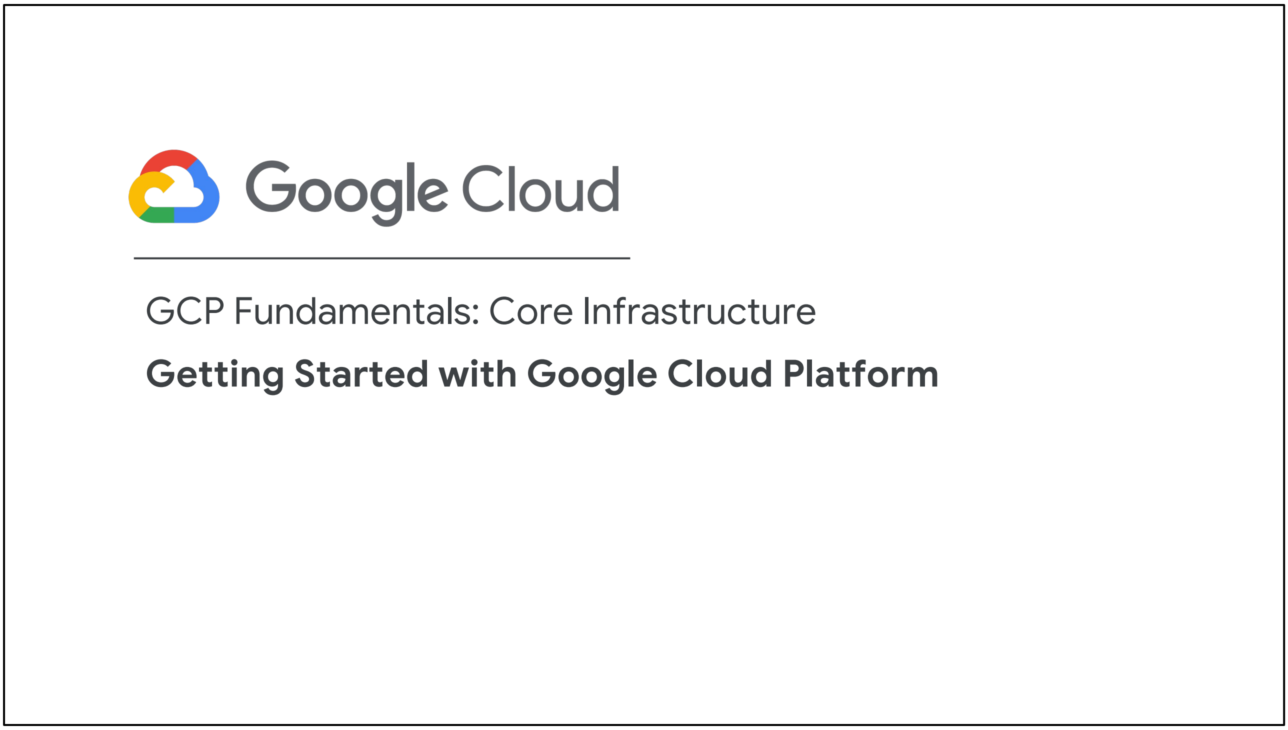
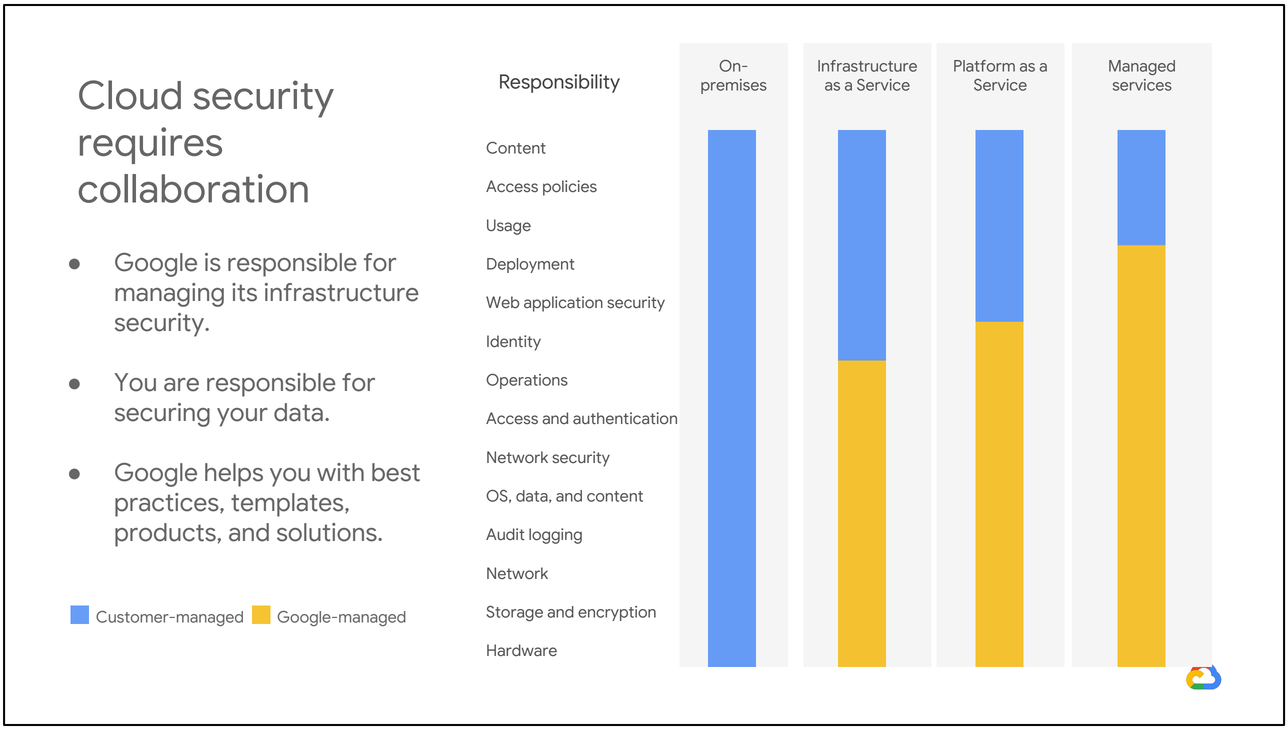


Google Cloud

GCP Fundamentals: Core Infrastructure

Getting Started with Google Cloud Platform



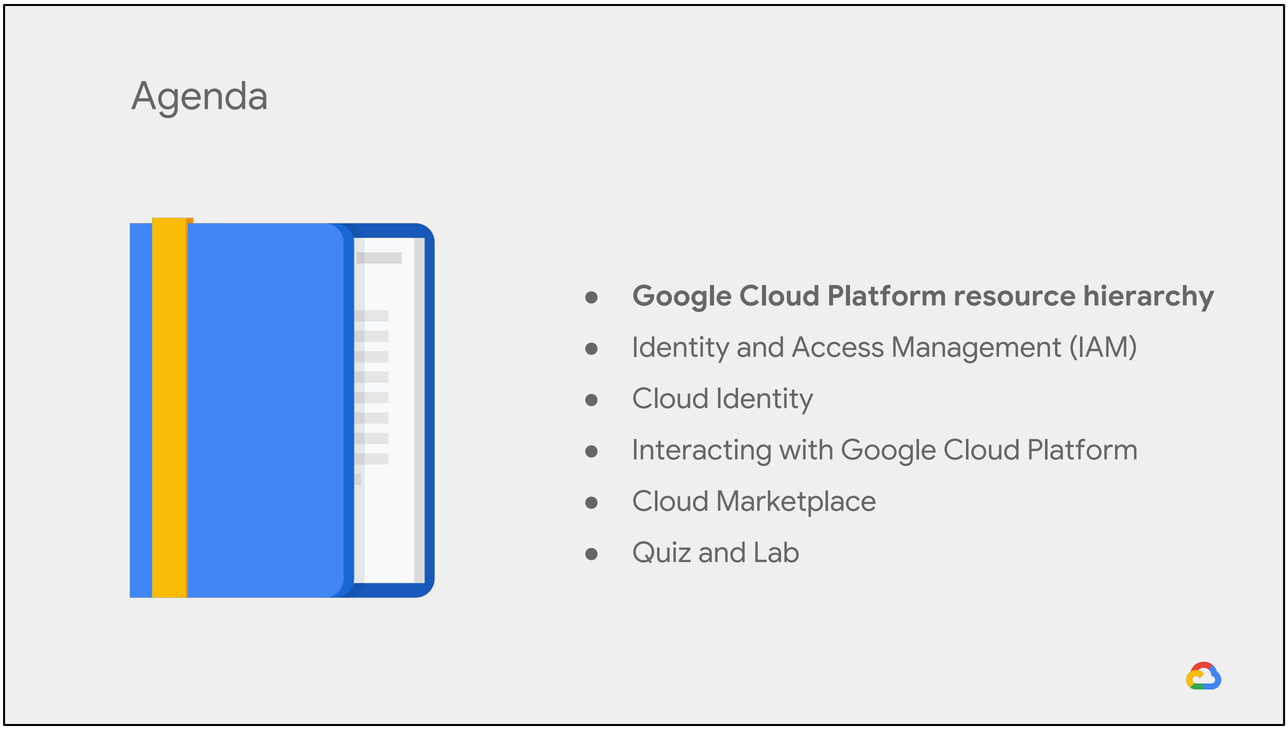
Cloud security requires collaboration

- Google is responsible for managing its infrastructure security.
- You are responsible for securing your data.
- Google helps you with best practices, templates, products, and solutions.

🟦 Customer-managed 🟨 Google-managed

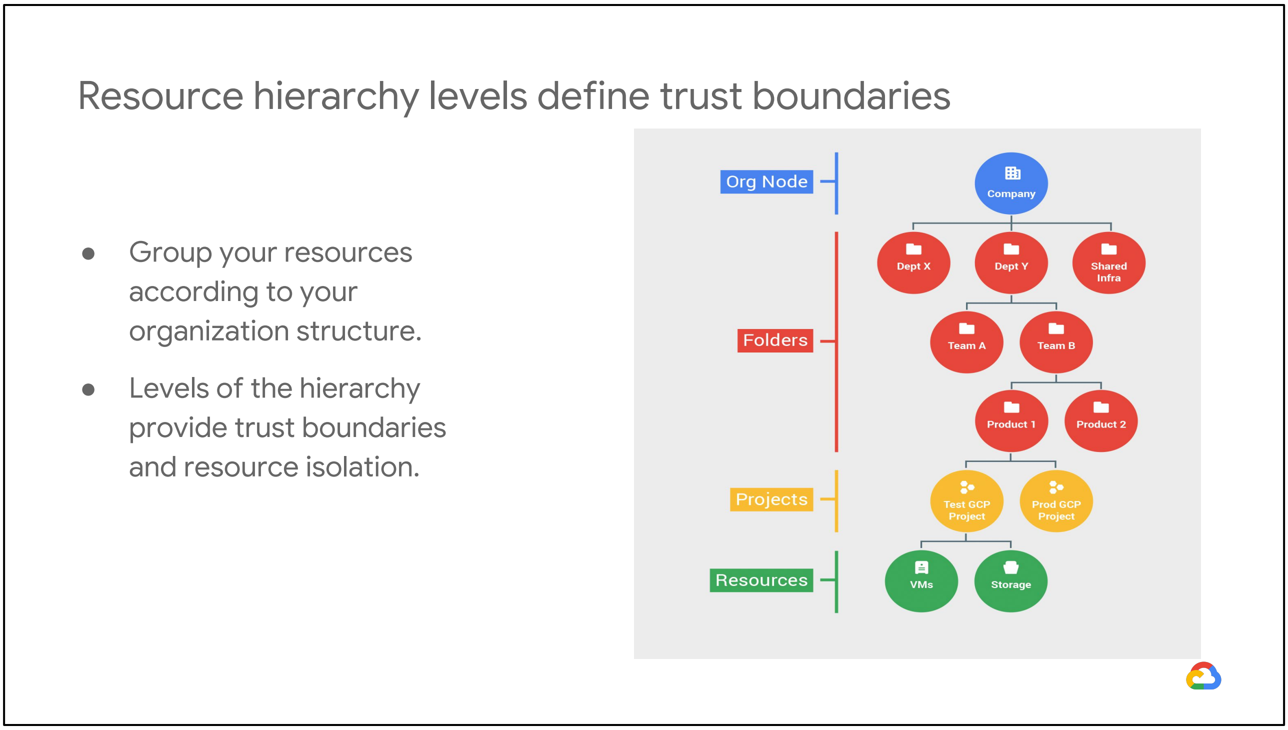
When you build an application on your on-premises infrastructure, you’re responsible for the entire stack’s security: from the physical security of the hardware and the premises in which they are housed, through the encryption of the data on disk, the integrity of your network, and all the way up to securing the content stored in those applications. When you move an application to Google Cloud Platform, Google handles many of the lower layers of security. Because of its scale, Google can deliver a higher level of security at these layers than most of its customers could afford to do on their own.

The upper layers of the security stack remain the customer’s responsibility. Google provides tools, such as IAM, to help customers implement the policies they choose at these layers.



Agenda

- **Google Cloud Platform resource hierarchy**
- Identity and Access Management (IAM)
- Cloud Identity
- Interacting with Google Cloud Platform
- Cloud Marketplace
- Quiz and Lab

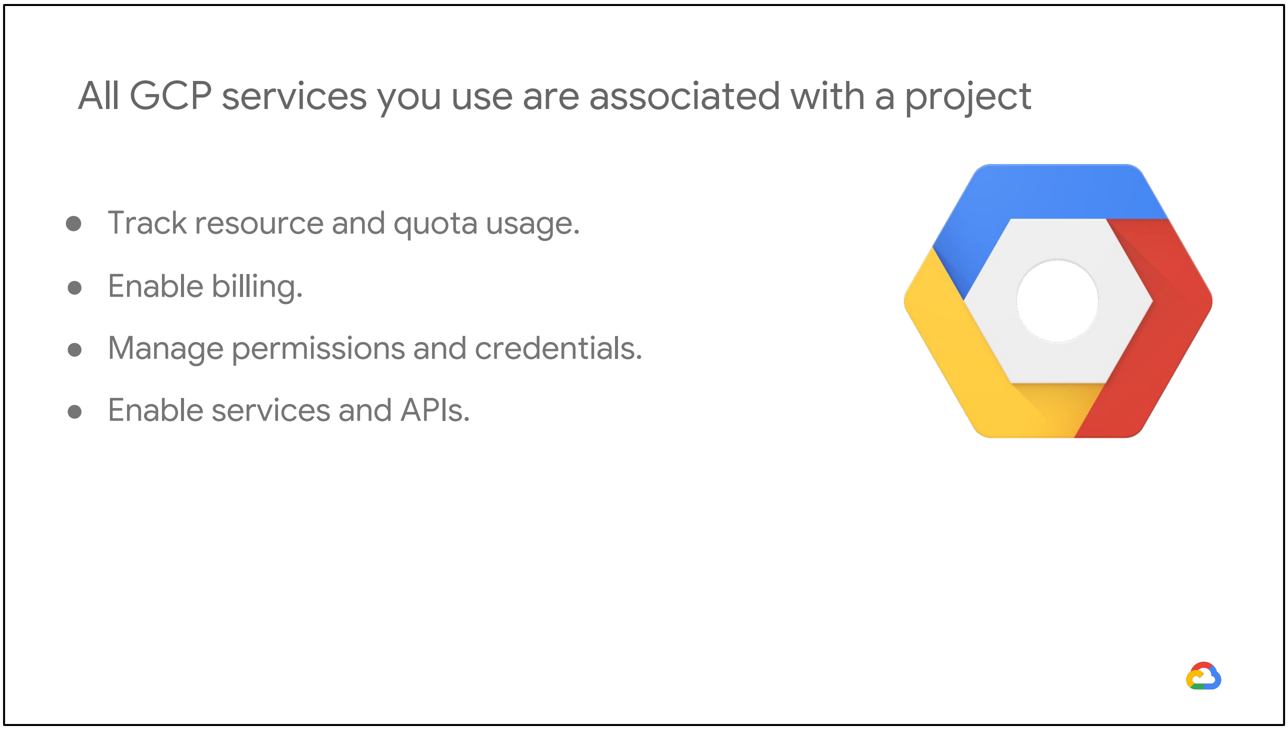


Resource hierarchy levels define trust boundaries

- Group your resources according to your organization structure.
- Levels of the hierarchy provide trust boundaries and resource isolation.

You may find it easiest to understand the GCP resource hierarchy from the bottom up. All the resources you use--whether they're virtual machines, Cloud Storage buckets, tables in BigQuery, or anything else in GCP--are organized into projects. Optionally, these projects may be organized into folders; folders can contain other folders. All the folders and projects used by your organization can be brought together under an organization node. Projects, folders, and organization nodes are all places where policies can be defined. Some GCP resources let you put policies on individual resources too, like Cloud Storage buckets. (This course discusses Cloud Storage buckets later in the course.)

Policies are inherited downwards in the hierarchy.



All GCP services you use are associated with a project

- Track resource and quota usage.
- Enable billing.
- Manage permissions and credentials.
- Enable services and APIs.

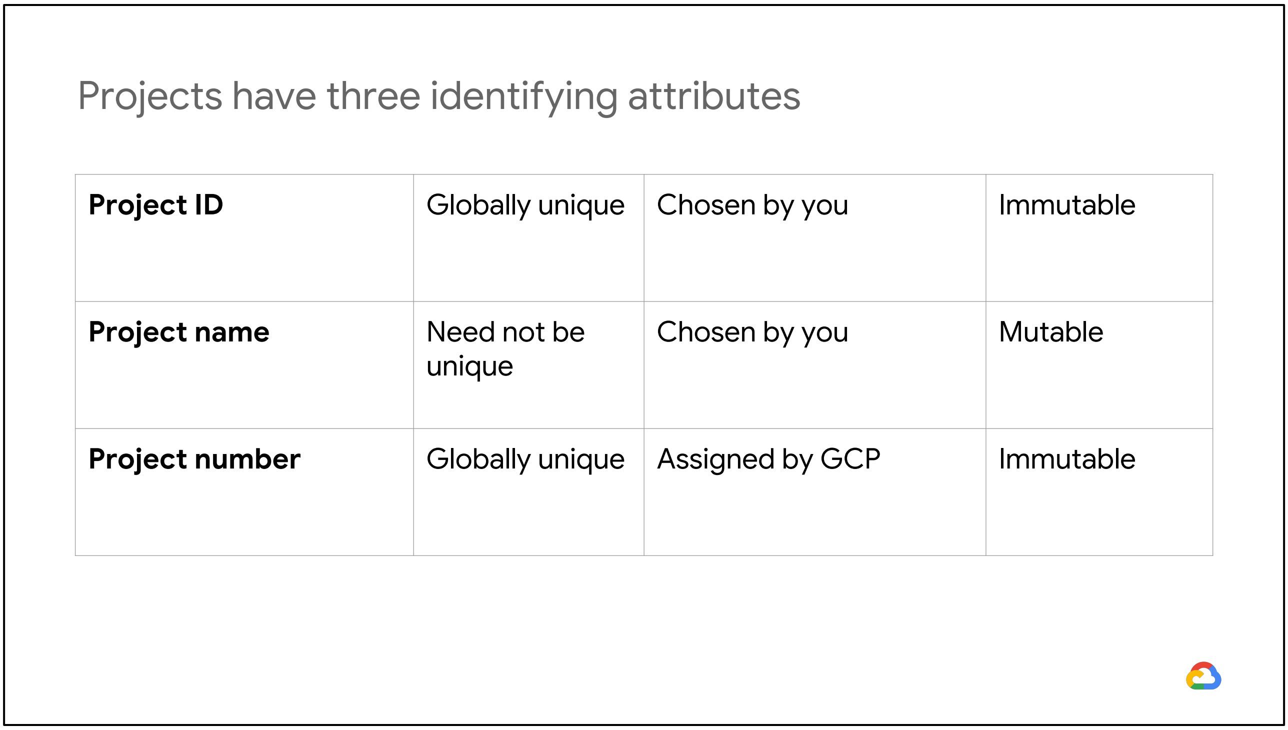
All Google Cloud Platform resources belong to a Google Cloud Platform Console project. Projects are the basis for enabling and using GCP services, like managing APIs, enabling billing, adding and removing collaborators, and enabling other Google services. Each project is a separate compartment, and each resource belongs to exactly one. Projects can have different owners and users. They're billed separately, and they're managed separately.

