

Cost Optimization through Secret Version Cleanup in GCP Secret Manager

Objective:

The goal of this initiative is to optimize costs associated with managing secrets in GCP Secret Manager by implementing a process to destroy older versions, avoiding unnecessary storage costs.

Background:

In our infrastructure, we utilize GCP Secret Manager to store sensitive information and configuration data securely. Over time, as secrets are updated, GCP Secret Manager creates new versions. However, retaining older versions can lead to increased storage costs.

Previous Approach:

Previously, each update to a secret created a new version, and older versions were retained. This led to unnecessary storage costs, as all versions, including outdated ones, were kept in the Secret Manager.

Optimization Strategy:

To reduce costs associated with storing older versions of secrets, we implemented a script in our Jenkins pipeline. The script identifies the latest version of a secret and destroys the older versions, thereby minimizing storage requirements and associated costs.

Implementation Steps:

Script Creation:

- A Bash script (`destroy_old_versions.sh`) was created to interact with GCP Secret Manager.
- The script identifies the latest version of a secret and destroys all older versions.

Jenkins Pipeline Integration:

- The script was integrated into our Jenkins pipeline under a dedicated stage.
- The pipeline runs periodically or as needed to keep the secret versions optimized.

Execution Flow:

- The script fetches all versions of a secret from GCP Secret Manager.
- It identifies the latest version and destroys all older versions.
- Any errors during the destruction process are handled, ensuring the pipeline continues to the next secret.

Results:

Cost Savings:

- By removing older versions, unnecessary storage costs associated with maintaining outdated secrets have been significantly reduced.

Storage Optimization:

- The Secret Manager now only retains the latest version of each secret, optimizing storage and access times.

Enhanced Security:

- Outdated secrets pose a security risk. By automating version cleanup, the system ensures that only the latest and most secure versions are retained.

Conclusion:

Implementing the secret version cleanup script in our Jenkins pipeline has resulted in tangible benefits, including cost savings, storage optimization, and enhanced security. By periodically removing older versions of secrets, we have achieved a more streamlined and cost-effective management of sensitive information in GCP Secret Manager.