This page explains how to use audit logging in your Google Kubernetes Engine clusters.

All Kubernetes-powered clusters have Kubernetes Audit Logging (https://kubernetes.io/docs/tasks/debug-application-cluster/audit/), which keeps a chronological record of calls that have been made to the Kubernetes API server. Kubernetes audit log entries are useful for investigating suspicious API requests, for collecting statistics, or for creating monitoring alerts for unwanted API calls.

GKE clusters integrate Kubernetes Audit Logging with Cloud Audit Logs (/logging/docs/audit/) and Stackdriver Logging (/logging/docs). You can see Kubernetes audit log entries in your Google Cloud project.

In addition to entries written by Kubernetes, your project's audit logs have entries written by Kubernetes Engine.

Audit Logging GA is available in GKE 1.11.4 and later.

Before you start, make sure you have performed the following tasks:

- Ensure that you have enabled the Google Kubernetes Engine API.

  Enable Google Kubernetes Engine API (https://console.cloud.google.com/apis/library/container.googleapis.com?q=kubernetes%20engine)

- Ensure that you have installed the Cloud SDK (/sdk/downloads).

Set up default `gcloud` settings using one of the following methods:

- Using `gcloud init`, if you want to be walked through setting defaults.
- Using `gcloud config`, to individually set your project ID, zone, and region.

You need to have a Kubernetes Engine cluster in your project. You can use an existing cluster, or you can create a new one (/kubernetes-engine/docs/how-to/creating-a-container-cluster) for the exercises in this topic. If you choose to use an existing cluster, make sure the cluster has had some recent activity. For example, if you haven't created a Deployment recently, you could create a Deployment now by entering this command:

Read about Kubernetes Audit Logging (https://kubernetes.io/docs/tasks/debug-application-cluster/audit/).

Your Cloud project has these audit logs:

- Admin Activity log
- Data Access log

Admin Activity logging is enabled by default and has no extra cost.

Data Access logging is disabled by default, and enabling it can result in extra billing. To learn more about enabling Data Access logging, and the associated costs, see Configuring Data Access Logs (/logging/docs/audit/configure-data-access).

Kubernetes Engine does not support Access Transparency logging (/logging/docs/audit/access-transparency-overview).

Various Google Cloud Platform services (/logging/docs/audit/services) write entries to your project's logs. The Kubernetes service also writes entries to your project's audit logs. For Kubernetes Engine clusters, log entries written by these services are the most relevant:

Service	Display name
k8s.io	Kubernetes
container.googleapis.com	Kubernetes Engine

In the Cloud Console, the Logs page (https://console.cloud.google.com/logs/viewer) has two filtering interfaces: basic and advanced. For information about the two filtering interfaces, see Logs Viewer filter interfaces (/logging/docs/view/overview#the_user_interfaces).

Each log entry in your Admin Activity log applies to a certain type of resource. These are the resource types that are the most relevant to Kubernetes clusters:

Resource type	Display name
k8s_cluster	Kubernetes Cluster
gke_cluster	GKE Cluster Operations

Log entries written by the Kubernetes API server apply to the `k8s_cluster` resource type. These log entries describe operations on Kubernetes resources in your cluster, for example, Pods, Deployments, and Secrets.

Log entries written by the Kubernetes Engine API server apply to the `gke_cluster` resource. These log entries describe operations like cluster creation and deletion.

Here are some examples of filters that you can try in the Cloud Console (https://console.cloud.google.com/logs/viewer). In each case, replace ***[PROJECT_ID]*** with your project ID.

Find changes to Role Based Access Control, excluding automated system changes.

```
          [PROJECT_ID]
          [PROJECT_ID]
```

```
          [PROJECT_ID]
```

You can use similar queries to find changes to `clusterroles` and `clusterrolebindings`.

Find certificate signing requests.

```
          [PROJECT_ID]
```

Find unauthenticated web requests.

```
          [PROJECT_ID]
```

Find kubelet bootstrap identity calls.

```
          [PROJECT_ID]
```

Find node authenticated requests.

```
          [PROJECT_ID]
```

Find calls outside an IP range.

```
          [PROJECT_ID]

                                        [IP Prefix]
```

Find entries in your Admin Activity log that apply to the `k8s_cluster` resource type and describe creating a Deployment.

```
            [PROJECT_ID]
```

Find entries in your Admin Activity log that apply to the `k8s_cluster` resource type and have a `principalEmail` value of `system:anonymous`. These entries probably represent failed attempts to authenticate.

```
            [PROJECT_ID]
```

Find entries in your Admin Activity log that apply to the `gke_cluster` resource type and describe cluster creation:

```
            [PROJECT_ID]
```

Find entries in your Admin Activity log that apply to the `gke_cluster` resource type and have a `severity` value of `ERROR`:

```
            [PROJECT_ID]
```

Find entries in your Admin Activity log that apply to the `k8s_cluster` resource type and describe a write request to a Secret:

```
            [PROJECT_ID]
```

Find entries in your Admin Activity log that apply to the `k8s_cluster` resource type and describe a Pod request from a particular user:

```
            [PROJECT_ID]
```

For more information about how to construct filters, see Advanced Logs Filters (/logging/docs/view/advanced-filters).

Every log entry is an object of type LogEntry (/logging/docs/reference/v2/rest/v2/LogEntry). For more information, go to Understanding audit logs (/logging/docs/audit/understanding-audit-logs).

Get the Cloud Identity and Access Management (Cloud IAM) policy for your project:

where ***[PROJECT_ID]*** is your project ID.

Open `my-policy.yaml` to view your Cloud IAM policy. Your policy probably contains a `bindings` object similar to this:

In `my-policy.yaml`, create an `auditConfigs` object, or add to your existing `auditConfigs` object, so that `ADMIN_READ`, `DATA_WRITE`, AND `DATA_READ` are listed under `auditLogConfigs`.

Do not change the value of `etag`.

Save your updated file as `my-policy-2.yaml`.

n: Do not change any part of your policy that is not related to audit logging. You are editing a policy object that contains critical information about w
cess your project or organization. Accidentally altering that information could make your project or organization unusable.

Set the Cloud IAM policy for your project:

where ***[PROJECT_ID]*** is your project ID.

Log entries are held in Stackdriver Logging for a limited time known as the retention period (/logging/quotas#logs_retention_periods). After that, the entries are deleted.

If you want to keep your log entries longer, you can export (/logging/docs/export/) them to a Google service like Cloud Storage, BigQuery, or Cloud Pub/Sub.

You can use Stackdriver Monitoring to set up metrics (/logging/docs/logs-based-metrics/) based on your log entries. And you can use log-based metrics to set up charts and alerts (/logging/docs/logs-based-metrics/charts-and-alerts).

The Kubernetes audit policy determines which log entries are exported by the Kubernetes API server. The Kubernetes Engine audit policy determines which entries go to your Admin Activity log and which entries go to your Data Access log.

For more information about audit policies in Kubernetes Engine, see Kubernetes Engine Audit Policy (/kubernetes-engine/docs/concepts/audit-policy).

- Kubernetes Audit Logging (https://kubernetes.io/docs/tasks/debug-application-cluster/audit/)
- Kubernetes Engine Audit Policy (/kubernetes-engine/docs/concepts/audit-policy)
- Kubernetes Engine Security Overview (/kubernetes-engine/docs/concepts/security-overview)
- [Cloud Audit Logging]