



The State of **API Security** 2024.



Table of Contents.

- **Introduction**
 - **Contributing Authors**
 - **About this Publication**
 - **Research Methodology**
 - **Disclaimer**
- **Executive Summary**
 - **Key Findings**
 - **Major Events**
 - **Securing Your APIs**
- **APIs on the Rise**
 - **APIs Are Everywhere**
 - **Driving Value**
 - **Low-Hanging Fruit for Attackers**
- **API Threat Landscape**
 - **OWASP API Security Top 10 Update**
 - **AI & API Calling Capabilities**
 - **Software Supply Chain Risks**
 - **Case Studies**
 - **Public API Breaches**
 - **Fortune 500 API Analysis**
 - **GitHub Repository Scanning**
 - **FireTail Platform Data**
- **API Attack Vectors**
 - **High-Impact Attack Vectors**
 - **Other Notable Attack Vectors**
 - **Failure of Network Security for API**
- **Effective API Security Strategies**
 - **6 Pillars of API Security**
 - **Context is King**
 - **Code to Cloud**

Introduction.



Contributing Authors.

Meet the team behind this report:

Jeremy Snyder

The co-founder and CEO at FireTail, Jeremy started his career with 13 years in cyber and IT operations. Jeremy has an MBA from Mason, a BA in computational linguistics from UNC, and has completed additional studies in Finland at Aalto University. Jeremy speaks 5 languages and has lived in 5 countries.



Riley Priddle

The co-founder and CTO at FireTail, Riley has a long background in cloud computing and cloud security, having run cloud initiatives and security programs at companies such as HP, SkyTV, Otro and Nekta. Originally from New Zealand, Riley has a BS in Information Technology from the Wellington Institute of Technology.



Timo Ruppell

Timo is VP of Product at FireTail and a bona fide expert when it comes to API security. A former researcher in theoretical high energy physics, Timo transitioned from simple stuff like quantum gravity and supersymmetry to really hard stuff like writing maintainable Javascript and developing innovative approaches to API security.



Viktor Markopoulos

An information security consultant and independent researcher for FireTail, Viktor specializes in web application and API pentesting. Originally from Greece, Viktor is renowned for uncovering massive data breaches and security vulnerabilities. Viktor is passionate about what he does and advocates for stronger cybersecurity measures globally.



About this Publication.

This document is based on research conducted by the FireTail team, employees of FireTail Inc. and FireTail International Limited, as well as third-party publications. Third-party research and analysis is quoted and cited correspondingly.

Research Methodology.

FireTail uses responsible research methodologies. Please see our disclosure policies for both reporting to and reporting by FireTail. All customer data used in this report is anonymized and aggregated, under the FireTail terms of service. Direct research involved the testing of publicly accessible APIs and only completing code or design analysis on these resources. No data was exfiltrated and vulnerabilities were reported in line with our responsible disclosure policies.

Disclaimer.

Research in this document is based on analysis of both third party publications and research and analysis conducted by the FireTail team. FireTail is not responsible for the content of any external sources, nor the quality or completeness thereof. Research is based on the following data sources:

- **Publicly disclosed information, which may not always include full details on a data breach.**
- **A sampling of publicly available APIs.**
- **Analysis of anonymized and aggregated FireTail customer data.**

Except for third-party publications or documents hyperlinked from this publication, all content herein is © 2024 FireTail Inc. and its subsidiaries. All rights reserved.



Executive Summary.



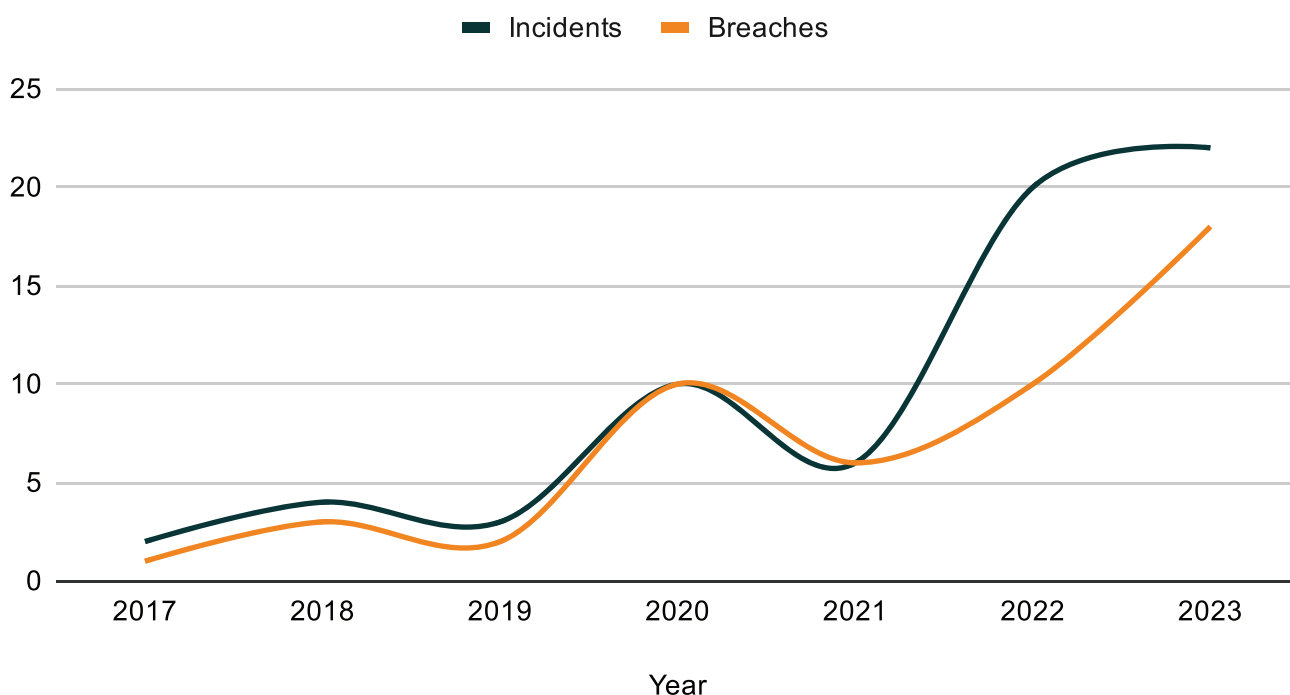
APIs are everywhere and their rise continues. In the last 12 months, the pace of adoption has accelerated. APIs are fundamental to microservice-based architectures, containerization and the proliferation of AI. As a result, the API attack surface has grown dramatically. APIs enable innovation and drive enormous value, yet they remain low-hanging fruit for attackers.

Key Findings.

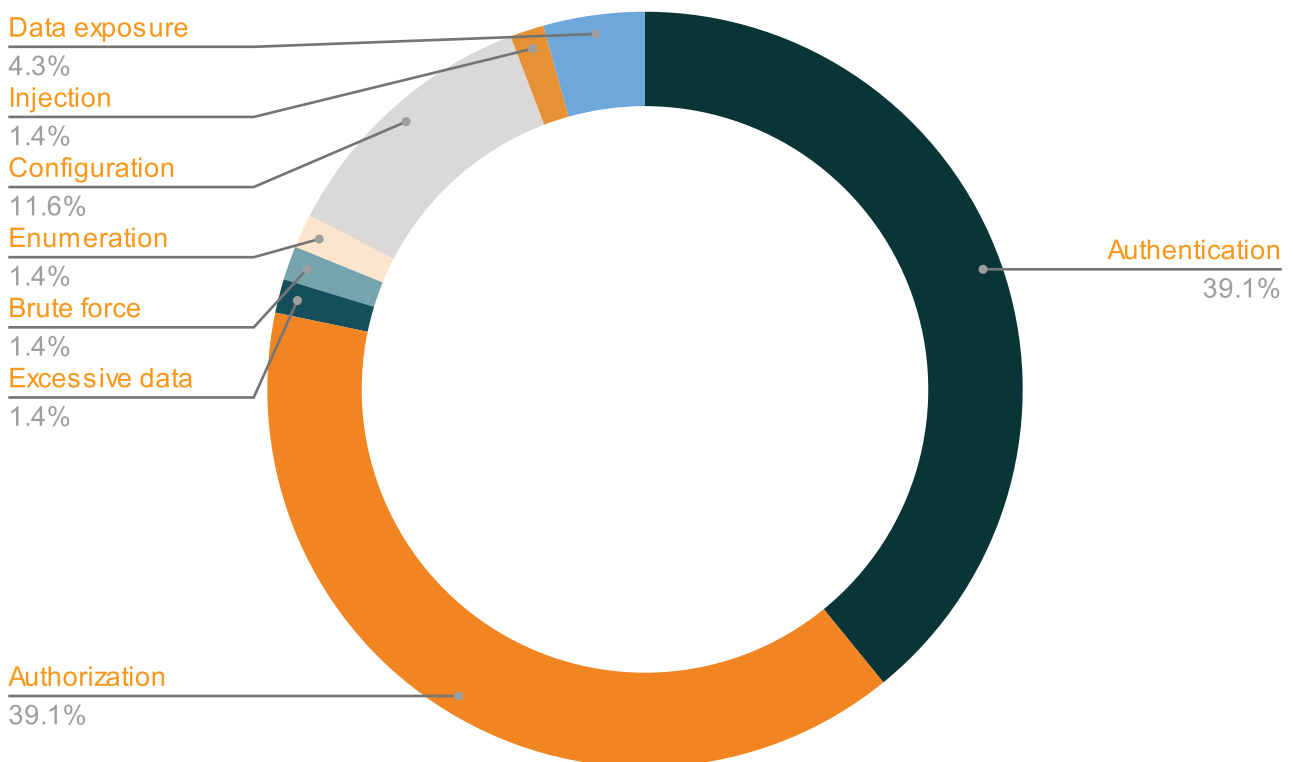
Direct research into the scale of the API threat landscape and the nature of successful API breaches shows that attackers find themselves in an increasingly target-rich environment and that the pace of attacks is accelerating significantly.

