

[Contact Us](#)[Start free](#)

Google Cloud Observability > Logging > Documentation > Guides

AuditLog

[Send feedback](#)

Common audit log format for Google Cloud Platform API operations.

JSON representation

```
{
  "serviceName": string,
  "methodName": string,
  "resourceName": string,
  "resourceLocation": {
    object (ResourceLocation)
  },
  "resourceOriginalState": {
    object
  },
  "numResponseItems": string,
  "status": {
    object (Status)
  },
  "authenticationInfo": {
    object (AuthenticationInfo)
  },
  "authorizationInfo": [
    {
      object (AuthorizationInfo)
    }
  ],
  "policyViolationInfo": {
    object (PolicyViolationInfo)
  },
  "requestMetadata": {
    object (RequestMetadata)
  },
  "request": {
```

```

    object
  },
  "response": {
    object
  },
  "metadata": {
    object
  },
  "serviceData": {
    "@type": string,
    field1: ...,
    ...
  }
}

```

Fields

serviceName

string

The name of the API service performing the operation. For example, "compute.googleapis.com".

methodName

string

The name of the service method or operation. For API calls, this should be the name of the API method. For example,

```

"google.cloud.bigquery.v2.TableService.InsertTable"
"google.logging.v2.ConfigServiceV2.CreateSink"

```

resourceName

string

The resource or collection that is the target of the operation. The name is a scheme-less URI, not including the API service name. For example:

```

"projects/PROJECT_ID/zones/us-central1-a/instances"
"projects/PROJECT_ID/datasets/DATASET_ID"

```

resourceLocation

object ([ResourceLocation](#))

The resource location information.

resourceOriginalState

object ([Struct](#) format)

The resource's original state before mutation. Present only for operations which have successfully modified the targeted resource(s). In general, this field should contain all changed fields, except those that are already been included in `request`, `response`, `metadata` or `serviceData` fields. When the JSON object represented here has a proto equivalent, the proto name will be indicated in the `@type` property.

numResponseItems

string ([int64](#) format)

The number of items returned from a List or Query API method, if applicable.

status

object ([Status](#))

The status of the overall operation.

authenticationInfo

object ([AuthenticationInfo](#))

Authentication information.

authorizationInfo[]

object ([AuthorizationInfo](#))

Authorization information. If there are multiple resources or permissions involved, then there is one `AuthorizationInfo` element for each {resource, permission} tuple.

policyViolationInfo

object ([PolicyViolationInfo](#))

Indicates the policy violations for this request. If the request is denied by the policy, violation information will be logged here.

requestMetadata

object ([RequestMetadata](#))

Metadata about the operation.

request

object ([Struct](#) format)

The operation request. This may not include all request parameters, such as those that are too large, privacy-sensitive, or duplicated elsewhere in the log record. It should never include user-generated data, such as file contents. When the JSON object represented here has a proto equivalent, the proto name will be indicated in the `@type` property.

response

object ([Struct](#) format)

The operation response. This may not include all response elements, such as those that are too large, privacy-sensitive, or duplicated elsewhere in the log record. It should never include user-generated data, such as file contents. When the JSON object represented here has a proto equivalent, the proto name will be indicated in the `@type` property.

metadata

object ([Struct](#) format)

Other service-specific data about the request, response, and other information associated with the current audited event.

serviceData (deprecated)

object



This item is deprecated!

Deprecated. Use the `metadata` field instead. Other service-specific data about the request, response, and other activities.

An object containing fields of an arbitrary type. An additional field `"@type"` contains a URI identifying the type. Example: `{ "id": 1234, "@type": "types.example.com/standard/id" }`.

ResourceLocation

Location information about a resource.

