Cloud Operations

Cloud Monitoring

In 28 Minutes

- To operate cloud applications effectively, you should know:
  - Is my application healthy?
  - Are the users experiencing any issues?
  - Does my database has enough space?
  - Are my servers running in an optimum capacity?
- **Cloud Monitoring** - Tools to monitor your infrastructure
  - Measures key aspects of services (Metrics)
  - Create visualizations (Graphs and Dashboard)
  - Configure Alerts (when metrics are NOT healthy)
    - Define Alerting Policies:
      - Condition
      - Notifications - Multiple channels
      - Documentation

Cloud Monitoring - Workspace

- You can use Cloud Monitoring to monitor one or more GCP projects and one or more AWS accounts
- How do you group all the information from multiple GCP projects or AWS Accounts?
- **Create a Workspace**
- Workspaces are needed to organize monitoring information
  - A workspace allows you to see monitoring information from multiple projects
  - Step I: Create workspace in a specific project (Host Project)
  - Step II: Add other GCP projects (or AWS accounts) to the workspace

Cloud Monitoring - Virtual Machines

- **Default metrics monitored** include:
  - CPU utilization
  - Some disk traffic metrics
  - Network traffic, and
  - Uptime information
- Install **Cloud Monitoring agent** on the VM to get more disk, CPU, network, and process metrics:
  - collectd-based daemon
  - Gathers metrics from VM and sends them to Cloud Monitoring

Cloud Logging

In28 Minutes

- Real time log management and analysis tool
- Allows to store, search, analyze and alert on massive volume of data
- Exabyte scale, fully managed service
  - No server provisioning, patching etc
- Ingest Log data from any source
- Key Features:
  - Logs Explorer - Search, sort & analyze using flexible queries
  - Logs Dashboard - Rich visualization
  - Logs Metrics - Capture metrics from logs (using queries/matching strings)
  - Logs Router - Route different log entries to different destinations

Cloud Logging - Collection

In28 Minutes

- Most **GCP Managed services automatically send logs** to Cloud Logging:
  - GKE
  - App Engine
  - Cloud Run
- Ingest logs from GCE VMs:
  - Install **Logging Agent** (based on fluentd)
  - (Recommended) Run Logging Agent on all VM instances
- Ingest logs from on-premises:
  - (Recommended) Use the BindPlane tool from Blue Medora
  - Use the Cloud Logging API

Cloud Logging - Audit and Security Logs

In 28 Minutes

- **Access Transparency Log:** Captures Actions performed by GCP team on your content (NOT supported by all services):
  - ONLY for organizations with Gold support level & above
- **Cloud Audit Logs**: Answers who did what, when and where:
  - Admin activity Logs
  - Data Access Logs
  - System Event Audit Logs
  - Policy Denied Audit Logs

Cloud Logging - Audit Logs

In 28 Minutes

- Which service?
  - **protoPayload.serviceName**
- Which operation?
  - **protoPayload.methodName**
- What resource is audited?
  - **resource.Type**
- Who is making the call?
  - **authenticationInfo.principalEmail**


```
{
  protoPayload: {
    @type: "type.googleapis.com/google.cloud.audit.AuditLog",
    status: {},
    authenticationInfo: {
      principalEmail: "user@example.com"
    },
    serviceName: "appengine.googleapis.com",
    methodName: "SetIamPolicy",
    authorizationInfo: [...],
    serviceData: {
      @type: "type.googleapis.com/google.appengine.legacy.AuditData",
      policyDelta: { bindingDeltas: [
          action: "ADD",
          role: "roles/logging.privateLogViewer",
          member: "user:user@example.com"
      ], }
    },
    request: {
      resource: "my-gcp-project-id",
      policy: { bindings: [...], }
    },
    response: {
      bindings: [
        {
          role: "roles/logging.privateLogViewer",
          members: [ "user:user@example.com" ]
        }
      ],
    }
  },
  insertId: "53179D9A9B559.AD6ACC7.B40604EF",
  resource: {
    type: "gae_app",
    labels: { project_id: "my-gcp-project-id" }
  },
  timestamp: "2019-05-27T16:24:56.135Z",
  severity: "NOTICE",
  logName: "projects/my-gcp-project-id/logs/cloudaudit.googleapis.com%2Factivity",
}
```

