

Contrato Interadministrativo Nro. 3021063 de 2021
Suscrito Entre la Empresa de Telecomunicaciones de Bogotá y el Fondo Financiero
Distrital de Salud - Secretaría Distrital de Salud

Fase 3 del Proyecto
Grupo de Entregables E20 – Infraestructura, Soporte, Mantenimiento y Operación

Aprovisionamiento, Despliegue y Configuración de Infraestructura de Nuevos Desarrollos

Código:
SISEM-3-E20-5

Bogotá D.C. 22 de noviembre 2023

FICHA DE IDENTIFICACIÓN DEL DOCUMENTO

Título del Documento	“Aprovisionamiento, Despliegue y Configuración de Infraestructura de Nuevos Desarrollos”	
Resumen	Este entregable presenta el plan de alta disponibilidad y despliegue, configuración del monitoreo y gestión de los ambientes del proyecto SISEM. Este documento es dinámico por lo que se versionara a medida que se vayan ejecutando las fases del plan de Aprovisionamiento, gestión y monitoreo de la Infraestructura y se generen productos a partir de la actualización de la plataforma SISEM – EMB. El entregable forma parte de las actividades de Análisis, Diseño y Arquitectura del SISEM, de la Fase 3 del proyecto.	
Código	SISEM-3-E20-5	
Autor	ETB	
Estado	V1	22/11/2023
Revisado por		

CONTROL DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
V1	Versión inicial	José Peña / Fernando Rocha	22/11/2023

DERECHOS DE AUTOR

Derechos de autor
ETB. Todo el material de este entregable puede ser utilizado libremente por la SDS y las personas a quienes ésta asigne en el marco del convenio citado.

CRÉDITOS

Créditos
Grupos de Diseño de Arquitecturas de ETB.

Tabla de Contenido

1.	INTRODUCCIÓN	8
2.	ALCANCE	9
3.	AUDIENCIA	10
4.	DEFINICIONES Y CONCEPTOS	11
5.	METODOLOGÍA DE DESARROLLO DEL DOCUMENTO	13
5.1.	HERRAMIENTAS UTILIZADAS	13
5.2.	OTRAS CONSIDERACIONES PREVIAS	13
6.	CONCEPTUALIZACIÓN	13
6.1.	ALTA DISPONIBILIDAD	13
6.2.	DESCRIPCIÓN DE LA CONFIGURACIÓN DE ALTA DISPONIBILIDAD BASES DE DATOS.....	14
6.3.	DESCRIPCIÓN DE LA CONFIGURACIÓN DE ALTA DISPONIBILIDAD EQUIPOS DE COMPUTO	15
6.4.	SCRIPTS DE INICIO Y APAGADO.....	16
6.5.	ALTA DISPONIBILIDAD DE SISEM CON KUBERNETES.....	17
7.	GESTIÓN DE IDENTIDADES Y ACCESOS	18
8.	ADMINISTRACIÓN Y MONITOREO	18
8.1.	MONITOREO GENERAL DE INFRAESTRUCTURA DEL AMBIENTE	18
8.2.	MONITOREO COMPUTE ENGINE	20
8.2.1.	CPU	21
8.2.2.	PROCESOS	21
8.2.3.	MEMORIA	21
8.2.4.	RED.....	22
8.2.5.	DISCO	23
8.3.	MONITOREO A NIVEL DE API.....	24
8.4.	MONITOREO A NIVEL DE BASE DE DATOS	25
8.5.	MONITOREO GKE.....	26
8.6.	NAMESPACES.....	26
8.7.	NODOS	27
8.8.	CARGAS DE TRABAJO	27
8.9.	PODS.....	27
8.10.	CONTENEDORES.....	28
9.	ALERTAS.....	28
10.	HARDENING	30
10.1.	HARDENING A NIVEL CLOUD	30
10.2.	HARDENING A NIVEL DE BASE DE DATOS.....	30
11.	IPV6.....	31

12.	CONTINUIDAD DE NEGOCIO Y RECUPERACIÓN ANTE DESASTRES	32
13.	AMBIENTE DE DESARROLLO (DEV).....	32
13.1.	MÁQUINAS Y SERVICIOS	33
13.1.1.	ANÁLISIS-DATOS	33
13.1.2.	DEMOSIDCRUE	33
13.1.3.	DEV-SISEM.....	34
13.1.4.	WINDOWS-POWER-BI.....	34
13.1.5.	WINSERVER	35
13.2.	VARIABLES DE ENTORNO	37
13.3.	TOPOLOGÍA DE RED	38
13.4.	TOPOLOGÍA INFRAESTRUCTURA	38
13.4.1.	TOPOLOGÍA DESARROLLO (DEV)	38
13.4.2.	TOPOLOGÍA PREPRODUCCIÓN	39
13.4.3.	TOPOLOGÍA PRUEBAS (QA)	40
13.4.4.	TOPOLOGÍA DESARROLLO MÓVIL	41
13.5.	FIREWALL Y PUERTOS	42
13.6.	SEGURIDAD	45
13.6.1.	SERVICIO CONTRA ATAQUE DDOS	46
13.6.1.1.	CREACIÓN DE LA REGLA DDOS.....	47
13.6.1.2.	POLÍTICAS CONFIGURADAS.....	48
13.6.2.	BALANCEADOR DE CARGA	48
13.6.3.	FIREWALL VPC GOOGLE	49
13.6.4.	HTTPS/SSL.....	49
13.6.5.	FIREWALL NODO	50
13.6.6.	ENRUTAMIENTO KUBERNETES Y API GATEWAY	51
13.6.7.	POD.....	51
13.6.8.	VALIDACIÓN LLAVE PÚBLICA.....	52
13.6.9.	VALIDACIÓN AUTENTICACIÓN	52
13.6.10.	VALIDACIÓN AUTORIZACIÓN.....	52
14.	AMBIENTE DE PRODUCCIÓN	53
14.1.	PROYECTO	53
14.2.	TOPOLOGÍA INFRAESTRUCTURA PRODUCCIÓN.....	54
14.3.	CLUSTER.....	55
14.4.	NODOS	56
14.5.	RESERVAR IP	57
14.6.	CLOUD VPC	57
14.7.	TOPOLOGÍA DE RED	58

14.8. SEGURIDAD 58

14.8.1. FIREWALL 59

14.8.2. CLOUD ARMOR 59

15. REFERENCIAS 61

Tabla de Tablas

TABLA 1.	CONFIGURACIÓN ANÁLISIS-DATOS	33
TABLA 2.	CONFIGURACIÓN DEMOSIDCRUE	34
TABLA 3.	CONFIGURACIÓN DEV-SISEM	34
TABLA 4.	WINDOWS-POWER-BI	35
TABLA 5.	WINSERVER	35

Tabla de Ilustraciones

ILUSTRACIÓN 1 GOOGLE CLOUD SQL	15
ILUSTRACIÓN 2 VIRTUAL PRIVATE CLOUD	17
ILUSTRACIÓN 3 MONITOREO GENERAL	19
ILUSTRACIÓN 4 MONITOREO POR REGIÓN, CPU Y MEMORIA.....	19
ILUSTRACIÓN 5 MONITOREO DE RED	20
ILUSTRACIÓN 6 MONITOREO DE DISCO.....	20
ILUSTRACIÓN 7 MONITOREO DE PROCEDIMIENTOS	20
ILUSTRACIÓN 8 MONITOREO DE CPU POR INSTANCIA	21
ILUSTRACIÓN 9 MONITOREO DE PROCESOS DE CPU POR INSTANCIA.....	21
ILUSTRACIÓN 10 MONITOREO DE MEMORIA POR INSTANCIA.....	22
ILUSTRACIÓN 11 MONITOREO DE RED POR INSTANCIA	23
ILUSTRACIÓN 12 MONITOREO DE DISCO POR INSTANCIA	23
ILUSTRACIÓN 13 REGISTROS POR INSTANCIA.....	24
ILUSTRACIÓN 14 MONITOREO A NIVEL DE API	25
ILUSTRACIÓN 15 MONITOREO A NIVEL DE DB	26
ILUSTRACIÓN 16 MONITOREO DE CLUSTER GKE	26
ILUSTRACIÓN 17 MONITOREO DE NAMESPACES GKE	27
ILUSTRACIÓN 18 MONITOREO DE NODOS GKE	27
ILUSTRACIÓN 19 MONITOREO DE CARGAS DE TRABAJO GKE	27
ILUSTRACIÓN 20 MONITOREO DE PODS GKE	28
ILUSTRACIÓN 21 MONITOREO DE CONTENEDORES GKE.....	28
ILUSTRACIÓN 22 ALERTA CPU.....	29
ILUSTRACIÓN 23 ALERTA DISCO	29
ILUSTRACIÓN 24 CARACTERÍSTICAS GENERALES GKE AMBIENTES DESARROLLO, DESARROLLO MÓVIL, PREPRODUCCIÓN Y PRUEBAS (QA)	32
ILUSTRACIÓN 25 INSTANCIAS CLOUD SQL ENTORNO DESARROLLO	36
ILUSTRACIÓN 26 TOPOLOGÍA DE RED AMBIENTE DESARROLLO (DEV)	38
ILUSTRACIÓN 27 FLUJO GENERAL.....	45
ILUSTRACIÓN 28 DDOS	46
ILUSTRACIÓN 29 DDOS MULTICAPA	46
ILUSTRACIÓN 30 CONFIGURACIÓN DE POLÍTICAS EN CLOUD ARMOR	47
ILUSTRACIÓN 31 SECURITY POLICIES - SISEM PREPRODUCCIÓN.....	48
ILUSTRACIÓN 32 INGRESS CONTROLLER (1).....	48
ILUSTRACIÓN 33 INGRESS CONTROLLER (2).....	49
ILUSTRACIÓN 34 FIREWALL	49
ILUSTRACIÓN 35 EJ CERTIFICADO PREPROD.....	50
ILUSTRACIÓN 36 FIREWALL NODO	50
ILUSTRACIÓN 37 GATEWAY SERVICE	51
ILUSTRACIÓN 38 PODS DEL SISTEMA	51
ILUSTRACIÓN 39 CONFIGURACIÓN PUERTO POD.....	52
ILUSTRACIÓN 40 CREACIÓN DE PERMISO KEYCLOAK	53
ILUSTRACIÓN 41 ADMINISTRACIÓN DE ROLES KEYCLOAK	53
ILUSTRACIÓN 42 PROYECTO SISEM	54
ILUSTRACIÓN 43 CLUSTER	56
ILUSTRACIÓN 44 NODOS PRODUCCIÓN	57
ILUSTRACIÓN 45 RESERVA IP ESTÁTICA.....	57
ILUSTRACIÓN 46 CLOUD VPC PRODUCCIÓN.....	58
ILUSTRACIÓN 47 FIREWALL PRODUCCIÓN.....	59
ILUSTRACIÓN 48 WAF ARMOR-SISEM-PROD.....	60
ILUSTRACIÓN 49 REGLAS DEFINIDAS EN WAF	60

1. Introducción

El presente documento tiene como objetivo principal detallar la infraestructura y la gestión operativa del entorno del Sistema de Emergencias Médicas (SISEM). Se centra en aspectos de alta disponibilidad, gestión de identidades y accesos, administración y monitoreo, seguridad, continuidad del negocio, desarrollo y producción, entre otros aspectos técnicos.

En el apartado de "Conceptualización", se aborda la arquitectura de alta disponibilidad, detallando la configuración de la infraestructura y los sistemas para asegurar la continuidad operativa del SISEM.

La sección de "Administración y Monitoreo" presenta los diferentes niveles de monitoreo, desde la infraestructura general hasta los detalles específicos de cada componente, incluyendo Compute Engine, API, bases de datos y Kubernetes (GKE), garantizando un control sobre el rendimiento y la disponibilidad del entorno.

Los apartados posteriores abordan áreas como la seguridad (Hardening, IPv6, Firewall), la continuidad del negocio y la recuperación ante desastres, así como los entornos de desarrollo y producción, proporcionando una visión integral de las configuraciones, topologías y políticas de seguridad implementadas.

2. Alcance

Este documento abarca un análisis de la infraestructura, configuración y políticas implementadas en el entorno del Sistema de Emergencias Médicas (SISEM). Se centra en aspectos técnicos relacionados con la alta disponibilidad, gestión de identidades y accesos, monitoreo, seguridad, continuidad del negocio, entornos de desarrollo y producción, entre otros elementos cruciales para el correcto funcionamiento y la estabilidad del sistema.

En su contenido, se encuentran descripciones completas de la configuración de alta disponibilidad tanto para las bases de datos como para los equipos de cómputo, además de la implementación específica en Kubernetes para asegurar la escalabilidad y la confiabilidad del SISEM.

El documento también proporciona una descripción de los entornos de desarrollo (Dev) y producción, mostrando las topologías de red, la configuración de seguridad y las variables de entorno implementadas. Esto, combinado con la información detallada sobre la alta disponibilidad, Hardening, IPv6, y niveles de monitoreo y alertas configuradas, conforma una visión integral de la gestión y configuración de la infraestructura en cada fase del proyecto SISEM.

3. Audiencia

Este documento está dirigido a los usuarios técnicos del sistema y más específicamente a aquellos que se desempeñan en la Dirección de Tecnologías de la Información y la Comunicación - TIC, en su calidad de solicitantes de servicios, procesos, datos, información y demás herramientas funcionales y no funcionales que les debe proveer la solución contratada, en el marco del SISEM.

El propósito del entregable es validar la completitud y pertinencia de las actividades realizadas por el contratista respecto del diseño de las arquitecturas del SISEM, para sus componentes de software y de infraestructura.

4. Definiciones y Conceptos

Alta disponibilidad	Es un protocolo de diseño del sistema y su implementación asociada que asegura un cierto grado absoluto de continuidad operacional durante un período de medición dado. Alta disponibilidad no solo implica realizar replicación de datos, sino además contar con mecanismos de detección y recuperación ante fallas eventuales o permanentes.
Docker	Docker es un proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización a nivel de sistema operativo en Linux. Docker permite una verdadera independencia entre las aplicaciones, la infraestructura, los desarrolladores y las operaciones de TI, para desbloquear su potencial y crea un modelo para una mejor colaboración e innovación. [2] Características: Portabilidad, Ligereza, Autosuficiencia.
Kubernetes	Kubernetes es una plataforma portable y extensible de código abierto para administrar cargas de trabajo y servicios. Kubernetes facilita la automatización y la configuración declarativa. Kubernetes ofrece un entorno de administración centrado en contenedores. Orquesta la infraestructura de cómputo, redes y almacenamiento para que las cargas de trabajo de los usuarios no tengan que hacerlo. Además de esto, permite la portabilidad entre proveedores de infraestructura.[3]
Hardening	Hardening (palabra en inglés que significa endurecimiento) en seguridad informática es un método o proceso de asegurar el sistema por medio de la reducción de vulnerabilidades de este.

Firewall	Un cortafuegos (o firewall en inglés) es un elemento de hardware o software que se utiliza en una red de computadoras para controlar las comunicaciones, permitiéndolas o prohibiéndolas según las políticas de red que haya definido la organización responsable de la red. Su modo de funcionamiento es indicado por la recomendación RFC 2979, que define las características de comportamiento y requerimientos de interoperabilidad. [4]
IPv6	IPv6 es el protocolo de Internet versión 6 (IP, Internet Protocol) que permite conectar diversos dispositivos a internet, identificándolos con una dirección única. Este protocolo viene a sustituir al IPv4 mucho más limitado en cuanto al número de direcciones IPs disponibles. [5]
Balanceo de Carga	El balanceo de carga es una estrategia que permite la redistribución de tráfico que se solía hacer a un servidor, a distintos servidores réplica que cumplen con el mismo servicio para así prevenir que se generen retrasos debido a congestión.
Backup	El término Backup hace referencia a una copia de seguridad de la información de una empresa, proyecto o aplicación. Estas copias de seguridad permiten tener un plan de acción en caso de que se produzca algún problema en el que se vea involucrada la disponibilidad de la información.
Compute Engine	Un servicio de computación seguro y personalizable con el que puedes crear y ejecutar máquinas virtuales en la infraestructura de Google. [6]
GKE	Las aplicaciones de Kubernetes son soluciones en contenedores creadas para el entorno empresarial que ofrecen plantillas de despliegue predefinidas, portabilidad, asignación de licencias simplificada y facturación consolidada. [7]
VPC	La nube privada virtual (VPC) de Google Cloud ofrece funcionalidad de red a las instancias de máquinas virtuales (VM) de Compute Engine, los contenedores de Google Kubernetes Engine (GKE) y el entorno flexible de App Engine. La VPC brinda herramientas de redes globales, escalables y flexibles a los servicios basados en la nube. [8]
Cluster	Un clúster de Kubernetes es un conjunto de máquinas de nodos que ejecutan aplicaciones en contenedores.

Nodo	Un nodo es una máquina de trabajo en Kubernetes, previamente conocida como minion. Un nodo puede ser una máquina virtual o física, dependiendo del tipo de clúster. [9]
Pod	Un Pod es un grupo de uno o más contenedores (como contenedores Docker), con almacenamiento/red compartidos y unas especificaciones de cómo ejecutar los contenedores. Los contenidos de un Pod son siempre coubicados, coprogramados y ejecutados en un contexto compartido. [10]

5. Metodología de Desarrollo del Documento

Para facilitar la lectura del documento, se ha dividido en las siguientes secciones:

- **Alta Disponibilidad:** Se muestran las características implementadas que apoyan a un sistema de alta disponibilidad como SISEM en distintos niveles (bases de datos, instancias de compute engine, kubernetes y más)
- **Gestión y Monitoreo:** En el cual se muestra la configuración y métricas que se tiene establecidas en lo correspondiente al monitoreo de la aplicación a distintos niveles (infraestructura general, instancias, API, base de datos, registros, alertas y más)
- **Ambientes de Trabajo:** Análisis detallado del aprovisionamiento y configuración de los ambientes de trabajo (desarrollo, pruebas, preproducción y producción) de SISEM

5.1. Herramientas Utilizadas

Se han utilizado las interfaces que nos dispone la nube, para nuestro caso Google Cloud Platform.