JSON representation
<pre>{ "currentLocations": [string], "originalLocations": [string] }</pre>
Fields
currentLocations[]
string
The locations of a resource after the execution of the operation. Requests to create or delete a location based resource must populate the 'currentLocations' field and not the 'originalLocations' field. For example:
<pre>"europe-west1-a" "us-east1" "nam3"</pre>
originalLocations[]
string
The locations of a resource prior to the execution of the operation. Requests that mutate the resource's location must populate both the 'originalLocations' as well as the 'currentLocations' fields. For example:
<pre>"europe-west1-a" "us-east1"</pre>

```
"nam3"
```

Status

The `Status` type defines a logical error model that is suitable for different programming environments, including REST APIs and RPC APIs. It is used by [gRPC](#). Each `Status` message contains three pieces of data: error code, error message, and error details.

You can find out more about this error model and how to work with it in the [API Design Guide](#).

JSON representation

```
{
  "code": integer,
  "message": string,
  "details": [
    {
      "@type": string,
      field1: ...,
      ...
    }
  ]
}
```

Fields

code

integer

The status code, which should be an enum value of `google.rpc.Code`.

message

string

A developer-facing error message, which should be in English. Any user-facing error message should be localized and sent in the [google.rpc.Status.details](#) field, or localized by the client.

`details[]`

`object`

A list of messages that carry the error details. There is a common set of message types for APIs to use.

An object containing fields of an arbitrary type. An additional field "`@type`" contains a URI identifying the type. Example: { "`id`": 1234, "`@type`": "`types.example.com/standard/id`" }.

AuthenticationInfo

Authentication information for the operation.

JSON representation

```
{
  "principalEmail": string,
  "authoritySelector": string,
  "thirdPartyPrincipal": {
    object
  },
  "serviceAccountKeyName": string,
  "serviceAccountDelegationInfo": [
    {
      object (ServiceAccountDelegationInfo)
    }
  ],
  "principalSubject": string
}
```

Fields

`principalEmail`

`string`

The email address of the authenticated user (or service account on behalf of third party principal) making the request. For third party identity callers, the `principalSubject` field is populated

instead of this field. For privacy reasons, the principal email address is sometimes redacted. For more information, see [Caller identities in audit logs](#).

authoritySelector

string

The authority selector specified by the requestor, if any. It is not guaranteed that the principal was allowed to use this authority.

thirdPartyPrincipal

object ([Struct](#) format)

The third party identification (if any) of the authenticated user making the request. When the JSON object represented here has a proto equivalent, the proto name will be indicated in the `@type` property.

serviceAccountKeyName

string

The name of the service account key used to create or exchange credentials for authenticating the service account making the request. This is a scheme-less URI full resource name. For example:

"//iam.googleapis.com/projects/{PROJECT_ID}/serviceAccounts/{ACCOUNT}/keys/{key}"

serviceAccountDelegationInfo[]

object ([ServiceAccountDelegationInfo](#))

Identity delegation history of an authenticated service account that makes the request. It contains information on the real authorities that try to access GCP resources by delegating on a service account. When multiple authorities present, they are guaranteed to be sorted based on the original ordering of the identity delegation events.

principalSubject

string

String representation of identity of requesting party. Populated for both first and third party identities.

ServiceAccountDelegationInfo

Identity delegation history of an authenticated service account.

JSON representation

```
{
  "principalSubject": string,

  // Union field Authority can be only one of the following:
  "firstPartyPrincipal": {
    object (FirstPartyPrincipal)
  },
  "thirdPartyPrincipal": {
    object (ThirdPartyPrincipal)
  }
  // End of list of possible types for union field Authority.
}
```

Fields

principalSubject

string

A string representing the principalSubject associated with the identity. For most identities, the format will be `principal://iam.googleapis.com/{identity pool name}/subject/{subject}` except for some GKE identities (GKE_WORKLOAD, FREEFORM, GKE_HUB_WORKLOAD) that are still in the legacy format `serviceAccount:{identity pool name}[{subject}]`

Union field Authority. Entity that creates credentials for service account and assumes its identity for authentication. Authority can be only one of the following:

firstPartyPrincipal

object (FirstPartyPrincipal)

First party (Google) identity as the real authority.

thirdPartyPrincipal

object (ThirdPartyPrincipal)

Third party identity as the real authority.

FirstPartyPrincipal

First party identity principal.

