

AYGO - Pruebas de Carga Distribuida en AWS

Documento de Arquitectura Profesional

Daniel Ricardo Vergara Fonseca Juan Felipe Santos Carvajal

María José Torres Nieves

Problemática

En el entorno dinámico del desarrollo de software en una empresa colombiana, nos enfrentamos a desafíos cruciales relacionados con el rendimiento y la escalabilidad de las APIs REST. La complejidad creciente de estas APIs hace imperativo para la empresa garantizar la calidad y cumplir con las expectativas de sus clientes. En este contexto, surgen necesidades específicas que deben abordarse de manera estratégica para asegurar el éxito continuo de la empresa.

Desafíos y Necesidades:

- 1. Identificación de Cuellos de Botella:** Evaluar el rendimiento bajo cargas intensivas es esencial para identificar cuellos de botella y abordar problemas antes de que impacten a los usuarios reales.
- 2. Validación de la Resistencia:** La evaluación de la resistencia bajo cargas sostenidas es clave para garantizar un rendimiento confiable en condiciones de uso continuo.
- 3. Validación de Escalabilidad:** Evaluar la capacidad de la aplicación para escalar horizontalmente es crucial para determinar si los recursos de la nube pueden manejar eficientemente un aumento en la carga de trabajo.
- 4. Optimización de Recursos y Costos:** Identificar los recursos óptimos para la carga prevista permite optimizar la infraestructura y reducir costos operativos, evitando sub o sobre aprovisionamiento.
- 5. Monitoreo y Diagnóstico en Tiempo Real:** Monitorear en tiempo real las pruebas de carga distribuidas permite identificar problemas de rendimiento proactivamente, facilitando correcciones eficientes.

Solución Estratégica

La implementación de pruebas de carga distribuidas en AWS se presenta como una solución estratégica para abordar estos desafíos. Al simular condiciones del mundo real, estas pruebas permiten evaluar el rendimiento bajo cargas significativas, mejorando así la robustez de las aplicaciones y asegurando una evolución efectiva con requisitos cambiantes. Esta estrategia no solo respalda la reputación de la empresa, sino que también fortalece su competitividad en el dinámico mercado del desarrollo de software.

Beneficios Específicos

La implementación incluye la prueba de capacidades de carga mediante contenedores independientes de Amazon ECS en AWS Fargate. Además, ofrece la automatización de pruebas de carga al programar su inicio de forma automática en fechas específicas o recurrentes. La posibilidad de probar aplicaciones personalizadas mediante la creación de scripts personalizados de JMeter y visualizar datos en tiempo real a través de la consola web de la solución contribuye significativamente a la eficiencia y efectividad del proceso de pruebas.

Arquitectura de la Solución

La arquitectura de la solución se presenta a continuación:

