Best Practices for Enterprise Organizations

Learn key concepts and best practices for getting the most out of Google Cloud Platform in an enterprise setting. This guide covers non-functional (https://wikipedia.org/wiki/Non-functional_requirement) topics relevant to enterprise-sized organizations.

The guide is organized into the following sections:

- Projects and access (#projects-and-access)
- Authentication and identity (#authentication-and-identity)
- Networking and security (#networking-and-security)
- Logging, monitoring, and auditing (#logging-monitoring-auditing)
- Support and training (#support-and-training)
- Billing and cost attribution (#billing-and-cost)
- Risk management (#risk-management)

Note: For the sake of completeness, some concepts are discussed in multiple sections.

Projects and access

Projects are the core organizational component of Google Cloud Platform. Projects provide an abstract grouping that you can use to associate resources with a particular department or functional team. All Cloud Platform resources belong to a project. You can manage projects by using the Google Cloud Platform Console (https://console.cloud.google.com/project), as well as the Resource Manager API (https://cloud.google.com/resource-manager/reference/rest/). The Cloud IAM API (https://cloud.google.com/iam/docs/managing-policies#access_control_via_api) includes a set of methods to manage project permissions through the Resource Manager API.

Use projects to control access to resources

Projects provide an isolation boundary, except where interconnects are explicitly granted, between the Cloud Platform resources used by your organization. Users and groups can be

granted different roles, such as **`viewer`**, **`editor`**, and **`owner`**, for different projects. To assign roles, you can use the IAM & Admin page (https://console.cloud.google.com/iam-admin/iam) in the Cloud Platform Console or the Cloud IAM API (https://cloud.google.com/iam/docs/managing-policies#access_control_via_api). This API currently doesn't allow you to create custom roles and assign permissions to those roles.

Further, you can delegate control over who has access to a particular project. Users granted the **`owner`** role can grant and revoke access for users, groups, and service accounts (https://developers.google.com/identity/protocols/OAuth2ServiceAccount).

Any Google account can be granted access to a project

You can grant any Google account, such as a Gmail account, access to a project. Any Google account can also be given access by virtue of belonging to a group that has access to the project. This feature can be helpful because it enables you to give access quickly to external parties, such as contractors. However, your organization might not want to allow this flexibility, given your security policies. You can manage this risk with the Cloud IAM API. You can use the API to build monitoring functionality that watches for off-policy assignments, and then raises an alert or revokes them automatically.

Projects are identified by universally unique identifiers

Each project has a universally unique project ID, which is a short string of lowercase letters, digits, and dashes, and a *project number*. When creating a project, you can specify the project ID. Google assigns the project number automatically. After they have been created, the project ID and project number cannot be changed.

We recommend that you spend some time planning your project IDs for manageability. A typical project ID naming convention might use the following pattern:

```
[company tag]-[group tag]-[system name]-[environment (dev, test, uat, stag
```

For example, the development environment for the human resources department's compensation system might be named **`acmeco-hr-comp-dev`**.

You can provide a human-readable *project name*; these names don't have to be globally unique and they can be edited. Project IDs can be from 6 to 30 characters long, while project names can be from 4 to 30 characters long.

Projects facilitate accurate costing

Google organizes your bill, including the exportable CSV or JSON version, by project. The project forms the top-level grouping of your invoice. There are further breakdowns within that grouping by SKU, but the top-level project grouping provides excellent visibility into the overall cost for a group of compute and networking resources.

To understand the contents of the exportable billing file, see Export billing data (https://support.google.com/cloud/answer/6293835).

Cross-project access is possible but must be explicitly allowed

Like users and groups, service accounts can be granted access to multiple projects. This cross-project access makes it possible for processes in one project to directly access resources within another project.

When a service account from project A accesses a resource in project B, the costs are attributed to the project that owns the resources: project B.

The exception to this rule is when you run queries in Google BigQuery. If from within project B, a user or service account queries BigQuery data stored in project A, the query costs are charged to the project that originated the query: project B. However, the data storage costs remain with project A. This approach makes it easier to understand how different groups are utilizing BigQuery for data analysis.

Each project belongs to a specific billing account

Billing accounts are an organization-wide resource that can only be edited by billing administrators. Billing administrators are managed by using the Billing page (https://console.cloud.google.com/billing) in the Cloud Platform Console. Each project is associated with exactly one billing account. Projects can be associated with a billing account only by a user who is both a project owner and billing administrator. Support charges and invoices are associated with a single billing account.

