API Security Best Practices: Key Considerations for API Protection

Successful API security requires vigilance on multiple fronts.

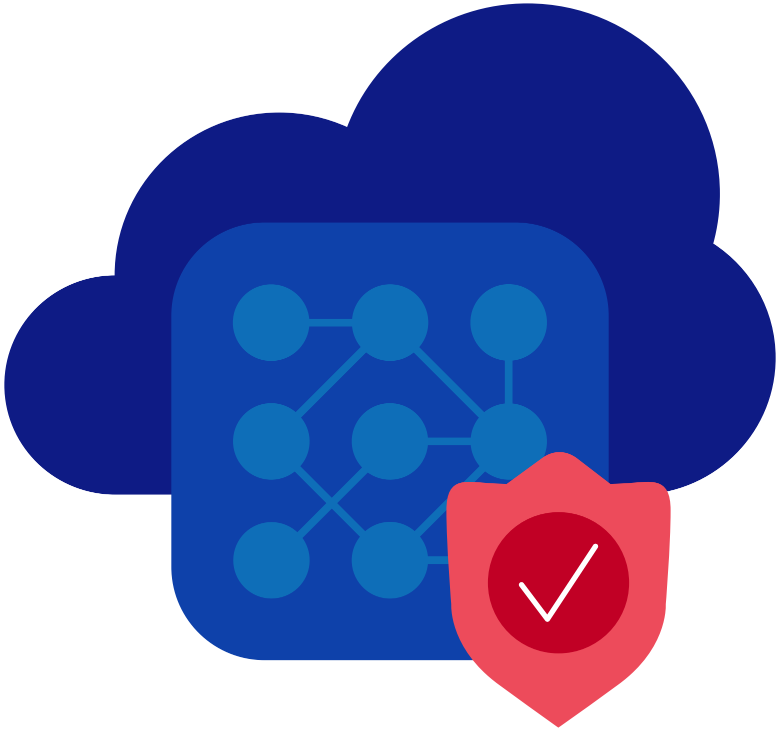
Contents

Introduction

Application Programming Interfaces, or APIs, were developed to streamline the way digital ecosystems communicate and interact with each other. At their core, APIs abstract the complexity of connecting multiple disparate systems. This allows developers to integrate third-party content or services quickly and easily into their applications, automate mundane tasks, and increase convenience for online shopping, one-stop travel planning, and other digital miracles.

In our increasingly connected world, APIs have evolved from vehicles that allow developers to quickly implement new functionality without re-inventing the wheel to become the essential fabric of the digital economy—from mobile apps to modern digital experiences that interconnect across data centers, private/public clouds, and the edge. As applications constantly evolve, they increasingly depend on third-party APIs to provide the bridge to modernization. APIs now serve as the cornerstone for more complete, convenient, and powerful digital experiences, as opposed to traditional apps built on classic 3-tier web stacks in the data center.

The emphasis on APIs as a fundamental building block for app development has resulted in a cascade of infrastructure and deployment scenarios and, in particular, has dramatically decentralized architecture. However, the growing reliance on APIs has also greatly expanded the risk surface for data breach, abuse, and attacks that bypass access controls. Many existing security tools struggle to detect and mitigate these risks, especially as API proliferation sprawls infrastructure across hybrid and multi-cloud environments.



The Challenge of Securing APIs in Decentralized and Distributed Architectures

Because of their ability to facilitate connections among applications and easily integrate new content and services, APIs have become a foundation of today's digital economy. APIs are one of the quickest and most efficient ways of adding features to existing digital offerings. APIs make it easy to add new services, data, and functionality—such as digital map services or online retail shopping carts—to existing applications. As a result, APIs are indispensable for modernizing legacy apps and improving the speed and efficiency ofept application development.

As fast, responsive, and frictionless digital experiences become a defining element and a differentiator for today's organizations, APIs represent a powerful and pervasive way for businesses to connect with their customers, partners, suppliers, employees, and other users.

APIs also offer stable and efficient ways to provide data, content, and functionality to a range of development platforms or frameworks, where mobile apps, single page apps, and microservices reign. APIs enable improved ease-of-use, performance, and reliability across ecosystems. This allows stable and secure collaboration, and innovation with customers and partners by delivering a better user experience and fast access to valuable data and information. The de-coupling of monolithic features into microservices allows app teams, and the infrastructure teams that support them, to scale independently while still maintaining a unified customer front.

Think of a travel app that offers not just airline reservations but also rental car services, online travel guides, weather information, and real-time flight status. Each of these services and content resources is delivered through third-party APIs. Developers can leverage these APIs for more efficient app development and faster time to market while also creating a differentiated customer experience by offering value-added services without the need for custom development. If one of the dependencies change, the entire application does not necessarily need to be repackaged and redeployed. However, this has real implications for security and risk teams that are tasked with protecting the entire digital estate.

