SOTI

V10 ISSUE 01

10 YEARS
OF SECURITY INSIGHT

Lurking in the Shadows

Attack Trends Shine Light on API Threats

Akamai

State of the Internet/Security

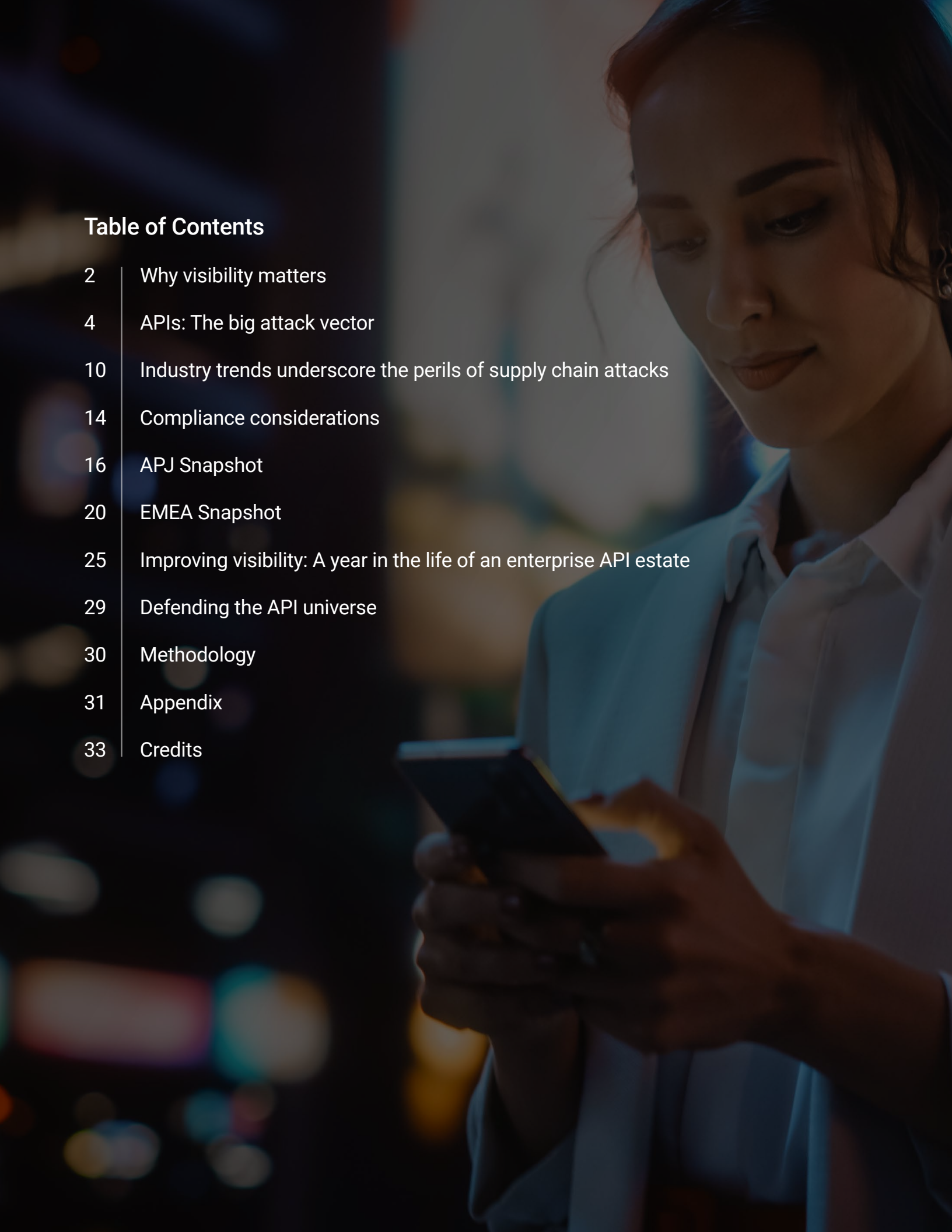


Table of Contents

Why visibility matters

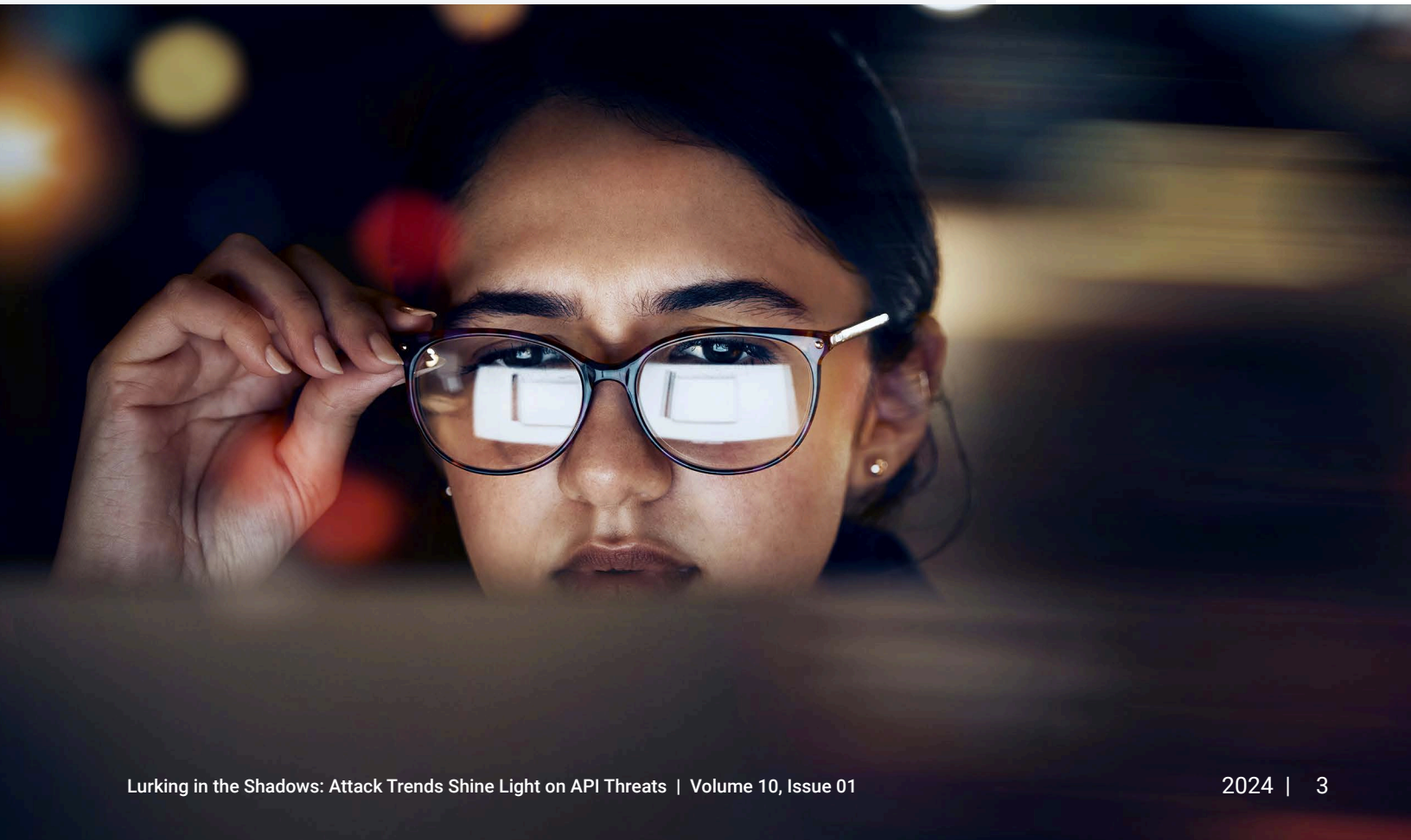
This year marks 10 years of Akamai's sharing of security research via our State of the Internet (SOTI) reports. The focus of these reports has changed over the years as both the operational and threat ecosystems have evolved. Starting in 2024, instead of looking at web application and API attacks as one single subject, we will use a new data set that allows Akamai researchers to distinguish between the two types of attacks. In this report, we focus on the percentage of web attacks that are targeting APIs (for more details, please read the Methodology section). This will help us better understand how adversaries are attacking APIs and offer more effective mitigation strategies.

APIs are foundational to many of the recent changes within companies that have improved both employee and customer experiences. Unfortunately, this digital innovation and the rapid expansion of the API economy have presented cybercriminals with new opportunities for exploitation. Therefore, visibility is a critical aspect of API security. Once blind spots like shadow APIs or rogue APIs are illuminated, security teams can start to address vulnerabilities that they were previously unaware of.

In this first SOTI report of 2024, we highlight the array of attacks that are hitting APIs, including traditional web attacks, and tackle the dangers of API abuse through common problem areas such as posture and runtime challenges that we have visibility into via our data. Additionally, we illustrate the dangers by industry and region so you can more accurately evaluate the risk to your company. We also present several real-world case studies, reinforce compliance requirements, and show how legislation trends can shape your security strategies. We conclude the report with steps to improve your visibility into your API landscape, which can enhance your overall security posture.

Key insights of the report

- A total of 29% of web attacks targeted APIs over 12 months (January through December 2023), indicating that APIs are a focus area for cybercriminals.
- The attacks on APIs include the risks that are highlighted in both the Open Web Application Security Project (OWASP) API Security Top 10 and the OWASP Top 10 Web Application Security Risks, with adversaries using tried-and-true methods like Structured Query Language injection (SQLi) and Cross-Site Scripting (XSS) to infiltrate their targets.
- Business logic abuse is a critical concern as it is challenging to detect abnormal API activity without establishing a baseline for API behavior. Organizations without solutions to monitor anomalies in their API activity are at risk of runtime attacks like data scraping — a new data breach vector that uses authenticated APIs to slowly scrape data from within.
- APIs are at the heart of most digital transformations today so it is paramount to understand the industry trends and relevant use cases, such as loyalty fraud, abuse, authorization, and carding attacks.
- Organizations need to think about compliance requirements and emerging legislation early in their security strategy process to avoid the need to re-architect.



APIs: The big attack vector

By design, APIs are conduits of data — and, as such, can expose that data to attackers once they gain unauthorized access, often by way of vulnerability exploitation or attacks against the business logic. In 2022, Gartner predicted that API abuse and data breaches will nearly double by 2024. Fast-forward to the present when high-profile API incidents are more common than ever. In fact, last year, the Open Web Application Security Project (OWASP), a nonprofit organization known for their list of top 10 security risks, released a separate list of API-specific risks, the OWASP API Security Top 10, which recognizes the unique threats posed by APIs.