Projects are not based on geography or equivalent to zones

A project is a logical grouping of resources that does not correspond to a particular geographic region (https://cloud.google.com/docs/geography-and-regions). Similarly, a project does not

represent an availability zone. Resources within a given project can exist in multiple geographic regions and multiple zones.

For example, project A can contain resources in Central USA, Europe, and Asia, each with multiple zones, while project B might contain resources only in Europe, also with multiple zones.

You don't need to set a geographic region or zone when creating a project. The only exception to this rule is in Google App Engine. App Engine apps always have a geographic location. When creating a new project, you can change the App Engine location from the default location. Note that after creating a project, certain resources can be pinned to specific geographic locations; this concept is explained in greater detail in upcoming sections.

Authentication and identity

This section provides best practices for creating and managing authentication and identities in Google Cloud Platform.

Use corporate login credentials

Google Cloud Platform uses Google Accounts (https://support.google.com/work/android/answer/6371476) for authentication and access management. Google recommends using fully managed corporate Google accounts for increased visibility, auditing, and control over access to Cloud Platform resources.

Cloud Identity (https://support.google.com/a/answer/7319251) provides free, managed Google Accounts you can use with Google services including Cloud Platform. Using Cloud Identity accounts for each of your users, you can manage all users across your entire domain from the Google Admin console.

If you're a G Suite administrator, you can manage all of your users and settings through the G Suite Admin Console. By default, all new users are assigned a G Suite license. If you have a subset of developers who don't require G Suite licenses, you can add Cloud Identity accounts instead.

For more information, see Get started with Cloud Identity (https://support.google.com/a/topic/7386474).

Provision users to Google's directory

The G Suite global Directory provides a structured data store of users and groups used for authentication and authorization. The global Directory is available to both Cloud Platform and G Suite resources and can be provisioned by a number of means. Provisioned users can take advantage of rich authentication features including single sign-on (SSO), OAuth, and two-factor verification.

You can provision users automatically using one of the following tools and services:

- Google Cloud Directory Sync (GCDS) (https://support.google.com/a/answer/106368)
- Google Admin SDK (https://developers.google.com/admin-sdk/)
- A third-party connector

GCDS is a connector that can provision users and groups on your behalf for both Cloud Platform and G Suite. Using GCDS, you can automate the addition, modification, and deletion of users, groups, and non-employee contacts. You can synchronize the data from your LDAP directory server to your Cloud Platform domain by using LDAP queries. This synchronization is one-way: the data in your LDAP directory server is never modified.

GCDS runs inside your network on a machine that you control. It connects to your LDAP server inside your network through standard LDAP or secure LDAP plus Secure Sockets Layer (SSL). This connection occurs on any port you specify, but defaults to standard LDAP ports. GCDS connects to the G Suite domain through the Internet through HTTPS on port 443. This connection can also run through a proxy host in your network.

If you prefer to build your own solution to provision users and groups, you can use the Google Admin SDK (https://developers.google.com/admin-sdk/). The Admin SDK provides methods for managing Cloud Platform users and their associations with groups and organizations. The SDK supports HTTPS REST endpoints, and also provides Python, Java, .NET, and other client libraries.

In addition to these native tools and services, many leading identity management vendors provide connectors for the G Suite global Directory. These connectors typically provision G Suite users and their associations with groups and organizations by using the Google Admin SDK's Directory API (https://developers.google.com/admin-sdk/directory/). Because Cloud Platform and G Suite share a common directory infrastructure, these connectors are applicable to both Cloud Platform and G Suite.

To manually provision users for testing or other purposes, Cloud Platform administrators can provision users and their associations with groups and organizations manually by using the G Suite Admin Console (https://support.google.com/a/answer/179832).

To learn more about the G Suite global Directory, see Manage the global Directory (https://support.google.com/a/answer/1628009).

Resolve conflicting accounts during user provisioning

If members of your domain have used their corporate email addresses to create a personal Google Account—for example, to sign up for a Google service such as YouTube or Blogger—these accounts will cause conflicts when you add them to G Suite or Cloud Platform. The authorizations for the existing accounts must be manually revoked and transferred to the new user accounts.

