ASSOCIATE CLOUD ENGINEER · CHEAT SHEET · CERTIFYDECK

Google Cloud Certified Associate Cloud Engineer

Dominios, pesos oficiales y habilidades clave del temario de Associate Cloud Engineer, para repasar antes del examen.

Nivel: Intermediate (Associate) Duración: 120 min Fuente: guía de estudio oficial de Microsoft Learn

1. Setting up a cloud solution environment

20–20%

Setting up cloud projects and accounts

- La jerarquía de recursos de Google Cloud va de la organización (raíz, que exige Cloud Identity o Google Workspace) a carpetas opcionales y proyectos; cada recurso tiene un único padre. Los roles de IAM y las políticas de organización concedidos en un nivel se heredan hacia abajo, por lo que las carpetas sirven para agrupar departamentos o entornos y delegar su administración. Las carpetas se crean con gcloud resource-manager folders create.
- Las políticas de organización limitan qué configuraciones se permiten (ubicaciones, claves de cuentas de servicio, acceso público a Cloud Storage), mientras que IAM decide quién actúa. Se establecen en la organización, carpetas o proyectos y se heredan; en reglas de lista prevalecen los valores denegados y las reglas booleanas del hijo sustituyen a las del padre. La mayoría no son retroactivas.
- Para dar acceso en un proyecto se concede un rol a un principal (usuario, grupo o cuenta de servicio) con gcloud projects add-iam-policy-binding, que añade una vinculación sin sobrescribir la política. Se recomiendan roles predefinidos o personalizados en vez de básicos y conceder a grupos. Las condiciones de IAM permiten accesos que caducan, pero no con roles básicos heredados.
- Cloud Identity gestiona de forma centralizada usuarios y grupos gestionados por la empresa. Se pueden crear a mano en la consola o automatizar con Google Cloud Directory Sync desde Active Directory (sincronización unidireccional, sin contraseñas) y SSO SAML, o con gcloud identity groups. Las unidades organizativas aplican ajustes como la verificación en dos pasos; el acceso se gestiona con grupos.
- Cada servicio de Google Cloud debe activarse en el proyecto antes de usarlo, desde la biblioteca de API de la consola o con gcloud services enable, que admite varios servicios a la vez; gcloud services list muestra los activados o disponibles. Hace falta el rol Service Usage Admin. Desactivar una API no borra los datos de Cloud Storage o BigQuery, y --force desactiva también los servicios dependientes.

Managing billing configuration

- Una cuenta de facturación de Cloud define quién paga los proyectos vinculados, funciona en una sola moneda y se asocia a un perfil de pagos de Google. En una organización hace falta el rol Billing Account Creator para crearla. El país y la moneda no se pueden cambiar, así que otro país exige otra cuenta. Un perfil de pagos de tipo Organización admite varios usuarios; el Individual, no.
- Un proyecto solo puede usar servicios de Google Cloud si está vinculado a una cuenta de facturación de Cloud activa; cada proyecto se vincula a una sola cuenta, que puede pagar muchos proyectos. Se vincula o se cambia con gcloud billing projects link, con Billing Account User en la cuenta y Project Billing Manager o Project Owner en el proyecto. Deshabilitar la facturación detiene los servicios y puede eliminar recursos.
- Los presupuestos de Cloud Billing comparan el gasto real o previsto con un importe y envían alertas al cruzar umbrales; pueden abarcar la cuenta, proyectos, carpetas, servicios o una etiqueta. Por defecto avisan a Billing Account Administrator y User; con canales de Cloud Monitoring se añaden otros destinatarios. Un presupuesto solo de alertas no limita el gasto: para automatizar respuestas se usan notificaciones a Pub/Sub.
- Facturación de Cloud exporta a BigQuery costes de uso estándar, costes detallados por recurso, precios y metadatos de CUD; cada tipo se habilita por separado. Se recomienda un proyecto de FinOps vinculado a la misma cuenta, un conjunto de datos multirregional (añade el mes anterior) sin caducidad de tablas, y el rol Billing Account Costs Manager o Administrator; los precios exigen la API BigQuery Data Transfer Service.

