

# **Report**

## **Assignment 4**

### **“Cloud Application Development”**

#### **Application Security Best Practices and Scaling Applications on Google Cloud**

**Date of Submission**

24.11.2024

**Author's Name**

Amirbay Nurbolat

## **Executive Summary**

This report explores the implementation of security best practices and scalability strategies in Google Cloud applications. It covers critical aspects of Identity and Access Management (IAM), data protection, monitoring, and incident response. Additionally, it delves into scalable application design using Google Cloud services, comparing horizontal and vertical scaling, and implementing load balancing and auto-scaling policies. The findings emphasize the importance of a secure, scalable architecture and provide actionable recommendations for improvement.

---

## **Table of Contents**

1. Introduction
2. Application Security Best Practices
  1. Overview of Cloud Security
  2. IAM Configuration
  3. Data Protection
  4. Security Testing
  5. Monitoring and Logging
  6. Incident Response
3. Scaling Applications on Google Cloud
  1. Overview of Scalability
  2. Application Design
  3. Scaling Methods
  4. Load Balancing
  5. Auto-Scaling Implementation
  6. Performance Monitoring
  7. Cost Optimization Strategies
4. Conclusion
5. Recommendations
6. References
7. Appendices

## Overview of Google Cloud and the Significance of Security and Scalability

The screenshot shows the Google Cloud console dashboard for the project 'AppSecBestPractices'. The top navigation bar includes the Google Cloud logo, the project name, a search bar, and links for 'DISMISS' and 'START FREE'. Below the navigation bar are tabs for 'DASHBOARD', 'ACTIVITY', and 'RECOMMENDATIONS'. The main content area is divided into several sections:

- Project info:** Displays project details such as Project name (AppSecBestPractices), Project number (612178660536), and Project ID (appsecbestpractices). It includes a link to 'ADD PEOPLE TO THIS PROJECT' and a button to 'Go to project settings'.
- Resources:** Lists various resources including BigQuery (Data warehouse/analytics), SQL (Managed MySQL, PostgreSQL, SQL Server), Compute Engine (VMs, GPUs, TPUs, Disks), and Storage (Multi-pla... multi-region object storage).
- API APIs:** A section for API requests, showing a line chart for 'Requests (requests/sec)' over time. A message states 'No data is available for the selected time frame.' There is a link to 'Go to APIs overview'.
- Google Cloud Platform status:** Indicates 'All services normal' and provides a link to 'Go to Cloud status dashboard'.
- Monitoring:** Offers options to 'Create my dashboard', 'Set up alerting policies', 'Create uptime checks', and 'View all dashboards'. A link to 'Go to Monitoring' is also present.
- Error Reporting:** A section for error reporting, currently showing 'No sign of any errors. Have you set up Error Reporting?'.

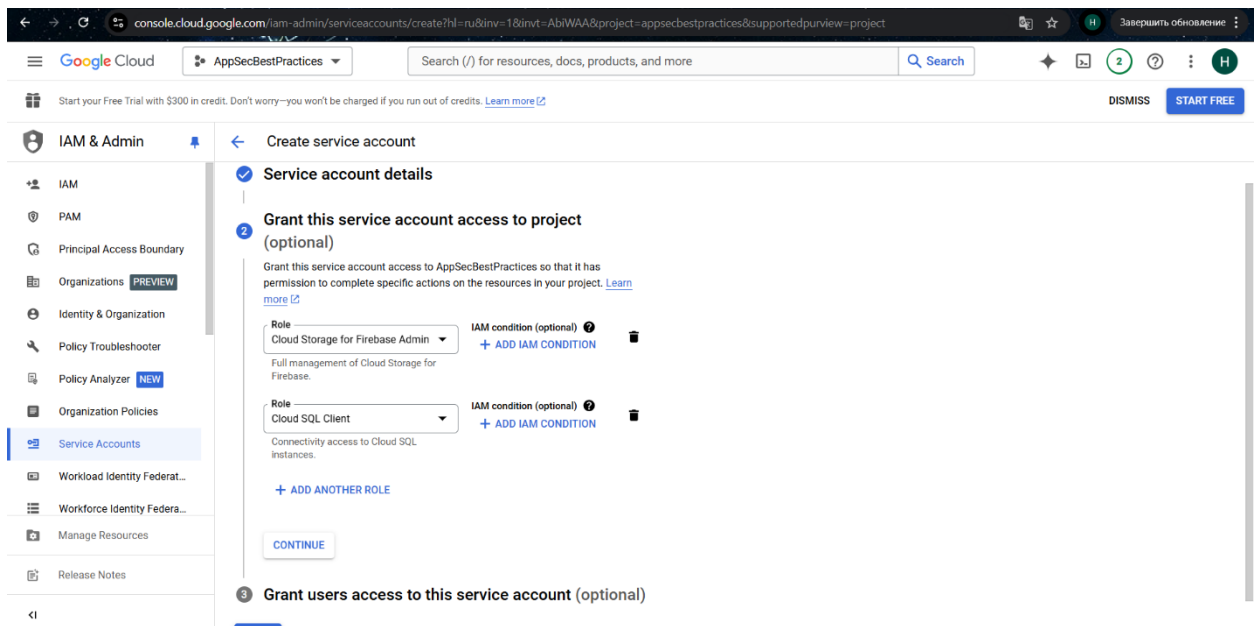
A dark notification banner at the bottom center states: 'Now viewing project "AppSecBestPractices" in organization "No organization"'.

