

# Api Gateway: Kong vs. AWS

Bravo Peña  
Darlyn  
UAGRM  
Santa Cruz, Bolivia  
bpdarlyn@gmail.com

Torrejón Mendez  
Joel Gabriel  
UAGRM  
Santa Cruz, Bolivia  
joel.torreon.mendez@gmail.com

Valle Tamayo  
Brandon Jason  
UAGRM  
Santa Cruz, Bolivia  
bjvtamayo78@gmail.com



## API Gateway OVERVIEW

Figure 1. Seattle Mariners at Spring Training, 2010.

### Abstract

A clear and well-documented  $\text{\LaTeX}$  document is presented as an article formatted for publication by ACM in a conference proceedings or journal publication. Based on the “acmart” document class, this article presents and explains many of the common variations, as well as many of the formatting

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org). *Cloud computing e infraestructura para Big Data*, Agosto 31-08, 2024, Santa Cruz, Bolivia

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-XXXX-X/18/06  
<https://doi.org/XXXXXXX.XXXXXXX>

elements an author may use in the preparation of the documentation of their work.

**CCS Concepts:** • **Do Not Use This Code → Generate the Correct Terms for Your Paper;** *Generate the Correct Terms for Your Paper;* Generate the Correct Terms for Your Paper; Generate the Correct Terms for Your Paper.

**Keywords:** Do, Not, Us, This, Code, Put, the, Correct, Terms, for, Your, Paper

### ACM Reference Format:

Bravo Peña, Darlyn, Torrejón Mendez, Joel Gabriel, Valle Tamayo, and Brandon Jason. 2018. Api Gateway: Kong vs. AWS. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Cloud computing e infraestructura para Big Data)*. ACM, New York, NY, USA, 5 pages. <https://doi.org/XXXXXXX.XXXXXXX>

## 1 Introducción

El propósito de este artículo es de hacer una investigación sobre las opciones de software Api Gateway, a efectos de este artículo elegimos dos opciones: Kong y AWS, para determinar cuál es la mejor opción para un proyecto de desarrollo de software. Se hará un análisis comparativo de las características de ambos productos, sus pros y contras, y se determinará cuál es la mejor opción para un proyecto de desarrollo de software. Se espera que este artículo sea útil para desarrolladores de software que estén considerando utilizar un Api Gateway.

## 2 Api Gateway

Antes de hablar de las diferencias entre Kong y AWS API Gateway, es importante entender qué es un API Gateway. Un API Gateway es un servidor que actúa como un punto de entrada para una API. Es similar a un proxy inverso, pero con funcionalidades adicionales. Un API Gateway puede realizar tareas como autenticación, autorización, enrutamiento, transformación de datos, monitoreo y administración de versiones. También puede proporcionar una capa de seguridad adicional al ocultar la estructura de la red interna de una organización. [1]

### 2.1 Funciones

La tecnología API Gateway ofrece una serie de ventajas, como la gestión eficiente de las solicitudes entrantes, que las enruta fácilmente a los servicios backend pertinentes. Además, puede traducir automáticamente los protocolos para que los clientes puedan interactuar con el servicio sin esfuerzo. [2]

**2.1.1 Baja latencia :** Al distribuir las solicitudes entrantes y descargar tareas comunes como la terminación SSL y el almacenamiento en caché, las puertas de enlace de API optimizan el enrutamiento del tráfico y el equilibrio de carga entre los servicios de backend para garantizar un rendimiento y una utilización de recursos óptimos. De este modo, las puertas de enlace de API minimizan la carga del servidor y el uso del ancho de banda, lo que reduce la necesidad de capacidad adicional del servidor y del ancho de banda de la red y mejora la experiencia del usuario.

**2.1.2 Gestión del tráfico :** Las puertas de enlace API limitan y gestionan el tráfico a través de diversos mecanismos diseñados para controlar la velocidad y el volumen de las solicitudes entrantes y garantizar un rendimiento y una utilización de recursos óptimos.

- Las políticas de limitación de velocidad
- Las políticas de limitación de solicitudes
- Las políticas de control de concurrencia
- Las políticas de interrupción de circuitos
- El equilibrio de carga dinámico

**2.1.3 Aprovechamiento de la infraestructura de red global :** Las puertas de enlace API pueden escalar dinámicamente los recursos de infraestructura en respuesta a los cambios en los patrones de tráfico y las demandas de carga de trabajo. De esta manera, las puertas de enlace API ayudan a las empresas a optimizar el uso de los recursos y minimizar los costos de infraestructura, lo que garantiza que solo paguen por los recursos que realmente utilizan.

**2.1.4 Rentabilidad.** Las puertas de enlace de API desempeñan un papel en la gestión de la rentabilidad de la entrega de aplicaciones y la integración de API al proporcionar una plataforma centralizada para gestionar el tráfico de API, aplicar políticas de seguridad, implementar reglas de gestión del tráfico y facilitar la integración con los servicios de backend. Las puertas de enlace de API también permiten el consumo escalonado de servicios para mantener la rentabilidad. Los diferentes tipos de API pueden afectar la rentabilidad de una aplicación de varias maneras.