The report shows that:

- **API Data Breaches Up 80%:** The volume of breaches where records were confirmed to have been compromised grew **80%** year on year. The compound annual growth rate for breaches from 2017 to 2023 stands at **61.87%** and for incidents stands at **49.13%**.



- **1.6 Billion Records Exposed:** 2023 saw **175M** records breached, up **214%** on 2022. In total, since 2017 the 50 breaches recorded on FireTail's API data breach tracker show **1,623,978,957** records exposed in the 7 year period. The average number of records exposed per breach is greater than **32M**.
- **158,336 Issues Identified:** Across the **206** Fortune 500 APIs, our researchers discovered more than **158K** issues, an average of **769** per API.
- **Authentication and Authorization Still Dominate:** the top two categories of primary attack vector are still authorization and authentication in terms of both number of breaches and the volume of records breached. **78.2%** of all incidents relied on AuthZ or AuthN issues as a primary attack vector.



Major Events.

The previous year in API security has been shaped by three key events:



- 1. OWASP Top 10 API Security Risks Update:** the long overdue refresh of the de facto standard in API security frameworks brought welcome changes but limitations persist. The need for clear controls rather than a list of risks is more pronounced than ever.
- 2. AI & API Calling Capabilities:** the expansion of API calling capabilities introduced by OpenAI in November 2023 radically lowered the bar for attackers. Now everyone, everywhere, regardless of expertise will have the ability to prod and probe APIs across the globe, at pace and at scale. This will be a game-changer for those charged with protecting APIs.
- 3. Software Supply Chain Risks:** One of the most significant changes in API security over the last year is the rise of API breaches in third-party, COTS (commercial off the shelf software) packages. While the threat has existed for many years, in 2023 it came to the fore. Examples such as Ivanti, FortiSIEM and MOVEit, which was perhaps the largest-scale cybersecurity incident of 2023, put API risks in the software supply chain center-stage.

Securing Your APIs.

While the research clearly shows that both the size of the API attack surface and the volume of attacks are steadily increasing, the nature of these attacks is largely unchanged.

To achieve robust API security, organizations must adopt a multi-pronged and continuous approach:

- **6 Pillars:** Establish strong foundations through discovery, visibility, observability, assessments, audits and enforcement.
- **Contextual Awareness:** Inspect payloads and deeply understand API behavior for accurate threat recognition.
- **Code to Cloud:** Embed security into development and adapt protection measures to each environment across the API's life cycle.

APIs on the Rise.



The volume of API communication is ever-increasing, as is the volume of sensitive data flowing over those APIs.

”

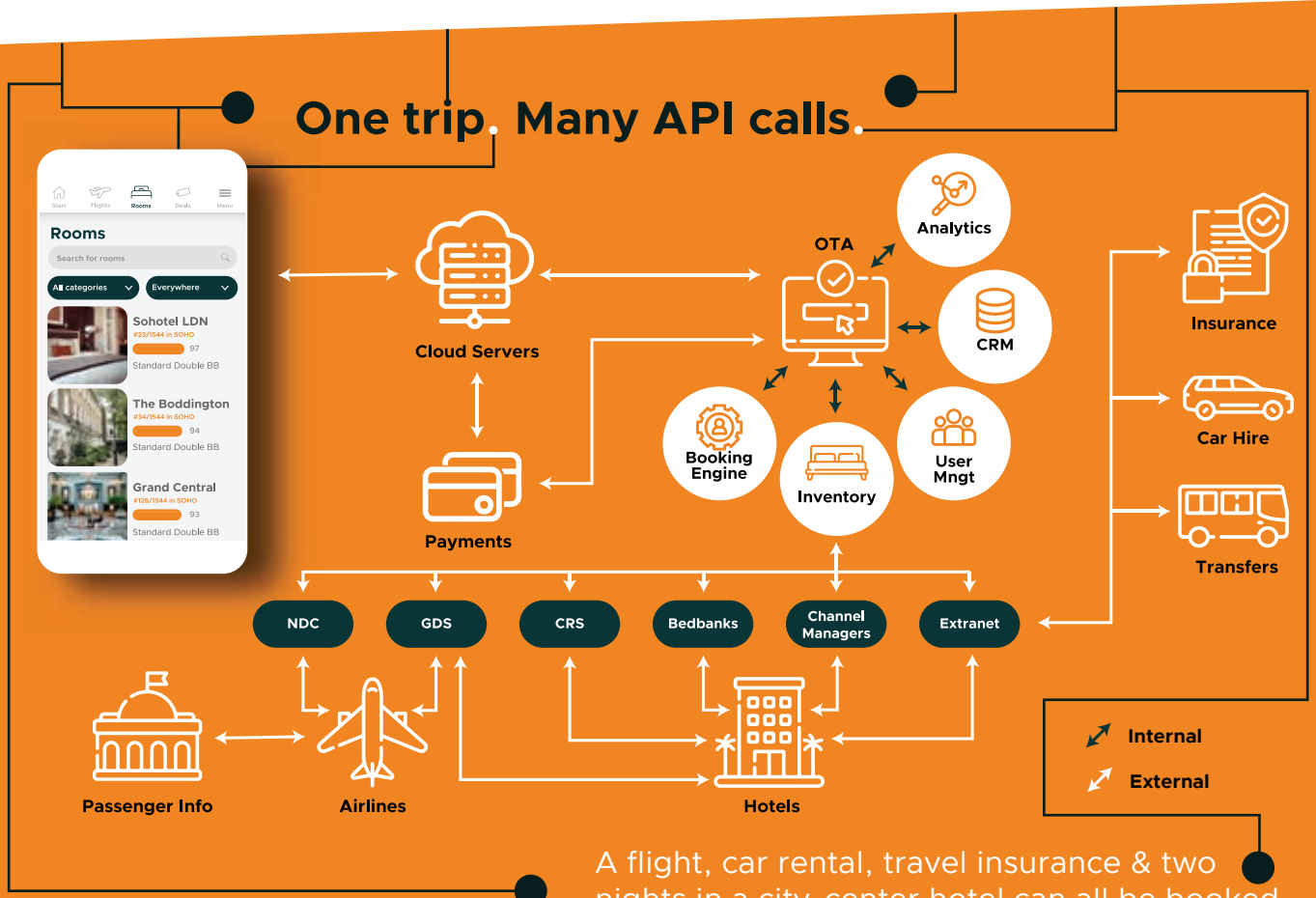
APIs on the Rise.

In this section, we look at the continued proliferation of APIs, their importance in modern technology, their capacity for value creation and what makes them an attractive target for attackers.

- **APIs Are Everywhere**
- **Driving Value**
- **Low-Hanging Fruit for Attackers**

APIs Are Everywhere.

The proliferation of APIs continues. Currently, over 83%¹ of all web requests are API calls, a figure set to rise steadily and it's easy to see why. Consider the itinerary of a simple trip:



A flight, car rental, travel insurance & two nights in a city-center hotel can all be booked in just a few clicks. But this process relies on a vast array of interconnected systems and results in hundreds of API calls.

As more and more commerce moves to the cloud and organizations everywhere adopt containerized and microservice-driven architectures, the importance of APIs grows further. And not just in travel. Whether ordering out, hiring a cab or connecting with friends, digital interactions and transactions across every industry increasingly rely on a complex array of interconnected systems and a dizzying symphony of connections made possible by APIs.

Driving Value.

The strategic significance of APIs cannot be overstated. They are conduits for external value creation, enabling innovation and efficiency.

Imagine that you run a food delivery app. If you didn't have or engage with third-party APIs, you'd have to build every bit of that value chain.

From payment processing to logistics coordination, vertical integration would be mandatory. That slows down the development and deployment of the service, while increasing costs, risks, as well as regulatory and security requirements. Instead, by leveraging APIs and finding purpose-based partners for specific needs, companies can launch quicker, iterate over their learnings and create new products and services for their customers quickly and cost-effectively.

