# Guía de seguridad TIC
# CCN-STIC 888C

# Guía de configuración segura para Contenedores


Noviembre 2021

MINISTERIO DE DEFENSA

**Catálogo de Publicaciones de la Administración General del Estado**
**https://cpage.mpr.gob.es**

Edita:

MINISTERIO DE DEFENSA
SUBSECRETARÍA DE DEFENSA
SECRETARÍA GENERAL TÉCNICA

centro criptológico nacional

2.5.4.13=Qualified Certificate: AAPP-SEP-M-SW-KPSC, ou=sello electrónico, serialNumber=S2800155J, o=CENTRO CRIPTOLOGICO NACIONAL, c=ES
2021.11.02 17:04:23 +01'00'

Pº de la Castellana 109, 28046 Madrid

NIPO: 083-21-205-1

Fecha de Edición: noviembre de 2021
Davinci Tecnologías de la Información, S.L ha participado en la realización y modificación del presente documento y sus anexos, que ha sido financiado por Google.

# PRÓLOGO

En un mundo cada vez más complejo y globalizado, en el que las tecnologías de la información y la comunicación (TIC) desempeñan un papel de suma importancia, hemos de ser conscientes de que la gestión adecuada de la ciberseguridad constituye un reto colectivo al que necesariamente hemos de enfrentar. Resulta necesario garantizar la protección de la capacidad económica, tecnológica y política de nuestro país, máxime cuando la proliferación de ataques dirigidos y el robo de información sensible representan una realidad incontestable.

Por ello, resulta imprescindible estar al día de las amenazas y vulnerabilidades asociadas al uso de las nuevas tecnologías. El conocimiento de los riesgos que se ciernen sobre el ciberespacio ha de servir para implementar con garantías las medidas, tanto procedimentales como técnicas y organizativas, que permitan un entorno seguro y confiable.

La Ley 11/2002, de 6 de mayo, reguladora del Centro Nacional de Inteligencia (CNI), encomienda al Centro Nacional de Inteligencia el ejercicio de las funciones relativas a la seguridad de las tecnologías de la información y de protección de la información clasificada, a la vez que confiere a su Secretario de Estado Director la responsabilidad de dirigir el Centro Criptológico Nacional (CCN)

Partiendo del conocimiento y la experiencia del CNI sobre amenazas y vulnerabilidades en materia de riesgos emergentes, el Centro realiza, a través del Centro Criptológico Nacional, regulado por el Real Decreto 421/2004, de 12 de marzo, diversas actividades directamente relacionadas con la seguridad de las TIC, orientadas a la formación de personal experto, al empleo de tecnologías de seguridad adecuadas y a la aplicación de políticas y procedimientos de seguridad.

Precisamente, esta serie de documentos CCN-STIC es un claro reflejo de la labor que este organismo lleva a cabo en materia de implementación de seguridad, permitiendo la aplicación de políticas y procedimientos, pues las guías han sido elaboradas con un claro objetivo: mejorar el grado de ciberseguridad de las organizaciones, conscientes de la importancia que tiene el establecimiento de un marco de referencia en esta materia que sirva de apoyo para que el personal de la Administración lleve a cabo la difícil tarea de proporcionar seguridad a los sistemas de las TIC bajo su responsabilidad.

Con esta serie de documentos, el Centro Criptológico Nacional, en cumplimiento de sus cometidos y de lo reflejado en el Real Decreto 3/2010 por el que se regula el Esquema Nacional de Seguridad en el ámbito de la Administración electrónica, contribuye a mejorar la ciberseguridad española y mantener las infraestructuras y los sistemas de información de todas las administraciones públicas con unos niveles óptimos de seguridad. Todo ello, con el fin de generar confianza y garantías en el uso de estas tecnologías, protegiendo la confidencialidad de los datos y garantizando su autenticidad, integridad y disponibilidad.

Noviembre de 2021

Paz Esteban López
Secretaria de Estado
Directora del Centro Criptológico Nacional

# ÍNDICE

# 1. GUÍA DE CONFIGURACIÓN SEGURA PARA GCP

## 1.1. Descripción del uso de esta guía

El contenido de esta guía muestra el despliegue y configuración para cargas de trabajo con contenedores en la nube pública de Google Cloud Platform (GCP) siguiendo las exigencias del Esquema Nacional de Seguridad (ENS).

Una de las principales utilidades de esta guía es explicar y referir a los servicios ofrecidos por GCP, que permiten el uso de contenedores, para cumplir con las diferentes medidas del ENS. Además, se referencian las buenas prácticas que exige el fabricante para la configuración de una máquina virtual o de un cluster de kubernetes.

# 2. SERVICIOS DE GCP DISPONIBLES PARA CONTENEDORES

GCP ofrece diferentes servicios que permiten el despliegue de contenedores, ya sea utilizando un cluster de kubernetes (GKE), desplegando una máquina virtual con un contenedor predefinido (GCE), bien desplegando esos contenedores en una infraestructura gestionada por Google APP Engine o usando la función serverless de Cloud Run.

## 2.1. Google Kubernetes Engine

Google Kubernetes Engine (GKE) proporciona un entorno administrado para implementar, administrar y escalar las aplicaciones en contenedores mediante la infraestructura de Google. El entorno de GKE consta de varias máquinas (en particular, instancias de Compute Engine) que se agrupan para formar un clúster.

Los clústeres de GKE funcionan con Kubernetes. Kubernetes proporciona los mecanismos a través de los cuales interactúa con el clúster. Se pueden usar comandos y recursos de Kubernetes para implementar y administrar las aplicaciones, realizar tareas de administración, establecer políticas y supervisar el estado de las cargas de trabajo implementadas.

Kubernetes ofrece los siguientes beneficios: administración automática, supervisión y sondeos de capacidad de funcionamiento de los contenedores de aplicaciones, ajuste de escala automático, actualizaciones progresivas y mucho más.


*Ilustración 1 Logo de Google Kubernetes Engine.*

Los clústeres de GKE tienen dos modos de operación entre los que se puede elegir:

- Autopilot: Administra toda la infraestructura de clúster y nodo de manera automática. Autopilot proporciona una experiencia práctica de Kubernetes que permite enfocarse en las cargas de trabajo. Los clústeres de Autopilot están preconfigurados con una configuración de clúster optimizada que está lista para las cargas de trabajo de producción.

- Estándar: Proporciona flexibilidad de configuración de nodos y control total sobre la administración de los clústeres y la infraestructura de nodos. En el caso de los clústeres creados con el modo estándar, se puede determinar la configuración necesaria para las cargas de trabajo de producción.

Según el modo de operación elegido anteriormente se podrá definir la configuración del clúster de la siguiente manera:

| **Opciones de clúster** | **Modo** | |
|---|---|---|
| | **Autopilot** | **Estándar** |
| **Tipo de disponibilidad** | Regional | Regional o zonal |
| **Versión** | Canal de versiones | Canal de versiones, predeterminado o específico |
| **Enrutamiento de herramientas de red** | VPC nativa | VPC nativa o basada en rutas |
| **Aislamiento de red** | Privado o público | Privado o público |
| **Características de kubernetes** | Producción | Producción o Alfa |

Dentro de GKE se puede encontrar también diferentes tipos de clústeres en función de su disponibilidad:

- Clúster zonal: Por defecto, un clúster se crea un solo plano de control. Un clúster zonal ejecuta nodos en múltiples zonas dentro de la misma región. Todos los nodos de un clúster de una o varias zonas están controlados por el mismo clúster maestro.
- Clúster regional: Tiene varias réplicas del plano de control que se ejecutan en varias zonas dentro de una región determinada. Los nodos de un clúster regional pueden ejecutarse en varias zonas o en una sola zona según las ubicaciones donde se configuren los nodos.