2. Planning and configuring a cloud solution

17–18%

Planning and configuring compute resources

- Compute Engine es IaaS con VMs autogestionadas y control total del sistema operativo. GKE ejecuta cargas en Kubernetes; en Autopilot Google gestiona los nodos y pagas por lo que piden los Pods. Cloud Run es serverless: los servicios atienden HTTP y escalan a cero, los jobs se ejecutan hasta terminar, los worker pools procesan colas y las funciones de Cloud Run responden a eventos.
- Las Spot VMs (--provisioning-model=SPOT) usan capacidad sobrante a gran descuento, pero pueden expropiarse en cualquier momento, no tienen SLA ni reinicio automático; sirven para lotes y cargas sin estado, mejor en un MIG cuya plantilla use spot. Los tipos de máquina personalizados (n2-custom-VCPU-MB) ajustan vCPU y memoria cuando ningún tipo predefinido encaja, por ejemplo con licencias por CPU.

Planning and configuring data storage options

- Cloud SQL es el servicio relacional gestionado para MySQL, PostgreSQL y SQL Server. Spanner es relacional con coherencia global fuerte y escala horizontal de lecturas y escrituras. Bigtable es NoSQL de clave única para miles de millones de filas con baja latencia y compatible con HBase. Firestore es una base de datos de documentos serverless. BigQuery es la plataforma analítica serverless con SQL.
- Cloud Storage ofrece Standard para datos de acceso frecuente, Nearline para lecturas mensuales, Coldline para trimestrales y Archive para menos de una vez al año; las clases frías bajan el coste de almacenamiento a cambio de cobrar el acceso. En bloques, el Persistent Disk zonal vive en una zona y el regional replica de forma síncrona en dos zonas de la región para alta disponibilidad.

Planning and configuring network resources

- Para elegir un balanceador se mira el tipo de tráfico (aplicación para HTTP/HTTPS, red de proxy para TCP, red de transferencia para TCP, UDP y otros protocolos), si es externo o interno y si es global o regional. Los globales usan una IP anycast y backends en varias regiones; los regionales sirven para terminar TLS en una región concreta. Solo los de transferencia conservan la IP de origen del cliente.
- Los recursos de Google Cloud son globales, regionales o zonales. Las redes de VPC, sus rutas y reglas de cortafuegos, las imágenes y las snapshots son globales; las subredes y las IP externas estáticas regionales solo sirven en su región, las VMs y los discos zonales solo en su zona. Repartir recursos entre zonas y regiones reduce el riesgo de que una caída los afecte a la vez.
- El nivel Premium lleva el tráfico por la red troncal de Google y optimiza el rendimiento; el nivel Standard usa redes de ISP y tránsito y optimiza el coste. El nivel se elige por recurso o como valor predeterminado del proyecto, que solo afecta a recursos nuevos. Los balanceadores externos globales, Cloud CDN y Cloud VPN solo funcionan en Premium; VMs, Cloud NAT y balanceadores regionales admiten Standard.

3. Deploying and implementing a cloud solution

25–25%

Deploying and implementing Compute Engine resources

- Al crear una VM se eligen discos (pd-ssd para IOPS aleatorias, pd-standard para E/S secuencial, discos adicionales conectados con attach-disk), la política de mantenimiento (MIGRATE o TERMINATE, obligatoria con GPU) y el reinicio automático (--restart-on-failure). El acceso SSH se gestiona con OS Login (enable-oslogin=TRUE e IAM) o con claves en metadatos, que pueden bloquearse con block-project-ssh-keys.
- Un MIG se crea siempre a partir de una plantilla de instancia, que no se puede modificar; para cambiar la configuración se crea otra y se asigna al grupo. Con --region el MIG reparte las VMs entre zonas. El autoescalado (set-autoscaling, con --max-num-replicas obligatorio) usa CPU, capacidad del balanceador, métricas de Monitoring o programaciones, y el periodo de inicialización evita escalar por picos de arranque.
- OS Login vincula las cuentas Linux de las VMs a identidades de Google y controla el acceso SSH con IAM: roles/compute.osLogin da acceso sin sudo y roles/compute.osAdminLogin con sudo. Se activa con la clave de metadatos enable-oslogin=TRUE en el proyecto o en la instancia (la de instancia prevalece), y la restricción compute.requireOsLogin lo impone en toda una organización, carpeta o proyecto.
- VM Manager agrupa Patch (parches puntuales o programados, con presupuesto de interrupción y scripts previos), OS policies (instalar y mantener software en VMs filtradas por etiqueta, SO o zona) y OS inventory management. Se activa habilitando la API de OS Config y fijando enable-osconfig=TRUE en los metadatos; el agente de OS Config necesita acceso a los repositorios de paquetes.