5.2. Otras Consideraciones Previas

Esta versión del documento plan de Alta Disponibilidad, se basa en un análisis inicial para el despliegue de los conceptos y patrones de arquitectura que se emplearán en el desarrollo de la aplicación, siguiendo las recomendaciones indicadas por el cliente.

6. Conceptualización

6.1. Alta Disponibilidad

Esta variación implica que, en una ejecución de operaciones de lectura o escritura, los primeros N nodos activos o “saludables” de una lista de preferencias para un ítem de datos son tomados en cuenta. Estos nodos no necesariamente son los primeros N en sentido horario alrededor del anillo de Consistent Hashing. [1]

Implica mantener un registro sobre los nodos consultados y que no respondieron. Las notificaciones se realizan a través de hints (indicios) que se incluyen en la meta data. [1]

6.2. Descripción de la Configuración de Alta Disponibilidad Bases de Datos

El propósito de una configuración de Alta Disponibilidad es reducir el tiempo de inactividad cuando una zona o instancia deja de estar disponible. Esto puede suceder durante una interrupción zonal o cuando una instancia se daña. Con Alta Disponibilidad, los datos siguen estando disponibles para las aplicaciones del cliente.

La configuración de Alta Disponibilidad, a veces denominada clúster, proporciona redundancia de datos. Una instancia de Cloud SQL configurada para alta disponibilidad también se denomina instancia regional, se encuentra en una zona principal y secundaria dentro de la región configurada. Dentro de una instancia regional, la configuración se compone de una instancia principal y una instancia en espera. A través de la replicación síncrona en el disco persistente de cada zona, todas las escrituras realizadas en la instancia principal se replican en los discos de ambas zonas antes de que se notifique que una transacción está confirmada. En caso de que falle una instancia o una zona, la instancia en espera se convierte en la nueva instancia principal. A continuación, los usuarios se redirigen a la nueva instancia principal. Este proceso se llama conmutación por error.

Después de una conmutación por error, la instancia que recibió la conmutación por error sigue siendo la instancia principal, incluso después de que la instancia original vuelva a estar en línea. Una vez que la zona o instancia que experimentó una interrupción vuelve a estar disponible, la instancia principal original se destruye y se vuelve a crear. Luego se convierte en la nueva instancia en espera. Si se produce una conmutación por error en el futuro, la nueva instancia principal se conmutará por error a la instancia original en la zona original.

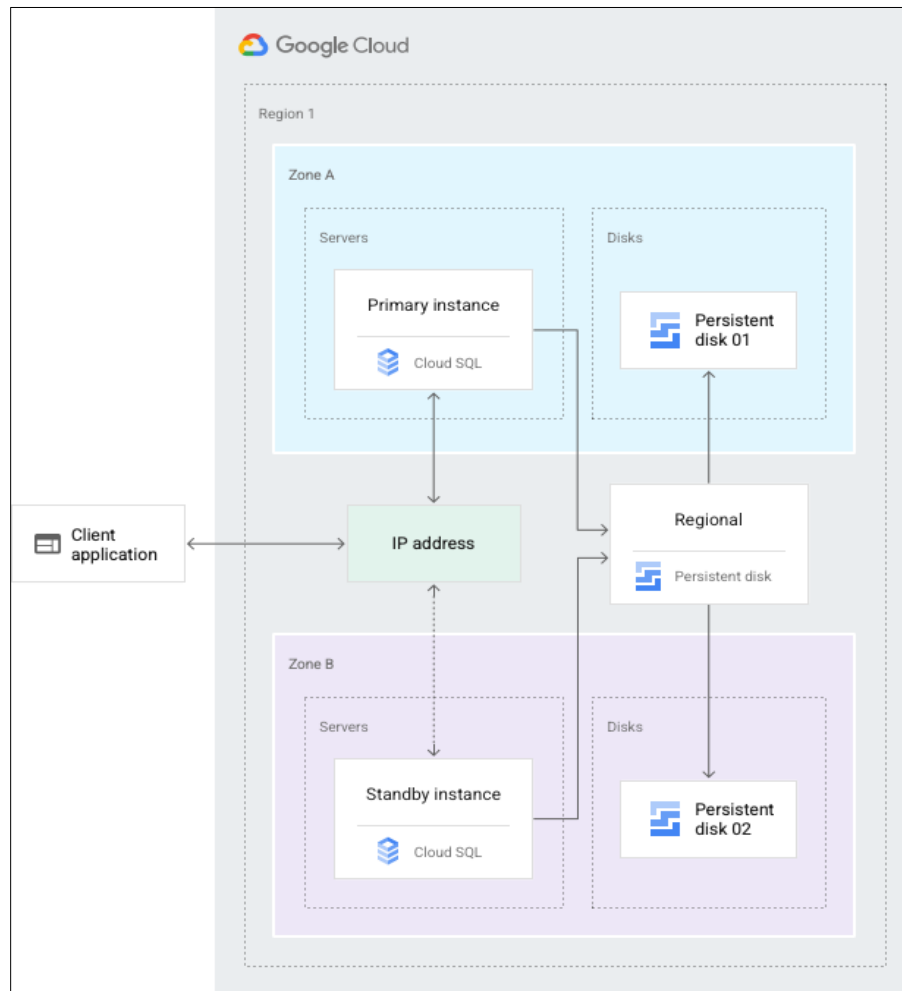


Ilustración 1 Google Cloud SQL

6.3. Descripción de la Configuración de Alta Disponibilidad Equipos de Computo

Para minimizar las fallas, se diseñan las máquinas de trabajo de kubernetes (Instancias de Compute Engine) para que sean tolerantes a errores, fallas de red y desastres inesperados. Un sistema robusto que es capaz de manejar los errores correctamente, por ejemplo; al redirigir el tráfico de la instancia fallida a una instancia activa y poder ejecutarse automáticamente después de un reinicio, esto con el fin de no permitir perder disponibilidad en dado de presentarse un fallo.

También se realiza un diseño de un sistema tolerante a errores, donde se crean instancias de máquinas virtuales en al menos dos zonas de disponibilidad ubicadas en dos regiones. Esto asegura que SISEM continúe su funcionamiento de manera correcta independientemente de:

- Falla en una región completa
- Falla en una zona completa

Para el proyecto SISEM se proporcionan grupos de instancias administradas, un grupo de instancias de máquinas virtuales que tienen un propósito común (máquinas de trabajo o nodos de kubernetes). Con un grupo de instancias administradas, el tráfico se puede enrutar a varias

máquinas virtuales a través del balanceador de carga, por lo que, si falla alguna instancia individual, el servicio no se interrumpe.

Estos grupos de instancias administrados también brindan las siguientes características:

- **Escalado Automático:** Escala automáticamente la cantidad de instancias de VM en el grupo cuando aumentan las cargas
- **Reparación Automática:** Realiza comprobaciones periódicas del estado y si una instancia de máquina virtual no está en buen estado, la vuelve a crear automáticamente
- **Admite Varias Zonas:** Puede crear un grupo de instancias administradas en varias zonas de disponibilidad en la misma región de Google Cloud
- Balanceo de carga

El proyecto SISEM proporciona equilibrio de carga administrado que lo ayuda a gestionar grandes cargas de tráfico para evitar abrumar las instancias de VM. El servicio de equilibrador de carga proporciona:

Reglas de Reenvío: Se implementan aplicaciones en varias regiones mediante grupos de instancias administrados regionalmente. También se configuran reglas de reenvío para distribuir el tráfico a todas las máquinas virtuales de esa región. Cada regla de reenvío puede usar una sola dirección IP externa, a través de la cual los usuarios pueden acceder a la aplicación.

- **Equilibrio de Carga Global:** Se implementan instancias en varias regiones mediante el equilibrio de carga global. El tráfico de equilibrio de carga HTTP/HTTPS nos permite ejecutar esta aplicación cerca de la ubicación geográfica de los usuarios (Bogotá)
- **Redundancia:** Se logra la redundancia equilibrando la carga entre regiones. Si una región no está disponible, el tráfico se enruta automáticamente a otra región, mientras que aún se puede acceder al servicio utilizando la misma dirección del dominio asignada (IP externa)
- **Escalado Automático:** Se agregarán o eliminarán instancias automáticamente cuando la carga aumente o disminuya, y equilibre la carga entre instancias en un grupo de instancias administrado

6.4. Scripts de Inicio y Apagado

Los equipos creados para SISEM tienen secuencias de comandos de inicio y apagado que se ejecutan cuando una instancia se inicia o se detiene. Estos scripts pueden automatizar tareas como:

- Instalación de software y dependencias
- Actualizaciones del sistema
- Copia de seguridad
- Generación de registros

Estos scripts son una forma efectiva de crear un procedimiento de arranque para este proyecto y cerrarlas limpiamente. En lugar de usar una imagen personalizada para configurar la instancia, puede usar un script de inicio. Después de cada reinicio, la secuencia de comandos de inicio se ejecuta y se usa para instalar o actualizar el software y asegurarse de que se estén ejecutando los servicios adecuados.

En el script de apagado se automatizan acciones importantes como:

- Cerrar conexiones
- Guardar el estado de las transacciones
- Realizar copias de seguridad de los datos

6.5. Alta Disponibilidad de SISEM con Kubernetes

El número de pods de backend es el número total de pods que son miembros de los recursos MultiClusterService en todos los clústeres de GKE del proyecto como se muestra a continuación.

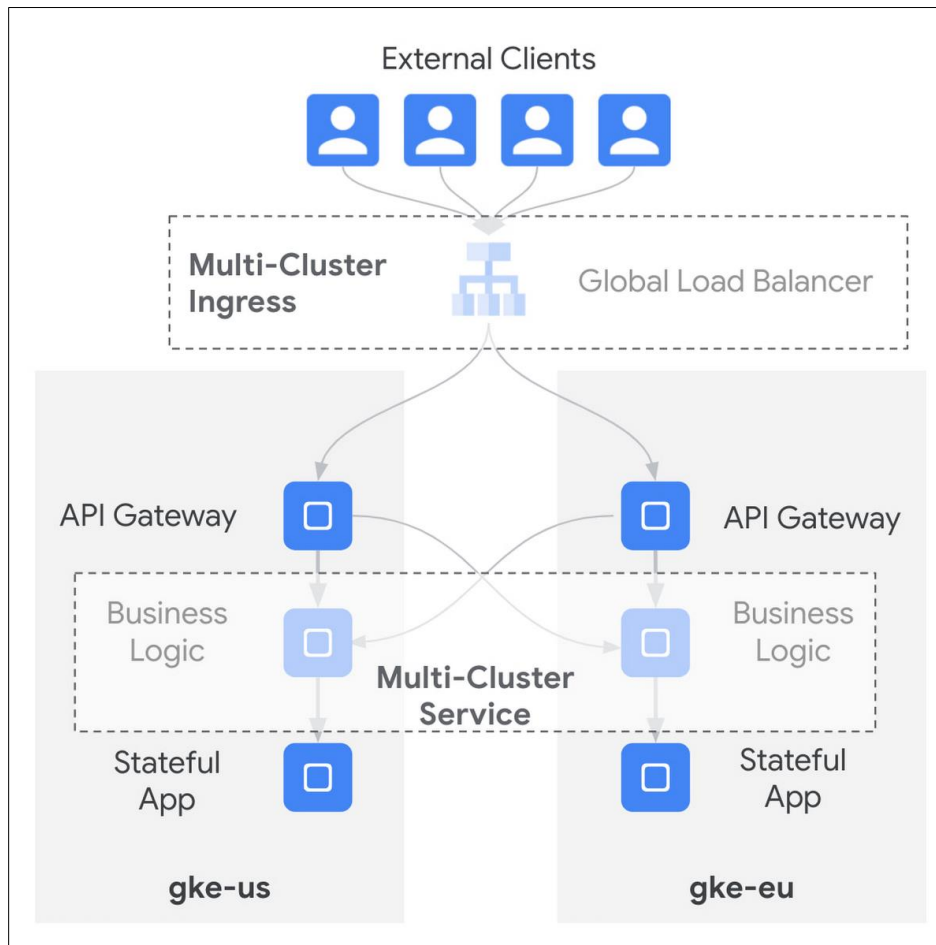


Ilustración 2 Virtual Private Cloud

GKE es el primer servicio de Kubernetes totalmente gestionado que es compatible con todas las APIs de Kubernetes, el auto escalado en cuatro vías, los canales de lanzamiento y la gestión de

varios clústeres. El auto escalado de pods horizontal puede regirse por el uso de CPU u otras métricas personalizadas. El auto escalado de clústeres funciona por grupo de nodos, mientras que el auto escalado de pods vertical analiza continuamente el uso de CPU y de memoria de los pods para ajustar automáticamente y en consecuencia las solicitudes de CPU y de memoria.

De igual manera GKE gestiona los posibles fallos que se puedan generar en los pods de ejecución esto mediante el mecanismo de control de TensorFlow, por lo cual los trabajos que se estén realizando si presentan algún fallo se volverán a ejecutar automáticamente con la lectura del estado del punto de control.

7. Gestión de Identidades y Accesos

A nivel de gestión de identidades para el proyecto SISEM se tienen dos momentos, para el aplicativo donde un usuario se identifica plenamente con su correo corporativo y contraseña, cada sesión está asociada a un token de autenticación que permite custodiar las sesiones activas e inhabilitarlas. Las contraseñas están encriptadas en el sistema de una sola vía, para que no se vean comprometidas.

Una vez un usuario ingresa al sistema no solo pasa por el proceso de autenticación, también por el proceso de validación para identificar qué información tiene acceso o puede ver. Esta validación está basada en dos conceptos: Gestión de privilegios y Gestión basada en roles.

La autenticación no está dada solo para usuarios internos, también: Contratistas, usuarios remotos y móviles, dispositivos, API's y microservicios.

Un segundo momento es la custodia de los accesos a la nube de Google, son responsabilidad del equipo de infraestructura y solo están a cargo del coordinador de área.

8. Administración y Monitoreo

La administración y monitoreo para los ambientes de desarrollo, QA, preproducción y producción, está configurado a distintos niveles de manera similar, en primera instancia se tiene un monitoreo general a nivel de infraestructura que se tiene en el entorno sobre la(s) instancias de compute engine pertenecientes al mismo, en el cual se muestran métricas establecidas por distintos segmentos de tiempo, estas métricas son orientadas principalmente a medir:

- CPU
- Red
- Memoria
- Disco

De igual manera este apartado de monitoreo nos permite la convergencia de las métricas de cada una de las instancias para así tener puntos de comparación entre unas y otras.

8.1. Monitoreo General de Infraestructura del Ambiente

Algunas de las métricas establecidas para el monitoreo a nivel general de infraestructura sobre el entorno son:

- Número de instancias activas del entorno

- % de uso máximo de procesamiento que ha tenido alguna instancia del entorno
- % promedio de uso de procesamiento de las instancias

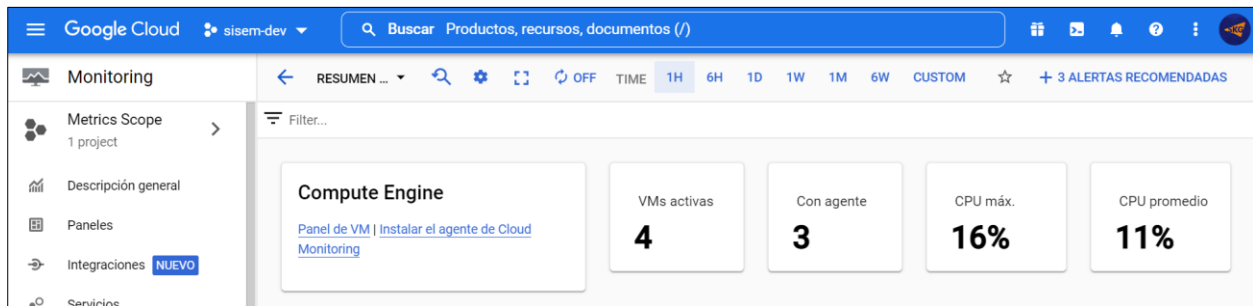


Ilustración 3 Monitoreo General

De igual manera como se mencionaba anteriormente se tiene no solo valores o porcentajes correspondiente a la CPU de las instancias, sino que se configuraron diagramas que permiten medir:

- Uso de memoria
- Tráfico enviado
- Tráfico recibido
- Latencia
- Índices de lecto/escritura en disco

A continuación, se muestran algunas de las métricas establecidas para el entorno de desarrollo, estas métricas reflejan el comportamiento de las instancias de compute engine en la última hora.