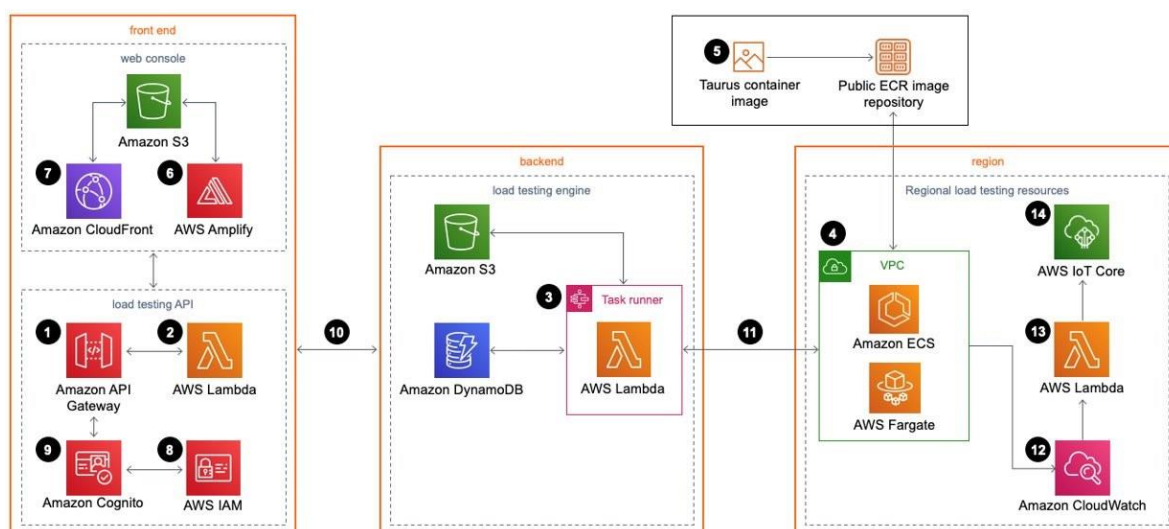


Figura 1. Diagrama de flujo de la arquitectura de Pruebas de carga distribuidas en AWS

La arquitectura implementa diversos recursos, entre ellos:

1. Una API de prueba de carga distribuida, que aprovecha Amazon API Gateway para invocar los microservicios de la solución (funciones de AWS Lambda).
2. Los microservicios proporcionan la lógica empresarial para gestionar los datos de prueba y ejecutar las pruebas.
3. Estos microservicios interactúan con Amazon Simple Storage Service (Amazon S3), Amazon DynamoDB y AWS Step Functions para proporcionar almacenamiento para los detalles y resultados del escenario de prueba y ejecutar escenarios de prueba.
4. Se implementa una topología de red de Amazon Virtual Private Cloud (Amazon VPC) que contiene los contenedores de Amazon Elastic Container Service (Amazon ECS) de la solución que se ejecutan en AWS Fargate.
5. Los contenedores incluyen la imagen de contenedor compatible con Open Container Initiative (OCI) de prueba de carga de Taurus (marco de automatización de pruebas de código abierto) que se utiliza para generar carga para probar el rendimiento de su

aplicación. La imagen del contenedor está alojada en AWS en un repositorio público de Amazon Elastic Container Registry (Amazon ECR).

6. Se implementa una consola web con tecnología de AWS Amplify en un depósito de Amazon S3 configurado para alojamiento web estático.
7. Amazon CloudFront proporciona acceso público seguro al contenido del depósito del sitio web de la solución.
8. Durante la configuración inicial, esta solución también crea una función de administrador de solución predeterminada (rol IAM) y envía una invitación de acceso a una dirección de correo electrónico de usuario especificada por el cliente.
9. Un grupo de usuarios de Amazon Cognito administra el acceso de los usuarios a la consola y a la API del probador de carga distribuida.
10. Después de implementar esta solución, puede utilizar la consola web para crear un escenario de prueba que defina una serie de tareas.
11. Los microservicios utilizan este escenario de prueba para ejecutar Amazon ECS en tareas de AWS Fargate en las regiones especificadas.
12. Además de almacenar los resultados en Amazon S3 y DynamoDB, una vez que se completa la prueba, el resultado se registra en Amazon CloudWatch.
13. Si selecciona la opción de datos en vivo, la solución envía los registros de Amazon CloudWatch para las tareas de AWS Fargate a una función Lambda durante la prueba, para cada región en la que se ejecutó la prueba.
14. Luego, la función Lambda publica los datos en el tema correspondiente en AWS IoT Core en la región donde se implementó la pila principal. La consola web se suscribe al tema y podrá ver los datos mientras se ejecuta la prueba en la consola web.

Detalles Técnicos

La solución se fundamenta en la utilización de Amazon ECS en conjunto con AWS Fargate para la ejecución de contenedores que generan carga a través de Taurus. La consola web, potenciada por AWS Amplify, proporciona una interfaz amigable para la creación y monitoreo de pruebas de carga distribuidas.

Costos y Consideraciones Económicas

El costo total de la solución depende del número y duración de las pruebas de carga realizadas. La estimación para la configuración predeterminada en la región Este de EE. UU. es de aproximadamente \$30.90 por mes. Es crucial tener en cuenta las variaciones en los recursos, especialmente a partir de la versión 1.3.0, que incrementa la CPU y la memoria, afectando el costo.