“ APIs allow businesses to monetize data, forge profitable partnerships and open new pathways for innovation & growth. ”

The value creation has been proven with rigorous studies - companies that adopted external APIs created \$8.4B in additional market cap value over a 20-month period, relative to their competitors who did not create. This equated to a 38% market cap gain over a longer period of time (16 years).³

By leveraging APIs, organizations unlock a spectrum of benefits:

Efficiency gains: APIs streamline operations by enabling seamless integration with third-party services, sparing companies the need for extensive vertical integration.

Agility: Rapid deployment of services and products becomes feasible through API-based partnerships, accelerating time-to-market and fostering iterative development.

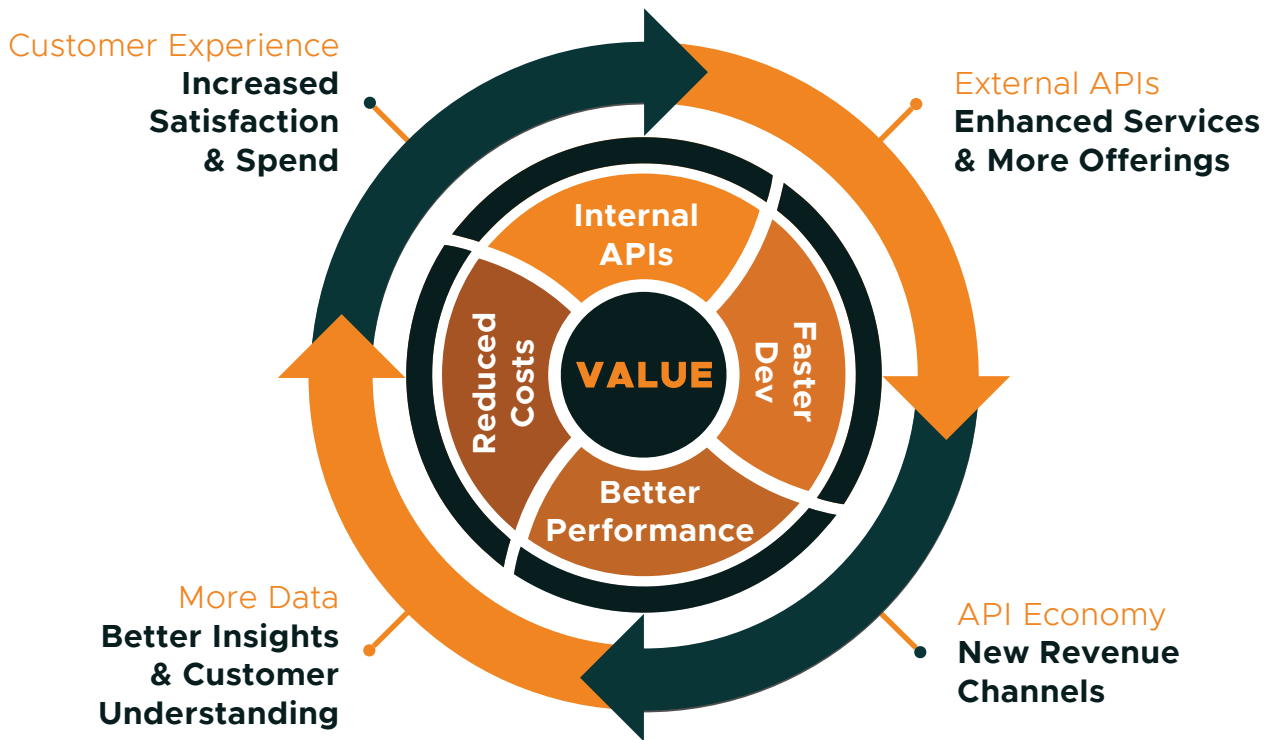
Market capitalization: Studies demonstrate that companies embracing external APIs witness significant market cap growth, outperforming competitors who eschew API integration.



²<https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/what-it-really-takes-to-capture-the-value-of-apis>

³<https://www.youtube.com/watch?v=nynBDfk8Fi4>

In essence, APIs serve as linchpins in the creation of a digital flywheel of value generation.



Low-Hanging Fruit for Attackers.

The openness inherent in API ecosystems exposes organizations to myriad risks. Transmitting data to third parties entails vulnerabilities ranging from technical encryption lapses to contractual breaches. Studies underscore the heightened risk landscape:

API-centric data breaches: Organizations opening external APIs witness a 13.5%⁴ uptick in the occurrence of data breaches, underscoring the vulnerabilities inherent in API integration.

Attack surface expansion: Estimates suggest that APIs represent 40% to 90%⁵ of the attack surface for web-based applications, making them prime targets for cyber assailants.

Elevated threat landscape: API-based attacks have surged and are expected to see a cumulative growth rate of 996%⁶ over the course of this decade.

In the next section, we take a closer look at the API threat landscape.

API Threat Landscape.



API activity is the biggest blind spot for many security teams.



API Threat Landscape.

In this section, we examine the biggest developments in API security across the last 12 months; the update to the OWASP API Security Top 10, OpenAI's massive expansion of API calling capabilities and the acceleration of the supply chain risk in 2023. We also share direct research into public API breaches, an analysis of Fortune 500 APIs, a review of FireTail platform data and recent case studies in order to build a comprehensive picture of the API threat landscape in 2024.

- **OWASP API Security Top 10 Update**
- **AI & API Calling Capabilities**
- **Software Supply Chain Risks**
- **Case Studies**
- **Public API Breaches**
- **Fortune 500 API Analysis**
- **GitHub Repository Scanning**
- **FireTail Platform Data**



OWASP API Security Top 10 Update.

Through the collective efforts of security experts, developers and enthusiasts, OWASP provides free resources, tools, and best practices to help organizations enhance their security posture and protect against emerging cyber threats.

In 2019, having recognized critical issues related to the security of APIs, OWASP launched the API Security Top 10 to raise awareness about the most prevalent API security risks and provide guidance on mitigating them effectively.

The OWASP API Security Top 10 is a comprehensive document that outlines the ten most critical security risks facing APIs. The list is curated based on real-world data, expert insights, and input from the cybersecurity community. It serves as a valuable resource for developers, security professionals, and organizations, assisting them in prioritizing their security efforts and ensuring the safe design and implementation of APIs.

The Top 10 quickly became the de facto standard in API security frameworks. Development, however, was slow, with no significant updates to the Top 10 list in more than 4 years. Then in June 2023, OWASP announced an update to the framework. This was big news in the world of API security.

The new OWASP Top 10 aligns much better with the problems that affect API owners in the real world. Authorization and Authentication unsurprisingly still take the two top spots. Three of the new Top 10 weren't present in any form in the previous list; 06, 07 & 10. They reflect the rising importance of APIs for organizations and the growing interdependence of software and business logic.

While this update is welcome and addresses some of the criticisms that were leveled at the previous Top 10, it is important to remember that this is a list of risks and not a set of prescriptive controls. It requires a deep understanding of the risks and an ability to translate what they mean for each specific organization and tech stack.

Below is a side-by-side comparison showing everything that changed on the Top 10 between 2019 and 2023.



The table below gives a clear breakdown of the differences between the OWASP API Security Top 10 from 2019 and 2023⁷.

OWASP Number	2019	Changes	2023	Status
1	Broken Object Level Authorization	No Change	Broken Object Level Authorization	No Change
2	Broken User Authentication	Update	Broken Authentication	Update: Now includes exploits of non-user API endpoints, microservices, AuthN best practices updates
3	Excessive Data Exposure	Combined with Mass Assignment to form BOPLA	Broken Object Property Level Authorization	Combination of Excessive Data Exposure & Mass Assignment
4	Lack of Resources & Rate Limiting	Update	Unrestricted Resource Consumption	Title Change reflecting outcome of vulnerability
5	Broken Function Level Authorization	No Change	Broken Function Level Authorization	No Change
6	Mass Assignment	Combined with Excessive Data Exposure to form BOPLA	Unrestricted Access to Sensitive Business Flows	NEW - Automated threats to APIs are becoming more common
7	Security Misconfiguration	Priority Reduced	Server Side Request Forgery	NEW - Reflects trend of applications accepting more user inputs
8	Injection	Removed	Security Misconfiguration	Priority Reduced
9	Improper Assets Management	Update	Improper Inventory Management	Title Change reflecting changes in IT infrastructure mgmt vocabulary
10	Insufficient Logging & Monitoring	Removed	Unsafe Consumption of APIs	NEW - Reflects trend of interdependence of APIs linking services

AI & API Calling Capabilities.