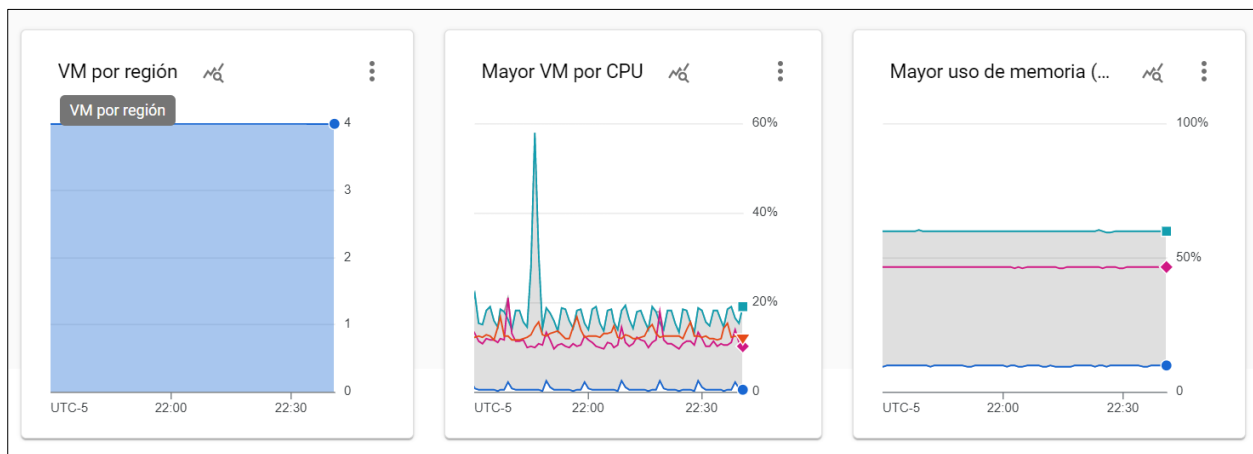


Ilustración 4 Monitoreo por Región, CPU y Memoria

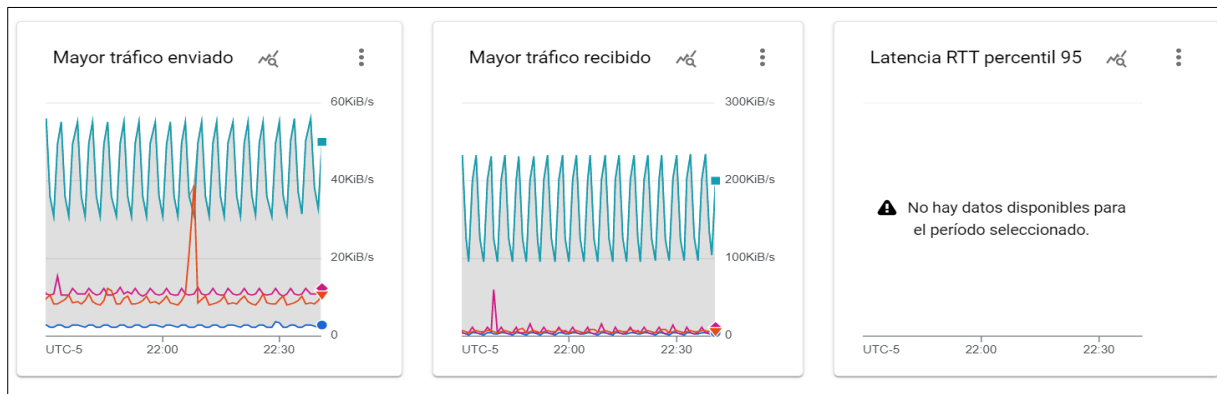


Ilustración 5 Monitoreo de Red



Ilustración 6 Monitoreo de Disco

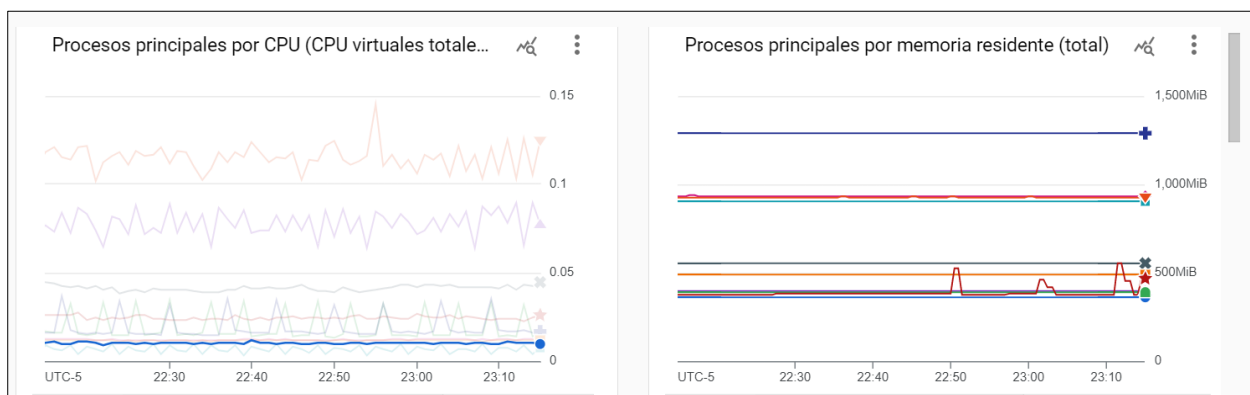


Ilustración 7 Monitoreo de Procedimientos

8.2. Monitoreo Compute Engine

A nivel de cada instancia también se configura un apartado monitoreo y métricas, que son más puntuales. Aunque los parámetros de las métricas son los mismos que en el monitoreo a nivel de infraestructura (CPU, procesador, memoria, red, disco) y también se puede definir un rango de tiempo de monitoreo, los datos y métricas son más puntuales al nivel que se permite tener medidas como:

8.2.1. CPU

- Porcentaje de uso de CPU por diferentes procesos ejecutados en la instancia
- Porcentaje de uso de CPU por el SO
- Procesos principales por porcentaje de uso de CPU de la CPU
- Procesos principales por uso de la CPU (CPU virtuales)

A continuación, se muestran algunas de las métricas mencionadas anteriormente:

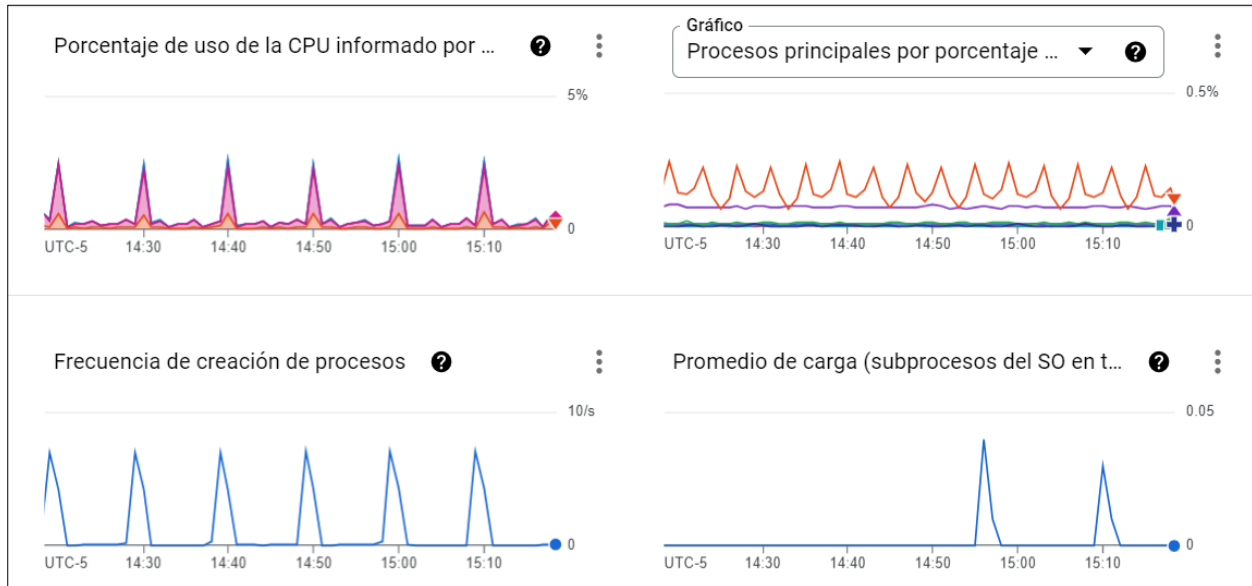


Ilustración 8 Monitoreo de CPU por Instancia

8.2.2. Procesos

- Procesos principales por porcentaje de uso de CPU
- Procesos principales por lectura o escritura de disco

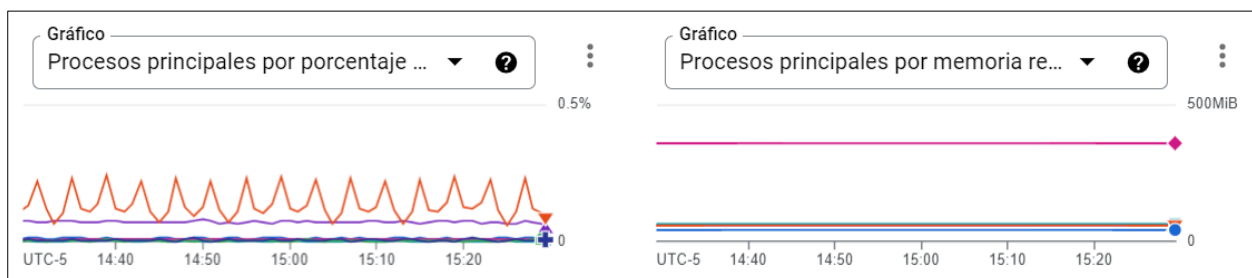


Ilustración 9 Monitoreo de Procesos de CPU por Instancia

8.2.3. Memoria

- Uso de memoria
- Memoria total utilizada por el SO
- Procesos principales por (memoria residente, porcentaje de memoria y memoria virtual)

- Detalle de memoria

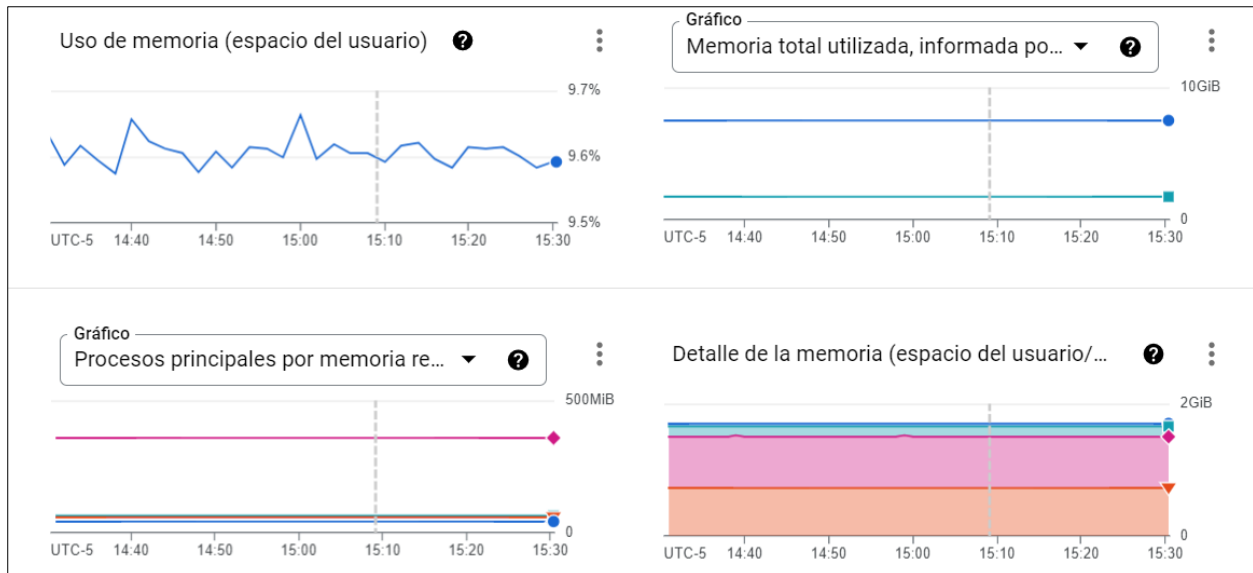


Ilustración 10 Monitoreo de Memoria por Instancia

8.2.4. Red

- Conexiones nuevas
- Totales de tráfico de red
- Total de paquetes de red
- Tráfico enviado
- Tráfico recibido
- Paquetes entrantes de firewall rechazados

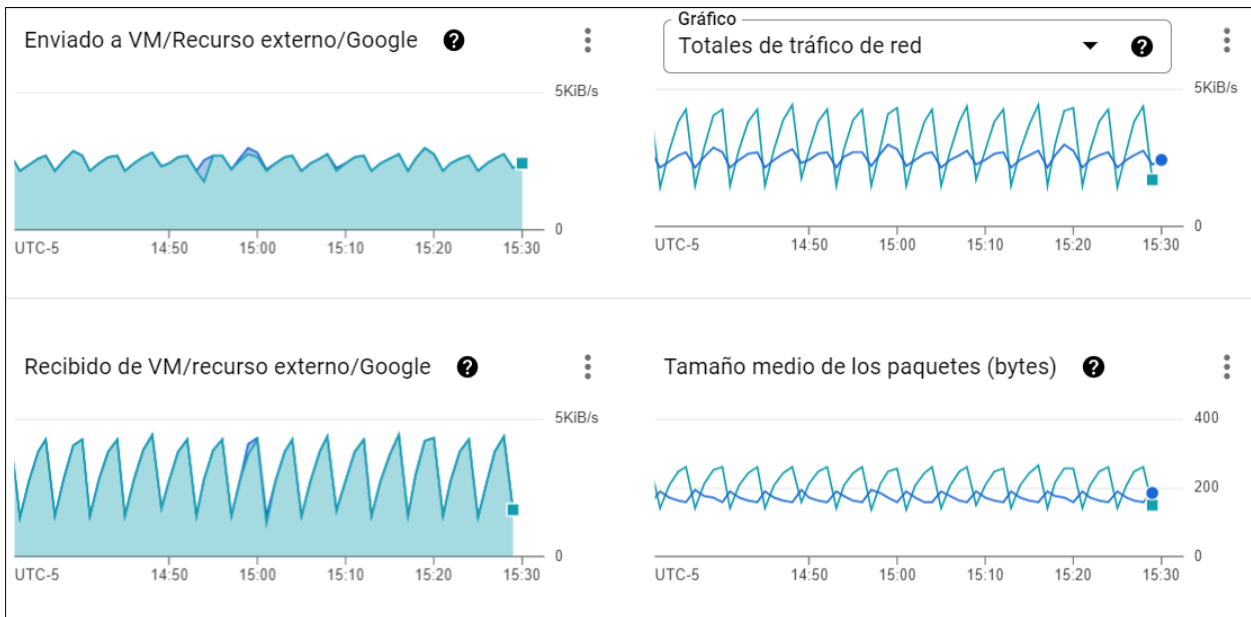


Ilustración 11 Monitoreo de Red por Instancia

8.2.5. Disco

- Capacidad de procesamiento MB/s
- Operaciones (IOPS)
- Promedio de latencia E/S

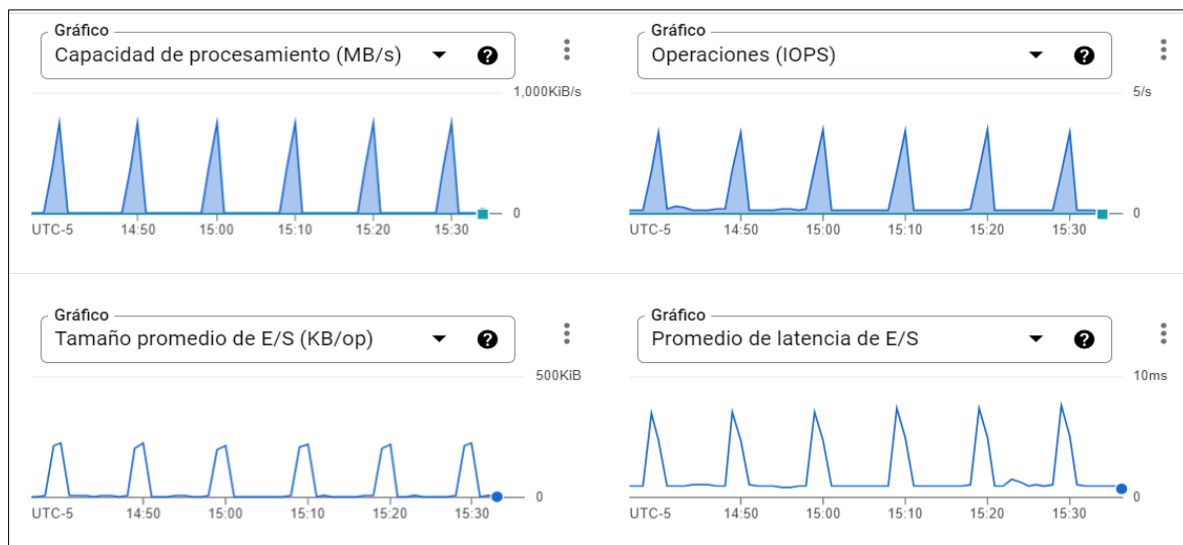


Ilustración 12 Monitoreo de Disco por Instancia

De igual manera se tiene un registro de los logs de la instancia en donde se puede filtrar dependiendo la necesidad como se muestra a continuación.

