

EBOOK | MARCH 2023

API Gateways:

Productivity, Resilience, and Security for Next-Generation
Cloud Applications



Contents

- 3** Introduction
- 5** Microservices Paradigm Shift:
New Problems and New Solutions
- 7** A Cloud-Native Modern Solution:
Developer Portal and API Gateway
- 11** Conclusion

INTRODUCTION

API gateways have become increasingly important in this era of cloud-based microservices architecture. Today's fast-paced business environment demands that companies continuously look for ways to improve their digital capabilities and stay ahead of the curve by delivering innovative and high-quality applications to their customers. API gateways play a crucial role in allowing companies to achieve their goals.

The popularity of microservices architecture has led to a need for more efficient ways to build, connect, and deploy smaller and more focused services. In a microservices architecture, each service is designed to perform a specific task

and can be deployed, updated, and managed independently, allowing for greater flexibility and scalability. However, as the number of microservices deployed increases, so too does the complexity of their management. This is where API gateways come into play.

API gateways are the entry point for incoming API calls. They handle traffic management and security, allowing developers to focus on building the core functionality of their microservices without having to worry about these sometimes tedious operational tasks. Additionally, API gateways provide a central location for monitoring, logging, and debugging microservices, making it easier to diagnose problems and maintain the system's overall health.

Netflix has successfully implemented API gateways in its microservices architecture, handling [over 1 billion API requests per day](#). Their API gateway is a vital component in Netflix's ability to provide a seamless, high-quality streaming experience for their customers. Another prime example is eBay, which uses API gateways to manage traffic flow to its various microservices, ensuring that the platform remains fast, reliable, and secure.

In this eBook, we will delve deeper into the role of API gateways and their significance in next-generation cloud applications, including technical details, benefits, and real-life examples.

MICROSERVICES PARADIGM SHIFT: NEW PROBLEMS AND NEW SOLUTIONS

The rise of microservices architecture has brought about a significant shift in how software is developed and deployed, and as such, has presented new technical and organizational challenges. In order to manage microservices effectively, businesses need to adopt new strategies and technologies suited to today's microservice-dominated software environment and its specific challenges. These new challenges can be broken down into four categories:

- 1. Potentially unwieldy bureaucratic processes:** This problem is caused by the need for independent deployment, management, and monitoring of each component, which in turn increases operational overhead.
- 2. Software-based tooling:** Microservices infrastructures are often more complex, and have more moving parts, than monolithic architectures. This contributes to management challenges.
- 3. The need for high levels of scalability:** Individual microservices need to handle high levels of traffic, and the overall system needs to take an increase in demand.
- 4. Evolving best practices:** It can be challenging to keep abreast of how to manage and deploy microservices as effectively as possible.

These challenges are a potential minefield, posing a multifaceted challenge to the management of microservices infrastructures. There is a real risk that management overhead may spiral out of control, which could jeopardize the quality of the entire application. A managed API gateway solution can assist in resolving all of these problems in a single location by providing a unified interface for accessing microservices.

A managed API gateway solution's developer self-service portal provides easy access to best practices and resources, making it easier to build, connect, and deploy microservices applications. By providing a centralized platform for managing and deploying microservices, the need for manual intervention is reduced. For example, an API gateway solution can provide automatic load balancing and traffic management, ensuring that

microservices are deployed and managed in a scalable manner.

To summarize, the shift to microservices architecture presents new challenges that businesses must address in order to prosper. A managed API gateway solution with a developer self-service portal can address these challenges in one fell swoop, and provide a centralized platform for managing and deploying microservices applications. As a result, improvements in next-generation cloud applications' productivity, resilience, and security can be realized.

A CLOUD-NATIVE MODERN SOLUTION: DEVELOPER PORTAL AND API GATEWAY

API gateways and developer portals are effective, cloud-native solutions to the challenges of API management microservice applications. They offer a way to manage and control traffic to microservices applications, enforce security, and provide extensibility.

TRAFFIC MANAGEMENT

API gateways are a crucial component in managing traffic flow to microservices applications. They act as a single entry point for all client requests and help regulate traffic, enhancing performance and security. According to a 2022 [report by Forrester on API development and consumption](#), the use of API gateways for

traffic management is on the rise; 59% of the respondents understand that API management is beneficial for increasing their traffic and revenue.

API gateways offer fine-grained control over traffic, allowing for optimized management of the microservices infrastructure. For example, they can allow developers to ensure that only valid requests are processed and sent to the right microservices. Let's assume two versions of the same application are deployed: v1 and v2. Old and new clients (with v1 and v2 versions) are connecting to the server side simultaneously. API gateways direct the versioned client requests to the appropriate microservices, thus protecting the continuity of the whole stack.

Modern deployment strategies can be implemented via API gateways—such as blue/green, canary, or A/B testing—with minimal impact on the rest of the system. In addition,

API gateways provide valuable insights into traffic patterns, helping organizations to make informed decisions about their microservices architecture. For instance, the traffic patterns can uncover underperforming microservices and identify any microservices that are creating a bottleneck in the application stack. With features such as fine-grained control over traffic, improved performance, and valuable insights into traffic patterns, API gateways enable organizations to optimize their microservices for a seamless and reliable user experience.

