

Diseño de una API serverless para gestión de datos

Este documento presenta el diseño de una API sin servidor basada en AWS Lambda y API Gateway, para la gestión de registros de usuarios almacenados en una base de datos DynamoDB. La API implementa las operaciones básicas de alta, consulta y eliminación de registros.

Escenario

La organización necesita una API ligera y escalable que permita gestionar usuarios en DynamoDB. La API debe exponer tres endpoints principales: POST, GET y DELETE, los cuales se conectan a funciones Lambda que procesan las solicitudes y se comunican con DynamoDB.

Endpoints definidos

Endpoint	Función Lambda asociada	Acción en DynamoDB
POST /usuarios	LambdaCrearUsuario	PutItem (crear nuevo registro)
GET /usuarios	LambdaListarUsuarios	Scan (listar todos los registros)
DELETE /usuarios/{id}	LambdaEliminarUsuario	DeleteItem (eliminar por ID)

Conexión con DynamoDB

- Cada función Lambda utiliza un rol IAM con permisos mínimos necesarios:
 - * dynamodb:PutItem para creación.
 - * dynamodb:Scan para consulta.
 - * dynamodb>DeleteItem para eliminación.
- Se habilita CloudWatch para registrar invocaciones y errores.

Flujo general

1. El cliente realiza una petición HTTP (POST, GET o DELETE).
2. API Gateway recibe la solicitud y la enruta al endpoint correspondiente.
3. La función Lambda procesa la lógica de negocio y accede a DynamoDB.
4. La respuesta es devuelta al cliente a través de API Gateway.

Ventajas frente a una API tradicional

- Escalabilidad automática sin necesidad de administrar servidores.
- Costos por uso, se paga únicamente cuando hay invocaciones.
- Arquitectura simple y modular, fácil de extender.
- Integración nativa con DynamoDB y monitoreo con CloudWatch.

Conclusión

La API serverless propuesta con Lambda y API Gateway permite gestionar registros en DynamoDB de forma eficiente, segura y escalable, reduciendo costos operativos y eliminando la necesidad de infraestructura servidor dedicada.