JSON representation
<pre>{ "principalEmail": string, "serviceMetadata": { object } }</pre>
Fields
principalEmail
string
The email address of a Google account. .
serviceMetadata
object (Struct format)
Metadata about the service that uses the service account. .

ThirdPartyPrincipal

Third party identity principal.

JSON representation
<pre>{ "thirdPartyClaims": { object } }</pre>
Fields
thirdPartyClaims

object ([Struct](#) format)

Metadata about third party identity.

AuthorizationInfo

Authorization information for the operation.

JSON representation

```
{
  "resource": string,
  "permission": string,
  "granted": boolean,
  "resourceAttributes": {
    object (Resource)
  }
}
```

Fields

resource

string

The resource being accessed, as a REST-style or cloud resource string. For example:

bigquery.googleapis.com/projects/PROJECTID/datasets/DATASETID

or projects/PROJECTID/datasets/DATASETID

permission

string

The required IAM permission.

granted

boolean

Whether or not authorization for resource and permission was granted.

resourceAttributes

object (Resource)

Resource attributes used in IAM condition evaluation. This field contains resource attributes like resource type and resource name.

To get the whole view of the attributes used in IAM condition evaluation, the user must also look into AuditLog.request_metadata.request_attributes.

Resource

This message defines core attributes for a resource. A resource is an addressable (named) entity provided by the destination service. For example, a file stored on a network storage service.

JSON representation

```
{
  "service": string,
  "name": string,
  "type": string,
  "labels": {
    string: string,
    ...
  },
  "uid": string,
  "annotations": {
    string: string,
    ...
  },
  "displayName": string,
  "createTime": string,
  "updateTime": string,
  "deleteTime": string,
  "etag": string,
  "location": string
}
```

Fields

service

string

The name of the service that this resource belongs to, such as `pubsub.googleapis.com`. The service may be different from the DNS hostname that actually serves the request.

name

string

The stable identifier (name) of a resource on the `service`. A resource can be logically identified as `"/{resource.service}/{resource.name}"`. The differences between a resource name and a URI are:

- Resource name is a logical identifier, independent of network protocol and API version. For example, `//pubsub.googleapis.com/projects/123/topics/news-feed`.
- URI often includes protocol and version information, so it can be used directly by applications. For example, `https://pubsub.googleapis.com/v1/projects/123/topics/news-feed`.

See <https://cloud.google.com/apis/design/resourceNames> for details.

type

string

The type of the resource. The syntax is platform-specific because different platforms define their resources differently.

For Google APIs, the type format must be `"{service}/{kind}"`, such as `"pubsub.googleapis.com/Topic"`.

labels

`map (key: string, value: string)`

The labels or tags on the resource, such as AWS resource tags and Kubernetes resource labels.

An object containing a list of `"key": value` pairs. Example: `{ "name": "wrench", "mass": "1.3kg", "count": "3" }`.

uid

string

The unique identifier of the resource. UID is unique in the time and space for this resource within the scope of the service. It is typically generated by the server on successful creation of a resource

and must not be changed. UID is used to uniquely identify resources with resource name reuses. This should be a UUID4.

annotations

map (key: string, value: string)

Annotations is an unstructured key-value map stored with a resource that may be set by external tools to store and retrieve arbitrary metadata. They are not queryable and should be preserved when modifying objects.

More info: <https://kubernetes.io/docs/user-guide/annotations>

An object containing a list of "key": value pairs. Example: { "name": "wrench", "mass": "1.3kg", "count": "3" }.

displayName

string

Mutable. The display name set by clients. Must be <= 63 characters.

createTime

string (Timestamp format)

Output only. The timestamp when the resource was created. This may be either the time creation was initiated or when it was completed.

A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: "2014-10-02T15:01:23Z" and "2014-10-02T15:01:23.045123456Z".

updateTime

string (Timestamp format)

Output only. The timestamp when the resource was last updated. Any change to the resource made by users must refresh this value. Changes to a resource made by the service should refresh this value.

