D782 - Task 2 - Attempt 1

Silver Alcid - #011933973

9/02/2025

SUBMISSION: PAPER, LINK, SLIDES

Security for CloudCampusIQ

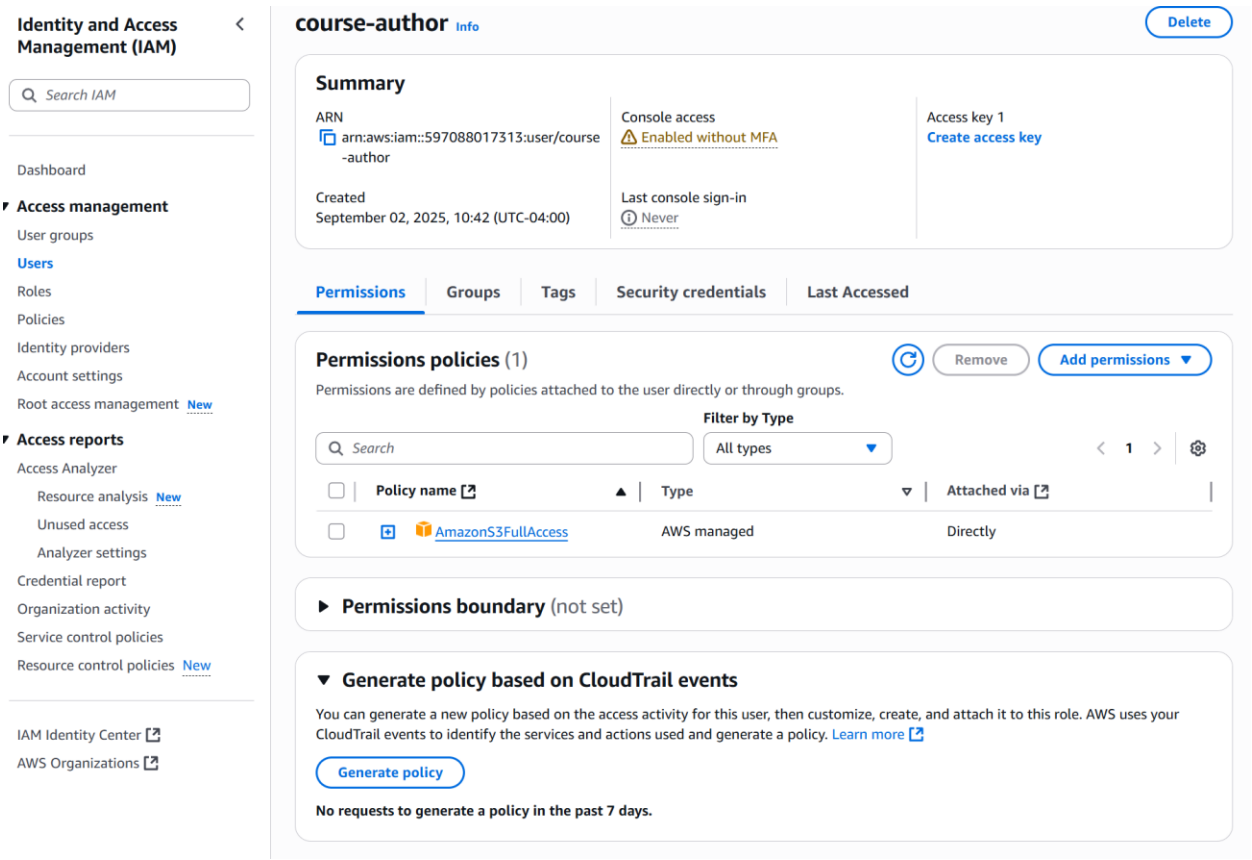
A1. Identity and Access Management Roles Setup

I created an IAM user called course-author and assigned it permissions limited to Amazon S3 and Elastic Beanstalk resources. This ensures only authenticated and authorized users can access application storage and deployment resources. Using role-based access control prevents unauthorized access by enforcing the principle of least privilege, so users only have the permissions required for their tasks.

A2. Access Monitoring (HTTPS, Firewall, Logging)

I enabled HTTPS by requesting a free SSL/TLS certificate from AWS Certificate Manager and attaching it to my Elastic Beanstalk environment. In the Security Group settings, I restricted inbound rules to only allow HTTPS traffic on port 443, blocking all other unnecessary ports. To monitor access and detect issues, I enabled logging through Amazon CloudWatch, which collects and stores request and error logs for the application.