The Cloud Resource Manager provides methods that you can use to programmatically manage your projects in Google Cloud Platform. With this API, you can do the following:

- Get a list of all projects associated with an account.
- Create new projects.
- Update existing projects.
- Delete projects.
- Undelete, or recover, projects that you don't want to delete.

You can access Cloud Resource Manager in either of the following ways:

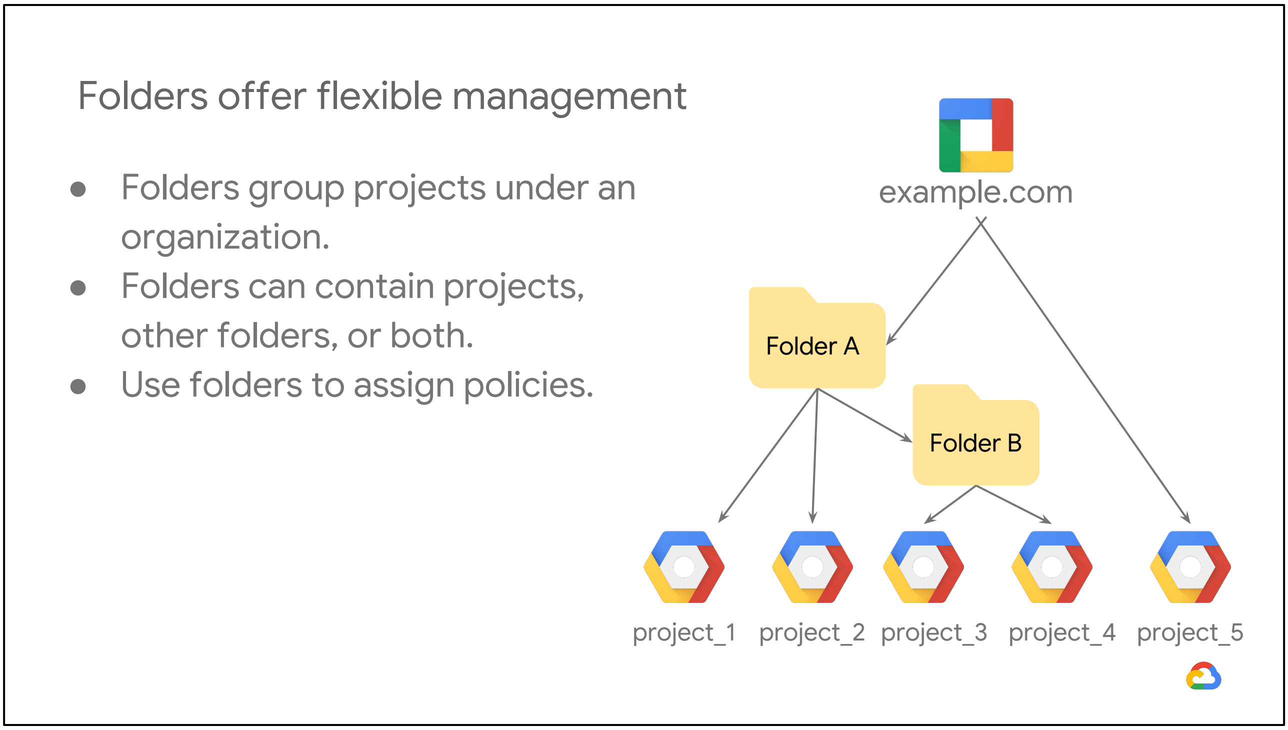
- Through the RPC API
- Through the REST API



Projects have three identifying attributes

Project ID	Globally unique	Chosen by you	Immutable
Project name	Need not be unique	Chosen by you	Mutable
Project number	Globally unique	Assigned by GCP	Immutable

Each GCP project has a name and project ID you assign. The project ID is a permanent, unchangeable identifier, and it has to be unique across GCP. You'll use project IDs in several contexts to tell GCP which project you want to work with. On the other hand, project names are for your convenience, and you can change them. GCP also assigns each of your projects a unique project number, and you'll see it displayed to you in various contexts, but using it is mostly outside the scope of this course. In general, project IDs are made to be human-readable strings, and you'll use them frequently to refer to projects.



Folders offer flexible management

- Folders group projects under an organization.
- Folders can contain projects, other folders, or both.
- Use folders to assign policies.

example.com

Folder A

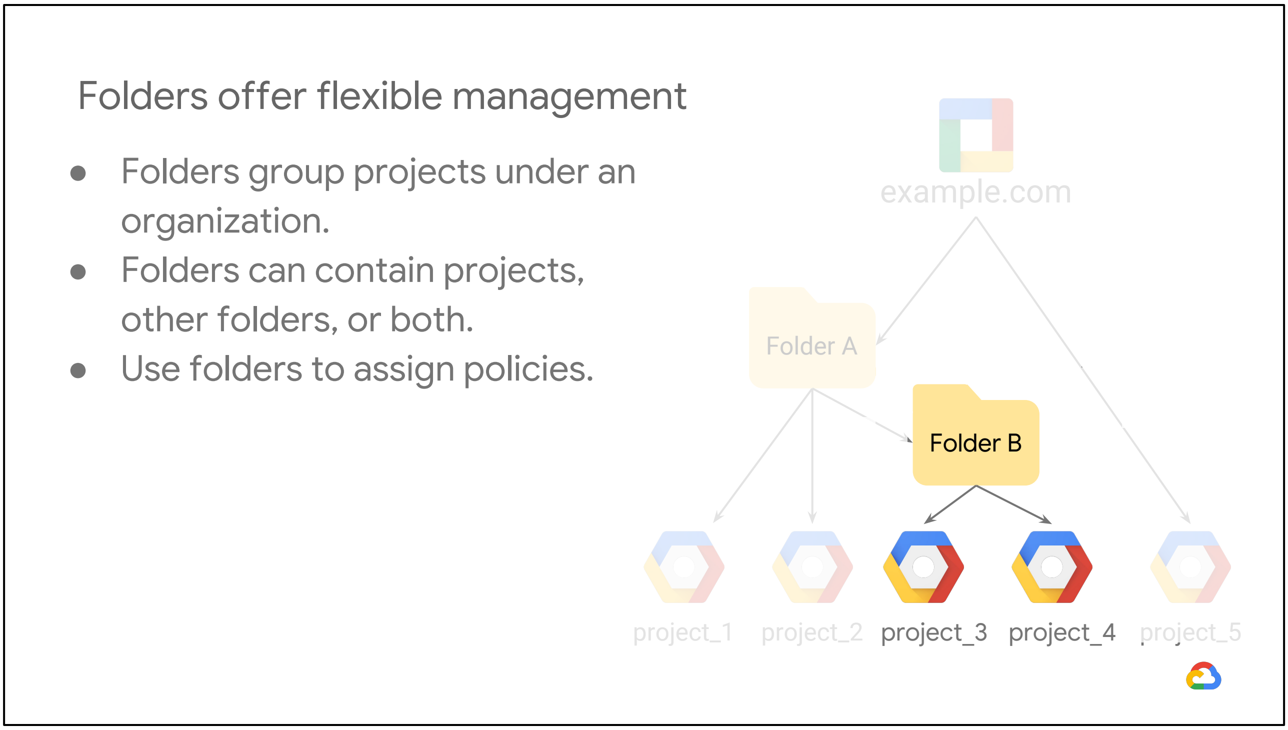
Folder B

project_1 project_2 project_3 project_4 project_5

The Cloud IAM Folders feature lets you assign policies to resources at a level of granularity you choose. The resources in a folder inherit IAM policies assigned to the folder.

A folder can contain projects, other folders, or a combination of both. You can use folders to group projects under an organization in a hierarchy. For example, your organization might contain multiple departments, each with its own set of GCP resources. Folders allows you to group these resources on a per-department basis

Folders let teams have the ability to delegate administrative rights, so that they can work independently.

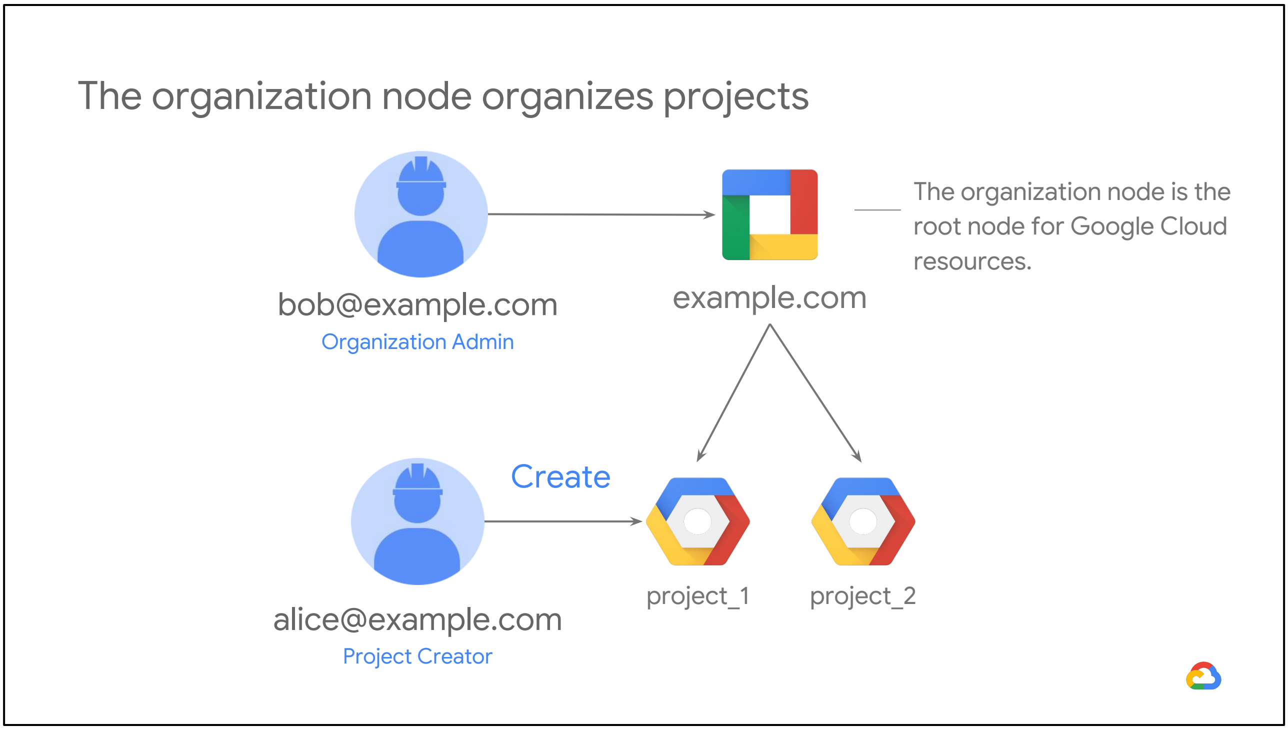


Folders offer flexible management

- Folders group projects under an organization.
- Folders can contain projects, other folders, or both.
- Use folders to assign policies.

The resources in a folder inherit IAM policies from the folder. So, if project 3 and project 4 are administered by the same team by design, you can put IAM policies onto Folder B instead. Doing it the other way-- putting duplicate copies of those policies on project 3 and project 4-- would be tedious and error-prone.

One word of caution: to use folders, you need an organization node at the top of the hierarchy.



The organization node organizes projects

bob@example.com
Organization Admin

example.com

The organization node is the root node for Google Cloud resources.

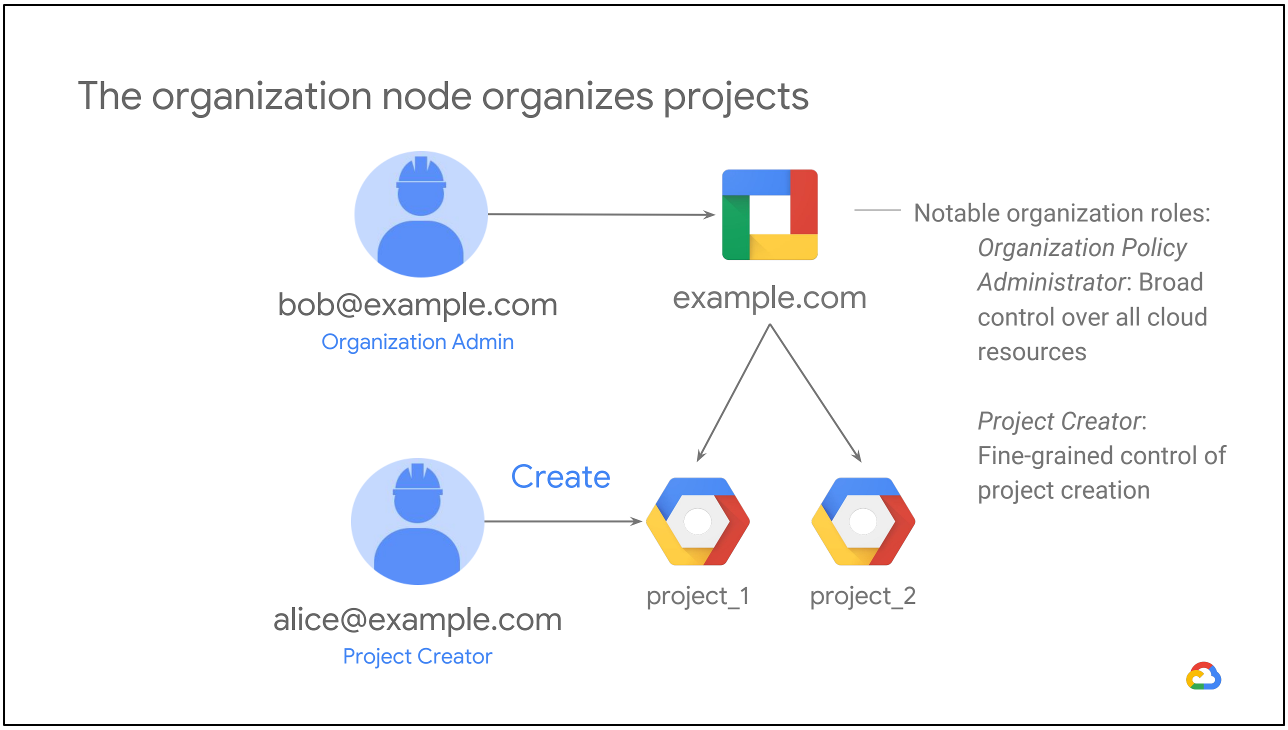
Create

alice@example.com
Project Creator

project_1

project_2

You probably want to organize all the projects in your company into a single structure. Most companies want the ability to apply to have centralized visibility of how resources are being used, and also to apply policies centrally. That's what the organization node is for. It's the top of the hierarchy.



The organization node organizes projects

bob@example.com

Organization Admin

example.com

Notable organization roles:

Organization Policy Administrator: Broad control over all cloud resources

Project Creator: Fine-grained control of project creation

Create

alice@example.com

Project Creator

project_1

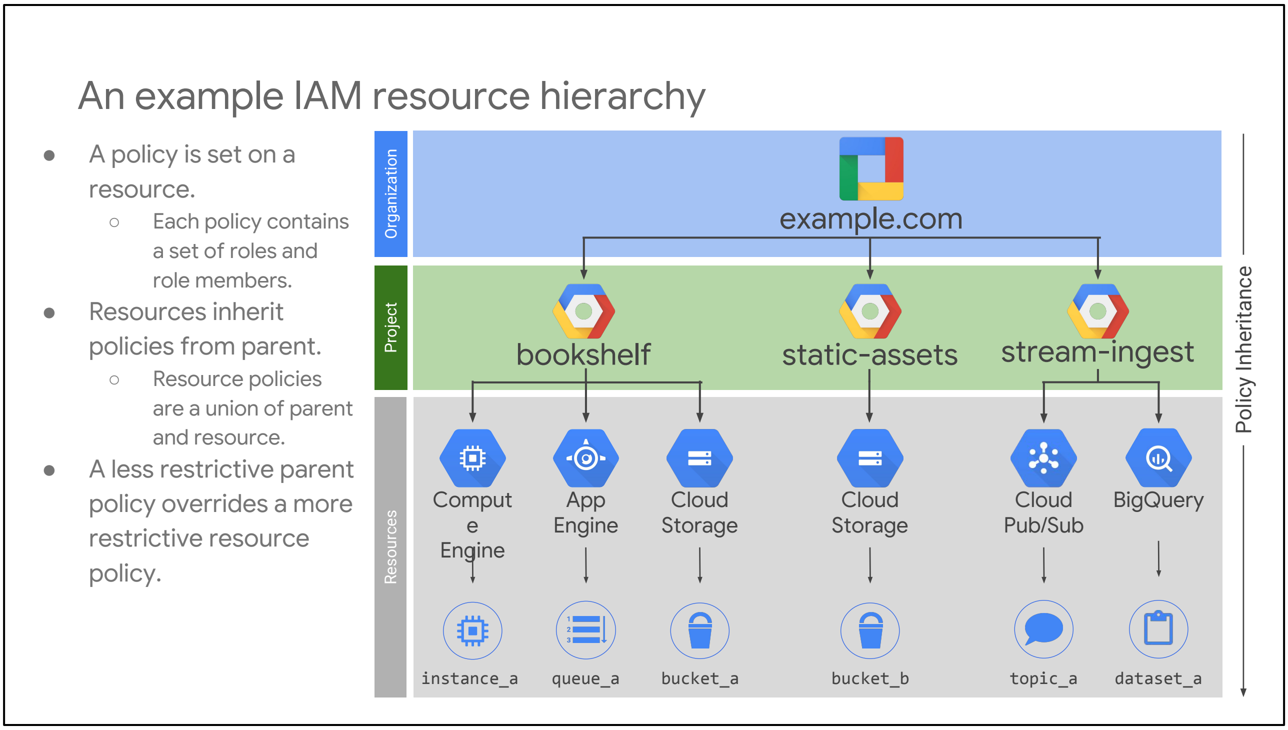
project_2

There are some special roles associated with it. For example, you can designate an organization policy administrator, so that only people with privilege can change policies. You can also assign a project creator role, which is a great way to control who can spend money.

