kasada

WHITE PAPER

The Hidden Risks of APIs: How to Protect Your Brand, Your Business, and Your Data

Executive Summary

While APIs (application programming interfaces) are typically the purview of software engineers, these days it's important for business leaders to understand them as well. That's because APIs can literally be an open door to your enterprise's most valuable digital assets.

In their simplest forms, APIs are a way to easily connect applications and services. They enable engineers to break down code into smaller, more discrete software components that use APIs to communicate with each other—letting them bring products and capabilities to market far faster than ever before. The downside is that as the number of APIs in each company grows into the hundreds or thousands, cybercriminals have discovered that they provide a new and expanding vector for attack.

What does this mean for your business?

Automated attacks on APIs are inflicting serious damage on businesses of all kinds—from financial services, retail, travel, and hospitality to gaming, real estate, media and entertainment, and others. The consequences of successful attacks can be devastating, including lost revenue and market share, a drop in brand value and stock prices, mitigation costs, breach and notification costs and, in some cases, penalties, and more.

If securing your APIs isn't at the top of your company's priorities, then it's time to re-evaluate those priorities. Security, IT, and business leaders should be making the case that protecting APIs needs to be a top focus and a core component of the IT security team's mission and objectives. This white paper outlines why APIs are at great risk today and introduces some best practices for mitigating that risk to protect the brand, the business, corporate intellectual property, and valuable data from attack.

Introduction

Cybercriminals are opportunistic and APIs today are proving to be ripe opportunities for successful attack. Why? First, it's the sheer number of APIs in use. Today, 60 percent of companies report having more than 400 APIs,[1] and APIs now represent 83 percent of all web traffic.[2]

This makes them an enormous and growing target for cybercriminals. Already, APIs account for 40 percent of the attack surface for all web-enabled apps and are predicted to account for 90 percent by 2021, according to Gartner. The industry analyst firm predicts that by 2022 API abuses will become the most-frequent attack vector.[3]

The second reason why APIs are a favored target is that they are significantly under-defended—even though they can directly expose a company's business logic and data. Increasingly, headlines confirm that even the largest enterprises are guilty of API-related security failures. Companies such as Venmo, Capital One, First American Financial, GitHub, Facebook, Instagram, and Google have all been in the news over the past few years for their API vulnerabilities.[4,5]

Protecting APIs can be challenging, not just because of their ubiquity but also because they are created and used by both developers and business users, so security teams don't even know about all of them. Non-technical developers using no-code/low-code platforms can undermine security such as allowing one user to see data belonging to another, or posting sensitive information to a public location. These factors make them an enormous and growing target for cybercriminals.

If your company is primarily focusing on protecting your web applications, but has not taken steps to protect your APIs, you're putting your business at serious risk of attack.

APIs for Everything

While APIs have been around for decades, it's only been more recently that they've become an essential enabler of modern applications and systems. APIs connect things such as applications and services, make it easier to share information, and let developers take advantage of existing internal or third-party services without having to develop the capabilities themselves.

As consumers, we might use dozens of APIs every day—without even knowing it—to accomplish things such as connecting to our bank account from a mobile app, finding the location of nearby drivers for on-demand transportation, or getting directions to a particular location.

In general, APIs fall into three main categories based on who can access them:[6]

- **Private APIs** are intended for internal use only within a company and make up slightly more than half (53 percent) of all APIs. There are many private API use cases common across industries such as retail, ecommerce, travel, gaming, and financial services, among others. These include account creation, universal login pages, and mobile app access, for example.
- **Partner APIs** are ones that provide a specific type of access to partners for B2B use cases, such as a bank providing access to a real-time interest rate service that its partners can integrate into their own applications. Partner APIs represent 28 percent of APIs.
- **Public APIs** (also known as open APIs) are publicly available to any software developer, with no restrictions on access. Public APIs represent 19 percent of all APIs.

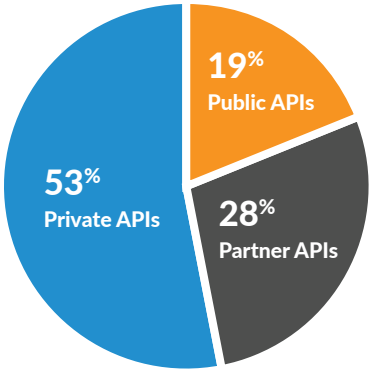

Figure 1: Three Main Types of APIs

By their nature, any of these API types are able to expose application logic and data, including personally identifiable information (PII). This means that any of these APIs are at risk from attack—even private APIs. Especially by malicious insiders or bad actors who have breached perimeter defenses.