A3. Configuration Screenshots and Security Policies



I applied a least-privilege security policy for IAM, ensuring users can only access assigned resources. Encryption in transit is enforced through HTTPS, and CloudWatch logs ensure that activity is tracked for auditing and troubleshooting.

B1. Application Host Justification

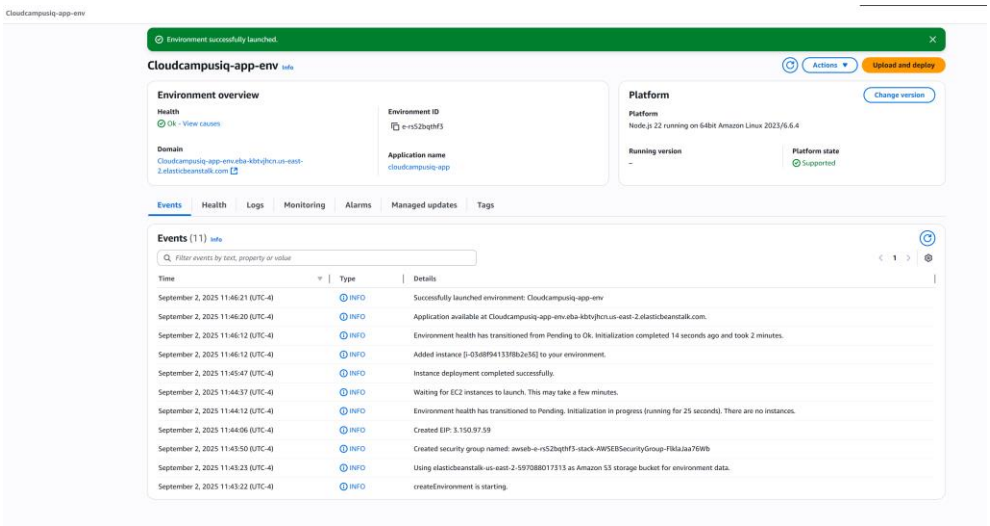
I chose AWS Elastic Beanstalk (PaaS) to host the application. Elastic Beanstalk simplifies deployment by automatically handling server provisioning, load balancing, and scaling, which allows CloudCampusIQ to focus on the application rather than managing infrastructure.