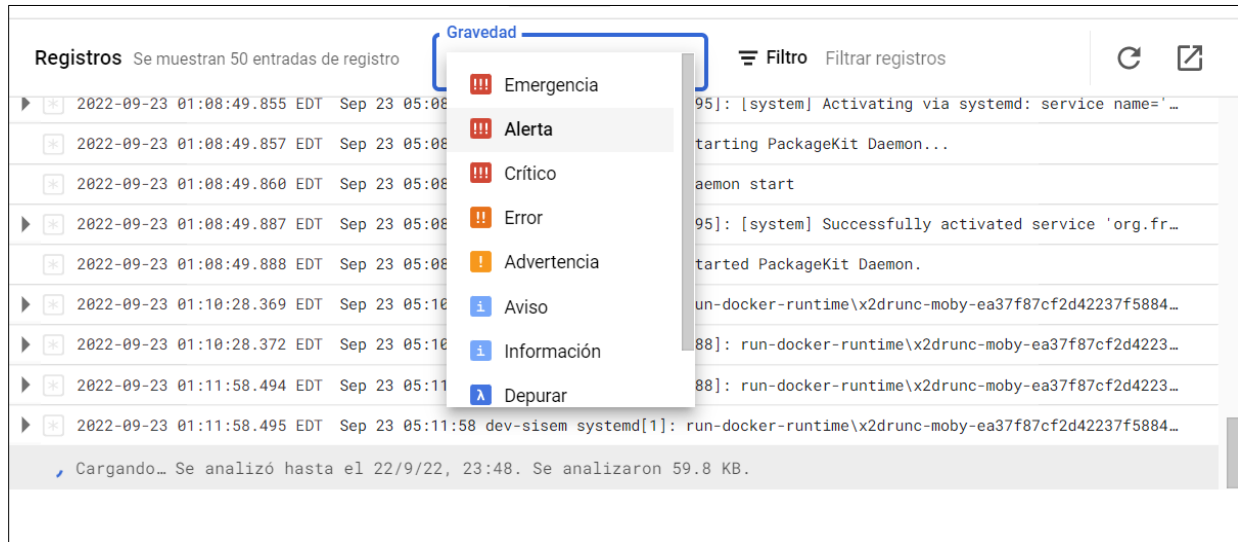


Ilustración 13 Registros por Instancia

8.3. Monitoreo a Nivel de API

Otro de los niveles de monitoreo que se tiene configurado es a nivel de API, este monitoreo se tiene configurado a través del servicio de cloud monitoring API, donde por medio del servicio de logging configurado con cloud logging API nos permite tener acceso a los registros de la instancia y respecto a estos obtener métricas a nivel de API.

En este nivel de monitoreo se pueden definir métricas tales como:

- Tráfico por código de respuesta
- Errores por método de API
- Latencia general
- Latencia por método API



Ilustración 14 Monitoreo a Nivel de API

8.4. Monitoreo a Nivel de Base de Datos

El monitoreo a nivel de base de datos nos permite medir métricas relacionadas con:

- Uso de CPU
- Conexiones activas
- Operaciones de lectura y escritura
- Entradas de registro por gravedad
- Uso de almacenamiento
- Transacciones

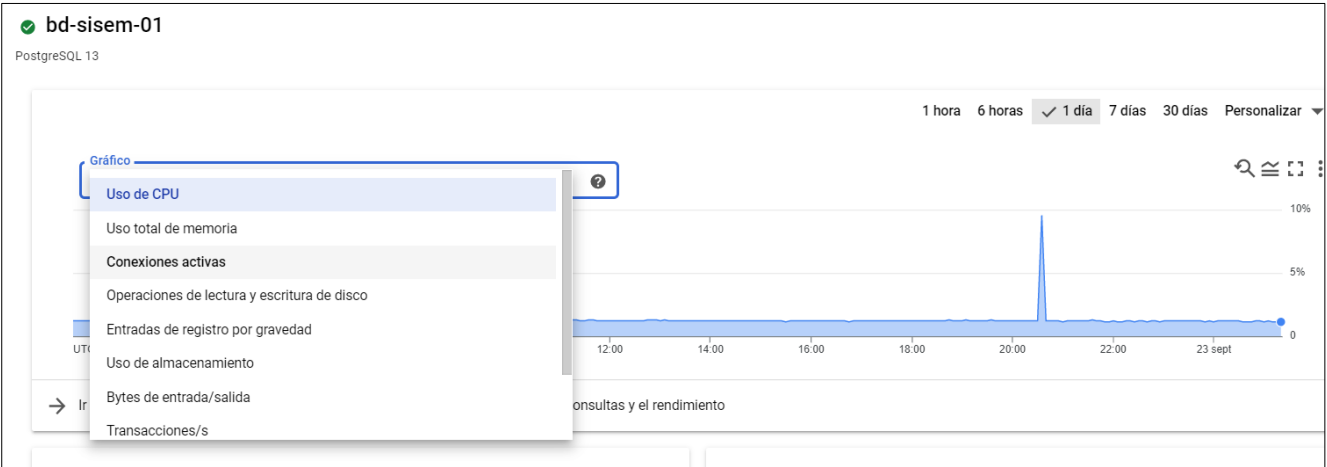


Ilustración 15 Monitoreo a Nivel de DB

8.5. Monitoreo GKE

El monitoreo a nivel de Kubernetes se considera como el monitoreo más importante de la aplicación puesto que va a ser el servicio que estará ejecutando SISEM tanto en el entorno de preproducción como en producción.

El monitoreo para Kubernetes nos permite tener una serie de métricas a distintos niveles dentro del servicio de GKE sobre un intervalo de tiempo, como lo son:

- Clusters

A continuación, se muestra las métricas y monitoreo a nivel de los clusters que se tienen en el servicio GKE en preproducción.

Clústeres					
No hay alertas activas					
NUM_ENTITIES,plural, =0{0 {PLURAL_ENTITY_NAME} con alertas activas=1{1 {SINGULAR_ENTITY_NAME} con alertas activas					
Nombre	Alertas ?	Registros de errores ?	Uso de CPU ?	Uso de memoria ?	Uso de disco ?
sisem-pre-prod-cluster...	0	218	27.07 % de 2.75 CPU	23.09 % de 6.76 GiB	0.22 % de 5.19 TiB

Ilustración 16 Monitoreo de Cluster GKE

8.6. Namespaces

A continuación, se muestran las métricas y monitoreo a nivel de a los namespaces que se tienen en el servicio GKE en preproducción.

Espacios de nombres					
No hay alertas activas					
NUM_ENTITIES,plural, =0{0 {PLURAL_ENTITY_NAME} con alertas activas=1{1 {SINGULAR_ENTITY_NAME} con alertas activas					
Nombre	Alertas ?	Reinicios de contenedor ?	Registros de errores ?	Uso de CPU ?	Uso de memoria ?
default	0	0	0	0 CPU ?	0 B ?
kube-node-lease	0	0	0	0 CPU ?	0 B ?
kube-public	0	0	0	0 CPU ?	0 B ?
kube-system	0	0	31	36 % de 1.05 CPU	25.77 % de 964.54 ...
sisem-pre-prod	0	1	187	11.32 % de 1.7 CPU	20.23 % de 5.82 GiB

Ilustración 17 Monitoreo de NameSpaces GKE

8.7. Nodos

A continuación, se muestran las métricas y monitoreo a nivel de los nodos que se tienen en el servicio GKE en preproducción.

Nodos					
No hay alertas activas					
NUM_ENTITIES,plural, =0{0 {PLURAL_ENTITY_NAME} con alertas activas=1{1 {SINGULAR_ENTITY_NAME} con alertas activas					
Nombre	Alertas ?	Reinicios de contenedor ?	Registros de errores ?	Uso de CPU ?	Uso de memoria ?
gke-sisem-pre-prod-cl...	0	0	23	8.19 % de 1.42 CPU	24.44 % de 1.72 GiB
gke-sisem-pre-prod-cl...	0	1	195	47.82 % de 1.33 CPU	21.89 % de 5.04 GiB

Ilustración 18 Monitoreo de Nodos GKE

8.8. Cargas de Trabajo

A continuación, se muestran las métricas y monitoreo a nivel de las cargas de trabajo que se tienen en el servicio GKE en preproducción.

Cargas de trabajo					
No hay alertas activas					
NUM_ENTITIES,plural, =0{0 {PLURAL_ENTITY_NAME} con alertas activas=1{1 {SINGULAR_ENTITY_NAME} con alertas					
Nombre	Alertas ?	Registros de errores ?	Uso de CPU ?	Uso de memoria ?	Uso de disco ?
event-exporter-gke	0	0	0 CPU ?	22.16 MiB ?	0 % de 5.9 GiB
konnnectivity-agent	0	31	4.01 % de 0.02 CPU	17.68 % de 60 MiB	0.01 % de 500 MiB
konnnectivity-agent-aut...	0	0	2.22 % de 0.01 CPU	125.11 % de 9.54 ...	0 % de 5.9 GiB
kube-dns	0	0	6.66 % de 0.52 CPU	15.49 % de 220 MiB	0 % de 200.3 GiB
kube-dns-autoscaler	0	0	1.15 % de 0.02 CPU	36.43 % de 10 MiB	0 % de 5.9 GiB

Ilustración 19 Monitoreo de Cargas de Trabajo GKE

8.9. Pods

A continuación, se muestran las métricas y monitoreo a nivel de pods que se tienen en el servicio GKE en preproducción.

No hay alertas activas					
NUM_ENTITIES,plural,=0{0 {PLURAL_ENTITY_NAME} con alertas activas=1{1 {SINGULAR_ENTITY_NAME} con alertas activas					
Nombre	Alertas ?	Registros de errores ?	Uso de CPU ?	Uso de memoria ?	Uso de disco ?
event-exporter-gke-54...	0	0	0 CPU ?	22.16 MiB ?	0 % de 5.9 GiB
fluentbit-gke-786wp	0	0	6.13 % de 0.1 CPU	12.79 % de 200 MiB	0 % de 94.74 GiB
fluentbit-gke-lwr4w	0	0	3 % de 0.1 CPU	10.28 % de 200 MiB	0 % de 94.74 GiB
gke-metrics-agent-8lgpk	0	0	81.87 % de 0 CPU	30.84 % de 50 MiB	0.01 % de 94.3 GiB
gke-metrics-agent-x9h...	0	0	98.56 % de 0 CPU	35.34 % de 50 MiB	0.01 % de 94.3 GiB

Ilustración 20 Monitoreo de Pods GKE

8.10. Contenedores

A continuación, se muestran las métricas y monitoreo a nivel de contenedores que se tienen en el servicio GKE en preproducción.

No hay alertas activas					
NUM_ENTITIES,plural,=0{0 {PLURAL_ENTITY_NAME} con alertas activas=1{1 {SINGULAR_ENTITY_NAME} con alertas					
Nombre	Alertas ?	Etiquetas	Reinicios de contenedor ?	Registros de errores ?	Uso de CPU ?
event-exporter	0	Clúster: sisem-pre-... +4	0	0	0 CPU ?
prometheus-to-sd-exp...	0	Clúster: sisem-pre-... +4	0	0	0 CPU ?
fluentbit	0	Clúster: sisem-pre-... +4	0	0	7.26 % de 0.05 CP
fluentbit-gke	0	Clúster: sisem-pre-... +4	0	0	4.99 % de 0.05 CP
fluentbit	0	Clúster: sisem-pre-... +4	0	0	4.42 % de 0.05 CP

Ilustración 21 Monitoreo de Contenedores GKE

9. Alertas

En lo que corresponde con las alertas, se establecen dos alertas para el entorno de desarrollo, que son comunes para cada una de las instancias pertenecientes al entorno que son:

- **Alerta de Procesamiento:** Se activa cuando alguna de las instancias supera el umbral establecido (70%)

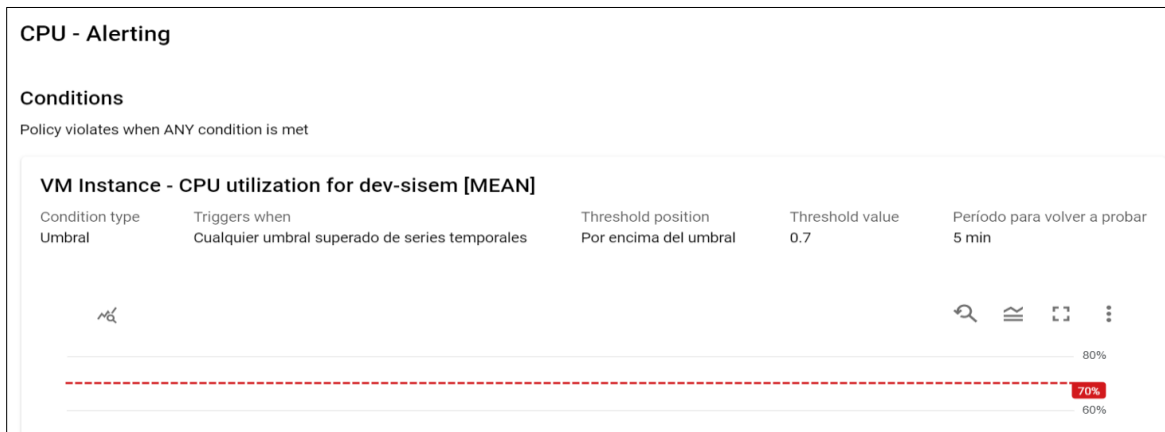


Ilustración 22 Alerta CPU

- **Alerta de Espacio de Almacenamiento:** Se activa cuando alguna de las instancias supera un margen de espacio de almacenamiento en el disco

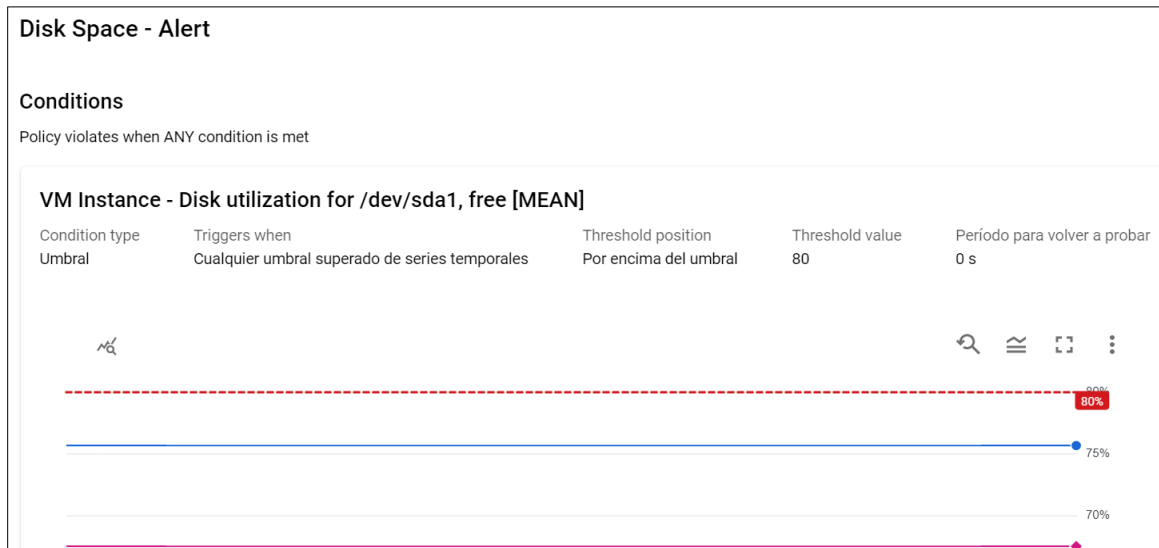


Ilustración 23 Alerta Disco

Cabe recordar que como se aclaró al principio el monitoreo de los entornos está configurado y administrado de manera similar pero no igual, por lo cual cabe resaltar que:

- El monitoreo a nivel de base de datos se encuentra solo en los ambientes de desarrollo y de producción, puesto que para el entorno de preproducción la base de datos se encuentra como servicio y su monitoreo es por medio del monitoreo de GKE
- En el entorno de producción cabe resaltar que al ser los nodos instancias de compute engine su monitoreo se puede medir tanto desde el monitoreo de GKE, como de compute engine, teniendo claridad que las métricas y datos que nos brinda GKE nos permite tener una mejor información del funcionamiento del sistema como uno solo
- Las alertas mencionadas anteriormente por el momento solo están configuradas en el ambiente de desarrollo

10. Hardening

10.1. Hardening a Nivel Cloud

El apartado de Hardening que se maneja cuenta con una configuración y administración conjunta entre el líder de infraestructura y la plataforma de Google, para lo cual Google se encarga de administrar todo lo correspondiente al funcionamiento y características de seguridad de las instancias y servicios para los apartados de virtualización, servidores, almacenamiento y red, es el líder de infraestructura el encargado de su configuración, donde en dado caso de algún fallo en este nivel, es la misma plataforma de Google quien se encarga de la administración, estos fallos administrados por Google están relacionados principalmente con:

- Fallos de servicios y máquinas en zonas
- Fallos de máquinas y servicios por regiones
- Fallos a nivel de infraestructura física de red
- Fallos a nivel de hardware de las instancias

En este apartado cabe aclarar que Google no solo proporciona este manejo de errores, sino que también gestiona el apartado de la escalabilidad de la aplicación, en caso de tener que actualizar más memoria o alguna otra característica de hardware o en caso de tener que realizar una réplica de alguna instancia, como ya se mencionó anteriormente.

La administración y configuración de lo correspondiente a sistema operativo, entorno de ejecución, aplicación, middlewares y datos corresponde a los encargados de cada nivel.

Donde a nivel de aplicación se establecen una serie de reglas que permiten manejar roles de usuario sobre la aplicación, encriptación de datos, patrones de desarrollo y otras.

A nivel de datos se definen privilegios y permisos de acceso sobre las instancias de Cloud SQL, de igual manera se manejan tanto copias de seguridad de información como réplicas redundantes como medidas de tolerancia a fallos.