1. "6 Lessons from Venmo's Lax Approach to API Security," Maria Korolov, CSO, July 2019.
2. "What You Need to Know About the New OWASP API Security Top 10 List," Maria Korolov, CSO, November 2019.
3. "What You Need to Know About the New OWASP API Security Top 10 List," Maria Korolov, CSO, November 2019.
4. "6 Lessons from Venmo's Lax Approach to API Security," Maria Korolov, CSO, July 2019.
5. "APIs Get Their Own Top 10 Security List," Robert Lemos, Dark Reading, September 2019.
6. "2019 Postman State of the API Report," Postman, 2019.

Figure 2: Example of Automated Attacks: Credential Abuse via an API for User Authentication

Rise of the Bot Army: Automated Attacks on APIs

Automation lets businesses get more done, faster, with the same or fewer staff. This applies equally to illegal businesses such as cybercrime operations. They develop and use bad bots (software designed to carry out a specific task) and botnets (networks of bots) to launch automated attacks. Increasingly, these automated attacks are targeting APIs.

There's no shortage of ways that automated attacks on APIs can inflict damage to your business. Many attacks are the same ones that have been carried out against web applications for years, but now cybercriminals have many new targets in the form of APIs. Some of the most frequent types of automated attacks include:

- Login and credential abuse via authentication endpoints (e.g. to gain access to sensitive data)—often considered the gateway of access to other API endpoints
- Creation of fake accounts (e.g. mobile social accounts)
- Credit card fraud (washing, cracking, stealing)
- Gift card fraud (balance checking, harvesting)
- Content scraping
- Application layer distributed denial of service (DDoS)

Alarmingly, nearly half of companies (45 percent) do not have confidence that they can detect malicious use of their APIs. Half (51 percent) are not confident that their security team even knows about all of the APIs that exist in their organization.[7] It's impossible to protect what you don't know exists.

The Ultimate Victim: Your Bottom Line

Any successful attack on your APIs will result in unforeseen costs to your business—the only question is how high will the damages be? Aside from the remediation costs incurred to stop the attack and investigate how it happened, most of the financial damages come from:

- Lost revenue and market share
- Lower brand value and stock prices
- Breach and notification costs and penalties

Consider the average cost of a data breach—US$3.92 million[8]—and the effect that would have on your business. Lost business is the largest contributor to data breach costs at US$1.42 million, which is 36 percent of the total average cost.

Other types of attacks are costly as well. For instance, fraud (such as credit card fraud) costs financial services companies US$3.25 in true costs for every US$1 in a fraudulent transaction.[9] For retail, every US$1 of fraud (such as gift card fraud) costs US$3.13 for U.S. retailers.[10]

Content scraping, while not necessarily illegal at this point, extracts valuable enterprise content from APIs using malicious bots. Bot operators can then sell or use the data for malicious purposes including share price manipulation, price undercutting, search engine manipulation, data theft, and brand damage.

Figure 3: Average Costs of a Data Breach and Industry Fraud

OWASP Tackles API Security

Since 2004, the Open Web Application Security Project (OWASP) has been working to improve software security. Well known for its Top Ten Web Application Security Risks, OWASP created a new Top Ten list for APIs in acknowledgement of the growing threat. It published its first release of the new API Security Top 10 list in September 2019.

Protecting Your APIs Should Be a Top Priority

What can companies do to better protect their APIs from automated attack? The first place to start is by assessing your current environment and understanding the level of risk your business faces today. This will help inform the development of a strategy and associated policies for securing APIs.

Astoundingly, many security teams can't assess the risk to their companies for their APIs because they don't have visibility into all of the APIs in use. Often, APIs and API security are in the hands of developers and DevOps teams. Each team may have its own set of APIs that it uses. In that situation, no one has visibility into all of the APIs being developed and used across the company. For example, a Kasada client discovered that it had five different authentication endpoints across the business, which represented a much larger attack surface than the company had previously realized.

That's why any security strategy for protecting APIs must begin with a complete understanding of all the APIs developed by the company:

- How many APIs are deployed?
- Who manages/owns the APIs?
- Who is using the APIs?
- Which APIs are exposed to partners?
- Which ones are exposed publicly?
- Which APIs are driving traffic?
- How is that traffic being monitored?