So how do you get an organization node? In part the answer depends on whether your company is also a G Suite customer. If you have a G Suite domain, GCP projects will automatically belong to your organization node. Otherwise, you can use Google Cloud Identity to create one.

Here’s a tip: when you get a new organization node, it lets anyone in the domain create projects and billing accounts, just as they could before. That’s to avoid surprises and disruption. But it’d be a great first step with a new organization node to decide who on your team really should be able to do those things.

Once you have an organization node, you can create folders underneath it and put projects in.



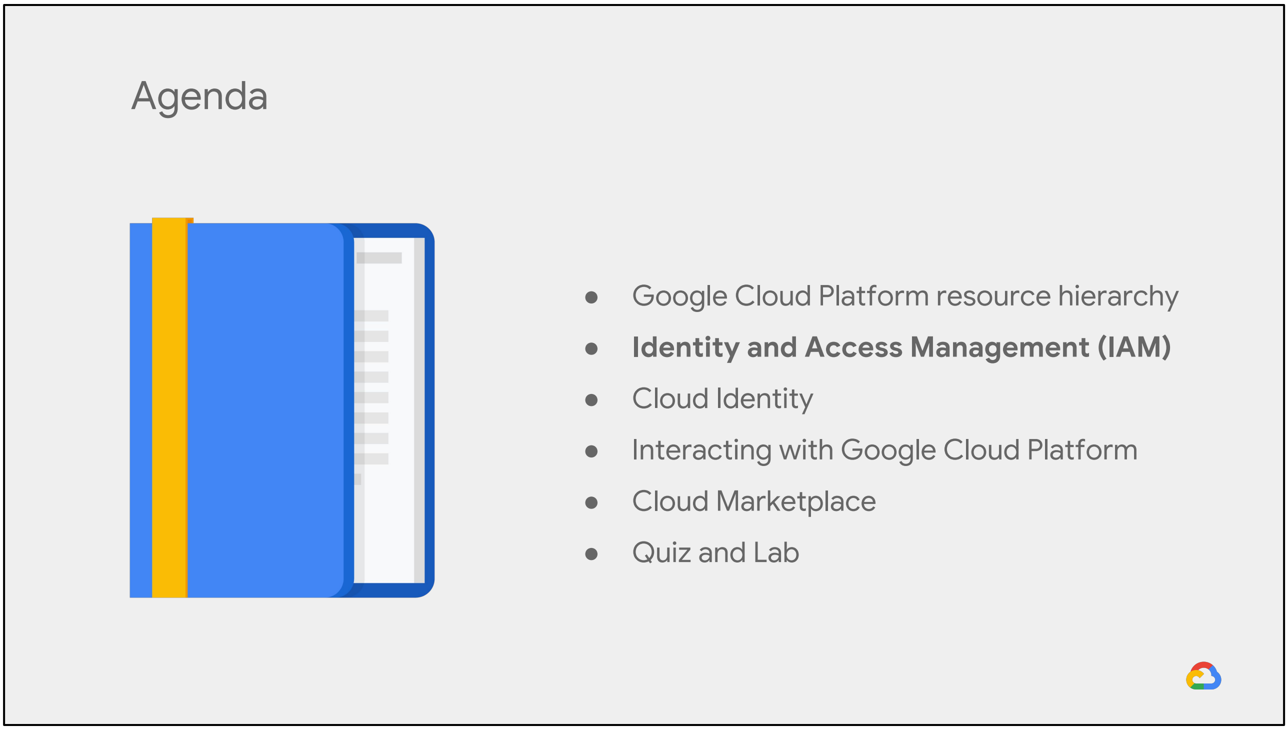
An example IAM resource hierarchy

- A policy is set on a resource.
    - Each policy contains a set of roles and role members.
- Resources inherit policies from parent.
    - Resource policies are a union of parent and resource.
- A less restrictive parent policy overrides a more restrictive resource policy.

Here's an example of how you might organize your resources. There are three projects, each of which uses resources from several GCP services. In this example, we haven't used any folders, although we always could move projects into folders if that became helpful.

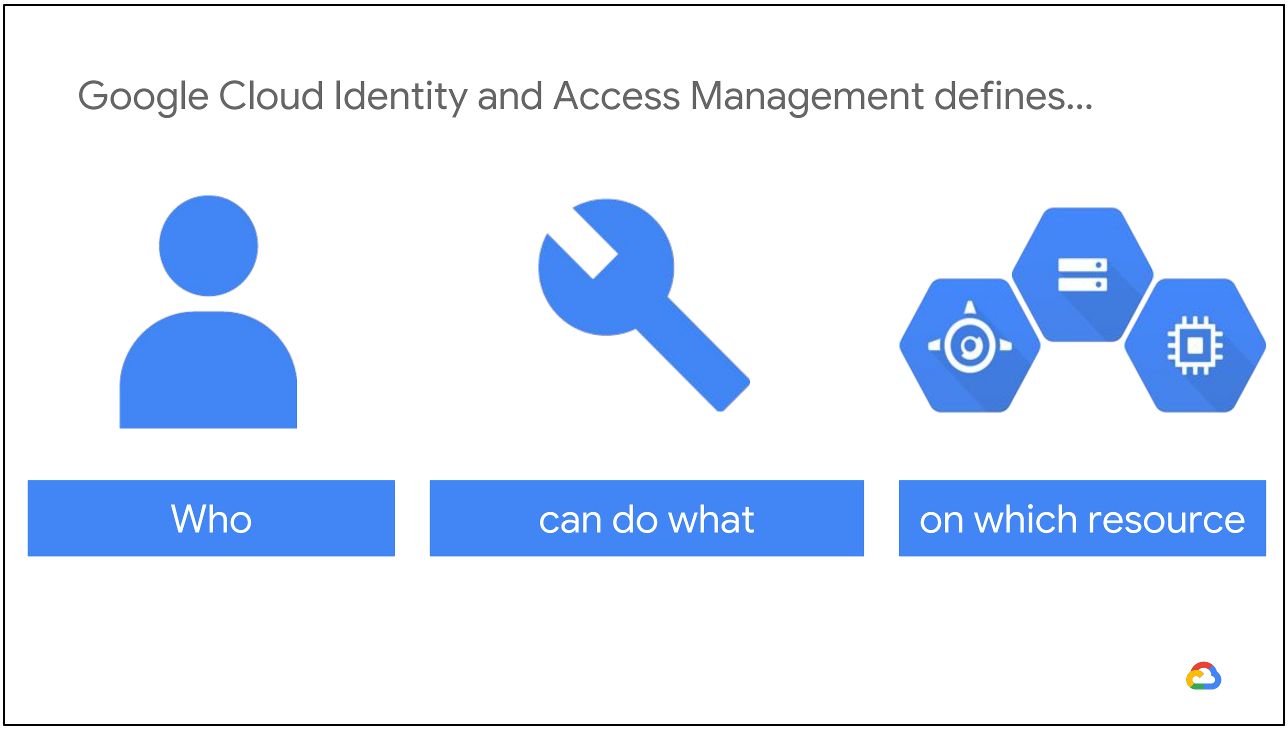
Resources inherit the policies of their parent resource. For instance, if you set a policy at the organization level, it is automatically inherited by all its children projects. And this inheritance is transitive, which means that all the resources in those projects inherit the policy too.

There's one important rule to keep in mind. The policies implemented at a higher level in this hierarchy can't take away access that's granted at lower level. For example, suppose that a policy applied on the "bookshelf" project gives user Pat the right to modify a Cloud Storage bucket. But a policy at the organization level says that Pat can only view Cloud Storage buckets, not change them. The more generous policy takes effect. Keep this in mind as you design your policies.



Agenda

- Google Cloud Platform resource hierarchy
- **Identity and Access Management (IAM)**
- Cloud Identity
- Interacting with Google Cloud Platform
- Cloud Marketplace
- Quiz and Lab



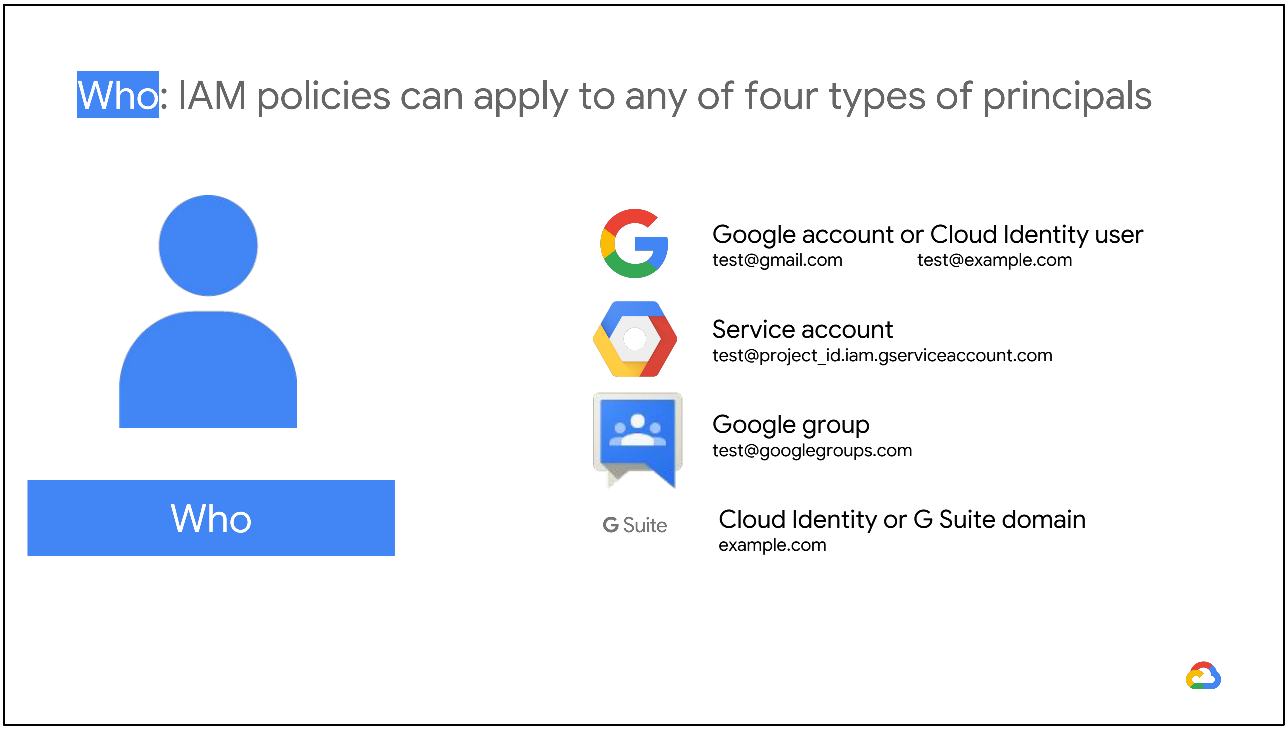
Google Cloud Identity and Access Management defines...

Who

can do what

on which resource

IAM lets administrators authorize who can take action on specific resources. An IAM policy has a “who” part, a “can do what” part, and an “on which resource” part.



Who: IAM policies can apply to any of four types of principals

Who

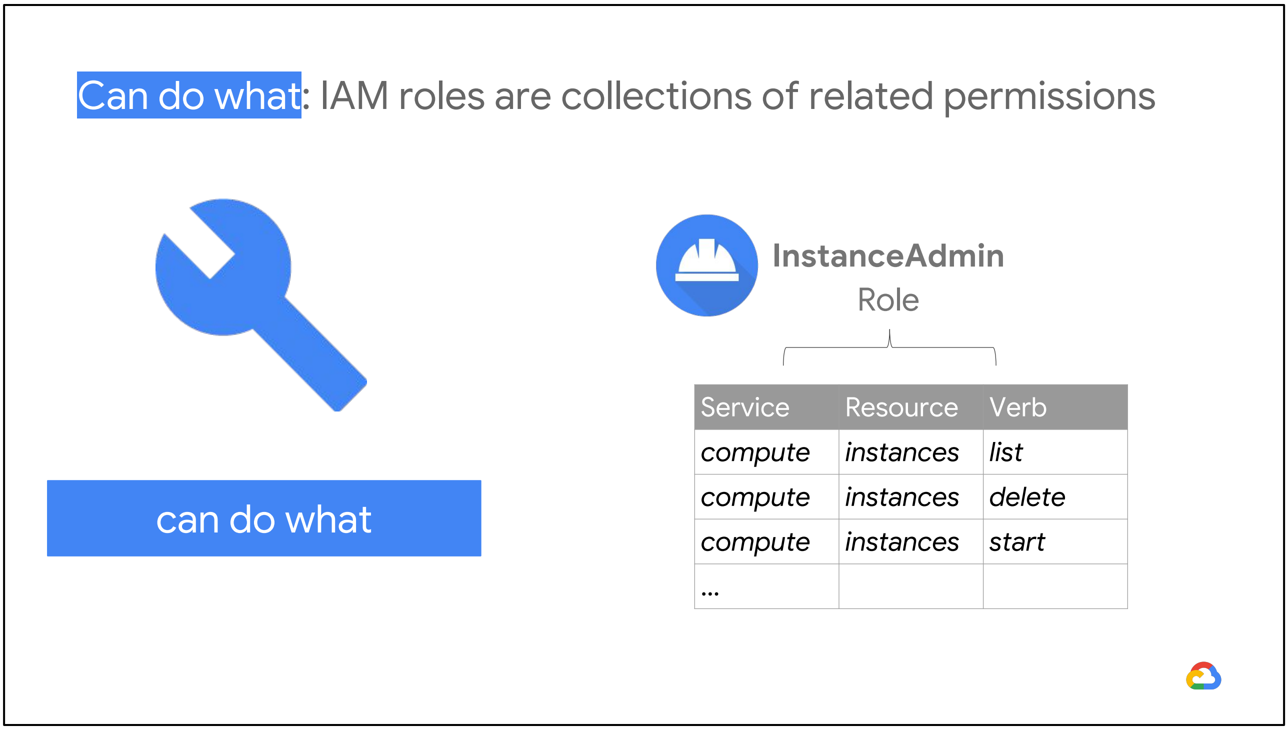
Google account or Cloud Identity user
test@gmail.com test@example.com

Service account
test@project_id.iam.gserviceaccount.com

Google group
test@googlegroups.com

Cloud Identity or G Suite domain
example.com

The “who” part of an IAM policy can be a Google account, a Google group, a service account, or an entire G Suite or Cloud Identity domain.



Can do what: IAM roles are collections of related permissions

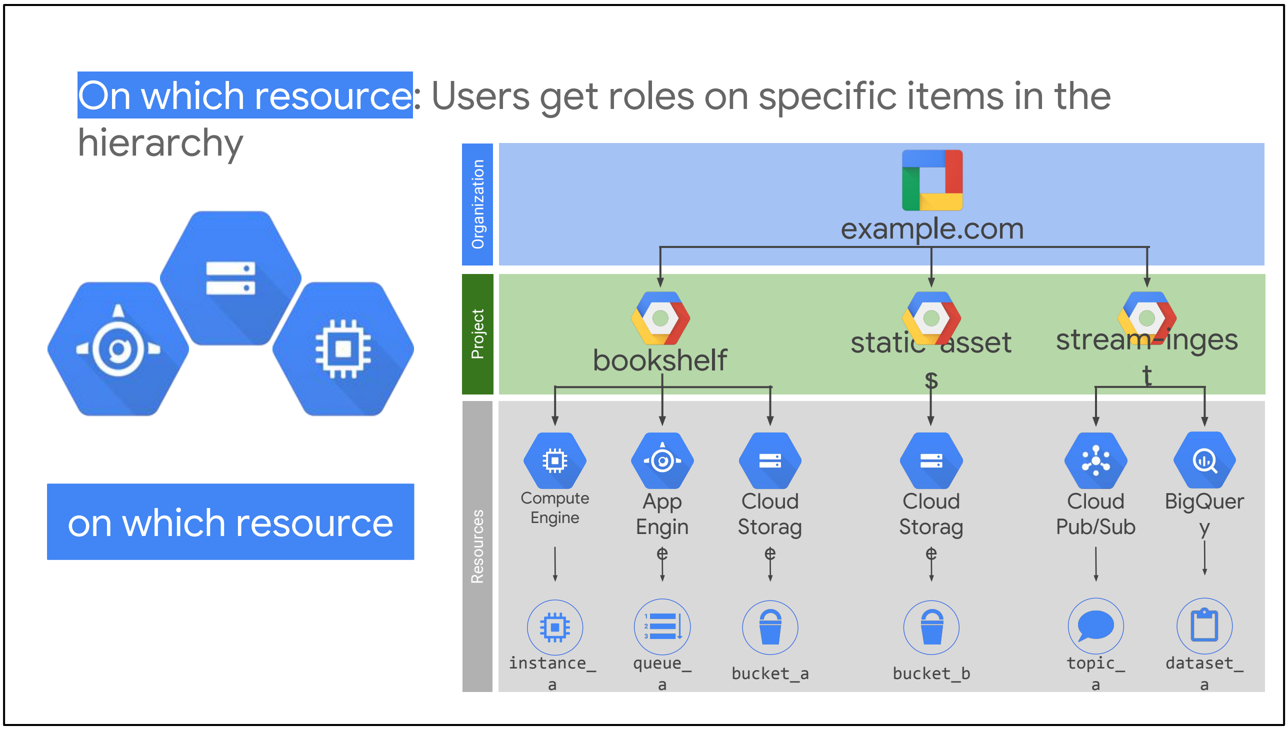
can do what

InstanceAdmin

Role

Service	Resource	Verb
compute	*instances*	*list*
compute	*instances*	*delete*
compute	*instances*	*start*
...		

The “can do what” part is defined by an IAM role. An IAM role is a collection of permissions. Most of the time, to do any meaningful operations, you need more than 1 permission. For example, to manage instances in a project, you need to create, delete, start, stop and change an instance. So the permissions are grouped together into a role to make them easier to manage.