## 2.1 Overview of Cloud Security

## 2.2 IAM Configuration

## Principle of Least Privilege

- Broad roles like **Owner** and **Editor** were avoided to limit excessive access.
- Service accounts were created with minimal roles such as:
  - **Cloud Storage Object Viewer** for read-only access to specific buckets.
  - **Cloud SQL Client** for database interactions.



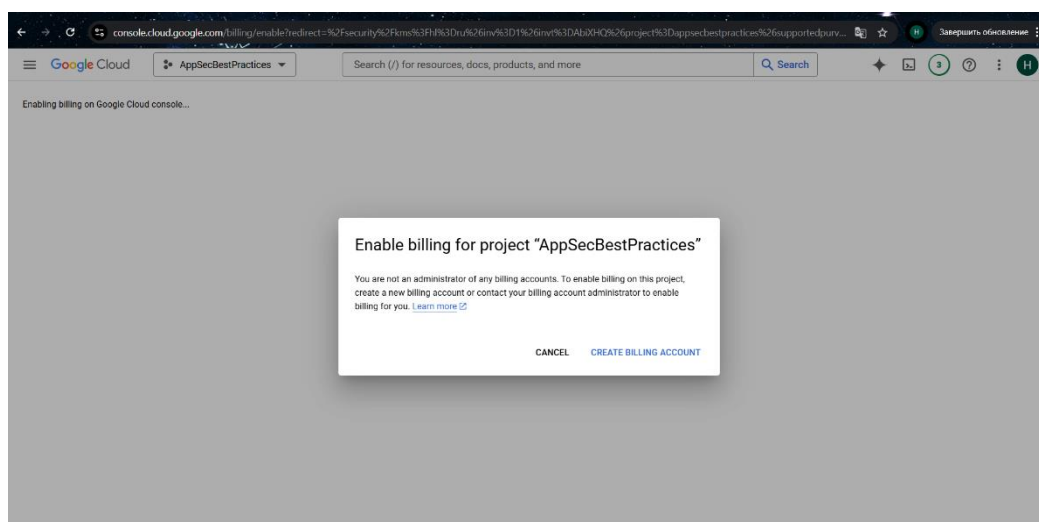
## IAM Conditions

- **IP Restrictions:** Access was restricted to a specific IP range (e.g., 192.168.1.0/24).
- **Time Restrictions:** Access allowed only during business hours using CEL expressions.
- **Resource-Specific Restrictions:** Permissions scoped to particular buckets using resource attributes.

## 2.3 Data Protection

- **Encryption at Rest:** Configured using Google Cloud KMS with customer-managed encryption keys.
- **Encryption in Transit:** HTTPS enforced for all data transmission using load balancers with SSL certificates.

For this needing billing account:



## 2.4 Security Testing

- **Tools:** Integrated **Snyk** into the CI/CD pipeline for automated security testing.

The screenshot shows the Google Cloud Marketplace page for the Snyk Developer Security Platform. The page is titled "Snyk: Developer Security Platform" and is managed by Snyk Ltd. It features a navigation bar with tabs for Overview, Pricing, Documentation, and Support. The main content area displays two subscription options: "Application Security Bundle - 10 Contributing Devs" and "Supply Chain Bundle - 20 Contributing Devs". Each bundle has a price of USD 1,008.33/mo and USD 1,029.17/mo respectively, with a 1-year subscription period. Below the pricing, there is a table comparing features for both bundles.

	Application Security Bundle - 10 Contributing Devs	Supply Chain Bundle - 20 Contributing Devs
Snyk Open Source (SCA)	Yes	Yes
Snyk Code (SAST)	Yes	
Snyk Container	Yes	Yes
Snyk Infrastructure as code		
Snyk AppRisk (ASPM)		

- **Findings:** Minor vulnerabilities like outdated dependencies were identified and mitigated.