A pesar de la administración de Google en algunos apartados como se menciona anteriormente, es el líder de infraestructura quien determina una serie de reglas y mecanismos que permitan brindar una capa de seguridad a las instancias, tales como:

- Firewall: Configuración de reglas y brinde de puertos
- Monitoreo a distintos niveles
- Logs de servicios

10.2. Hardening a Nivel de Base de Datos

Anteriormente se ha hablado de algunas características de seguridad y alta disponibilidad a nivel de base de datos con las que cuenta el proyecto, en este apartado se hace énfasis en el apartado de seguridad en la migración de los datos por lo cual se definen e implementan una serie de estrategias y tecnologías que permiten proteger la información, su acceso no autorizado, corrupción o robo en su ciclo de vida, las estrategias implementadas son:

- **Autenticación:** La autenticación es un proceso que permite identificar a los usuarios del sistema, de esta manera se desbloquean datos bloqueados para que

las partes autorizadas los utilicen. Esta autenticación se aplica a nivel de la red, la aplicación y el archivo

- **Control de Acceso:** El control de acceso permite definir qué datos pueden ver los usuarios respecto a los roles que manejen en el proyecto, esto garantiza que los distintos usuarios que necesiten ver datos o contenido confidencial estén debidamente autorizados para hacerlo, y no que cualquier persona o usuario del sistema pueda acceder a contenido exclusivo para otros roles
- **Respaldo y Recuperación:** Este apartado hace referencia a lo mencionado anteriormente en la alta disponibilidad de las bases de datos, para lo cual se tiene una capa de seguridad que permite que en dado caso de que una máquina o una zona falle, se tenga un respaldo en otra zona con información redundante, que permita el correcto funcionamiento del sistema en todo momento
- **Resiliencia de Datos:** Para este apartado hace referencia a la rapidez en que se puede recuperar o restaurar las operaciones en caso de que se produzca una violación de datos o una falla, esto se mitiga mediante la implementación de Google Cloud y todas las ventajas que tiene la misma, en la cual ya brinda distintas capas de automatización y seguridad que un proyecto de esta magnitud necesita, como son; los distintos servicios y características orientados a la alta disponibilidad en base de datos, que se presentaron anteriormente en el apartado de alta disponibilidad
- **Control de Datos:** El control de datos abarca políticas y prácticas que sirven como base para la administración de los datos, permite determinar la responsabilidad y define cómo se almacenan, transfieren, acceden a los datos, de igual manera como se conservan y destruyen, además brindan orientación a los usuarios del sistema de cómo mantener los datos seguros y limpios

11. IPv6

IPv6 es compatible con los servicios proporcionados por Google, servicios como VPC permiten la configuración de la red bajo IPv6 con las siguientes aclaraciones:

- La subred a la que está configurada la instancia de VM tiene un rango IPv6 configurado, esto quiere decir que se tiene una subred dentro del servicio de VPC que cuenta con rangos de direcciones tanto de IPv4 e IPv6 (subred de pila doble)
- Por defecto el firewall deniega las conexiones entrantes a las direcciones IPv6 de las instancias de VM, por lo cual se debe tener una regla de alta prioridad que permita el acceso entrante
- Las instancias de VM se les asigna una dirección IPv6 mediante DHCPv6
- La configuración de IPv6 se puede hacer tanto externa, como comunicación únicamente de las instancias de VM o tanto interna como externa (Internet)

De igual manera se aclara que la mayoría de los proyectos se trabajan internamente con direcciones IP versión 4, por lo cual la solución más factible a un posible cambio a IPv6 sería configurar un gateway que reciba las peticiones a través de IPv6, pero internamente se siga trabajando con IPv4 para la comunicación de las instancias.

12. Continuidad de Negocio y Recuperación Ante Desastres

A nivel de continuidad del negocio se tienen ambientes de producción en dos regiones de US con multi zona este y oeste, de igual manera se tiene un sistema espejo (en pausa) en la región de Sudamérica solo con el servicio de réplica de datos para ser activado en caso de pérdida de la funcionalidad del servidor primario.

Es de esta manera que se implementa una tolerancia a fallos y continuidad del negocio a distintos niveles:

- **GKE:** Estableciendo copia de seguridad para GKE, las copias de seguridad de cargas de trabajo de GKE son útiles para la recuperación ante desastres, las canalizaciones de CI/CD, la clonación de cargas de trabajo o las situaciones de actualización y/o recuperación.
- **Pods:** A través del punto de control de Tensor Flow, en caso de falla de algún pod reiniciar su servicio desde su última versión estable
- **Instancias:** (Nodos), en donde se verifica el funcionamiento de cada instancia y en caso de falla una entra a tomar el tráfico de la instancia que ya no responde
- **Base de Datos:** En donde se verifica que el funcionamiento de la instancia principal sea el correcto y en caso de no serlo se cambia a una instancia réplica para que los datos se puedan seguir proporcionando de la manera correcta a la aplicación
- **Firewall / Cloud Armor** Estableciendo reglas y configuraciones de seguridad que permitan solventar distintos ataques

13. Ambiente de Desarrollo (Dev)

En Google Cloud Platform se ha desplegado el ambiente de desarrollo Dev y dentro de este ambiente se encuentra embebido los ambientes de Preproducción, pruebas (QA), desarrollo (Dev) y desarrollo móvil.

El ambiente de desarrollo se despliega en alta disponibilidad utilizando compute engine de Google Cloud y GKE Google Kubernetes Engine. Donde se desplegarán cada uno de los componentes en contenedores. Esta infraestructura nos permitirá tener control y visualización de logs para apoyar la labor de desarrollo, desarrollo móvil, preproducción y pruebas (QA).

Google Cloud

Sisem - Dev

Buscar (/) recursos, documentos, productos y más

Clústeres de Kubernetes

CREAR

IMPLEMENTAR

ACTUALIZAR

DESCRIPCIÓN GENERAL

OBSERVABILIDAD

OPTIMIZACIÓN DE COSTOS

Filtro

Ingresar el nombre o el valor de la propiedad

☐ Estado	Nombre ↑	Ubicación	Cantidad de nodos	CPU virtuales totales	Memoria total
☒	sisem-dev-cluster-1	us-east1-b	3	12	48 GB
☒	sisem-mobile-cluster-dev	us-east1-b	2	4	16 GB
☒	sisem-preprod-cluster-1	us-east1-c	3	12	48 GB
☒	sisem-qa-cluster-1	us-east1-b	3	12	48 GB

Ilustración 24 Características Generales GKE Ambientes Desarrollo, Desarrollo Móvil, Preproducción y Pruebas (QA)

13.1. Máquinas y Servicios

Para el ambiente de desarrollo se tienen activas cinco instancias de compute engine como se muestra a continuación:

Estado	Nombre ↑	Zona	Fecha y hora de creación
✓	analisis-datos	us-east1-b	mar 16, 2022, 10:30:31 a. m. UTC-05:00
✓	demosidcrue	us-east1-b	feb 24, 2022, 9:31:04 a. m. UTC-05:00
✓	dev-sisem	us-east1-b	nov 23, 2021, 3:02:43 p. m. UTC-05:00
⊗	windows-power-bi	us-east1-b	jul 25, 2023, 10:53:49 a. m. UTC-05:00
✓	winserver	us-east1-b	sept 5, 2023, 10:44:34 a. m. UTC-05:00

Ilustración 25 Instancias Ambiente de Desarrollo

13.1.1. análisis-datos

Esta instancia de compute engine está configurada así:

Tipo de máquina	n1-standard-2
CPU	2vCPUs
Memoria	7.5GB
Disco	50GB
SO	Debian GNU/Linux 10 (buster)
Región	us-east1-b
IP Interna	10.130.0.5
IP Externa	35.237.51.164

Tabla 1. Configuración análisis-datos

13.1.2. demosidcrue

Esta instancia de compute engine está configurada así:

Tipo de máquina	n1-standard-2
CPU	2vCPUs
Memoria	7.5GB
Disco	100GB
SO	Windows server 2016
Región	us-east1-b
IP Interna	10.130.0.4
IP Externa	35.185.5.59

Tabla 2. Configuración demosidcrue

13.1.3. dev-sisem

Esta instancia de compute engine está configurada así:

Tipo de máquina	e2-standard-2
CPU	2vCPUs
Memoria	8GB
Disco	100GB
SO	Ubuntu 18.04.6 LTS
Región	us-east1-b
IP Interna	10.130.0.6
IP Externa	35.243.247.134

Tabla 3. Configuración dev-sisem

13.1.4. windows-power-bi

Esta instancia de compute engine está configurada así:

Tipo de máquina	e2-standard-4
CPU	4vCPUs
Memoria	16GB
Disco	80GB
SO	windows-server-2022-dc-v20230712
Región	us-east1-b
IP Interna	10.130.0.248
IP Externa	Efímera

Tabla 4. windows-power-bi

13.1.5. winserver

Esta instancia de compute engine está configurada así:

Tipo de máquina	e2-medium
CPU	1-2vCPUs
Memoria	4GB
Disco	50GB
SO	windows-server-2022-dc-v20230712
Región	us-east1-b
IP Interna	10.130.0.251
IP Externa	34.73.215.147

Tabla 5. winserver

De igual manera se tiene configurado 4 instancias del servicio de cloud SQL compartidas entre los entornos de desarrollo, desarrollo móvil y pruebas, para lo correspondiente a las bases de datos, que son:

☐	ID de instancia ? ↑	Edición de Cloud SQL	Tipo	Dirección IP pública
☐	bd-sisem-01	Enterprise	PostgreSQL 13	34.148.74.49 ?
☐	✓ premier-one	Enterprise	SQL Server 2022 Express	34.74.154.194
☐	✓ quejas	Enterprise	MySQL 5.7	104.198.67.167
☐	✓ sisem-mobile-test	Enterprise	PostgreSQL 14	35.196.142.87 ?

Ilustración 25 Instancias cloud SQL entorno desarrollo

Cuenta con un servicio de cloud logging compartido con el entorno de pruebas para el almacenamiento de los logs de las instancias de cloud engine.



Ilustración 26 Cloud Logging Ambiente Desarrollo

Y por último a nivel de red se tiene configurado el servicio de VPC, que permite la comunicación entre las instancias, como se muestra a continuación.

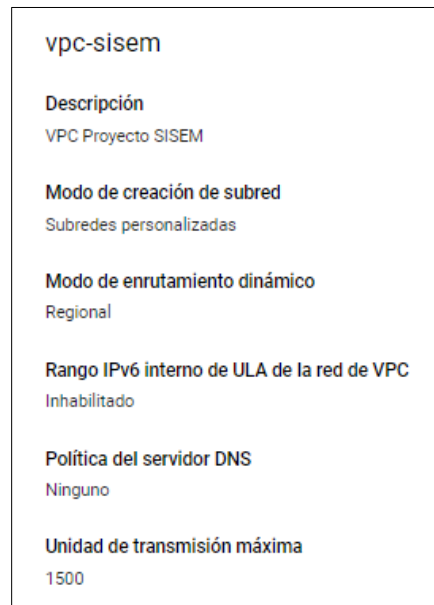


Ilustración 27 VPC Ambiente Desarrollo

y una subred de vpc-sisem, configurada de la siguiente manera:

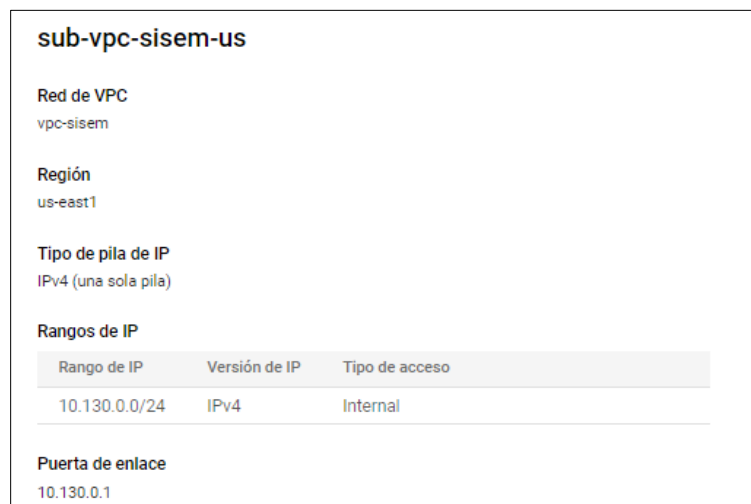


Ilustración 28 Subred de VPC Ambiente Desarrollo

13.2. Variables de Entorno

Las variables de entorno en el ambiente de pruebas se definen a dos niveles, primero se definen variables secretas y sensibles dentro del SO de la instancia de compute engine y algunas dentro de la configuración del contenedor.

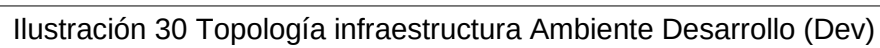
Ilustración 29 Variables de Entorno

A continuación, se muestra el diagrama de red del ambiente de desarrollo dev en donde se subraya embebido las redes de los ambientes de preproducción, desarrollo (Dev), desarrollo móvil, pruebas (QA) y como estas se enrutan internamente y externamente hacia cada una de las zonas dispuestas en GCP:



A continuación, se muestra la topología de la infraestructura separado por los ambientes desarrollo (Dev), preproducción, pruebas (QA) y desarrollo Móvil.

La infraestructura en el siguiente diagrama se ha implementado en Google cloud platform GCP



La infraestructura en el siguiente diagrama se ha implementado en Google cloud platform GCP

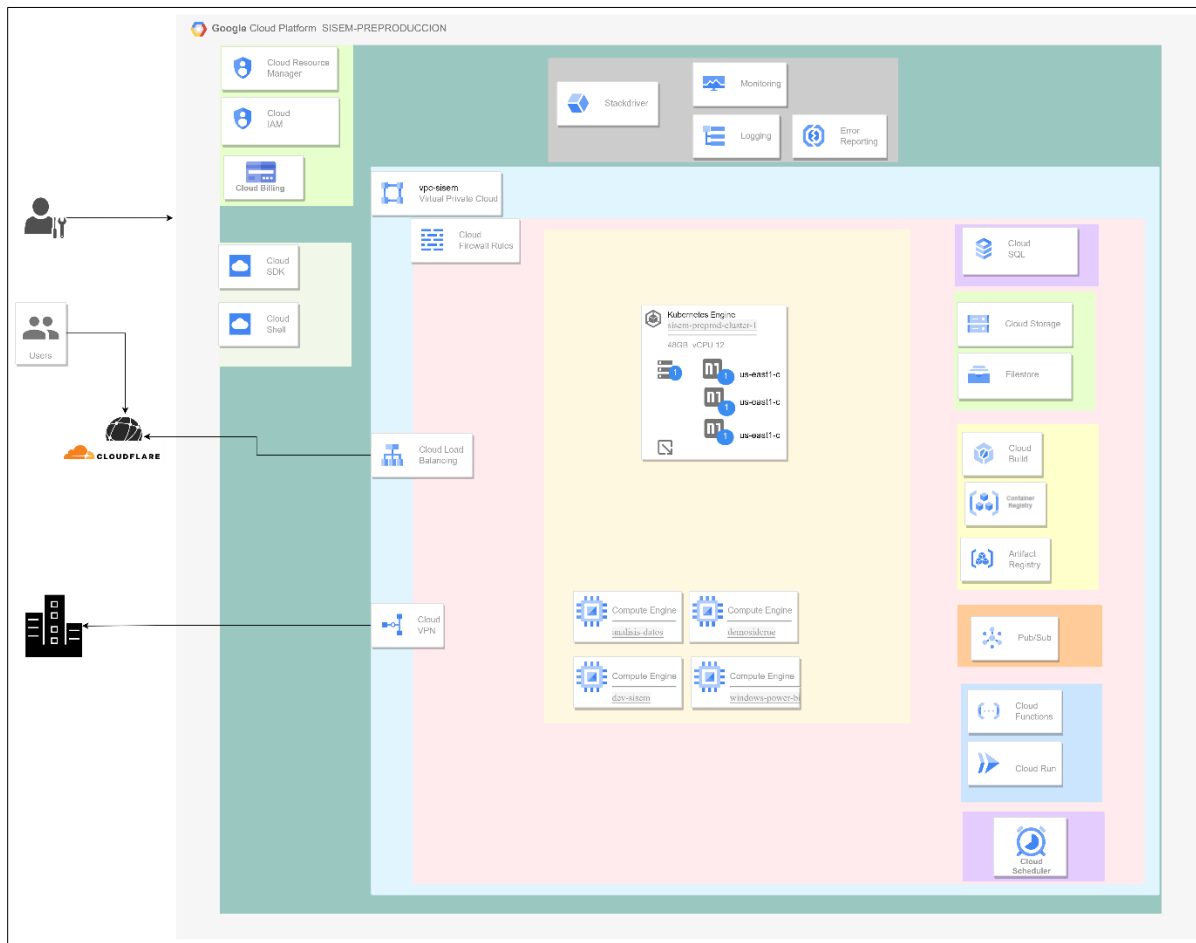


Ilustración 31 Topología infraestructura Ambiente preproducción

13.4.3. Topología pruebas (QA)

La infraestructura en el siguiente diagrama se ha implementado en Google cloud platform GCP



La infraestructura en el siguiente diagrama se ha implementado en Google cloud platform GCP