On which resource: Users get roles on specific items in the hierarchy

on which resource

Organization

example.com

Project

bookshelf

static-assets

stream-ingest

Resources

Compute Engine

instance_a

App Engine

queue_a

Cloud Storage

bucket_a

Cloud Storage

bucket_b

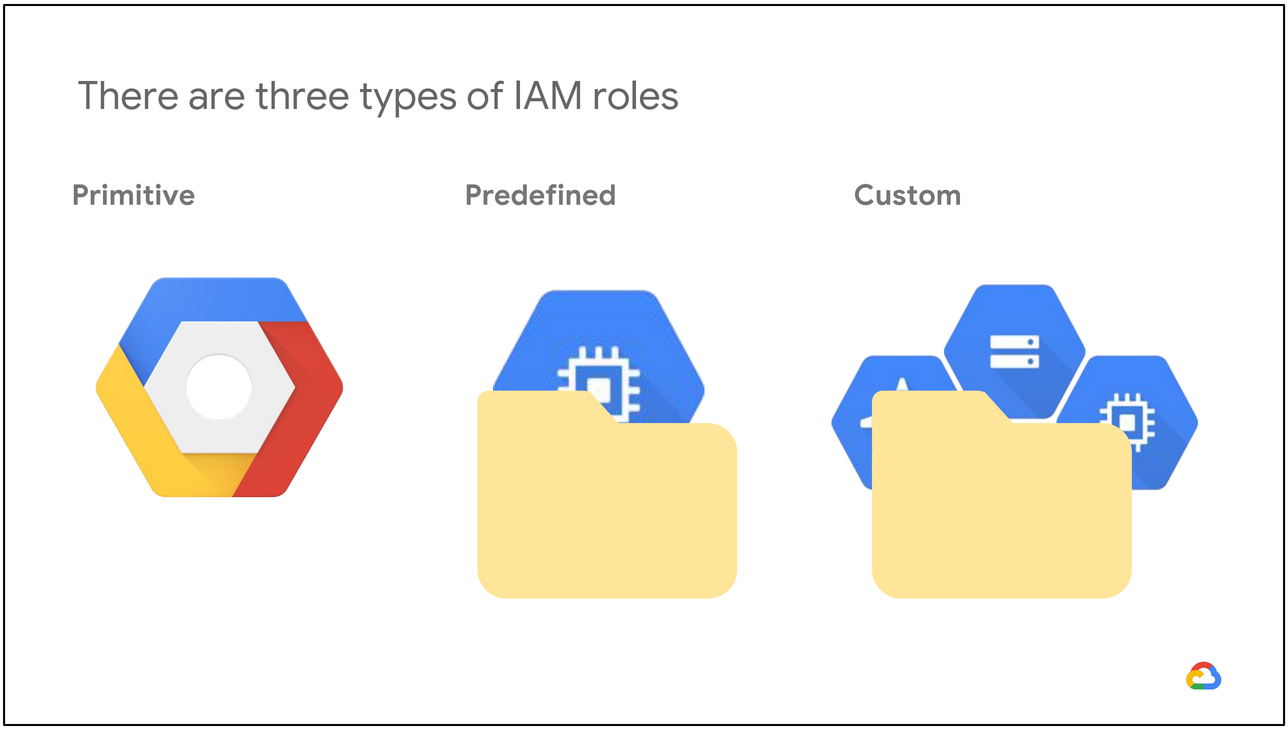
Cloud Pub/Sub

topic_a

BigQuery

dataset_a

When you give a user, group, or service account a role on a specific element of the resource hierarchy, the resulting policy applies to the element you chose, as well as to elements below it in the hierarchy.



There are three types of IAM roles

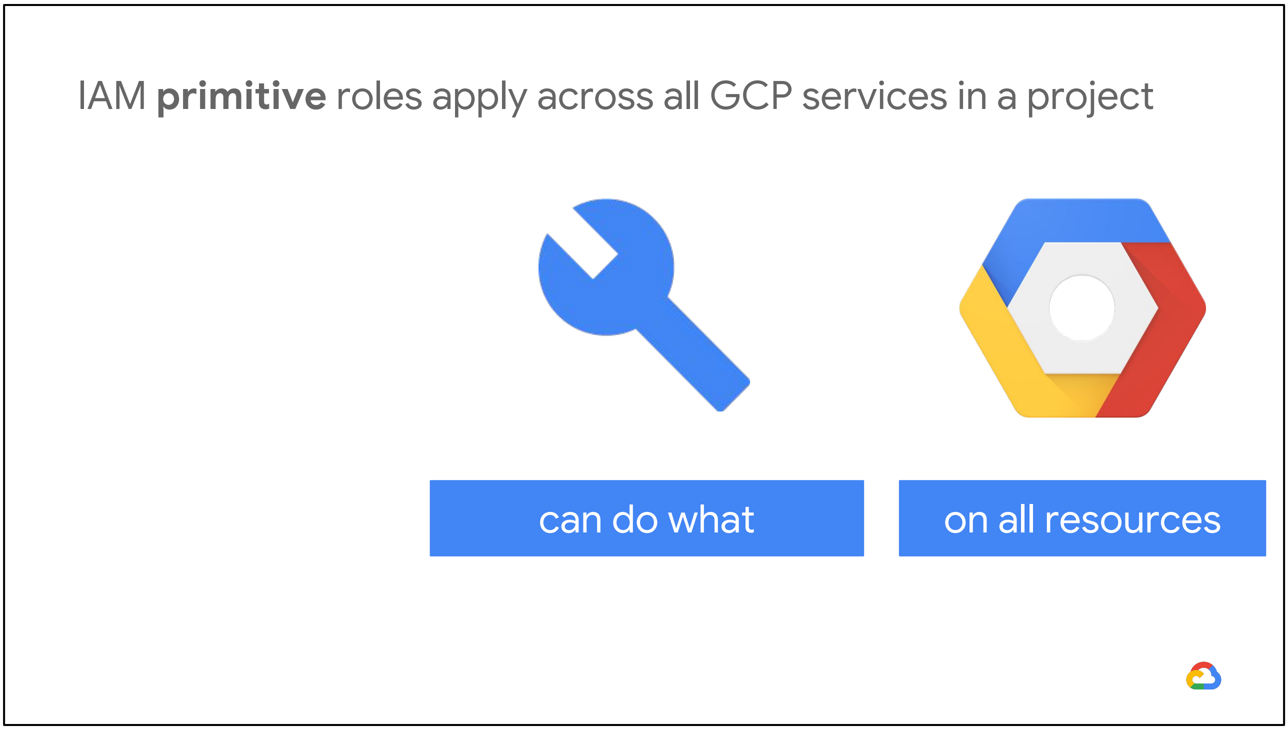
Primitive

Predefined

Custom

The “can do what” part of an IAM policy is defined by a role. An IAM role is a collection of permissions, because, most of the time you need more than 1 permission to do meaningful work. For example, to manage virtual machine instances in a project, you have to be able to create, delete, start, stop and change virtual machines. So these permissions are grouped together into a role to make them easier to understand and easier to manage.

There are three kinds of roles in Cloud IAM. Let’s talk about each in turn.

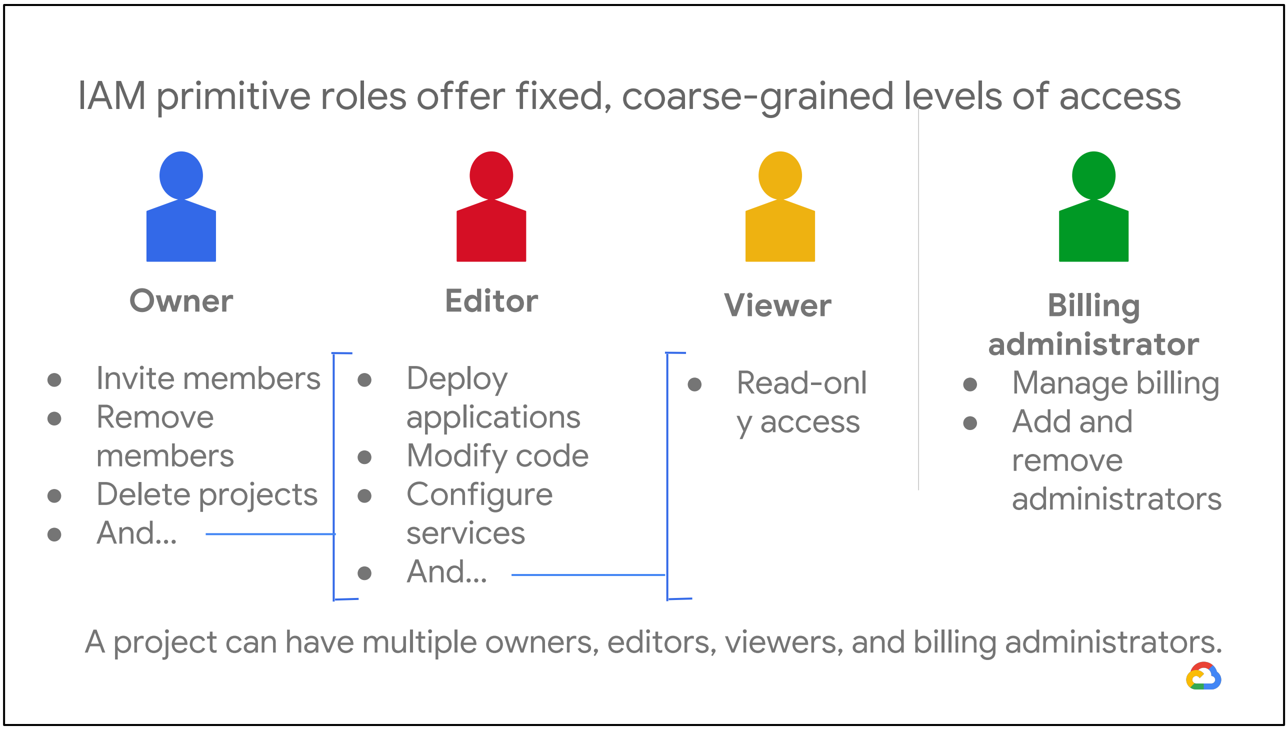


IAM **primitive** roles apply across all GCP services in a project

can do what

on all resources

Primitive roles are broad. You apply them to a GCP project, and they affect all resources in that project.



IAM primitive roles offer fixed, coarse-grained levels of access

Owner

- Invite members
- Remove members
- Delete projects
- And...

Editor

- Deploy applications
- Modify code
- Configure services
- And...

Viewer

- Read-only access

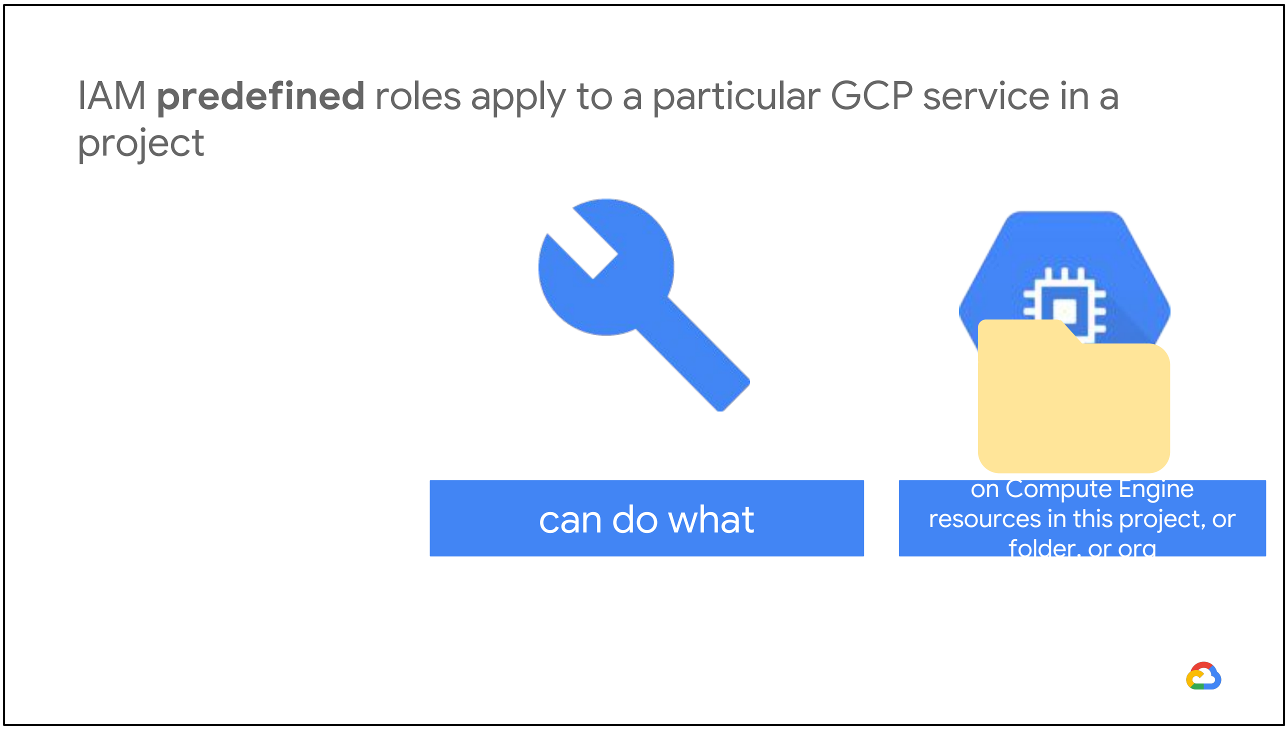
Billing administrator

- Manage billing
- Add and remove administrators

A project can have multiple owners, editors, viewers, and billing administrators.

These are the Owner, Editor, and Viewer roles. If you're a viewer on a given resource, you can examine it but not change its state. If you're an editor, you can do everything a viewer can do plus change its state. And if you're an owner, you can do everything an editor can do plus manage roles and permissions on the resource. The owner role on a project lets you do one more thing too: you can set up billing. Often companies want someone to be able to control the billing for a project without the right to change the resources in the project, and that's why you can grant someone the billing administrator role.

Be careful! If you have several people working together on a project that contains sensitive data, primitive roles are probably too coarse a tool. Fortunately, GCP IAM provides finer-grained types of roles.



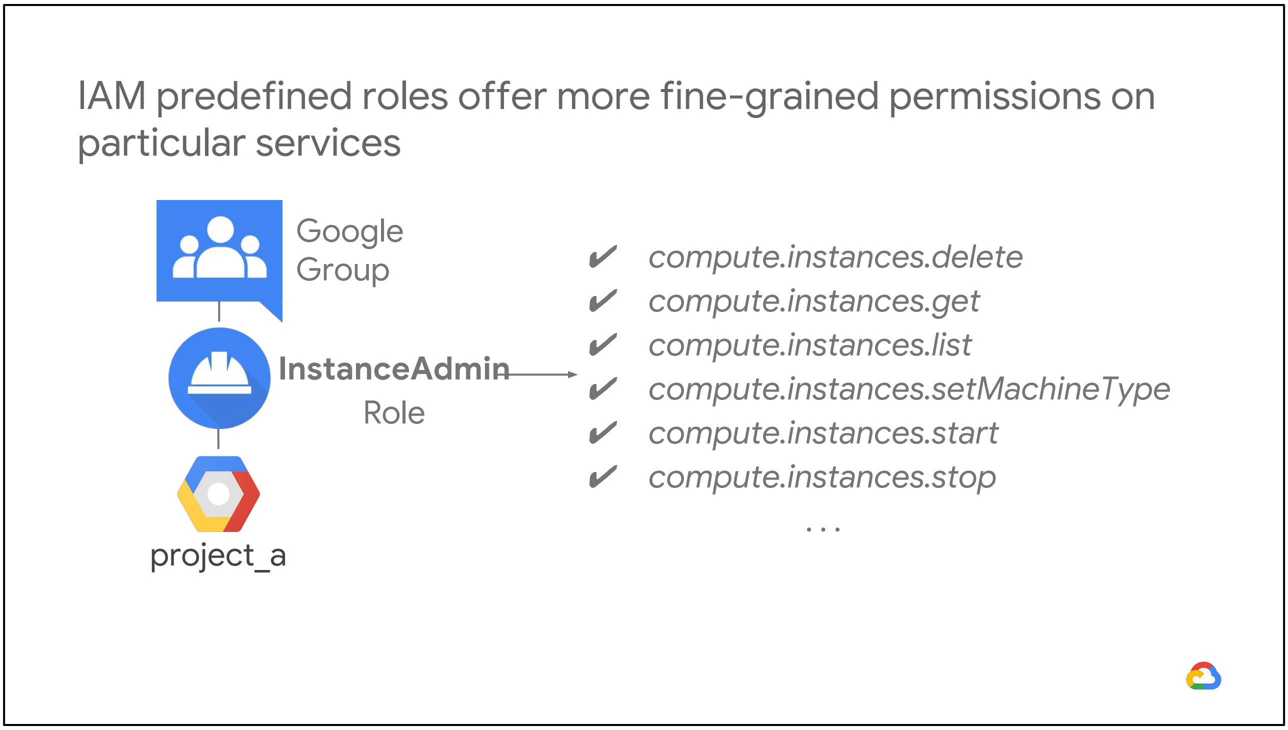
IAM **predefined** roles apply to a particular GCP service in a project

can do what

on Compute Engine resources in this project, or folder, or org

GCP services offers their own sets of predefined roles, and they define where those roles can be applied. For example, later in this course, we'll talk more about Compute Engine, which offers virtual machines as a service. Compute Engine offers a set of predefined roles, and you can apply them to Compute Engine resources in a given project, a given folder, or an entire organization.

Another example: consider Cloud Bigtable, which is a managed database service. Cloud Bigtable offers roles that can apply across an entire organization, to a particular project, or even to individual Bigtable database instances.



IAM predefined roles offer more fine-grained permissions on particular services

Google Group

InstanceAdmin Role

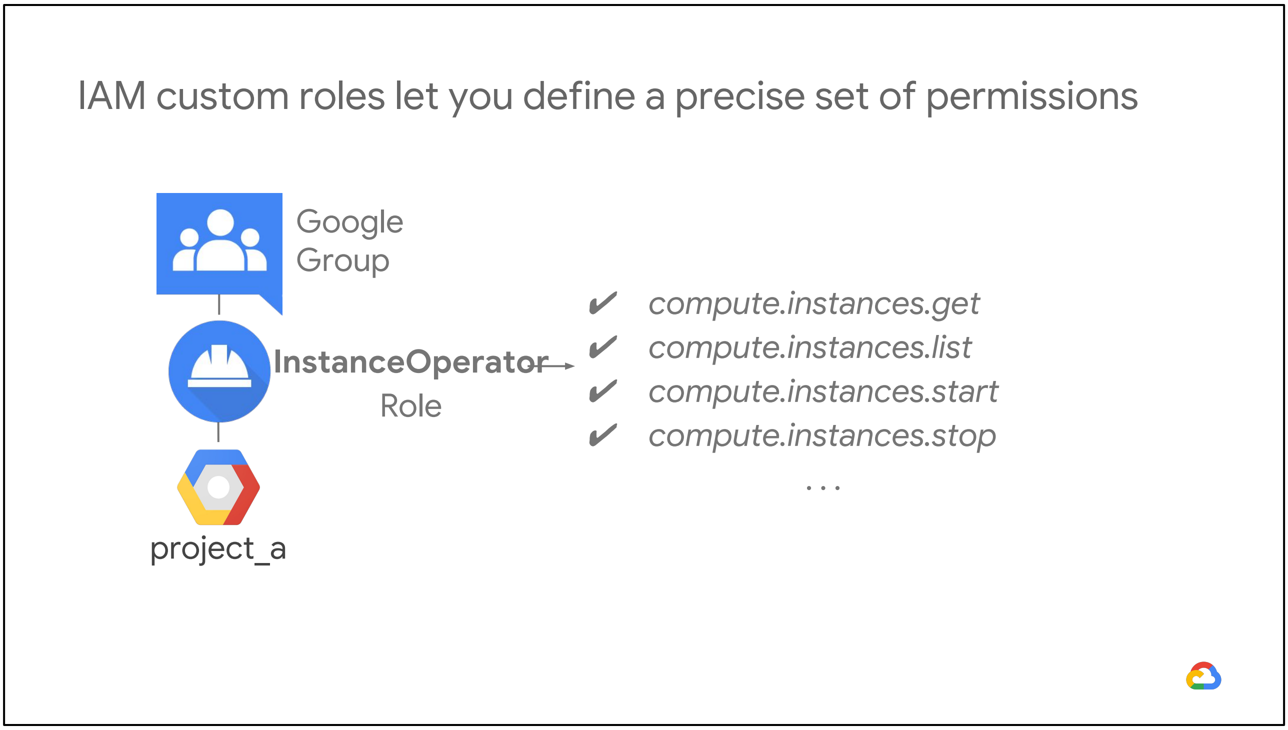
project_a

- ✔ *compute.instances.delete*
- ✔ *compute.instances.get*
- ✔ *compute.instances.list*
- ✔ *compute.instances.setMachineType*
- ✔ *compute.instances.start*
- ✔ *compute.instances.stop*

. . .

