Google Cloud Platform Forensics

Overview of Incident Response in GCP

Cloud compromise involves understanding existing infrastructure and investigating malicious activity derived from control plane activity characterized by four categories of cloud services:

1. **Identity and Access (Admin Console, IAM, Cloud KMS)**
2. **Compute (GCE, GAE, GKE, Cloud Functions, PubSub)**
3. **Storage (GCS, Persistent Disk, Local SSD)**
4. **Database (Cloud SQL, BigQuery, Firestore)**

Here are five categories of forensic data that play a critical role during incident response:

1. **Alerts (SIEM, SCC, Cloud Armor & Alert Center)**
2. **Logs**
3. **Configurations (role bindings, service metadata, etc.)**
4. **Reports (Cloud Billing & Admin Console)**
5. **Service Data (Compute Workload, GCS bucket, etc.)**

GCP native tooling for threat hunting and incident response:

- **Security Command Center**
- **Logs Explorer**
- **BigQuery**
- **Metrics Explorer**
- **Policy Analyzer**
- **Asset Inventory**

Logs for Threat Hunting and Incident Response

Admin Console Cloud Logging

Category	Log Name	Description	Default Enabled	Location	Default Retention
Identity	Admin	Tracks actions performed in the Google Admin console	Yes	Admin Console > Reporting > Audit	180 days
	User	Tracks user events (logins, password change, 2FA, etc)	Yes	Admin Console > Reporting > Audit	180 days
	OAuth	Tracks 3rd party data access requests and app usage	Yes	Admin Console > Reporting > Audit	180 days
	SAML	Tracks successful and failed sign-ins to SAML apps	Yes	Admin Console > Reporting > Audit	180 days
	Groups	Tracks changes to groups and memberships via Groups interface	Yes	Admin Console > Reporting > Audit	180 days
	Groups Enterprise	Tracks changes to groups and memberships via Admin console, Cloud console, Admin SDK API, Cloud Identity API, and Groups user interface	Yes	Admin Console > Reporting > Audit	180 days
Security	Admin Activity	Tracks API calls and other actions that modify the configuration or metadata	Yes	Log Bucket (_Required)	400 days
	Data Access	Tracks API calls that read the configuration or metadata of resources; additionally, this tracks user-driven API calls that create, modify, or read user-provided resource data	No	Log Bucket (_Default)	30 days
	Policy Denied	Tracks when a GCP service denies access to a member due to a security policy violation, which must be manually configured in VPC Service Controls	Yes	Log Bucket (_Required)	30 days
Platform	VPC Flow	Tracks network connection metadata to and from VM instances (including instances used as GKE nodes)	No	Log Bucket (_Default)	30 days
	GCS Usage	Tracks HTTP web requests made to a specific bucket	No	Configured Storage Bucket	N/A
	Cloud DNS	Tracks queries that name servers resolve for VPC networks	No	Log Bucket (_Default)	30 days
	Firewall Rules	Tracks the effects of configured VPC firewall rules	No	Log Bucket (_Default)	30 days
	HTTP/S Load Balancing	Tracks load balanced web connections to backend services	No	Log Bucket (_Default)	30 days
User-written	Agent	Host-based logs (standard OS logs, metrics, etc) collected from Compute-oriented service resources (GCE instances, GKE nodes, etc)	No	Log Bucket (_Default)	30 days

Log Analysis in GCP

Security Log Analysis

- Identify anomalies in creation, deletion, and modification events across Identity and Access, Compute, Storage, Database, and Data Transfer services, while attributing actions to user or service accounts
  - Identify and pivot on indicators of compromise (affected user accounts, service accounts, service account keys, IPs, user agents, resource data)