With APIs serving as the interconnection, applications have moved toward an increasingly distributed and decentralized model. Because APIs are typically designed to be exposed externally, they can be a doorway to sensitive data that the business needs to protect.

Modern API delivery designs are innovative and fluid, and application components may be scattered across multiple environments—namely the data center, across clouds, and at the edge—making it difficult to manage and secure all elements consistently. API components may be housed within multiple public cloud providers to take advantage of specific cloud-native tools and may exist in a mix of data centers, private clouds, and public cloud environments—creating a hybrid IT environment.[1] In addition, containers and serverless systems create ephemeral components that must be secured. In each of these decentralized models, APIs distributed across heterogenous infrastructures are outside the realm of centralized security controls, increasing the attack surface and potential entry points for malicious actors.

APIs are indispensable for modernizing legacy apps and improving the speed and efficiency of application development.

APIs Are Subject to the Same Attacks as Web Apps

API security incidents have been responsible for many high-profile data breaches, as APIs are susceptible to many of the same attacks that target web applications, including vulnerability exploits, abuse from bots, and attacks that bypass weak authentication/authorization controls.[2] These kinds of attacks can result in data breaches, in which hackers gain access to a sensitive data and extract confidential information without authorization, or account takeover and fraud if critical business logic is compromised.

APIs are also vulnerable to attacks from bots and malicious automation. Bots are pervasive on the Internet—in fact, according to an F5 Labs analysis, an increasing proportion of automated attacks are coming via the Mobile channel over time, and as more industries adopt modern application architectures and move towards APIs, this trend is expected to continue.[3]

Bots are software programs that imitate human behavior on the Internet by carrying out automated, repetitive, pre-defined tasks. They can perform useful functions, such as automating customer service, simulating human communication on social networks, or indexing search engines. Bots can also be directed to perform malicious tasks, including theft of financial data and personal information through credential stuffing, and other forms of fraud and abuse stemming from account takeover (ATO).

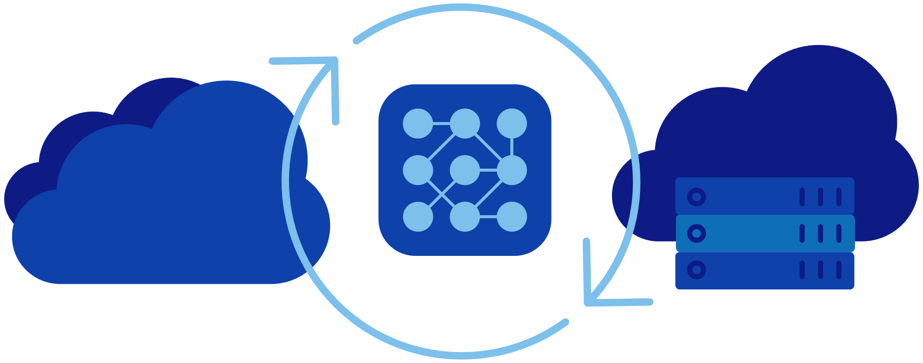
Bots can also inflict high-volume distributed denial of service (DDoS) attacks that overwhelm legitimate web services, networks, or APIs by overloading systems with fake network traffic to cause a disruption in service. Automated DDoS attacks consume server and network resources, cripple web services that companies rely on for business, or completely overburden a company's network—taking down customer facing services. APIs, in particular, are often the conduit to critical business logic, which can be abused when attackers flood endpoints with requests or send a targeted query that results in a resource intensive task. Both scenarios can lead to performance degradation, customer frustration, and transaction abandonment.

Therefore, risk management leaders need to be concerned with bot-driven attacks that lead to compromise and data breach as well as those that impact uptime and reliability, for both legacy web apps and modern API fabrics. Multi-vector DDoS attacks require multi-layer defenses for all critical infrastructure—the network, servers, and the application interfaces themselves—which are increasingly implemented through APIs.

Existing Security Controls May Not Protect APIs

Traditional security controls, such as basic web app firewalls (WAFs) and security information and event management (SIEM) systems, are not sufficient to identify and prevent attacks on APIs, in part because of the high amount of machine-to-machine, or API-to-API traffic. Attacks and breaches can seem like normal app behavior on the surface, but behind the scenes APIs can be exploited and abused—allowing attackers to elude detection until it's too late.

East-west API traffic—network communications between servers and applications within the data center—often bypasses centralized controls. This is unlike north-south network traffic, which refers to client-server communications that enter and exit the perimeter of the data center and are typically protected by various types of firewalls, intrusion detection systems, DDoS mitigation devices, and other layered protections at the data center entrance. However, with flows of east-west traffic, there is no single-point-of-detection to monitor for malicious activity—even though large organizations may have massive volumes of east-west traffic flowing between data centers and multiple private and public clouds. APIs are a primary driver for the de-centralization of architecture and deployment of digital ecosystems across hybrid and multi-cloud environments.