Compute Engine’s instanceAdmin role let's whoever has it perform a certain set of actions on virtual machines. What set of actions? Those listed here: listing them, reading and changing their configurations, and starting and stopping them. And which virtual machines? Well, that depends on where the role is applied.

In this example, all the users of a certain Google group have the role, and they have it on all the virtual machines in project A.



IAM custom roles let you define a precise set of permissions

Google Group

InstanceOperator → Role

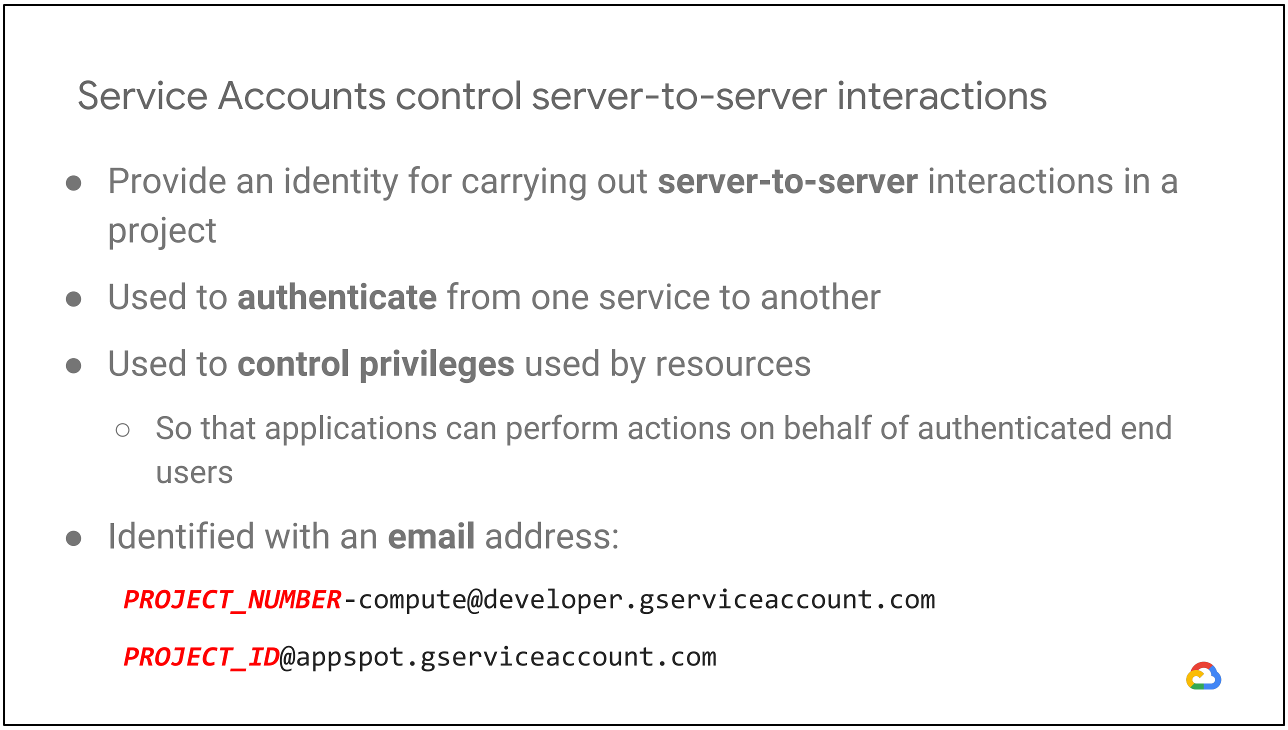
- ✔ *compute.instances.get*
- ✔ *compute.instances.list*
- ✔ *compute.instances.start*
- ✔ *compute.instances.stop*

. . .

project_a

What if you need something even finer-grained? That's what custom roles permit. A lot of companies use a "least-privilege" model, in which each person in your organization the minimal amount of privilege needed to do his or her job. So, for example, maybe I want to define an "instanceOperator" role, to allow some users to stop and start Compute Engine virtual machines but not reconfigure them. Custom roles allow me to do that.

A couple of cautions about custom roles. First, if you decide to use custom roles, you'll need to manage the permissions that make them up. Some companies decide they'd rather stick with the predefined roles. Second, custom roles can only be used at the project or organization levels. They can't be used at the folder level.



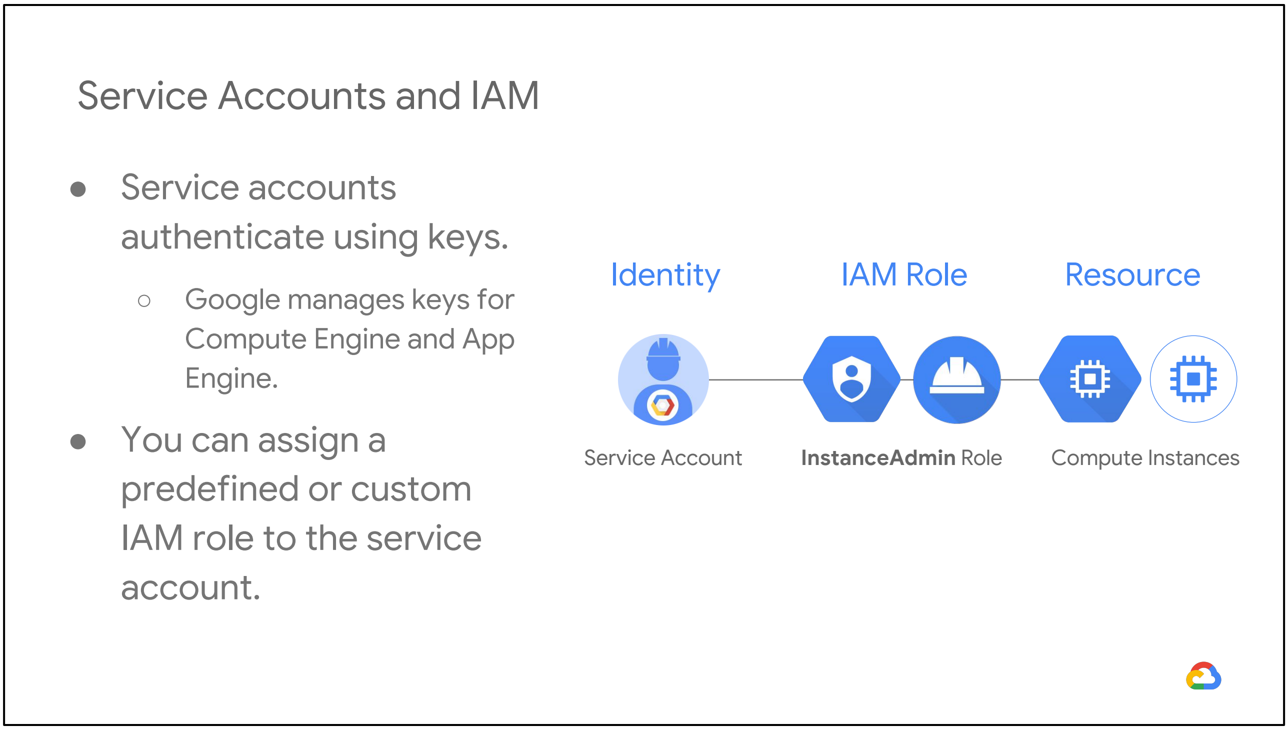
Service Accounts control server-to-server interactions

- Provide an identity for carrying out **server-to-server** interactions in a project
- Used to **authenticate** from one service to another
- Used to **control privileges** used by resources
  - So that applications can perform actions on behalf of authenticated end users
- Identified with an **email** address:

```
PROJECT_NUMBER-compute@developer.gserviceaccount.com

PROJECT_ID@appspot.gserviceaccount.com
```

What if you want to give permissions to a Compute Engine virtual machine rather than to a person? That's what service accounts are for. For instance, maybe you have an application running in a virtual machine that needs to store data in Google Cloud Storage. But you don't want to let just anyone on the Internet have access to that data; only that virtual machine. So you'd create a service account to authenticate your VM to Cloud Storage. Service accounts are named with an email address, but instead of passwords they use cryptographic keys to access resources.

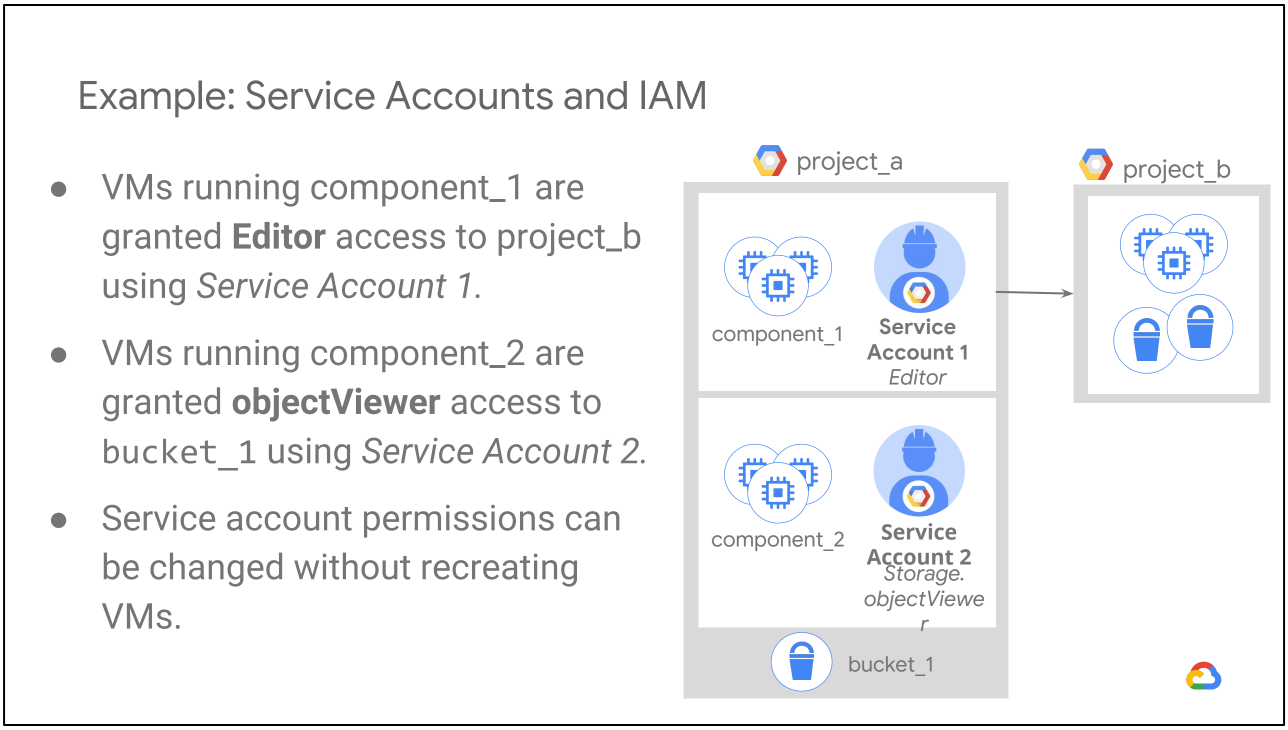


Service Accounts and IAM

- Service accounts authenticate using keys.
  - Google manages keys for Compute Engine and App Engine.
- You can assign a predefined or custom IAM role to the service account.

In this simple example, a service account has been granted Compute Engine's Instance Admin role. This would allow an application running in a VM with that service account to create, modify, and delete other VMs.

Incidentally, service accounts need to be managed too! For example, maybe Alice needs to manage what can act as a given service account, while Bob just needs to be able to view what can. Fortunately, in addition to being an identity, a service account is also a resource! So it can have IAM policies of its own attached to it. For instance, Alice can have the editor role on a service account and Bob can have the viewer role. This is just like granting roles for any other GCP resource.

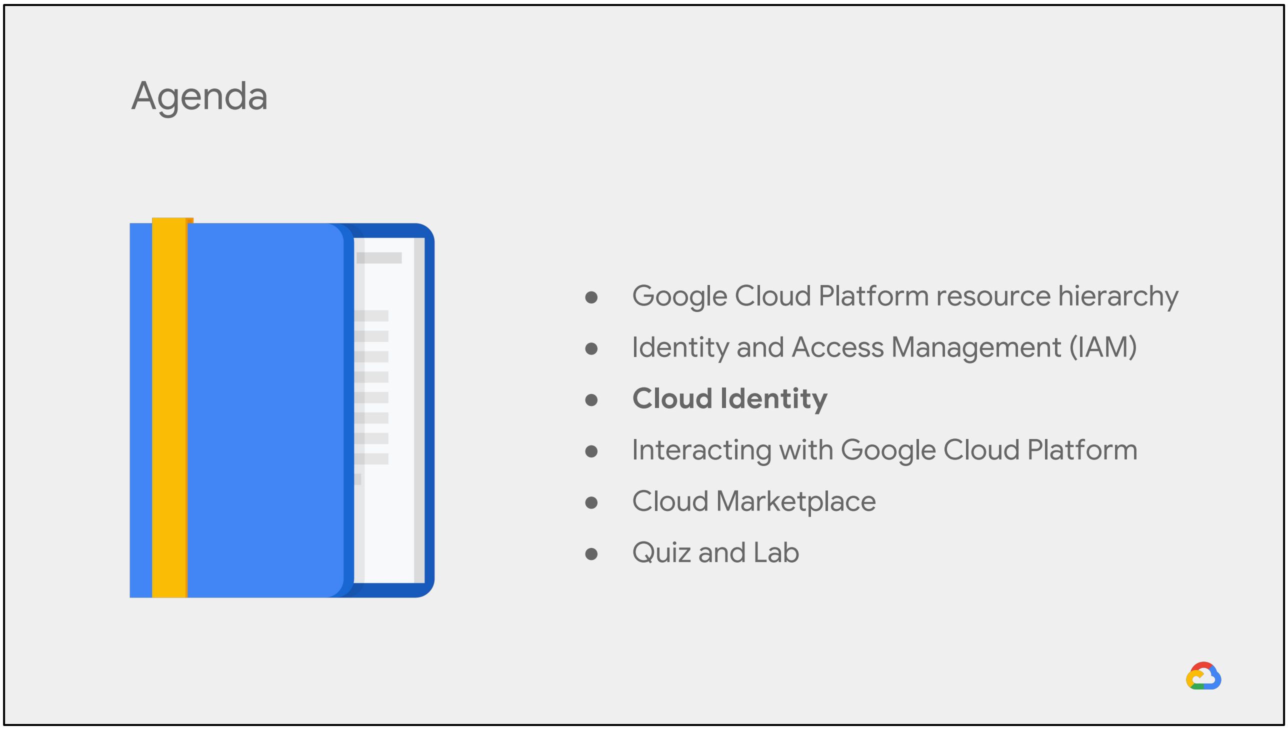


Example: Service Accounts and IAM

- VMs running component_1 are granted **Editor** access to project_b using *Service Account 1.*
- VMs running component_2 are granted **objectViewer** access to `bucket_1` using *Service Account 2.*
- Service account permissions can be changed without recreating VMs.

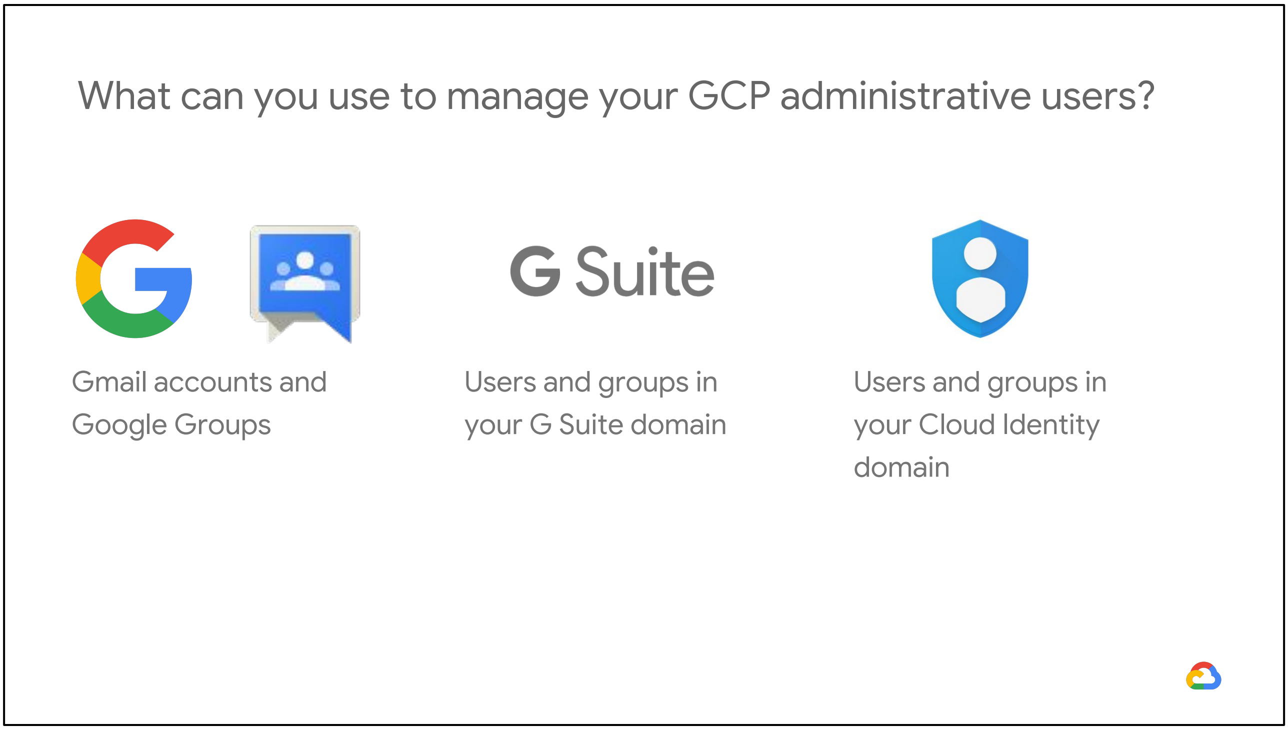
You can grant different groups of VMs in your project different identities. This makes it easier to manage different permissions for each group. You also can change the permissions of the service accounts without having to recreate the VMs.

