Google Cloud Fundamentals: Core Infraestructure

jueves, 15 de febrero de 2024 18:27

Computacion en la nube: Forma de usar las tecnologías de la información con 5 importantes características.
1. On demand y self service
2. Acceso a los recursos mediante internet.
3. El proveedor asigna recursos de cómputo o demás a los usuarios.
4. Recursos elásticos: escalamiento y de escalamiento de una manera rápida y sencilla.
5. Los usuarios pagan por lo que usan.

Tres olas en la nube:
1. Colocación: Se retan datacenters
2. Virtualización: Los datacenters tienen software virtualizado lo que permite más configuración por parte del usuario.
3. Contenerizacion: Las aplicaciones son manejadas por contenedores que automáticamente configuran y aprovisionan la infraestructura.

IaaS
- Computo crudo.
- Almacenamiento.
- Capacidades de red.
- Pagar por lo que se asigna

PaaS
- Código para dar acceso a la infraestructura que la aplicación necesita.
- Pagar por lo que se usa.

Serverless: Enfocarse en el código y no en la infraestructura, al eliminar la necesidad del manejo de la misma (la infra).
{
Cloud functions: Manejar código orientado por eventos.
Cloud Run: Desplegar una aplicación contenerizada basada en microservicios en un entorno totalmente manejado por google.
}
SaaS: Corren en la nube y son consumidas por medio de Internet por los usuarios.

La ubicación donde se ejecutan las aplicaciones afectan la disponibilidad, durabilidad y latencia (tiempo que demora un paquete en viajar de una fuente a un destino) de las mismas.

Regiones: Áreas geográficas independientes que se componen de zonas.
Zona: Área en la que los servicios de Google Cloud son desplegados.

Seguridad:
- Encriptamiento de la comunicación entre servicios.
- Los servicios usan el protocolo RPC para hacer llamadas entre sí.
- Identidad de usuario.
- Encriptación en reposo: Para bases de datos y storage se encriptan los datos una vez están dentro de lo servicios.
- Protección ante ataques de Denial Of Service

Quotas: Protegen ante la posibilidad de consumir más plata de la que se puede en GCP; las cuotas se aplican a nivel de proyecto.
Quotas de asignación: Se restablecen después de un tiempo determinado.
Quotas de frecuencia: Controlan la cantidad de recursos que puedes tener en tu proyecto. Se puede solicitar más al equipo de Google.

Jerarquías de recursos de Google Cloud:

4. Recursos: Cualquier elemento de Google Cloud. Cada uno de los recursos pertenece a un solo proyecto. Se puede asignar roles para personas encargadas de establecer políticas a nivel de organización y personas con la capacidad de crear proyectos.
3. Proyectos: Los recursos se organizan en proyectos. Desde este nivel se pueden habilitar el uso de los recursos de GCP.
Atributos únicos de cada proyecto: ID (único a nivel global y escogido por el cliente, inmutable), name y number.
2. Carpetas: Los proyectos se pueden organizar en carpetas o sub-carpetas. Permiten establecer políticas a los recursos con el nivel de granularidad que se desee. Da la posibilidad a los equipos de delegar los permisos de administración para que trabajen de forma autónoma.
1. Organización: Abarca todas las carpetas, proyectos y recursos dentro de la organización.

Políticas: Se pueden definir a nivel de proyecto, carpeta u organización. Las políticas se heredan hacia abajo, por ende, si se aplica una política a nivel de organización, las carpetas, proyectos y algunos recursos la aplicaran.

IAM:
Políticas que definen:
Quien (cuenta de Google, grupo de Google, cuenta de servicio o dominio de cloud identity)
puede hacer que (Rol: colección de permisos),
en que recurso.

Tres (3) tipos de roles:
Básico: Cuando son aplicados a un proyecto afectan a todos los recursos en el.
Owner: Pueden acceder, realizar cambios a los recursos, configurar la facturación y configurar y asignar permisos.
Editor: Pueden acceder y realizar cambios a los recursos,
Viewer: Pueden acceder a recursos pero no realizar cambios.
Billing Admin: Puede ver los recursos y controlar la facturación.
(!) Si se trabaja en un proyecto con datos sensibles, los roles básicos puede que sean demasiado amplios.
Predefinido: Roles administrados por Google con cierta cantidad de permisos.
Custom: No administrados por Google, solo pueden ser aplicados a nivel de proyecto u organización.

(!) Usar el principio de privilegios mínimos, donde cada persona se le dan los privilegios necesarios para realizar su trabajo.

Cuentas de servicio:
- Requieren ser administradas.
- Se nombran por medio de una dirección de correo, en lugar de contraseñas usan claves criptográficas para acceder a los recursos.
- Puede tener políticas de IAM asociadas a ella.