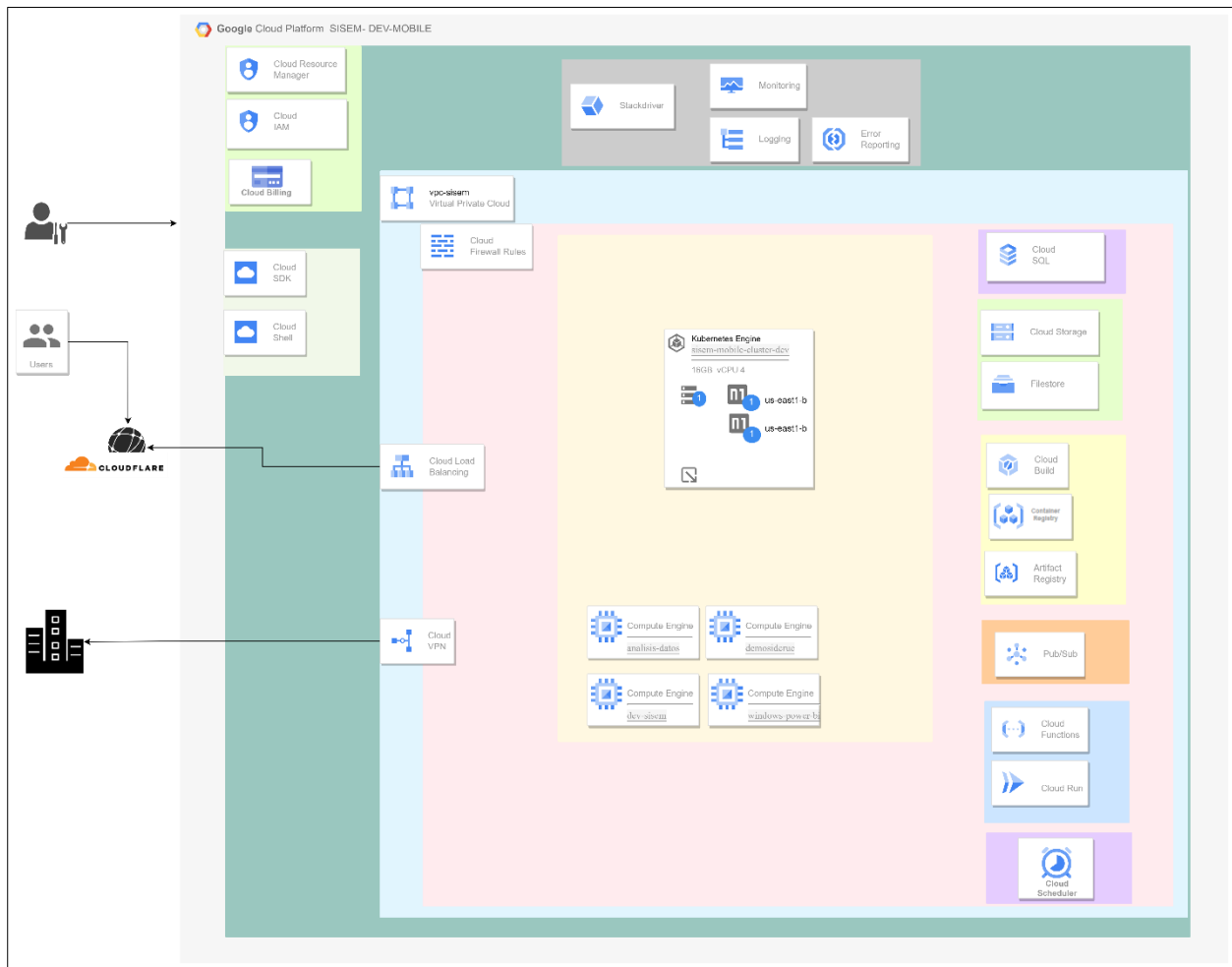


Ilustración 33 Topología infraestructura Ambiente desarrollo móvil

13.5. Firewall y Puertos

La configuración de firewall y puertos de la instancia está configurada de la siguiente manera:

Nombre	Tipo	Destinos	Filtros	Protocolos/puertos	Acción	Prioridad	Red ↑	Registros
default-allow-ssh	Entrada	Aplicar a tod:	Intervalos	tcp:22	Permitir	1000	vpc-sisem	Activado
default-postgress-ingress	Entrada	Aplicar a tod:	Intervalos	tcp:5432 udp:5432	Permitir	1000	vpc-sisem	Desactivado
fr-sisem-dev-ftp	Entrada	Aplicar a tod:	Intervalos	tcp:21	Permitir	1000	vpc-sisem	Desactivado
fr-sisem-http	Entrada	Aplicar a tod:	Intervalos	tcp:80, 81	Permitir	1000	vpc-sisem	Desactivado
fr-sisem-https	Entrada	Aplicar a tod:	Intervalos	tcp:443	Permitir	1000	vpc-sisem	Desactivado
fr-sisem-icmp	Entrada	Aplicar a tod:	Intervalos	icmp	Permitir	1000	vpc-sisem	Desactivado
fr-sisem-rdp	Entrada	Aplicar a tod:	Intervalos	tcp:3389	Permitir	1000	vpc-sisem	Desactivado
fr-sisem-ssh	Entrada	Aplicar a tod:	Intervalos	tcp:22	Permitir	1000	vpc-sisem	Desactivado
ingres-jupyter	Entrada	Aplicar a tod:	Intervalos	tcp:9092 udp:9092	Permitir	1000	vpc-sisem	Desactivado
portainer-agent	Entrada	Aplicar a tod:	Intervalos	tcp:9123	Permitir	1000	vpc-sisem	Desactivado
rabbitmq	Entrada	Aplicar a tod:	Intervalos	tcp:5672, 9891 udp:5672, 9891	Permitir	1000	vpc-sisem	Desactivado
rtc	Entrada	Aplicar a tod:	Intervalos	tcp:3478, 3479, 4442, 4444, 5349, 5350 udp:3478, 3479, 4442, 4444, 5349, 5350	Permitir	1000	vpc-sisem	Desactivado
sisem-admin-ports	Entrada	Aplicar a tod:	Intervalos	tcp:9000, 9001	Permitir	1000	vpc-sisem	Desactivado
sisem-postgres-redis	Entrada	Aplicar a tod:	Intervalos	tcp:5432, 6379	Permitir	1000	vpc-sisem	Desactivado
vpc-sisem-allow-http	Entrada	http-server	Intervalos	tcp:80	Permitir	1000	vpc-sisem	Desactivado
vpc-sisem-allow-https	Entrada	https-server	Intervalos	tcp:443	Permitir	1000	vpc-sisem	Desactivado
default-postgress-egress	Salida	Aplicar a tod:	Intervalos	tcp:5432 udp:5432	Permitir	1000	vpc-sisem	Desactivado
egress-jupyter	Salida	Aplicar a tod:	Intervalos	tcp:9092 udp:9092	Permitir	1000	vpc-sisem	Desactivado

Ilustración 34 Configuración Firewall y Puertos Ambiente Desarrollo

Secrets y ConfigMaps	Nombre ↑	Tipo	Espacio de nombres
Almacenamiento	default-token-mg9l2	Secret	kube-node-lease
Navegador de objetos	default-token-tl4pt	Secret	sisem-pre-prod
Migrate to Containers	ingress-controller-leader	Config Map	sisem-pre-prod
Copia de seguridad para GKE	kube-root-ca.crt	Config Map	kube-node-lease
Administración de la configu...	kube-root-ca.crt	Config Map	default
Postura de seguridad	kube-root-ca.crt	Config Map	sisem-pre-prod
	kube-root-ca.crt	Config Map	kube-public
	letsencrypt-production	Secret	sisem-pre-prod
	letsencrypt-staging	Secret	sisem-pre-prod
	minio-sa-token-tmdc6	Secret	sisem-pre-prod
	sh.helm.release.v1.sisem.v19	Secret	sisem-pre-prod
	sh.helm.release.v1.sisem.v20	Secret	sisem-pre-prod
	sh.helm.release.v1.sisem.v21	Secret	sisem-pre-prod
	sh.helm.release.v1.sisem.v22	Secret	sisem-pre-prod
	sh.helm.release.v1.sisem.v23	Secret	sisem-pre-prod
	sh.helm.release.v1.sisem.v24	Secret	sisem-pre-prod
	sh.helm.release.v1.sisem.v25	Secret	sisem-pre-prod
Marketplace	sh.helm.release.v1.sisem.v26	Secret	sisem-pre-prod
Notas de versión	sh.helm.release.v1.sisem.v27	Secret	sisem-pre-prod
	sh.helm.release.v1.sisem.v28	Secret	sisem-pre-prod
	sisem-actors-ms	Config Map	sisem-pre-prod

Ilustración 35 Configuración de Secrets y Configmaps Gke Ambiente Desarrollo

Cabe aclarar que el manejo de las variables de entorno del ambiente se maneja tanto del cluster en general como algunas configuradas directamente a nivel del contenedor, como se muestra a continuación, como ejemplo se tiene un pod del servicio de base de datos:

✓ sisem-pg-notifications-ms-pgpool-cd4d665c7-95gtg	
DETALLES	EVENTOS
REGISTROS	YAML
COPIAR DESCARGAR <pre> 232 weight: 1 233 containers: 234 - env: 235 - name: BITNAMI_DEBUG 236 value: "false" 237 - name: PGPOOL_BACKEND_NODES 238 value: 0:sisem-pg-notifications-ms-postgresql-0.sisem-pg-notifications-ms-postgresql-headless:5432,1:sisem-pg-notifications-ms-postgr 239 - name: PGPOOL_SR_CHECK_USER 240 value: repmgr 241 - name: PGPOOL_SR_CHECK_PASSWORD 242 valueFrom: 243 secretKeyRef: 244 key: repmgr-password 245 name: sisem-pg-notifications-ms 246 - name: PGPOOL_SR_CHECK_DATABASE 247 value: postgres 248 - name: PGPOOL_ENABLE_LDAP 249 value: "no" 250 - name: PGPOOL_POSTGRES_USERNAME </pre>	

Ilustración 36 Configuración Variables de Entorno Contenedores

13.6. Seguridad

Propuesta general de medidas de aseguramiento de seguridad y privacidad de los datos, así como validación de permisos y medidas para prevención de potenciales ataques, a continuación, se describe el flujo general de una petición a nivel de validaciones y flujos.

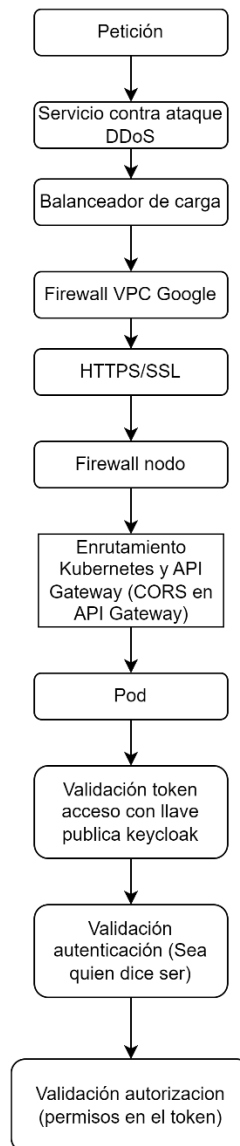


Ilustración 27 Flujo General

13.6.1. Servicio Contra Ataque DDOS

La prevención contra ataques DDOS es una regla implícita que viene en la implementación del servicio de cloud armor de Google, esta regla viene a distintos niveles como se muestra a continuación:

Defensa frente a DDoS de nivel empresarial

Cloud Armor se basa en nuestra experiencia en la protección de servicios clave de Internet, como la Búsqueda de Google, Gmail y YouTube. Proporciona un sistema de defensa integrado frente a ataques DDoS de capa 3 y capa 4.

Ilustración 28 DDos

De igual manera brinda una capa en la cual si se requiere permite configuraciones de reglas que permitan solventar esta problemática a niveles más altos a través del siguiente servicio:

Adaptive Protection

Detecta y ayuda a reducir automáticamente los ataques DDoS de capa de volumen alto con un sistema de aprendizaje automático entrenado en tus aplicaciones de forma local. [Más información](#)

Ilustración 29 DDos multicapa

13.6.1.1. Creación de la Regla DDoS

[←](#) Crear política de seguridad

Una política de seguridad contiene una o más reglas. Las reglas le indican a la política de seguridad qué hacer (la acción) y cuándo (la condición). Los objetivos son los elementos a los que se aplica la regla. [Más información](#)

- **Configura la política**

Nombre *

ddos

?

Se permiten letras minúsculas, números y guiones

Descripción

Tipo de política
☒ Política de seguridad del backend
☐ Política de seguridad de Edge

Acción de la regla predeterminada ?
☐  Permitida
☒  Rechazada

Código de denegación

403 (Prohibido)

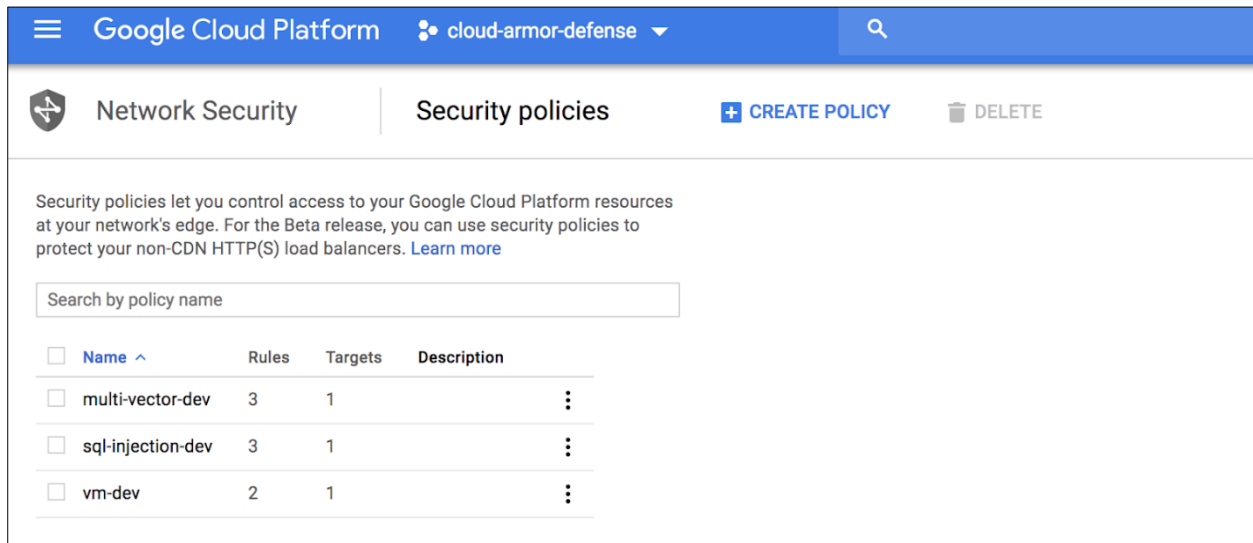
▼

?

PRÓXIMO PASO

Ilustración 30 configuración de políticas en Cloud Armor

13.6.1.2. Políticas Configuradas



Google Cloud Platform cloud-armor-defense

Network Security Security policies + CREATE POLICY DELETE

Security policies let you control access to your Google Cloud Platform resources at your network's edge. For the Beta release, you can use security policies to protect your non-CDN HTTP(S) load balancers. [Learn more](#)

Search by policy name

☐ Name ^	Rules	Targets	Description
☐ multi-vector-dev	3	1	⋮
☐ sql-injection-dev	3	1	⋮
☐ vm-dev	2	1	⋮

Ilustración 31 Security policies - SISEM Preproducción

Se crean tres reglas en el cloud armor como primera capa que protege todo el sistema:

- **multi-vector:** Protección de ataques DDoS con tres reglas de temporización entre llamadas, origen, tipo de llamado
- **sql-injection:** Protección y revisión de las peticiones
- **vm:** Reglas para la protección de las VPC de cada proyecto

13.6.2. Balanceador de Carga

Servicio de balanceo de carga de Google, se encarga de la distribución de las peticiones al cluster de kubernetes para escalamiento horizontal para esto se configura el servicio de ingress controller (configurado con nginx) en el clúster de kubernetes, que es el encargado del balanceo de carga y cuenta con las siguientes características:

Clúster	sisem-pre-prod-cluster-1
Espacio de nombres	sisem-pre-prod
Fecha de creación	14 oct 2022 22:54:18
Etiquetas	app.kubernetes.io/component: controller app.kubernetes.io/instance: sisem app.kubernetes.io/name: nginx-ingress pod-template-hash: 659d779945
Anotaciones	<i>Sin establecer</i>
Registros	Container logs , Audit logs
Nodo	gke-sisem-pre-prod-cluster-1-pool-1-bbd34110-q7c2
Período de gracia de terminación	5 min
Política de reinicio	Always
Fase	Running
Hora de inicio	14 oct 2022 22:54:18
Condiciones	Initialized: True Ready: True ContainersReady: True PodScheduled: True
Controladores	Replica Set: sisem-nginx-ingress-controller-659d779945
Volúmenes	webhook-cert kube-api-access-dvmvs

Ilustración 32 Ingress controller (1)

Contenedores				
Nombre ↑	Estado	Imagen	Conteo de reinicios	Registros
controller	Running	registry.k8s.io/ingress-nginx/controller:v1.3.0@sha256:d1707ca76d3b044ab8a28277a2466a02100ee9f58a86af1535a3edf9323ea1b5	0	Ver registros

Servicios expuestos		
Nombre ↑	Tipo	Extremos
sisem-nginx-ingress-controller	Balanceador de cargas	35.196.154.63:80
sisem-nginx-ingress-controller-admission	IP del clúster	10.40.6.224

Ilustración 33 Ingress controller (2)

13.6.3. Firewall VPC Google

Los nodos del cluster de kubernetes son instancias de la GCP de Compute Engine, por lo tanto, ya están en una VPC propia de ellos la cual proporciona otro firewall para reglas de acceso y egreso, las cuales se muestran a continuación:

Nombre	Tipo	Destinos	Filtros	Protocolos/puertos	Acción	Prioridad	Red ↑	Registros
fr-sisem-preprod-admin	Entrada	Aplicar a todos	Intervalos de IP: t	tcp:5432, 6379, 9000, 9001, 9123	Permitir	1000	vpc-sisem-preprod	Desactivado
fr-sisem-preprod-http	Entrada	Aplicar a todos	Intervalos de IP: t	tcp:80	Permitir	1000	vpc-sisem-preprod	Desactivado
fr-sisem-preprod-https	Entrada	Aplicar a todos	Intervalos de IP: t	tcp:443	Permitir	1000	vpc-sisem-preprod	Desactivado
fr-sisem-preprod-ssh	Entrada	Aplicar a todos	Intervalos de IP: t	tcp:22	Permitir	1000	vpc-sisem-preprod	Desactivado
gke-sisem-pre-prod-cluster-1-f63a535b-all	Entrada	gke-sisem-pr	Intervalos de IP: t	tcp, udp, icmp, esp, ah, sctp	Permitir	1000	vpc-sisem-preprod	Desactivado
gke-sisem-pre-prod-cluster-1-f63a535b-ssh	Entrada	gke-sisem-pr	Intervalos de IP: t	tcp:22	Permitir	1000	vpc-sisem-preprod	Desactivado
gke-sisem-pre-prod-cluster-1-f63a535b-vms	Entrada	gke-sisem-pr	Intervalos de IP: t	tcp:1-65535, udp:1-65535, icmp	Permitir	1000	vpc-sisem-preprod	Desactivado
k8s-a614c6db771d64f5fa0642bb4f8a0722-http-hc	Entrada	gke-sisem-pr	Intervalos de IP: t	tcp:32278	Permitir	1000	vpc-sisem-preprod	Desactivado
k8s-fw-a614c6db771d64f5fa0642bb4f8a0722	Entrada	gke-sisem-pr	Intervalos de IP: t	tcp:80, 443	Permitir	1000	vpc-sisem-preprod	Desactivado

Ilustración 34 Firewall

Las reglas de firewall de VPC permiten o rechazan las conexiones hacia o desde las instancias de máquina virtual (VM) configuradas para ejecutar o montar los nodos de KVM (kubernetes).

