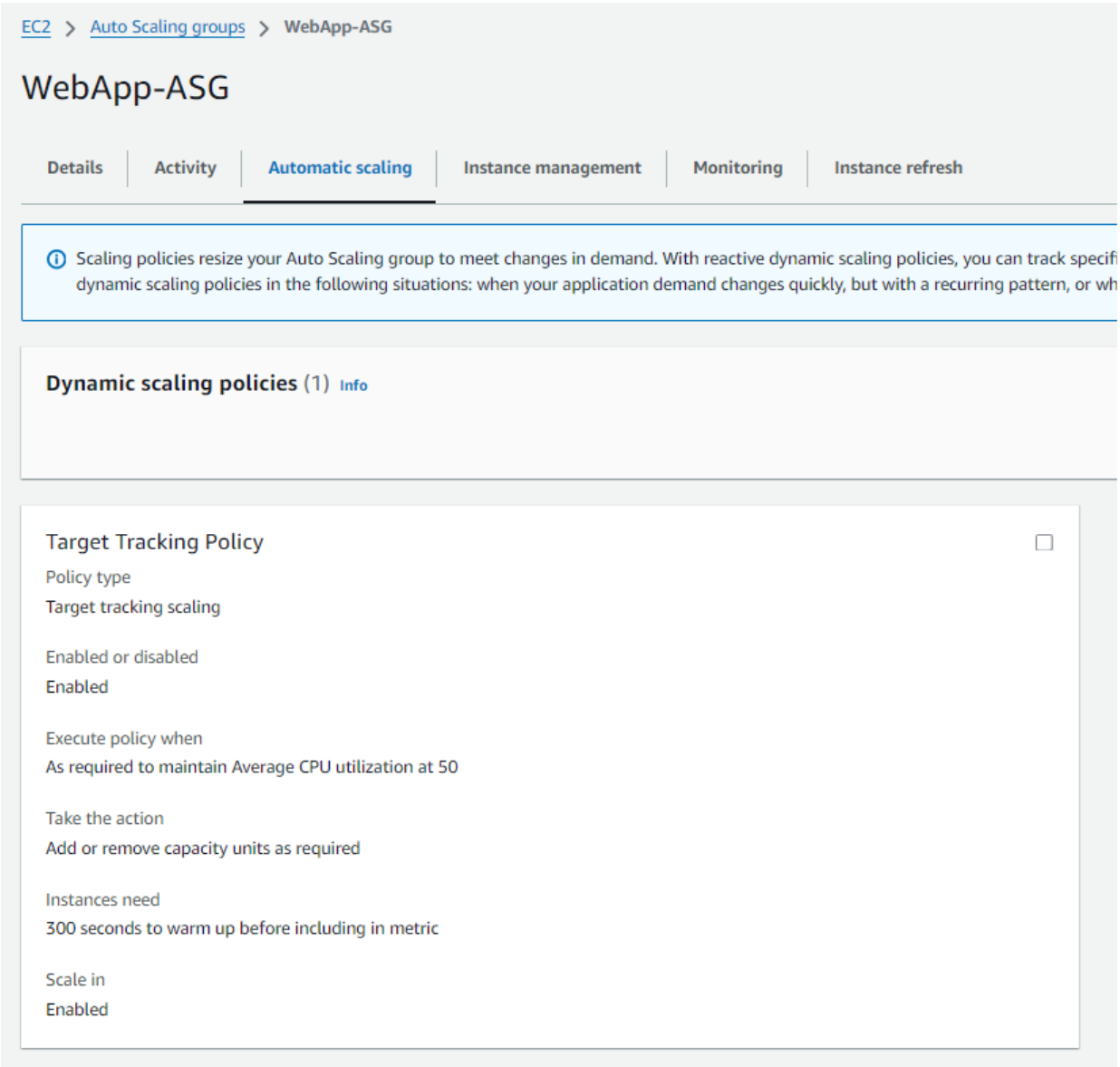
6. **Application Load Balancer Setup:** Configure an ALB to balance traffic across multiple instances.





7. **Auto Scaling Group Setup:** Set up an ASG to scale EC2 instances dynamically.





IV. Conclusion

The AWS-based solution designed meets the project requirements for high availability, scalability, load balancing, and security. By leveraging a combination of Amazon EC2, RDS, ALB, Auto Scaling, and AWS Secrets Manager, the application handles high traffic, ensures resilience to server failures, and optimizes costs.

---

5. Contact Information

- **Email:** mazen_magdy@hotmail.com
- **Phone:** +201090305555
- **LinkedIn:** linkedin.com/in/MazenMagdy