For guidance about how to avoid and resolve conflicting account issues, see Resolve conflicting accounts (https://support.google.com/a/answer/185186).

Define domain administration roles

When a project is associated with a domain, a new role called *Super Admin* is created. The Super Admin role applies to the domain itself. Use one of the predefined roles or create a custom role and define its management scope:

- Users: organization, which is a subset of users, or global.
- Privileges: such as group management or security management.

These roles can be managed through the Admin Console, or through the Roles API (https://developers.google.com/admin-sdk/directory/v1/guides/manage-roles). For more information about domain administrator roles, see About administrator roles (https://support.google.com/a/answer/33325).

Implement SSO with SAML exchange

Cloud Platform supports SAML 2.0-based SSO, which provides seamless SSO against Cloud Platform Console, web- and command-line-based SSH, and OAuth authorization prompts. Cloud Platform's command-line interface tools, such as **`gcloud`**, **`gsutil`**, and **`bq`**, use SAML 2.0-based SSO for browser-based authentication as well.

If the user accesses any of these tools without having previously authenticated against the SSO provider, either directly or by accessing another application, the user is prompted for their username but not their password. This step helps the Google authentication infrastructure determine the domain for which the user wants to be authenticated.

For information about setting up Google SSO, see Set up Single Sign-On for G Suite accounts (https://support.google.com/a/answer/60224). This guide applies to both Cloud Platform and G Suite, because both products share a common directory, auth, and SSO infrastructure.

Consider using Google as an identity provider for other services

You can move to a 100% cloud-based authentication solution by authenticating your own applications with Google's OpenID Connect, and by using Google as a SAML 2.0 identity provider to authenticate commercial, off-the-shelf applications.

Google and third parties provide libraries that you can use to take care of many of the implementation details of using OpenID Connect to authenticate users. You can leverage Google as your SAML 2.0 identity provider rather than using a third-party directory, such as Microsoft Active Directory. G Suite supports more than 15 popular SaaS providers (https://gsuite.google.com/partner/recommended/). Recommended G Suite identity partners include Ping (https://www.pingidentity.com/) and Okta (https://www.okta.com/). You can also add new custom SAML app integrations (https://support.google.com/a/answer/6087519).

Using Google as your identity provider allows you to leverage Google's rich authentication infrastructure, including two-factor authentication and FIDO U2F Security Key devices (https://support.google.com/accounts/answer/6103523).

Assign project roles to groups of users

You can use Google Groups (https://support.google.com/a/answer/33329) to manage project authorization across a large organization. If you're already familiar with Role-Based Access Control (RBAC) (https://developers.google.com/admin-sdk/directory/v1/guides/manage-roles), you can think of Google Groups as being analogous to RBAC roles, while project roles in Cloud Platform are analogous to RBAC permissions. You can assign project roles to groups in a similar way that you assign them to users, and manage group membership as previously described.

Authorize server-to-server interactions with service accounts

Like users, service accounts can be granted authorization to specific resources, by using similar techniques. Examples include granting an application the ability to read and write to Google Cloud Storage buckets or access particular datasets within BigQuery. Service accounts are useful for running automated processes because they are authenticated with a private key, not with a password.

Delegate application authorization with OAuth2

Cloud Platform APIs support OAuth 2.0 (https://developers.google.com/identity/protocols/OAuth2), and scopes provide granular authorization over the methods that are supported. Cloud Platform supports both service-account and user-account OAuth, also called three-legged OAuth.

OAuth 2.0 is typically used to authorize web applications. It also has an installed applications mode (https://developers.google.com/identity/protocols/OAuth2InstalledApp) which supports desktop applications.

Cloud Platform supports Application Default Credentials (https://developers.google.com/identity/protocols/application-default-credentials), which are well suited to non-user-dependent access to APIs, such as when everyone has access to the same data. For operations from a command line terminal, you can authorize the `gcloud` command-line tool (https://cloud.google.com/sdk/gcloud/) to access APIs on behalf of your user account or a service account. `gsutil` (https://cloud.google.com/storage/docs/gsutil) can also use these credentials.

Verify instance identity before transmitting data

Instances can generate a JSON Web Token (JWT) that includes details about the instance as well as Google's RS256 signature. Your applications verify the signature against Google's public Oauth2 certificates (https://www.googleapis.com/oauth2/v1/certs) to confirm the identity of the instance with which they have established a connection.