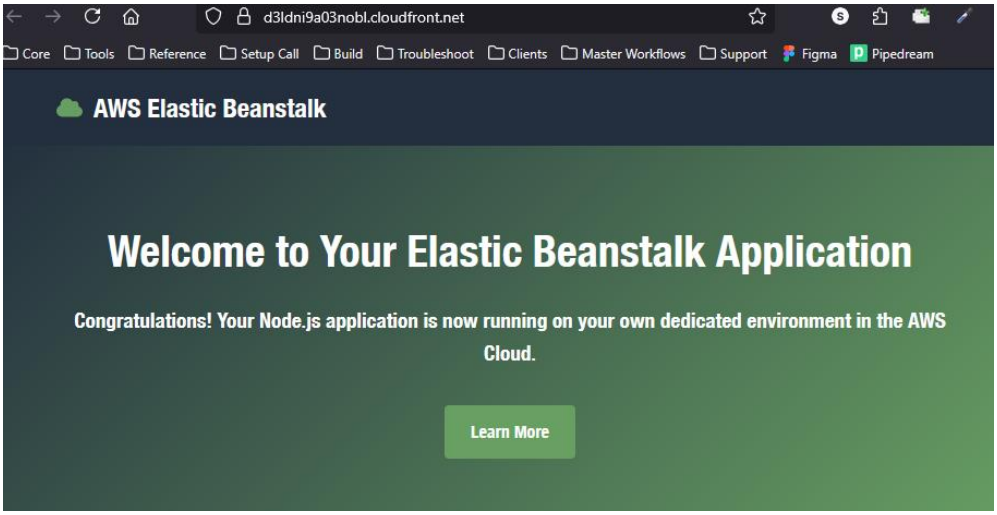
B2. Networking Choice Justification

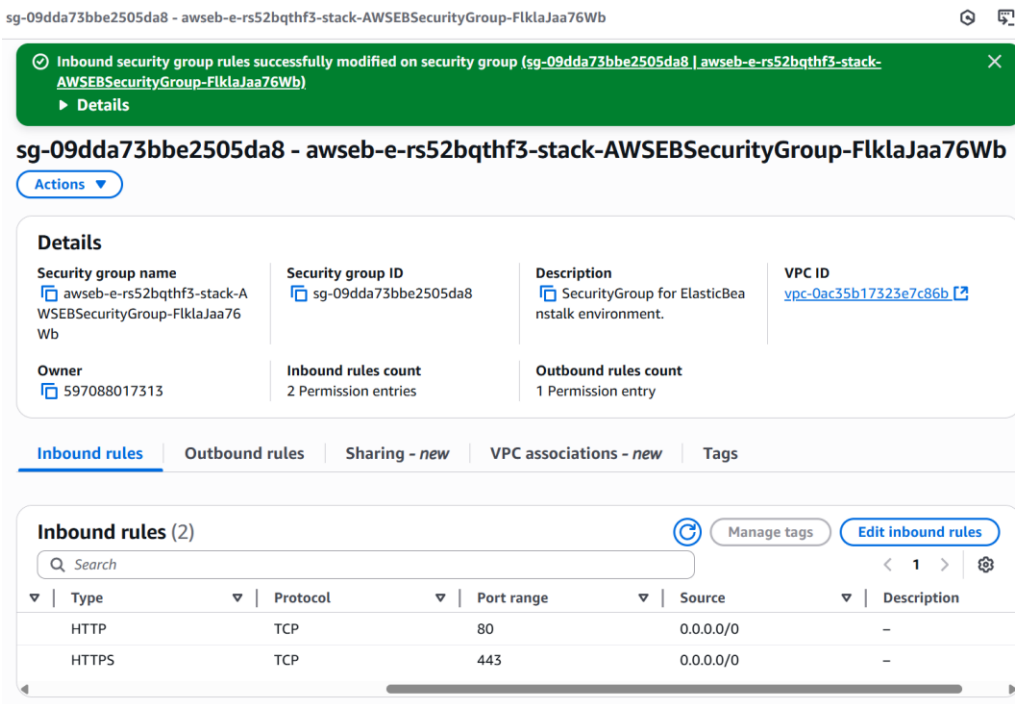
The application is accessible through the public DNS name automatically provided by Elastic

Beanstalk. The networking configuration uses Security Groups to restrict access to port 443 (HTTPS) only. This setup ensures that students and instructors can securely reach the application over the internet while protecting it from unauthorized connections.

B3. Application Screenshots and Setup Description







I deployed a sample application by uploading the source code package through the Elastic Beanstalk console. Elastic Beanstalk created the necessary environment, provisioned resources, and launched the application automatically. Once deployment completed, I verified the application by accessing it at the provided URL.