Se puede consultar más información sobre GKE en la documentación proporcionada por el fabricante:
https://cloud.google.com/kubernetes-engine/docs/concepts/kubernetes-engine-overview

## 2.2. Google Compute Engine

Las instancias de Compute Engine pueden ejecutar las imágenes públicas de Linux y Windows Server que proporciona Google, así como las imágenes personalizadas privadas que puedes crear o importar desde los sistemas existentes. También permite implementar contenedores de Docker, que se inician de forma automática en instancias que ejecutan la imagen pública de Container-Optimized OS.

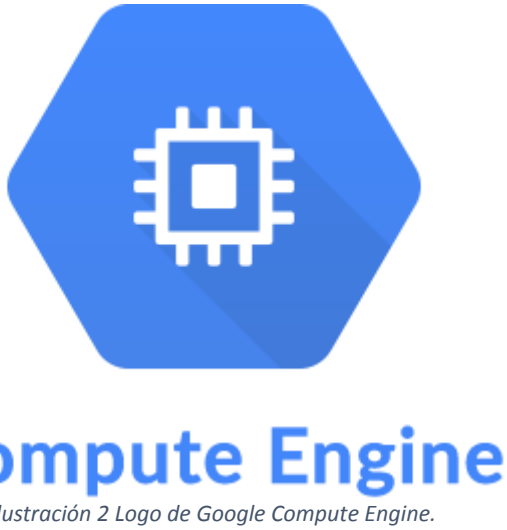


*Ilustración 2 Logo de Google Compute Engine.*

Las instancias de Compute Engine admiten un método declarativo para iniciar las aplicaciones con contenedores. Cuando se crea una VM o una plantilla de instancias, se puede proporcionar un nombre de imagen de Docker y, luego, iniciar la configuración. Compute Engine se encargará del resto, incluido el suministro de una imagen actualizada de Container-Optimized OS con Docker instalado y el lanzamiento del contenedor cuando se inicia la VM.

Este servicio tiene algunas limitaciones:

- Solo permite implementar un contenedor para cada instancia de VM.
- Solo se pueden implementar contenedores desde un repositorio público o desde un repositorio privado de Container Registry al que se tenga acceso. No se admiten otros repositorios privados.
- Con este método de implementación, solo puedes usar las imágenes de Container-Optimized OS.

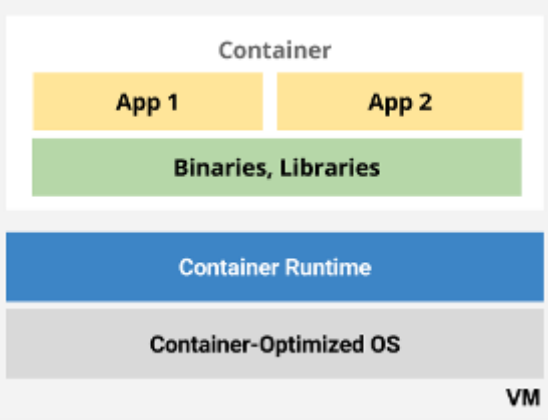


*Ilustración 3 Arquitectura de una máquina virtual con contenedor.*

Se puede consultar más información sobre GKE en la documentación proporcionada por el fabricante:
https://cloud.google.com/compute/docs/containers/?hl=es-419

## 2.3. Google APP Engine

App Engine es una plataforma sin servidores completamente administrada para desarrollar y alojar aplicaciones web a gran escala. Permite elegir entre varios lenguajes, bibliotecas y marcos de trabajo para desarrollar la aplicación. Luego, App Engine se encarga del aprovisionamiento de servidores y del escalamiento de las instancias de la aplicación según demanda.


*Ilustración 4 Logo de Google APP Engine.*

App Engine permite dos entornos donde desplegar las aplicaciones:

- Entorno Estándar: El entorno estándar de App Engine se basa en instancias de contenedores que se ejecutan en la infraestructura de Google. Los contenedores se configuran previamente con uno de los varios entornos de ejecución disponibles.
- Entorno Flexible: El entorno flexible de App Engine aumenta y disminuye la escala de la aplicación de forma automática y balancea la carga. Las instancias de aplicación se ejecutan dentro de contenedores Docker en máquinas virtuales (VM) de Compute Engine.

App Engine es muy adecuado para aplicaciones diseñadas con una arquitectura de microservicio, especialmente si se decide usar ambos entornos.

| Función | Entorno estándar | Entorno flexible |
|---|---|---|
| Tiempo de inicio de instancia | Segundos | Minutos |
| Procesos en segundo plano | No | Sí |
| Depuración de SSH | No | Sí |
| Escalamiento | Manual, Básico, Automático | Manual, Automático |
| Modificación del entorno de ejecución | No | Sí (a través de Dockerfile) |
| Tiempo de implementación | Segundos | Minutos |

Se puede consultar con más detalle sobre las principales diferencias entre los entornos en la documentación del fabricante:
https://cloud.google.com/appengine/docs/the-appengine-environments

## 2.4. Cloud Run

Cloud Run es una plataforma sin servidores, lo que significa que quita la complejidad de la administración de infraestructura.

Cloud Run simplifica la administración de la infraestructura mediante un escalamiento vertical automático que aumenta o disminuye desde cero, según el tráfico, y de modo casi instantáneo.

*Ilustración 5 Logo de Cloud Run.*

Los servicios de Cloud Run son regionales y se replican de forma automática en varias zonas, ayudando a la continuidad del servicio.

## 2.5. Container Registry

Container Registry proporciona almacenamiento privado y seguro de imágenes de Docker en Google Cloud Platform.

Gracias a Container Registry, un equipo puede gestionar imágenes Docker, analizar vulnerabilidades y decidir quiénes tienen acceso a qué elementos.

Google Container Registry

*Ilustración 6 Logo de Google Container Registry.*

# 3. CONFIGURACIÓN SEGURA PARA CONTENEDORES

En las secciones siguientes se presentan las medidas de aplicación comprendidas en los ámbitos Marco Operacional y Medidas de Protección del Esquema Nacional de Seguridad.

## 3.1. Marco Operacional

El marco operacional está constituido por las medidas a tomar para proteger la operación del sistema como conjunto integral de componentes para un fin.

### 3.1.1. Control de Acceso

El control de acceso cubre el conjunto de actividades preparatorias y ejecutivas para que una determinada entidad, usuario o proceso, pueda, o no, acceder a un recurso del sistema para realizar una determinada acción. El control de acceso que se implante en un sistema real será un punto de equilibrio entre la usabilidad y la protección de la información.

En GCP el control de accesos está principalmente gobernado por el servicio *GCP Identity and Access Management* (IAM). La correcta configuración de IAM otorga acceso detallado a recursos específicos de Google Cloud y ayuda a evitar el acceso a otros recursos.

Una Política de IAM define y aplica qué funciones se otorgan a qué miembros; esta política se vincula a un recurso. Cuando un miembro autenticado intenta acceder a un recurso, IAM verifica la política del recurso para determinar si la acción está permitida.

Se puede obtener más información sobre el control de acceso de GCP en la guía ***CCN 888B Guía de Configuración Segura de Google Cloud Platform***.

#### 3.1.1.1. Identificación

Dentro de GCP se definen dos principales conceptos a diferenciar antes de tratar el siguiente punto, que son, la autenticación y la autorización:

- Autenticación: La autenticación garantiza que el usuario es quien dice ser. Los nombres de usuario y las contraseñas son los tipos de autenticación más comunes, pero también se puede trabajar con otras formas, como la autenticación basada en tokens o en datos biométricos como una huella dactilar. La autenticación simplemente responde a la pregunta: "¿Eres quien dices ser?".
- Autorización: La autorización es el proceso de dar permiso a los usuarios para acceder a los recursos y servicios de AWS. La autorización determina si el usuario puede realizar una acción, ya sea leer, editar, eliminar o crear recursos. La autorización responde a la pregunta: "¿Qué acciones puede realizar?".