- **Identify anomalous behavior in GCP:**
  - Irregular status codes [1]
    - Code 3: Invalid Argument
    - Code 5: Not Found
    - Code 7: Permission Denied
    - Code 9: Failed Precondition
    - Code 16: Unauthenticated
  - Service account impersonation
  - Service account creation, deletion, or modification events
  - User account creation, deletion, or modification events
  - IAM (policy bindings, org policy, etc) creation, deletion, or modification events
  - Key creation, deletion, or modification events
  - Compute creation, deletion, modification, enumeration, or access events
  - VPC Firewall rule creation, modification, or enumeration events
  - Bucket creation, deletion, enumeration, modification, or access events
  - Database deletion, modification, enumeration, or access events

- **Actionable fields for threat hunting and incident response:**
  - protoPayload.methodName
  - timestamp
  - protoPayload.authenticationInfo.principalEmail
  - protoPayload.authenticationInfo.serviceAccountDelegationInfo[]
  - protoPayload.resourceName
  - protoPayload.requestMetadata.callerIp
  - protoPayload.requestMetadata.callerSuppliedUserAgent
  - protoPayload.authorizationInfo.granted
  - protoPayload.status.code
  - protoPayload.status.details[].fieldViolations[].description
  - logName

Identity Log Analysis

- **Common analysis via Admin, User, Groups, and Groups Enterprise logs** [3, 4a, 4b, 5, 6]
  - Login events by IP/login type/frequency (User log)
  - Login events that occur without 2-Step Verification (User log)
  - High-risk user-driven events (2sv_disable, password_edit, suspicious_login*)
  - High-risk admin-driven events (create_user, change_password, assign_role)
  - Group membership modification (add_member, remove_member, add_member_role, remove_member role, change_security_setting add_service_account_permission, remove_service_account_permission)

- **Common analysis for OAuth log** [7]
  - Activity (should "X" app call "Y" method on behalf of "Z" actor?)
  - Authorize (should "X" app be authorized the "Y" scopes by "Z" actor?)
  - Revoke (should "X" app have its "Y" scopes removed by "Z" actor?)

- **Common analysis via SAML log** [8]
  - Successful login events by IP/app name/frequency
  - Failed login events by IP/app name/failure type/frequency
    - failure_request_denied
    - failure_malformed_request
    - failure_app_not_configured_for_user
    - failure_app_not_enabled_for_user

Platform Log Analysis

- **VPC Flow log** [9]
  - Anomalous IPs (OSINT/Geo/Threat Intel Feeds)
  - Remote access port activity (21, 22, 23, 445, 3389)
  - Staging (high ingress bytes)
  - Exfiltration (high egress bytes)

- **GCS Usage** [10, 11]
  - Access and Exfiltration (GET_Object)
  - Destruction (DELETE_Bucket, DELETE_Object)
  - Manipulation (PUT_Object)

- **Cloud DNS** [12]
  - DNS communication over TCP with an unknown server (potential C2)
  - Unusual DNS query failures (jsonPayload.responseCode: "ERROR")
  - Unusual domain names (OSINT, DGA, Fast Flux, irregular TLDs, newly registered)

- **Firewall Rules** [13]
  - Disposition (DENIED vs ALLOWED)

- **HTTP/S Load Balancing** [14]
  - Anomalous IPs and user agents (OSINT/Geo/Threat Intel Feeds)
  - Abnormal request URIs, GET vs POST ratios, and return code ratios
  - Web request length or payload size
  - URIs missing or uncommon HTTP referrer

Detecting Service Account Impersonation

Impersonation involves giving a principal (user or service account) permission to a service account without having to give direct permission to the end resource [2].

Detecting anomalous impersonation:

- Upon impersonation, the impersonee must generate an access token. This event produces the **GenerateAccessToken** API call in the **protoPayload.methodName** field. Each identity involved in the impersonation activity is tracked through the **protoPayload.metadata.identityDelegationChain[].<#>** field.

- To determine what action occurred via impersonation, identify log events where the **protoPayload.authenticationInfo.serviceAccountDelegationInfo[].firstPartyPrincipal.principalEmail** field exists and observe the corresponding API method. The first party principal field represents the impersonator. The **protoPayload.authenticationInfo.principalEmail** field represents the impersonee.

