FinOps Dashboard Project Report

Introduction

The FinOps Dashboard is a simulation-based project aimed at helping users understand cloud cost monitoring and optimization practices without using actual cloud billing APIs. By generating mock usage data and visualizing it in Grafana, the dashboard provides a cost breakdown of multiple cloud services in a locally hosted environment.

Objective

To design a lightweight, offline dashboard to simulate daily cloud spending across services such as Compute Engine, Cloud Functions, and Cloud Storage using Python, SQLite, and Grafana.

Tech Stack

- Python 3: Used for generating mock data
- SQLite: Local database to store usage metrics
- Grafana: Visualization of usage and cost data

Architecture Workflow

1. Python script (`mock_usage.py`) simulates usage data.
2. Data is stored in `usage.db` (SQLite format).
3. Grafana connects to the database for dashboard creation and data visualization.

FinOps Dashboard Project Report

Mock Data Generation

The script `mock_usage.py` generates mock billing data for selected services by assigning random daily usage and cost values. The data covers a range of dates and service categories for a realistic overview.

Database Setup

The script outputs the data into a SQLite database (`usage.db`). Each record consists of fields like date, service type, usage amount, and cost.

Grafana Dashboard

Grafana is configured to read from the SQLite database. Users can create custom dashboards to visualize trends, such as daily spend per service or total cloud cost over time.

Observations

- Clear visualization of how each cloud service contributes to daily spend
- Ability to simulate cost anomalies or spikes for test scenarios
- Local deployment ensures no dependency on actual cloud environments

Conclusion

This project provides a practical understanding of FinOps practices by simulating cost data in an offline setting. It demonstrates how data can be effectively tracked and visualized to make informed budgeting and scaling decisions.