A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: "2014-10-02T15:01:23Z" and "2014-10-02T15:01:23.045123456Z".

deleteTime

string (Timestamp format)

Output only. The timestamp when the resource was deleted. If the resource is not deleted, this must be empty.

A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: "2014-10-02T15:01:23Z" and "2014-10-02T15:01:23.045123456Z".

etag

string

Output only. An opaque value that uniquely identifies a version or generation of a resource. It can be used to confirm that the client and server agree on the ordering of a resource being written.

location

string

Immutable. The location of the resource. The location encoding is specific to the service provider, and new encoding may be introduced as the service evolves.

For Google Cloud products, the encoding is what is used by Google Cloud APIs, such as `us-east1`, `aws-us-east-1`, and `azure-eastus2`. The semantics of `location` is identical to the `cloud.googleapis.com/location` label used by some Google Cloud APIs.

PolicyViolationInfo

Information related to policy violations for this request.

JSON representation

```
{
  "orgPolicyViolationInfo": {
    object (OrgPolicyViolationInfo)
  }
}
```

Fields

orgPolicyViolationInfo

object (OrgPolicyViolationInfo)

Indicates the orgpolicy violations for this resource.

OrgPolicyViolationInfo

Represents OrgPolicy Violation information.

JSON representation
<pre>{ "payload": { object }, "resourceType": string, "resourceTags": { string: string, ... }, "violationInfo": [{ object (ViolationInfo) }] }</pre>
Fields
<p>payload</p> <p>object (Struct format)</p> <p>Optional. Resource payload that is currently in scope and is subjected to orgpolicy conditions. This payload may be the subset of the actual Resource that may come in the request. This payload should not contain any core content.</p>
<p>resourceType</p> <p>string</p> <p>Optional. Resource type that the orgpolicy is checked against. Example: compute.googleapis.com/Instance, store.googleapis.com/bucket</p>
<p>resourceTags</p> <p>map (key: string, value: string)</p> <p>Optional. Tags referenced on the resource at the time of evaluation. These also include the federated tags, if they are supplied in the CheckOrgPolicy or CheckCustomConstraints Requests.</p>

Optional field as of now. These tags are the Cloud tags that are available on the resource during the policy evaluation and will be available as part of the OrgPolicy check response for logging purposes.

An object containing a list of "key": value pairs. Example: { "name": "wrench", "mass": "1.3kg", "count": "3" }.

violationInfo[]

object ([ViolationInfo](#))

Optional. Policy violations

ViolationInfo

Provides information about the Policy violation info for this request.

JSON representation

```
{
  "constraint": string,
  "errorMessage": string,
  "checkedValue": string,
  "policyType": enum (PolicyType)
}
```

Fields

constraint

string

Optional. Constraint name

errorMessage

string

Optional. Error message that policy is indicating.

checkedValue

string

Optional. Value that is being checked for the policy. This could be in encrypted form (if pii sensitive). This field will only be emitted in LIST_POLICY types

policyType

enum ([PolicyType](#))

Optional. Indicates the type of the policy.

RequestMetadata

Metadata about the request.

JSON representation

```
{
  "callerIp": string,
  "callerSuppliedUserAgent": string,
  "callerNetwork": string,
  "requestAttributes": {
    object (Request)
  },
  "destinationAttributes": {
    object (Peer)
  }
}
```

Fields

callerIp

string

The IP address of the caller. For a caller from the internet, this will be the public IPv4 or IPv6 address. For calls made from inside Google's internal production network from one GCP service to another, `callerIp` will be redacted to "private". For a caller from a Compute Engine VM with a external IP address, `callerIp` will be the VM's external IP address. For a caller from a Compute Engine VM without a external IP address, if the VM is in the same organization (or project) as the accessed resource, `callerIp` will be the VM's internal IPv4 address, otherwise `callerIp` will be redacted to "gce-internal-ip". See <https://cloud.google.com/compute/docs/vpc/> for more information.

callerSuppliedUserAgent

string

The user agent of the caller. This information is not authenticated and should be treated accordingly. For example:

