Google Cloud

The Importance of API Security

Why it’s time for CISOs to protect APIs with a defense-in-depth strategy

Authored by Anton Chuvakin, Sakshi Dalmia, Etienne De Burgh, Ashwath Desai, Ilya Kabanov, and Sachin Kalra

Office of the CISO

API security is more critical than ever

Now is the time for CISOs to prioritize defense in depth for API security

Application programming interfaces, or APIs, are driving the digital economy and opening up paths for new business opportunities. Companies use APIs to not only provide access to their own services, but to also integrate applications within an organization as well as with third-party services. Think of Google Maps Platform APIs delivering functionality that offers services based on location, or Google Pay APIs being integrated with the eCommerce websites of Google customers.

Now API traffic is dominating the internet. Case in point: Google Cloud's Apigee saw a 46% increase in API traffic when comparing Black Friday 2020 and 2021. This is because APIs are a gateway for services and data and can unlock the value of data that enables business growth.

At the same time, APIs represent a large, dynamically changing attack surface. Just one web or mobile application can rely on hundreds of APIs that provide data to the application that's executing in the user's browser or on a mobile device. Therefore, those hundreds of APIs expose data through external interfaces that need to be properly protected. Third-party APIs that are used ubiquitously also contribute to a growing API attack surface. For example, a stolen customer key for accessing a third-party API can be used by an attacker to consume the third-party service, resulting in excessive payment obligation for the customer.

Attack surfaces are expanding dramatically as the result of API proliferation.

This means that as a security professional, you need to leverage your organizational knowledge, security strengths, and technical know-how to protect APIs along their entire life cycle. You need to engage in API design and implementation, and then move to API development, deployment, change control, operations, and retirement. Since business logic forms part of the API attack surface, you must also ensure engagement with API business owners as part of the threat modeling process to determine likely abuse scenarios involving business logic.

This report outlines our recommendations for building a defense in depth for APIs that CISOs and their teams should establish to enable business growth in a safe and sustainable way.

The threat is very real

At Google, we've seen how the growing API footprint attracts threat actors of all sorts. In 2020, Apigee observed a 172% increase in abusive API traffic from 2019. In 2022, half of the 500 technology leaders surveyed in the United States reported that they experienced an API security incident in the past 12 months.

Attacks on APIs can be broadly split into five categories:

01 Data scraping
02 Denial of service (DoS)
03 Injections or malware
04 Account takeover (ATO)
05 Scalping and bots

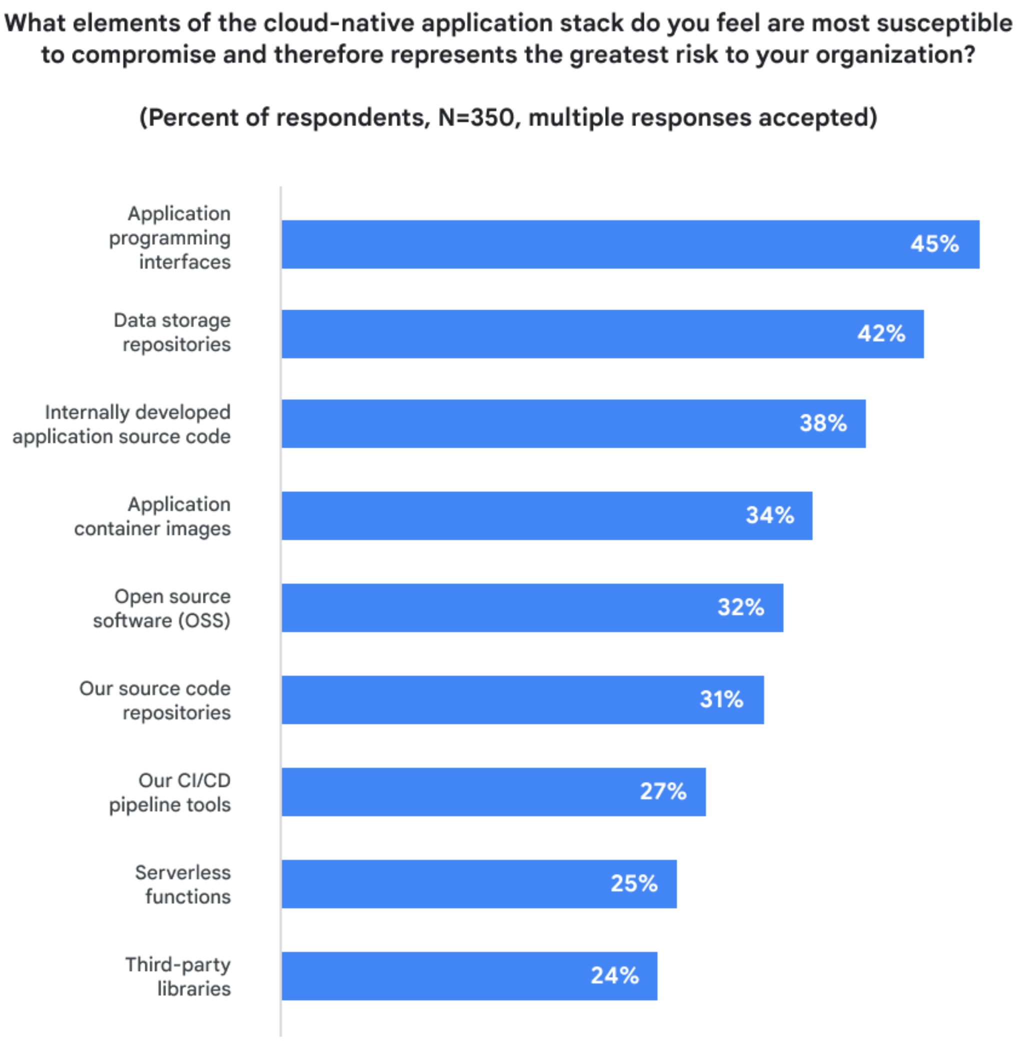
While DoS, injections, and ATO are well-known attacks that came to the API world from web applications, abuse and bots are growing threats for APIs that are by their very nature different from security issues. Abuse does not occur due to security bugs in code. It happens mainly because of the malicious use of legitimate features, such as a free account tier.