13.6.4. HTTPS/SSL

El punto de entrada es un Ingress el cual implemente SSL y solo recibe peticiones en HTTPS, para lo cual se implementa el servicio de cert-manager, lo que nos permite implementar y gestionar los certificados SSL.



Ilustración 35 Ej certificado preprod

13.6.5. Firewall Nodo

Cada nodo implementa un firewall propio del sistema operativo.

Seguridad		
Sandbox with gVisor	Inhabilitados	
Cuenta de servicio ?	default	
Permiso de acceso	API de Stackdriver Monitoring	Versión completa
	API de registro de Stackdriver	Solo escritura
	Administración de servicio	Solo lectura
	Control de servicio	Habilitado
	Google Storage	Solo lectura
	Stackdriver Trace	Solo escritura
Servidor de metadatos de GKE	Inhabilitada	
Supervisión de integridad	Habilitada	
Inicio seguro	Inhabilitada	

Ilustración 36 Firewall Nodo

13.6.6. Enrutamiento Kubernetes y API Gateway

Como capa extra de abstracción, las peticiones están enrutadas en el cluster por el motor de kubernetes y varios servicios, la API de SISEM, está detrás de un API Gateway el cual puede implementar CORS más estricto para mejorar seguridad.

✓ sisem-api-gateway-service	
DESCRIPCIÓN GENERAL	DETALLES
Clúster	sisem-pre-prod-cluster-1
Espacio de nombres	sisem-pre-prod
Fecha de creación	14 oct 2022 22:54:16
Etiquetas	app: sisem-api-gateway app.kubernetes.io/managed-by: Helm type: backend
Anotaciones	cloud.google.com/neg: {"ingress": true} meta.helm.sh/release-name: sisem meta.helm.sh/release-namespace: sisem-pre-prod
Selector de etiquetas	app = sisem-api-gateway type = backend
Pods	3 actual/3 deseado
Tipo	ClusterIP
IP del clúster	
IP del clúster	10.40.9.91

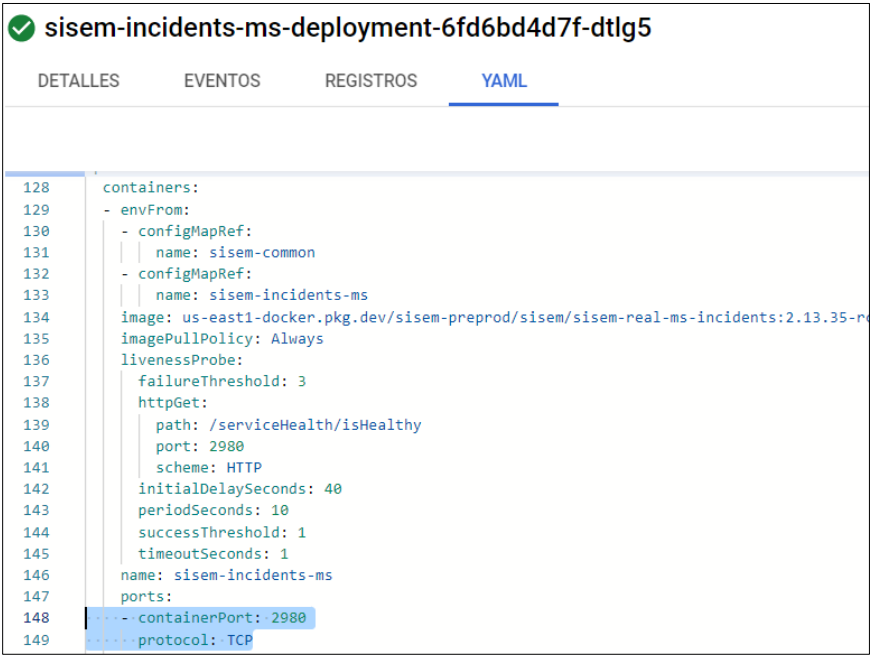
Ilustración 37 Gateway service

13.6.7. Pod

Cada petición la recibe el servicio que regula los pods, tanto los servicios como los pods tienen puertos puntuales para su operación y rechazan peticiones en puertos diferentes.

Pods					
Filtro Espacio de nombres : sisem-pre-prod Estado : OK: Running Filtra los pods					
Nombre	Estado	↓ CPU solicitada	Memoria solicitada	Almacenamiento solicitado	Espacio de nombres
sisem-elasticsearch-master-0	✓ Running	100 mCPU	1.07 GB	0 B	sisem-pre-prod
sisem-rabbitmq-service-1	✓ Running	100 mCPU	536.87 MB	0 B	sisem-pre-prod
sisem-incidents-ms-deployment-6fd6bd4d7f-8564n	✓ Running	100 mCPU	209.72 MB	0 B	sisem-pre-prod
sisem-incidents-ms-deployment-6fd6bd4d7f-dtlg5	✓ Running	100 mCPU	209.72 MB	0 B	sisem-pre-prod
sisem-incidents-ms-deployment-6fd6bd4d7f-nldbj	✓ Running	100 mCPU	209.72 MB	0 B	sisem-pre-prod

Ilustración 38 Pods del sistema



```
✓ sisem-incidents-ms-deployment-6fd6bd4d7f-dtlg5
DETALLES  EVENTOS  REGISTROS  YAML
128 containers:
129   - envFrom:
130     - configMapRef:
131       name: sisem-common
132     - configMapRef:
133       name: sisem-incidents-ms
134   image: us-east1-docker.pkg.dev/sisem-preprod/sisem/sisem-real-ms-incidents:2.13.35-r
135   imagePullPolicy: Always
136   livenessProbe:
137     failureThreshold: 3
138     httpGet:
139       path: /serviceHealth/isHealthy
140       port: 2980
141       scheme: HTTP
142     initialDelaySeconds: 40
143     periodSeconds: 10
144     successThreshold: 1
145     timeoutSeconds: 1
146   name: sisem-incidents-ms
147   ports:
148     - containerPort: 2980
149     - protocol: TCP
```

Ilustración 39 Configuración puerto pod

13.6.8. Validación Llave Pública

Los tokens del sistema son generados mediante keycloak, el cual expone la llave pública con la cual fueron generados para su validación desde los servicios.

Los servicios van a rechazar peticiones que no tengan la firma esperada (cada servicio obtiene la firma de keycloak para validar los tokens), por lo tanto, un token por más válido que sea, si no es generado por keycloak de SISEM será rechazado.

13.6.9. Validación Autenticación

Cuando el token sea válido se puede confirmar si la persona que dice ser realmente lo sea.

13.6.10. Validación Autorización

Diferentes endpoints no basta con confirmar la autenticación, muchas veces se requiere un permiso que esté asignado al usuario, el cual se ve reflejado como un rol dentro del token (Refiérase a un rol de un cliente/grupo de keycloak, no confundir con concepto de rol de SISEM).

Create resource-based permission

Name * ⓘ Bank account permission

Description ⓘ Defines the policies that apply to all bank accounts

Apply to resource type ☐ Off ⓘ

Resources * ⓘ bank-account ✕

Policies ⓘ Country policy ✕ Only Owner Policy ✕

Decision strategy ⓘ ☒ Unanimous ☐ Affirmative ☐ Consensus

Ilustración 40 Creación de permiso keycloak

Realm roles
Realm roles are the roles that you define for use in the current realm. [Learn more](#)

Search role by name → [Create role](#)

Role name	Composite	Description
admin	True	`\${role_admin}`
create-realm	False	`\${role_create-realm}`
default-roles-master ⓘ	True	`\${role_default-roles}`
offline_access	False	`\${role_offline-access}`
uma_authorization	False	`\${role_uma_authorization}`

Ilustración 41 Administración de roles keycloak

14. Ambiente de Producción

El ambiente de producción estará en alta disponibilidad utilizando GKE de Google Cloud, este componente nos permite desplegar en contenedores directamente sobre la nube y se administran por kubernetes, similar al ambiente de preproducción, pero configurando las reglas de escalabilidad con reglas manuales y no automáticas.

14.1. Proyecto

Para el ambiente productivo sobre GCP se crea un proyecto específico para el despliegue de SISEM:

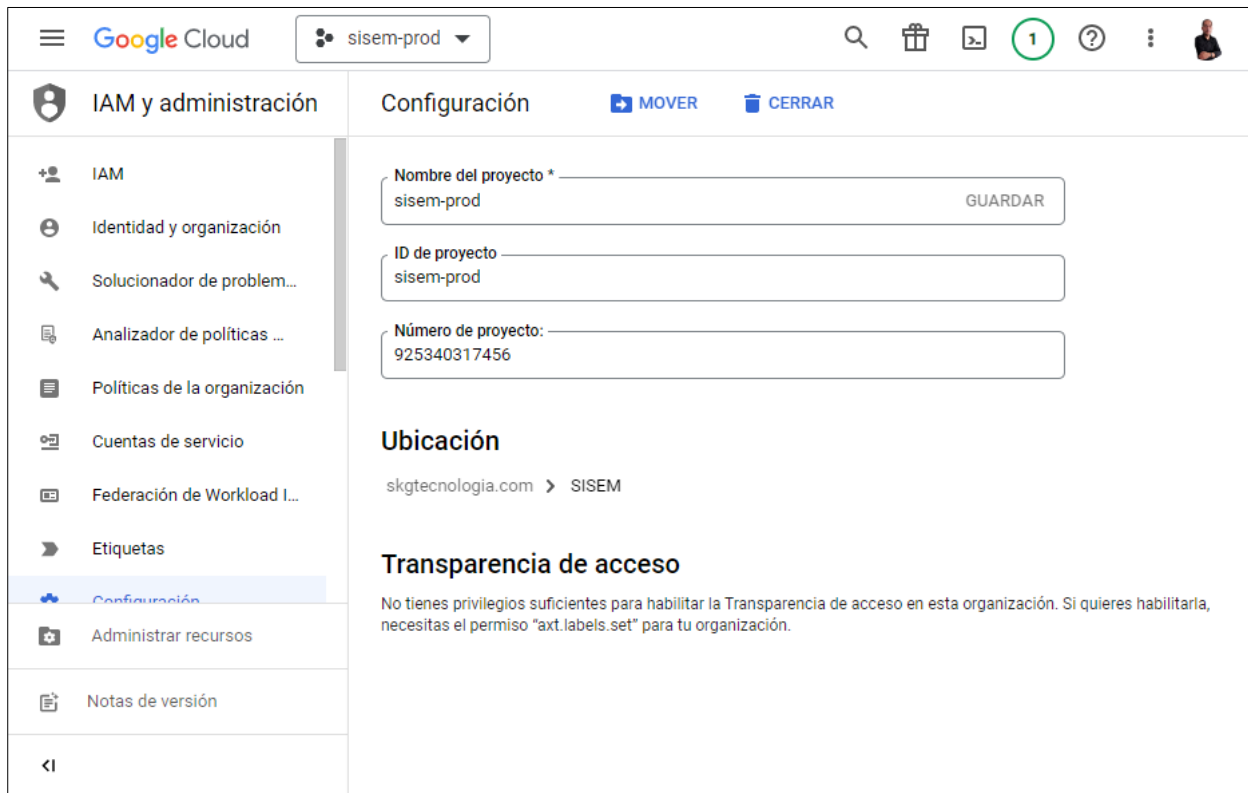


Ilustración 42 Proyecto SISEM

Este proyecto es configurado desde la opción del IAM y con los detalles:

Nombre: sisem-prod

ID: sisem-prod

14.2. Topología infraestructura Producción

A continuación, se muestra la topología de la infraestructura de ambiente de producción implementada en GCP Google Cloud Platform.

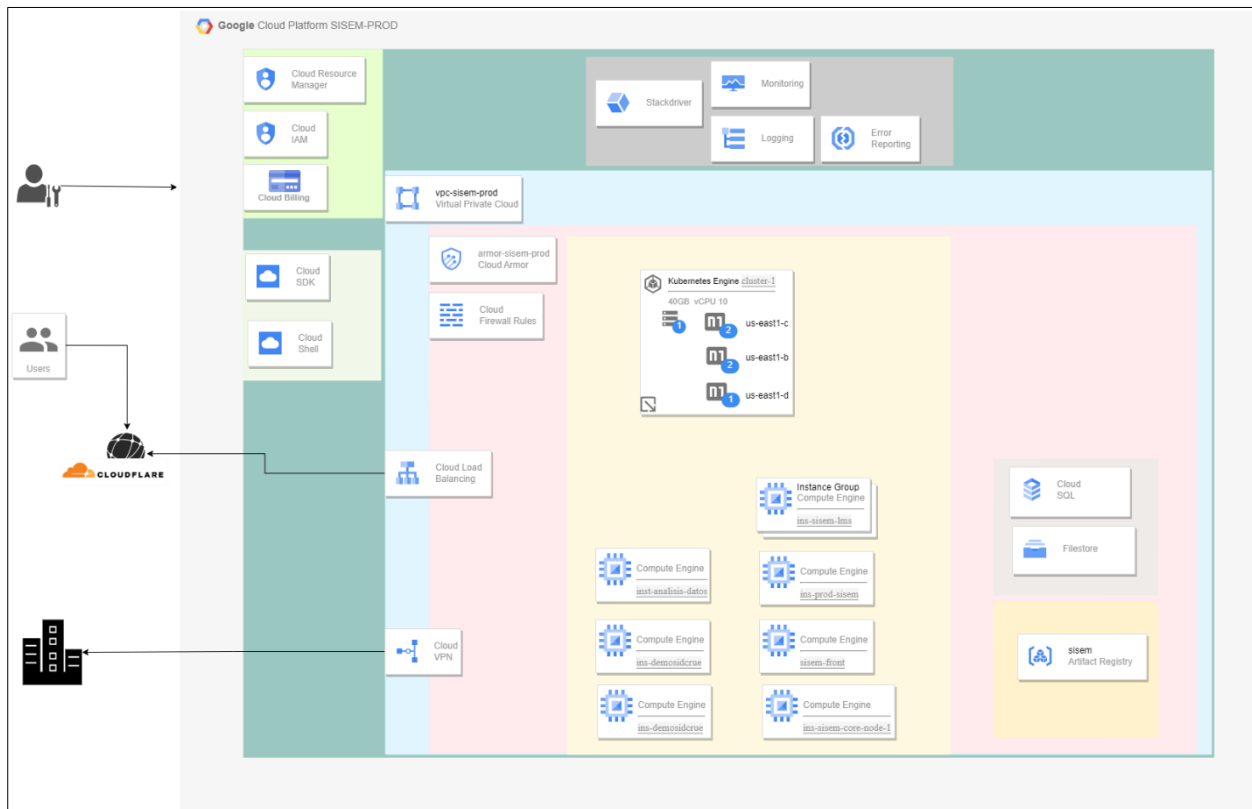


Ilustración 67 Topología infraestructura Producción

14.3. Cluster

Se crea el clúster para el ambiente de producción del SISEM cluster-prod con ubicación regional en tres zonas:

- us-east1-c
- us-east1-d
- us-east1-b

DETALLES	NODOS	ALMACENAMIENTO	OBSERVABILIDAD	REGISTROS
Aspectos básicos del clúster				
Nombre	cluster-1			🔒
Tipo de ubicación	Regional			🔒
Región	us-east1			🔒
Zonas predeterminadas de los nodos [?]	us-east1-c us-east1-d us-east1-b			✎
Canal de versiones	Canal regular			✎ ACTUALIZACIÓN DISPONIBLE
Versión	1.24.5-gke.600			
Tamaño total	6			ⓘ
Extremo externo	34.73.64.24 Mostrar certificado del clúster			✎
Extremo interno	172.20.0.7 Mostrar certificado del clúster			🔒
Automatización				
Período de mantenimiento	En cualquier momento			✎
Exclusiones de mantenimiento	Ninguno			
Notificaciones	Inhabilitado			✎
Ajuste de escala automático vertical de Pods	Inhabilitado			✎
Aprovisionamiento automático de nodos	Inhabilitadas			✎
Aprovisionamiento automático de etiquetas de red				✎
Perfil de ajuste de escala automático	Equilibrado			✎
Redes				
Clúster privado	Inhabilitado			🔒
Red	vpc-sisem-prod			🔒
Subred	sub-vpc-sisem-prod			🔒
Subred del extremo del plano de control privado	sub-vpc-sisem-prod			🔒
Enrutamiento del tráfico nativo de la VPC	Habilitado			🔒
Rango de direcciones del Pod del clúster (predeterminado)	10.188.0.0/14			🔒
Cantidad máxima de Pods por nodo	110			🔒
Rango de direcciones del servicio	10.192.0.0/20			🔒
Visibilidad dentro de los nodos	Inhabilitado			✎
Balanceo de cargas HTTP	Habilitado			✎
Subdivisión de balanceadores de cargas internos L4	Inhabilitado			✎
Redes autorizadas del plano de control	Inhabilitado			✎
Política de red	Inhabilitado			✎
Dataplane V2	Inhabilitado			🔒
Proveedor de DNS	Kube-dns			✎
NodeLocal DNSCache	Inhabilitadas			

Ilustración 43 Cluster

14.4. Nodos

Para el clúster de producción se crean 6 nodos, 2 por zona como se presenta en la siguiente imagen.