- `google-api-python-client/1.4.0`: The request was made by the Google API client for Python.
- `Cloud SDK Command Line Tool apitools-client/1.0 gcloud/0.9.62`: The request was made by the Google Cloud SDK CLI (`gcloud`).
- `AppEngine-Google; (+http://code.google.com/appengine; appid: s~my-project:`
The request was made from the `my-project` App Engine app.

callerNetwork

string

The network of the caller. Set only if the network host project is part of the same GCP organization (or project) as the accessed resource. See <https://cloud.google.com/compute/docs/vpc/> for more information. This is a scheme-less URI full resource name. For example:

```
//compute.googleapis.com/projects/PROJECT_ID/global/networks/NETWORK_ID"
```

requestAttributes

object ([Request](#))

Request attributes used in IAM condition evaluation. This field contains request attributes like request time and access levels associated with the request.

To get the whole view of the attributes used in IAM condition evaluation, the user must also look into `AuditLog.authentication_info.resource_attributes`.

destinationAttributes

object ([Peer](#))

The destination of a network activity, such as accepting a TCP connection. In a multi hop network activity, the destination represents the receiver of the last hop. Only two fields are used in this message, `Peer.port` and `Peer.ip`. These fields are optionally populated by those services utilizing the IAM condition feature.

Request

This message defines attributes for an HTTP request. If the actual request is not an HTTP request, the runtime system should try to map the actual request to an equivalent HTTP request.

JSON representation
<pre>{ "id": string, "method": string, "headers": { string: string, ... }, "path": string, "host": string, "scheme": string, "query": string, "time": string, "size": string, "protocol": string, "reason": string, "auth": { object (Auth) } }</pre>
Fields
<div><div>id</div><div>string</div><div>The unique ID for a request, which can be propagated to downstream systems. The ID should have low probability of collision within a single day for a specific service.</div></div>
<div><div>method</div><div>string</div><div>The HTTP request method, such as GET, POST.</div></div>
<div><div>headers</div></div>

map (key: string, value: string)

The HTTP request headers. If multiple headers share the same key, they must be merged according to the HTTP spec. All header keys must be lowercased, because HTTP header keys are case-insensitive.

An object containing a list of "key": value pairs. Example: { "name": "wrench", "mass": "1.3kg", "count": "3" }.

path

string

The HTTP URL path, excluding the query parameters.

host

string

The HTTP request Host header value.

scheme

string

The HTTP URL scheme, such as http and https.

query

string

The HTTP URL query in the format of name1=value1&name2=value2, as it appears in the first line of the HTTP request. No decoding is performed.

time

string ([Timestamp](#) format)

The timestamp when the destination service receives the last byte of the request.

A timestamp in RFC3339 UTC "Zulu" format, with nanosecond resolution and up to nine fractional digits. Examples: "2014-10-02T15:01:23Z" and "2014-10-02T15:01:23.045123456Z".

size

string ([int64](#) format)

The HTTP request size in bytes. If unknown, it must be -1.

protocol

string

The network protocol used with the request, such as "http/1.1", "spdy/3", "h2", "h2c", "webrtc", "tcp", "udp", "quic". See <https://www.iana.org/assignments/tls-extensiontype-values/tls-extensiontype-values.xhtml#alpn-protocol-ids> for details.

reason

string

A special parameter for request reason. It is used by security systems to associate auditing information with a request.

auth

object (Auth)

The request authentication. May be absent for unauthenticated requests. Derived from the HTTP request `Authorization` header or equivalent.

Auth

This message defines request authentication attributes. Terminology is based on the JSON Web Token (JWT) standard, but the terms also correlate to concepts in other standards.