Read Verifying the Identity of Instances (https://cloud.google.com/compute/docs/instances/verifying-instance-identity) to learn how to request and verify signed instance tokens.

Networking and security

Use projects to fully isolate resources

Google uses software-defined networking that enables you to subject every packet to security checks, thereby enabling complete isolation of Cloud Platform projects. You can read more about software-defined networking in Google's data centers on the Cloud Platform blog (https://googlecloudplatform.blogspot.com/2014/04/enter-andromeda-zone-google-cloud-platforms-latest-networking-stack.html)

.

You can read more about security on the Google Cloud Platform Security page (https://cloud.google.com/security/).

Use networks within projects to isolate groups of VM instances

Each project supports up to five isolated networks. Each network constitutes a global IP address space. This means that you can create resources, such as Google Compute Engine virtual machine (VM) *instances*, in multiple geographic regions, and the resources will share the same IP address space because they are on the same virtual network. Note that, despite the flat network address space, you still incur normal egress charges when you leave your zone. For more information, see Network pricing (https://cloud.google.com/compute/pricing#network).

You can request a quota increase to support up to 15 isolated networks in each project.

Leverage network-local DNS

Each network supports a local domain-name server (DNS) that enables VM instances to find each other quickly and easily, by name. Similar to the global address space, this local DNS operates over the entire network, regardless of region or zone of the VM instance using the service.

A virtual firewall and routes control all network access

Cloud Platform utilizes a software-defined network that provides lots of flexibility in virtual appliances. You can create firewall rules to control incoming traffic to instances and load balancers, both from the public Internet as well as from other instances on the same network. You can apply firewall rules to specific instance tags. For example, you can create a firewall

rule that applies dynamically to a set of instances marked with the same tag, which makes autoscaling of clusters easier.

Routes are similarly flexible. You define the routes for the network that define how the VM instances direct traffic to other instances on the same network, as well as outside resources. Routes can also be applied to specific instance tags, enabling you to set up rules that can be dynamically applied to instances as they come and go.

Use `iptables` and routes to limit or filter egress traffic

While the Cloud Platform firewall rules can control ingress traffic to your VM instances, network routes can control egress from VMs to IP address ranges. Examples include routing all external traffic through a NAT gateway, all traffic to corporate IP ranges through a VPN gateway, or denying access to IP ranges by routing to a non-existent IP. If you require control over the egress traffic from the VM instances to specific ports, such as filtering all traffic through port 80, configure `iptables` (https://help.ubuntu.com/community/IptablesHowTo) or another host-based filtering mechanism on the instance.

VM instances have an internal IP address and can have a public IP address

Every VM instance is assigned an internal address that fits the network address space in which it was created. This address can be used for traffic between instances on the same network. Optionally, you can attach an external, public IP address to a VM. This external address can either be *ephemeral* or, for a fee, *static*. Note that you are charged for a reserved static IP address when the VM instance isn't running, and also when the IP is not associated with any VM instance.

Your standard quota includes availability of a small number of external IP addresses: 23 ephemeral and 7 static, per region. With a quota increase, you can increase the number of external ephemeral IP addresses to 500 per region. If you require more external IP addresses than the quotas provide, contact Cloud support (https://enterprise.google.com/supportcenter/managecases) or your technical account manager to discuss your requirements.

VM instances require a public IP address to reach Cloud Platform services

To reach Internet resources, including Cloud Platform APIs and other services, each of your VM instances need an external IP address. While this practice might raise some concerns, we recommend that you limit the incoming traffic to these VM instances by using firewall rules.

If your security policy requires truly internal VM instances, you will need to set up a NAT proxy manually on your network and a corresponding route so that the internal instances can reach the Internet. It is important to note that you cannot connect to a fully internal VM instance directly by using SSH. To connect to such internal machines, you must set up a *bastion* instance that has an external IP address and then tunnel through it. For more information, see Bastion hosts (https://cloud.google.com/solutions/connecting-securely#bastion).

Put certain resources in regions and zones for lower latency and disaster recovery

A *region* represents a geographic region or, more specifically, the location of a data center campus. Within a region, there are multiple *zones* which are availability zones with respect to the underlying infrastructure. Certain Cloud Platform resources are global, but other resources can be pinned to specific regions or, in some cases, specific zones. VM Instances, persistent disks, Cloud Storage buckets, App Engine applications, Cloud Bigtable, Cloud Dataproc, BigQuery datasets, Cloud VPN, and some other resources can all be created within a specific geographic region.