- Flexibilidad
- Infraestructura
- Escalabilidad

## 3 Kong

Kong Gateway es una pasarela de API nativa en la nube ligera, rápida y flexible y un proxy inverso que le permite gestionar, configurar y dirigir las solicitudes a sus API.

### 3.1 Arquitectura descentralizada

Kong está construido sobre una arquitectura descentralizada, lo que permite desplegar y gestionar API en múltiples entornos, ya sea en la nube, on-premise o en una configuración híbrida. Esto facilita la escalabilidad y la resistencia del sistema.

**3.1.1 Automatización de Flujos de Trabajo y GitOps Modernos.** Kong se integra con prácticas modernas como GitOps, que permiten la automatización de flujos de trabajo a través del control de versiones de la infraestructura y las configuraciones. Esto asegura que las implementaciones sean consistentes y rastreables, lo cual es esencial en entornos descentralizados.

**3.1.2 Ecosistema de Desarrolladores de API.** Al descentralizar las aplicaciones y servicios, Kong Gateway facilita la creación de un ecosistema vibrante para los desarrolladores de API. Los equipos pueden trabajar de manera independiente en diferentes servicios, lo que acelera el desarrollo y la implementación de nuevas funcionalidades.

**3.1.3 Identificación Proactiva de Anomalías y Amenazas.** La arquitectura descentralizada, junto con las capacidades avanzadas de Kong, permite una identificación proactiva de anomalías y amenazas relacionadas con las API. Esto

es esencial para mantener la seguridad y el rendimiento en un entorno distribuido.

**3.1.4 Gobernanza y Seguridad de APIs.** Kong proporciona herramientas para la seguridad y la gobernanza de las API, mejorando la visibilidad y el control en toda la organización. Esto es crucial en arquitecturas descentralizadas, donde es necesario mantener un alto nivel de seguridad y cumplimiento normativo a través de múltiples servicios y entornos.

### 3.2 Plugins Personalizables

Kong ofrece una amplia variedad de plugins que pueden ser utilizados para añadir funcionalidades como autenticación, limitación de tasa, caché, entre otros. Además, permite a los usuarios crear sus propios plugins personalizados para satisfacer necesidades específicas.

### 3.3 Seguridad

Kong proporciona diversas herramientas para asegurar las API, incluyendo autenticación JWT (JSON Web Token), OAuth2, mTLS (mutual TLS), y cifrado de tráfico. La capacidad de implementar políticas de seguridad a través de sus plugins hace que Kong sea una opción confiable para la protección de API.

### 3.4 Monitoreo y Observabilidad

Kong ofrece integración con herramientas de monitoreo como Prometheus, Grafana, y otras soluciones de observabilidad para rastrear el rendimiento de las API, errores, y métricas clave. Esto permite una gestión proactiva y la resolución rápida de problemas.

Konnect Advanced Analytics

- **Visibilidad centralizada:** Proporciona una visión integral y centralizada de todo el panorama de tus API para todas las APIs, servicios y planos de datos.
- **Analítica contextual de API:** Konnect Advanced Analytics ofrece información sobre cada solicitud de API, incluyendo las rutas específicas, los consumidores involucrados, y los servicios accedidos.
- **Insights de datos democratizados:** Konnect empodera tanto a los equipos de negocio como a los de plataforma para generar informes para cualquier servicio, ruta, o consumidor, basados en sus requisitos específicos.
- **El tiempo más rápido para obtener insights:** Proporciona a los equipos de aplicaciones y plataformas métricas críticas de API para cada servicio en menos de un segundo, reduciendo así el tiempo de resolución.
- **Reducción del costo de propiedad:** Advanced Analytics es una solución de analítica llave en mano que elimina la necesidad de construir, mantener o integrarse con productos de terceros.

### Servicios y Rutas

Kong Gateway utiliza un modelo de objetos para definir políticas de gestión de tráfico. Los objetos clave, como servicios y rutas, se configuran de manera coordinada para establecer el flujo de solicitudes y respuestas en el sistema. Las solicitudes se enrutan a los servicios a través de las rutas definidas, mientras que las respuestas siguen el camino inverso, tal como se describe en la siguiente imagen.

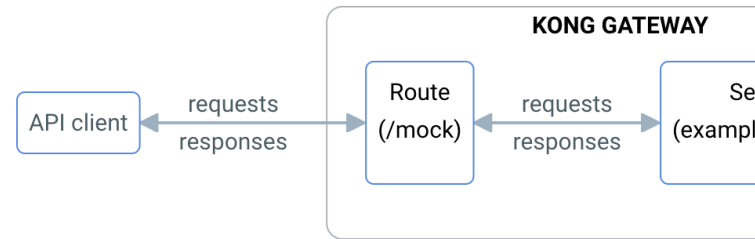


Figure 2. Kong Gateway - Rutas y servicios

**Servicio.** En Kong Gateway, un servicio es una abstracción de una aplicación upstream existente. Los servicios pueden almacenar colecciones de objetos como configuraciones de plugins y políticas, y pueden asociarse con rutas.

Al definir un servicio, se proporciona un nombre y la información de conexión a la aplicación upstream. Estos detalles de conexión pueden especificarse en un solo campo URL o separadamente en los campos de protocolo, host, puerto y ruta.