Another major development in API security over the previous 12 months came in November 2023. OpenAI announced a massive expansion of API calling capabilities as part of their 'CustomGPT' and 'Assistant API' initiatives.⁸

The exponential growth of AI, particularly Large Language Model (LLM) AI, has already fueled a surge in API consumption. AI's reliance on diverse and vast datasets for learning processes necessitates seamless data integration, normally enabled by APIs.

Furthermore, organizations integrating third-party AI models into their operations depend on APIs for user interaction with the models, increasing the volume of API calls, as well as the number of people using APIs for the first time.

Now that OpenAI is giving users everywhere, and of any technical ability, the power to call APIs from CustomGPTs and via the front-end Assistant API, that growth curve is only going to get steeper.

This expansion of API calling capabilities is great news for businesses and the economy. It will allow more people than ever to create and innovate, bringing together different systems to create powerful solutions that will undoubtedly improve all of our lives. It will remove engineering bottlenecks associated with API development, deployment and management. On the face of things, it's revolutionary and widely beneficial. However, the security implications need to be understood.

APIs are already the number one attack surface. In 2021, IBM X-force⁹ reported that more than two thirds of breaches involved the exploitation of API vulnerabilities.



The ability of AI to call APIs only exacerbates the problem. Now, OpenAI has opened up that ability to everyone. Previously, attackers needed a level of knowledge, sophistication and perseverance in order to successfully find, understand and exploit API vulnerabilities. Now everyone, everywhere, regardless of expertise will have the ability to prod and probe APIs across the globe, at pace and at scale. This will be a game-changer for those charged with protecting APIs.

AI makes it so much cheaper and more efficient to stage attacks. The pool of people with the technical ability to successfully breach an API has just grown exponentially. That means a ton of hitherto unattacked sites and systems will start to see increased attempts, all day, every day and everywhere. And that's in addition to the current norm - in our own testing labs, we see our APIs probed within 5 minutes of going online. The normal calculus used by attackers which weighs up the possibility of a payout against the time and cost of conducting an attack has been turned on its head. Attacks are set to explode. Get ready.

APIs are the new battleground and AI is about to supercharge the arms race.



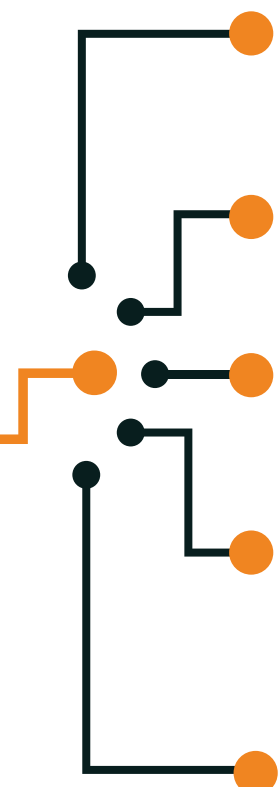
Software Supply Chain Risks.

One of the most significant changes in API security over the last year is the rise of API breaches in third-party, COTS (commercial off the shelf software) packages. **Often organizations become ‘unwitting API providers,’ failing to realize that packages and products they use expose APIs, in addition to web or other interface points.**

In fact, the largest-scale cybersecurity incident of 2023 - **MOVEit** file transfer package from Progress Software - included APIs as a key part of the attack path. Two incidents related to FortiSIEM from Fortinet and Ivanti’s Mobile EPM served as further example. We will cover all of these incidents in more detail below.

These API vulnerabilities highlight the interconnectedness of modern business ecosystems. Organizations downstream from compromised APIs are exposed to heightened risks of data breaches, unauthorized access, and exploitation of sensitive information.

5 reasons that API security is critical to supply chain security¹⁰:

- 
- 1. APIs are essential to the software supply chain:** Developing APIs relies on third-party packages, making supply chain security vulnerable. Continuous monitoring, updates and testing are crucial.
 - 2. APIs handle sensitive data:** Insecure data transmission through APIs can disrupt downstream services. Securing API data integrity and confidentiality is vital.
 - 3. APIs expand the risk landscape:** APIs grant access to functionalities and data, increasing the risk landscape.
 - 4. APIs can lead to authentication and authorization issues:** Unauthorized API access enables attackers to compromise data, bypass security controls, and execute malicious actions, posing supply chain risks.
 - 5. API vulnerabilities can hide supply chain attacks:** Exploiting API vulnerabilities allows attackers to compromise data integrity and confidentiality, potentially masquerading as valid endpoints. Managing APIs and addressing unknown or rogue ones is crucial for supply chain security.



FortiSIEM

Case in Point 1¹¹

In 2024, two critical vulnerabilities were discovered within FortiSIEM, a widely-deployed Security Information and Event Management (SIEM) solution. Designated as CVE-2024-23108 and CVE-2024-23109, these vulnerabilities posed a significant threat, enabling remote code execution without authentication.

The incident underscored the inherent risks associated with supply chain vulnerabilities, where weaknesses in one component can cascade through interconnected systems, leaving a trail of potential exploits and data breaches. As a cornerstone in the cybersecurity arsenal of organizations spanning healthcare, finance, retail and government sectors, FortiSIEM's vulnerabilities had the potential to compromise the security postures of numerous downstream organizations, amplifying the magnitude of the threat.

This incident highlights the heightened risks posed by vulnerabilities in widely-used security tools, particularly those entrusted with safeguarding critical digital infrastructures. Security lapses in such tools not only jeopardize the integrity of individual organizations but also undermine the overall resilience of digital ecosystems.

In response to the FortiSIEM incident, the cybersecurity community emphasized the urgent need for robust API security measures spanning the entire supply chain.



¹¹https://www.theregister.com/2024/02/06/fortinet_fortisiem_vulns/



Case in Point 2¹²

In mid-2023, a significant software vulnerability surfaced in MOVEit, a popular file transfer application utilized by numerous organizations.

This vulnerability, officially classified as CVE-2023-34362, entails a SQL injection flaw within the MOVEit Transfer web application, potentially granting unauthorized access to its database. Given MOVEit's widespread adoption for secure file transfer functionalities, the breach has raised concerns regarding data security and confidentiality.

The impact of the MOVEit breach is substantial, marking one of the largest API-enabled data breaches in recent history. Over 700 organizations have fallen victim to the breach, with more than 47 million data records compromised. The majority of affected organizations are based in the U.S., followed by Germany, Canada and the U.K., highlighting the global scale of the incident.

The attack path for exploiting the MOVEit vulnerability follows a multi-step process, enabling bad actors to gain unauthorized access and execute malicious actions.

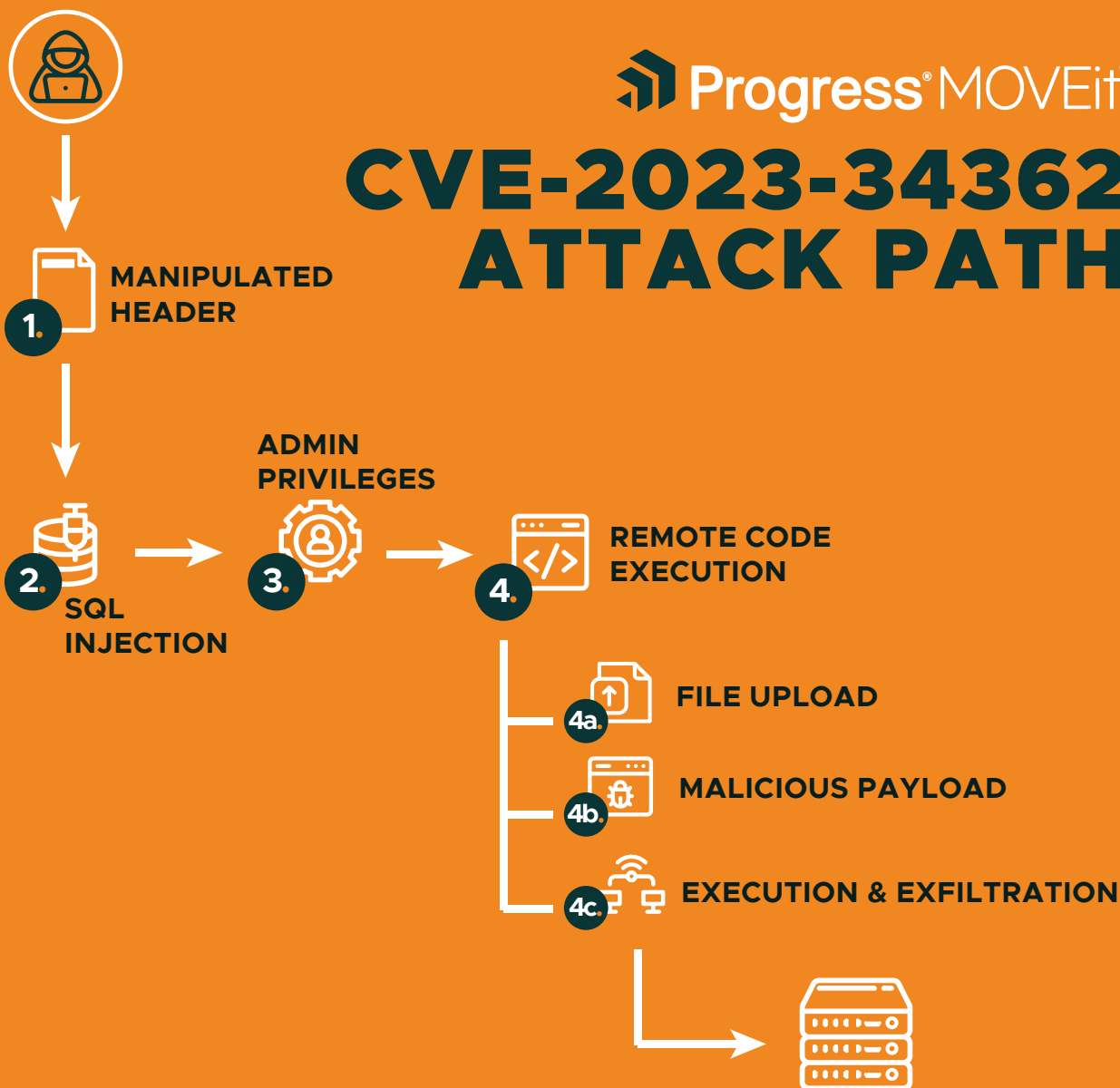
Initially, attackers manipulate API headers to bypass input sanitization functions, facilitating an SQL injection attack at a guest registration endpoint.