**GKE**

Google Kubernetes Engine admite varias opciones para administrar el acceso a los recursos de tu proyecto y sus clústeres mediante el control de acceso según la función (RBAC):

- Administración de identidades y accesos (IAM)
- Control de acceso según la función de Kubernetes (RBAC)

Estos mecanismos se superponen en cuanto a sus funciones en cierto punto, pero están dirigidos a diferentes tipos de recursos:

- El RBAC de Kubernetes está incorporado dentro de Kubernetes y otorga permisos detallados a los objetos dentro de los clústeres de Kubernetes. Los permisos existen como objetos Role o ClusterRole dentro del propio cluster. Además, se puede encontrar el objeto RoleBinding el cual otorga funciones a los usuarios de Kubernetes o Google Cloud además de a los grupos de Google. Si se usa GKE y se necesitan permisos específicos para cada objeto y operación dentro del clúster, el RBAC de Kubernetes es la opción que se deberá usar.
- IAM administra los recursos de Google Cloud, incluidos los clústeres y los tipos de objetos dentro de los clústeres. Los permisos se asignan a los miembros de IAM, que existen en Google Cloud, Google Workspace o Cloud Identity. Una función de IAM permite otorgar privilegios en todos los clústeres del proyecto.

Si se usan varios componentes de Google Cloud y no se necesita administrar permisos específicos detallados de Kubernetes, IAM es la opción que se deberá usar.

Kubernetes cuenta con compatibilidad integrada para **RBAC**. Esto permite crear funciones detalladas, que existen dentro del clúster de Kubernetes. Una función puede aplicarse a un objeto de Kubernetes específico o a un tipo de objeto de Kubernetes, y define qué acciones (llamadas verbos) otorga la función en relación con ese objeto. Un RoleBinding también es un objeto de Kubernetes y otorga funciones a los usuarios. En GKE, un usuario puede ser cualquiera de las siguientes opciones:

- Usuario de Google Cloud
- Cuenta de servicio de Google Cloud
- Cuenta de servicio de Kubernetes
- Usuario de Google Workspace
- Grupo de Google de Google Workspace(Beta)

**IAM** permite definir funciones y asignarlas a los miembros desde la consola fe GCP en el apartado de IAM. Una función es una colección de permisos, y cuando se le asigna a un miembro, controla el acceso a uno o más recursos de Google Cloud. Las funciones se dividen en tres categorías amplias:

- Las funciones básicas proporcionan permisos generales limitados a permisos de propietario, de editor y de visualizador.
- Las funciones predefinidas, como las funciones predefinidas de GKE, proporcionan un acceso más detallado que las funciones básicas y abordan muchos casos prácticos comunes.
- Las funciones personalizadas permiten crear combinaciones de permisos únicas.

Cualquiera de los siguientes puede ser un miembro:

- Cuenta de Google
- Cuenta de servicio
- Grupo de Google
- Dominio de Google Workspace
- Dominio de Cloud Identity

Una política de IAM asigna un conjunto de permisos a uno o más miembros de Google Cloud.

IAM y RBAC de Kubernetes trabajan juntos para ayudar a administrar el acceso al clúster. RBAC controla el acceso en el nivel de clúster y espacio de nombres, mientras que IAM funciona en el nivel de proyecto.

**Container Registry**

Todos los usuarios, las cuentas de servicio y otras entidades que interactúan con Container Registry deben tener los permisos adecuados de administración de identidades y accesos (IAM) para el almacenamiento de Cloud Storage.

Container Registry usa depósitos de Cloud Storage como almacenamiento subyacente para las imágenes de contenedor. Para controlar el acceso a las imágenes, se deberá otorgar los permisos de Cloud Storage adecuados a un usuario, grupo, cuenta de servicio o a otra identidad.

Los permisos de Cloud Storage otorgados a nivel de proyecto se aplican a todos los depósitos de almacenamiento del proyecto, no solo a los que usa Container Registry. Para configurar permisos específicos de Container Registry, se deberán otorgar en el bucket de almacenamiento que usa el registro.

Se pueden consultar todos los permisos y funciones disponibles para Cloud Storage en la siguiente tabla proporcionada por el fabricante:
https://cloud.google.com/container-registry/docs/access-control?hl=es#permissions_and_roles

**Cloud Run**
Cloud Run usa IAM para otorgar funciones a diferentes miembros. Estas funciones se encuentran dentro de la consola de GCP en el apartado de IAM.

Se puede consultar una lista completa de todos los permisos que se pueden asignar a los usuarios en la siguiente tabla del fabricante:
https://cloud.google.com/run/docs/reference/iam/roles?hl=es

#### 3.1.1.2. Segregación de funciones y tareas

**RBAC**
Dentro de Kubernetes y GKE se permiten crear los siguientes objetos de Kubernetes para definir los permisos de RBAC:

- ClusterRole o Role: Define un conjunto de tipos de recursos y operaciones que pueden asignarse a un usuario o grupo de usuarios en un clúster (ClusterRole) o un espacio de nombres (Role), pero no especifica el usuario o grupo de usuarios.
- ClusterRoleBinding o RoleBinding: Asigna un ClusterRole o Role a un usuario o grupo de usuarios. Una ClusterRoleBinding funciona con un ClusterRole y una RoleBinding funciona con un ClusterRole o un Role.

Las funciones de RBAC son completamente aditivas: no hay reglas de “denegación”.

Después de crear un Role o ClusterRole, se debe asignar a un usuario o grupo de usuarios mediante un RoleBinding o ClusterRoleBinding. Los usuarios y los grupos se llaman *subjects* y pueden ser cualquiera de los siguientes:

| Usuario | Kind | Valor |
|---|---|---|
| Cuenta de usuario de Google Cloud | User | Dirección de correo electrónico registrada en Google Cloud |
| Cuenta de servicio de Kubernetes | ServiceAccount | El nombre de un objeto ServiceAccount de Kubernetes en el clúster |

| Usuario | Kind | Valor |
|---|---|---|
| Cuenta de servicio de IAM | User | Dirección de correo electrónico de la cuenta de servicio de IAM generada de forma automática |
| Dirección del Grupo de Google (beta) en un dominio verificado | Group | Dirección de correo electrónico de un Grupo de Google que es miembro del Grupo de Google gke-security-groups@YOUR_DOMAIN |

La siguiente RoleBinding otorga la función pod-reader a un usuario, una cuenta de servicio de Kubernetes, una cuenta de servicio de IAM y un Grupo de Google:

```
kind: RoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: pod-reader-binding
  namespace: accounting
subjects:
# Cuenta de usuario de Google Cloud
- kind: User
  name: janedoe@example.com
# Cuenta de servicio de Kubernetes
- kind: ServiceAccount
  name: johndoe
# Cuenta de servicio de IAM
- kind: User
  name: test-account@test-project-123456.google.com.iam.gserviceaccount.com
# Grupo de Google
- kind: Group
  name: accounting-group@example.com
roleRef:
  kind: Role
  name: pod-reader
  apiGroup: rbac.authorization.k8s.io
```

#### 3.1.1.3. Proceso de gestión de derechos de acceso

**Workload Identity**

Workload Identity permite acceder a otros servicios de Google Cloud desde el clúster de GKE. Workload Identity permite mantener el principio de privilegio mínimo, ya que se puede vincular cuentas de servicio de Kubernetes a cuentas de servicio de Google.

Las cuentas de servicio de Kubernetes se pueden asignar a los Pods, lo que permite tener permisos detallados para acceder a las API de Google por carga de trabajo.