Akamai research observed that APIs are being targeted by both traditional attacks and API-specific techniques, which require a blend of protections. In fact, we saw that nearly 30% of overall web attacks targeted APIs from January through December 2023 (Figure 1). These attacks will likely continue to grow as the demand for API use increases unless organizations properly secure their APIs or account for all APIs in their environment. Understanding the full extent of an attack surface begins with a comprehensive and accurate inventory of APIs.

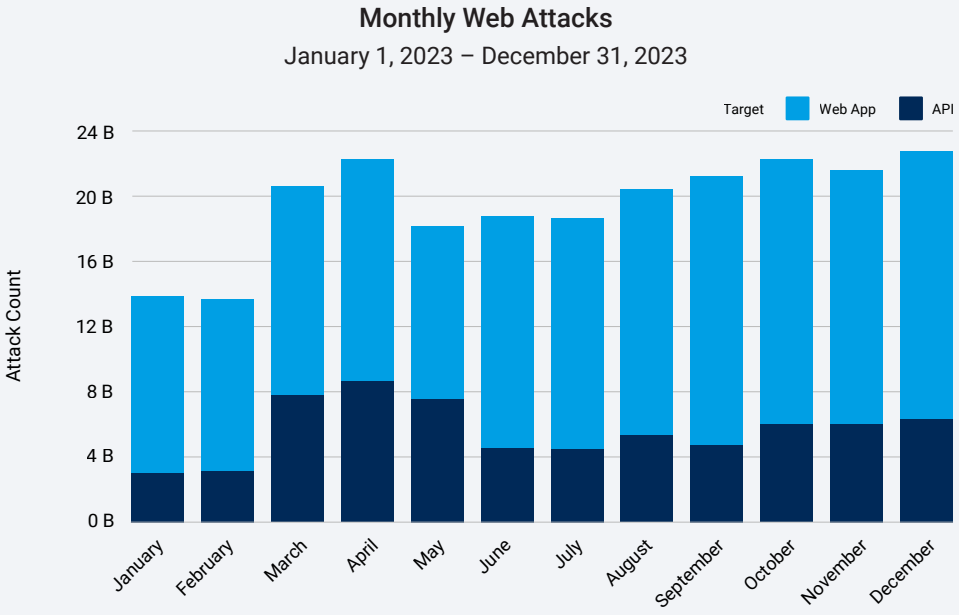


Fig. 1: Web attacks against APIs grew from 22% in January to 28% in December, with several fluctuations between March and May 2023

We also observed some interesting trends globally, with the Europe, Middle East, and Africa (EMEA) region experiencing the greatest ratio of web attacks that targeted APIs (47.5%), followed by North America (27.1%) and the Asia-Pacific and Japan (APJ) region (15%). At the country level, the top areas were Spain (94.8%), Portugal (84.5%), the Netherlands (71.9%), and Israel (67.1%). It is worth mentioning that, in comparison, only 27.6% of web attacks in the United States targeted APIs.

There are a number of reasons for differences in regional attacks, such as regulatory environments, geopolitical conflicts, infrastructure types, access and education variations, business models, and social factors. However, it is also important to note that you can see a cyberattack trend start in one region or industry, then migrate to others; it is, therefore, worth tracking broader trends. For a more detailed discussion of regional trends, read the APJ Snapshot and the EMEA Snapshot within this report.

APIs under attack

An examination of how adversaries are targeting companies' APIs, and the tactics they frequently use, can shed light on the areas of defense you should focus on. In the past 12 months, HTTP Protocol (HTTP), Structured Query Language injection (SQLi), and Data Harvesting attacks were some of the adversaries' favored techniques (Figure 2). In an HTTP attack, adversaries exploit vulnerabilities in various protocols for malicious purposes, such as to read sensitive data and spoof clients or servers, among others. Another popular technique, Active Session, pertains to any instance where suspicious attack traffic is flagged and blocked for the duration of that session. On the other hand, Data Harvesting as the name implies, pertains to attacks related to gathering or collecting information, which attackers can then use for other attacks in the future. (See the appendix at the end of this report for a complete list of attack vector definitions.)

Our latest dataset enables us to monitor for additional vectors against APIs. Case in point: Server-Side Request Forgery (SSRF) is one of the up-and-coming vectors we tackled in last year's SOTI, Slipping Through the Security Gaps, and it is employed to gain sensitive information or to execute commands.

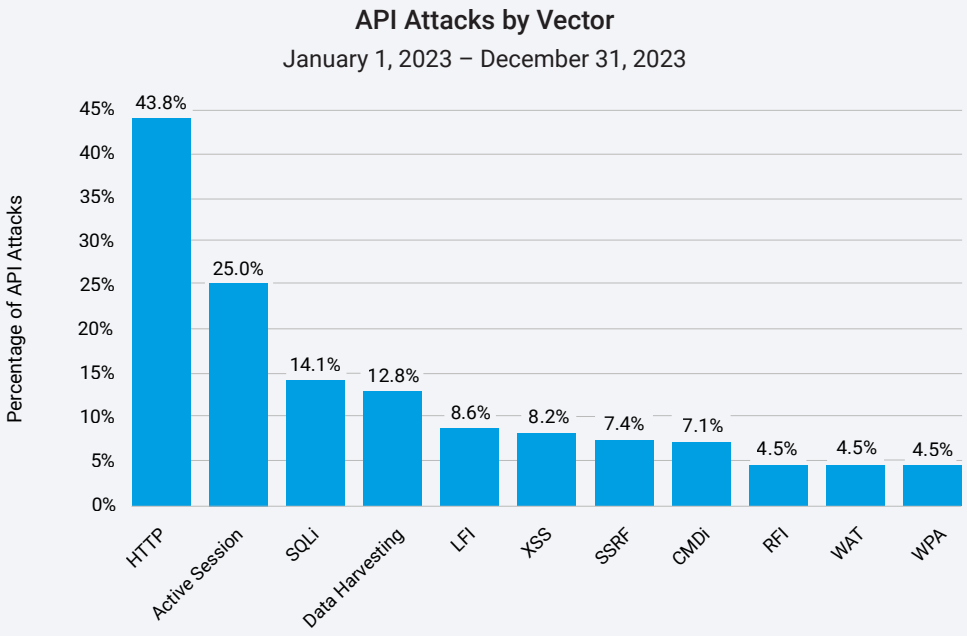


Fig. 2: Although Local File Inclusion (LFI) is not the top vector in APIs, it's still an area of concern as it can be used to infiltrate intended targets; a closer look at the distribution of attacks for both web applications and APIs, however, reveals LFI is still one of the top attack vectors

Findings from our research also revealed that bot requests are an area of concern. Based on Akamai data, almost one-third of suspicious bot requests were aimed at APIs globally in 2023. Although not necessarily all malicious, these bot requests can be weaponized to conduct credential stuffing attacks and data scraping, which may lead to information theft.

We highlight these types of attack to point out that beyond the OWASP API Security Top 10, attacks on access, data abuse/scraping, and configuration mistakes, there are still a number of direct attacks that companies need to track and have their pen test teams and red teams examine.

Real-world lessons in API security

Akamai works closely with enterprises globally to collect detailed information about API use and to perform advanced behavioral analytics to identify security vulnerabilities and indicators of API abuse. From Akamai's view of API activity, we generally see two distinct problems: posture problems and runtime problems.

1. **Posture problems** pertain to flaws in the enterprise's API implementation. Alerts indicating posture problems aid security teams in identifying and remediating high-priority vulnerabilities before they can be exploited by attackers.

2. **Runtime problems** are active threats or behaviors that require an urgent response. While often critical in nature, these alerts are more nuanced than other types of security alerts since they take the form of API abuse (as opposed to more explicit infrastructure breach attempts).

Most common posture problems

The following are the most common posture issues we observed, along with a brief overview of their potential impact on the enterprise if left unaddressed.

Shadow endpoints
Shadow endpoints are outdated or previous versions of APIs that have not been retired or documented. They are sometimes referred to as zombie, rogue, or legacy APIs, and they pose a higher risk of exploitation since they are not subjected to the organization's standard security controls and measures.

Unauthenticated resource access
Unauthorized resource access refers to scenarios in which a user or system is able to access API resources without providing any form of authentication, often as a result of flaws in the API implementation or configuration. Although many unauthorized resources are hidden through obscurity, it's possible that attackers who find them will exploit them to access sensitive data or application functionality.

Sensitive data in a URL
In some instances, sensitive data — such as passwords, authentication tokens, credit card details, and personally identifiable information (PII) — may be observed in the URL of an API request. Data in URLs is often stored in places that may become accessible to attackers, like logs and caches, creating a significant risk of sensitive data leakage and compliance issues.

Permissive CORS policy
A permissive cross-origin resource sharing (CORS) policy is a situation in which an API allows requests to originate from a wider range of origins (such as protocols, domains, and ports) than necessary. An overly permissive policy makes it easier for attackers to access sensitive resources from an untrusted source, as well as execute attack techniques like Cross-Site Scripting (XSS) more easily.

Excessive client errors
Excessive client error alerts are generated when an abnormally high number of failed API resource requests are observed. While many client-side errors are caused by misconfiguration and other nonmalicious errors, excessive client error alerts are a possible indication that an attacker is probing the API implementation for vulnerabilities.

Most common runtime problems

When we performed a similar analysis of the runtime alerts we observed, we identified these common API security issues that represent potential active threats.

Unauthenticated resource access attempt
This is a more urgent derivative of the unauthenticated resource access posture alert described in the previous section, where we see specific attempts to access sensitive API resources without the appropriate authentication. Even if the observed attempts are unsuccessful, these scenarios suggest an active attempt to find and exploit API vulnerabilities, which may eventually be successful without prompt intervention.

Abnormal JSON property
API activity with unusual JSON payloads, such as unexpected data types, abnormal size, or excessive complexity may indicate an active attempt to exploit a vulnerable API. This activity may indicate an attempt to perform a variety of malicious actions, such as injection attacks, denial of service, data exfiltration, or exploitation of API logic flaws.

