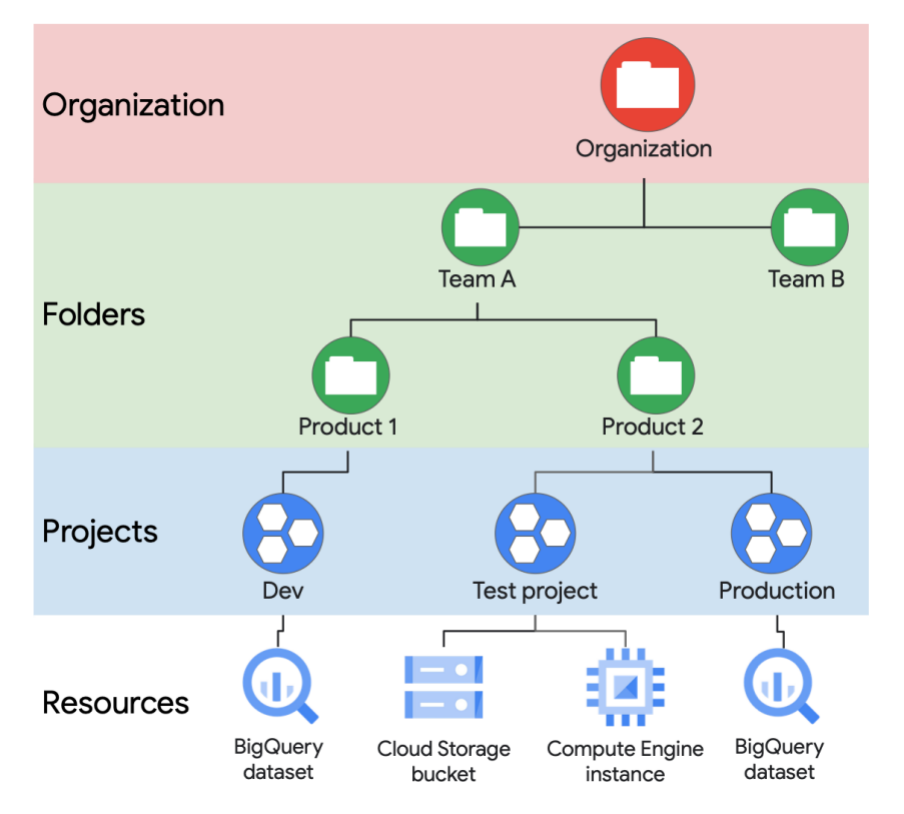
Cloud security premises requires collaboration

* 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.

# Google Cloud RESOURCE HIERARCHY

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



## PROJECT

A project is the base level organising entity for creating and using resources and services and managing billing, API’s and permissions.

Zones and regions physically organise the GCP resources you use, projects and logically organise them. Projects can be easily created, managed, deleted or even recovered from accidental deletion.

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.

## FOLDERS

**Folders** are another logical grouping you can have for collections of projects. Having an organisation is required to use folders, and the organisation is the root node of the entire GCP hierarchy.

Folders offer flexible management

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

## ORGANISATION

Lastly, while not required, an organisation is quite useful because it will allow you set policies that apply throughout the enterprise.

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 |

# IAM Roles and POLICIES

## POLICY

* 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

## ROLES

There are three types of IAM roles: -

1. Primitive
2. Predefined
3. Custom

##### IAM primitive roles

IAM primitive roles apply across all GCP services in a project. IAM primitive roles offer fixed, coarse-grained levels of access.

##### IAM predefined roles

IAM predefined roles apply to a particular GCP service in a project. IAM predefined roles offer more fine-grained permissions on particular services

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

##### IAM custom roles

IAM custom roles let you define a precise set of permissions.

##### Service Accounts

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](mailto:PROJECT_ID@appspot.gserviceaccount.com)

# Cloud Identity

There are four ways to interact with GCP:-

# Virtual Machines

## Virtual Private Cloud (VPC)

Your VPC networks connect your Google Cloud Platform resources to each other and to the internet.

* Each VPC network is contained in a GCP project.
* You can provision Cloud Platform resources, connect them to each other, and isolate them from one another.

Google Virtual Private Cloud networks that you define have global scope. They can have subnets in any GCP region worldwide. Subnets can span the zones that make up a region. You can dynamically increase the size of a subnet in a custom network by expanding the range of IP addresses allocated to it. Doing that doesn’t affect already configured VMs.

## Compute Engine

Compute Engine offers managed virtual machines.

* High CPU, high memory, standard and shared-core machine types.
* Persistent disks
  + Standard, SSD, local SSD
  + Snapshots
* Resize disks with no downtime
* Instance metadata and start-up scripts

Compute Engine offers customer friendly pricing

* Per-second billing, sustained use discounts, committed use discounts
* Preemptible instances
* High throughput to storage at no extra cost
* Custom machine types: Only pay for the hardware you need

# Big Data and Machine Learning in the Cloud

## Cloud Dataproc

Apache Hadoop is an open-source framework for big data. It is based on the MapReduce programming model, which Google invented and published. The MapReduce model, at its simplest, means that one function -- traditionally called the “map” function -- runs in parallel across a massive dataset to produce intermediate results; and another function -- traditionally called the “reduce” function -- builds a final result set based on all those intermediate results. The term “Hadoop” is often used informally to encompass Apache Hadoop itself and related projects, such as Apache Spark, Apache Pig, and Apache Hive.