Para habilitar Workload Identity en un cluster, se puede consultar la siguiente guía del fabricante:
https://cloud.google.com/anthos/clusters/docs/aws/how-to/workload-identity-cluster?hl=es-419#overview

**Compute Engine**
**Acceso a VM**
Google Compute Engine ofrece varios métodos de acceso a las máquinas virtuales (VM) de GCE:

- Acceso a SO: El acceso al SO permite administrar el acceso SSH a las instancias con la IAM sin tener que crear y administrar claves SSH individuales. El acceso al SO mantiene una identidad de usuario de Linux coherente en todas las instancias de VM y es la forma recomendada para administrar muchos usuarios en múltiples instancias o proyectos. El acceso a SO permite aplicar la autenticación multifactor añadiendo más seguridad a la hora de conectar con las VM.
- Administra Llaves SSH en metadatos: Cuando se crean y administran Llaves SSH, se puede permitir que los usuarios accedan a una instancia de Linux a través de herramientas de terceros. Se pueden controlar las Llaves SSH públicas que están disponibles para una instancia de Linux, editando los metadatos de la instancia. También se pueden definir llaves SSH a nivel de proyecto, dando acceso a la gran mayoría de instancias.
- Google Compute Engine también permite la conexión mediante RDP a las instancias con Windows.

Se puede obtener más información sobre los procesos de gestión de derechos de acceso en la guía ***CCN 888B Guía de Configuración Segura de Google Cloud Platform*** en la sección ***3.1.1.4 Proceso de gestión de derechos de acceso***.

#### 3.1.1.4. Acceso Local y Remoto

Al ser GCP un servicio cloud accesible por el usuario final a través de internet deberán tenerse en cuenta en todo caso las exigencias del capítulo del ENS relativo a acceso local y remoto.

**GKE**
Kubernetes usa un archivo YAML llamado kubeconfig a fin de almacenar información de autenticación de clúster para kubectl. Kubeconfig contiene una lista de contextos a los que kubectl se refiere cuando ejecuta comandos.

Un contexto es un grupo de parámetros de acceso. Cada contexto contiene un clúster de Kubernetes, un usuario y un espacio de nombres. El contexto actual es el clúster predeterminado para kubectl: todos los comandos kubectl se ejecutan en ese clúster.

Para añadir un cluster de GKE al archivo de configuración kubeconfig se puede seguir la siguiente guía del fabricante:
https://cloud.google.com/kubernetes-engine/docs/how-to/cluster-access-for-kubectl#generate_kubeconfig_entry

Todos los clústeres de GKE están configurados para aceptar identidades de usuario y cuenta de servicio de Google Cloud, mediante la validación de las credenciales que presenta kubectl.

Para configurar la herramienta kubectl con GKE se puede consultar la siguiente guía del fabricante:
https://cloud.google.com/kubernetes-engine/docs/how-to/cluster-access-for-kubectl#view_current_context

**Compute Engine**
Las instancias de VM con sistema operativo Linux, permiten el acceso SSH. De forma predeterminada, Compute Engine crea reglas de firewall que permiten conexiones TCP a través del puerto 22.

De manera opcional, se pueden almacenar las claves de host como atributos de invitado en las instancias de Linux para agregar una capa de seguridad adicional. Para la implementación de estas claves de host se puede consultar la siguiente guía del fabricante:
https://cloud.google.com/solutions/connecting-securely?hl=es-419#storing_host_keys_by_enabling_guest_attributes

### 3.1.2. Explotación

Este apartado del marco operacional del Esquema Nacional de Seguridad exige el mantenimiento de un inventario actualizado de activos, incluyendo detalles de naturaleza y responsable.

#### 3.1.2.1. Inventario de activos

**Tags (Etiquetas)**
Una etiqueta es un par clave-valor que ayuda a organizar los recursos de Google Cloud. Se puede adjuntar una etiqueta a cada recurso y, luego, usarlas para filtrarlos.

No se recomienda crear grandes cantidades de etiquetas únicas, como marcas de tiempo o valores individuales. Estos son algunos casos prácticos comunes de las etiquetas de clúster:

- Etiquetas de clúster por equipo o del centro de costos: Agregar etiquetas por equipo o centro de costos para distinguir los clústeres que pertenecen a distintos equipos (**team:devops** y **team:security**).
- Etiquetas de clúster de componentes (**component:redis**, **component:frontend**, **component:ingest**, **component:dashboard**).
- Etiquetas de clúster de entorno o etapa (**environment:production**, **environment:test**).
- Etiquetas de clúster de estado (**state:active**, **state:readytodelete**, **state:archive**).

#### 3.1.2.2. Mantenimiento

**GKE**

Dentro del servicio de GKE, se disponen de herramientas y configuraciones que permiten realizar un mantenimiento controlado.

Las exclusiones y los períodos de mantenimiento brindan un control detallado acerca de cuándo puede ocurrir el mantenimiento automático en los clústeres.

Un período de mantenimiento es un período arbitrario y recurrente durante el cual puede ocurrir el mantenimiento automático.

Una exclusión de mantenimiento es un período arbitrario no recurrente durante el cual se prohíbe el mantenimiento automático. Un clúster puede tener hasta tres exclusiones de mantenimiento a la vez.

Se pueden configurar las exclusiones de mantenimiento y los períodos de mantenimiento por separado y de forma independiente.

Los períodos de mantenimiento permiten controlar cuándo pueden ocurrir las actualizaciones automáticas de los planos de control y nodos a fin de mitigar posibles interrupciones transitorias en las cargas de trabajo. Los períodos de mantenimiento son útiles en distintos tipos de situaciones, por ejemplo:

- Horas de menor demanda: para disminuir las posibilidades de que ocurran tiempos de inactividad, se deberá programar actualizaciones automáticas durante las horas de menor demanda, que es cuando el tráfico es reducido.
- De guardia: asegurarse de que las actualizaciones se realizan durante el horario laboral para que alguien pueda supervisar y administrar cualquier problema inesperado.
- Actualizaciones de varios clústeres: implementar actualizaciones en varios clústeres en diferentes regiones, una por una y en intervalos específicos.

Además de las actualizaciones automáticas, puede que Google necesite realizar otras tareas de mantenimiento y, si es posible, respetará el período de mantenimiento de un clúster.

Si la ejecución de las tareas excede el período de mantenimiento, GKE intenta pausar la operación y reanudarla durante el siguiente período de mantenimiento.

GKE se reserva el derecho de implementar actualizaciones de emergencia sin planificar fuera del período de mantenimiento. Además, las actualizaciones obligatorias para software obsoleto o desactualizado pueden ocurrir de manera automática fuera del período de mantenimiento.

**Nodos**

- Se recomienda inhabilitar la actualización automática de los nodos. Inhabilitar las actualizaciones automáticas de nodos no bloquea la actualización del plano de control de tu clúster.

Los grupos de nodos de un clúster no pueden tener más de dos versiones secundarias anteriores a la versión del plano de control para mantener la compatibilidad con la API del clúster.

- Se recomienda mantener los grupos de nodos actualizados a la versión del clúster.

#### 3.1.2.3. Registro de la actividad de los usuarios

**GKE**

GKE ofrece diferentes formas de acceder a los registros de actividad. Entre ellos están:

- Explorador de registros: Puedes ver los registros directamente desde el Explorador de registros. Para ello, se pueden usar los filtros de registro para seleccionar los recursos de Kubernetes, como los registros de clústeres, nodos, espacios de nombres, pods o registros de contenedores.
- Consola de GKE: En la sección Google Kubernetes Engine de Google Cloud Console, se puede seleccionar los recursos de Kubernetes que se enumeran en Cargas de trabajo y, luego, en los vínculos Contenedor o Registros de auditoría.
- Consola de Cloud Monitoring: Cloud Monitoring recopila medidas de los servicios y los recursos de Google Cloud que se usan.
- Herramienta de línea de comandos de gcloud: Con el comando gcloud logging read, se puede seleccionar el clúster, el nodo, el Pod y los registros del contenedor adecuados.