Here's a more complex example. Say you have an application that's implemented across a group of Compute Engine virtual machines. One component of your application needs to have an editor role on another project, but another component doesn't. So you would create two different service accounts, one for each subgroup of virtual machines. Only the first service account has privilege on the other project. That reduces the potential impact of a miscoded application or a compromised virtual machine.



Agenda

- Google Cloud Platform resource hierarchy
- Identity and Access Management (IAM)
- **Cloud Identity**
- Interacting with Google Cloud Platform
- Cloud Marketplace
- Quiz and Lab



What can you use to manage your GCP administrative users?

Gmail accounts and Google Groups

G Suite

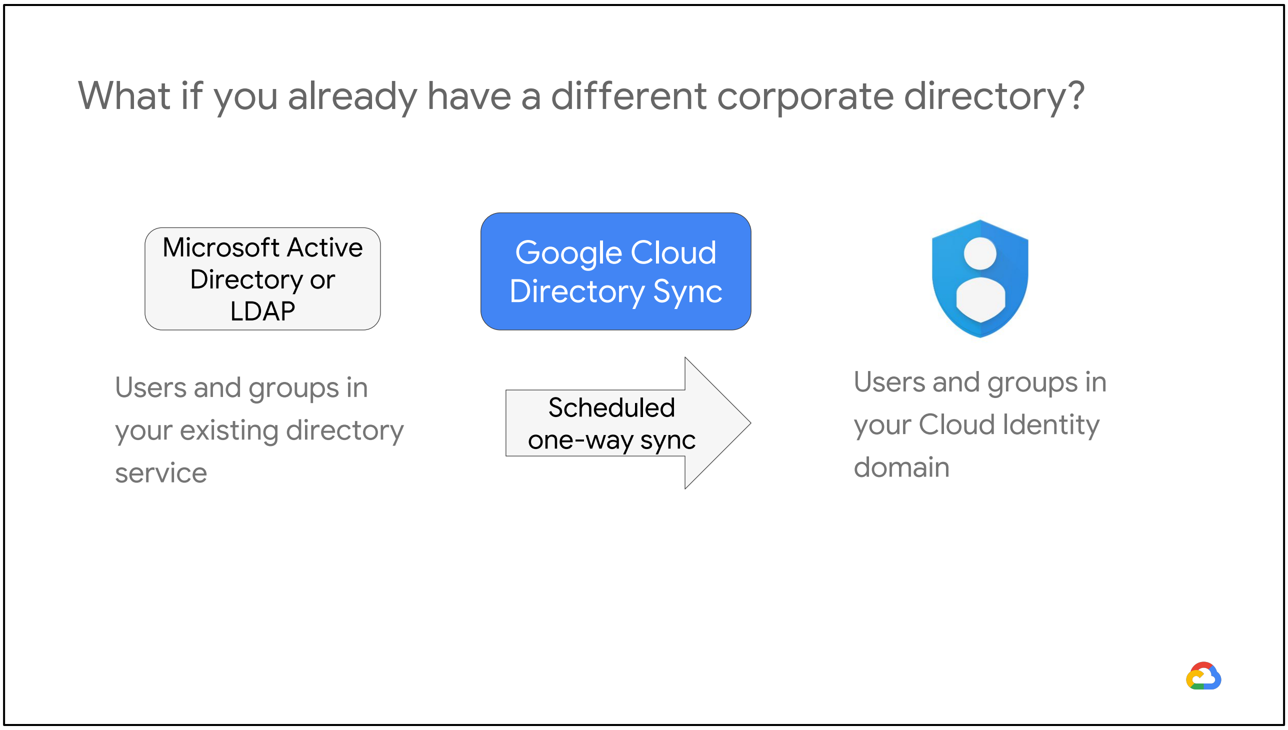
Users and groups in your G Suite domain

Users and groups in your Cloud Identity domain

Many new GCP customers get started by logging into the GCP console with a Gmail account. To collaborate with their teammates, they use Google Groups to gather together people who are in the same role. This approach is easy to get started with, but its disadvantage is that your team's identities are not centrally managed. For example, if someone leaves your organization, there is no centralized way to remove their access to your cloud resources immediately.

GCP customers who are also G Suite customers can define GCP policies in terms of G Suite users and groups. This way, when someone leaves your organization, an administrator can immediately disable their account and remove them from groups using the Google Admin Console.

GCP customers who are not G Suite customers can get these same capabilities through Cloud Identity. Cloud Identity lets you manage users and groups using the Google Admin Console, but you do not pay for or receive G Suite's collaboration products such as Gmail, Docs, Drive, and Calendar. Cloud Identity is available in a free and a premium edition. The premium edition adds capabilities for mobile device management.



What if you already have a different corporate directory?

Microsoft Active Directory or LDAP

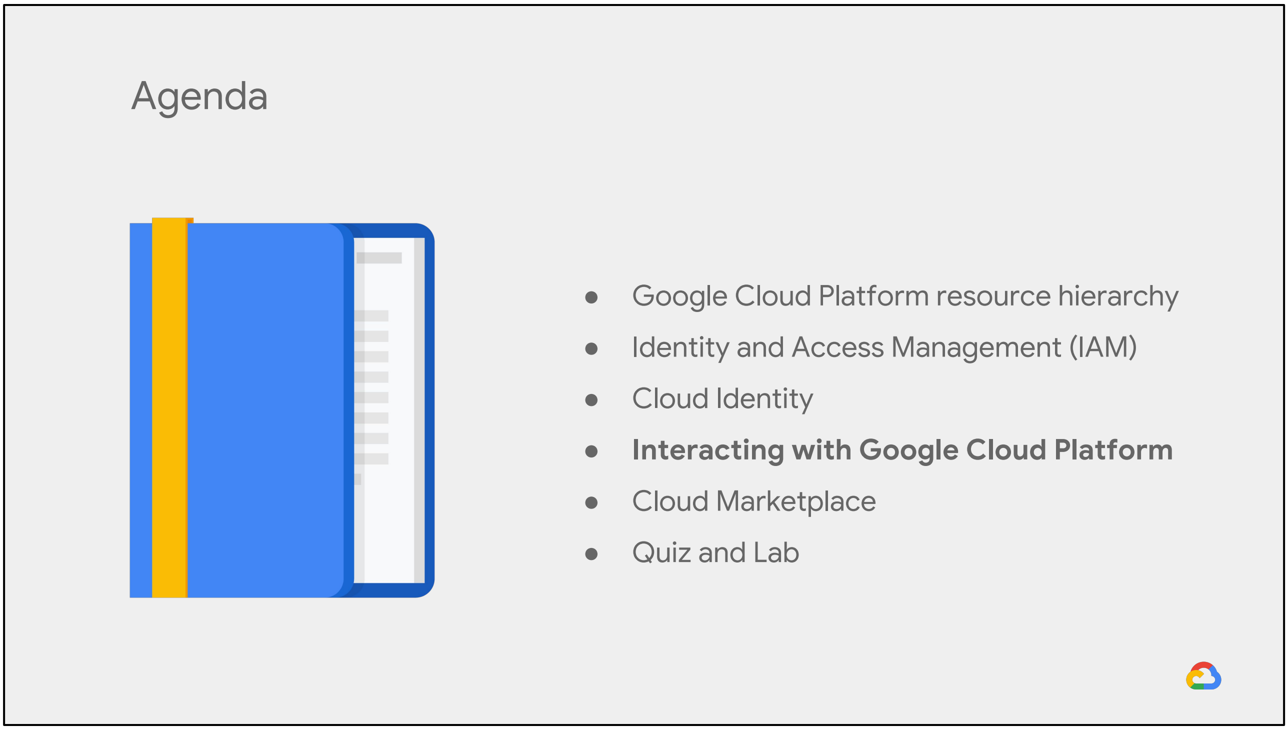
Users and groups in your existing directory service

Google Cloud Directory Sync

Scheduled one-way sync

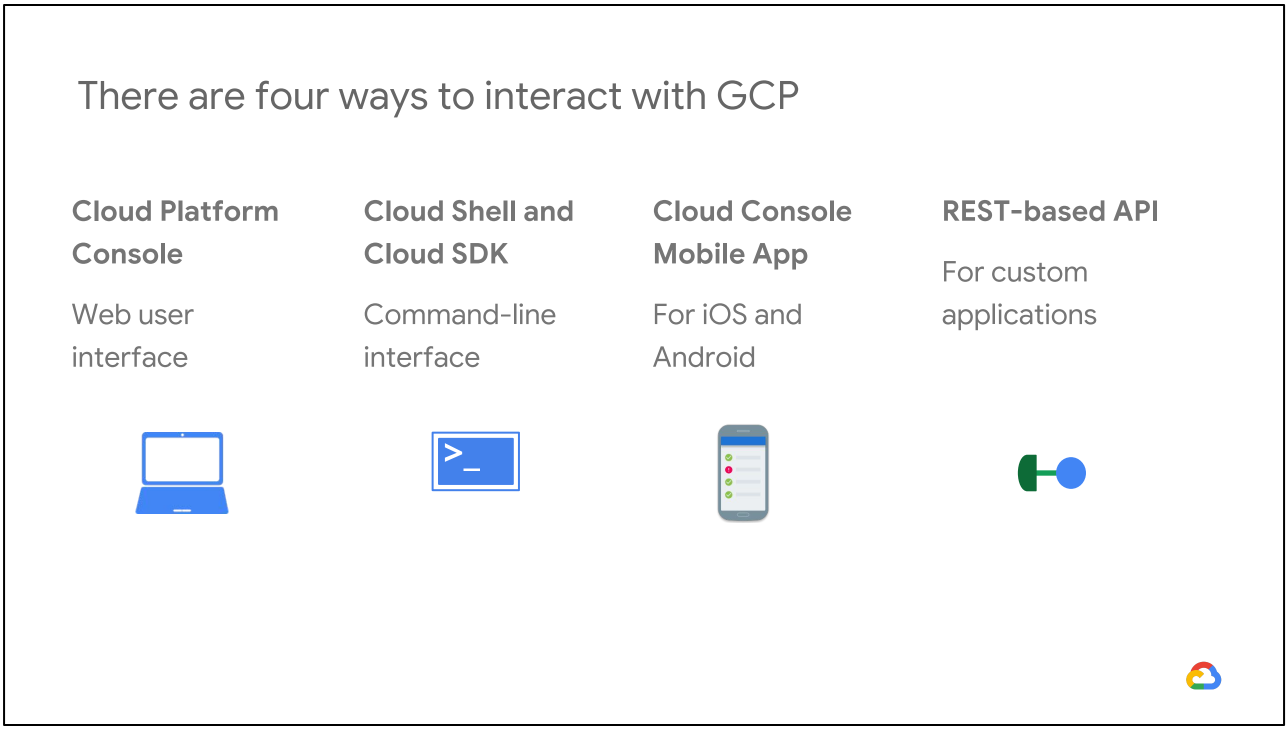
Users and groups in your Cloud Identity domain

Using Google Cloud Directory Sync, your administrators can log in and manage GCP resources using the same usernames and passwords they already use. This tool synchronizes users and groups from your existing Active Directory or LDAP system with the users and groups in your Cloud Identity domain. The synchronization is one-way only; no information in your Active Directory or LDAP map is modified. Google Cloud Directory Sync is designed to run scheduled synchronizations without supervision, after its synchronization rules are set up.



Agenda

- Google Cloud Platform resource hierarchy
- Identity and Access Management (IAM)
- Cloud Identity
- **Interacting with Google Cloud Platform**
- Cloud Marketplace
- Quiz and Lab



There are four ways to interact with GCP

Cloud Platform Console

Web user interface

Cloud Shell and Cloud SDK

Command-line interface

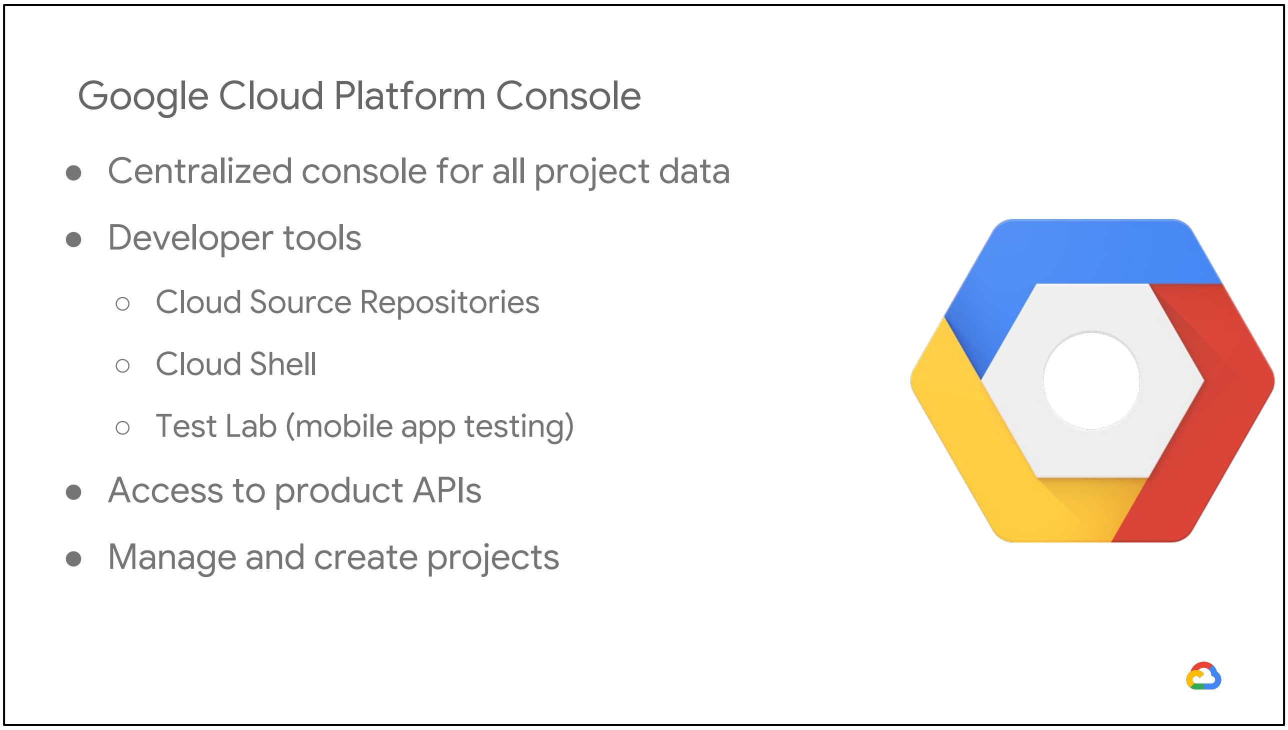
Cloud Console Mobile App

For iOS and Android

REST-based API

For custom applications

There are four ways you can interact with Google Cloud Platform, and we'll talk about each in turn: the Console, the SDK and Cloud Shell, the mobile app, and the APIs.

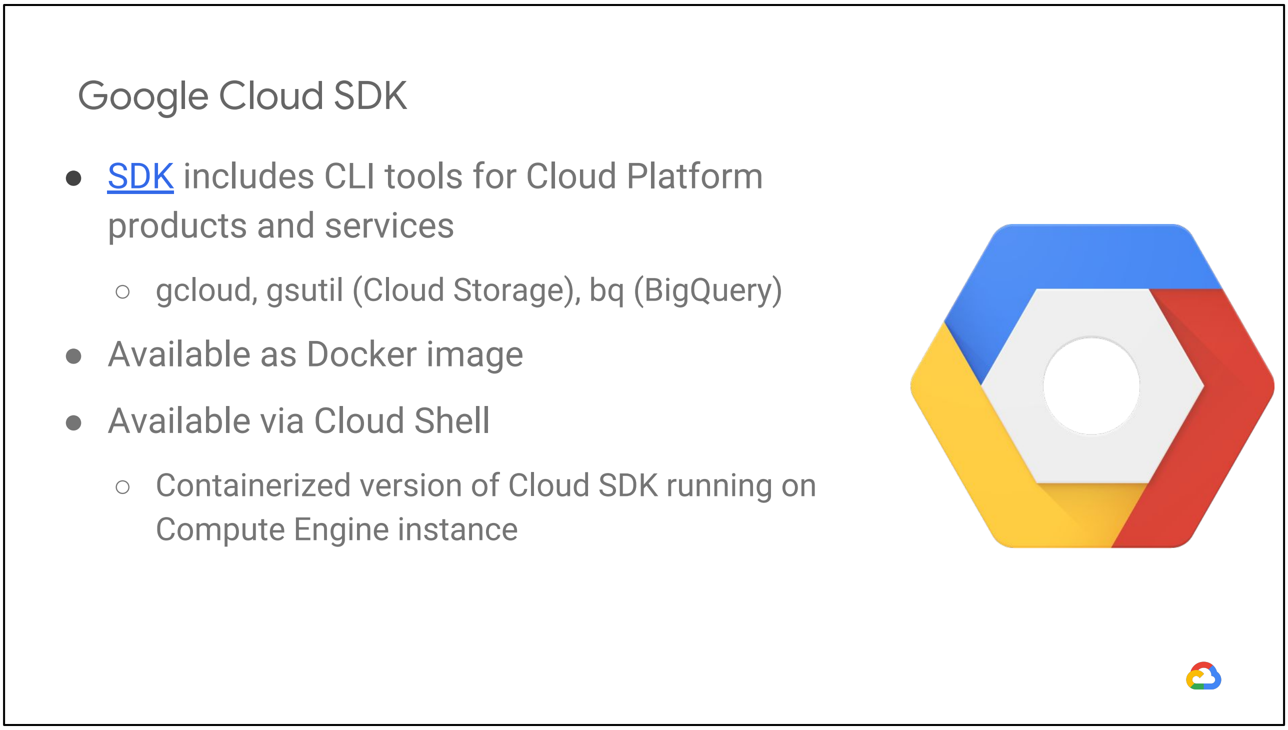


Google Cloud Platform Console

- Centralized console for all project data
- Developer tools
  - Cloud Source Repositories
  - Cloud Shell
  - Test Lab (mobile app testing)
- Access to product APIs
- Manage and create projects

Google Cloud Source Repositories provides Git version control to support collaborative development of any application or service, including those that run on Google App Engine and Google Compute Engine. If you are using the Stackdriver Debugger, you can use Cloud Source Repositories and related tools to view debugging information alongside your code during application runtime. Cloud Source Repositories also provides a source editor that you can use to browse, view, edit, and commit changes to repository files from within the Cloud Platform Console.

Google Cloud Shell provides you with command-line access to your cloud resources directly from your browser. You can easily manage your projects and resources without having to install the Google Cloud SDK or other tools on your system. With Cloud Shell, the Cloud SDK gcloud command and other utilities you need are always available, up to date, and fully authenticated when you need them.



Google Cloud SDK

- SDK includes CLI tools for Cloud Platform products and services
    - gcloud, gsutil (Cloud Storage), bq (BigQuery)
- Available as Docker image
- Available via Cloud Shell
    - Containerized version of Cloud SDK running on Compute Engine instance

The Google Cloud SDK is a set of tools that you can use to manage resources and applications hosted on Google Cloud Platform. These include the gcloud tool, which provides the main command-line interface for Google Cloud Platform products and services, as well as gsutil and bq. All of the tools are located under the bin directory.