JSON representation

```
{
  "principal": string,
  "audiences": [
    string
  ],
  "presenter": string,
  "claims": {
    object
  },
  "accessLevels": [
    string
  ]
}
```

Fields

principal

string

The authenticated principal. Reflects the issuer (`iss`) and subject (`sub`) claims within a JWT. The issuer and subject should be / delimited, with / percent-encoded within the subject fragment. For Google accounts, the principal format is: "https://accounts.google.com/{id}"

audiences[]

string

The intended audience(s) for this authentication information. Reflects the audience (`aud`) claim within a JWT. The audience value(s) depends on the `issuer`, but typically include one or more of the following pieces of information:

- The services intended to receive the credential. For example, ["https://pubsub.googleapis.com/", "https://storage.googleapis.com/"].
- A set of service-based scopes. For example, ["https://www.googleapis.com/auth/cloud-platform"].
- The client id of an app, such as the Firebase project id for JWTs from Firebase Auth.

Consult the documentation for the credential issuer to determine the information provided.

presenter

string

The authorized presenter of the credential. Reflects the optional Authorized Presenter (`azp`) claim within a JWT or the OAuth client id. For example, a Google Cloud Platform client id looks as follows: "123456789012.apps.googleusercontent.com".

claims

object ([Struct](#) format)

Structured claims presented with the credential. JWTs include {`key`: `value`} pairs for standard and private claims. The following is a subset of the standard required and optional claims that would typically be presented for a Google-based JWT:

```
{'iss': 'accounts.google.com', 'sub': '113289723416554971153', 'aud': ['123456789012',
'pubsub.googleapis.com'], 'azp': '123456789012.apps.googleusercontent.com', 'email':
'jsmith@example.com', 'iat': 1353601026, 'exp': 1353604926}
```

SAML assertions are similarly specified, but with an identity provider dependent structure.

accessLevels[]

string

A list of access level resource names that allow resources to be accessed by authenticated requester. It is part of Secure GCP processing for the incoming request. An access level string has the format: "{api_service_name}/accessPolicies/{policy_id}/accessLevels/{short_name}"

Example:

"//accesscontextmanager.googleapis.com/accessPolicies/MY_POLICY_ID/accessLevels/MY_LEVEL"

Peer

This message defines attributes for a node that handles a network request. The node can be either a service or an application that sends, forwards, or receives the request. Service peers should fill in **principal** and **labels** as appropriate.

JSON representation

```
{
  "ip": string,
  "port": string,
  "labels": {
    string: string,
    ...
  },
  "principal": string,
  "regionCode": string
}
```

Fields

ip

string

The IP address of the peer.

port

string ([int64](#) format)

The network port of the peer.

labels

map (key: string, value: string)

The labels associated with the peer.

An object containing a list of "key": value pairs. Example: { "name": "wrench", "mass": "1.3kg", "count": "3" }.

principal

string

The identity of this peer. Similar to `Request.auth.principal`, but relative to the peer instead of the request. For example, the identity associated with a load balancer that forwarded the request.

regionCode

string

The CLDR country/region code associated with the above IP address. If the IP address is private, the `regionCode` should reflect the physical location where this peer is running.

Send feedback

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see the [Google Developers Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated 2022-10-03 UTC.

Why Google

Choosing Google Cloud

Trust and security

Modern Infrastructure Cloud

Products and pricing

Google Cloud pricing

Google Workspace pricing

See all products

Solutions

Infrastructure modernization

Databases

Application modernization

Smart analytics

Resources

Google Cloud Affiliate Program

Google Cloud documentation

Google Cloud quickstarts

Engage

Contact sales

Find a Partner


Become a Partner

Events

Podcasts

Multicloud	Artificial Intelligence	Google Cloud Marketplace	Developer Center
Global infrastructure	Security	Learn about cloud computing	Press Corner
Customers and case studies	Productivity & work transformation	Support	Google Cloud on YouTube
Analyst reports	Industry solutions	Code samples	Google Cloud Tech on YouTube
Whitepapers	DevOps solutions	Cloud Architecture Center	Follow on X
Blog	Small business solutions	Training	Join User Research
	See all solutions	Certifications	We're hiring. Join Google Cloud!
		Google for Developers	Google Cloud Community
		Google Cloud for Startups	
		System status	
		Release Notes	

[About Google](#) | [Privacy](#) | [Site terms](#) | [Google Cloud terms](#)

 Our third decade of climate action: join us

Sign up for the Google Cloud newsletter

Subscribe