AWS service	Dimensions	Cost [USD]
AWS Fargate	10 on-demand tasks (using two vCPUs and 4 GB memory) running for 30 hours	\$29.62
Amazon DynamoDB	1,000 on-demand write capacity units 1,000 on-demand read capacity units	\$0.0015
AWS Lambda	1,000 requests 10 minutes total duration	\$1.25
AWS Step Functions	1,000 state transitions	\$0.025
Total:		\$30.90 per month

Se puede ajustar los recursos según tus necesidades para optimizar los costos.

Seguridad en la Implementación

La implementación de esta solución aprovecha la robustez de la seguridad de AWS. Los Roles de IAM permiten asignar políticas de acceso granulares, garantizando la seguridad en el manejo de servicios y usuarios en AWS. Además, se implementa Amazon CloudFront para proporcionar acceso seguro y público al contenido del bucket del sitio web.

Grupo de seguridad de AWS Fargate

De manera predeterminada, esta solución expone la regla de salida del grupo de seguridad de AWS Fargate al público. En caso de que desees restringir a AWS Fargate para que no envíe tráfico indiscriminado, se sugiere modificar la regla de salida para apuntar a una Ruta de Inter-Dominio sin Clase (CIDR) específica. Dentro de este grupo de seguridad, también se encuentra una regla de entrada que habilita el tráfico local en el puerto 50,000 para cualquier fuente asociada al mismo grupo de seguridad. Este ajuste se implementa con el propósito de facilitar la comunicación entre los contenedores.

Roles de IAM

La solución crea varios roles de IAM, incluyendo roles que otorgan a la función AWS Lambda acceso a los otros servicios de AWS utilizados en esta solución.

Amazon CloudFront

Esta implementación despliega un sitio web estático que se encuentra alojado en un depósito de Amazon Simple Storage Service (Amazon S3). Con el objetivo de disminuir la latencia y reforzar la seguridad, se ha incorporado una distribución de Amazon CloudFront, la cual cuenta con una identidad de origen. Esta identidad de origen representa un usuario especial de CloudFront que contribuye a facilitar un acceso seguro y público al contenido almacenado en el depósito del sitio web de esta solución. Para obtener detalles adicionales, se recomienda

revisar la sección sobre la restricción del acceso al contenido de Amazon S3 mediante el uso de una Identidad de Origen.

Prueba de estrés de red

Esta política cubre situaciones como si estás planeando ejecutar pruebas de red de alto volumen directamente desde tus instancias de Amazon EC2 a otros lugares como otras instancias de Amazon EC2, propiedades/servicios de AWS, o puntos finales externos. Estas pruebas a veces se llaman pruebas de estrés, pruebas de carga, o pruebas de día de juego. La mayoría de las pruebas de los clientes no caerán bajo esta política, sin embargo, consulta esta política si crees que estarás generando tráfico que se sostenga, en agregado, durante más de 1 minuto, sobre 1 Gbps (1 billón de bits por segundo) o sobre 1 Gpps (1 billón de paquetes por segundo).

Restringiendo el Acceso a la Interfaz de Usuario Pública

Para restringir el acceso a la interfaz de usuario más allá de los mecanismos de autenticación y autorización proporcionados por IAM y Amazon Cognito, se recomienda utilizar la solución de Automatizaciones de Seguridad de AWS WAF (firewall de aplicaciones web).

Conclusiones y Recomendaciones

La implementación de pruebas de carga distribuidas en AWS proporciona una solución integral y eficaz para enfrentar los desafíos asociados con el rendimiento y la escalabilidad en el desarrollo de software. La arquitectura propuesta respalda de manera sólida los objetivos estratégicos de la empresa al ofrecer una infraestructura automatizada, segura y escalable para llevar a cabo pruebas de carga distribuidas. Los beneficios tanto económicos como de seguridad hacen de esta solución una elección altamente recomendable para aquellas empresas que buscan mejorar la calidad y el rendimiento de sus aplicaciones.

En conclusión, la adopción de AYGO Pruebas de Carga Distribuida en AWS no solo conduce a mejoras tangibles en la capacidad de respuesta y rendimiento de las aplicaciones, sino que también consolida la posición competitiva de la empresa en un mercado de desarrollo de software dinámico. Esta estrategia global no solo satisface las necesidades actuales, sino que también prepara a la empresa para enfrentar con confianza los desafíos tecnológicos futuros.