Leverage regions and zones (https://cloud.google.com/docs/geography-and-regions) to achieve appropriate service redundancy or to minimize latency by considering geography.

Use Cloud VPN to securely connect remote networks

Google Cloud VPN (https://cloud.google.com/vpn/docs/) is a flexible tool that can be used to connect remote networks securely, including between Cloud Platform and an on-premises network, two networks in different projects, or two networks in the same project. Cloud VPN tunnels are billed at a static monthly rate plus standard egress charges. Note that connecting two networks on the same project still incurs standard egress charges.

Using instance tags on routes, you can dynamically control which VM instances can send traffic down the VPN.

Scan for common website vulnerabilities

Detect common website vulnerabilities with Google Cloud Security Scanner (https://cloud.google.com/security-scanner/). This tool scans your App Engine applications for cross-site scripting and mixed- content issues.

As with all dynamic vulnerability scanners, a clean scan does not ensure that your site is secure. You must still perform a manual security review.

Cloud Router can add routes dynamically to Cloud VPN

Cloud VPN, though flexible, still requires that IP address ranges be statically added and removed to and from tunnels. Google Cloud Router (https://cloud.google.com/router/docs/) alleviates this limitation leveraging Border Gateway Protocol (BGP) to dynamically update the tunnels.

A network can be divided into subnetworks

Subnetworks (https://cloud.google.com/compute/docs/subnetworks) on Compute Engine enable you to control the address space in which VM instances are created, while maintaining the ability to route between them. Subnetworks can be any non-public IP range, which means they don't need to belong to a common, parent network. Subnetworks within a particular network grouping can reach one another.

You can achieve further isolation between subnetworks by using firewall rules and routes.

Logging, monitoring, and auditing

To recognize threats and mitigate risks, monitor access and usage of your account and system resources. Cloud Platform has a very rich set of monitoring and alerting tools that can be an important part of your overall solution.

Use Cloud Logging as a centralized location for logs

Google Cloud Logging (https://cloud.google.com/logging/docs/) provides a generalized, centralized location for logs. Logs from VM instances, App Engine and managed services automatically appear in this centralized logging area. You can also send other arbitrary logs to Cloud

Logging. Use this feature to collect all of your system and application logs together in one place, for easier analysis and monitoring.

Monitor access of system resources

To monitor how your system resources are accessed, monitor the following items:

- Compute Engine resource modifications such as creating and deleting instances and disks, changing firewall rules, and configuring load balancing and autoscaling by using the RegionOperations API (https://cloud.google.com/compute/docs/reference/latest/regionOperations).
- Requests for Compute Engine information such as **`instance/get`** invocations by using Activity Logs (https://cloud.google.com/compute/docs/activity-logs).
- Compute Engine application logs and authentication attempts by using the Google Cloud Logging Agent (https://cloud.google.com/logging/docs/agent/).
- VPN gateway-to-gateway traffic, including source and IP addresses, by using the private network log (https://cloud.google.com/vpn/docs/how-to/logging).
- BigQuery queries and table, dataset, and view operations by using the BigQuery API (https://cloud.google.com/bigquery/docs/reference/v2/jobs/list).
- Access of Cloud Storage objects by using the Cloud Storage access logs (https://cloud.google.com/storage/docs/access-logs).
- App Engine deployments, version promotions, cron, index, queue or other configuration changes by using the App Engine Admin logs.
- Cloud SQL operations by using the Cloud SQL operations.list method (https://cloud.google.com/sql/docs/admin-api/v1beta4/operations/list).
- Google Cloud Deployment Manager operations by using the Deployment Manager API (https://cloud.google.com/deployment-manager/latest/operations/list).

Monitor account access

Monitor all attempts to access your domain account by using the Reports API (https://developers.google.com/admin-sdk/reports/v1/guides/manage-audit-login) and Login audit log in the Admin console (https://support.google.com/a/answer/4580120?ref_topic=6046029).

Monitor administrative actions

Monitor changes to project roles by using the IAM API (https://cloud.google.com/iam/docs/managing-policies#access_control_via_api). Monitor all operations performed in the domain admin console with the Admin console audit log (https://support.google.com/a/answer/4579579?ref_topic=6046029).