B4. URL of Deployed Application

https://d3ldni9a03nobl.cloudfront.net/

C1. Monitoring Alerts Configuration

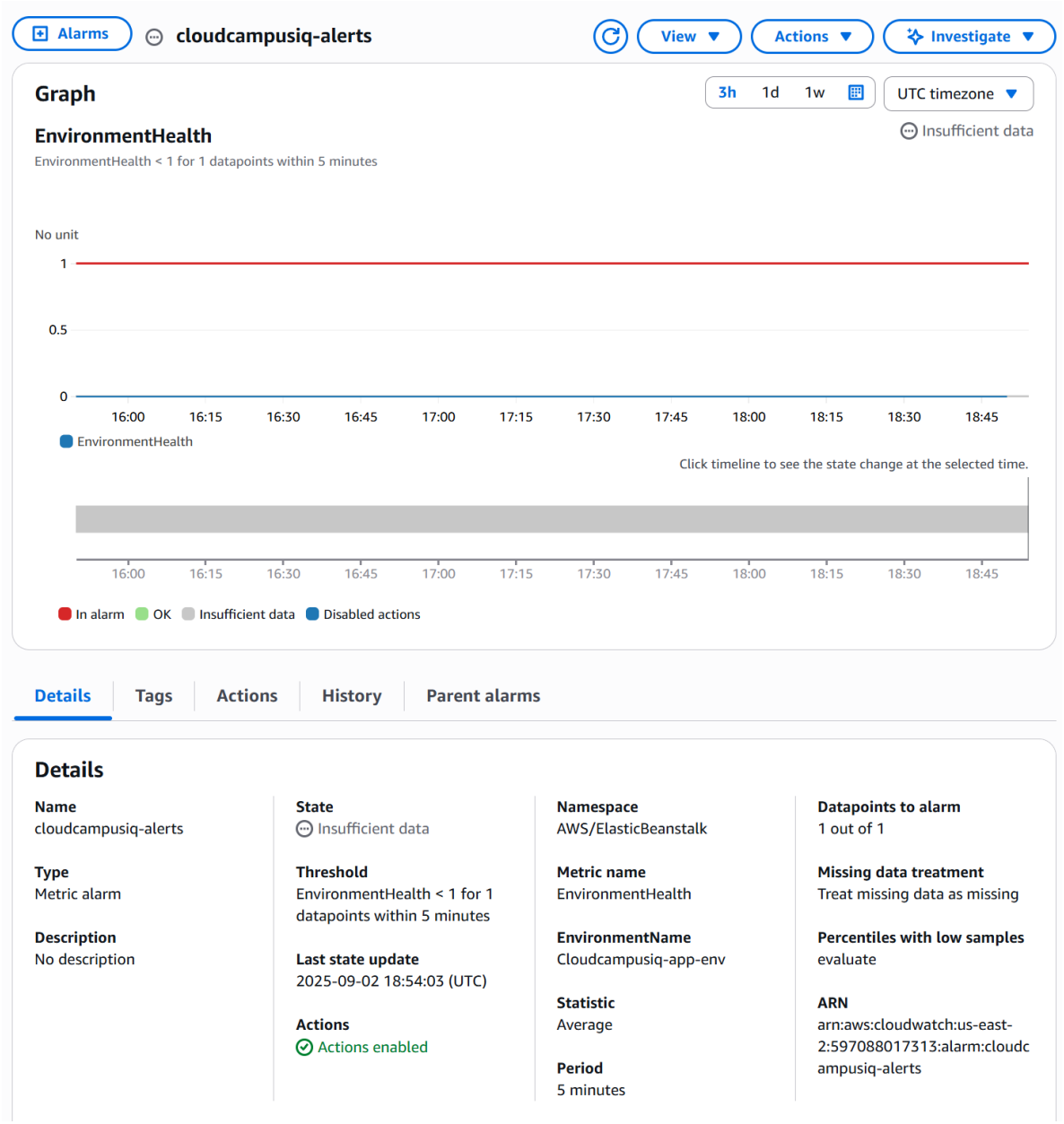
I created a CloudWatch alarm that triggers if the application's CPU utilization exceeds 70% for more than five minutes. This alert ensures that I am notified if the environment experiences performance issues, such as excessive load or insufficient scaling.