Once you have an inventory of APIs, you can begin evaluating your risk by looking for common API security weaknesses, such as authorization flaws, excessive data exposure, lack of rate limiting, security misconfigurations, insufficient logging, and others. A great place to start is the OWASP API Security Project and its API Security Top Ten report.

7. "Survey says: Security and IT Professionals Are Concerned About Enterprise API Growth," Candace Flynn, Ping Identity, November 2018.
8. "Cost of a Data Breach Report 2019," Study by Ponemon Institute, Results sponsored, analyzed and reported by IBM Security, 2019.
9. "2019 True Cost of Fraud Study: Financial Services and Lending," LexisNexis, 2019.
10. "2019 True Cost of Fraud Study: E-Commerce/Retail Edition," LexisNexis, 2019.

Three Best Practices to Implement

Once you have an understanding of the APIs in your company, common API weaknesses, and the types of threats that can be used against them, it's time to make sure your company is using recommended best practices to help protect your APIs. Starting with the APIs that represent the greatest risk for your business if they are attacked, check to see that the following three important best practices are being followed.

1) LOCK DOWN ACCESS TO THE API

The ability to control API access is a cornerstone of effective API security. Make sure you're authenticating both end users and applications, and make sure that access policies and authentication mechanisms are set up correctly.

The authentication mechanism is a popular target for attack and as such, should be a top priority for extra layers of protection. OWASP's Top Ten report says that authentication mechanisms "are often implemented incorrectly, allowing attackers to compromise authentication tokens or to exploit implementation flaws to assume other users' identities temporarily or permanently." It's critical to understand the authentication mechanisms your organization has in place and then apply authentication best practices to these endpoints.

2) MONITOR AND LOG EVERYTHING

You can't protect your APIs if you don't have visibility into what is happening, and you can't mitigate damage from an attack if you don't know what was impacted. Continuous logging and monitoring gives you that visibility so that you can track and respond to suspicious activity in a timely way.

Logging and monitoring are especially relevant to authentications and preventing an API endpoint from being a gateway to other endpoints. Log all authentication attempts, denied access, validation errors, and response codes so you can track ratios to detect when something suspicious, such as a credential stuffing attack, is happening.

For example, normal users typically have an authentication success rate greater than 50 percent and most likely closer to 70 to 80 percent—as opposed to attackers, who have a much lower success rate, often in the range of 1 to 5 percent. By monitoring the success/fail ratio per hour, this gives you a good signal for suspicious activity that could indicate an attack.

3) IMPLEMENT RATE LIMITING

Rate limiting or throttling helps protect against brute-force attacks, but often the API doesn't impose or enforce any restrictions on the size or number of resources that can be requested by the user. For example, a bad actor might use automated software to generate a large number of consecutive login attempts by systematically guessing passwords. If the API is not protected by rate limits, it may allow this attack to continue indefinitely or until it succeeds—even if that means accessing the API a million times per second, which could make the API unresponsive or lead to denial of service (DoS), both of which impact legitimate users.

Impose rate limits such as the number of requests per user and number of requests per user within a defined timeframe, number of records per page return, request payload size, memory, and CPU usage.

Layers of Protection Against Automated Attack

Best practices around authentication, logging, and rate limiting are worthwhile and effective as a first layer of protection. However, they aren't enough to secure your APIs and protect them from more sophisticated forms of automated attacks. For greater protection, you need additional layers of security that can identify suspicious activity and block it.

For your most sensitive API endpoints (for example, those involved in authentication, account creation, and handling sensitive data) that are at greatest risk from automated attacks, you need to fight bad automation with good automation to detect and stop attacks in real time. An API security solution should:

- Visualize all your traffic including good bots, bad bots, and humans
- Detect bad bots attempting to attack your APIs
- Make it economically infeasible for bots to be successful

While some solutions can accomplish the above, they tend to be complex to implement, time-consuming to manage, and may degrade the end-user experience. Instead, the better solution is one that is easy to deploy and use, doesn't impact user experience or API performance, and not only prevents automated attacks from being successful, but does so in such a way that it effectively deters cybercriminals from attacking that API in the future.