Deploying and implementing Google Kubernetes Engine resources

- kubectl se instala con gcloud components install kubectl o con apt/yum, y necesita el complemento gke-gcloud-auth-plugin para autenticarse en GKE. gcloud container clusters get-credentials crea la entrada del clúster en kubeconfig y cambia el contexto actual; requiere container.clusters.get, incluido en roles/container.clusterViewer. Para un solo comando contra otro clúster se usa la opción --cluster de kubectl.
- En Autopilot (gcloud container clusters create-auto) Google gestiona nodos, escalado y seguridad, y el clúster siempre es regional. Un clúster regional replica el plano de control en varias zonas; uno zonal tiene un único plano de control aunque sus nodos estén en varias zonas. Con --enable-private-nodes los nodos solo tienen IP internas, y las redes autorizadas o --enable-private-endpoint limitan el acceso al plano de control.
- En GKE, las aplicaciones sin estado se despliegan como Deployment y las que necesitan identidad única y almacenamiento persistente por Pod, como StatefulSet. Se actualizan con kubectl set image, se revierten con kubectl rollout undo y se exponen con un Service; LoadBalancer crea un balanceador con IP externa estable. Para extraer de Artifact Registry la cuenta de los nodos necesita roles/artifactregistry.reader.

Deploying and implementing Cloud Run and Cloud Functions resources

- Cloud Run despliega con gcloud run deploy desde una imagen (--image) o desde el código (--source), que se compila con Dockerfile o con buildpacks mediante Cloud Build; una función usa además --function. --allow-unauthenticated da roles/run.invoker a allUsers. Desplegar requiere roles/run.developer, roles/iam.serviceAccountUser en la identidad de servicio y lectura en Artifact Registry.
- Eventarc entrega a Cloud Run, como peticiones HTTP, eventos de auditoría, eventos directos como los de Cloud Storage y mensajes de Pub/Sub. El activador (gcloud eventarc triggers create) usa una cuenta con roles/run.invoker. En Cloud Storage, el bucket debe estar en el proyecto y la ubicación del activador, y su agente de servicio necesita roles/pubsub.publisher; un tema existente se indica con --transport-topic.
- Cloud Run (totalmente gestionado) ejecuta servicios HTTP sin clústeres, con endpoint HTTPS en *.run.app, escalado a cero y sistema de archivos efímero; sus trabajos ejecutan tareas hasta terminar. Cloud Run functions despliega solo el código de una función, que se ejecuta como servicio de Cloud Run. Knative serving, la oferta gestionada de Knative de Google, lleva cargas sin servidor a tus clústeres de GKE.

Deploying and implementing data solutions

- Cloud SQL ofrece alta disponibilidad con --availability-type=REGIONAL, con réplica síncrona entre dos zonas y conmutación automática. En Firestore, --delete-protection impide borrar la base de datos por error. Un bucket de doble región configurable usa --location y --placement, y --uniform-bucket-level-access desactiva las ACL. Pub/Sub escribe en BigQuery con una suscripción de BigQuery o con plantillas de Dataflow.
- Los archivos locales se suben a Cloud Storage con gcloud storage cp (-r recursivo para directorios) o gcloud storage rsync, bq load carga CSV desde Cloud Storage; añade por defecto, --replace sobrescribe y --skip_leading_rows omite cabeceras. Storage Transfer Service hace transferencias gestionadas y programadas desde S3, Azure u otros buckets. Al importar en Cloud SQL, la cuenta de servicio de la instancia necesita acceso al bucket.

Deploying and implementing networking resources

- Una red VPC en modo automático crea una subred por región dentro de 10.128.0.0/9; en modo personalizado eliges subredes y rangos, y Google lo recomienda en producción. Convertir de automático a personalizado es irreversible.
- Las reglas de firewall de VPC son con estado, de entrada o de salida, y gana la de número de prioridad más bajo; a igual prioridad, deny prevalece. Si nada coincide, se deniega la entrada y se permite la salida. Los objetivos se definen con etiquetas de red o con cuentas de servicio, más estrictas, y las políticas jerárquicas se asocian a la organización o a carpetas.
- En Shared VPC, un proyecto host comparte subredes con proyectos de servicio y compute.networkUser se concede en todo el host o en subredes concretas.
- VPC Network Peering conecta dos redes VPC, se configura en ambos lados, no es transitivo y exige rangos que no se solapen, así que dos redes automáticas no se pueden emparejar; las rutas dinámicas se comparten con --export-custom-routes y --import-custom-routes. Cloud VPN cifra con IPsec el tráfico hacia redes externas y HA VPN usa BGP mediante Cloud Router.