Path parameter fuzzing attempt
Path parameter fuzzing is another example of deliberately sending unexpected or malformed data as a part of API requests, with a focus on the parts of the URL that RESTful APIs use to specify certain resources or operations. It's another technique that attackers use to perform reconnaissance to discover potentially vulnerable APIs that can be targeted with data exfiltration or service disruption attempts.

Impossible time travel
When analyzing API activity, there are scenarios in which the timestamps, geolocation, or sequence of API calls are illogical, which suggests that attackers are attempting to manipulate them in some way. Additionally, this type of behavior may represent several possible threats such as data manipulation as part of fraudulent activity.

Data scraping
Data scraping refers to the automated extraction of data from an API in a manner and volume that does not align with the intended use and terms of service for the API. Attackers often collect this data slowly to avoid detection and steal intellectual property, gather sensitive customer data, or gain some kind of profit. When it goes undiscovered within APIs, low and slow data scraping is a potentially massive data breach attack.

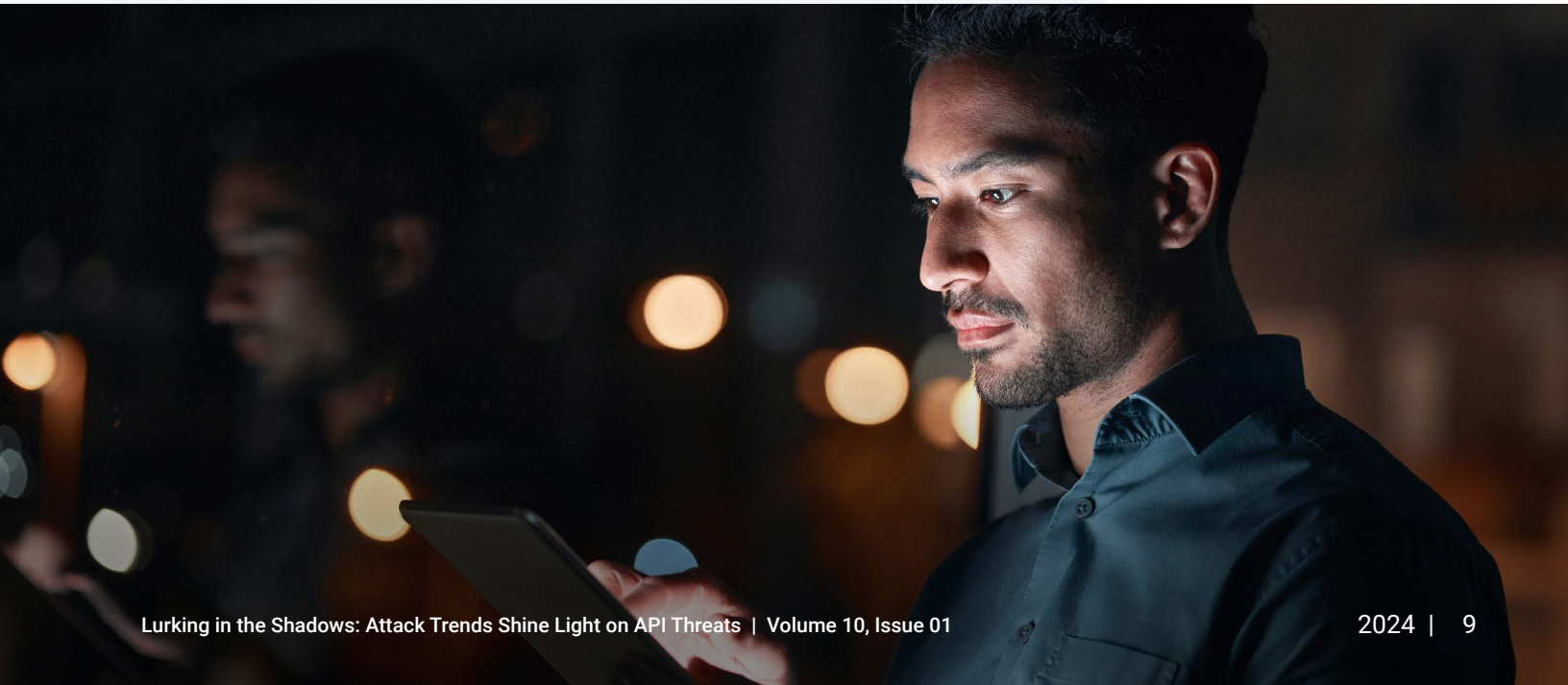
Finally, as you think about posture and runtime problems, it can be helpful to step back and look at three more general challenges APIs face:

1. **Visibility** — Do you have process and technical controls to ensure all APIs are protected by your program? This is a key issue as APIs are often part of the transformation or embedded in new products, so many do not have the same level of directions, protections, and validations as a traditional web presence.

2. **Vulnerabilities** — Are your APIs following best practices for development? Are you avoiding OWASP's most common poor coding issues? Furthermore, are you tracking and checking for vulnerabilities?

3. **Business logic abuse** — Do you have a baseline of expected traffic? Have you established what constitutes suspicious activities?

It is crucial to have visibility into your APIs and the ability to conduct investigations — and to have processes established to rapidly mitigate threats. This is true for both customer-facing and internal APIs.



Industry trends underscore the perils of supply chain attacks

APIs are at the heart of digital transformation in organizations. However, the existence of APIs heightens the risk exposure of businesses and poses a significant security challenge. During the reporting period, 44.2% of web attacks that impact organizations in the commerce industry targeted APIs, followed by organizations in the business services industry at 31.8% (Figure 3). The heavy slant on commerce is due to a multitude of factors, including the complex nature of its ecosystem, high reliance on APIs, and troves of confidential customer information.

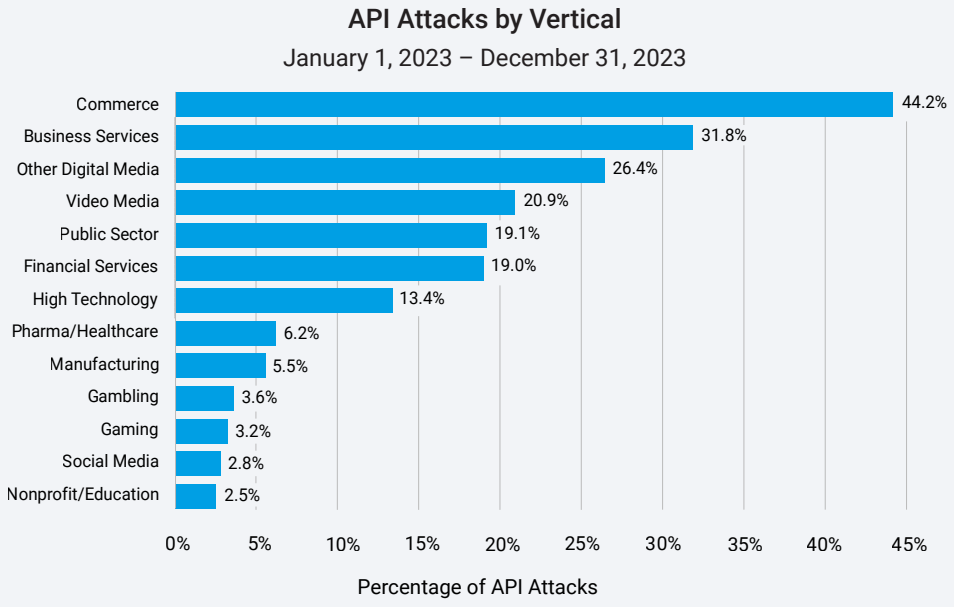


Fig. 3: The commerce and business services verticals experienced the highest percentage of API attacks in 2023. Financial services in the United States — which, unlike the financial services industry in EMEA, has not adopted open banking — was not among the top five.

It should be of concern that business services ranks second because of the potential dangers of supply chain attacks. Third-party companies that offer business services may possess confidential information about their affiliated organization, or even have access to their environment, which attackers may use as a pathway to high-value targets.

A closer inspection of our data reveals that no vertical is immune to API attacks. For example, healthcare's Internet of Medical Things (IoMT) explosion and data interoperability efforts have fueled their API use. The healthcare industry is at significant risk as the security implications of APIs in healthcare are not yet fully understood.

Case studies

To give you insights into what types of attacks we've seen against various industries, we present several case studies, including their real-world implications for the organizations and customers.

Loyalty fraud in the commerce vertical

Fraudsters are targeting loyalty program accounts as they contain valuable currency, such as points, miles, or credits, that is redeemable for real-world goods or cash. Akamai detected a behavior on an API where a user was accessing more than five loyalty accounts. Upon investigation, we identified that this behavior was fraudulent within these accounts. Most accounts are accessed by only a small number of authorized users, so users who access multiple accounts may potentially indicate abusive behavior. To help reduce fraud, it is best to understand the differences between normal behavior and abusive behavior.

API abuse in SaaS notification service

The Akamai Hunt team spotted API abuse in a software-as-a-service (SaaS) notification system that belonged to a financial services company. The authorization header and signature inside the payload were missing, which prevents the company from checking who is making requests in the system and sending notifications. As such, anyone who has access to the API can possibly abuse it to send messages to the company employees and customers.

Potential BOLA attack

The team found a potential Broken Object Level Authorization (BOLA) attack in an airline company wherein authenticated users from nearly 200 IPs fuzzed the {customer_id} path parameter, which then returns sensitive user information if the supplied parameter is valid (Figure 4). Because this ID is a simple integer, fuzzing is very easy, but the impact can be damaging because it can lead to data exfiltration of customer information. The API being targeted returns sensitive information such as first name, middle name, and last name, together with government ID, nationality, country of residence, and date of birth.