For more information on the SDK command-line tools, see: https://cloud.google.com/sdk/cloudplatform

Note: Currently, the App Engine SDKs are separate downloads. For more information, see: https://cloud.google.com/appengine/downloads

Cloud Shell provides the following:

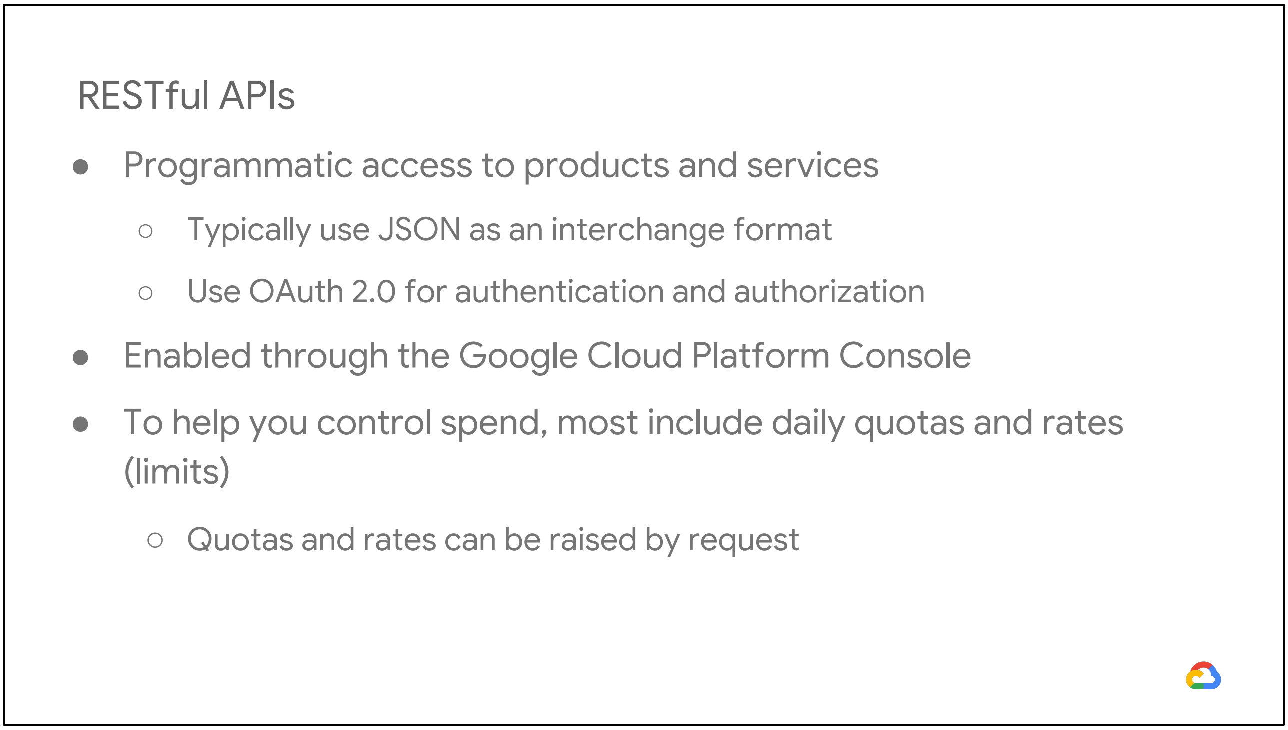
- A temporary Compute Engine virtual machine instance running a Debian-based Linux operating system
- Command-line access to the instance from a web browser using terminal windows in the Cloud Platform Console
- 5 GB of persistent disk storage per user, mounted as your $HOME directory in Cloud Shell sessions across projects and instances
- Google Cloud SDK and other tools pre-installed on the Compute Engine instance

- Language support, including SDKs, libraries, runtime environments and compilers for Java, Go, Python, Node.js, PHP and Ruby
- Web preview functionality, which allows you to preview web applications running on the Cloud Shell instance through a secure proxy
- Built-in authorization for access to projects and resources

You can use Cloud Shell to:

- Create and manage Google Compute Engine instances.
- Create and access Google Cloud SQL databases.
- Manage Google Cloud Storage data.
- Interact with hosted or remote Git repositories, including Google Cloud Source Repositories.
- Build and deploy Google App Engine applications.

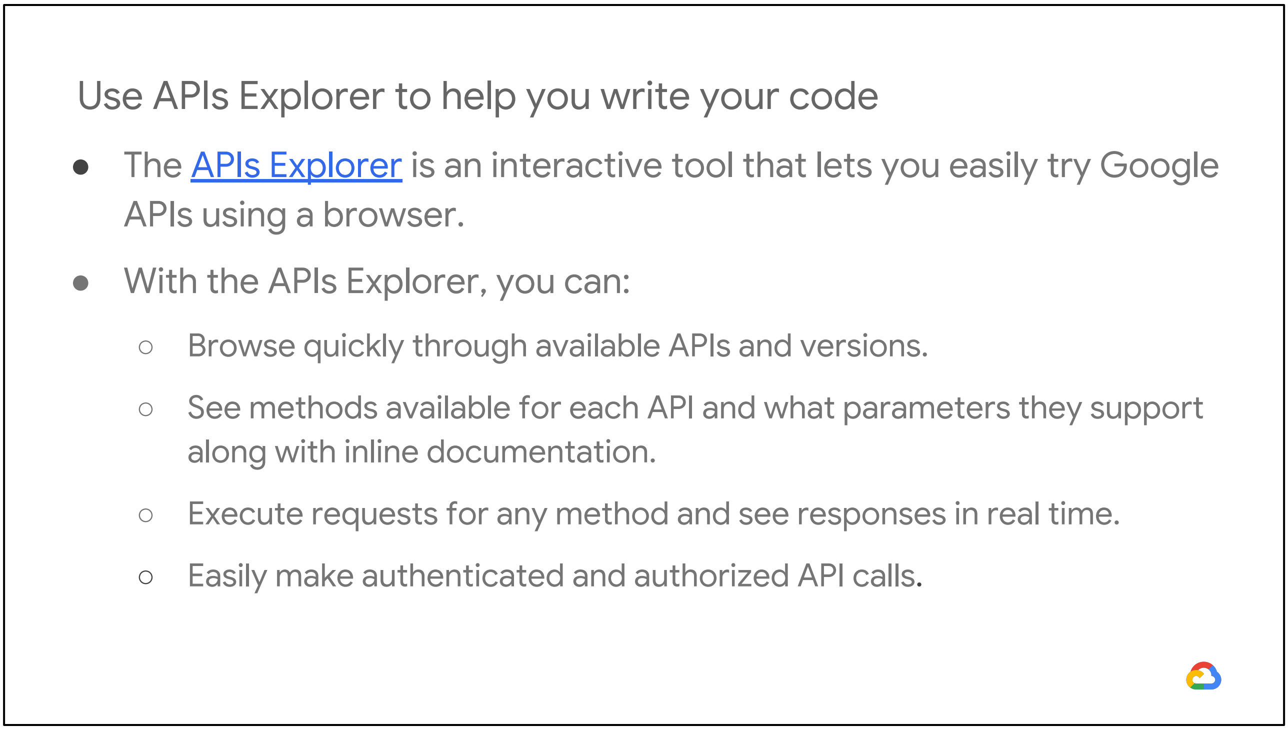
You can also use Cloud Shell to perform other management tasks related to your projects and resources, using either the gcloud command or other available tools.



RESTful APIs

- Programmatic access to products and services
  - Typically use JSON as an interchange format
  - Use OAuth 2.0 for authentication and authorization
- Enabled through the Google Cloud Platform Console
- To help you control spend, most include daily quotas and rates (limits)
  - Quotas and rates can be raised by request

The services that make up GCP offer Application Programming Interfaces, so that code you write can control them. These APIs are what's called "RESTful"; in other words, they follow the "Representational state transfer" paradigm. In a broad sense, that means that your code can use Google services in much the same way that web browsers talk to web servers. The APIs name resources in GCP with URLs. Your code can pass information to the APIs using JSON, which is a very popular way of passing textual information over the Web. And there's an open system, OAuth2, for user login and access control.

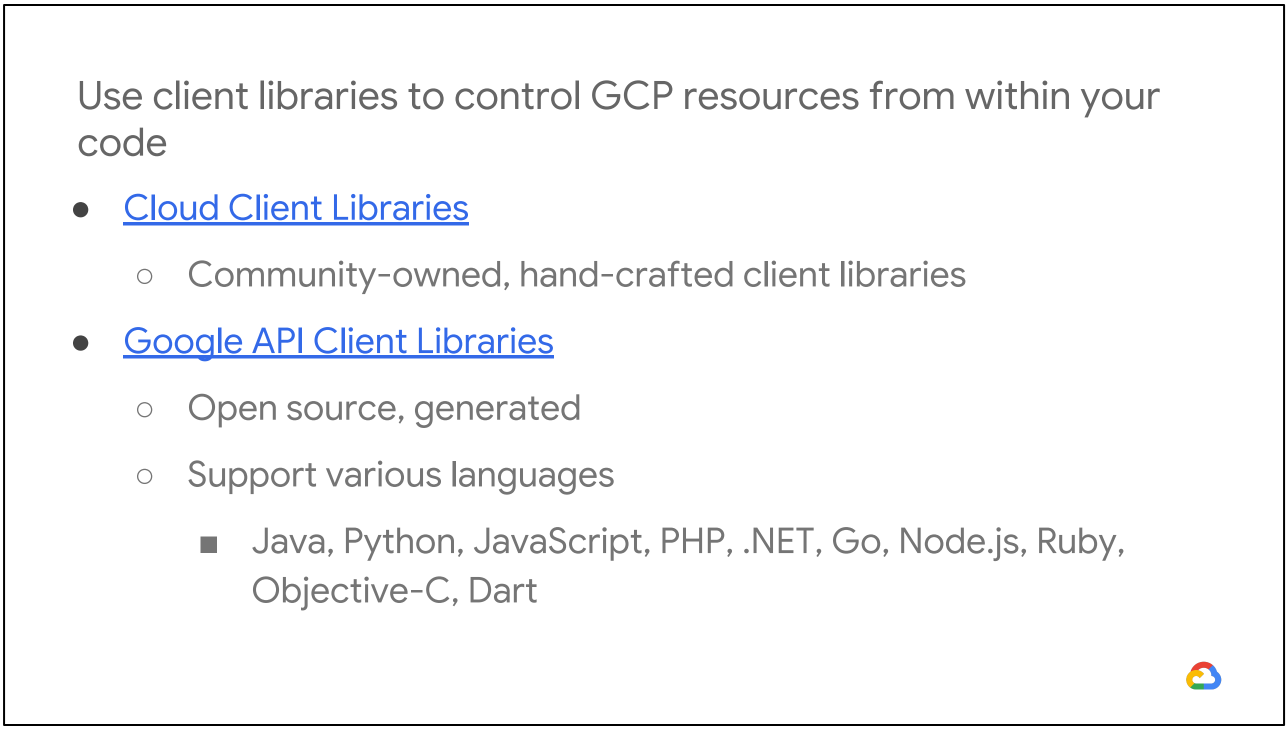


Use APIs Explorer to help you write your code

- The APIs Explorer is an interactive tool that lets you easily try Google APIs using a browser.
- With the APIs Explorer, you can:
    - Browse quickly through available APIs and versions.
    - See methods available for each API and what parameters they support along with inline documentation.
    - Execute requests for any method and see responses in real time.
    - Easily make authenticated and authorized API calls.

The GCP Console includes a tool called the APIs Explorer that helps you learn about the APIs interactively. It lets you see what APIs are available, and in what versions. These APIs expect parameters, and documentation on them is built-in. You can try the APIs interactively, even with user authentication.

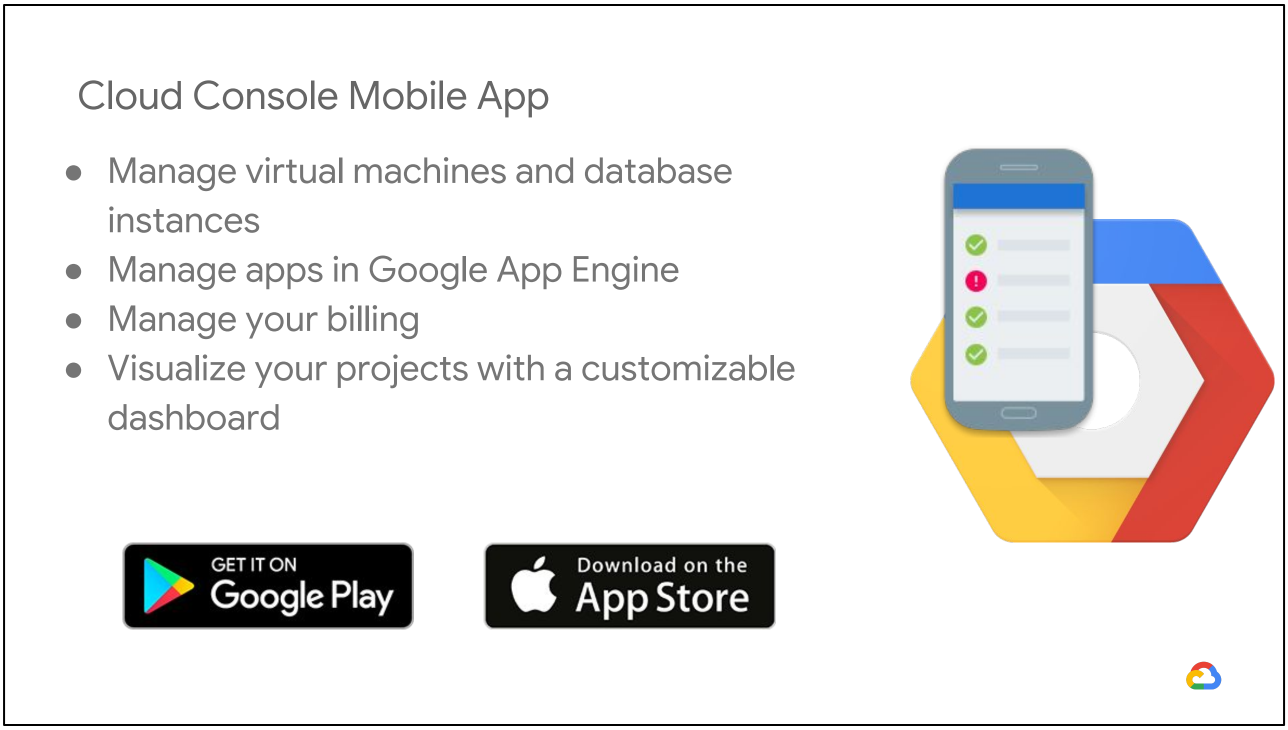
Suppose you've explored an API, and you're ready to build an application that uses it. Do you have to start coding from scratch? No. Google provides client libraries to take a lot of the drudgery out of the task of calling GCP from your code.



Use client libraries to control GCP resources from within your code

- Cloud Client Libraries
  - Community-owned, hand-crafted client libraries
- Google API Client Libraries
  - Open source, generated
  - Support various languages
    - Java, Python, JavaScript, PHP, .NET, Go, Node.js, Ruby, Objective-C, Dart

There are two kinds of libraries. The Cloud Client libraries are Google Cloud's latest and recommended libraries for its APIs. They adopt the native styles and idioms of each language. On the other hand, sometimes a Cloud Client library doesn't support the newest services and features. In that case, you can use the Google API Client library for your desired languages. These libraries are designed for generality and completeness.



Cloud Console Mobile App

- Manage virtual machines and database instances
- Manage apps in Google App Engine
- Manage your billing
- Visualize your projects with a customizable dashboard

GET IT ON Google Play

Download on the App Store

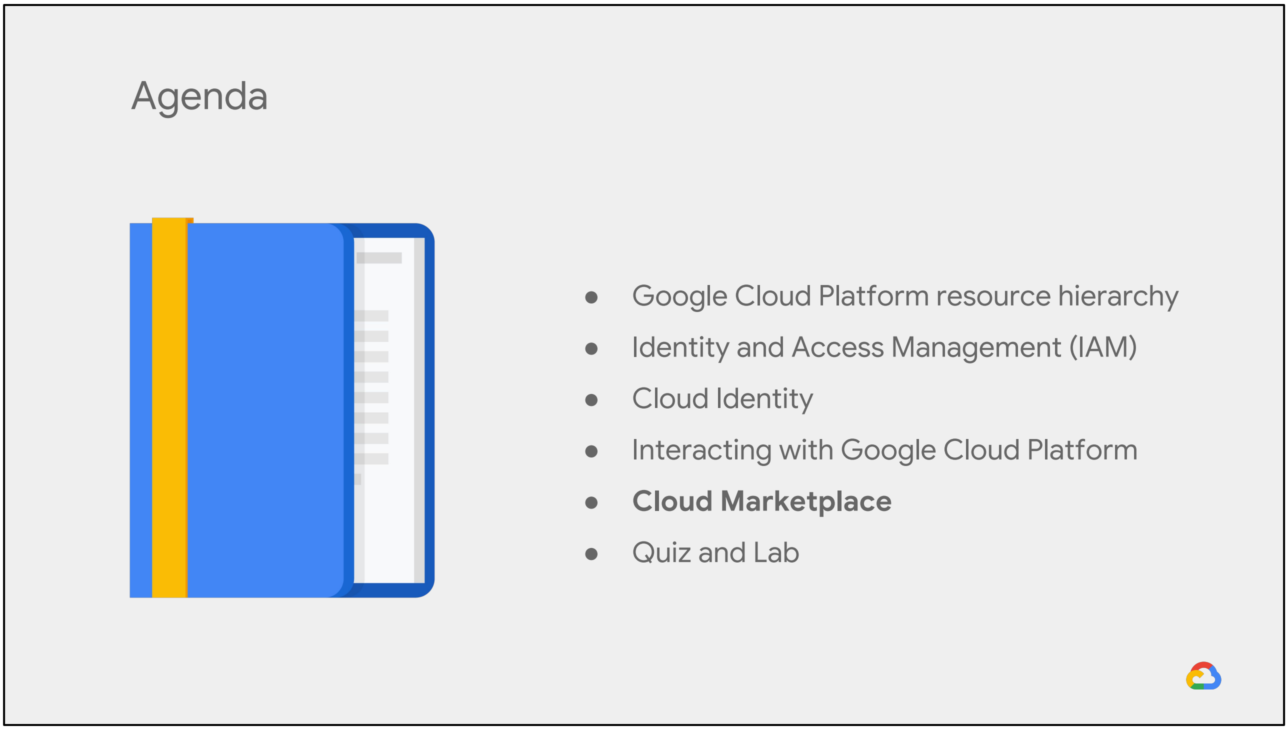
The mobile app allows you to start, stop, and SSH into Compute Engine instances, and to see logs from each instance. You can stop and start Cloud SQL instances. You can also administer applications deployed on Google App Engine, by viewing errors, rolling back deployments, and changing traffic splitting.

You can also get up-to-date billing information for your projects and get billing alerts for projects that are going over budget.

You can set up customizable graphs showing key metrics such as CPU usage, network usage, requests per second, and server errors.