Subsequently, attackers exploit this SQL injection to acquire administrative privileges via a manipulated JSON Web Token (JWT) sent to an external endpoint controlled by the attacker. With administrative rights obtained, attackers exploit a flaw in the file upload functionality, triggering remote code execution and potentially facilitating data exfiltration.



¹²<https://www.firetail.io/blog/moveit-series-breach-enabled-apis>

CVE-2023-34362 ATTACK PATH



This breach underscores several API-related vulnerabilities outlined in the OWASP API Top 10. Specifically, the attack exploits unauthenticated access (OWASP API 2023:2) and manipulated API calls triggering undesirable behavior (OWASP API 2019:8 Injection). However, the authentication method utilized in the attack does not directly align with OWASP API Top 10 recommendations, highlighting the evolving nature of API security threats and the need for comprehensive mitigation strategies.



Case in Point 3¹³

In January 2024, a critical zero-day vulnerability, identified as CVE-2023-35078, in Ivanti Endpoint Manager Mobile (EPMM), formerly known as MobileIron Core, was exploited in the wild, resulting in limited attacks. This authentication bypass vulnerability allowed unauthenticated remote attackers to access the server's API, typically accessible only to authenticated users. Successful exploitation enabled attackers to access specific API paths, potentially obtaining personally identifiable information (PII) and mobile device details managed by EPMM.

Attackers could leverage the unrestricted API paths to modify server configuration files, potentially creating admin accounts for further system compromise. The vulnerability's severity, with a CVSS v3 score of 10.0, posed significant risks to organizations relying on Ivanti's EPMM solution for mobile device management (MDM), mobile application management (MAM) and mobile content management (MCM). Although Ivanti released public advisories, detailed information was restricted to a customer-only knowledge base article, limiting broader awareness and mitigation efforts.

Confirmed exploitation of CVE-2023-35078 was linked to a small number of customers, emphasizing the targeted nature of the attacks. Notably, a cyber attack against twelve Norwegian government ministries was attributed to the exploitation of this vulnerability, highlighting its real-world impact. Security experts observed probing of vulnerable EPMM systems shortly after the vulnerability's disclosure, further underscoring the urgency of patching and securing affected environments.



¹³<https://www.tenable.com/blog/cve-2023-35078-ivanti-endpoint-manager-mobile-epmm-mobileiron-core-unauthenticated-api-access>

Public API Breaches.

Since 2022, FireTail has maintained an API breach tracker¹⁴ to log and analyze all publicly-reported incidents and vulnerabilities involving APIs. For the most part, these API vulnerabilities and misconfigurations were not found by the companies themselves, even when they performed audits internally or via a traditional IT security firm.

In many cases, outside security firms or bug bounty hunters identified these issues, which were subsequently fixed before any evidence of malicious activity was found. But it is hard to prove that malicious activity did not happen. The impact of API data breaches is massive today, and will continue to grow in the future.

The following insights are taken from an analysis of that API Data Breach Tracker.

Source Data

Year	2017	2018	2019	2020	2021	2022	2023	TOTAL
Incidents	2	4	3	10	6	20	22	67
Breaches	1	3	2	10	6	10	18	50
Records Breached	2,200,000	310,000,000	800,013,021	214,996,739	65,670,000	55,770,518	175,328,679	1,623,978,957
Avg Records	2,200,000	103,333,333	400,006,511	21,499,674	10,945,000	5,577,052	9,740,482	32,479,579

Notes

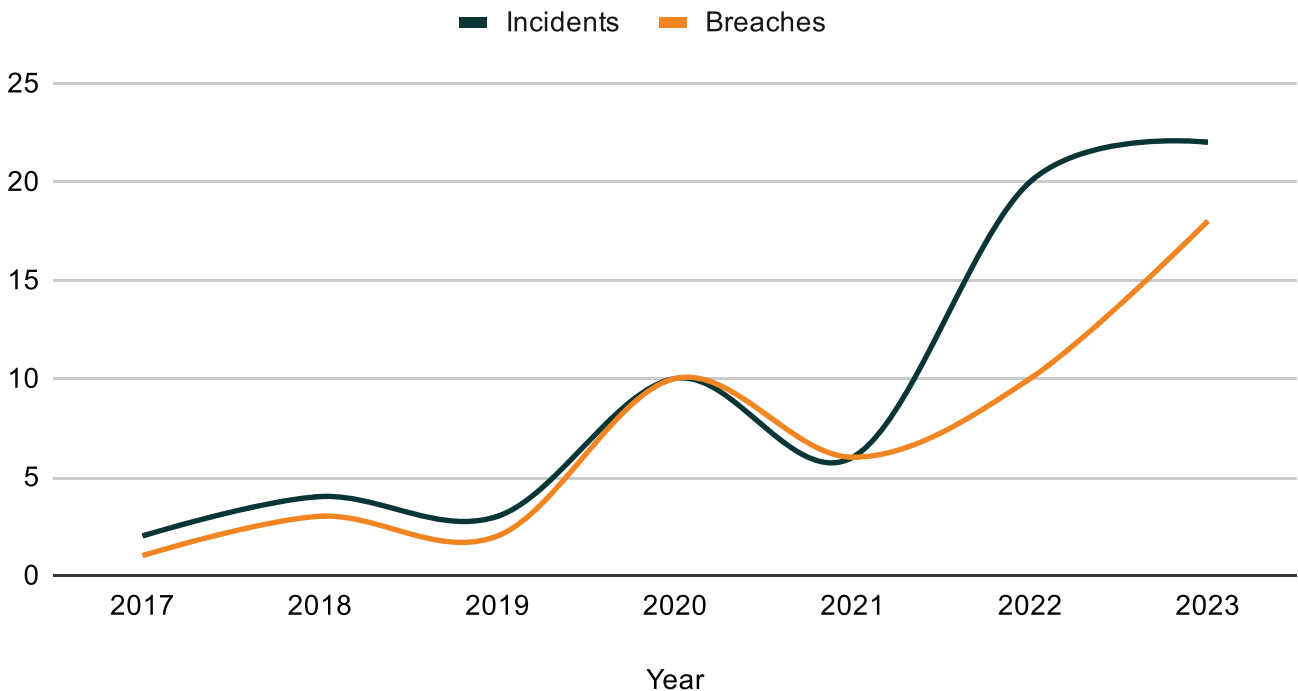
- Years are calculated 1 Feb to 31 Jan in order to correspond to FireTail's financial year.
- Incidents are all events logged on the breach tracker including misconfigurations and vulnerabilities identified and resolved before any confirmed threat actor access.
- Breaches are events where there was a confirmed number of records publicly exposed.
- Totals and data for 2021 through 2023 do not correspond to data published in our 'State of API Security Report 2023'. New events are added to the breach tracker as they are discovered. Since the publication of our previous report in May 2023, additional events that occurred during this time period have been included.



¹⁴<https://www.firetail.io/api-data-breach-tracker>

API Data Breaches Up 80%

The volume of breaches where records were confirmed to have been compromised grew **80%** year on year. The compound annual growth rate for breaches from 2017 to 2023 stands at **61.87%** and for incidents stands at **49.13%**.



1.6 Billion Records Exposed

2023 saw **175M** records breached up **214%** on 2022. In total, since 2017 the 50 breaches recorded on the tracker show **1,623,978,957** records exposed in the 7 year period. The average number of records exposed per breach is greater than **32M**.

