**Cloud-Based Ransomware Detection and Recovery System on GCP**

**1. SETUP GCP PROJECT & BILLING**

- Console > IAM & Admin > Manage Resources > Create Project - Console > Billing > Link billing account to project - Enable APIs (Cloud Shell):

gcloud services enable logging.googleapis.com bigquery.googleapis.com \ cloudfunctions.googleapis.com cloudscheduler.googleapis.com \ pubsub.googleapis.com securitycenter.googleapis.com

**2. DETECTION SYSTEM**

a. Enable Data Access Logs:

* Console > IAM & Admin > Audit Logs > Enable for Cloud Storage & IAMb. BigQuery Logging Sink:
* Console > Logging > Logs Router > Create Sink > BigQuery
* Filter:

resource.type=("gcs\_bucket" OR "project") AND

(protoPayload.serviceName="storage.googleapis.com" OR protoPayload.serviceName="iam.googleapis.com")

b. Anomaly Detection Function (Python):

Runs every 5 minutes via Cloud Scheduler + Pub/Sub.

Backup Function Code (Python):

**3. RECOVERY SYSTEM**

a. Enable Object Versioning:

- Console > Cloud Storage > Bucket > Edit > Enable Versioning

b. Backup Function (Python):

Triggered by Cloud Scheduler via Pub/Sub daily.

**4. MITIGATION SYSTEM**

a. IAM Least-Privilege:

* Console > IAM & Admin > IAM > Review members- Remove Owner/Editor roles not needed; assign specific roles
* b. Network Firewall:
* Console > VPC > Firewall Rules
* Disable default-allow-ssh/rdp or restrict to internal IPs- Add deny-all and explicit allow rules c. Security Command Center: - Console > Security > Enable Standard Tier
* Monitor findings and alerts for risky configurations

**5. TESTING**

a. Mass File Encryption Simulation:

* Upload and overwrite 50+ files in main bucket using a script- Check if alert email is received, restore from versioned bucket

b. IAM Lockout Simulation:

* Modify IAM roles to simulate unauthorized changes
* Expect detection alert and use backup account to revert

**6. DELIVERABLES**

* GCP prototype with detection, mitigation, and recovery systems- GitHub repository with:
* detection/anomaly\_detector\_function.py
* recovery/backup\_function.py
* mitigation/firewall\_rules.yaml
* Architecture diagram (PNG)
* Test report (PDF/DOC)