Data scraping is ranked at the top of the list as a result of large-scale scrapings of Clubhouse, LinkedIn, Facebook, and Instagram. Scraping publicly available data is the most common abuse type that is interestingly not particularly illegal, according to the Supreme Court's decision in April 2022. However, it's not always publicly available data that's scraped. Attackers may identify a flaw in API user authorization, or just poor authentication, that provides them with access to personal identifiable information (PII) stored in a company's database. The PII leakage can result in the violation of user data privacy and impact a business's reputation, similar to what happened with Parler. Data scraping is especially dangerous for enterprises whose business models are based on selling proprietary data through APIs. If an attacker succeeds with a large-scale scraping, they can potentially ruin the business by offering the same data for a fraction of the price.

Scalping is another emerging API abuse, where an attacker attempts to buy limited-edition products or reserve appointments before a real user gets to it. From the business perspective, scalping can damage a brand, cause DoS to a website that cannot handle volumes of bot traffic, and eventually turn away loyal customers who suddenly cannot buy a product or a ticket to a concert at a reasonable price. In addition, abusers often use fraudulent payment methods as part of a scalping attack, with such fraud leading to direct business losses.

The API-specific threats and incidents are slowing the pace of API-enabled innovation for many organizations. Recent Google research on API security shows that more than half (53%) of the surveyed organizations have delayed the rollout of a new service or application due to API security concerns. Of those organizations that have experienced an incident in the past 12 months, more than three-quarters (77%) have delayed the rollout of a new service or application. Therefore, the growing API footprint and novel API-specific attack vectors demand that API security and anti-abuse be at the forefront of how organizations build and operate APIs to enable rapid and safe business innovations.

Elements of the cloud-first application stack considered most susceptible to compromise



What elements of the cloud-native application stack do you feel are most susceptible to compromise and therefore represents the greatest risk to your organization?

(Percent of respondents, N=350, multiple responses accepted)

Source: ESG, a division of TechTarget, Inc.

Why traditional WAF is not enough

The web application firewall (WAF) concept emerged in the late 1990s as a response to the increasing amount of attacks on web applications. Early innovators wanted to present a security solution, which was often a physical device, that would serve as a sort of "firewall." Such firewalls included protections against exploitations of SQL injection (SQLI) and cross-site scripting (XSS) vulnerabilities. Later, many WAFs expanded protections to DoS and distributed denial-of-service (DDoS) attacks. Your security team most likely has at least one WAF protecting web resources.

In the wake of API growth, WAF vendors have adapted their solutions to protect APIs. As a result, many companies continue relying on WAFs as their first line of defense for web resources, including web APIs. However, WAFs alone don't provide comprehensive protection to APIs and may struggle to address API-specific threats coming from misconfiguration, bot, and credential-related attacks. Internal APIs used for application integration also remain unprotected by WAFs. That's why your organization needs to take a holistic approach to protecting application footprints, including web, mobile applications, and APIs – and you need to build a defense in depth to address a complex and rapidly evolving threat landscape.

Build defense-in-depth to secure APIs

Defense-in-depth strategies are widely adopted on premises and in the cloud. The idea of properly layering defense mechanisms that support, but don't depend on each other, has proven to be effective at stopping adversaries. At Google, our API defense-in-depth strategy is built on four pillars:

01 First, we ensure that essential API security controls and protections are in place for all our APIs. These controls are enforced through a common API management platform that guarantees their proper application to all APIs, thus leaving no space for any uncontrolled API exposure.

02 Second, we protect our APIs from DDoS and exploits with an adaptive cloud protection suite that includes a WAF, machine-learning-based DDoS protection, and a threat intelligence capability.