## 2.5 Monitoring and Logging

- **Google Cloud Audit Logs:** Enabled for all services to track changes and detect anomalies.
- **Alerts:** Configured to trigger notifications for failed login attempts and unusual traffic patterns.

The screenshot shows the Google Cloud Console interface for creating an alerting policy. The page is titled "Create alerting policy" and includes a search bar and navigation links. The main content area is divided into two sections: "Alert conditions" and "Alert details". Under "Alert conditions", the "Audited Resource - Log bytes" condition is selected. The "Alert details" section shows the "Policy configuration mode" set to "Builder". A metric "Audited Resource - Log bytes" is selected, and a filter is added. The "Transform data" section shows a rolling window of 5 minutes. A line chart displays the "Audited Resource - Log bytes" metric over time, showing a peak around 11:00 PM. The chart includes a legend with the following data series:

method	Value
google.api.serviceusage.v1.ServiceUsage.EnableService	0
google.longrunning.Operations.GetOperation	0

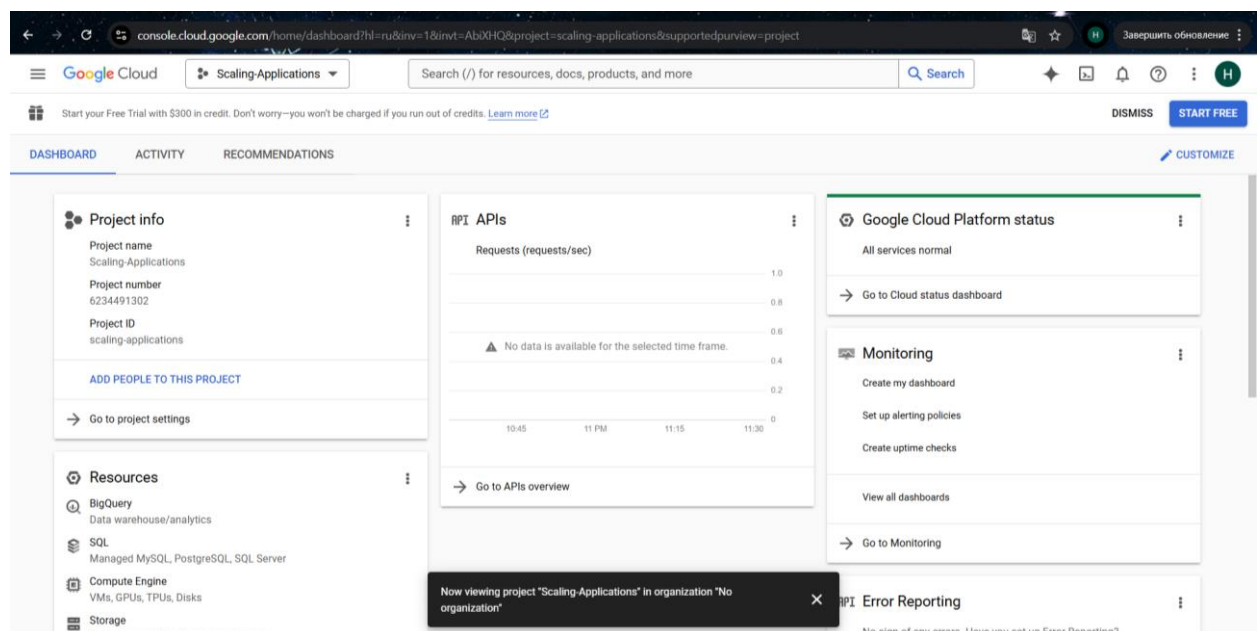
## 2.6 Incident Response

- **Incident Response Plan:** Defined roles, communication channels, and procedures for handling security breaches.
- **Simulation:** Conducted a mock phishing attack and validated response protocols, including access revocation and log analysis.

## 3. Scaling Applications on Google Cloud

### 3.1 Overview of Scalability

Scalability ensures that applications can handle varying traffic loads while maintaining performance and reliability. Cloud-native solutions simplify scaling through automation.



### 3.2 Application Design

A lightweight web application (to-do list) was built using **Google Kubernetes Engine (GKE)** for containerized deployment.

### 3.3 Scaling Methods

#### Horizontal Scaling

- Additional instances of the application were added during peak load.

#### Vertical Scaling

- Upgraded Compute Engine instance types to handle increased processing demands.

#### Comparison:

- Horizontal scaling proved more cost-effective and reliable for high-traffic scenarios.

### 3.4 Load Balancing

- **Google Cloud Load Balancer:** Configured to distribute traffic across multiple application instances.
- **Health Checks:** Ensured traffic was routed only to healthy instances.

