Prueba Técnica para Administrador de Nube 🔑☁️

Duración: Tienes un plazo de 24 horas a partir de la recepción de este correo para completar y enviar la prueba.

Envío: Una vez completada la prueba, por favor envíala como un archivo adjunto en formato PDF al correo electrónico

Prepárate para sustentar tus respuestas.

Escenario 1: Preguntas de Selección Múltiple 📝

Instrucciones: Selecciona la respuesta correcta para cada una de las siguientes preguntas y de una justificación corta del por qué.

1. ¿Cuál es la herramienta de GCP recomendada para la Infraestructura como Código (IaC)?
    a. Chef
    b. Puppet
    c. Terraform
    d. SaltStack
2. Si necesitas crear un clúster de Kubernetes en GCP, ¿qué servicio utilizarías?
    a. Cloud Functions
    b. Compute Engine
    c. App Engine
    d. GKE
3. ¿Cuál es la función principal de Cloud Armor?
    a. Monitorear el rendimiento de las aplicaciones.
    b. Almacenar objetos de forma duradera.
    c. Proteger las aplicaciones de ataques DDoS y WAF (Web Application Firewall).
    d. Administrar las identidades y accesos.
4. En GCP, ¿qué servicio se utiliza para ejecutar código sin servidor en respuesta a eventos?
    a. Cloud Run
    b. Cloud Functions
    c. Compute Engine
    d. App Engine

5. ¿Qué servicio de GCP debes usar para un despliegue de contenedores completamente gestionado y sin servidor que escala automáticamente?
    a. GKE
    b. Cloud Functions
    c. **Cloud Run**
    d. Compute Engine

6. Para almacenar archivos estáticos y objetos de manera segura y escalable, ¿qué servicio de GCP es el más adecuado?
    a. Cloud Storage
    b. Cloud SQL
    c. BigQuery
    d. Datastore

Escenario 2: Ejercicio Práctico - Despliegue de una Aplicación Web Sencilla 💻

Se requiere realizar el despliegue de una aplicación web sencilla utilizando las líneas de comando de Gcloud y Terraform. Para ello debes realizar las siguientes tareas:

Configuración de la Infraestructura con Terraform

1. Cree un archivo main.tf para definir los siguientes recursos:
    - Una red VPC llamada web-app-vpc.
    - Una subred llamada web-app-subnet en la región us-central1.
    - Una instancia de Compute Engine (e2-medium) con el sistema operativo ubuntu, llamada web-server-1, ubicada en la subred.
    - Una regla de firewall que permita el tráfico HTTP (puerto 443) desde cualquier dirección IP (0.0.0.0/0) hacia la instancia.
2. Ejecute los comandos terraform init, terraform plan y terraform apply.

Despliegue de la Aplicación y Verificación

- Conéctese por SSH a la instancia de Compute Engine.
- Instale Nginx como servidor web.
- Cree un archivo index.html con el contenid.
- Verifique que la aplicación es accesible desde algún lugar.

Entregables Esperados

- Url, IP o link de acceso a la aplicación
- Evidencias de lo ejecutado, arquitectura seleccionada y demás información relevante
- Archivo .tf para la IaC
- Comandos Gcloud utilizados

Escenario 3: Ejercicio Práctico Completo con CI/CD y Seguridad 🚀

Debe realizar el despliegue de una aplicación web simple que se encuentre en un repositorio de GitHub, utilizando IaC con cloud build y desplegada en cloud run, a la vez debe implementar buenas prácticas de seguridad.

Para ello debe tener en cuenta las siguientes tareas

Requisito de Repositorio: Clone un repositorio de GitHub que encuentre en internet.

Tarea 1: Despliegue con Cloud Build y Cloud Run

- Configure un disparador de Cloud Build que se active con cada push en la rama main del repositorio.
- El cloudbuild.yaml debe construir la imagen de Docker, empujarla a Artifact Registry y desplegar el servicio en Cloud Run, asegurando que solo reciba tráfico HTTPS.
- Realice un commit y push para demostrar que el pipeline se ejecuta y la aplicación se actualiza.

Tarea 2: Gestión de la Seguridad (IAM y Cloud Armor)

- Cree un rol de IAM personalizado que solo permita la gestión de servicios de Cloud Run (roles/run.admin).
- Utilice Terraform para crear una política de Cloud Armor que bloquee el tráfico de una dirección IP específica (ej: 1.2.3.4).
- Asocie esta política al servicio de Cloud Run.

Verificación: Intente acceder a la URL del servicio desde la IP bloqueada para confirmar que la regla de Cloud Armor está funcionando.

Entregables esperados

- Un repositorio de GitHub que contenga todos los archivos necesarios (app.py, Dockerfile, cloudbuild.yaml, main.tf).
- Una URL de servicio de Cloud Run que esté activa y responda a las solicitudes.
- La confirmación de que la política de seguridad de Cloud Armor está funcionando, demostrando que una IP de prueba es bloqueada.
- Evidencias de lo ejecutado, arquitectura seleccionada y demás información relevante