Input validation, sometimes called data validation, is fundamental to application security. Input validation is a risk control method for testing any input supplied by a user to prevent improperly formed data from executing business logic. Because APIs are designed for machine-to-machine communication and data exchange, however, they may represent a direct route to an organization's most sensitive data. As a result, input validation for APIs may not be as robust—or as tested—as that on user-facing web forms.

APIs are a primary driver for the de-centralization of architecture and deployment of digital ecosystems across hybrid and multi-cloud environments.

Multi-cloud architectures, and those that use multiple data centers, can complicate API security and contribute to API sprawl, which occurs when APIs become widely distributed without a comprehensive governance strategy. Such architectures are difficult to secure, in part because they lack predictable and trusted observability across environments. Inconsistent API attack detection and policy enforcement—common in many such architectures due to the complexity of maintaining application deployments that span multiple clouds and data centers—can create unknown risk and dangerous security gaps. For example, the potential for misconfiguration across clouds, and the presence of unmaintained or even rogue API endpoints—often referred to as shadow and zombie APIs.

Modern Application Lifecycles Introduce Unintended Risk

Many organizations use a continuous improvement and continuous delivery (CI/CD) development process, which allows them to deploy code changes and new versions of software quickly and reliably. Yet, any software change may introduce or increase risk. As application development teams continue to use CI/CD pipelines for faster innovation, a growing number of API calls may end up hidden deep within business logic, making them extremely hard to discover and identify, and increasing the risk of attack through third-party APIs.

Using third-party APIs can save an organization a lot of time, simplifying the development process and providing easy access to features and services created by other companies. Yet, while organizations can control the APIs they develop, using third-party APIs can provide an open window to their house and give cybercriminals easy access to their data and applications. Seemingly innocuous functionality, such as an open-source logging library, can be the source of critical weaknesses and vulnerabilities.[4] When developers deploy public APIs, such as those made available through open-source libraries, they often bypass established security processes and procedures.

Today, organizations face a continuously changing risk landscape, with new threat vectors emerging all the time. A good example is third-party aggregators, like the companies that offer consumers and merchants alternative payment options or other financial services. Third-party aggregators provide services that consumers and merchants value, but they also introduce a new level of risk that could compromise a consumer's personal information, leading to fraud or identity theft, and expose a merchant's system to possible attack.

When it comes to security, CI/CD pipelines are also a mixed blessing for many organizations. As companies continue to accelerate their development and deployment cycles through the use of automated CI/CD pipelines, many have security policies that still use manual processes for the discovery and assessment of software components. That approach is impractical at best, and it may increase an organization's risk. Meanwhile, reactive security—waiting to act until a threat arises within a system—can impede the software development and release cycle, and slow time to market. As a result, many organizations sacrifice security in favor of speed, which amplifies their risk of attack.

APIs Introduce More Risk by Default

APIs are the foundation of many modern applications and a key component of innovation and digital transformation at multiple levels. They can enable disparate systems to work together seamlessly, speed time to market, and enhance customer experiences by giving developers easy access to third-party features, functionality, and services. At the same time, APIs inevitably introduce new, hard-to-manage risks by opening multiple windows into an organization's house.

APIs inevitably introduce new, hard-to-manage risks by opening multiple windows into an organization's house.

Still, the use of APIs continues to skyrocket—the number of APIs is projected to reach a billion by 2031—and the attack surface keeps expanding.[5] As organizations continue to use APIs to innovate and modernize their app portfolios, they also increase the number of endpoints and parameters, and raise the risk of attack.

Fundamentally, APIs provide more information for attackers to glean during reconnaissance. The availability and information contained within APIs can provide insight into system internals, which can help hackers create targeted attacks. This is especially true with open-source vulnerabilities like log4j2, where attackers immediately started scanning the Internet for vulnerable hosts as soon as an exploit kit was published. The complexity of software supply chains, and the difficulty of evaluating their risks, underscores the need for more alignment across application and security teams.[6]

Key Considerations for API Security

There are several things for organizations to consider as they prepare to strengthen their API security:

Continuously monitor and protect API endpoints to identify changing app integrations, detect vulnerable components, and mitigate attacks through third-party integrations using automation and machine learning.

Implement a positive security model by supporting OpenAPI specifications and API Swagger file imports, validating schema and enforcing protocol compliance, and automatically baselining normal traffic patterns and detecting anomalous behavior indicative of malicious users.

Embrace zero-trust and risk-based security principles by limiting available methods, inspecting payloads and preventing unauthorized data exposure, and implementing access control and risk-based authentication for objects and functions.

