

Caso Práctico: Gestión de infraestructura como código (IaC) y ambientes de nube AWS

Contexto:

Como parte del equipo de infraestructura, se te han delegado un conjunto de historias de usuario, repartidas en varios sprints, orientadas a la gestión de IaC y los ambientes de nube de deuna!.

Objetivo:

Resolver las historias a continuación descritas, o plantear una solución según corresponda, tomando en cuenta aspectos de: costo-eficiencia, seguridad, operación, observabilidad.

Tareas:

1. Monitoreo del escalamiento de microservicios:

- **Descripción del escenario:** se mantiene implementado un clúster de ECS con varios servicios ejecutándose en el. Cada servicio mantiene una configuración de tareas deseadas, mínimas y máximas diferente. Todos los servicios tienen una misma política de escalamiento definida, en donde, se creará una nueva tarea cuando el uso del procesador y/o memoria pase el umbral del 60% en promedio. Deuna! mantiene una herramienta de observabilidad diferente a la provista por Cloudwatch, que provee la capacidad de ingestar el contenido de cloudwatch logs.
- **Requerimiento de la historia:** plantee y describa con el mayor detalle un escenario de solución en donde se pueda monitorear y alertar cuando un microservicio ha escalado al máximo de tareas permitido.

2. Monitoreo de costos de transacciones monetarias y no monetarias:

- **Descripción del escenario:** deuna! app dispone de un conjunto de transacciones monetarias (recargas, pagos, etc) y no monetarias (onboarding) que hacen uso de un conjunto de servicios de AWS. Cada servicio de AWS consumido tiene su propio esquema de costeo.
- **Requerimiento de la historia:** plantee y describa con el mayor detalle una solución al escenario antes indicado, en donde se busca de manera automática identificar y distinguir cada transacción monetaria y no monetaria y en base al modelo de costo de cada servicio AWS utilizado, obtener el costode cada transacción.

3. Implementación de recursos AWS mediante IaC:

- Diseñe e implemente un repositorio de IaC basado en Terraform en donde se implementen los siguiente recursos:
 - Una llave KMS

- Un bucket S3 encriptado con la llave KMS antes creada. Incluir en el bucket una política en la que se apliquen mejores prácticas de seguridad para este tipo de recursos
- Un cluster de RDS basado en una instancia tipo Aurora Postgres.
- Un secreto en donde se almacene las credenciales del usuario pgadmin.