DETALLES

NODOS

ALMACENAMIENTO

OBSERVABILIDAD

REGISTROS

Grupos de nodos

Filtro

Filtra grupos de nodos

Nombre ↑	Estado	Versión	Cantidad de nodos	Tipo de máquina	Tipo de imagen	Ajuste de escala automático	Rango de direcciones IP del Pod	
default-pool	✓ Aceptar	1.24.5-gke.600	6 (2 por zona)	e2-medium	Container-Optimized OS con containerd (cos_containerd)	Desactivada	10.188.0.0/14	

Nodos

Filtro

Filtra los nodos

Nombre ↑	Estado	CPU solicitada	CPU asignable	Memoria solicitada	Memoria asignable	Almacenamiento solicitado	Almacenamiento asignable
gke-cluster-1-default-pool-43f74211-3mg0	✓ Ready	488 mCPU	940 mCPU	492.83 MB	2.95 GB	0 B	0 B
gke-cluster-1-default-pool-43f74211-lmk5	✓ Ready	248 mCPU	940 mCPU	387.97 MB	2.95 GB	0 B	0 B
gke-cluster-1-default-pool-44521c47-8r66	✓ Ready	488 mCPU	940 mCPU	492.83 MB	2.95 GB	0 B	0 B
gke-cluster-1-default-pool-44521c47-qsgx	✓ Ready	228 mCPU	940 mCPU	377.49 MB	2.95 GB	0 B	0 B
gke-cluster-1-default-pool-ade8f933-5lf4	✓ Ready	228 mCPU	940 mCPU	377.49 MB	2.95 GB	0 B	0 B
gke-cluster-1-default-pool-ade8f933-clhr	✓ Ready	297 mCPU	940 mCPU	526.95 MB	2.95 GB	0 B	0 B

Ilustración 44 Nodos producción

14.5. Reservar IP

Para producción se reserva una ip premium como se muestra:

Direcciones IP

RESERVAR DIRECCIÓN ESTÁTICA EXTERNA

ACTUALIZAR

LIBERAR DIRECCIÓN ESTÁTICA

TODOS

DIRECCIONES IP INTERNAS

DIRECCIONES IP EXTERNAS

DIRECCIONES IPV4

DIRECCIONES IPV6

Filtro

Ingresar el nombre o el valor de la propiedad

?

III

	Nombre	Dirección IP	Tipo de acceso	Región	Tipo ↓	Versión	En uso por	Subred	Red de VPC	Nivel de red ?	Etiquetas
☐	gke-cluster-1-7593cdb1-46df48f8-pe	172.20.0.7	Interno	us-east1	Estática	IPv4	Regla de reenvío gke-cluster-1-7593cdb1-46df48f8-pe	sub-vpc-sisem-prod		Premium	
☐	ippu-sisem-lms	34.111.20.61	Externo		Estática	IPv4	Regla de reenvío https-sisem-lms Regla de reenvío sisem-http			Premium	CAMBIAR
☐	lms-moodle	34.135.81.129	Externo	us-central1	Estática	IPv4	Instancia de VM ins-sisem-lms (Zona us-central1-a)			Premium	CAMBIAR
☒	main-ip	35.231.48.181	Externo	us-east1	Estática	IPv4	⚠ Ninguno			Premium	CAMBIAR
☐	vpc-sisem-prod-ip-range	10.107.112.0	Interno		Estática	IPv4	⚠ Ninguno	vpc-sisem-prod		Premium	

Ilustración 45 Reserva IP estática

14.6. Cloud VPC

Se define la red de VPC vpc-sisem-prod con 2 subredes para el manejo de los componentes de este proyecto y así mismo se establezcan reglas de seguridad sobre la misma.

Redes de VPC [CREAR RED DE VPC](#) [ACTUALIZAR](#)

REDES DEL PROYECTO ACTUAL SUBREDES DEL PROYECTO ACTUAL **NUEVO**

⚠ El puerto SMTP 25 está autorizado en este proyecto ⓘ

Redes de VPC

Filtro Ingresar el nombre o el valor de la propiedad ⓘ

Nombre ↑	Subredes	MTU ⓘ	Modo	Rangos de IP internas	Puertas de enlace	Reglas de firewall	Enrutamiento dinámico global
vpc-sisem-prod	2	1500	Personalizada			15	Desactivado

Ilustración 46 Cloud VPC Producción

14.7. Topología de Red

A continuación, se muestra el diagrama de red del ambiente producción, en donde se evidencia el flujo de red internamente y externamente hacia cada una de las zonas dispuestas en GCP:

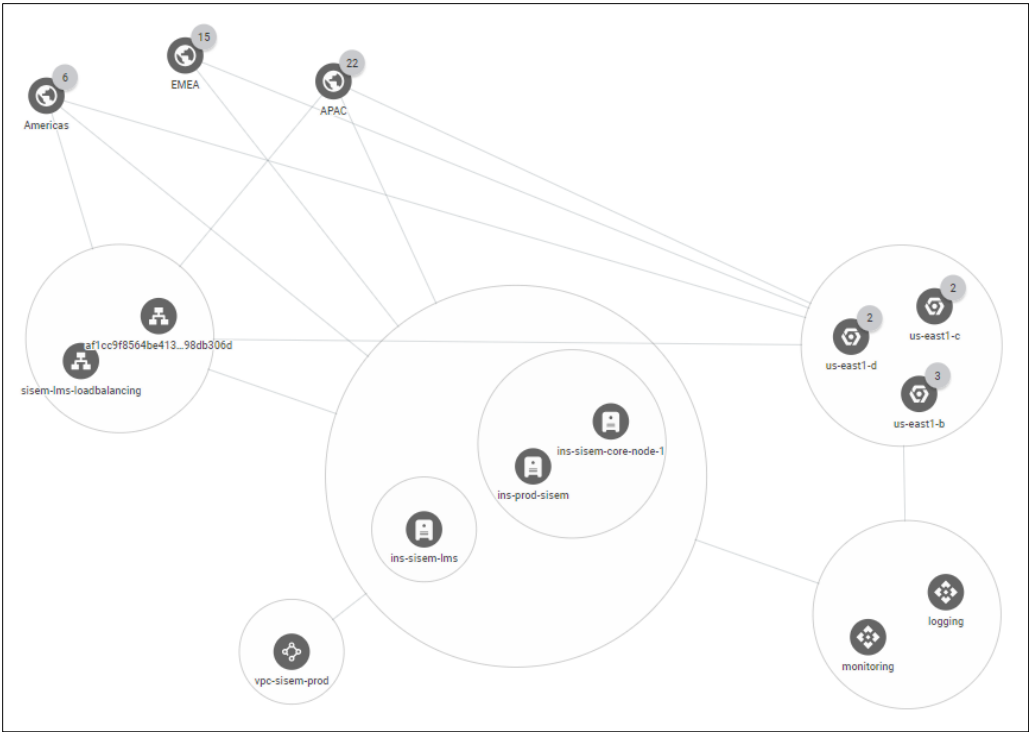


Ilustración 71 Topología de red

14.8. Seguridad

Para el apartado de seguridad aplica la misma configuración que se tiene en preproducción, puesto que este apartado es igual en ambos ambientes.

14.8.1. Firewall

Los nodos del cluster de kubernetes son instancias de la GCP de Compute Engine, por lo tanto, ya están en una VPC propia de ellos la cual proporciona otro firewall para reglas de acceso y egreso, las cuales se muestran a continuación:

☐	Nombre	Tipo	Destinos	Filtros	Protocolos/puertos	Acción	Prioridad	Red ↑	Registros	Recuento de hits ?	Último hit ?
☐	gke-cluster-1-7593cdb1-inkubelet	Entrada	gke-cluster-1	Intervalos de IP: 10	tcp:10255	Permitir	999	vpc-sisem-prod	Desactivado	—	—
☐	gke-cluster-1-7593cdb1-exkubelet	Entrada	gke-cluster-1	Intervalos de IP: 0.0	tcp:10255	Rechazar	1000	vpc-sisem-prod	Desactivado	—	—
☐	allow-gcp-iap	Entrada	allow-gcp-iap	Intervalos de IP: 35	tcp:22, 80, 443	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	allow-gcp-traffic-lb	Entrada	allow-gcp-tra	Intervalos de IP: 13	tcp:80, 443	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	fr-sisem-http	Entrada	Aplicar a tod	Intervalos de IP: 0.0	tcp:80	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	fr-sisem-https	Entrada	Aplicar a tod	Intervalos de IP: 0.0	tcp:443	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	fr-sisem-icmp	Entrada	Aplicar a tod	Intervalos de IP: 0.0	icmp	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	fr-sisem-ssh	Entrada	Aplicar a tod	Intervalos de IP: 0.0	tcp:22	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	fw-sisem-health-check	Entrada	allow-health-	Intervalos de IP: 13	tcp:80	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	gke-cluster-1-7593cdb1-all	Entrada	gke-cluster-1	Intervalos de IP: 10	tcp udp icmp,esp,ah,sctp	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	gke-cluster-1-7593cdb1-vms	Entrada	gke-cluster-1	Intervalos de IP: 17	tcp:1-65535 udp:1-65535 icmp	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	sisem-allow-nfs-in	Entrada	Aplicar a tod	Intervalos de IP: 10	all	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	vpc-sisem-prod-allow-http	Entrada	http-server	Intervalos de IP: 0.0	tcp:80	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	vpc-sisem-prod-allow-https	Entrada	https-server	Intervalos de IP: 0.0	tcp:443	Permitir	1000	vpc-sisem-prod	Desactivado	—	—
☐	sisem-allow-nfs-out	Salida	Aplicar a tod	Intervalos de IP: 10	all	Permitir	1000	vpc-sisem-prod	Desactivado	—	—

Ilustración 47 Firewall Producción

14.8.2. Cloud Armor

Adicionalmente con Google Cloud Armor podemos implementar en el ambiente un WAF, para lo cual se tiene propuesto el siguiente WAF:

armor-sisem-prod

Tipo

Política de seguridad del backend

Descripción

Políticas de seguridad para proyecto sisem-prod (OZA - SKG)

Contiene

4 reglas

Se aplica a

1 objetivo

Protección adaptable

Inhabilitada

REGLAS

DESTINOS

REGISTROS

Se evalúan las reglas por prioridad, es decir, se analizan las cifras más bajas en primer lugar. [Más información](#)

AGREGAR REGLA

BORRAR

MÁS ▾

Ilustración 48 WAF armor-sisem-prod

Para el cual se definen una serie de reglas con el fin de proteger el backend del proyecto SISEM, las reglas definidas por el momento se muestran a continuación, en la cual podemos ver la acción que se realiza, las coincidencias, su descripción y por último su respectiva prioridad en el sistema:

☐	Acción	Tipo	Coincidencia	Descripción	Prioridad ↑	
☐	Denegar (502)		evaluatePreconfiguredExpr('rce-canary',['owasp-crs-v030001-id932160-rce','owasp-crs-v030001-id932150-rce','owasp-crs-v030001-id932130-rce','owasp-crs-v030001-id932105-rce','owasp-crs-v030001-id932110-rce','owasp-crs-v030001-id932115-rce','owasp-crs-v030001-id932100-rce'])	Ejecución de código remoto (RCE)	994	⋮
☐	Denegar (502)		evaluatePreconfiguredExpr('sqli-stable',['owasp-crs-v030001-id942420-sqli','owasp-crs-v030001-id942421-sqli','owasp-crs-v030001-id942200-sqli','owasp-crs-v030001-id942260-sqli','owasp-crs-v030001-id942432-sqli','owasp-crs-v030001-id942340-sqli','owasp-crs-v030001-id942430-sqli','owasp-crs-v030001-id942380-sqli','owasp-crs-v030001-id942431-sqli','owasp-crs-v030001-id942460-sqli','owasp-crs-v030001-id942440-sqli','owasp-crs-v030001-id942110-sqli','owasp-crs-v030001-id942120-sqli','owasp-crs-v030001-id942330-sqli','owasp-crs-v030001-id942210-sqli','owasp-crs-v030001-id942450-sqli','owasp-crs-v030001-id942300-sqli','owasp-crs-v030001-id942310-sqli','owasp-crs-v030001-id942350-sqli','owasp-crs-v030001-id942410-sqli','owasp-crs-v030001-id942180-sqli','owasp-crs-v030001-id942320-sqli','owasp-crs-v030001-id942150-sqli'])	SQL Injection Nivel 3	995	⋮
☐	Permitir		origin.region_code == 'CO'	Permitir Regiones norma internacional ISO 3166-1	1,000	⋮
☐	Denegar (403)	Direcciones IP o rangos de IP	*(Todas las direcciones IP)	Regla predeterminada, una prioridad más alta la anula	2,147,483,647	⋮

Ilustración 49 Reglas definidas en WAF

Cabe aclarar que el WAF será implementado el día de paso a producción de SISEM. La configuración del correo para el envío de comunicaciones será implementada el día de paso a producción de SISEM.

Se agregarán las reglas de excepciones de sistemas desde el exterior (Ej. Azure, gov.co). Un mes después luego del paso a producción, las estadísticas de: Cloud armor, Firewall y WAF generadas por GCP serán presentadas en un tablero de estadísticas a SDS.

15. Referencias

- [1] DeCandia, Giuseppe; Hastorun, Deniz; Jampani, Madan; et.al. (2007). Dynamo: Amazon's Highly Available Key-valueStore. Communications of the ACM.
- [2] DOCKER, Docker. What is Docker. [En línea] [Consultado 16 mayo 2022]. Disponible en <<https://www.docker.com/what-docker>>
- [3] KUBERNATES, Kubernetes, ¿Qué es Kubernetes? [En línea] [Consultado 16 mayo 2022]. Disponible en <<https://kubernetes.io/es/docs/concepts/overview/what-is-kubernetes/>>
- [4] Comunidad DragonJAR. Hardening, Conceptos básicos [En línea] [Consultado 20 septiembre 2022]. Disponible en <https://www.dragonjar.org/hardening-conceptos-basicos.xhtml>.
- [5] Robledano Angel. (2019). Qué es IPv6. [En línea] [Consultado 20 septiembre 2022]. Disponible en <https://openwebinars.net/blog/que-es-ipv6/>
- [6] <https://cloud.google.com/compute>
- [7] https://cloud.google.com/kubernetes-applications/?utm_source=google&utm_medium=cpc&utm_campaign=latam-CO-all-es-dr-BKWS-all-all-trial-e-dr-1011454-LUAC0016124&utm_content=text-ad-none-any-DEV_c-CRE_542358547663-ADGP_Hybrid%20%7C%20BKWS%20-%20EXA%20%7C%20Txt%20~%20Containers_Kubernetes-KWID_43700070988485070-kwd-1653888087293&utm_term=KW_google%20cloud%20kubernetes%20use-ST_Google%20Cloud%20kubernetes%20use&gclid=Cj0KCQjwj7CZBhDHARIsAPPWv3cy9ZBGGEF_QGZLRf_a32GuXkXlhGQgw-7cxqwYmszfavX5LzavNM4aAn5TEALw_wcB&gclsrc=aw.ds
- [8] <https://cloud.google.com/network-connectivity/docs/vpn?hl=es-419>
- [9] <https://kubernetes.io/es/docs/concepts/architecture/nodes/>
- [10] <https://kubernetes.io/es/docs/concepts/workloads/pods/pod/>