Export logs to BigQuery for analysis and long term storage

BigQuery is an excellent tool for rapidly analyzing huge amounts of data. BigQuery is also a very cost-effective way to store data for analysis. You can configure Cloud Logging to export logs directly to BigQuery, which gives you a powerful tool to perform detailed analysis to identify trends in your logs.

Prevent unwanted changes to logs

Logs are stored to Cloud Storage in the originating project. By default, project owners and editors have ownership permissions for all Cloud Storage buckets in the project and objects under the bucket's hierarchical permissions model.

To minimize the risk of inadvertent or malicious changes to your logs, apply the following principles.

Least privilege

Grant the least-broad permissions that are required to do the job. Restrict the usage of `owner` role for projects and log-buckets. The intent of the `owner` role is to manage team membership, authorization, and so on.

Team members with the `editor` role still can deploy applications and modify or configure their resources.

You could provide managed access to buckets and objects to a broader population through a custom application that uses the Cloud Storage API (https://cloud.google.com/storage/docs/json_api/v1/) with a dedicated service account, and then add auditing in the application.

Non-repudiation

Cloud Storage automatically encrypts all data before it is written to disk. You can provide some additional assurance of non-repudiation by implementing object versioning (https://cloud.google.com/storage/docs/object-versioning) on the log-buckets. When an object is overwritten or deleted in a bucket, a copy of the object is automatically saved with generation properties that identify it. Unfortunately, this feature can't protect against a project owner deleting the archived object or disabling the versioning.

Separation of duties

You can provide some additional assurance of separation of duties. For example, you might require two people to inspect and sign off on the logs. You can copy the log-buckets to a project that has a different owner by using gsutil cp (https://cloud.google.com/storage/docs/gsutil/commands/cp#copying-in-the-cloud-and-metadata-preservation) as part of a frequent cron job, or if the amount of data copied will be greater than 10TB of log data at a time, by using the Cloud Storage Transfer Service (https://cloud.google.com/storage/transfer/). This approach can't protect against a project owner who deletes the original bucket before the copy occurs or who disables the original logging.

This approach does cloud-based copy without copying to your local machine, so it is fast and efficient. There are no network ingress or egress charges for copies within a region (https://cloud.google.com/storage/pricing#network-pricing), though there will be operations charges (https://cloud.google.com/storage/pricing#operations-pricing) for the copying.

You can reduce the storage costs of log duplication by copying the objects to Nearline Storage (https://cloud.google.com/storage/docs/storage-classes#nearline), implementing object lifecycle management (https://cloud.google.com/storage/docs/lifecycle) on the originating bucket objects and removing them after you have verified that they've been replicated to the backup project, perhaps weekly.

Use Stackdriver Monitoring to monitor your resources and provide alerts

You can use Stackdriver Monitoring (https://cloud.google.com/monitoring/) to monitor various resources such as VM instances, App Engine, and Cloud Pub/Sub. Many of these monitoring integrations are provided automatically and you just need to configure thresholds for alerts.

Cloud Platform also provides the ability to detect particular log entries and build custom metrics around these items.

Unify all your logging and monitoring needs by using Cloud Platform

Stackdriver Monitoring enables you to ship monitoring metrics from VMs located outside of Cloud Platform. You can use an installable agent to report and alert over all your resources in a single dashboard, including resources that are on- premises and in other clouds.

Use the Activity page to see activity in your organization's projects

The Activity page (https://console.cloud.google.com/home/activity) in the Cloud Platform Console provides an organization-wide stream of activities in all your projects. It can be filtered by one or more projects, activity types, and date range to quickly zero in on changes or adjustments. A core set of products support the activity stream today, and more will be added over time.

Support and training

Production applications should have at least Gold support

A support package (https://cloud.google.com/support/) is purchased for your entire organization. While there is a spectrum of support packages available, we strongly recommend the Gold and Platinum packages to deal with critical issues in a timely fashion for your projects in production.

The community is an excellent source of support

Google employees actively support and encourage the community of users. For a list of common community locations where you can find detailed information about Cloud Platform, see Join a Google Cloud Platform community (https://support.google.com/cloud/answer/3466163).

Introductory and advanced training will help avoid common pitfalls

Google provides qualification training (https://cloud.google.com/training/) for Cloud Platform, ranging from high- level introductory overviews to detailed deep-dive sessions on specific technologies. You can find a class (https://cloud.google.com/training/courses) on the Google Cloud Platform website.