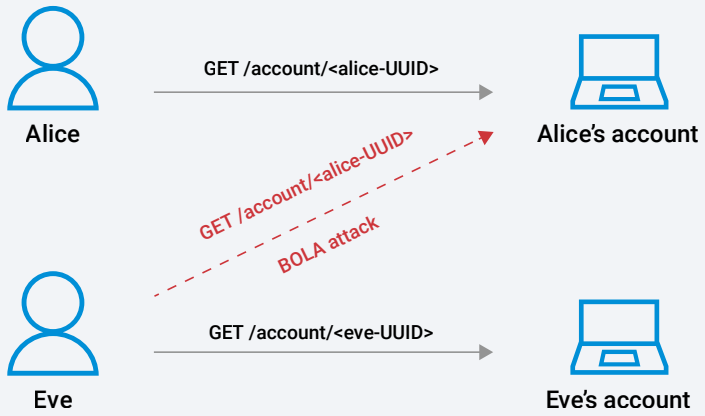


Fig. 4: BOLA attacks occur when adversaries attempt to access resources they are not authorized to access in order to steal sensitive data

Carding attack hidden in plain sight

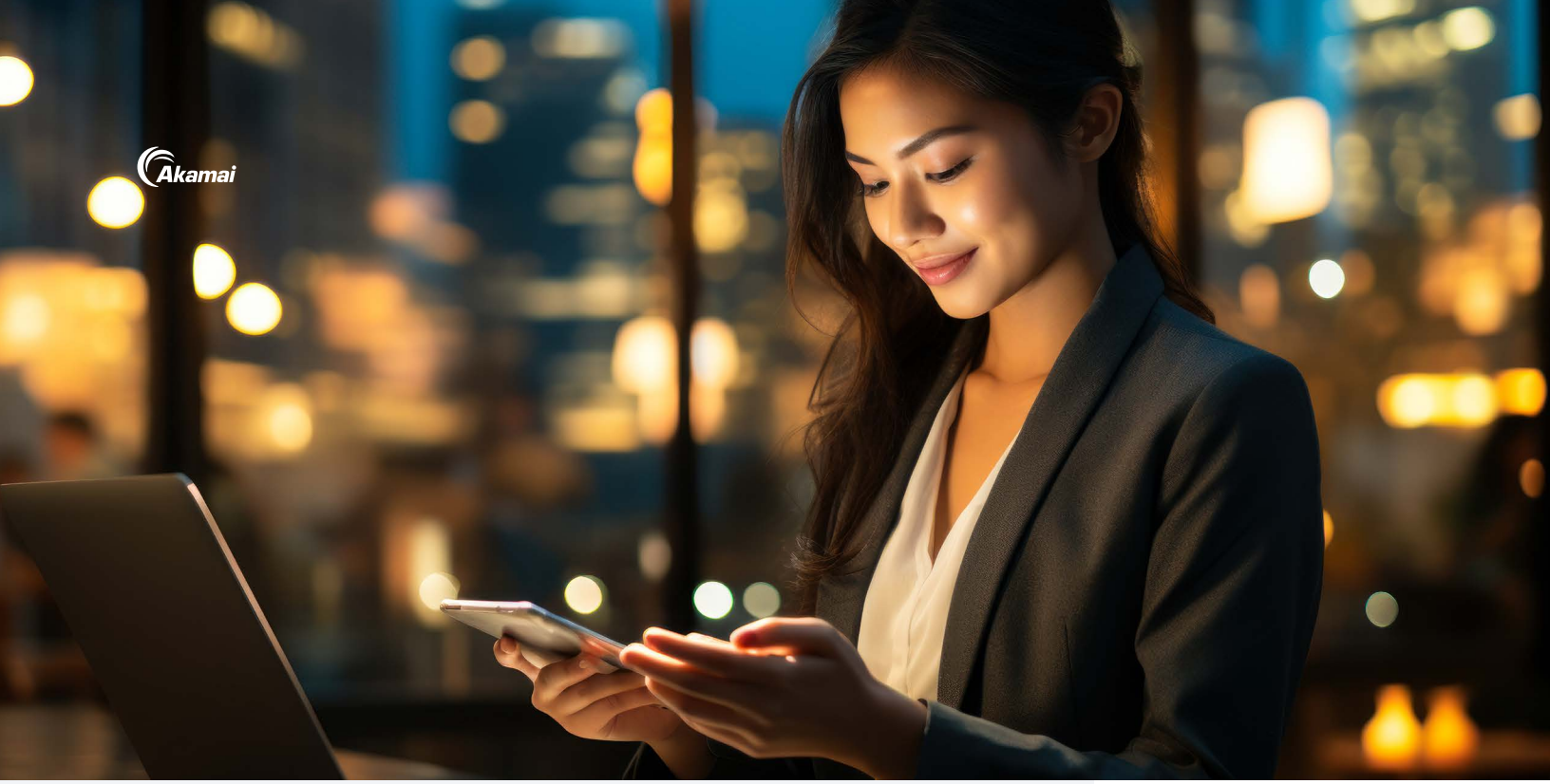
In another case, what started out as abnormal API traffic turned out to be a carding attack. Initially, the affected organization thought it was Distributed Denial-of-Service (DDoS) activities that were behind the spike in API traffic; however, upon closer inspection, it became clear that the attack attempted to validate credit cards, with the customer system responding with true or false (i.e., valid or not). In carding attacks, adversaries validate the stolen credit card numbers and once verified, the attacker can sell them in dark web marketplaces or perform other fraudulent transactions.

Why understanding API attacks is essential

A common business problem present in most API environments is a programming error or a configuration mistake that is detected during the discovery phase of maturing their API security program. Although the majority of these errors are never exploited, the potential damage is apparent to security teams once they gain visibility into the API estate and the traffic running on each API.

Too often, applications and business processes involving APIs are initiated and deployed faster than security teams can evaluate their posture. This seems to make misconfigurations and vulnerabilities inevitable. Add to the mix the lack of API security expertise inside most organizations, and you have all the variables for an unpalatable security equation.

> **The conditions for API security errors**
>
> Rapid deployment of business-critical process using APIs + Lack of visibility into APIs = Misconfigured or vulnerable APIs



Best practices to prevent a data breach

Unknown API vulnerabilities exploited over time are often heavily linked to programming errors. Today, publicized data breaches involving APIs are commonplace, indicating that attackers are now exploring API estates and performing reconnaissance to identify specific APIs to exploit. This exploration, together with the automated threat of data scraping, means that APIs are the new data breach vector. The ramifications of such successful attacks include damages to brand and reputation, the loss of confidential data and customer trust, and — depending on where you are located — compliance and legislative issues that could result in financial losses. As such, API security is more vital than ever.

The conditions for API data breaches

Misconfigured or vulnerable APIs in production	+	Data scraping automated threat (bot)	=	Low-and-slow data breach over weeks or months

The first critical aspect of protecting against a data breach via API vulnerabilities is having visibility into your environment with an understanding of what is normal and what data is on which APIs. This includes putting all APIs behind security controls and having automated responses to mitigate attacks or to alert the security operations team. Next, practicing shift-left testing during development can address these vulnerabilities and weaknesses at the onset, before attackers can exploit them. Finally, you need to run exercises to validate both preventive measures and crisis response.

A **2023 study** conducted in part by Kong analysts stated that "API attacks ... currently cost $10.6 billion in the U.S., and that's set to jump to $198 billion annually by 2030."

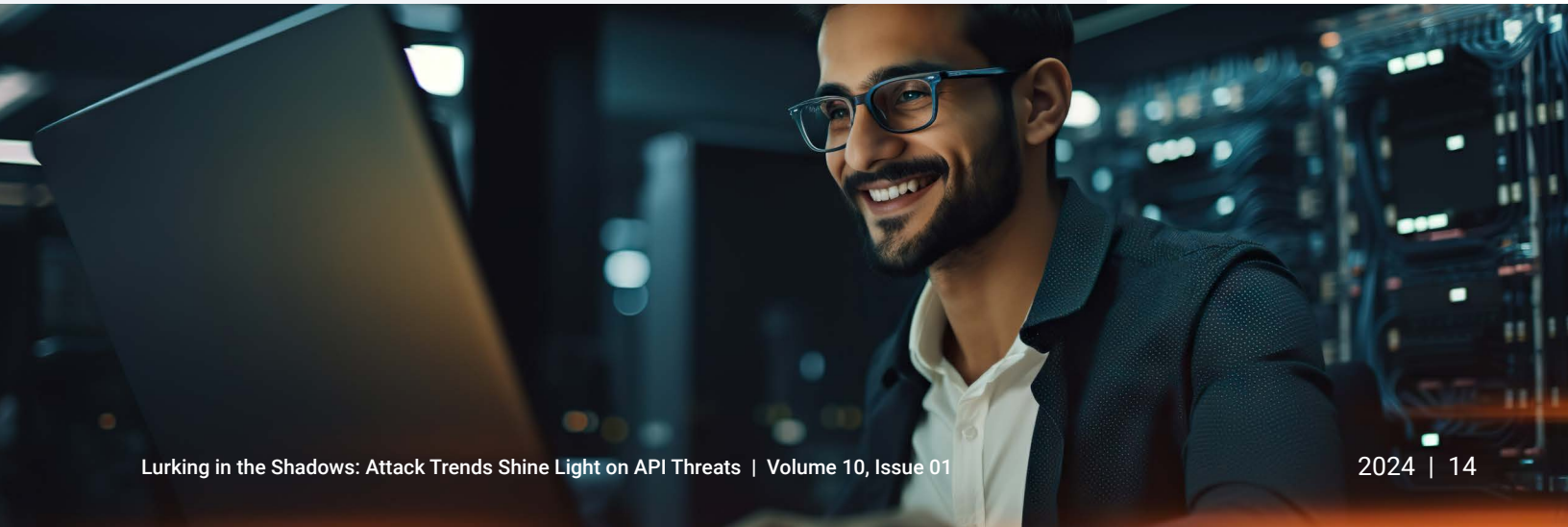
Compliance considerations

Securing APIs and shutting off these entry points as a rising vector of attack is a clear imperative from the classic enterprise security and risk management point of view. Recent legal and enforcement trends related to security and data protection provide additional compelling reasons to address API security and observability issues.

Security has always been a part of data protection laws around the world — you cannot have good privacy without good security. Article 32 of the European Union's Global Data Protection Regulation (GDPR), for example, requires that entities processing PII "shall implement appropriate technical and organisational measures to ensure a level of security appropriate to the risk." Other laws, such as the California Privacy Rights Act (CPRA), include similar requirements to implement "reasonable security procedures and practices," and to take appropriate measures to safeguard the confidentiality, integrity, and availability of PII.

Regulation and enforcement actions are similarly raising the bars for transparency and accountability. The United States Securities and Exchange Commission (SEC), for example, has recently enacted new rules for public companies that require the disclosure of material security incidents, as well as detailed information about risks, security governance, and oversight.

