proyecto.

Infraestructura
Viva

S A N D R A
G O N Z Á L E Z

julio 2025

GUÍA DE BUENAS PRÁCTICAS

Guía de Buenas Prácticas

1. Seguridad

La arquitectura del sistema fue diseñada siguiendo principios de seguridad en profundidad, asegurando el aislamiento de componentes, el control granular de accesos y la respuesta proactiva ante incidentes. A continuación se detallan las principales medidas aplicadas:

1.1 Aislamiento de redes y control de acceso

Subredes segmentadas

La red VPC se dividió en subredes públicas y privadas para aislar los distintos componentes del sistema:

- En las subredes públicas se desplegaron los servicios expuestos al exterior, como el balanceador de carga y el sitio web estático alojado en Amazon S3.
- En las subredes privadas se ubicaron los recursos internos, como la instancia EC2, las bases de datos RDS y DynamoDB.

Grupos de Seguridad (SG)

Se implementaron reglas estrictas para cada SG, aplicando el **principio de menor privilegio**:

- La instancia EC2 solo permite tráfico HTTP (puerto 80) y SSH (puerto 22) desde direcciones IP confiables.
- Las bases de datos RDS y DynamoDB fueron configuradas para aceptar conexiones **únicamente desde la EC2** dentro de la misma VPC, sin exposición pública.

Listas de Control de Acceso a la Red (NACLs)

Las NACLs se configuraron a nivel de subred, reforzando el control de tráfico:

- Se definieron reglas específicas para permitir solo conexiones entrantes y salientes estrictamente necesarias.
- Toda conexión no autorizada se bloquea por defecto.

IAM Roles y Políticas

Cada servicio en la arquitectura (como EC2, Lambda y CloudWatch) cuenta con un **rol IAM asociado**, limitado a los permisos estrictamente necesarios para su funcionamiento.

Esto evita el uso de credenciales permanentes y permite auditar cada acción ejecutada.

1.2 Seguridad en capas

La seguridad fue implementada en múltiples niveles, considerando la arquitectura por capas:

- Cada componente (web, lógica, base de datos) tiene su propio grupo de seguridad.
- El acceso SSH a la capa de aplicación se restringió exclusivamente a direcciones IP específicas autorizadas.
- Esta segmentación asegura que un eventual compromiso en una capa no se propague a las demás.

1.3 Alarmas y Monitoreo (CloudWatch + SNS)

Se habilitó la supervisión de recursos mediante **Amazon CloudWatch**, configurando alarmas críticas:

- Uso excesivo de CPU, tráfico de red y fallos de estado en la instancia EC2.
- Las alarmas están integradas con **Amazon SNS**, que notifica al correo electrónico del administrador en tiempo real.
- Durante las pruebas no se detectaron eventos críticos, pero la infraestructura está lista para escalar su monitoreo en ambientes productivos.

2. Escalabilidad y optimización de costos

Amazon EC2:

- Se implementó un balanceador de carga (ALB) que permite distribuir el tráfico entrante entre múltiples instancias EC2.
- Aunque en esta etapa no se habilitó Auto Scaling, la arquitectura está preparada para integrarse fácilmente con grupos de Auto Scaling basados en métricas como uso de CPU o tráfico.

Amazon RDS y DynamoDB:

- **RDS** permite escalar verticalmente (cambiando a instancias más potentes) o agregar réplicas de lectura para mejorar rendimiento y disponibilidad.
- **DynamoDB** está configurado en modo on-demand, escalando automáticamente según la demanda de uso.
- Aunque no se desplegó RDS en esta fase, se considera la política de habilitar backups automáticos para recuperación ante pérdida o corrupción de datos.

Amazon S3:

- Almacenamiento con escalabilidad prácticamente ilimitada y sin intervención manual.
- Se aplicó una política de ciclo de vida para mover objetos antiguos hacia la clase de almacenamiento Glacier, optimizando costos.

Control de versiones y colaboración (GitHub):

- Todo el código fuente, scripts de prueba y documentación se mantiene versionado en un repositorio GitHub, siguiendo buenas prácticas de trazabilidad, colaboración y control de cambios.

3. Administración de red

Para garantizar un entorno seguro, escalable y controlado, se implementó una arquitectura de red basada en Amazon VPC (Virtual Private Cloud). A continuación, se describen sus principales configuraciones:

3.1 Amazon VPC personalizada

Se creó una VPC propia para aislar los recursos internos de la nube pública, incrementando la seguridad y el control sobre el tráfico de red.

Subredes:

- **2 subredes públicas**, destinadas a componentes que requieren acceso desde Internet, como el Application Load Balancer.

- **1 subred privada**, utilizada para alojar servicios internos como instancias EC2 y, en entornos futuros, una base de datos en Amazon RDS.

3.2 Conectividad a Internet

- **Internet Gateway**: Se asoció un Internet Gateway a la VPC, permitiendo que las subredes públicas tengan acceso a Internet.
- **Tablas de enrutamiento**: Se configuraron tablas de rutas diferenciadas para dirigir correctamente el tráfico entre subredes públicas, privadas e Internet.

3.3 DNS interno

Se activó el DNS interno de la VPC, permitiendo que los recursos se comuniquen entre sí utilizando nombres en lugar de direcciones IP. Esto facilita el despliegue automatizado y la gestión de servicios internos.

3.4 IP elástica (EIP)

No se utilizó Elastic IP en este entorno para evitar el consumo innecesario del plan gratuito. Sin embargo, se contempla su uso en ambientes productivos donde se requiere una dirección IP estática para acceder a servicios críticos.

4. Buenas prácticas adicionales

Se incorporaron varias medidas que mejoran la disponibilidad, seguridad y mantenibilidad del sistema:

- **Backups automáticos en RDS**: Se configuraron copias de seguridad diarias para garantizar la recuperación ante pérdida de datos.
- **Política de ciclo de vida en S3**: Los archivos antiguos se migran automáticamente a Amazon S3 Glacier, optimizando costos de almacenamiento a largo plazo.
- **Monitoreo con Amazon CloudWatch**: Se activaron alarmas para detectar anomalías en el rendimiento, consumo o disponibilidad del sistema antes de que generen fallos críticos.
- **Alertas mediante Amazon SNS**: Las alarmas de CloudWatch se integran con Amazon SNS para notificar automáticamente al equipo técnico en caso de incidentes.
- **Gestión de código fuente con GitHub**: Se mantiene un repositorio con control de versiones, documentación asociada y trazabilidad de cambios, facilitando la colaboración y mantenibilidad del sistema.