We recommend that you have at least key people in your organization become qualified on Cloud Platform. The **CPO200: Google Cloud Platform for Systems Operations Professionals** course is particularly helpful for people deploying and managing Compute Engine applications.

Build internal centers of excellence for products

Cloud Platform provides an extensive set of products, services, and APIs that can be combined in creative and unexpected ways. Google continues to invest in these products, and new features are continually being rolled out. It can be valuable to capture your organization's information, experience, and patterns in an internal knowledge base, such as in a wiki, Google Site, or intranet site.

In your wiki, include a list of product experts who can advise new users and help them use particular Cloud Platform products according to your organization's best practices.

Establish an internal support clearinghouse

There is a maximum number of people that you can authorize to submit support tickets on behalf of your organization. We recommend that you set up an internal support clearinghouse or triage desk. This approach helps to avoid ticket duplication and miscommunication, and keeps your communication with Google Cloud Support as clear as possible.

Leverage Google Partners

On occasion, you might need to supplement your organization's expertise and capacity with outside help. Google has built a rich partner ecosystem that can help fill these gaps. For more information about partners, see Google Cloud Platform Partners (https://cloud.google.com/partners/).

Keep up with the latest Cloud Platform news and announcements

Stay up to date on the latest news, announcements, and customer stories by subscribing to the Google Cloud Platform blog (https://googlecloudplatform.blogspot.com/).

Billing and cost attribution

Your monthly bill breaks down costs by project, then by resource type

Your monthly bill is delivered by email to your billing administrators. The bill itself contains a lot of detail. Recall that the bill breaks down resource usage first by project, then by resource type. Careful naming of projects makes it very easy to see which teams or products are consuming resources. You can use this information to facilitate easy chargebacks to those departments within your organization.

Use billing export daily to get a machine-readable version of your bill

You can enable the billing export feature to publish the equivalent of the detailed invoice line items on a daily basis. This file can be formatted as either JSON or CSV and published to a Cloud Storage bucket. This raw data can facilitate daily reports on your Cloud Platform usage and costs. Enable billing export in the Billing Account administration (https://console.cloud.google.com/billing) area of the Cloud Platform Console.

Use project labels to further categorize projects in billing export

In the Cloud Platform Console (https://console.cloud.google.com/project), on the **Projects** page, you can add custom labels to projects as key-value pairs. These labels appear in the billing export file. Apply these labels to enable further categorization of projects.

Credits are applied to a billing account, not to a specific project

If you receive credits, such as for a refund or for promotional reasons, the credits are applied to the billing account, not to a specific project. This means that all projects draw against the pool of credits until they are consumed, which can make accurate costing a bit more challenging. However, credits are generally applied infrequently enough that they shouldn't have a major impact on your costing.

Project spend can be capped

Compute Engine costs are controlled through the quota system. Each project has a specific quota that represents the maximum amount of resources that can be consumed by that project. To adjust these quotas, complete the Quota Change Request form

(https://docs.google.com/a/google.com/forms/d/1vb2MkAr9JcHrp6myQ3oTxCyBv2c7Iyc5wqIKqE3K4IE/viewform)

.

For App Engine, which is an auto-scaling resource, you can open a support ticket to have a feature enabled that lets you specify a maximum budget for each day. If you exceed the budget threshold, the application completely stops serving, so be careful how you use this feature. Note that the cap is an approximation; computing perfectly accurate consumption costs requires an offline job.

Set alerts for monthly spending thresholds

You can configure a monthly threshold for each billing account. Note that this is not a cap to the spend for the billing account. Instead, it is a monitoring mechanism. Billing administrators receive notifications when the account reaches 50%, 90%, and 100% of this monthly threshold. Actual billed amounts can be higher after all the billing processing is complete. You can configure these settings in the **Alerts** page for your billing account in the Cloud Platform Console.

Monitor and analyze BigQuery costs

You can get an estimate of BigQuery costs prior to billing by using the BigQuery Jobs:list (https://cloud.google.com/bigquery/docs/reference/v2/jobs/list) method to get a billed-bytes-by-query for the last six months. You can drill down on individual query costs by using the Jobs:get (https://cloud.google.com/bigquery/docs/reference/v2/jobs/get) method to find the query and its submitter (https://cloud.google.com/bigquery/docs/reference/v2/jobs#resource).