Around the world, companies are being fined for the failure to protect PII. For example, the SEC recently brought suit against the CISO of SolarWinds alleging fraud and internal control failures relating to allegedly known cybersecurity risks and vulnerabilities. The suit stated that SolarWinds and its CISO defrauded investors by overstating the company's cybersecurity practices and understating or failing to disclose known risks.



So, although there are very few laws or regulations that are focused solely on APIs, there are a number of them that mention APIs or drive companies to be compliant. For example, the revised Payment Services Directive (PSD2) in the European Union and the 21st Century Cures Act in the United States are driving transparency requirements for healthcare providers to use APIs. The challenge is that the data involved is both highly regulated and liable to be targeted by cybercriminals. This has led to groups like American National Standards Institute (ANSI) and the International Organization for Standardization and the International Electrotechnical Commission to produce guidelines (ISO/IEC 27001). Emerging regulations like the Payment Card Industry Data Security Standard (PCI DSS) v4.0 are also calling out APIs. On the technical side, the Open Web Application Security Project (OWASP) is a great reference for training. The bottom line is: Build a system that allows for discovery, monitoring, investigations, and remediation and you will be able to map to compliance requirements.

As these trends toward legal action try to raise the bar on cybersecurity programs, transparency and accountability must continue to advance as well. The risks associated with a lack of visibility into a company's API space and a corresponding gap in security posture — because you cannot protect what you cannot see — create potentially serious legal and regulatory concerns.

For more information on the API attack trends in the Asia-Pacific and Japan (APJ) and Europe, Middle East, and Africa (EMEA) regions, please refer to the regional reports found in the next sections.

Akamai

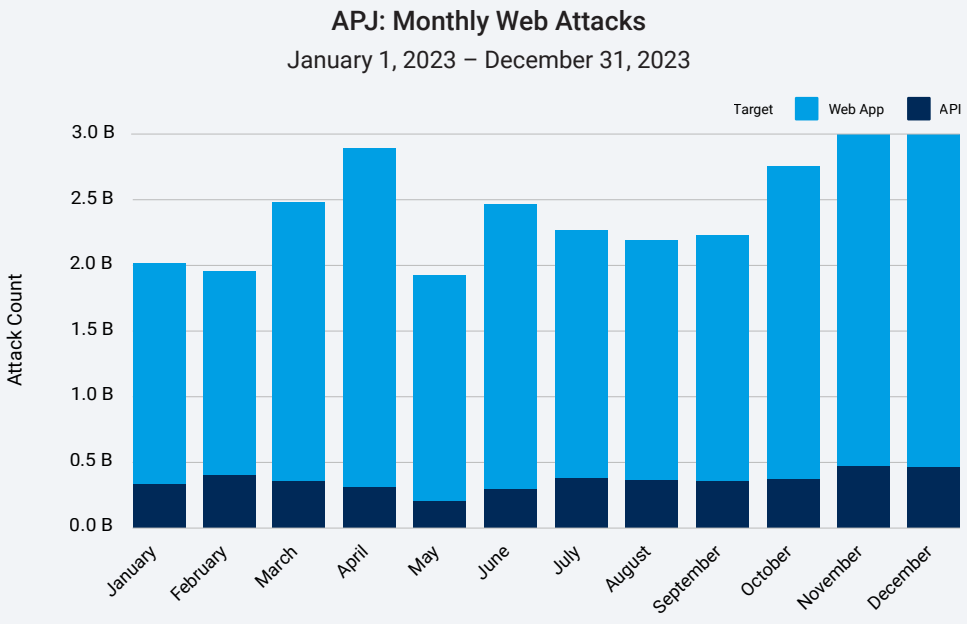
APJ Snapshot

The APJ Snapshot is a companion piece to our larger API security SOTI report, Lurking in the Shadows: Attack Trends Shine Light on API Threats (available in English only). Please refer to that report for detailed descriptions of how adversaries leverage the attack vectors we discuss in this snapshot, recommendations to safeguard your organization, and an explanation of our research methodologies and new dataset.

API attacks notable in APJ

By leveraging a new dataset that specifically tracks API attack traffic, Akamai research revealed that 15.0% of all overall web attacks in the Asia-Pacific and Japan (APJ) region targeted APIs. On a global basis, the APJ region had the third-highest percentage of API attacks behind the Europe, Middle East, and Africa (EMEA) region at 47.5%, and North America at 27.1%.

During the reporting period from January through December 2023, web attacks targeting APIs fluctuated between 11% and 21% on a monthly basis (APJ Figure 1). We may be able to attribute this relatively low percentage of attacks (compared with the percentage of attacks on other regions) in part to the relatively small open API market size versus Europe and North America and thus lower adoption rates by organizations in APJ.

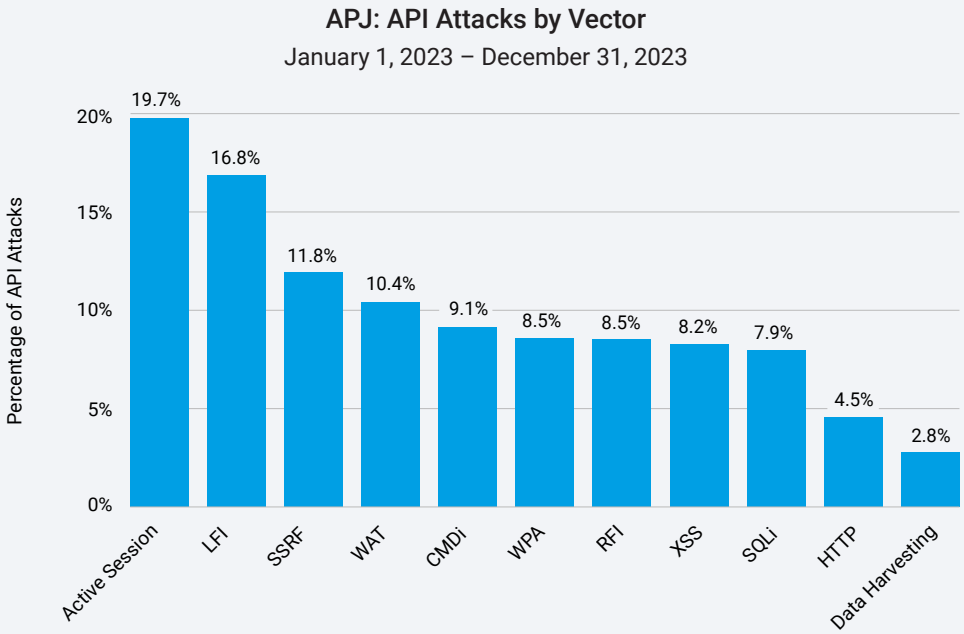


APJ Fig. 1: Attacks targeting APIs averaged 15.0%, even as overall web attacks increased

Within APJ, the areas with the highest percentage of web attacks targeting APIs were South Korea (47.9%), Indonesia (39.6%), Hong Kong SAR (38.7%), Malaysia (26.4%), Japan (23.4%), India (19.0%), Australia (15.6%), Singapore (5.8%), the Philippines (5.5%), and New Zealand (4.8%).

APIs under attack: Traffic analysis

Consistent with previous reports in which we looked at overall web attacks, LFI remains a top attack vector for APIs in APJ. However, Cross-Site Scripting (XSS) and Structured Query Language Injection (SQLi) have moved further down the list as related to API attacks specifically (APJ Figure 2).



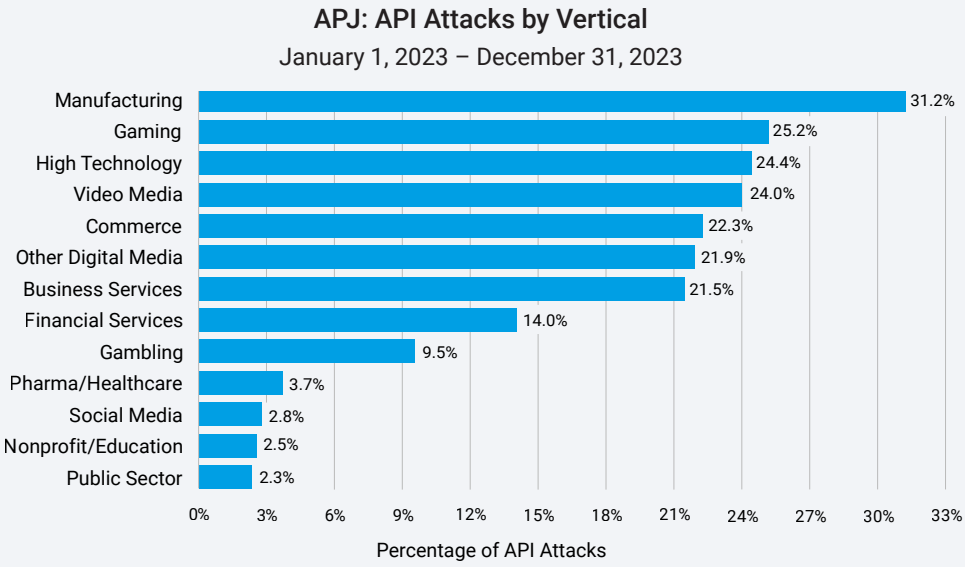
APJ Fig. 2: LFI remains a prevalent attack vector and our new dataset reveals additional favored attack techniques against APIs

The new dataset enables us to surface additional favored API attack vectors. For example, Command injection (CMDi) is a popular technique in API attacks, and Server-Side Request Forgery (SSRF; which we discussed in our 2023 report) is now among the most frequently used vectors. Of note, Active Session indicates suspicious behavior during that session, which results in a temporary block. (See the appendix at the end of the global report for a complete list of attack vector definitions.)

Our research also revealed that bot requests are an area of concern. During the same 12-month reporting period, 40% of the more than two trillion suspicious bot requests were aimed at APIs.

API attacks across industries

During the reporting period, Akamai researchers found that the manufacturing industry had the highest percentage of overall web attacks that targeted APIs at 31.2%, followed by gaming at 25.2%, high tech at 24.4%, video media at 24.0%, and commerce at 22.3% (APJ Figure 3).



APJ Fig. 3: The manufacturing vertical had the highest percentage of API attacks, partially due to the increasing connectivity of this critical infrastructure sector via APIs and the potential for supply chain disruption

APJ Snapshot conclusion

Defending APIs is a clear imperative from a security and risk management perspective. In addition, the existing laws and regulations and emerging reforms to keep cybersecurity legislation apace with the threat landscape also make it imperative to protect APIs.