Introducing Kasada API

Kasada API protects an organization's web and mobile APIs from automated attacks, botnets, and targeted fraud. Companies with revenue-focused web and mobile applications that have exposed APIs can benefit from Kasada API, helping digital enterprises across all industries, including retail, ecommerce, travel, gaming, and financial services. Kasada API can be quickly implemented to mitigate online fraud losses, lower operating costs, and create a frictionless user experience. Future software releases will deepen Kasada's capabilities for protecting the assortment of private, public, and partner APIs across organizations.

Use Cases

Kasada API, delivered as a cloud-based service, has a simple deployment model that protects APIs with long-term efficacy from the very first request. This allows application developers to innovate rapidly using APIs without sacrificing security and no additional resources required from a customer's IT department.

Protect your customers, end users, and organization from:

- **Account Takeover**
  Protect your users from having their accounts hacked by abusing stolen credentials.
- **Fake Account Creation**
  Prevent attacks and illegitimate actors from creating fake accounts for scams or accessing your content.
- **Loyalty Program Abuse**
  Protect your customer loyalty program by defending your APIs that serve users with rewards services.
- **API Scraping**
  Get in control of your content. Prevent competitors and illegitimate actors from scraping your web and mobile APIs.

How It Works

Kasada API quickly identifies malicious bot requests through APIs to protect web and mobile apps from automated threats. An organization's most sensitive API endpoints (i.e. authentication, account creation, and handling sensitive data) are at the greatest risk and require that automated attacks be deterred and stopped in real time.

Kasada API deploys the same technology that is utilized for its enterprise solution, Kasada Web, which deters synthetic traffic with sensor detection and inspection process and an increasingly difficult cryptographic challenge that makes it arduous and expensive for bots to continue their attacks.

Specifically, malicious automation leaves traces of itself in the client-side environment (browser, mobile phone). Kasada uses "sensors" to collect advanced attributes from the browser or mobile device through a "client interrogation" process, which occurs in the background.

Kasada API customers are provided with either embedded JavaScript SDKs for web applications or mobile SDKs for native Android and iOS apps. Using proprietary techniques, Kasada API presents a myriad of obstacles to frustrate and disrupt the operating model of bot attacks, preventing hackers from using automation and challenging critical aspects of the attack process.

Each SDK uses sensors to collect signals from the environment. This sensor data is then processed by Kasada's data engine of analyzers to identify:

- if the mobile device is real or fake
- If the browser is being automated using a bot

For example, when your mobile app calls to your API, Kasada's SDK will decorate the request with a token that's computed from the sensor data. Kasada will mitigate API requests from malicious clients with a deceptive response.