Cuando GKE escribe los registros del clúster, cada entrada de registro incluye el tipo de recurso.

| Tipo de recurso | Descripción |
|---|---|

| | |
|---|---|
| k8s_cluster | Registros de clústeres de Kubernetes |
| k8s_node | Registros del grupo de nodos de GKE |
| k8s_pod | Registros de los Pods de GKE |
| k8s_container | Registros de contenedores de GKE |

Los registros de auditoría del sistema aparecerán en Cloud Logging con los siguientes nombres:

- proyectos/[ID_DEL_PROYECTO]/logs/cloudaudit.googleapis.com%2Fdata_access: Registros de acceso a los datos
- proyectos/[ID_DEL_PROYECTO]/logs/cloudaudit.googleapis.com%2Factivity: Registros de actividad del administrador
- proyectos/[ ID_DEL_PROYECTO]/logs/events: Registros de acontecimientos

Los contenedores de Kubernetes recopilan registros para las cargas de trabajo escritas en STDOUT y STDERR. Los registros aparecerán en Logging con estos nombres::

- projects/[YOUR_PROJECT_ID]/logs/stderr: Registros escritos en el error estándar
- projects/[YOUR_PROJECT_ID]/logs/stdout: Registros escritos en la salida estándar

Para obtener más detalles sobre los registros de auditoría y su configuración, se puede consultar la siguiente guía del fabricante:
https://cloud.google.com/kubernetes-engine/docs/how-to/audit-logging?hl=es-419

#### 3.1.2.4. Protección de los registros de actividad

El nivel de seguridad ALTO para esta medida exige no sólo el registro de actividades de usuarios del sistema sino su centralización y la automatización de la recolección y correlación de eventos.

Cloud Logging recibe entradas de registro a través de la API de Cloud Logging donde pasan a través del enrutador de registros. Los receptores del enrutador de registros verifican cada entrada de registro con los filtros de inclusión y exclusión existentes que determinan si la entrada de registro se debe enviar a destinos de almacenamiento, incluidos en los depósitos de Cloud Logging, o si se excluye por completo.

Los receptores enrutan las entradas de registro a los destinos de almacenamiento. Cloud Logging proporciona dos receptores de registros predefinidos para cada proyecto de Google Cloud: **_Required** y **_Default**. Todos los registros que se generan en un proyecto de Google Cloud se procesan de forma automática a través de estos

dos receptores de registro y, luego, se almacenan en los depósitos de registro de nombres **_Required** y **_Default** correspondientes.

Se puede obtener más información sobre los procesos de gestión de derechos de acceso en la guía ***CCN 888B Guía de Configuración Segura de Google Cloud Platform*** en la sección ***3.1.2.4 Protección de los registros de actividad***.

### 3.1.3. Continuidad del servicio

**Alta disponibilidad**
Los clústeres regionales permiten aumentar la disponibilidad de las aplicaciones en un clúster porque el plano de control y los nodos del clúster se distribuyen en varias zonas.

- Se recomienda usar el escalador automático del clúster para asegurar de que el clúster pueda manejar la carga requerida en cualquier momento.

Las principales ventajas de un cluster regional son:

- Si una zona de una región experimenta una interrupción, el plano de control del clúster permanece accesible siempre que dos réplicas del plano de control permanezcan disponibles.
- Durante el mantenimiento del clúster, como una actualización de este, solo hay una réplica del plano de control que no está disponible a la vez, y el clúster permanece operativo.

Los clústeres regionales replican el plano de control y los nodos del clúster en varias zonas dentro de una sola región. Por ejemplo, con la configuración predeterminada, un clúster regional en la región us-east1 crea réplicas del plano de control y nodos en tres zonas us-east1: us-east1-b, us-east1-c, y us-east1-d. En el caso de una interrupción de la infraestructura, las cargas de trabajo continúan ejecutándose, y los nodos se pueden volver a balancear de forma manual o mediante el escalador automático del clúster.

Para la implementación de un cluster regional se puede consultar la guía del fabricante:
https://cloud.google.com/kubernetes-engine/docs/how-to/creating-a-regional-cluster

### 3.1.4. Monitorización del sistema

#### 3.1.4.1. Detección de intrusión

Container Threat Detection detecta los ataques más comunes en el entorno de ejecución del contenedor y muestra las alertas dentro del Security Command Center.

La instrumentación de detección de Container Threat Detection puede detectar los siguientes eventos:

- Ejecución de un binario dañino
- Carga de bibliotecas externas
- Shells inversas

| Identificador | Descripción | Detección |
|---|---|---|
| Se ejecutó el objeto binario añadido | Se ejecutó un objeto binario que no era parte de la imagen del contenedor original.<br><br>Si un atacante agrega un objeto binario agregado, es posible que tenga el control de la carga de trabajo y ejecute comandos arbitrarios. | El detector busca un objeto binario en ejecución que no formaba parte de la imagen del contenedor original o se modificó a partir de la imagen del contenedor original. |
| Se cargó la biblioteca agregada | Se cargó una biblioteca que no formaba parte de la imagen del contenedor original.<br><br>Si se carga una biblioteca agregada, es posible que un atacante tenga el control de la carga de trabajo y ejecute un código arbitrario. | El detector busca una biblioteca que se carga y que no formaba parte de la imagen del contenedor original, o que se modificó a partir de la imagen del contenedor original. |
| Shells inversas | Un proceso iniciado con redireccionamiento de transmisión a un socket remoto conectado<br><br>Con una shell inversa, un atacante puede comunicarse desde una carga de trabajo comprometida a una máquina controlada por atacantes. Luego, el atacante puede dirigir y controlar la carga de trabajo para realizar las acciones deseadas, por ejemplo, como parte de un botnet. | El detector busca "stdin" vinculada a un socket remoto. |

Para habilitar la detección de amenazas para contenedores se puede consultar la documentación del fabricante: https://cloud.google.com/security-command-center/docs/how-to-use-container-threat-detection?hl=es-419#enabling

## 3.2. Medidas de protección

### 3.2.1. Protección de las comunicaciones

**GKE**

Existen dos tipos de aislamiento de red para los clústeres: públicos y privados. Los clústeres públicos tienen direcciones IP privadas y públicas en los nodos y solo un extremo público para los nodos del plano de control. Los clústeres privados proporcionan más aislamiento porque solo tienen direcciones IP privadas en los nodos y tienen extremos privados y públicos para los nodos del plano de control. En los clústeres privados, también se puede acceder a las API de Google con el Acceso privado a Google.

- Se recomienda elegir clusters privados para el aislamiento de red.

En un clúster privado, los Pods se aíslan de la comunicación entrante y saliente (el perímetro del clúster). Se pueden controlar estos flujos direccionales mediante la exposición de servicios mediante el balanceo de cargas y Cloud NAT.