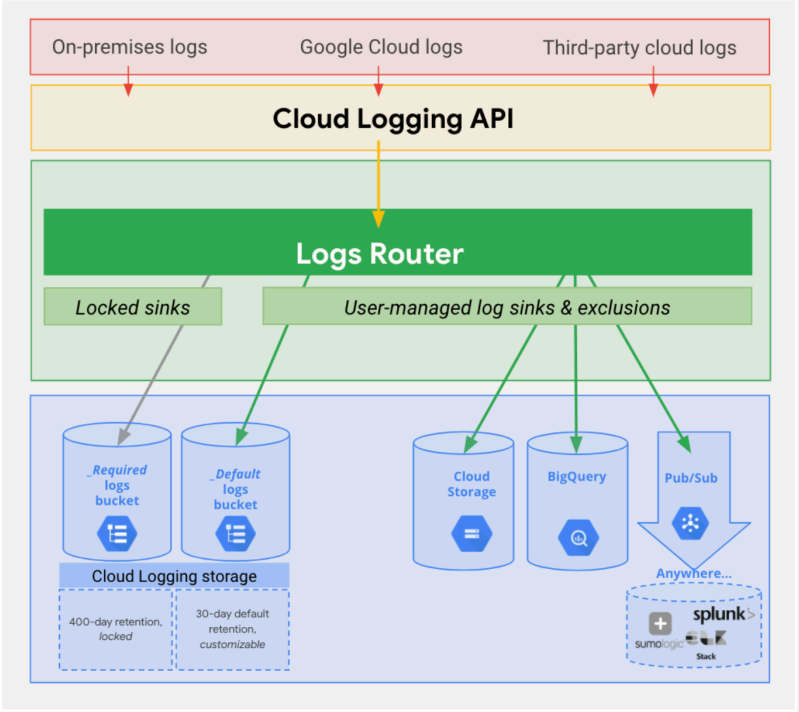
Cloud Audit Logs

In28 Minutes

Feature	Admin Activity Logs	Data Access Logs	System Event Logs	Policy Denied Logs
Logs for	API calls or other actions that modify the configuration of resources	Reading configuration of resources	Google Cloud administrative actions	When user or service account is denied access
Default Enabled	✓	X	✓	✓
VM Examples	VM Creation, Patching resources, Change in IAM permissions	Listing resources (vms, images etc)	On host maintenance, Instance preemption, Automatic restart	Security policy violation logs
Cloud Storage	Modify bucket or object	Modify/Read bucket or object		
Recommended	Logging/Logs Viewer	Logging/Private	Logging/Logs Viewer	Logging/Logs

Cloud Logging - Controlling & Routing

- How do you manage your logs?
  - Logs from various sources reaches **Log Router**
  - Log Router checks against configured rules
    - What to ingest? what to discard?
    - Where to route?
- Two types of Logs buckets:
  - **_Required**: Holds Admin activity, System Events & Access Transparency Logs (retained for 400 days)
    - ZERO charge
    - You cannot delete the bucket
    - You cannot change retention period
  - **_Default**: All other logs (retained for 30 days)
    - You are billed based on Cloud Logging pricing
    - You cannot delete the bucket:
      - But you can disable the **_Default** log sink route to disable ingestion!
    - You can edit retention settings (1 to 3650 days (10 years))



source: (*https://cloud.google.com*)