Los servicios tienen una relación de uno a muchos con las aplicaciones upstream, lo que permite crear comportamientos avanzados de gestión de tráfico.

**Ruta.** Una ruta es un camino hacia un recurso dentro de una aplicación upstream. Las rutas se añaden a los servicios para permitir el acceso a la aplicación subyacente. En Kong Gateway, las rutas suelen mapearse a endpoints expuestos a través de la aplicación Kong Gateway. Además, las rutas pueden definir reglas que relacionan solicitudes con servicios asociados, lo que permite que una ruta pueda referenciar múltiples endpoints. Una ruta básica debe tener un nombre, una o varias rutas, y referenciar a un servicio existente.

## 4 TBD

Modifying the template — including but not limited to: adjusting margins, typeface sizes, line spacing, paragraph and list definitions, and the use of the \vspace command to manually adjust the vertical spacing between elements of your work — is not allowed.

**Your document will be returned to you for revision if modifications are discovered.**

## 5 Benchmark

Authors of any work published by ACM will need to complete a rights form. Depending on the kind of work, and the rights management choice made by the author, this may be copyright transfer, permission, license, or an OA (open access) agreement.

Regardless of the rights management choice, the author will receive a copy of the completed rights form once it has been submitted. This form contains  $\LaTeX$  commands that must be copied into the source document. When the document source is compiled, these commands and their parameters add formatted text to several areas of the final document:

- the “ACM Reference Format” text on the first page.
- the “rights management” text on the first page.
- the conference information in the page header(s).

Rights information is unique to the work; if you are preparing several works for an event, make sure to use the correct set of commands with each of the works.

The ACM Reference Format text is required for all articles over one page in length, and is optional for one-page articles (abstracts).

## 6 Conclusión

The “figure” environment should be used for figures. One or more images can be placed within a figure. If your figure contains third-party material, you must clearly identify it as such, as shown in the example below.



**Figure 3.** 1907 Franklin Model D roadster. Photograph by Harris & Ewing, Inc. [Public domain], via Wikimedia Commons. (<https://goo.gl/VLCRBB>).

Your figures should contain a caption which describes the figure to the reader.

Figure captions are placed *below* the figure.

Every figure should also have a figure description unless it is purely decorative. These descriptions convey what’s in

the image to someone who cannot see it. They are also used by search engine crawlers for indexing images, and when images cannot be loaded.

A figure description must be unformatted plain text less than 2000 characters long (including spaces). **Figure descriptions should not repeat the figure caption – their purpose is to capture important information that is not already provided in the caption or the main text of the paper.** For figures that convey important and complex new information, a short text description may not be adequate. More complex alternative descriptions can be placed in an appendix and referenced in a short figure description. For example, provide a data table capturing the information in a bar chart, or a structured list representing a graph. For additional information regarding how best to write figure descriptions and why doing this is so important, please see <https://www.acm.org/publications/taps/describing-figures/>.

### 6.1 The “Teaser Figure”

A “teaser figure” is an image, or set of images in one figure, that are placed after all author and affiliation information, and before the body of the article, spanning the page. If you wish to have such a figure in your article, place the command immediately before the `\maketitle` command:

```
\begin{teaserfigure}
\includegraphics[width=\textwidth]{sampleteaser}
\caption{figure caption}
\Description{figure description}
\end{teaserfigure}
```

## Acknowledgments

To Robert, for the bagels and explaining CMYK and color spaces.

## References

- [1] Red Hat. 2024. *What does an API gateway do?* Red Hat. <https://www.redhat.com/en/topics/api/what-does-an-api-gateway-do>
- [2] Kong. 2024. *What is an API Gateway?* Kong. <https://konghq.com/blog/learning-center/what-is-an-api-gateway>

## A Research Methods

### A.1 Part One

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi malesuada, quam in pulvinar varius, metus nunc fermentum urna, id sollicitudin purus odio sit amet enim. Aliquam ullamcorper eu ipsum vel mollis. Curabitur quis dictum nisl. Phasellus vel semper risus, et lacinia dolor. Integer ultricies commodo sem nec semper.

### A.2 Part Two

Etiam commodo feugiat nisl pulvinar pellentesque. Etiam auctor sodales ligula, non varius nibh pulvinar semper. Suspendisse nec lectus non ipsum convallis congue hendrerit

vitae sapien. Donec at laoreet eros. Vivamus non purus placerat, scelerisque diam eu, cursus ante. Etiam aliquam tortor auctor efficitur mattis.

## **B Online Resources**

Nam id fermentum dui. Suspendisse sagittis tortor a nulla mollis, in pulvinar ex pretium. Sed interdum orci quis metus euismod, et sagittis enim maximus. Vestibulum gravida

massa ut felis suscipit congue. Quisque mattis elit a risus ultrices commodo venenatis eget dui. Etiam sagittis eleifend elementum.

Nam interdum magna at lectus dignissim, ac dignissim lorem rhoncus. Maecenas eu arcu ac neque placerat aliquam. Nunc pulvinar massa et mattis lacinia.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009