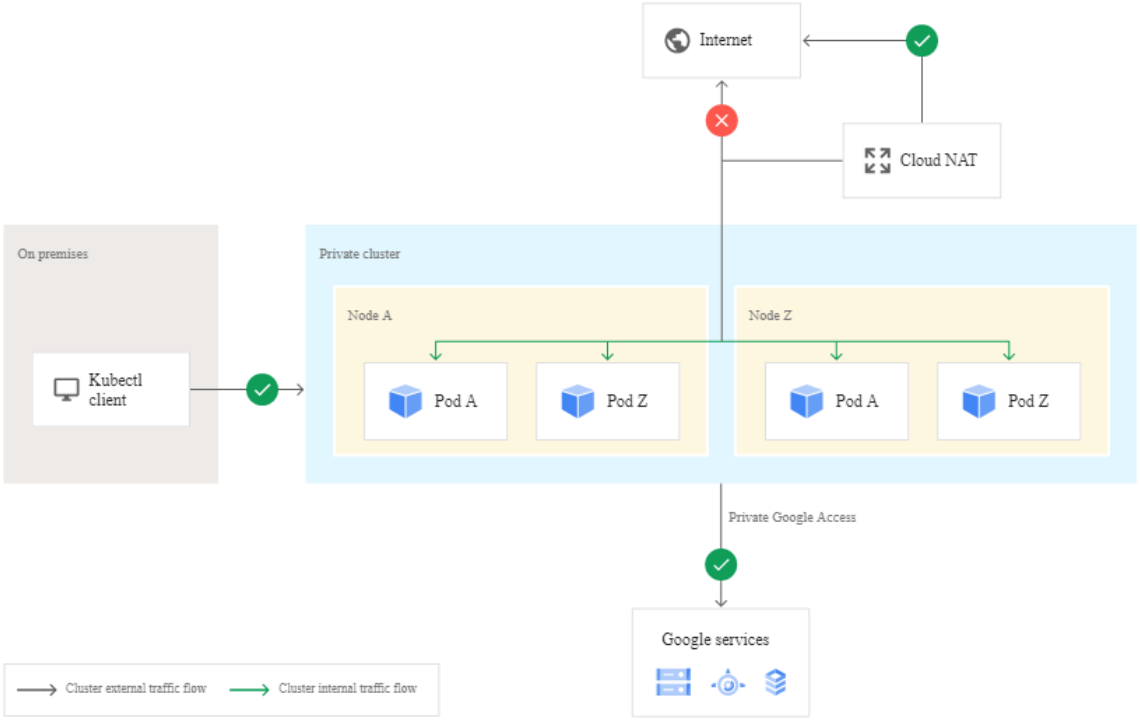


*Ilustración 7 Comunicación de un cluster privado.*

#### 3.2.1.1. Protección de la confidencialidad

**GKE**

Si se desea aceptar solicitudes HTTP en el cluster, el balanceador de cargas de HTTP(S) interno o externo debe tener un certificado para demostrar su identidad a los clientes. También debe tener una clave privada para completar el protocolo de enlace HTTPS.

Cuando el balanceador de cargas acepta una solicitud HTTP(S) de un cliente, el tráfico entre el cliente y el balanceador de cargas se encripta mediante TLS. Sin embargo, el balanceador de cargas finaliza la encriptación TLS y reenvía la solicitud sin encriptación a la aplicación. Cuando se configura un balanceador de cargas de HTTP(S) a través de Ingress, puedes configurarlo para que presente hasta diez certificados TLS al cliente. El recurso Ingress es el que va a definir las reglas que se tienen que cumplir para enrutar la petición a un pod u a otro. Un controlador de Ingress es un Pod o conjunto de Pods que se ejecutan dentro del Cluster y cuya función es asegurarse de que el tráfico entrante se administra del modo que nosotros hayamos especificado.

El balanceador de cargas usa la indicación de nombre del servidor (SNI) para determinar qué certificado presentar al cliente, según el nombre de dominio en el protocolo de enlace TLS. Si el cliente no usa SNI o usa un nombre de dominio que no coincide con el nombre común (CN) en uno de los certificados, el balanceador de cargas usa el primer certificado que se encuentra en el Ingress.

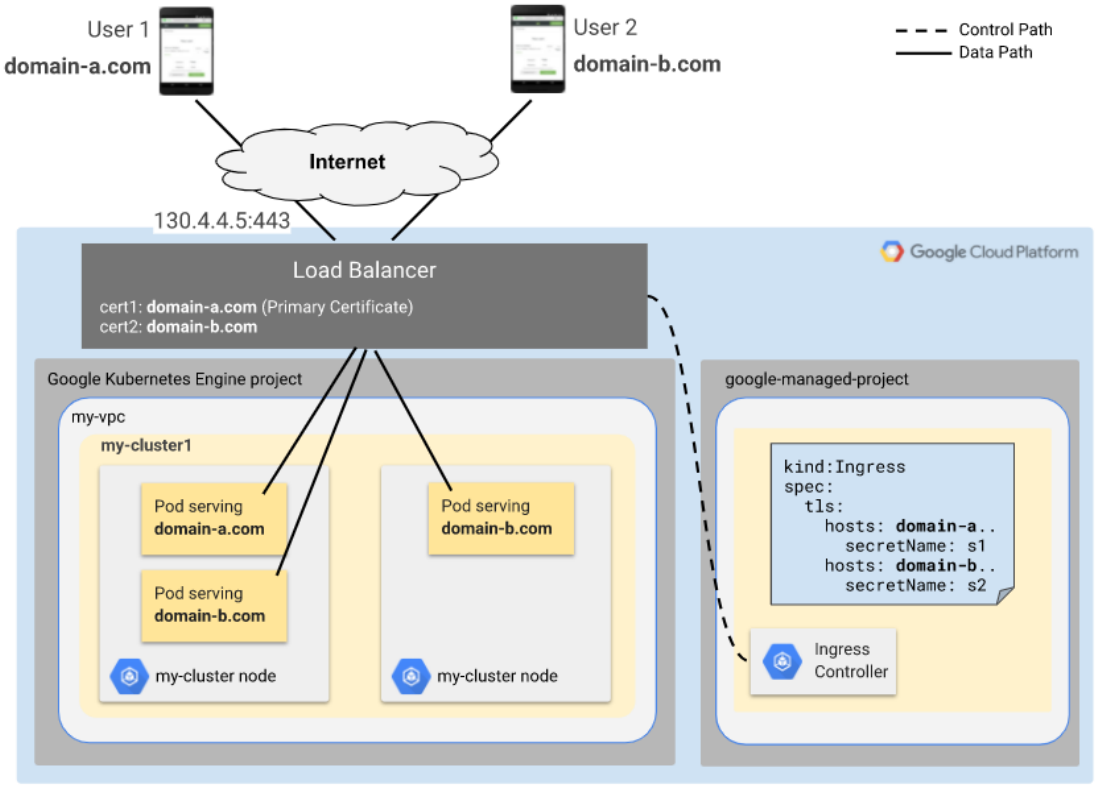


*Ilustración 8 Ejemplo de ingress con múltiples certificados.*

Para la implementación de un certificado con el objeto Ingress de kubernetes, se puede consultar la siguiente guía del fabricante:
https://cloud.google.com/kubernetes-engine/docs/how-to/ingress-multi-ssl?hl=es-419#specifying_certificates_for_your_ingress

**Agente de enmascaramiento de IP**

El enmascaramiento de IP es una forma de traducción de direcciones de red (NAT) que se usa para realizar traducciones de direcciones IP de varios a uno. Esto permite que varios clientes accedan a un destino con una sola dirección IP. Un clúster de GKE usa el enmascaramiento de IP para que los destinos fuera del clúster solo reciban paquetes de direcciones IP de nodo en lugar de direcciones IP de pod.
GKE usa reglas iptables, junto con el DaemonSet de ip-masq-agent, para cambiar la dirección IP de origen de los paquetes que se envían desde los pods a ciertos destinos. Cuando un Pod envía un paquete a una dirección IP de destino en un rango de enmascaramiento especificado, la dirección IP del nodo se usa como dirección de origen del paquete (en lugar de la dirección IP del Pod).