Implementing resources through infrastructure as code

- Terraform describe el estado final de la infraestructura y, en equipo, su estado debe ir a un bucket de Cloud Storage con el backend gcs. Infrastructure Manager ejecuta Terraform como servicio gestionado y Cloud Foundation Toolkit ofrece módulos reutilizables. Config Connector gestiona recursos de Google Cloud con manifiestos de Kubernetes y Helm instala charts guardados en Artifact Registry.

4. Ensuring successful operation of a cloud solution

20–20%

Managing Compute Engine resources

- Sin IP externa, se accede por SSH con IAP TCP forwarding (--tunnel-through-iap), que exige permitir 35.235.240.0/20 y el rol IAP-secured Tunnel User. OS Login controla SSH con IAM mediante roles/compute.osLogin y roles/compute.osAdminLogin. La consola serie sirve para depurar arranque y red, y en Windows las credenciales se crean con gcloud compute reset-windows-password.
- gcloud compute instances list muestra las instancias de todas las zonas del proyecto y admite --filter y --format; gcloud compute instances describe da la configuración completa de una VM, incluido su ID inmutable. Para toda una organización se usa gcloud asset search-all-resources de Cloud Asset Inventory, y los paquetes instalados se consultan con el inventario de VM Manager.
- Las instantáneas estándar y de archivo son incrementales y sobreviven al borrado del disco; las de archivo cuestan menos para retención larga y las inmediatas desaparecen con el disco. Se crean con gcloud compute snapshots create, se listan con gcloud compute snapshots list --filter y se programan con una política snapshot schedule asociada con gcloud compute disks add-resource-policies.
- Las imágenes personalizadas se crean con gcloud compute images create desde un disco (--source-disk, con --force si la VM sigue en ejecución) o desde una instantánea (--source-snapshot), y las familias apuntan a la imagen activa más reciente. Se listan con gcloud compute images list --no-standard-images, se retiran con images deprecate y se eliminan sin vuelta atrás con images delete.

Managing Google Kubernetes Engine resources

- Para consultar un clúster de GKE con kubectl primero se obtienen las credenciales con gcloud container clusters get-credentials, que actualiza kubeconfig. Después, kubectl get nodes, kubectl get pods y kubectl get services listan el inventario; -A o --all-namespaces abarca todos los espacios de nombres, y -o wide añade detalles como el nodo de cada Pod y Ready por etiquetas, por ejemplo cloud.google.com/gke-nodepool para los nodos de un grupo.
- GKE extrae imágenes de Artifact Registry con la cuenta de servicio de los nodos: en el mismo proyecto basta con roles/artifactregistry.reader y un nivel de acceso con lectura de almacenamiento. Si el repositorio está en otro proyecto, el rol se concede allí. Los niveles de acceso (storage-ro, storage-rw, cloud-platform) solo se fijan al crear el clúster o el grupo de nodos.
- En GKE Standard los grupos de nodos se crean con gcloud container node-pools create, se redimensionan con gcloud container clusters resize (en clústeres regionales --num-nodes es por zona) y se eliminan con gcloud container node-pools delete, que por defecto no respeta los PodDisruptionBudgets. Google recomienda una cuenta de servicio propia con roles/container.defaultNodeServiceAccount.
- Un StatefulSet da a cada Pod un identificador estable y con volumeClaimTemplates un PersistentVolumeClaim propio; se gobierna con un Service headless (clusterIP: None), que publica por DNS las IP de los Pods. Un Service ClusterIP ofrece una IP interna estable y reenvía de port a targetPort. kubectl scale y kubectl rollout undo escalan y revierten.
- El HorizontalPodAutoscaler cambia el número de réplicas según CPU, memoria o métricas personalizadas (kubectl autoscale) y necesita solicitudes de recursos para objetivos en porcentaje; no añade nodos, eso lo hace el escalador de clústeres. El VerticalPodAutoscaler ajusta solicitudes de CPU y memoria: Off solo recomienda y Auto desaloja Pods para aplicarlas.