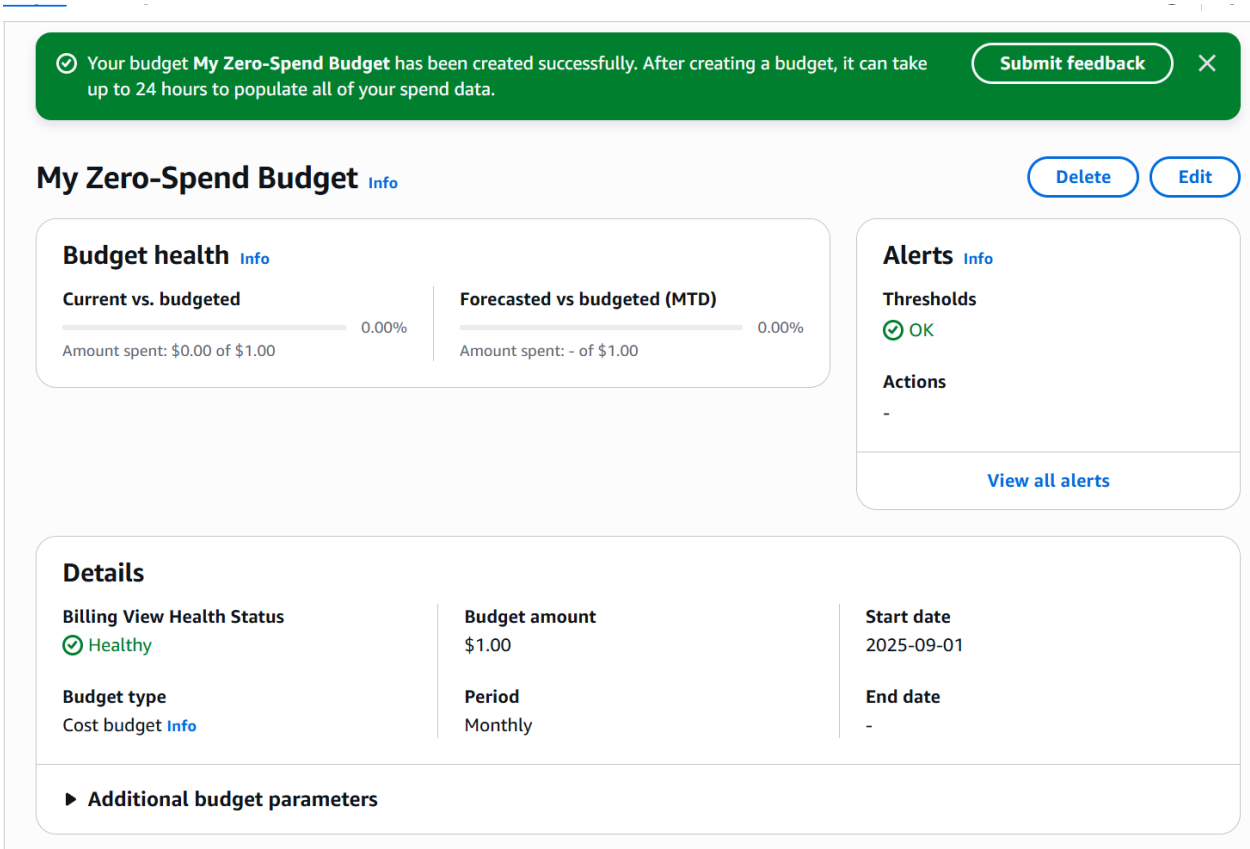
C2. Cost Management Dashboard

I set up AWS Cost Explorer to track expenses associated with running the application. The

dashboard is configured to show service-level costs, allowing me to see spending on Elastic Beanstalk, S3, and CloudFront. This helps ensure CloudCampusIQ remains within its budget.

C3. Monitoring and Cost Management Screenshots with Explanations





The CloudWatch alarm tracks health and CPU utilization, which is a key performance metric for ensuring the application scales appropriately under heavy load. The Budget alert dashboard shows budget tracking and alerts for managing costs.

C4. Explanation of Choices (C1–C3)

I selected a CPU utilization alert in CloudWatch because high processor usage is a direct indicator of performance issues, allowing the system to scale or administrators to respond before users are affected. Monitoring this metric ensures the application remains reliable during peak usage. I also implemented AWS Cost Explorer to track service-level spending since CloudCampusIQ must stay within its budget while supporting up to 10,000 concurrent users. Together, these tools provide a balance between performance and cost control, ensuring the platform can scale reliably without overspending.

D1. Summary of Project Parts

In section A, I implemented basic security controls by creating IAM roles with least-privilege access, enabling HTTPS with AWS Certificate Manager, and restricting network access through Security Groups. Logging was configured in CloudWatch to monitor traffic and provide audit visibility. In section B, I deployed the CloudCampusIQ application on AWS Elastic Beanstalk, which provided a managed PaaS environment that automatically handled scaling, load balancing, and deployment. Networking was configured to use the public DNS provided by Elastic Beanstalk with HTTPS enforced, ensuring secure global access. In section C, I configured CloudWatch alarms to monitor CPU utilization and created a cost dashboard in AWS Cost Explorer to track cloud expenses. These monitoring and cost tools ensure the application remains both reliable and financially sustainable. Overall, the solution balances performance, security, and cost efficiency while meeting CloudCampusIQ's requirement to support up to 10,000 concurrent users.

D2. Slide Deck Presentation

Slide 1 – Introduction

- CloudCampusIQ Cloud Solution: Secure, Scalable, and Cost-Efficient
- Supports up to 10,000 concurrent users
- Provides secure access to course materials
- Ensures global availability and performance

Slide 2 – Security Controls

- IAM roles created with least-privilege access
- HTTPS enabled with AWS Certificate Manager

- Security Groups restricted to port 443 (HTTPS only)
- CloudWatch logs enabled for access monitoring

Slide 3 – Application Deployment

- Application hosted on AWS Elastic Beanstalk (PaaS)
- Automatic scaling and load balancing
- Public DNS endpoint with HTTPS enforced
- Deployed sample app through Elastic Beanstalk console

Slide 4 – Monitoring & Cost Management

- CloudWatch alarm for CPU utilization > 70%
- AWS Cost Explorer dashboard tracks service-level costs
- Ensures reliability during peak usage
- Keeps monthly spending within budget limits

Slide 5 – Conclusion

- Secure and compliant platform for students and instructors
- Scales seamlessly to support global growth
- Reliable performance through monitoring and alerts
- Cloud costs: pay-as-you-go, ~$25K/month projected
- On-premises costs: ~$200K upfront for servers, storage, and networking + higher ongoing maintenance
- Cloud solution avoids over-provisioning and reduces IT overhead