Para configurar la lista se usa el objeto configmap. Un configmap es un objeto de la API utilizado para almacenar datos no confidenciales en el formato clave-valor. Los Pods pueden utilizar los ConfigMaps como variables de entorno, argumentos de la linea de comandos o como ficheros de configuración.

| Configuración del clúster | Comportamiento de la SNAT |
|---|---|
| El ip-masq-agent está instalado en el clúster y se especifica una lista de nonMasqueradeCIDRs en el ConfigMap de ip-masq-agent | GKE conserva las direcciones IP del Pod de origen para los paquetes enviados a los destinos especificados en la lista de nonMasqueradeCIDRs.<br><br>GKE cambia las direcciones IP del Pod de origen por las direcciones IP del nodo de origen para los paquetes enviados a destinos no especificados en la lista de nonMasqueradeCIDRs. |
| ip-masq-agent no está instalado o no especificaste una lista de ip-masq-agent en el ConfigMap de nonMasqueradeCIDRs, y se creó el clúster sin la marca --disable-default-snat | GKE conserva las direcciones IP del Pod de origen para los paquetes enviados a un conjunto de destinos predeterminados sin enmascarar. Estos destinos predeterminados dependen de la versión de GKE y del tipo de imagen de nodo.<br><br>GKE cambia las direcciones IP del Pod de origen por las direcciones IP de nodo de origen para los paquetes enviados a destinos fuera de los destinos predeterminados sin enmascarar. |
| ip-masq-agent no está instalado o no especificaste una lista de | GKE conserva las direcciones IP del Pod de origen para los paquetes |

| | |
|---|---|
| nonMasqueradeCIDRs en el ConfigMap de ip-masq-agent, y se creó el clúster con la marca --disable-default-snat | enviados a todos los destinos.<br><br>Para cambiar este comportamiento, hay que asegurarse de que ip-masq-agent esté instalado y que especificaste una lista de nonMasqueradeCIDRs en el ConfigMap de ip-masq-agent. |

Se puede obtener más información sobre la configuración de este agente en la web del fabricante:
https://cloud.google.com/kubernetes-engine/docs/how-to/ip-masquerade-agent#specify-non-masq-cidrs

#### 3.2.1.2. Segregación de redes

**GKE**

A la hora de crear un cluster de GKE se permite la creación de dos tipos de cluster en cuanto a la forma de enrutar el tráfico, basados en rutas o clúster nativo de la VPC.

En un clúster basado en rutas, a cada nodo se le asigna un rango de direcciones IP de /24 para los pods. Con un rango /24, existen 256 direcciones, pero la cantidad máxima de pods por nodo es de 110. Dado que hay dos direcciones IP disponibles por cada pod posible, Kubernetes puede mitigar la reutilización de la dirección IP a medida que los pods se agregan y quitan de un nodo.

Un clúster que usa rangos de direcciones IP de alias se denomina clúster nativo de la VPC. Si solo se ejecuta un servicio en una VM, se pueden hacer peticiones mediante la dirección IP principal de la interfaz. Si se ejecutan múltiples servicios en una VM, se tiene la opción de asignarle a cada uno una dirección IP interna distinta. Se puede hacer esto con rangos de IP de alias.

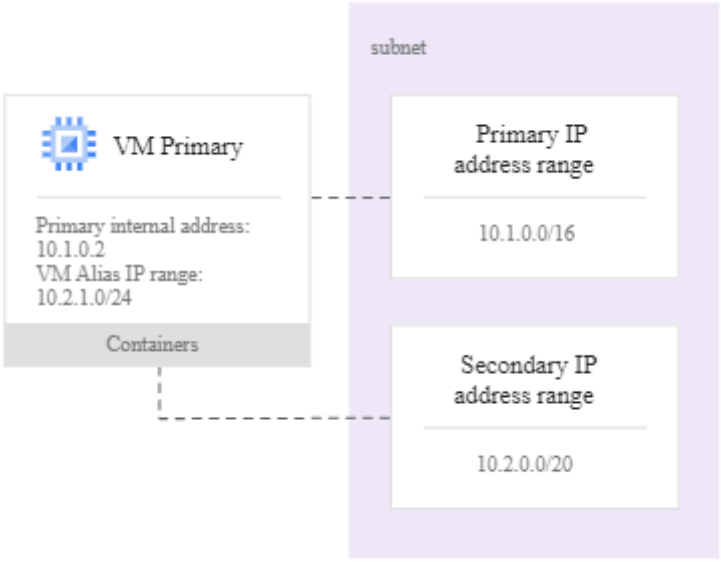


*Ilustración 9 Ejemplo de uso de alias de IP.*

- Se recomienda elegir un clúster nativo de la VPC porque usa rangos de direcciones IP de alias en los nodos de GKE y escala con mayor facilidad que los clústeres basados en rutas.

Para la implementación de un cluster nativo de la VPC se puede consultar la siguiente guía del fabricante:
https://cloud.google.com/kubernetes-engine/docs/how-to/alias-ips?hl=es

Los clústeres nativos de la VPC son necesarios para los clústeres de GKE privados y a fin de crear en las VPC compartidas. En el caso de los clusters creados en modo Autopilot, el modo nativo de la VPC siempre está activado y no se puede desactivar.

Los clústeres nativos de la VPC escalan con mayor facilidad que los clústeres basados en rutas sin consumir rutas de Google Cloud y, por lo tanto, son menos susceptibles de los límites de enrutamiento. Las ventajas de usar clústeres nativos de la VPC van de la mano con la compatibilidad de alias de IP. Por ejemplo, los grupos de extremos de red (NEG) solo se pueden usar con direcciones IP secundarias, por lo que solo se admiten en clústeres nativos de la VPC.

Los clústeres de GKE requieren una planificación cuidadosa de direcciones IP. En la arquitectura de la red de VPC compartida, un administrador de red puede crear subredes y compartirlas con miembros y funciones específicas. Luego, se pueden crear clústeres de GKE en proyectos de servicio de esas subredes.

En general, una red de VPC compartida es una arquitectura que es adecuada para la mayoría de las organizaciones con un equipo de administración centralizada.

- Se recomienda usar redes de VPC compartida a fin de crear las subredes para los clústeres de GKE y evitar conflictos de direcciones IP en la organización.

Para la implementación de una red VPC compartida, se puede consultar la siguiente guía del fabricante:
https://cloud.google.com/kubernetes-engine/docs/how-to/cluster-shared-vpc/?hl=es_419

### 3.2.2. Protección de la información

#### 3.2.2.1. Cifrado de la información

**Container Registry**
Container Registry almacena las imágenes de contenedor en Cloud Storage. Cloud Storage siempre encripta los datos en el lado del servidor.

Si se requiere de requisitos normativos o de cumplimiento, se pueden encriptar las imágenes de contenedor mediante las claves de encriptación administradas por el cliente (CMEK). Las claves CMEK se administran en Cloud Key Management Service. Cuando se usa CMEK, mediante la inhabilitación o destrucción de la clave puedes inhabilitar de forma temporal o permanente el acceso a una imagen de contenedor encriptada.

Container Registry no está integrado de forma directa en Cloud KMS. En su lugar, es compatible con CMEK cuando se almacenan las imágenes de contenedor en depósitos de almacenamiento configurados para usar CMEK. Para configurar dentro de Cloud Storage el uso de CMEK se puede seguir la siguiente guía del fabricante:
https://cloud.google.com/storage/docs/encryption/using-customer-managed-keys?hl=es

### 3.2.3. Protección de los servicios

#### 3.2.3.1. Protección de servicios y aplicaciones web

**Container Analysis**
Container Analysis proporciona análisis de vulnerabilidades y almacenamiento de metadatos para contenedores a través de Container Analysis. El servicio de análisis realiza análisis de vulnerabilidades en imágenes de Container Registry. Luego, almacena los metadatos resultantes y los pone a disposición mediante una API.