Managing Cloud Run resources

- Cada despliegue o cambio de configuración en Cloud Run crea una revisión inmutable; una etiqueta de imagen se resuelve a un digest al desplegar. gcloud run deploy --image despliega una imagen y --source compila el código con Cloud Build. Con --no-traffic y --tag se prueba una revisión en su propia URL sin tráfico de usuarios.
- Cloud Run reparte el tráfico entre revisiones con gcloud run services update-traffic: --to-revisions REV=PORCENTAJE para revertir o dividir, LATEST para la revisión más reciente y --to-latest para enviarle siempre el 100 %. Tras un reparto, los despliegues siguientes lo mantienen. En YAML los porcentajes deben sumar 100 y se aplican con gcloud run services replace.
- Cloud Run escala por solicitudes y puede bajar a cero. Las instancias mínimas (--min) mantienen instancias activas para evitar arranques en frío y se facturan; las máximas (--max) limitan el escalado para controlar costes o conexiones a una base de datos. La simultaneidad (--concurrency) fija cuántas solicitudes atiende cada instancia; 1 para código no patabelo.

Managing storage and database solutions

- Cloud Storage protege los objetos con roles de IAM por bucket (Object Viewer, Object Creator, Object User, Object Admin), con acceso uniforme a nivel de bucket, que desactiva las ACL, y con la prevención del acceso público. Las URL firmadas dan acceso temporal sin cuenta. Una política de retención bloqueada con Bucket Lock impide eliminar o sustituir objetos.
- La gestión del ciclo de vida de Cloud Storage aplica reglas con una acción (Delete, SetStorageClass o AbortIncompleteMultipartUpload) y condiciones como age, matchesPrefix, isLive o numNewerVersions. Si varias reglas coinciden, Delete tiene prioridad y gana la clase más barata. Delete espera a que se cumpla la retención. Se aplica con gcloud storage buckets update --lifecycle-file.
- Cada servicio tiene su vía de consulta: gcloud sql connect abre un cliente contra Cloud SQL, gcloud spanner databases execute-sql lanza SQL en Spanner y bq query lo hace en BigQuery. Las consultas federadas usan EXTERNAL_QUERY para leer Cloud SQL, Spanner o AlloyDB desde BigQuery sin copiar datos. En Firestore, las consultas compuestas con rango necesitan un índice compuesto.
- La calculadora de precios estima costes antes de desplegar a partir de tus hipótesis de uso. En Cloud Storage el coste combina almacenamiento (según la clase de cada objeto), procesamiento (operaciones y tarifas de recuperación en Nearline, Coldline y Archive) y red. En BigQuery, la vista INFORMATION_SCHEMA.TABLE_STORAGE permite prever el coste con facturación lógica o física.
- Cloud SQL ofrece copias automáticas y bajo demanda (gcloud sql backups create); restaurar en la misma instancia sobrescribe sus datos, y la recuperación a un momento dado siempre crea una instancia nueva. Firestore permite programar copias diarias o semanales con retención configurable, que se restauran siempre en una base de datos nueva y no incluyen políticas IAM ni Security Rules.

Managing networking resources

- Las redes de VPC son globales y las subredes regionales: se añaden con gcloud compute networks subnets create indicando red, rango principal y región. Los rangos no pueden solaparse con otras subredes de la red ni de redes emparejadas, y conviene evitar los rangos locales. Las subredes IPv6 solo existen en modo personalizado, y el acceso privado a Google se puede activar al crear la subred.
- El rango IPv4 principal de una subred se amplía con gcloud compute networks subnets expand-ip-range y una longitud de prefijo menor, sin interrumpir el tráfico. La ampliación no se puede deshacer ni reducir, así que conviene ampliar con prudencia. Después hay que revisar las reglas de cortafuegos que usan el rango original y, si hay GKE, las redes autorizadas del plano de control.
- Una dirección estática se reserva con gcloud compute addresses create: --global para balanceadores globales, --region para VM y balanceadores regionales, y --region con --subnet para direcciones internas. Una dirección efímera en uso se puede promover a estática sin cortar el tráfico. Las direcciones reservadas no se liberan al borrar el recurso, y las externas sin asignar tienen una tarifa más alta.
- Cloud DNS ofrece zonas públicas, visibles en internet, y privadas, visibles solo desde las redes de VPC autorizadas; para publicar una zona pública hay que apuntar en el registrador los servidores de nombres de Cloud DNS. Cloud NAT da salida a internet a recursos sin IP externa mediante una pasarela regional asociada a un Cloud Router, sin admitir conexiones entrantes no solicitadas.