Formas de acceder a GCP:
- Google Cloud Console: Web
- Cloud SDK and Shell: Terminal
- APIs: Google APIs Explorer muestra que API's estan disponibles y en que versiones.
- Cloud console: Mobile App.

VPC (Virtual Private Cloud): Modelo privado de computación en la nube alojado dentro de una nube pública.

Combinan la escalabilidad y conveniencia de la computación en la nube, con el aislamiento de las aplicaciones de la nube privada.
El tamaño de una sub red se puede expandir al expandir el número de direcciones IP asociadas a ella.
Las subredes tienen alcance regional en la VPC de Google.
Diferentes recursos en diferentes regiones pueden compartir la misma subred.

No requiere de manejar la tabla de enrutamiento
No requiere de administrar el firewall, las reglas se pueden definir por medio de etiquetas.

Con VPC peering se puede realizar intercambio entre 2 VPC
VPC compartida para controlar accesos de IAM.

Compute Engine: Crear y correr máquinas virtuales.
Pueden ser configuradas como un servidor físico.
Por cada máquina que corra más del 25% del mes, se da descuento por cada minuto.
Commited – use: Una cantidad especifica de VCPU y memoria se puede comprar con hasta el 57% de descuento, compromiso de uso de 1 a 3 años.
Spot: Las máquinas virtuales de Spot son maquinas que por ahorrar aproximadamente el 90% del precio Google tiene el permiso de detenerlas en caso de necesitar recursos.

Cloud Load Balancing se encarga de ser el balanceador de carga en el momento en que las máquinas virtuales escalen verticalmente.
Provee balanceo de las cargar en múltiples regiones, así como solo en una.

Cloud DNS: Resolucion de nombres de las aplicaciones.

Cloud CDN: Sistema de chache en la frontera (edge) que permite almacenar contenido más cerca de los usuarios finales.

Para conectarse a una red externa desde GCP hay varias opciones

IPsec VPN protocol: Inicia una conexión por internet y se trata de conectar a la VPN, se usa Cloud Router para que la conexión sea dinámica mediante el protocolo Border GateWay Protocol.

Intercambio de tráfico directo: Poner un router en el mismo datacenter público que un punto de presencia de Google.

Programa de intercambio de tráfico: Acceso a la red de Google por medio de la red de un ISP (Internet service provider), no está cubierto por Google SLA (Service Level Agreement).

Interconexión dedicada: Garantiza tiempo de conectividad altos, permite una o más conexiones privadas con Google, pueden ser soportadas por una VPN.

Interconexión por medio de un tercero (partner): Util si no se puede alcanzar un datacenter por medio de la interconexión dedicada o si las necesidades de tráfico no son más de 10 gigabytes por segundo.
Aplicaciones que pueden tolerar cierto tiempo de baja.
Google no es responsable de ningún problema de conectividad por fuera de la red de Google.

Servicios de almacenamiento:

- Cloud Storage: Object Storage, maneja los datos como objetos. Contiene el binario de los datos reales y sus metadatos asociados. Se almacenan BLOBs (Binary Large Objects) en el.
  Los BLOB's se almacenan en buckets, los cuales necesitan de un nombre único a nivel global.

La ubicación ideal de un bucket es donde la latencia sea minima.
Se puede habilitar el versionamiento de un objeto en un bucket.

Darle acceso a los usuarios unicamente a los recursos que estos necesiten para hacer su trabajo.

IAM, los roles son heredados del proyecto al bucket al objeto.
ACL's en caso de necesitar permisos mas espcificos, las ACL's se componen de definir un alcance y un permiso (que acciones se pueden ejecutar), quien tiene acceso y puede realizar una accion.

Cloud Storage ofrece políticas de ciclo de vida para borrar ciertos objetos dependiendo de su edad en el sistema.

Son inmutables, una nueva version del objeto se crea por cada cambio efectuado en este.

- File Storage: Maneja los datos como una jerarquía de archivos
- Block Storage: Maneja los datos como fragmentos de disco.

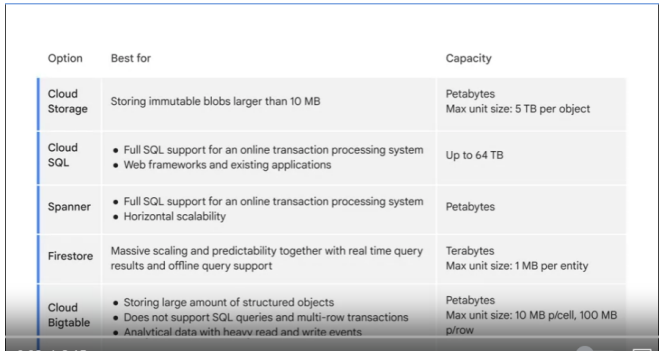
Cloud SQL: Bases de datos relacionales completamente administradas, MySQL, PostgresSQL y SQL Server como servicio.
El costo de una instancia cubre 7 backups
Puede escalar hasta 64 procesadores y 30TB de almacenamiento.
Encripta los datos en tránsito (en la VPC de Google) y en reposo (tablas, archivos temporales y backups).