SECURITY

A major challenge when shifting to a microservices architecture is ensuring security across all application stacks. API gateways play a crucial role in enforcing security measures in a microservices environment, offering features such as authentication, authorization, and encryption.

For instance, the API gateway can act as a single entry point for all microservices requests and enforce strict security policies such as OAuth 2.0, JWT, or API keys. The API gateway can also be configured to implement rate limiting and request throttling, preventing a malicious user from overwhelming the system with too many requests. Another use case is the implementation of traffic encryption between microservices and the API gateway, which allows data to be transmitted securely over public networks. This helps to prevent eavesdropping and tampering with sensitive data.

The healthcare industry makes powerful use of API gateways to ensure security of sensitive patient information. With the increased use of electronic medical records, this is naturally a top priority for the industry. API gateways can be used to ensure that only authorized users have access to patient information, and that all data being transmitted is encrypted.

Security is a critical aspect of microservices architecture, and API gateways play an essential role in ensuring that microservices applications are able to protect sensitive data and maintain their users' privacy, thus securing both brand reputation and customer trust.

EXTENSIBILITY

API gateways can add functionality and extend the capabilities of microservices applications since they act as a single entry point for all external requests to a microservices application. The API gateway can act as an intermediary between the external client and the microservices by providing additional functionality and improving the overall user experience. In other words, companies can focus on their core business competencies while removing the management burden. According to a Forrester [report on API security from 2022](#), 78% of respondents say the adoption and extension of APIs is important for their company to stay

competitive in the market.

Let's consider the case of a large e-commerce company with a complex architecture including numerous microservices. Each microservice handles a different aspect of the customer journey, such as browsing products, adding items to a shopping cart, and processing payments. The API gateway adds enhanced functionality, such as personalization, to the customer experience. For example, it can collect data on the customer's browsing and purchasing habits, and then use that information to suggest products that might interest the customer. This personalized experience will likely lead to higher revenue and sales and better customer engagement—desirable outcomes for any e-commerce platform.

By acting as an intermediary between the external client and the microservices, API gateways can add functionality, improve user experience, and increase the overall effectiveness of the microservices stack.

DEVELOPER PORTAL AND PROCESS

The process of building and deploying microservices applications is simplified through the use of API gateways. The developer portal manages the lifecycle of microservices from development to deployment, providing self-service tools and resources for developers. The developer portal helps to streamline the development process and to automate specific tasks, such as testing and deployment. It functions as a central hub for developers to collaborate, share information, and create a centralized view of the microservices landscape, making it easier to understand relationships between the various microservices in use.

Puppet's DevOps Research and Assessment group (DORA) indicates in their [State of DevOps Report 2020](#) that organizations that implemented

self-service portals for their developers saw a significant improvement in collaboration and communication among teams. The result was increased efficiency and productivity.

A developer portal and API gateway solution can significantly improve the microservice application development process by providing self-service tools and resources for developers, automating manual tasks, and simplifying the lifecycle of microservices. Correspondingly, developers often report increased productivity, efficiency, and collaboration.

CONCLUSION

API gateways and developer portals are crucial to the management and security of microservices architectures, which dominate next-generation cloud applications. They help to improve performance and scalability, enforce security measures, and simplify the process of building and deploying microservices. Moreover, API gateways and developer portals provide an additional layer of security by enforcing access control, authentication, and authorization policies, helping to ensure that only authorized users and services have access to the microservices, thus reducing the risk of security breaches and malicious attacks.

In addition to security, API gateways and

developer portals help organizations to monitor and manage their microservices' health. With real-time monitoring, organizations can detect and resolve issues before they escalate, thus reducing downtime so that microservices are always available to the end users. Rate limiting is also facilitated by API gateways, helping to prevent the microservices' overload and ensuring that they perform optimally.

ENTER SOLO.IO

Using a modern API gateway and developer portal solution—like [Solo.io](https://solo.io)—offers enterprises significant benefits in the realms of productivity, resilience, and security. With its cloud-native and comprehensive approach, Solo.io aims to combat the challenges of distributed microservices architectures while saving time, money, and effort. With its intuitive interface, organizations can quickly and easily set up and manage their

microservices. Solo.io also offers a range of customization options, ensuring that every organization's specific requirements are met through a tailor-made solution.

Solo.io's [developer portal](#) provides a centralized platform for developers to access resources and tools for building and deploying microservices. With self-service tools, developers can quickly and easily create, test, and deploy microservices without relying on IT or operations teams. This speeds up the development process, enabling organizations to launch new microservices faster and with greater agility.

API gateways and developer portals are crucial in managing and securing the microservices architecture of next-generation cloud applications. With its comprehensive range of features, Solo.io provides a top solution for organizations looking to optimize their microservices architecture. Whether it's improving performance, scalability, security, or simplifying the process of building and deploying microservices, Solo.io provides comprehensive service coverage. [Take a free trial of Solo.io today!](#)



About Solo.io

Solo.io, the leading application networking company, delivers a service mesh and API platform for Kubernetes, zero trust, and microservices. The three components of the Gloo Platform – Gloo Gateway, Gloo Mesh and Gloo Network – enable enterprise companies to rapidly adopt microservice applications as part of their cloud journey and digital transformation. Solo delivers open source solutions, and is a community leader in building the technologies of the future.

✉ contact@solo.io

🌐 www.solo.io