Responsible Disclosures on the Rise

There has also been a steady increase in the proportion of incidents where no records are confirmed to have been breached. From 2017 to 2020, the proportion of incidents that were not breaches was **15.79%**. From 2021 to 2023, that proportion almost doubled to **29.16%**. This suggests that greater awareness of API security risks and more focus by researchers are leading to more vulnerabilities being identified and remediated before threat actors can take advantage.



Fortune 500 API Analysis.

While FireTail's API Data Breach Tracker is a useful tool for sampling and understanding the wider API threat landscape, it has limitations. The events recorded represent only those breaches, misconfigurations and vulnerabilities that are publicly disclosed and spotted by FireTail's research team.

In order to further evaluate the scale and reality of the API threat that exists in 2024, our researchers also conducted an analysis of publicly-available, publicly-documented APIs belonging to Fortune 500 companies.

Findings

Our researchers identified and analyzed 206 publicly-available, publicly-documented APIs as part of this study.

# APIs	206
Findings	Spec-based findings showing level of severity for vulnerabilities and misconfigurations identified
Critical	0
High	3921
Medium	4415
Low	127000
Informational	23000
Total	158336
Avg per API	769

158,336 Issues Identified

Across the **206** Fortune 500 APIs, our researchers discovered more than **158K** issues and average of **769** per API.

3,921 High Severity Findings

Encouragingly, there were no 'Critical' severity issues identified among the **206** APIs analyzed, however, there were **3,921** 'High' severity and **4,415** 'Medium' severity findings.

Use of Numeric IDs Dominate High Severity Findings

Virtually all of the 'High' severity findings related to the use of numeric



IDs. In fact, **3,918** or **99.92%** fell into this category with just **3**, or **0.08%**, pertaining to ‘Credentials in URL.’

The pervasiveness of numeric IDs is concerning given the potential for enumeration and data scraping. Specifically, given that authorization is the leading breach vector, numeric IDs are particularly susceptible to BOLA and BOPLA (Broken Object Level Authorization and Broken Object Property Level Authorization, respectively). A malicious actor can simply change the request parameter that corresponds to the record ID in API requests by incrementing integer identifiers to obtain data. This exact attack pattern is responsible for a huge number of breached records. In most such cases, traditional security approaches have failed, as the API requests look like legitimate user behavior.

4,415 Medium Severity Findings

The ‘Medium’ severity findings identified typically fell into the following three categories:

- Unconstrained properties
- Missing global ‘security’
- Missing authentication on specific endpoints

Alarming Rates of Issues in High-Profile APIs

Publicly-available, publicly-documented APIs from Fortune 500 companies should represent some of the best secured APIs out there. These are highly-visible resources from sophisticated organizations with significant security and development resources. And while it’s encouraging that we did not identify any ‘Critical’ issues across the **206** APIs we identified, the **3,921** ‘high’ severity issues discovered are cause for concern. If these companies can’t get it right on their most public APIs, the reality is that the extent of vulnerabilities in smaller organizations or across internal, zombie and shadow APIs is much more pronounced.



GitHub Repository Scanning.

In order to further investigate the state of the modern threat to APIs, we conducted a similar exercise focused on APIs in publicly-accessible GitHub repositories.

Our researchers identified 2,879 APIs. Below is a breakdown of the spec-based findings and the associated OWASP Top 10 risks:

- **owasp:api4:2023-rate-limit: 2448**
- **owasp:api4:2023-string-limit: 2413**
- **owasp:api4:2023-rate-limit-responses-429: 2392**
- **owasp:api4:2023-string-restricted: 2392**
- **owasp:api3:2023-define-error-responses-401: 2239**
- **owasp:api3:2023-define-error-responses-500: 2158**
- **owasp:api3:2023-define-error-validation: 1993**
- **owasp:api4:2023-array-limit: 1819**
- **owasp:api2:2023-protection-global-safe: 1523**
- **owasp:api4:2023-integer-limit-legacy: 1450**
- **owasp:api2:2023-protection-global-unsafe-strict: 1370**
- **owasp:api2:2023-protection-global-unsafe: 1326**
- **owasp:api4:2023-integer-format: 1144**
- **owasp:api3:2023-no-additionalProperties: 494**
- **owasp:api2:2023-jwt-best-practices: 439**
- **owasp:api3:2023-constrained-additionalProperties: 429**
- **owasp:api1:2023-no-numeric-ids: 406**
- **owasp:api2:2023-no-http-basic: 286**
- **owasp:api4:2023-integer-limit: 176**
- **owasp:api2:2023-no-credentials-in-url: 101**
- **owasp:api4:2023-rate-limit-retry-after: 17**
- **owasp:api2:2023-auth-insecure-schemes: 3**



The top 3 categories of issue identified in these APIs are as follows:

- **Lack of rate limiting**
- **Lack of parameter validation**
- **Lack of error responses**

85% of APIs are missing rate limiting

Rate limiting and pagination are common defense mechanisms to slow down attackers and reduce the scale of any potential breach. In many cases, rate limiting can be implemented at the network layer, with tools like API Gateways. Still, including rate limiting in an API specification is a recommended best practice.

84% of APIs show string limit issues

This is the second highest category of issue identified and speaks to a wider problem. String limit, string restricted, array limit, integer limit legacy, integer format, constrained additional properties, no numeric IDs and integer limit can all be classified as problems with perimeter validation. These perimeter validation problems were prevalent in some form across 98% of APIs and this class represents 38% of all issues identified. While the OWASP API Top 10 for 2023 may have removed, renamed and de-prioritized injection attacks, the facts on the ground show that the single largest API attack (and largest cybersecurity incident of 2023), the MOVEit series of ransomware attacks, relied on an injection to breach an API. Proper input validation and sanitization is as critical for APIs as it is for traditional web or enterprise applications.

77% of API specs missing 401 responses

API endpoints that do not return HTTP status code descriptions are more difficult to use for developers, while being easier for bad actors. Without standard responses defined in API specifications, default server responses are likely, and may expose too much information if the server is in debug or verbose logging mode. Consistent and well-documented HTTP status codes in API specifications also play a vital role in understanding the usage of an API. When developers consume an API, clear status code definitions save them time and reduce the risk of misinterpreting a response. Similarly, clearly defined responses help identify specific attack vectors and pinpoint the needs for debugging and additional security measures.



FireTail Platform Data.

FireTail worked with customers to analyze real-world API traffic, the final piece of the end-to-end API security stack. Our researchers analyzed a random sample of 2,934,267 API calls across a cross-section of APIs belonging to clients in different industries. Below is a breakdown of the error responses:

Status code	Description	Number	Percentage
429	Too Many Requests	563646	19%
503	Service Unavailable	440712	15%
404	Not Found	103040	3%
502	Bad Gateway	4116	0.14%
400	Bad Request	2136	0.07%
401	Unauthorized	1192	0.04%
403	Forbidden	477	0.01%
500	Internal Server Error	272	0.01%
415	Unsupported Media Type	26	0.00%
405	Method Not Allowed	8	0.00%
406	Not Acceptable	3	0.00%
422	Unprocessable Entity	3	0.00%

Only 60% of all API traffic was successful. This suggests companies are overpaying for compute infrastructure because they lack the visibility of the API logs to show them that their APIs are not behaving in the expected ways.

By volume alone, '429' errors were the single largest category in the data set at 19% of traffic, with the closely related server error '503' at 15% of traffic. Together, that's 34% of all API calls.

These error messages can be attributed to two common API scenarios:

- **DDoS attacks**
- **Overwhelmed APIs that are poorly designed on the server side to withstand stressful utilization**



FireTail then conducted analysis to identify security vulnerabilities and misconfigurations across these APIs. Below is a breakdown of the findings:

Finding Name	Number of Instances	Severity
Endpoint authentication removed	10	Critical
No numeric IDs	499	High
No credentials in URL	320	High
No API keys in URL	6	High
Global Safe	2585	Medium
Global Unsafe Strict	1282	Medium
Constrained additional properties	127	Medium
Security hosts https OAS3	143	Medium
Security hosts https OAS2	52	Medium
No http basic	12	Medium
String Limit	95060	Low
String Restricted	110070	Low
Error 429	5929	Low
Error 500	5919	Low
Error 401	5827	Low
Integer Limit	10151	Low
Error validation	3432	Low
No additional properties	125	Low
JWT best practices	61	Low
Array Limit	11749	Informational
Rate Limit	12452	Informational
Integer Format	2057	Informational
Global Unsafe	1282	Informational
Majority status code 500	23	Informational

How do APIs in public repositories compare to the real world?