For example, India is in the process of drafting the Digital India Bill, which will be a major overhaul of the IT Act, starting with the passing of the Digital Personal Data Protection Act in August 2023. The Australian government released the 2023-2030 Australian Cyber Security Strategy on November 23, 2023, with a pillar focused on safe technology and ensuring trust in digital products and software. Additionally, Section 6 of the upcoming Payment Card Industry Data Security Standard (PCI DSS) v4.0 specifically includes new standards on the use of APIs in the development and maintenance of systems and software to reduce the risk of compromise.

Regulators are putting initiatives and policies in place to strengthen cybersecurity standards for APIs, which are increasingly being used to exchange sensitive financial information. Understanding best practices and guidelines is important so that you can integrate APIs into your security program to improve visibility, strengthen defenses, and map to compliance requirements.

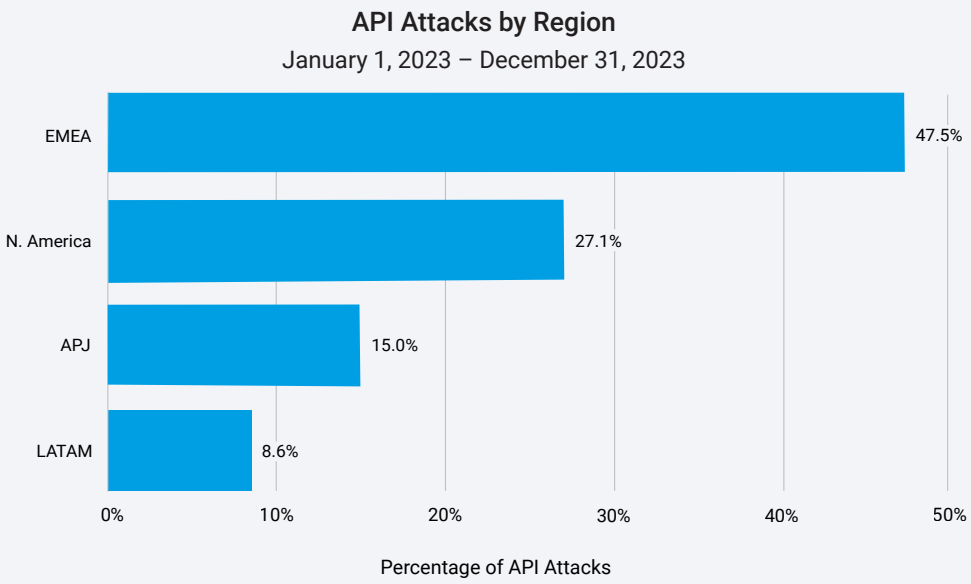
For more information, please refer to the global API security SOTI report, Lurking in the Shadows: Attack Trends Shine Light on API Threats.

EMEA Snapshot

The EMEA Snapshot is a companion piece to our larger API Security SOTI report, Lurking in the Shadows: Attack Trends Shine Light on API Threats (available in English only). Please refer to that report for detailed descriptions of how adversaries leverage the attack vectors we discuss in this snapshot, recommendations to safeguard your organization, and an explanation of our research methodologies and new dataset.

API attacks prevalent in EMEA

By leveraging a new dataset that specifically tracks API attack traffic, Akamai research revealed that the Europe, Middle East, and Africa (EMEA) region has the highest percentage of API attacks on a global basis at 47.5% — by far exceeding the next closest region, North America at 27.1% (EMEA Figure 1). This is based on the total number of web attacks in each region and shows that APIs are in more danger in EMEA than in other regions.

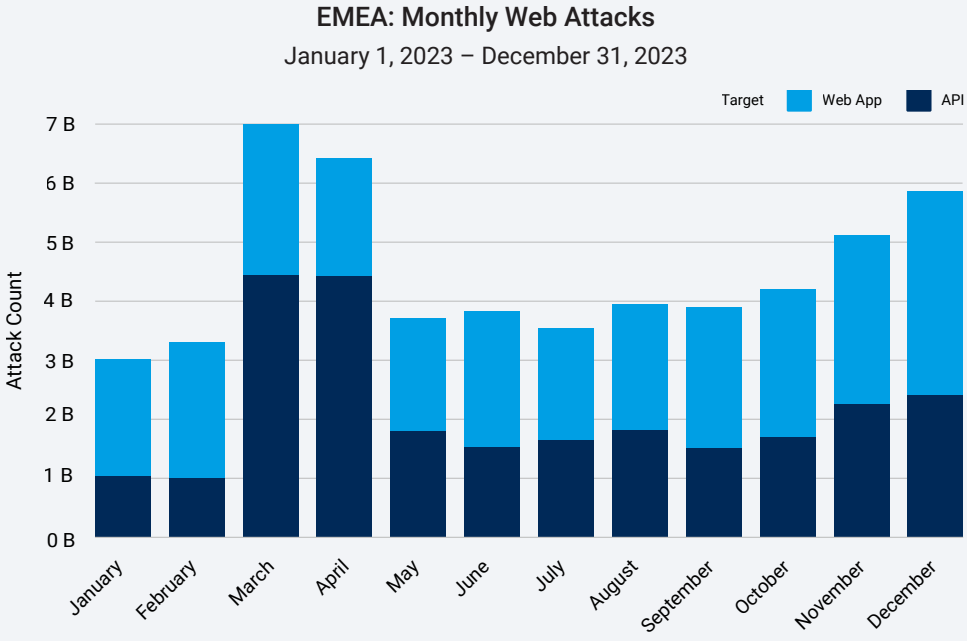


EMEA Fig. 1: Web attacks are significantly more likely to target APIs in EMEA than in any other region

We may be able to attribute this relatively high percentage of attacks in EMEA (when compared with the percentage of attacks on other regions) in part to the relatively large open API market size versus North America and Asia-Pacific, reflecting higher API adoption rates in EMEA, as well as to open banking and the Payment Card Industry Data Security Standard (PCI DSS) v4.0 that are driving the use of APIs and can introduce the security risks discussed in the global report.

Within EMEA, the areas with the highest percentage of web attacks that target APIs are Spain (94.8%), Portugal (84.5%), the Netherlands (71.9%), and Israel (67.1%). This is not to say that the number of web attacks overall is higher in these countries than in others in EMEA — rather, these countries face a much more concentrated risk from API abuse because of attackers' focus on that vector.

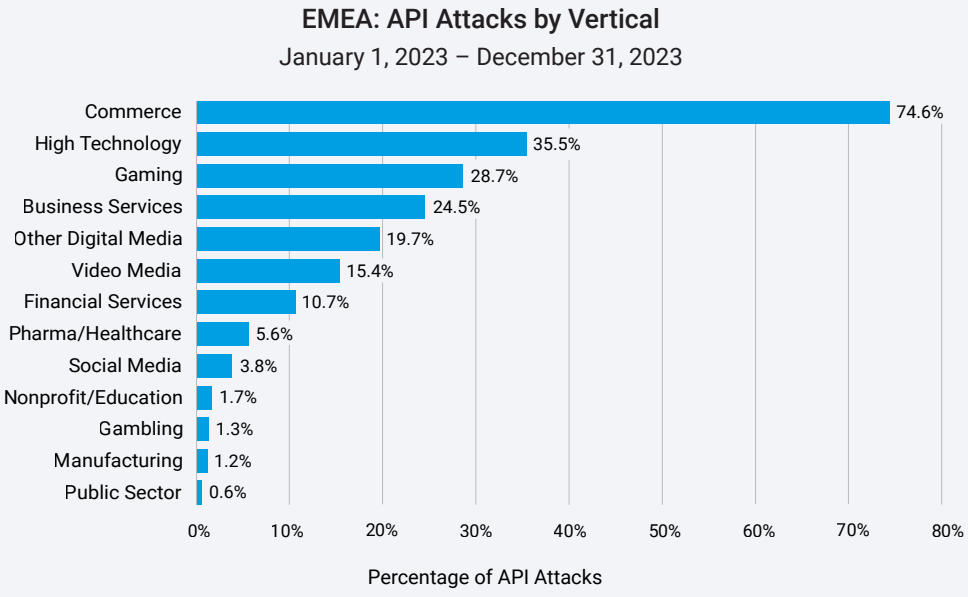
The monthly trends during the reporting period from January through December 2023 show that web attacks that targeted APIs in EMEA increased fairly steadily, starting at 34% in January and rising to 41% by the end of the year (EMEA Figure 2). The exceptions were in March and April when Akamai researchers saw a spike in API attacks as the commerce sector in Spain — a country with an already huge API attack concentration — experienced large-scale, focused attacks. This spike shows how quickly attackers can shift their focus among regions and industries, so it is worth tracking broader trends.



EMEA Fig. 2: With the exception of March and April, when API attacks spiked, API attacks slowly increased throughout 2023, rising to 41% of all attacks by the end of the year

API attacks across industries

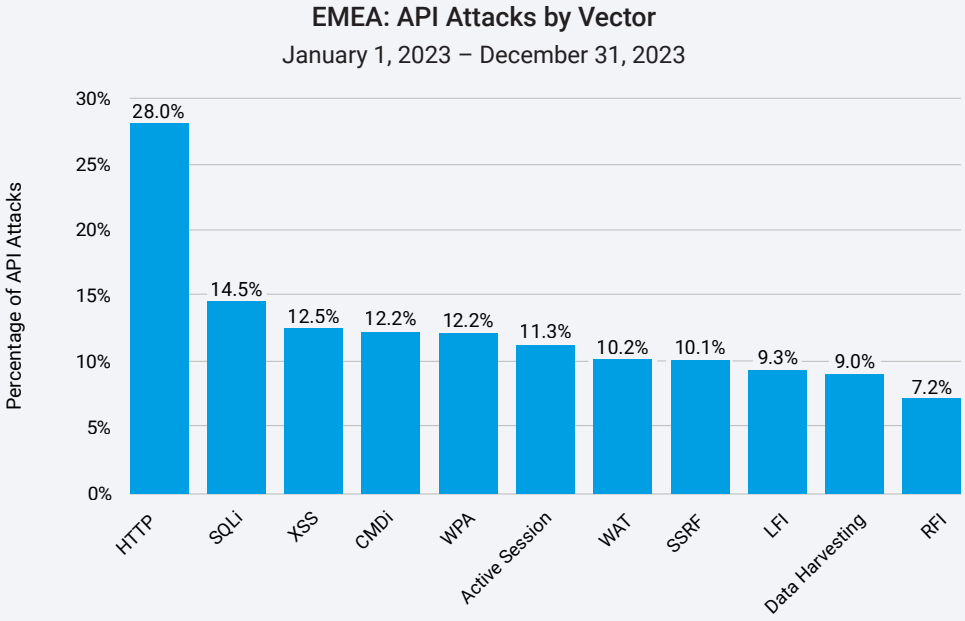
During the reporting period, Akamai researchers found that the commerce industry had the highest percentage of overall web attacks that impacted organizations at 74.6%, which is more than twice the percentage of the next closest industry — high tech at 35.5%. They were followed by gaming at 28.7%, business services at 24.5%, and other digital media at 19.7% (EMEA Figure 3).