03 Third, we add a layer of anti-bot protection to keep APIs and exposed resources safe from fraudulent activity, spam, and abuse.

04 Finally, we develop APIs with an adherence to safe coding principles to prevent the most common classes of security issues upfront.

API management platform

An API management platform is an essential component of any API security strategy. On the one hand, it brings together the best of API management and integration, so you can connect existing data and applications, and surface them as easily accessible APIs that can power new experiences. It allows IT teams to scale operations, accelerate developer velocity, and increase the speed to market. On the other hand, an API management platform protects APIs through their proper discovery, inventory, safe gateway, authentication, authorization, quotas, and throttling mechanisms in a developer-friendly way.

In addition, an API management platform provides governance over the full API life cycle, which typically includes design, build, operate, engage, and version management phases.

The three base components of API governance are: API hub and registry, governance engine, and API portfolio management.

These components help establish and apply policies to APIs across the whole enterprise, and make them discoverable and usable in a secure and compliant way.

Google customers are using Google Cloud's Apigee API Management to build and scale API programs in a safe and secure way to gain actionable insights across the entire API value chain, monetize API products, and maximize the business value of digital assets. For example, Apigee helps secure customer data at Experian and accelerate time to market, while Autodesk uses Apigee to offer secure access to services.

Web application and API protection

Web application and API protection, or WAAP, was introduced in 2021 by Gartner analysts. With this, Gartner expanded the evolution of WAFs to include DoS protection, bot management, and API protection. A WAF brings traditional capabilities to detect and block exploitation attempts of application vulnerabilities. An anti-DoS component ensures the availability of APIs by protecting them from DoS and DDoS volumetric and application-level attacks through excessive ad hoc bandwidth, throttling, and blocking requests.

API protection comes in the form of an autonomic API security layer that provides visibility and control over API security configurations, and ensures continuous compliance with internal security policies and external regulatory requirements. In addition, bot detection can identify malicious usage like data scraping, scalping, or ATO attempts at scale.

Major cloud and security providers have already included WAAP-related services in their offerings; however, since it's a relatively new term, exact WAAP features may vary from vendor to vendor.

Google's WAAP provides end-to-end protection to web applications, and to external and internal APIs through a solution suite that includes Apigee Advanced API Security, Google Cloud Armor for next-gen WAF DDoS protection, and reCAPTCHA Enterprise to combat automated bots and detect online fraud.

Apigee Advanced API Security also includes adaptive abuse protection that detects bot activities using privacy-preserving machine learning and allows it to automatically block the attacker's requests before an attack scales up. Google's WAAP solution also helps uncover and remediate vulnerabilities in web applications and APIs. In addition, Google Cloud provides necessary API security features such as customer-managed encryption keys (CMEK), Identity and Access Management (IAM), threat intelligence with Chronicle security operations, and other cloud-first security capabilities.

API safe coding

The use of API management infrastructure and advanced WAAP significantly reduces API security and abuse risks. However, it's critical to prevent security issues in APIs from the beginning. Unsurprisingly, injections, authentication/authorization, and business logic security flaws continue to represent major security threats.

At Google, we've determined that safe coding can readily scale to eliminate entire classes of security vulnerabilities. Google's safe coding practices help ensure code security and reliability in common frameworks, languages, and libraries. Ideally, libraries only expose an interface that makes writing code with common classes of security vulnerabilities impossible.

SQLI is a concrete example. SQLI holds the top spot on both the Open Web Application Security Project (OWASP) and the SysAdmin, Audit, Network, and Security (SANS) lists of common security vulnerabilities. An SQLI vulnerability arises when an application executes an SQL query on a database and allows parts of the query to be controlled by an attacker. An SQLI can have serious security consequences and it's been notoriously difficult to avoid in large-scale API developments. As shared in one of our recent books, Building Secure and Reliable Systems, when you use a hardened data library such as TrustedSqlString, these types of vulnerabilities become a significantly smaller issue. While this approach may not solve all security challenges, it dramatically increases the speed to market for APIs.

Launching your API-first security strategy

Organizations across the world are developing APIs to speed up innovation and enable easier, more standardized delivery of services and data for digital experiences. As API usage and traffic volumes have grown, so has the need to make API security a top priority.

As part of Office of the CISO, we provide actionable recommendations for CISOs and their teams on how to start or continue the journey to building a defense in depth for APIs – one that enables business growth in a safe and sustainable way. Building on what we've discussed in this report, here are six excellent ways to further support you in this journey:

01 Check out "Best practices for securing your applications and APIs using Apigee"
02 Read "Building Secure and Reliable Systems"
03 Get started with Apigee, Google Cloud Armor, and reCaptcha Enterprise
04 Explore Google security papers
05 Read about the layered security approach for APIs

06 Listen to “Protect Modern Applications in the Cloud: Union of API and Application Security” on Cloud Security Podcast