Interpreting "Caller Identities" in Security Logs

Principal Email

protoPayload.authenticationInfo.principalEmail [15, 16]

- If the domain is "**gserviceaccount.com**", a service account is responsible for the event
- If the domain is "**google.com**", a Google service is responsible for the admin event
- If the domain is "**gmail.com**", an unmanaged* Google account is responsible for the event
- If the domain is "**<example>.com**", a managed Google account/group is responsible for the event
- If the "protoPayload.methodName" is a read-only operation ("Get", "List", etc.) and fails with a "permission denied", status code 7, the principal email field will be absent from the log event unless it is a service account

IP Address

protoPayload.requestMetadata.callerIp [17, 18]

- When the Caller IP is "**<Public IPv4 or IPv6>**", caller is from the Internet
- When the Caller IP is "**<GCE VM external IPv4>**", caller is from a GCE VM with an external IP
- When the Caller IP is "**<GCE VM internal IPv4>**", caller is from a GCE VM without an external IP and is in the same organization or project
- When the Caller IP is "**gce-internal-ip",** caller is from a GCE VM without an external IP and is in a different organization or project
- When the Caller IP is "**private**", caller is communicating from inside Google's internal network between Google Cloud services

User Agent

protoPayload.requestMetadata.callerSuppliedUserAgent [19, 20]:

- When the User Agent contains strings correlating to a programming language (e.g., "**google-api-python-client/1.4.0**"), the request was made by an application or script using the Google API
- When the User Agent contains strings correlating to a Cloud SDK command line tool such as gcloud, gsutil, or bq (e.g., "**Cloud SDK Command Line Tool apitools-client/1.0 gcloud/0.9.62**"), the request was made manually by a user or dynamically through a script
- When the User Agent contains strings correlating to a Google Cloud service (e.g., "**AppEngine-Google; (+http://code.google.com/appengine; appid: s~my-project**"), the request was made by a Google Cloud service

Google Cloud Platform Forensics

Service Accounts Deconstructed

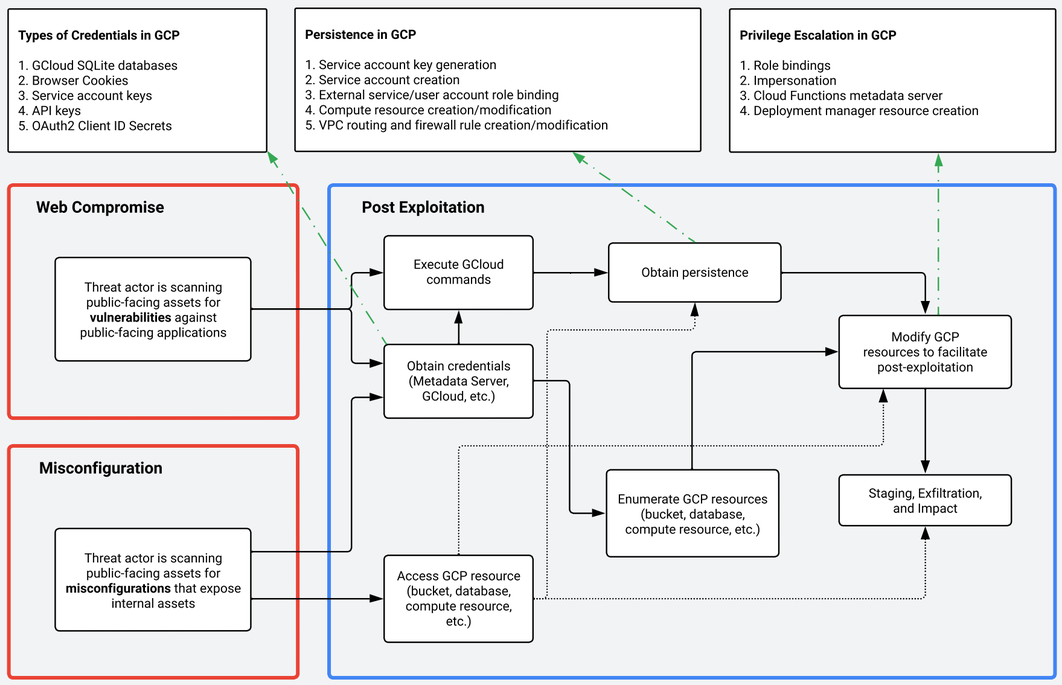
Service Account Type	Description	Format	Key Takeaways
User-managed [20]	Service accounts created and managed by users	**User Established** <service-account-name>@<project-id>.iam.gserviceaccount.com	• Authenticate and authorize applications and services (often used as "identities for workloads") • Manage and audit access to Google Cloud resources • Impersonate users
Default [21]	User-managed service accounts created automatically when certain Google Cloud services are enabled	**App Engine** <project-id>@appspot.gserviceaccount.com **Compute Engine** <project-number>-compute@developer.gserviceaccount.com	• By default, assigned Editor role at project-level • Compute Engine default service account is utilized by GKE, Cloud Run, Cloud Functions, and GCE • Associated with enabled GCP services • By default, access is limited by "access scopes" (legacy)
Google-managed [22]	Google-created and Google-managed service accounts that allow services to access resources on your behalf	**Google APIs Service Agent** <project-number>@cloudservices.gserviceaccount.com **Role Manager** service-agent-manager@system.gserviceaccount.com	• Automatically granted Editor role at project-level ◦ Runs internal Google processes on your behalf • Manages service auditing and roles granted to other Google-managed service