EMEA Fig. 3: The commerce vertical had the highest percentage of API attacks, partially due to the complex nature of its ecosystem, its high reliance on APIs, and the valuable data organizations in this sector possess

APIs under attack: Traffic analysis

Consistent with the global trend, HTTP Protocol (HTTP) and Structured Query Language injection (SQLi) have been the predominant ways in which adversaries have targeted APIs in EMEA during the last 12 months, and Local File Inclusion (LFI) has moved further down the list in comparison with its dominance in web application attacks (EMEA Figure 4).



EMEA Fig. 4: HTTP, SQLi, and XSS are the most relevant vectors to API attacks; LFI is less prevalent for API attacks, but still actively used for attacks against web applications

In EMEA, Cross-Site Scripting (XSS) remains a favored technique, even for API attacks, and Command injection (CMDi) is also prevalent. The new dataset enables us to monitor for additional attack vectors in APIs. For example, Server-Side Request Forgery (SSRF; which we discussed in our 2023 report) is now an up-and-coming vector. (See the appendix at the end of the global report for attack vector definitions.)

Our research also revealed that bot requests are an area of concern. During the same 12-month reporting period, 40% of the nearly four trillion suspicious bot requests were aimed at APIs.

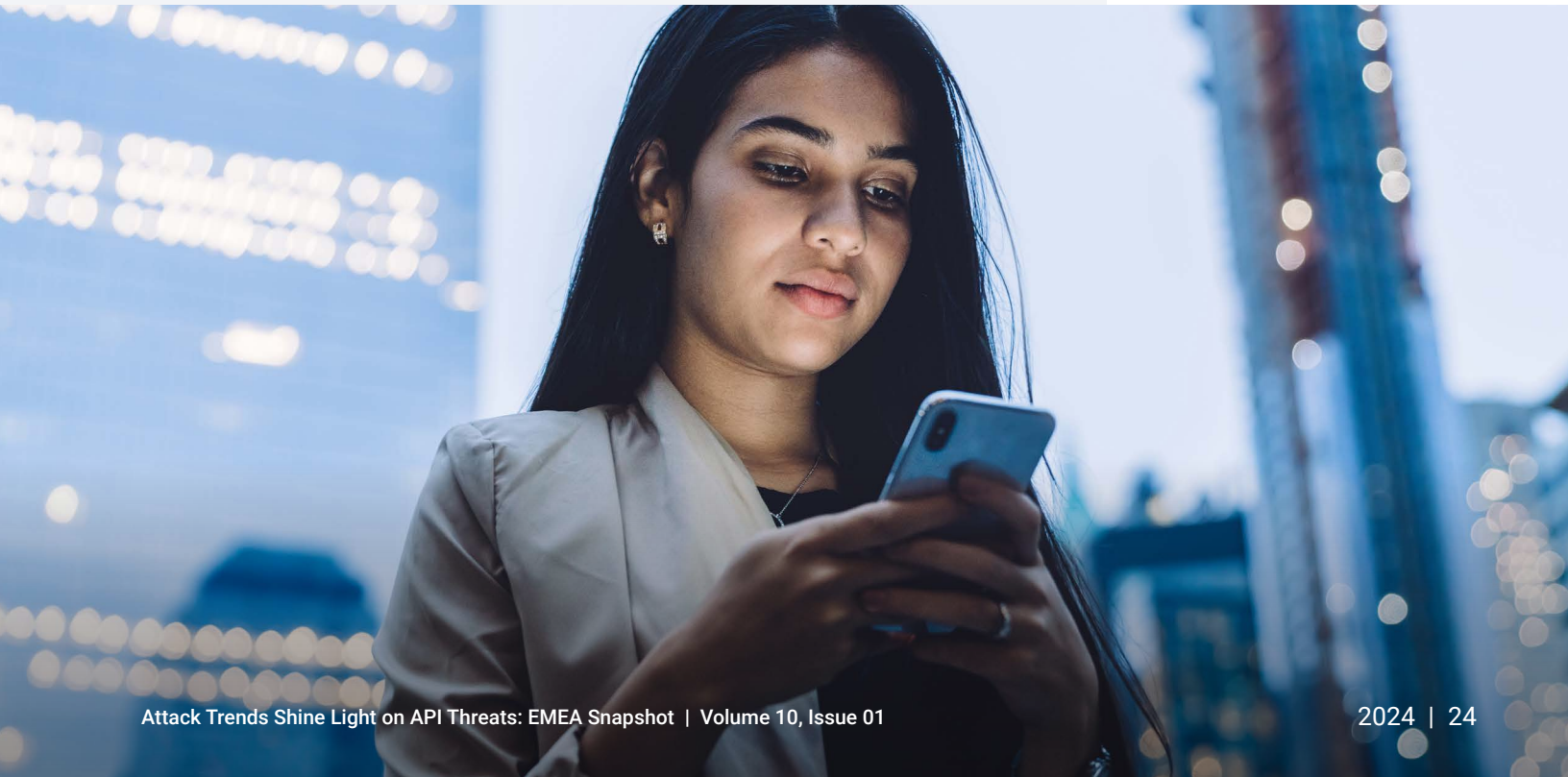
EMEA Snapshot conclusion

Defending APIs is a clear imperative from a security and risk management perspective. In addition, the existing laws and regulations and emerging reforms to keep cybersecurity legislation apace with the threat landscape also make it imperative to protect APIs.

For example, the European Union's Global Data Protection Regulation (GDPR) is focused on the protection of personal data, and APIs are now at the forefront of how this data is used and shared. Additionally, the new Network and Information Security Directive (NIS2) specifically calls for the establishment of a robust API security program. Outside the EU, countries such as Saudi Arabia have introduced data protection laws similar to the GDPR, which create obligations for entities dealing with personal data. Additionally, Section 6 of the upcoming Payment Card Industry Data Security Standard (PCI DSS) v4.0 specifically includes new standards on the use of APIs in the development and maintenance of systems and software to reduce the risk of data compromise.

As regulators put initiatives and policies in place to strengthen cybersecurity standards for APIs, it is important to understand best practices and guidelines so that you can integrate APIs into your security program to improve visibility, strengthen defenses, and map to compliance requirements.

For more information, please refer to the global API security SOTI report, Lurking in the Shadows: Attack Trends Shine Light on API Threats.



Akamai

Improving visibility: A year in the life of an enterprise API estate

At the beginning of the report, we tackled the perils posed by the lack of API visibility and highlighted some of the insights gained by organizations via security alerts. In this section, we will showcase how adopting a strong API security program leads to many different lenses of visibility, including:

- **Discovery** — Visibility into the inventory of APIs inside your organization
- **Risk audit** — Visibility into the risk posture of each discovered API
- **Behavioral detection** — Visibility into normal use versus abnormal abuse to see active threats on each API
- **Investigations and threat hunting** — Visibility into threats lurking inside your API estate found by expert human threat hunters

These lenses are not unique to APIs, but we find that because of the rapid deployment of APIs, many of them don't have the cybersecurity maturity of more established IT infrastructure. The issue is that many APIs have sensitive data that can be exploited. The organizations we work with that apply more sophisticated API security practices to their API footprint often follow a common pattern as they learn more about their API activity and begin to see both posture alerts and runtime alerts.

1. **Shining a light on the shadows**

   "You cannot protect what you cannot see" is an old adage that applies to today's APIs. One of the biggest surprises for many enterprises that increase their visibility into API activity is the number of shadow endpoints that were unknowingly operating in their environment. Security teams are grateful when rogue or zombie APIs are discovered because a light is now shining on a previous dark spot. Typically, the first step on the journey to API security maturity is to discover these shadow APIs systematically and ensure that each is either decommissioned or formally documented and incorporated into the organization's API security controls. This has an immediate impact on reducing the risk of unexpected API abuse and other threats. Typically, we see a large spike of alerts when we deploy API security tools, but then we start to discover the gaps in our process as, over time, more unmanaged or unauthorized APIs turn up.

2. **Getting organized**

   Once shadow APIs are addressed, there's still work to do in rationalizing and organizing the inventory of sanctioned APIs. This includes segmenting by broad categories, such as development, testing, and production, as well as establishing hierarchies that ensure security alerting and analytics have the appropriate context to allow the team to understand the risk associated with the API.

   Documenting each API is the next step toward improved visibility. Documentation allows security teams to react to posture alerts more efficiently, as it brings their alerts into context and aligns them with the way they think about their applications, APIs, and business processes. It is hard to determine suspicious activities until you establish a baseline of activity.

3. **Hardening the API posture**

   The initial wave of posture and runtime alerts that an enterprise receives often informs a set of high-priority changes to its API implementations. For example, security teams will generally look at their most common alert types and identify strategies and priorities to reduce their risk. This includes a combination of correcting flaws in API code, addressing misconfiguration issues, and implementing processes for preventing future vulnerabilities based on the lessons learned. This will lead to prioritizing pen test validation plans and can inform management of the necessary coding best practices to avoid vulnerabilities in the future.

4. **Sharpening threat detection and response**

   While the first three steps generally lead to a general downward trend in the overall number of API security alerts, we typically see occasional spikes as the year progresses. These can be caused by internally driven factors, such as broad changes to business models, the acquisition of new capabilities, or additions to the API footprint that introduce new vulnerabilities or rogue systems. Spikes can also be caused by external factors, including attack attempts from adversaries. The most effective organizations plan for these spikes and engage well-defined response procedures when they occur to bring risk and alert volume down to normal levels. They will also take steps to continuously accelerate the time it takes to respond, investigate, contain, and recover from active API threats. This can require new skills based on the API environments.