React to a changing application lifecycle by preventing security misconfiguration across heterogenous environments, mitigating abuse that can lead to account takeover and denial of service, and remediating threats consistently across clouds and architectures.

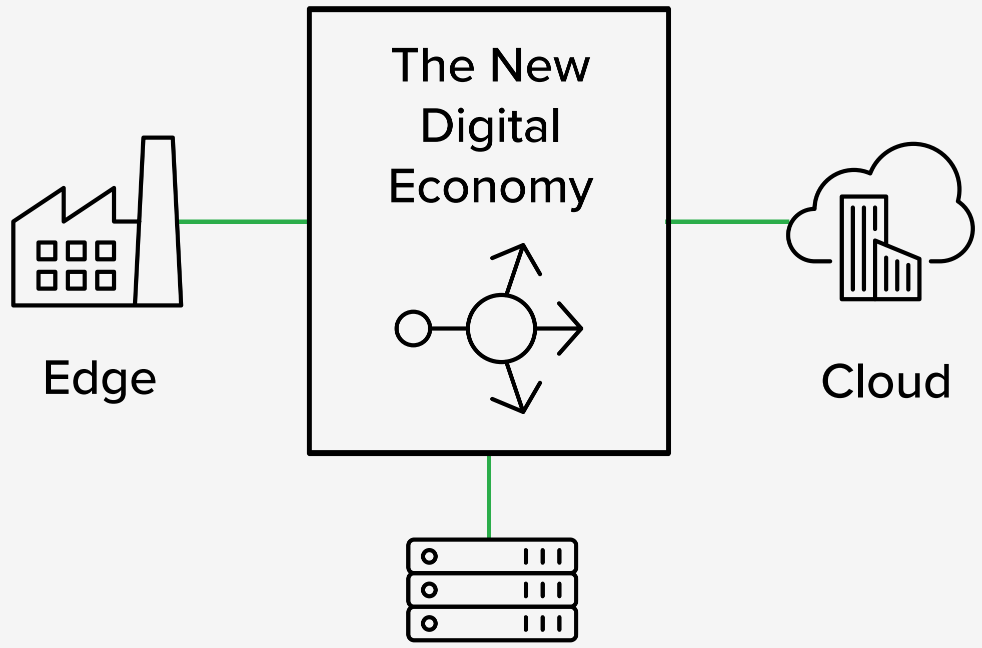


Figure 1: APIs are central to today's decentralized, distributed computing infrastructures.

The Bottom Line

APIs are pivotal to digital business yet are inherently more difficult to secure as compared to legacy architectures with more predictable use cases, such as custom three-tier web stacks in the data center. APIs facilitate a decentralized and distributed architecture with endless opportunities for third-party integration that fundamentally changes the calculus for security and risk teams.

Appendix

[1] Lori MacVittie, “What Hybrid IT Means for App and API Security,” F5 Office of the CTO (January 11, 2023) https://www.f5.com/company/blog/what-hybrid-it-means-for-app-and-api-security

[2] Lori MacVittie, "The Case for Integrated App and API Security Strategies”, F5 Office of the CTO (August 9, 2023) https://www.f5.com/company/blog/the-case-for-integrated-app-and-api-security-strategies

[3] Tafara Muwand, “Monthly Bot Stats Report: H1 2023”, F5 Labs (August 15, 2023) https://www.f5.com/labs/articles/threat-intelligence/monthly-bot-stats-report-h1-2023

[4] Andrew van der Stock, Brian Glas, Neil Smithline, Torsten Gigler, “The OWASP Top 10 for 2021: A New Wave of Risk,” OWASP (September 24, 2021) https://www.f5.com/resources/infographic/owasp-top-10-2021-infographic

[5] Rajesh Narayanan and Mike Wiley, “Continuous API Sprawl: Challenges and Opportunities in an API-Driven Economy,” (2021) https://www.f5.com/pdf/reports/f5-office-of-the-cto-report-continuous-api-sprawl.pdf

[6] Sander Vinberg, “Log4Shell: Rebooting (The Same Old) Security Principles In its Wake,” F5 Labs (December 17, 2021) https://www.f5.com/labs/articles/cisotociso/log4shell-rebooting-the-same-old-security-principles-in-its-wake

ABOUT F5

BRINGING A BETTER DIGITAL WORLD TO LIFE

F5 is a multi-cloud application services and security company committed to bringing a better digital world to life. F5 partners with the world's largest, most advanced organizations to secure and optimize apps and APIs anywhere—on premises, in the cloud, or at the edge. F5 enables organizations to provide exceptional, secure digital experiences for their customers and continuously stay ahead of threats. For more information, go to f5.com. (NASDAQ: FFIV).

Learn more at f5.com/solutions/web-app-and-api-protection