Risk management

Use projects to designate ownership of resources

You can use Cloud Platform projects to designate ownership of Cloud Platform resources within your organization. Keep in mind that a particular department can own multiple projects and the resources they contain. You can use project labels to denote organizational ownership or other dimensions you want to track. For example, you can add labels such as, `owner:marketing` or `cost-center:cc-123` to a project. These labels can be configured to be

visible in the Cloud Platform Console and the exported bill, and the labels can be used as a filter in API calls.

Only billing administrators can create new projects

Billing administrators are the only people who can create new billable projects associated with the billing accounts they administer. By making the billing administrators the people responsible for the resource spend, you can ensure responsibility for resource usage and cost.

Use project owners to delegate responsibility for controlling access to owned resources

Each project has a set of owners that must be individual people; groups are not allowed. These owners have the ability to determine who has access to the project and the type of access, or *role*. Use project owners to delegate responsibility for resources within your organization.

Use resource labels to further identify owners within a project or between projects

Some resources, such as VM Instances, enable you to add name-value labels. Use these labels to further classify resource ownership, such as `owner:johndoe`.

Grant users the appropriate permissions to facilitate least privilege

Cloud Platform provides a number of mechanisms to fine tune permissions for least privilege. Grant developers the least required privileges at the project level: `viewer`, `editor`, or `owner` role. These permissions are inherited by resources such as Compute Engine and Cloud Storage. The intent of the `owner` permission is to manage team membership, authorization, and so on.

Evaluate whether developers need project-level access at all, or whether granting access to a specific Cloud Platform resource is sufficient. For example, add SSH keys to your instances (https://cloud.google.com/compute/docs/instances/adding-removing-ssh-keys#instance-only) to grant developers access to individual virtual machine instances without granting access to your project or to all of the instances in your project.

Monitor policy compliance with ACL reports

Use Security reports (https://support.google.com/a/answer/6000269?ref_topic=6110458) to monitor domain roles such as super administrator, or use the Roles API (https://developers.google.com/admin-sdk/directory/v1/guides/manage-roles) to do this in a central reporting solution, and then raise an alert or revoke permission automatically when a compliance violation occurs.

Use the Cloud Platform Console to monitor project usage for out-of-policy assignments, or use the Cloud IAM API (https://cloud.google.com/iam/docs/managing-policies#access_control_via_api) instead in a central reporting solution and alert or revoke access automatically.

Use the Cloud Storage Bucket Access Controls API (https://cloud.google.com/storage/docs/json_api/v1/bucketAccessControls#resource) to monitor bucket usage for out of policy assignments and alert or revoke access automatically; this is particularly important for audit-log buckets.

Enforce policy with policy entitlement systems

Logging, monitoring, and auditing in Cloud Platform help you to recognize threats and mitigate risks. You can also define, manage, and enforce policy by integrating with third-party, policy-entitlement systems, or by building your own. This enables you to separate project ownership roles from policy management roles. Correctly implementing this sort of scheme requires good, advanced planning of project-owner accounts and permissions, particularly with respect to inherited permissions, such as those for Cloud Storage. You also should evaluate whether the APIs support your required policy-management operations.

Plan for disaster recovery

Service-interrupting events can happen at any time. Your network could have an outage, your latest application push might introduce a critical bug, or—in rare cases—you might even have to contend with a natural disaster.

When things go awry, it's important to have a robust, targeted, and well-tested disaster recovery plan (https://cloud.google.com/solutions/designing-a-disaster-recovery-plan). Cloud Platform provides many of the facilities you need to implement such a plan, such as redundancy, scalability, compliance, and security.

The Disaster Recovery Cookbook (https://cloud.google.com/solutions/disaster-recovery-cookbook) walks you through some scenarios to show you how Cloud Platform can help.

Familiarize yourself with terms of service and compliance certifications

Here are some key links:

- Terms of service (https://cloud.google.com/terms/)
- Compliance certifications (https://cloud.google.com/security/compliance)
- Privacy policy (https://www.google.com/intl/en/policies/privacy/)

What's next?

Try out other Google Cloud Platform features for yourself. Have a look at our tutorials (https://cloud.google.com/docs/tutorials).

All rights reserved. Java is a registered trademark of Oracle and/or its affiliates.

Last updated November 9, 2017.