Monitoring and logging

- Una política de alertas de umbral de métrica avisa cuando una métrica supera o queda por debajo de un umbral durante toda la ventana de nueva prueba, que evita alertas por una sola medición. Las series temporales se pueden agregar y agrupar por etiquetas, como la zona. Con varias condiciones, AND_WITH_MATCHING_RESOURCE exige que todas las cumpla el mismo recurso.
- Las métricas definidas por el usuario se recogen preferentemente con OpenTelemetry (receptor OTLP del Ops Agent en Compute Engine) o con la API de Monitoring, eligiendo un recurso supervisado adecuado como generic_task en vez de global. Las métricas basadas en registros se crean en Cloud Logging (contador o distribución), solo cuentan entradas recibidas tras su creación y sirven para gráficos y alertas.
- Para exportar registros se crean sinks: Pub/Sub para enviarlos casi en tiempo real a herramientas externas o locales, BigQuery para combinarlos con datos de negocio y Cloud Storage para archivarlos. La identidad de escritura del sink necesita permisos en el destino; los sinks agregados con --include-children cubren carpetas u organizaciones, y las entradas ya almacenadas solo se copian a Cloud Storage con gcloud logging copy.
- El enrutador de registros evalúa cada entrada con los sinks del recurso: _Required no se puede modificar y _Default admite filtros de exclusión, cambio de destino, inhabilitarse. Los buckets de registros definidos por el usuario permiten fijar la retención y bloquearse con --locked, si se actualizan a Observability Analytics admiten SQL y un conjunto de datos vinculado de BigQuery sin costes de ingesta.
- El Explorador de registros, la API y gcloud logging read usan el lenguaje de consultas de Logging: comparaciones campo operador valor unidas por AND, OR y NOT en mayúsculas, con AND implícito entre líneas. La gravedad se compara por orden numérico, NOT sobre un campo ausente es verdadero y != es falso, y en gcloud logging read la marca --freshness amplía la antigüedad de las entradas devueltas.

5. Configuring access and security

17–18%

Managing Identity and Access Management (IAM)

- Una política de permiso reúne enlaces de rol (principales, rol y condición opcional) más etag y version; se adjunta a un recurso y la heredan sus descendientes. gcloud projects add-iam-policy-binding añade un enlace, mientras que set-iam-policy sustituye la política entera y exige leer-modificar-escribir con el etag. get-iam-policy no muestra roles heredados; para eso se usa Policy Troubleshooter.
- IAM tiene roles básicos (muy amplios; no recomendados en producción), predefinidos (mantenidos por Google) y personalizados. Los personalizados se crean en una organización o un proyecto, nunca en una carpeta, y solo se conceden allí; se gestionan con gcloud iam roles create, update, delete y undelete, y la fase DISABLED los deja sin efecto sin quitarlos de las políticas.

Managing service accounts

- Una cuenta de servicio se crea con gcloud iam service-accounts create ID --display-name; su correo es ID@PROJECT_ID.iam.gserviceaccount.com y el ID no se puede cambiar. El rol necesario es roles/iam.serviceAccountCreator. Google recomienda una cuenta por aplicación, prefijos como vm- y una descripción con el contacto, y reintentar con espera exponencial si la cuenta nueva aún no es visible.
- Una cuenta de servicio debe recibir solo los roles que necesita y en el recurso más concreto posible, por ejemplo un bucket en lugar del proyecto. Los ámbitos de acceso no sustituyen a IAM, la restricción iam.automaticIamGrantsForDefaultServiceAccounts evita que las cuentas predeterminadas reciban Editor, y las recomendaciones de roles señalan los permisos que no se usan.
- Asociar una cuenta de servicio a un recurso (máquina virtual, servicio de Cloud Run) exige iam.serviceAccounts.actAs, incluido en roles/iam.serviceAccountUser sobre la cuenta. En Compute Engine hay que detener la máquina antes de cambiar la cuenta y conviene usar el ámbito cloud-platform; en GKE se usa Workload Identity Federation for GKE; entre proyectos, no se debe aplicar iam.disableCrossProjectServiceAccountUsage.
- Una cuenta de servicio también es un recurso con su propia política de permiso, que se consulta y modifica con gcloud iam service-accounts get-iam-policy y add-iam-policy-binding. Quien tiene roles/iam.serviceAccountAdmin puede concederse la suplantación, así que conviene conceder el acceso por cuenta y no por proyecto. Es mejor inhabilitar una cuenta antes de eliminarla, porque conserva sus enlaces.
- Suplantar una cuenta de servicio significa que un principal autenticado obtiene credenciales de corta duración de esa cuenta; requiere iam.serviceAccounts.getAccessToken, incluido en roles/iam.serviceAccountTokenCreator. Se usa con --impersonate-service-account, con la propiedad auth/impersonate_service_account o con gcloud auth application-default login para las bibliotecas de cliente, y evita las claves.