Cloud Dataproc is managed Hadoop

* Fast, easy, managed way to run Hadoop and Spark/Hive/Pig on GCP
* Create clusters in 90 seconds or less on average.
* Scale clusters up and down even when jobs are running.
* monitor your cluster using Stackdriver
* Use Spark/Spark SQL to quickly perform data mining and analysis.
* Use Spark Machine Learning Libraries (MLlib) to run classification algorithms.

## Cloud Dataflow

Cloud Dataproc is great when you have a dataset of known size, or when you want to manage your cluster size yourself. But what if your data shows up in **realtime**? Or it’s of unpredictable size or rate? That’s where Cloud Dataflow is a particularly good choice.

Cloud Dataflow offers managed data pipelines

* Processes data using Compute Engine instances.
  + Clusters are sized for you
  + Automated scaling, no instance provisioning required
* Write code once and get batch and streaming.
  + Transform-based programming model

### Cloud Dataflow features:

* **Resource Management**   
  Cloud Dataflow fully automates management of required processing resources. No more spinning up instances by hand.
* **On Demand**   
  All resources are provided on demand, enabling you to scale to meet your business needs. No need to buy reserved compute instances.
* **Intelligent Work Scheduling**  
  Automated and optimized work partitioning which can dynamically rebalance lagging work. No more chasing down “hot keys” or pre-processing your input data.
* **Auto Scaling**  
  Horizontal auto scaling of worker resources to meet optimum throughput requirements results in better overall price-to-performance.
* **Unified Programming Model**  
  The Dataflow API enables you to express MapReduce like operations, powerful data windowing, and fine grained correctness control regardless of data source.
* **Open Source**   
  Developers wishing to extend the Dataflow programming model can fork and or submit pull requests on the Java-based Cloud Dataflow SDK. Dataflow pipelines can also run on alternate runtimes like Spark and Flink.
* **Monitoring**   
  Integrated into the Google Cloud Platform Console, Cloud Dataflow provides statistics such as pipeline throughput and lag, as well as consolidated worker log inspection—all in near-real time.
* **Integrated**   
  Integrates with Cloud Storage, Cloud Pub/Sub, Cloud Datastore, Cloud Bigtable, and BigQuery for seamless data processing. And can be extended to interact with others sources and sinks like Apache Kafka and HDFS.
* **Reliable & Consistent Processing**   
  Cloud Dataflow provides built-in support for fault-tolerant execution that is consistent and correct regardless of data size, cluster size, processing pattern or pipeline complexity.

### Why use Cloud Dataflow?

* ETL (extract/transform/load) pipelines to move, filter, enrich, shape data
* Data analysis: batch computation or continuous computation using streaming
* Orchestration: create pipelines that coordinate services, including external services
* Integrates with GCP services like Cloud Storage, Cloud Pub/Sub, BigQuery, and Bigtable ○ Open source Java and Python SDKs

## BigQuery

BigQuery is a fully managed data warehouse. Compute and storage are separated with a terabit network in between.

* Query using SQL syntax (SQL 2011)
* No cluster maintenance is required.
* You only pay for storage and processing used
* Automatic discount for long-term data storage

### BigQuery’s features:

* **Flexible Data Ingestion**

Load your data from Cloud Storage or Cloud Datastore, or stream it into BigQuery at 100,000 rows per second to enable real-time analysis of your data.

* **Global Availability**

You have the option to store your BigQuery data in European locations while continuing to benefit from a fully managed service, now with the option of geographic data control, without low-level cluster maintenance.

* **Security and Permissions**

You have full control over who has access to the data stored in BigQuery. If you share datasets, doing so will not impact your cost or performance; those you share with pay for their own queries.

* **Cost Controls**

BigQuery provides cost control mechanisms that enable you to cap your daily costs at an amount that you choose. For more information, see Cost Controls.

* **Highly Available**

Transparent data replication in multiple geographies means that your data is available and durable even in the case of extreme failure modes.

* **Super Fast Performance**

Run super-fast SQL queries against multiple terabytes of data in seconds, using the processing power of Google's infrastructure.

* **Fully Integrated**

In addition to SQL queries, you can easily read and write data in BigQuery via Cloud Dataflow, Spark, and Hadoop.

* **Connect with Google Products**

You can automatically export your data from Google Analytics Premium into BigQuery and analyze datasets stored in Google Cloud Storage, Google Drive, and Google Sheets.

# Cloud Pub/Sub

Cloud Pub/Sub is a fully managed real-time messaging service that allows you to send and receive messages between independent applications. Cloud Pub/Sub is designed to provide “at least once” delivery at low latency with on-demand scalability to 1 million messages per second (and beyond).

Cloud Pub/Sub is scalable, reliable messaging system.

* Supports many-to-many asynchronous messaging
  + Application components make push/pull subscriptions to topics
* Includes support for offline consumers
* Based on proven Google technologies
* Integrates with Cloud Dataflow for data processing pipelines

##### Cloud Pub/Sub features:

* **Highly Scalable**

Any customer can send up to 10,000 messages per second, by default—and millions per second and beyond, upon request.

* **Push and Pull Delivery**

Subscribers have flexible delivery options, whether they are accessible from the internet or behind a firewall.

* **Encryption**

Encryption of all message data on the wire and at rest provides data security and protection.

* **Replicated Storage**

Designed to provide “at least once” message delivery by storing every message on multiple servers in multiple zones.

* **Message Queue**

Build a highly scalable queue of messages using a single topic and subscription to support a one-to-one communication pattern.

* **End-to-End Acknowledgement**

Building reliable applications is easier with explicit application-level acknowledgements.

* **Fan-out**

Publish messages to a topic once, and multiple subscribers receive copies to support one-to-many or many-to-many communication patterns.

* **REST API**

Simple, stateless interface using JSON messages with API libraries in many programming languages.