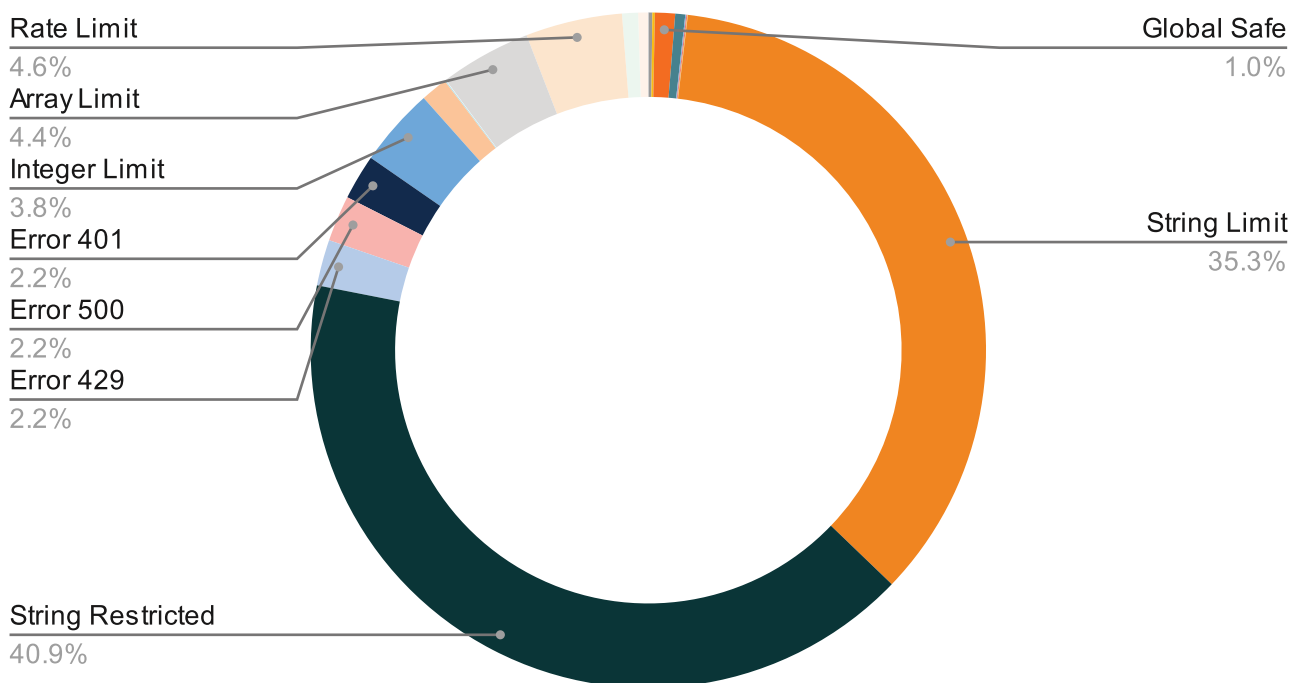
One potential question is whether publicly available APIs are representative of APIs found in actual production environments.



There are some key differences in both more and less secure directions. Across the most critical severity levels (Critical, High, Medium), these were the most commonly observed issues:

- **9.9% were numeric IDs**
- **6.7% were related to authentication issues - endpoints where authentication was removed, or endpoints using API keys or credentials in URLs**
- **Only 2.5% were in parameter**

Among the lower severity issues, request parameter issues dominated the findings.



Other common behaviors

Other common behaviors identified during our analysis included:

- Probing for secrets by requesting resources like environment variables and configuration files
- Enumeration of tech stack through follow-up API calls to check for common third-party software package APIs that have known vulnerabilities.



Another related observation, from both customer and FireTail internal lab environments, is that most APIs receive traffic – both standard HTTP probes and then more targeted follow-up APIs calls within less than 5 minutes of coming online. Security through obscurity is not a viable defense mechanism.

Novel observations

One other interesting observation is the attempt to distribute or plant malware on systems via properly formatted API calls. These included both Mirai botnet malware and n***a.sh malware.

There were other instances that involved attempts to deliver or download scripts via API request payloads that call for directory changes, permissions changes and other common Linux shell commands.

APIs Represent Massive Attack Surface for Modern Organizations

Having analyzed public API breaches, Fortune 500 APIs, GitHub specs and FireTail user data, it's clear that APIs represent a significant attack surface that remains a blind spot for many companies. In the next section, we look deeper into the root causes of API breaches and how to protect your organization from API attacks.

APIs are a clear, present & future danger.



API Attack Vectors.



In order for attackers to be successful, more than one thing has to go wrong.



API Attack Vectors.

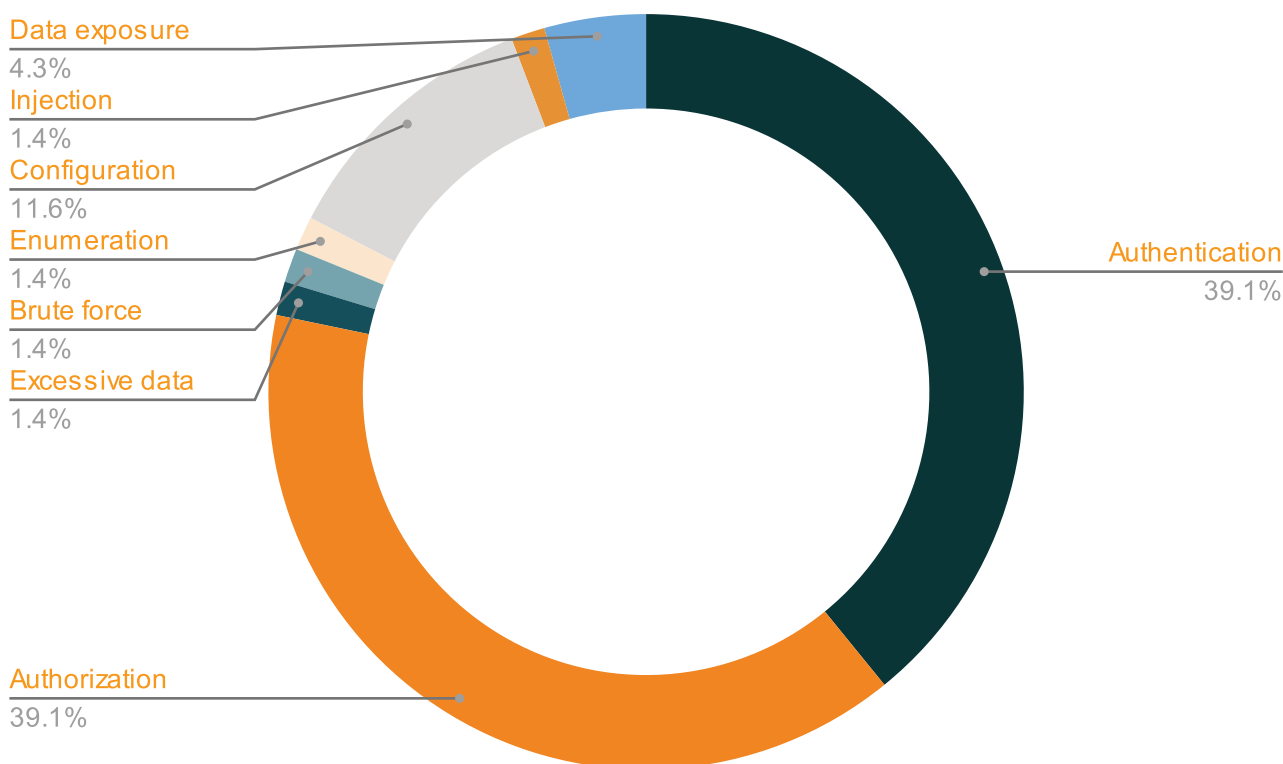
In this section, we analyze the common root causes for successful API attack and identify the most significant attack vectors.

- 
- **High-Impact Attack Vectors**
 - **Other Notable Attack Vectors**
 - **Failure of Network Security for API**

High-Impact Attack Vectors.

In order to understand the root causes of successful API breaches, we analyzed all incidents from our API Data Breach Tracker up to 31st of January 2024. The chart below shows the breakdown of primary attack vectors.

API Breaches by Primary Attack Vector



One of the factors that makes API security so difficult is that the attack vectors don't necessarily align to common defense tools or methodologies, especially in the age of the cloud. Historically, many organizations start their cybersecurity maturity process with a combination of TTPs - tools, techniques (or technology) and procedures.

As many cybersecurity teams evolved out of IT teams, there has been a natural gravitational pull in cybersecurity towards the common IT layers that these people have domain expertise in, such as network security, operating systems and related threats (malware, viruses, endpoint protection) as well as logging (think of security incident and event management, or SIEM).



Yet for the most part, the breaches that have happened via APIs would evade all of these TTPs.

Authentication and Authorization Still Dominate

As with our analysis in 2023, the top two categories of primary attack vector are still authorization and authentication in terms of both number of breaches and the volume of records breached. 78.2% of all incidents relied on AuthZ or AuthN issues as a primary attack vector.

Both of these fall under the broad category of identity, and are intrinsically linked to the application, where identity is normally established, verified and assigned permissions to which parts of the application (functions) and which records (data) can be accessed.