Cloud Logging - Export

In28 Minutes

- Logs are ideally stored in Cloud Logging for limited period
  - For long term retention (Compliance, Audit) logs can be exported to:
    - Cloud Storage bucket (ex: `bucket/syslog/2025/05/05`)
    - Big Query dataset (ex: `tables syslog_20250505 > columns timestamp, log`)
    - Cloud Pub/Sub topic (base64 encoded log entries)
- How do you export logs?
  - Create **sinks** to these destinations using Log Router:
    - You can create **include** or **exclude** filters to limit the logs

Cloud Logging - Export - Use Cases

In28 Minutes

- Use Case 1: Troubleshoot using VM Logs:
  - Install Cloud logging agent in all VM's and send logs to Cloud Logging
  - Search for logs in Cloud Logging
- Use Case 2: Export VM logs to BigQuery for querying using SQL like queries:
  - Install Cloud logging agent in all VM's and send logs to Cloud Logging
  - Create a BigQuery dataset for storing the logs
  - Create an export sink in Cloud Logging with BigQuery dataset as sink destination
- Use Case 3: You want to retain audit logs for external auditors at minimum cost
  - Create an export sink in Cloud Logging with Cloud Storage bucket as sink destination
  - Provide auditors with Storage Object Viewer role on the bucket
  - You can use Google Data Studio also (for visualization)

Cloud Trace

In 28 Minutes

- Distributed tracing system for GCP: Collect latency data from:
  - Supported Google Cloud Services
  - Instrumented applications (using tracing libraries) using **Cloud Trace API**
- Find out:
  - How long does a service take to handle requests?
  - What is the average latency of requests?
  - How are we doing over time? (increasing/decreasing trend)
- Supported for:
  - **Compute Engine, GKE, App Engine (Flexible/Standard)** etc
- Trace client libraries available for:
  - **C#, Go, Java, Node.js, PHP, Python & Ruby**

Cloud Debugger

In28 Minutes

- How to debug issues that are happening only in test or production environments?
- **Cloud Debugger**: Capture state of a running application
  - Inspect the state of the application directly in the GCP environment
  - Take snapshots of variables and call stack
  - No need to add logging statements
  - No need to redeploy
  - Very lightweight => Very little impact to users
    - Can be used in any environment: Test, Acceptance, Production

Cloud Profiler

In28 Minutes

- How do you identify performance bottlenecks in production?
- **Cloud Profiler** - Statistical, low-overhead profiler
  - Continuously gathers CPU and Memory usage from production systems
  - Connect profiling data with application source code
    - Easily identify performance bottlenecks
  - Two major components:
    - Profiling agent (collects profiling information)
    - Profiler interface (visualization)

Error Reporting

In 28 Minutes

- How do you identify production problems in real time?
- Real-time exception monitoring:
  - Aggregates and displays errors reported from cloud services (using stack traces)
  - **Centralized Error Management console**:
    - Identify & manage top errors or recent errors
  - Use **Firebase Crash Reporting** for errors from Android & iOS client applications
  - Supported for Go, Java, .NET, Node.js, PHP, Python, and Ruby
- Errors can be reported by:
  - Sending them to Cloud Logging OR
  - By calling Error Reporting API
- Error Reporting can be accessed from desktop
  - Also available in the Cloud Console mobile app for iOS and Android

Cloud Logging, Monitoring .. - Stackdriver

Stackdriver Service	New Service Name
Stackdriver Monitoring	Cloud Monitoring
Stackdriver Logging	Cloud Logging
Stackdriver Error Reporting	Error Reporting
Stackdriver Trace	Cloud Trace
Stackdriver Profiler	Cloud Profiler

Cloud Operations Scenarios

In28 Minutes

Scenario	Solution
You would like to record all operations/requests on all objects in a bucket (for auditing)	Turn on data access audit logging for the bucket
You want to trace a request across multiple microservices	Cloud Trace
You want to identify prominent exceptions (or errors) for a specific microservice	Error Reporting
You want to debug a problem in production by executing step by step	Cloud Debugger
You want to look at the logs for a specific request	Cloud Logging