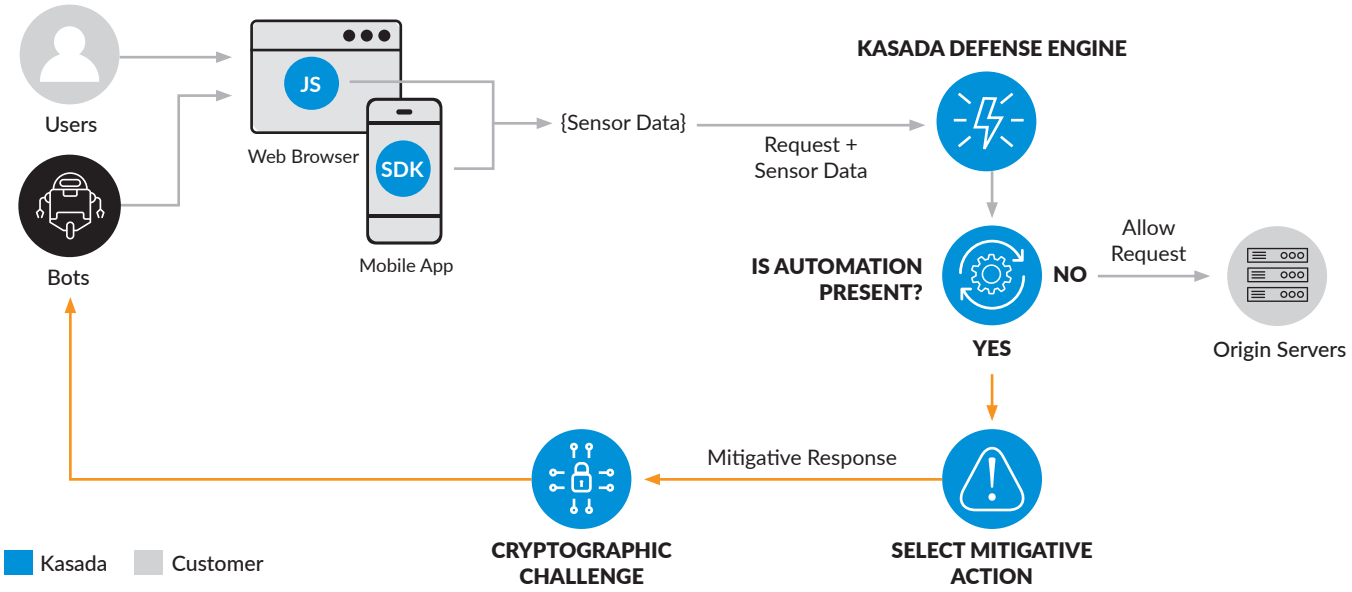

Figure 4: How Kasada API Works

In the Kasada Portal, customers can easily and instantly change between two modes to achieve their desired level of protection:

- **PROTECT** - reports and mitigates malicious traffic
- **MONITOR** - only reports on, makes no active decisions

Customers who want to see what would be detected as malicious have the option to start with MONITOR mode first.

How is Kasada Different?

Our client interrogation is focused on the complex relationship between various client-side attributes. It is not the same as "fingerprinting" used by first-generation bot mitigation vendors, which try to construct a tracking ID (fingerprint) by collecting uniquely identifying data and invading your users' privacy. In practice, fingerprinting doesn't work very well, since browsers are changing in a trend known as the "anti-tracking movement." In addition, instead of relying on contextual data from the past like other bot mitigation vendors, which takes time and ongoing maintenance, Kasada looks for immutable evidence of automation from the very first request and is able to provide long-term efficacy by obscuring its methods as attackers shift their tactics.

A cryptographic challenge is used as a proof of work that exponentially increases the difficulty level with the number of abusive requests over time, therefore exhausting the CPU resources of bad bots, without informing the attacker. This forces the attack to permanently cease, as its ROI inevitably collapses. Not only does Kasada neutralize the attack long-term, but also prevents the bot operator from quickly retooling or attacking other targets, as all CPU resources have been exhausted. Fraudsters then stay away from Kasada-protected properties, as it costs them virtually unlimited resources trying to break in.

Key Benefits

PROTECT YOUR REPUTATION AND REVENUE

Kasada lets you prevent automated attacks on your APIs before they cause you damage. We are devoted to researching attackers and their tools so you don't have to. This saves your team time and your company money.

PROTECT ALL CHANNELS

Kasada lets you detect attacks across all channels of traffic to your APIs. A simple deployment model with long-term efficacy allows application developers to innovate rapidly using APIs without sacrificing security.

REDUCE USER FRICTION

Kasada allows you to reduce friction for your end users. CAPTCHAs and other human-facing challenges cause high friction for users; that's why Kasada doesn't use them. Also, good security should be invisible and that's why our solution works in the background, invisible to your users. As well, by preventing attack traffic from putting load on your API servers, Kasada can enhance your API performance for legitimate users.

GET VISIBILITY

The Kasada Portal provides dashboards and rich visualizations to help you understand your traffic. You can see legitimate traffic coming from users of your web and mobile apps. You can see bot traffic imitating users or hitting your API directly.

We'd love to demonstrate the Kasada difference to you; please request a demo today.

Conclusion

Whether it's fraud, content scraping, or credential abuse that leads to a data breach, DDoS, or some other form of attack, under-protected APIs are a favorite target of cybercriminals. You can stop these attacks from being successful by:

- Making API security a top priority for your company and your IT and security teams
- Applying the best practices described here and in the OWASP Top Ten for API Security
- By layering an automated solution able to detect malicious automation on top of these best practices to protect your most sensitive and valuable APIs.

About Kasada

Operating globally since 2015 and trusted by enterprises worldwide, Kasada gives internet control and safety back to human beings through its category-defining digital traffic integrity solution. With Kasada, even the stealthiest cyber threats are foiled, from login to data scraping across web, mobile, and API channels. Scalable up to multi-billion-dollar companies, onboarded in just minutes, and designed to deliver clear ROI in multiple departments, Kasada's solution invisibly defends and enhances critical business assets while ensuring optimal online activity, with immediate and lasting traffic security. Kasada is based in New York and Sydney, with offices in Melbourne, San Francisco, and London. For more information, visit www.kasada.io.