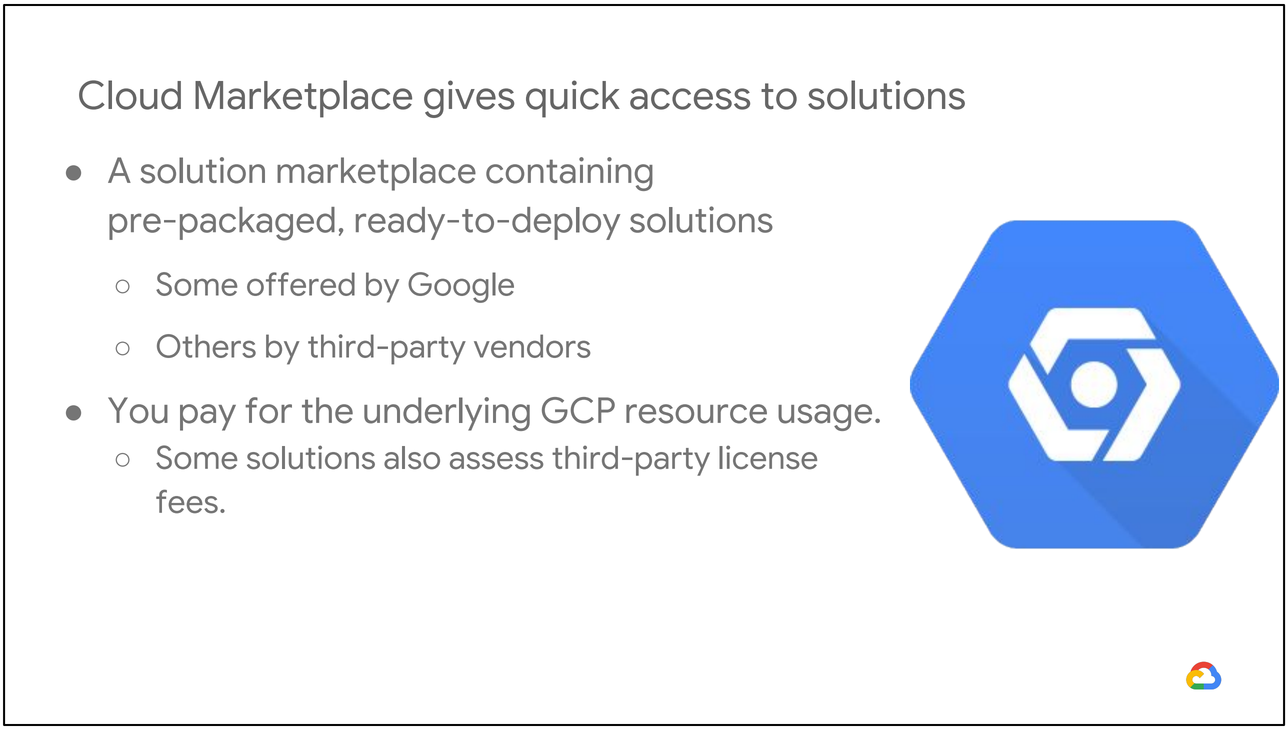
The mobile app also offers alerts and incident management.

Download the Google Cloud Console Mobile App from Google Play or from the iOS App Store.



Agenda

- Google Cloud Platform resource hierarchy
- Identity and Access Management (IAM)
- Cloud Identity
- Interacting with Google Cloud Platform
- **Cloud Marketplace**
- Quiz and Lab



Cloud Marketplace gives quick access to solutions

- A solution marketplace containing pre-packaged, ready-to-deploy solutions
  - Some offered by Google
  - Others by third-party vendors
- You pay for the underlying GCP resource usage.
  - Some solutions also assess third-party license fees.

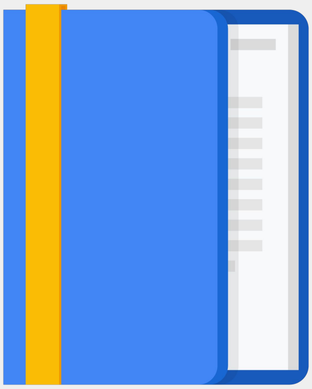
Google Cloud Marketplace lets you quickly deploy functional software packages that run on Google Cloud Platform. You can easily start up a familiar software package without having to manually configure the software, virtual machine instances, storage, or network settings.

Many software packages in Cloud Marketplace are free. The only costs to deploy these solutions are the normal usage fees for Google Cloud Platform resources. Estimated costs are based on the minimum recommended instance and storage configuration. The estimate does not include networking costs. You can modify the instance and storage configuration when you deploy the configuration.

Google Cloud Platform updates the images for these software packages to fix critical issues and vulnerabilities, but doesn't update software that you have already deployed.

Some Cloud Marketplace images assess usage fees, particularly those published by third parties and containing commercially licensed software. If an image does incur a usage fee, the fee appears on your monthly Google Cloud Platform invoice as a separate line item. See the Cloud Marketplace documentation for details.

Agenda



- Google Cloud Platform resource hierarchy
- Identity and Access Management (IAM)
- Cloud Identity
- Interacting with Google Cloud Platform
- Cloud Marketplace
- **Quiz and Lab**

Question #1

True or False: If a Google Cloud IAM policy gives you Owner permissions at the project level, your access to a resource in the project may be restricted by a more restrictive policy on that resource.

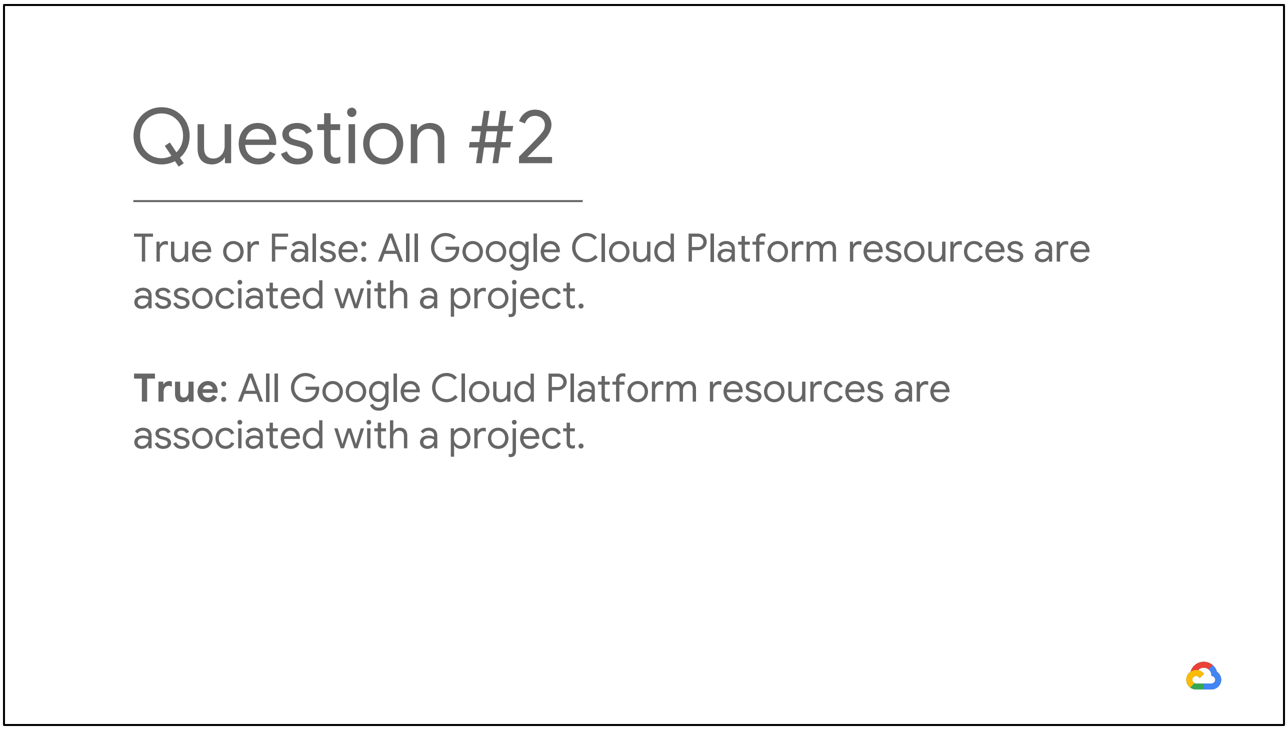
Question #1

True or False: If a Google Cloud IAM policy gives you Owner permissions at the project level, your access to a resource in the project may be restricted by a more restrictive policy on that resource.

False: Policies are a union of the parent and the resource. If a parent policy is less restrictive, it overrides a more restrictive resource policy.

Question #2

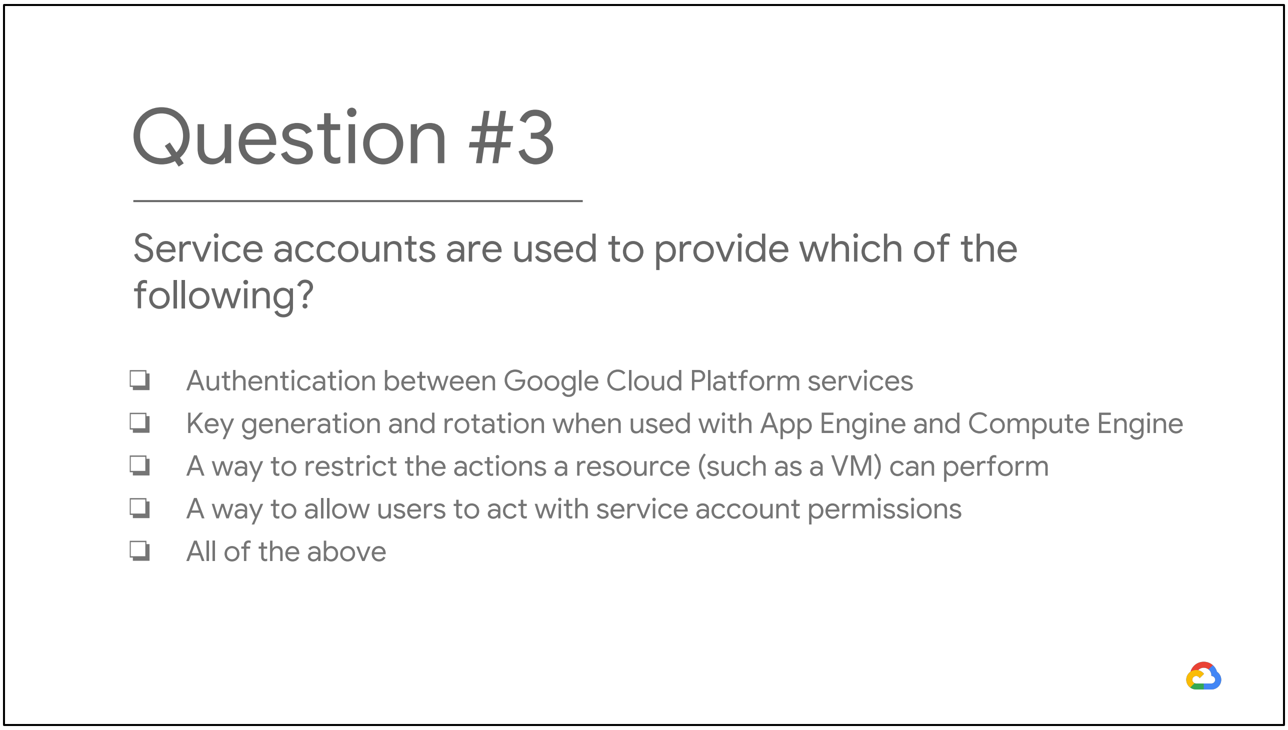
True or False: All Google Cloud Platform resources are associated with a project.



Question #2

True or False: All Google Cloud Platform resources are associated with a project.

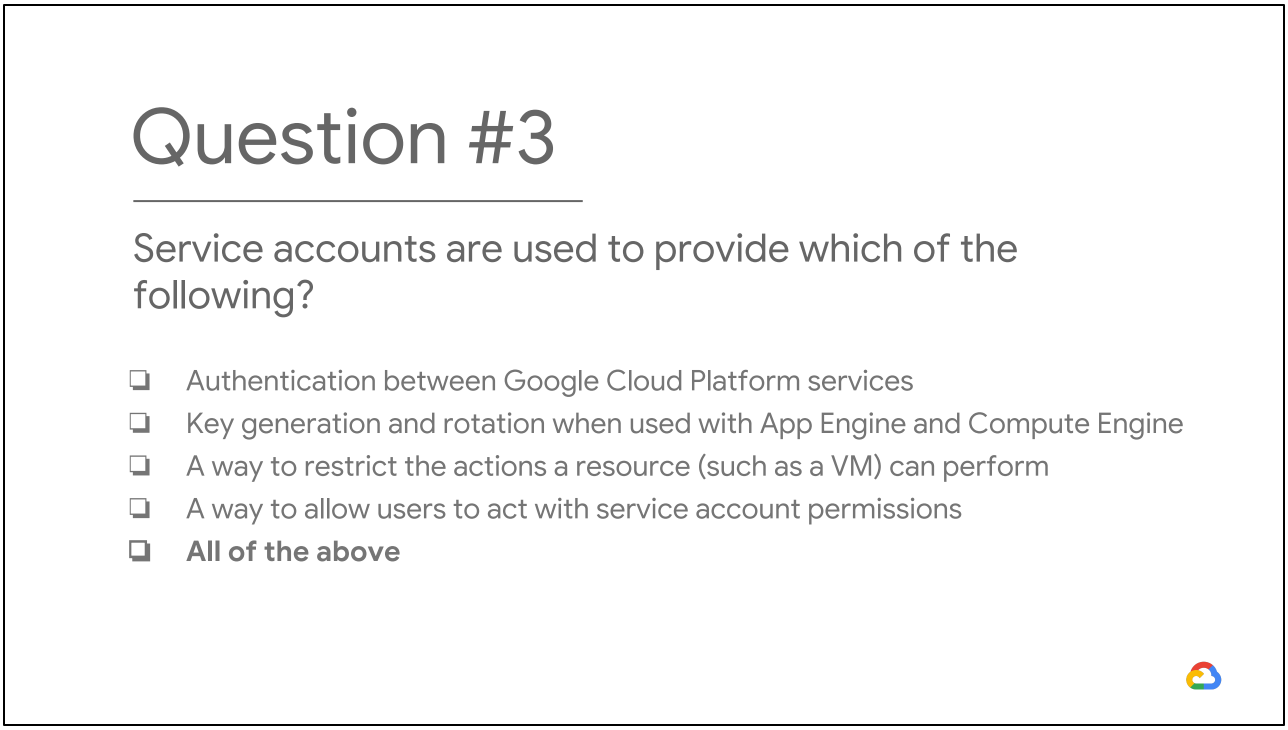
True: All Google Cloud Platform resources are associated with a project.



Question #3

Service accounts are used to provide which of the following?

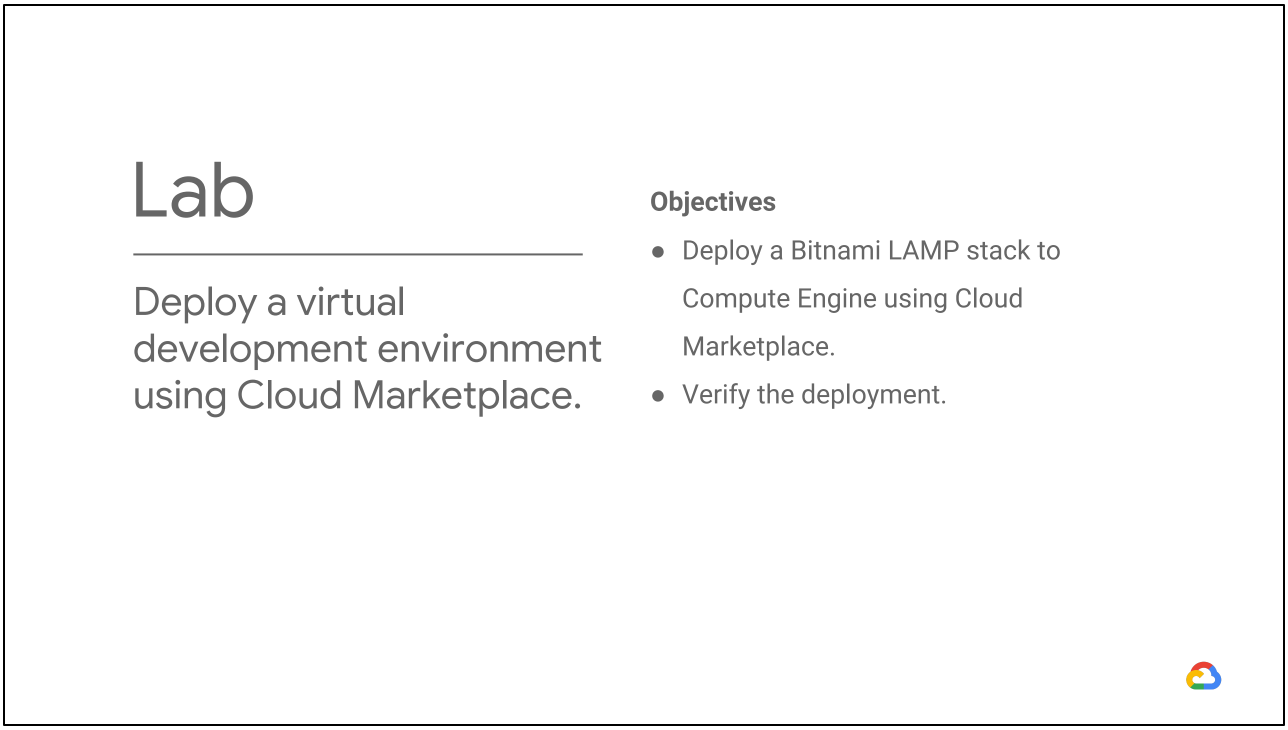
- □ Authentication between Google Cloud Platform services
- □ Key generation and rotation when used with App Engine and Compute Engine
- □ A way to restrict the actions a resource (such as a VM) can perform
- □ A way to allow users to act with service account permissions
- □ All of the above



Question #3

Service accounts are used to provide which of the following?

- ❑ Authentication between Google Cloud Platform services
- ❑ Key generation and rotation when used with App Engine and Compute Engine
- ❑ A way to restrict the actions a resource (such as a VM) can perform
- ❑ A way to allow users to act with service account permissions
- ❑ **All of the above**



Lab

Deploy a virtual development environment using Cloud Marketplace.

Objectives

- Deploy a Bitnami LAMP stack to Compute Engine using Cloud Marketplace.
- Verify the deployment.

In this lab, you create two nginx web servers and control external HTTP access to the web servers using tagged firewall rules. Then, you explore IAM roles and service accounts.

More resources

Google Cloud Platform security https://cloud.google.com/security/

Configuring permissions https://cloud.google.com/docs/permissions-overview

Identity and Access Management (IAM) https://cloud.google.com/iam/

Cloud SDK installation and quick start https://cloud.google.com/sdk/#Quick_Start

`gcloud` tool guide https://cloud.google.com/sdk/gcloud/