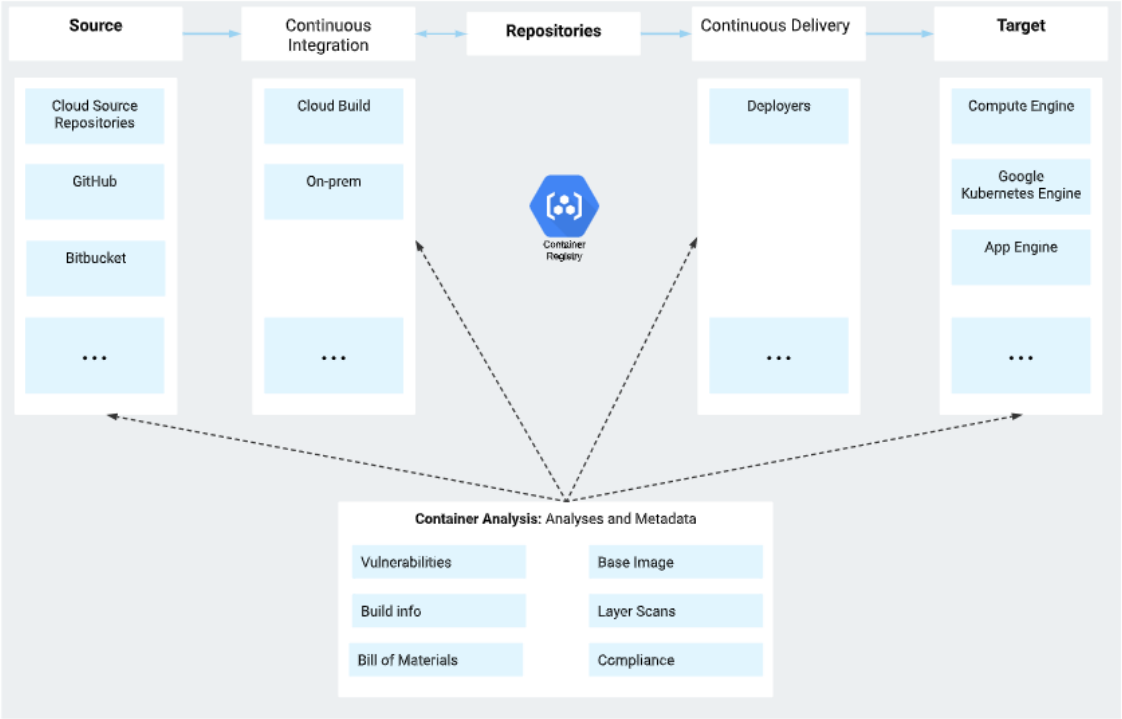


*Ilustración 10 Diagrama de ejemplo de diferentes servicios publicando en Container Registry y analizando la imagen.*

El análisis de vulnerabilidades puede ocurrir de forma automática o de manera manual:

- Cuando se habilita el análisis automático, el análisis se activa de forma automática cada vez que se envía una imagen nueva a Container Registry. La información de vulnerabilidad se actualiza continuamente cuando se descubren nuevas vulnerabilidades.
- Cuando se habilita el análisis de manera manual. El análisis manual brinda más flexibilidad cuando analiza contenedores. Por ejemplo, se puede analizar una imagen compilada de forma local y solucionar las vulnerabilidades antes de almacenarla en un registro.

Para habilitar Container Analysis se puede seguir la siguiente guía del fabricante: https://cloud.google.com/container-analysis/docs/enabling-disabling-container-analysis?hl=es-419

**Google Cloud Armor para Ingress**

Con las políticas de seguridad de Google Cloud Armor, se pueden proteger las aplicaciones que usan el balanceo de cargas de HTTP(S) externo contra ataques de DSD y otros ataques basados en la Web mediante el bloqueo de ese tráfico en la red perimetral. En GKE, se pueden habilitar las políticas de seguridad de Google Cloud Armor para las aplicaciones mediante Ingress para balanceo de cargas HTTP(S) externo y agrega una política de seguridad al BackendConfig que se adjunta al objeto Ingress.

Para la implementación de esta política de seguridad se puede seguir la siguiente guía del fabricante:

https://cloud.google.com/kubernetes-engine/docs/how-to/ingress-features?hl=es-419#cloud_armor

## 4. GLOSARIO DE TÉRMINOS

A continuación, se describen los términos, acrónimos y abreviaturas relacionados con la tecnología objeto de esta guía con el objeto de facilitar la comprensión de la misma

| Término | Definición |
|---|---|
| **Cloud KMS** | Cloud Key Management Service |
| **DSD** | Denegación de servicio distribuidos |
| **ENS** | Esquema Nacional de Seguridad |
| **FaaS** | Funciones como servicio |
| **GCE** | Google Compute Engine |
| **GCF** | Google Cloud Functions |
| **GCP** | Google Cloud Platform |
| **GCS** | Google Cloud Storage |
| **GKE** | Google Kubernetes Engine |
| **GW** | Google Workspace |
| **IAM** | Identity and Access Management |
| **IAP** | Identity-Aware Proxy |
| **MFA** | Autenticación de múltiples factores |
| **RBAC** | Acceso basado en roles |
| **SDN** | Redes definidas por software |
| **SSL** | Secure Sockets Layer |
| **TLS** | Seguridad de la capa de transporte |
| **VM** | Virtual Machine (máquina virtual) |
| **VPC** | Virtual Private Cloud |
| **WAF** | Web application firewall |
| **NEG** | Un grupo de extremos de red (NEG) es un objeto de configuración que especifica un grupo de extremos o servicios de backend. Un caso práctico común para esta configuración es la implementación de servicios en contenedores. También puedes distribuir el tráfico de manera detallada a las aplicaciones que se ejecutan en las instancias de backend. |

## 5. GLOSARIO DE SERVICIOS GCP

A continuación, se reúnen los diferentes servicios mencionados a lo largo de esta guía incluyendo enlaces a la documentación concreta de cada uno de ellos.

| Servicio | URL de documentación del servicio |
|---|---|
| Cloud Asset Inventory | https://cloud.google.com/asset-inventory |
| Cloud Logging | https://cloud.google.com/logging |
| Cloud Monitoring | https://cloud.google.com/monitoring |
| Container Threat Detection | https://cloud.google.com/security-command-center/docs/concepts-container-threat-detection-overview?hl=es-419 |
| Event Threat Detection | https://cloud.google.com/security-command-center/docs/concepts-event-threat-detection-overview?hl=es-419 |
| Firewall Insights | https://cloud.google.com/network-intelligence-center/docs/firewall-insights |
| Google Cloud Armor | https://cloud.google.com/armor?hl=es |
| Google Cloud Functions | https://cloud.google.com/functions?hl=es |
| Google Cloud SQL | https://cloud.google.com/sql?hl=es |
| Google Cloud Storage | https://cloud.google.com/storage |
| Google Compute Engine | https://cloud.google.com/compute?hl=es |
| Google Compute Engine Persistent Disks | https://cloud.google.com/persistent-disk?hl=es |
| Google Docs Editors | https://www.google.es/intl/es/docs/about/ |
| Google Kubernetes Engine | https://cloud.google.com/kubernetes-engine?hl=es-419 |
| Google VPC | https://cloud.google.com/vpc?hl=es-419 |
| Google Workspace | https://workspace.google.com/intl/es-419_ar/ |
| Identity-Aware Proxy | https://cloud.google.com/iap |
| Security Command Center | https://cloud.google.com/security-command-center?hl=es |

PUBLICACIONES DE DEFENSA

GOBIERNO DE ESPAÑA MINISTERIO DE DEFENSA
SUBSECRETARÍA DE DEFENSA
SECRETARÍA GENERAL TÉCNICA
SUBDIRECCIÓN GENERAL DE PUBLICACIONES Y PATRIMONIO CULTURAL