GCP Attack Matrix

Initial Access	Discovery	Credential Access & Lateral Movement	Privilege Escalation	Persistence & Execution	Defense Evasion	Exfiltration
Compromised Valid Account	IAM Enumeration	Access Secret Manager	Service Account Impersonation	User Account Creation	Audit Log Modification	Cloud Source Repository Exfiltration
Web Service Compromise	Compute Enumeration	Gather Credentials via Compute Metadata	User Account Impersonation	Service Account Creation	Sink Modification	Storage Exfiltration
Service Misconfiguration	Database Enumeration	Gather Credentials from Databases	Adding SSH keys to Compute Metadata	User Account Modification	Pub/Sub Modification	Compute Exfiltration
	Storage Enumeration	Gather GKE Cluster Credentials	Deployment Manager Service Account	Service Account Modification	VPC Service Control Modification	Pub/Sub Data Exfiltration
		Gather local Compute credentials	IAM Modification	Google Group Modification		Database Exfiltration
		Gather Credentials from Storage Buckets	Cloud Functions Service Account	Key Creation		
			Cloud Run Service Account	Key Modification		
			Cloud Scheduler Cron Jobs	VM Instance Startup Script		
				Cloud Shell Startup Script		
				Compute Resource Creation / Execution (GCE, GKE, GCF, GCR)		
				API Key Creation / Modification		
				VPC Firewall / Route Creation / Modification		

Further information on these techniques can be found in the external GCP Attack Matrix Google Doc (linked per cell)

Common Attack Paths in GCP



Common Misconfigurations and Missteps

Services with public-facing resources and the service accounts that support them have the highest potential for misconfiguration and missteps. Common examples include:

- Over-permissive service accounts
- Default VPC firewall rules for compute workloads
- Accidental public exposure of assets
- Publishing sensitive data (credentials, SSNs, etc)
- Lack of controls mitigating malicious data input
- Insufficient logging and visibility

Metadata Server

Every VM instance hosted in GCP stores its metadata on a "metadata server". Your VM automatically has access to the metadata server API without any additional authorization. In addition to custom metadata, there are a default set of metadata entries that are available for VMs running on Compute Engine accessible via curl or functionally equivalent tool:

- Primary/Associated Service Accounts (Token, Email, Scopes)
- Project Metadata
- SSH Keys

Noah McDonald & Wes Guerra