Cloud Spanner: Servicio de base de datos relacional, el cual escala horizontalmente.
Soporta el manejo de índices secundarios.

Firestore:
Base de datos NOSQL escalable horizontalmente, los datos se almacenan en documentos y se organizan mediante colecciones.
El performance del query depende del tamaño del conjunto de respuesta, no de todo el dataset.
Alamcena en cache los datos de alto uso por la aplicación, para que inclusive sin conexión la aplicación pueda hacer consultas a la base de datos.
Cuando el dispositivo vuelve a tener conexión Firestore sincroniza los cambios realizados.
Se cobra por cada documento, que se lea, escribe o borre. Los queries se cobran bajo la tarifa de un documento leído por query.

Cloud BigTable: Servicio NOSQL para BigData, maneja grandes cantidades de datos con una baja latencia.
Mejor en caso de que:

- Se esta trabajando con mas de 1TB de datos semiestructurados o estructurados.
- Los datos cambian rápidamente y requieren de alto rendimiento.
- Datos NOSQL, donde no se requiere de semántica relacional potente.
- Batch o Datos real time.
- Se están ejecutando algoritmos de ML con los datos.



Option	Best for	Capacity
Cloud Storage	Storing immutable blobs larger than 10 MB	Petabytes Max unit size: 5 TB per object
Cloud SQL	• Full SQL support for an online transaction processing system • Web frameworks and existing applications	Up to 64 TB
Spanner	• Full SQL support for an online transaction processing system • Horizontal scalability	Petabytes
Firestore	Massive scaling and predictability together with real time query results and offline query support	Terabytes Max unit size: 1 MB per entity
Cloud Bigtable	• Storing large amount of structured objects • Does not support SQL queries and multi-row transactions • Analytical data with heavy read and write events	Petabytes Max unit size: 10 MB p/cell, 100 MB p/row

The best performance is achieved by placing the client and the database close to each other.

Contenedor: Caja invisible alrededor del codigo y sus dependencias, la cual posee acceso limitado a su sistema de archivos de su propia partición y hardware.
El SO está siendo virtualizado, escala como PaaS pero de la flexibilidad como IaaS.

Kubernetes: Producto para administrar y escalar aplicaciones alojadas en contenedores.
Facilita el orquestamiento de muchos contenedores en muchos hosts, su escalamiento, y la implementación de lanzamientos y reversiones.
Un nodo = una instancia de procesamiento (una máquina).
Se describe el conjunto de aplicaciones y como interactúan entre ellas y Kubernetes decide como hacer que eso suceda.
Pod: Unidad mínima en Kubernetes que se puede crear o desplegar. Representa un proceso de ejecución en su cluster (como componente o toda la aplicación).
Generalmente solo se tiene un contenedor por Pod, en caso de tener una dependencia alta con otros contenedores se pueden meter en un solo Pod los contenedores.
Proporciona una IP de red única y una serie de puertos para los contenedores, asi como opciones que determinan como debería de correr el contenedor.

Despliegue: representa un grupo de réplicas del mismo pod y mantiene los pods en ejecución incluso si fallan los nodos donde se ejecutan
Kubernetes crea un servicio con una IP fija para los pods.

Servicio: Abstraccion que define un conjunto logico de pods y una politica para acceder a ellos

---

Cloud Run: Plataforma de cómputo administrada que corre contenedores sin estado por medio de solicitudes web o eventos Pub/Sub.
Es ServerLess
Puede aumentar o reducir la escala automáticamente casi de inmediato, solo cobra por los recursos usados.
Pasos contenedores:
1. Escribir la aplicación
2. Compilar y empaquetar la aplicación dentro de una imagen de un contenedor.
3. La imagen se envía a Artifact Registry para que Cloud Run pueda correrla.

Una vez desplegada se obtiene una dirección única HTTPS.

Pasos código:
1. Escribir la aplicación
2. Cloud run compila y empaqueta la aplicación dentro de una imagen de un contenedor usando BuildPacks.
3. Solo se debe de preocupar de manejar las requests, Cloud Run maneja la encriptacion.

Solo se paga por los recursos que usa un contenedor al estar respondiendo peticiones.

Cloud Functions: Escribir una funcion de unico proposito y se ejecute automaticamente cuando un evento suceda. Solución asíncrona. Se puede usar la invocación HTTP para sincronismo.