5. **Developing a stronger offense**

   As organizations improve their defensive API security measures, the next stage is to complement defensive measures with an offensive approach to API threat discovery and mitigation. This includes establishing a formal API threat hunting discipline and cadence, with the goal of identifying possible threats early — before they escalate into a reactive scenario. This can be challenging to execute since it requires highly specialized talent and an ability to isolate resources from interrupt-driven tasks. For this reason, some enterprises engage specialized third-party service providers, including Akamai, to deliver this important function.

The anonymized Figure 5 illustrates how this pattern unfolded for one of our enterprise customers. In January and February, there was an initial drop-off as the organization took steps to eliminate shadow APIs, get organized, and make some initial improvements to its API security posture. Occasional spikes in posture alerts were observed as changes to the API estate were made. Through their API visibility, the customer resolved possible vulnerabilities quickly rather than allow them to persist.

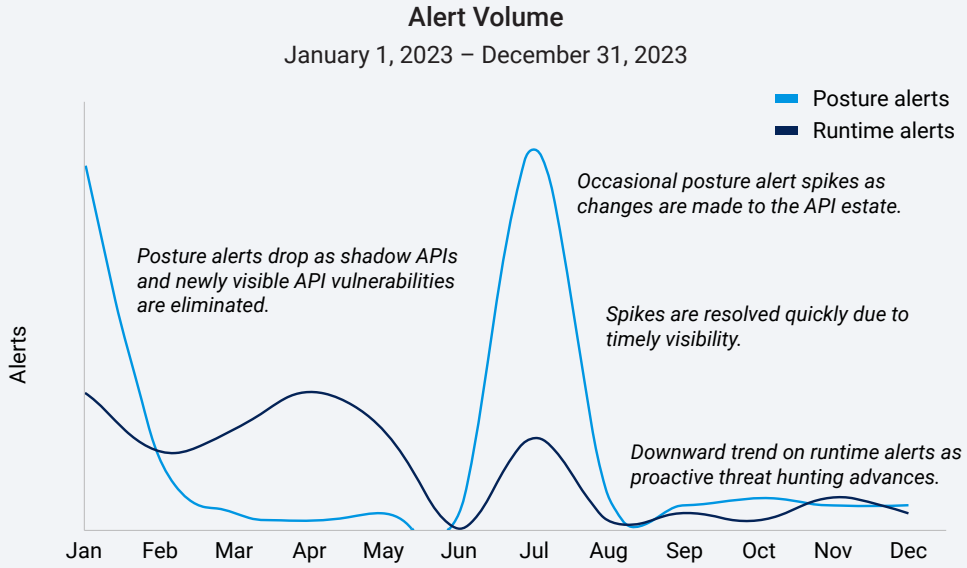


Fig. 5: The ebb and flow of the API estate with a significant decrease in alerts once organizations have visibility of their API landscape

Runtime alert volume is less predictable, since it is driven by external factors. However, a general drop-off can be observed as the organization made improvements to its overall security posture and ramped up its proactive threat hunting capabilities.

The bottom line is that APIs cannot remain the responsibility of just the IT team. This can require new tools and skills depending on the potential risk exposure of the APIs determined by the sensitivity of the data. As new tools are deployed, API defense is also an area in which leaders should be looking at skill set and staffing needs and, in many cases, reallocating engineering hours or moving to managed services. Overall, the level of effort for cybersecurity to support APIs needs to be tracked and analyzed for inefficiencies.

Defending the API universe

APIs are foundational for many of the new capabilities that companies are building — but, in most cases, the security of APIs is either not considered early enough in the planning process or not able to keep up with the rapid deployment of new technology. So, as we look at how to approach building an effective security program, we'll quote our favorite cyber gray beard, Bruce Schneier: "You can't defend. You can't prevent. The only thing you can do is detect and respond." That philosophy should drive us to focus on establishing better situational awareness. We should make sure that we have all APIs integrated into our security program, and that we actively monitor for attacks, vulnerabilities, and abuse. Our pen test and red teams should be testing the posture for authentication and exposed data, as well as for runtime issues like JSON properties and scraping. The tests should be built as purple team exercises in which the security information and event management team and the security operations center validate that they detect the attacks and have current processes in place to mitigate the impacts. The case studies we reviewed in this report (loyalty fraud and carding attack, etc.) are great templates to use for test plans.

We should use the OWASP guidance on coding practices to prevent the most common attacks. These proactive controls and the calls to action to develop strong processes around discovery, hardening, detection, and response are great bases around which to build quarterly action plans.

We must also consider compliance. Although there are not many API laws/regulations today, there are a number of best practices and standards we should take advantage of to ensure we are doing the right thing to protect our customers. Current regulations like the GDPR include APIs, new standards like the PCI DSS v4.0 call out APIs, and groups like ANSI are publishing guidelines.

This report was based on both the threat traffic we defended against and the best practices we learned from our customers. Furthermore, as we engage with customers, we continue to hear about the value of integration of security controls on fewer platforms, the need for flexible staffing solutions to meet transformation goals, and the importance of visibility to make decisions and evaluate performance. We hope the data from this report provides insights and visibility to help you update your program and develop best practices to protect your customers.

Stay plugged in to our latest research by checking out our security research hub.

Methodology

Web application and bot attacks

This data describes application-layer alerts on traffic seen through our web application firewall (WAF) and bot management tool. The web application attack alerts are triggered when we detect a malicious payload within a request to a protected website, application, or API. The bot alerts are triggered when we detect a bot payload within a request to a protected website, application, or API. These bot alerts can be triggered by both malicious and benign bots. The alerts do not indicate the successfulness of an attack. Although these products allow a high level of customization, we collected the data presented here in a manner that does not consider custom configurations of the protected properties. The data was drawn from an internal tool for analysis of security events detected on Akamai Connected Cloud, a global network of 4,000+ edge points of presence in 130+ countries. Our security teams use this data, measured in petabytes per month, to research attacks, flag malicious behavior, and feed additional intelligence into Akamai's solutions.

The data in this report covered the 12-month period from January 1, 2023, through December 31, 2023.

2024 data update

We are happy to announce some updates to our datasets for our 10th anniversary! Our web application and bot attack datasets have received a few upgrades. The collection method for each has been transformed, streamlined, and optimized. The range and depth of our insights have been broadened. Classifications for additional attack vectors, such as SSRF, have been added. Identification of attacks targeting API endpoints have also been added to each dataset. We enjoyed highlighting some of these new improvements in this report, and we are looking forward to continuing to share these updates throughout the year — and beyond — as we celebrate this State of the Internet/Security milestone with our readers.

Akamai API Security insights

Special thanks to our Akamai API Security Solution Engineering team for their contributions of real-world insights with a look into API risks and their potential impacts based on our API Security alerts.

Akamai

Appendix

Attack Vector	Definition
Active Session	Attack traffic has been recently flagged for the client and repeated requests will be blocked for the duration of the session
Command injection (CMDi)	An adversary injects new items into an existing command to modify the interpretation away from what was intended and toward actions of their choosing
Cross-Site Scripting (XSS)	An adversary embeds malicious scripts in content so that the target software executes the scripts with the users' privilege levels when the content is served to web browsers
Data Harvesting	An adversary exploits weaknesses in the design or configuration of the target and its communications to get it to reveal more information than intended; this is often executed to gather data in preparation for another type of attack, but gaining access to the information may also be the end goal of the adversary
HTTP Protocol (HTTP)	An adversary takes advantage of weaknesses in the protocol by which a client and server are communicating to perform unexpected actions; exploiting different types of protocols can lead to different end goals of attacks
Local File Inclusion (LFI)	An attacker manipulates inputs to the target software to gain access to, and perhaps modify, areas of the file system that were not intended to be accessible

Attack Vector	Definition
Remote File Inclusion (RFI)	The adversary loads and executes remote arbitrary code, subsequently hijacking the targeted application and forcing it to execute their own instructions
Server-Side Request Forgery (SSRF)	The attacker abuses the functionality of the server to read or update internal resources
Structured Query Language injection (SQLi)	An attacker crafts input strings so that when the target software intends to construct SQL statements based on user input, the resulting SQL statement instead performs actions the attacker intended; successful injections can cause information disclosure as well as the ability to add or modify data in the database
Web Attack Tool (WAT)	An adversary actively probes the target in a manner that is designed to solicit information that could be leveraged for malicious purposes; as a result of these probes, the adversary is able to obtain information from the target that aids the attacker in making inferences about its security, configuration, or potential vulnerabilities
Web Platform Attack (WPA)	An attack against a software platform (cloud, web, or application layer) that is not categorized in another attack group

Akamai

Credits

Editorial and writing

Badette Tribbey – Editor in Chief
Charlotte Pelliccia – Lead Writer (regionals)

Editorial contributors

James Casey
Edward Roberts
Steve Winterfeld

Review and subject matter contribution

Tom Emmons
Reuben Koh
Rob Lester
Richard Meeus
Abigail Ojeda
Menachem Perlman
Yariv Shivek

Data analysis

Chelsea Tuttle

Marketing and publishing

Georgina Morales Hampe
Emily Spinks

More State of the Internet/Security

Read back issues and watch for upcoming releases of Akamai's acclaimed State of the Internet/Security reports. **akamai.com/soti**

More Akamai threat research

Stay updated with the latest threat intelligence analyses, security reports, and cybersecurity research. **akamai.com/security-research**

Access data from this report

View high-quality versions of the graphs and charts referenced in this report. These images are free to use and reference, provided Akamai is duly credited as a source and the Akamai logo is retained. **akamai.com/sotidata**

More on Akamai solutions

To learn more information on Akamai solutions for API attacks, visit our **App and API Security page**.

Akamai

Akamai protects your customer experience, workforce, systems, and data by helping to embed security into everything you create — anywhere you build it and everywhere you deliver it. Our platform's visibility into global threats helps us adapt and evolve your security posture — to enable Zero Trust, stop ransomware, secure apps and APIs, or fight off DDoS attacks — giving you the confidence to continually innovate, expand, and transform what's possible. Learn more about Akamai's cloud computing, security, and content delivery solutions at akamai.com and akamai.com/blog, or follow Akamai Technologies on X, formerly known as Twitter, and LinkedIn. Published 03/24.