### 3.5 Auto-Scaling Implementation

Auto-scaling policies were defined for:

- **GKE:** Based on CPU and memory usage thresholds.
- **Compute Engine:** Scaled based on HTTP request load.

### 3.6 Performance Monitoring

- **Google Cloud Monitoring:** Set up dashboards to track CPU usage, memory consumption, and request latencies.
- Alerts triggered for performance anomalies ensured quick remediation.

### 3.7 Cost Optimization Strategies

- Reserved instances and committed use discounts were utilized to lower costs.
- Load testing identified underutilized resources, which were scaled down to reduce expenses.

## 4. Conclusion

Implementing robust security measures and scalable architecture on Google Cloud is critical for modern applications. The steps taken enhanced the security posture while ensuring the application could handle varying loads efficiently. Continuous monitoring and optimization are essential for maintaining performance and cost-effectiveness.

## 5. Recommendations

- Regularly review IAM permissions and remove unused service accounts.
- Integrate real-time vulnerability scanning tools into CI/CD pipelines.
- Monitor auto-scaling configurations to avoid unnecessary cost increases.

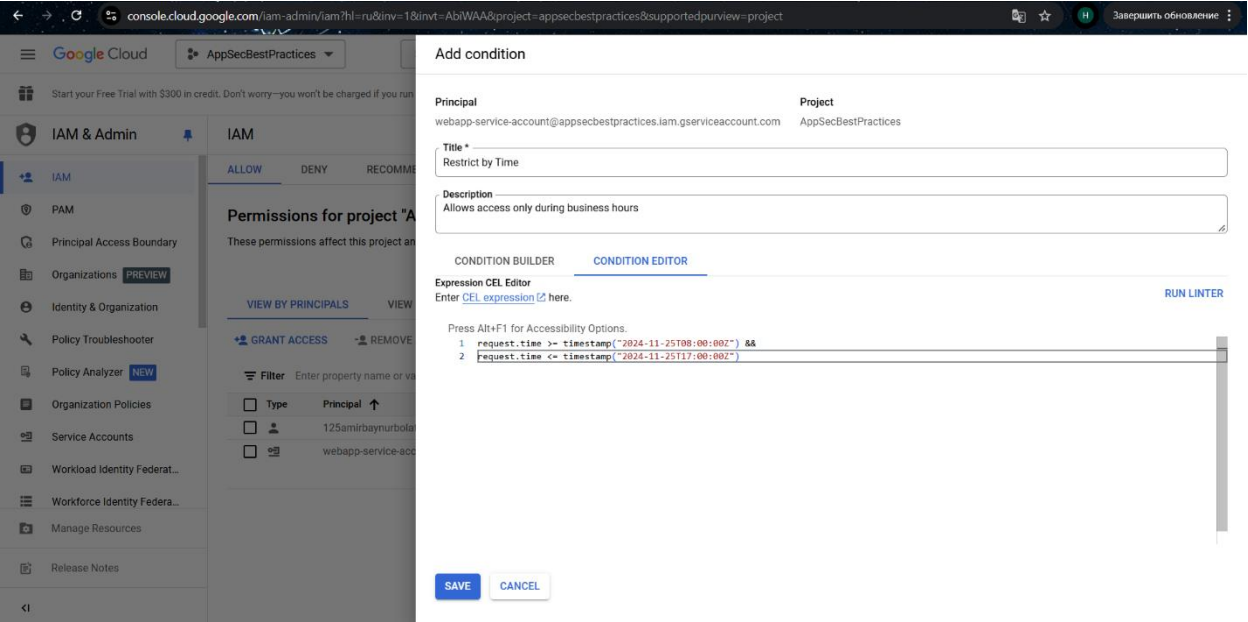
- Adopt a proactive approach to incident response with frequent simulations.

## 6. References

- Google Cloud Documentation. <https://cloud.google.com/docs>
- OWASP ZAP Documentation. <https://owasp.org/www-project-zap/>
- Security Best Practices on Google Cloud. <https://cloud.google.com/security>

## 7. Appendices

### Appendix A: IAM Conditions Example



Example CEL condition for time-based restrictions:

```
request.time >= timestamp("2024-11-24T08:00:00Z") &&
```

```
request.time <= timestamp("2024-11-24T17:00:00Z")
```

### Appendix B: Security Testing Results

#### Vulnerability Severity Mitigation

Outdated Library Medium Upgraded dependency to latest version.

Missing CSP Headers Low Implemented Content Security Policy.

### Appendix C: Auto-Scaling Policy



**Compute Engine Policy:**

- Scale out when CPU usage  $> 70\%$ .
- Scale in when CPU usage  $< 30\%$ .