One often overlooked consideration in the authentication process is validating authentication credentials repeatedly, and binding credentials to an active session. Long-lived credentials, like static API keys, are subject to secret sprawl, including the risk of those secrets leaving your organization when an employee leaves.

Another authentication challenge is related to a different hot topic in cybersecurity - supply chain risk. Some common authentication mechanisms may actually introduce vulnerabilities into APIs. For that reason, it's important that APIs are designed in a way to force authentication on a regular basis, including checking whether a token is valid in an identity or secret store, rather than only checking whether a token conforms to the expected format.

Making Use of Malware

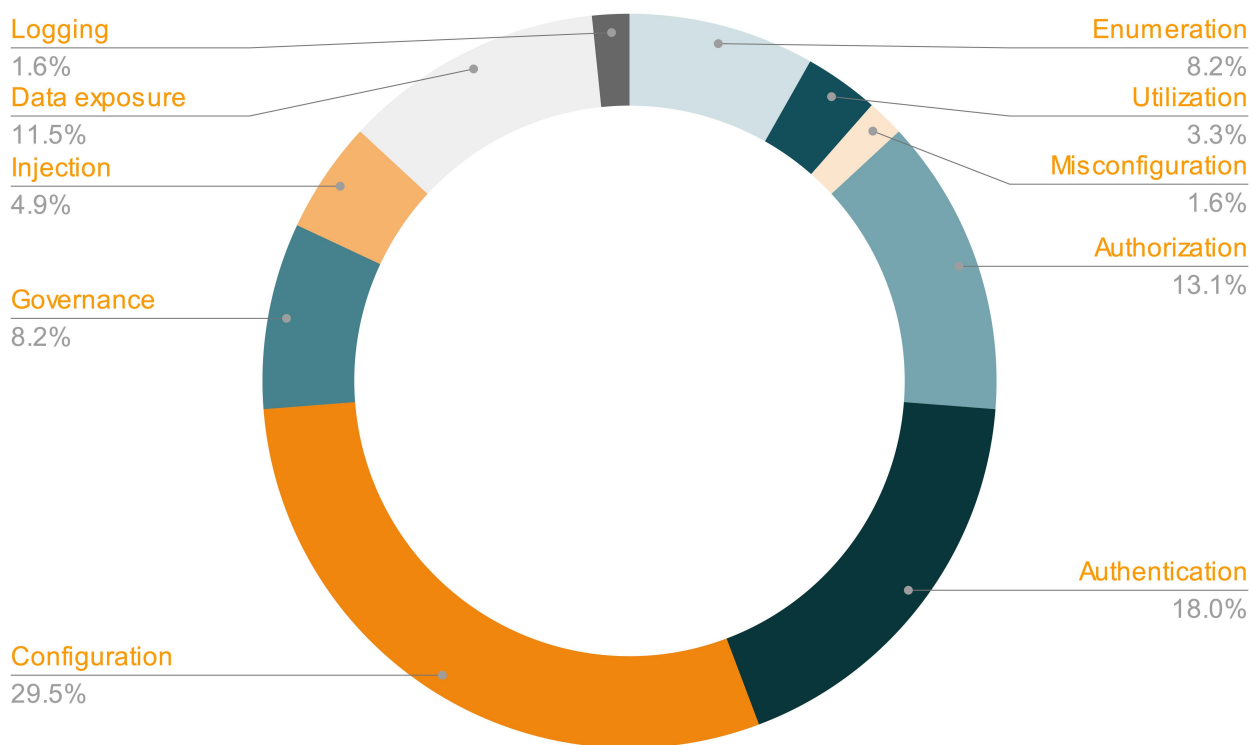
One other very interesting development this year is that APIs are being attacked with malware and exploits. For instance, the infamous Mirai Botnet, known for targeting Internet of Things (IoT) devices, has evolved to include APIs in its scope, leveraging them for reconnaissance and propagation. Additionally, recent incidents, such as the exploitation attempts observed in API traffic logs targeting the Ivanti CVE, highlight the increasing sophistication of attackers in targeting API vulnerabilities using a combination of malware and exploits.



Other Notable Vectors.

Our analysis shows that in order for attackers to successfully exploit an API vulnerability, more than one thing has to go wrong. The vast majority of breaches involved a secondary attack vector. The chart below shows a breakdown.

API Breaches by Secondary Attack Vector



This year's update to the OWASP API Security Top 10 combined 'Excessive Data Exposure' and 'Mass Assignment' to create 'Broken Object Property Level Authorization' or BOPLA. Renewed focus on this area is warranted. Our analysis indicates that these application logic vulnerabilities are implicated in approximately 65% of exposed data records.

The intricate nature of these breaches often makes it challenging to pinpoint primary versus secondary root causes. Authentication or authorization flaws frequently pave the way for these breaches, facilitating unauthorized access to sensitive data.



Failure of Network Security for APIs.

To reiterate, these cases will look like normal network traffic, and network security approaches are extremely unlikely to have visibility into the data returned to an attacker.

This implies that at best, NTA (network traffic analysis) or NDR (network detection and response) will only flag the exposures after the data has left your network, and in cases where traffic patterns show marked deviation from normal traffic, if anomaly detection is turned on.

Furthermore, overall cybersecurity governance, typically defined as a combination of oversight and accountability, coupled with mitigation strategies and plans, is often a contributing factor. This is a typical symptom of advanced technical adoption that outpaces cybersecurity control systems.

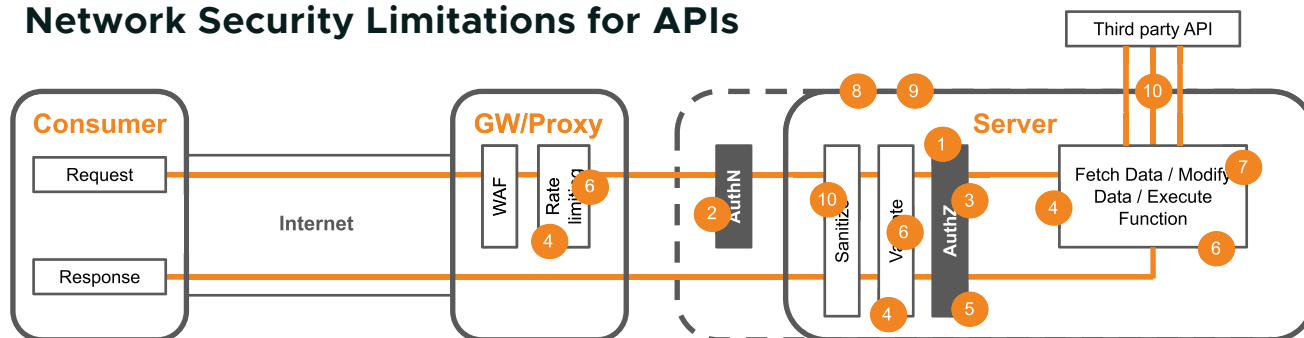
Our analysis of the API data breaches shows that in all the cases where “security misconfiguration” was flagged as a breach vector, the cause was universally an API that was believed to be private, accidentally becoming publicly available. In those cases, the breaches bypassed non-existent or trivial (string-based, not tokenized) authentication, so arguably authentication was equally a root cause. In fact, rate limiting, a free and recommended feature of all leading API Gateways, was only cited as a primary or secondary breach vector of 0.1% of the records exfiltrated.

API attacks look like normal requests.



The diagram below illustrates how network security techniques will fail to prevent the vast majority of API attack vectors covered by frameworks such as the OWASP API Security Top 10.

Network Security Limitations for APIs



- | | |
|---------------------------------------|-----------------------------------|
| 1. BOLA. | 6. Unrestricted Process Access. |
| 2. Broken AuthN. | 7. SSRF. |
| 3. BOPLA. | 8. Misconfiguration. |
| 4. Unrestricted Resource Consumption. | 9. Improper Inventory Management. |
| 5. BFLA. | 10. Unsafe consumption of APIs. |

It has also been proven that other network security tools, like WAFs, can be bypassed with simple techniques.

Enumeration and Probing

One valid concern around network traffic is whether bots or other attacker-driven automations are mapping your APIs to discover weak points. While not identified on the OWASP Top 10 list, FireTail lab testing has shown that 3% of all traffic received by APIs is bot traffic looking for credentials, secrets, access points or ways to query data from an API.

As demonstrated by our analysis of Fortune 500 APIs, 99.8% of the 'High' severity findings discovered related to the use of numeric IDs. The risk of these types of breaches is on the rise with the accelerated pace of cloud adoption, the growing ease of automation offered and the widespread expansion of API calling capabilities offered by AI.



Effective API Security Strategies.




The threat has grown but the principles of protection remain the same.



Effective API Security Strategies.

In this section, we look at what it takes to protect your APIs in the face of accelerated threats:

- 
- **6 Pillars of API Security**
 - **Context is King**
 - **Code to Cloud**

6 Pillars of Effective API Security

While the overall size of the API attack surface has increased and the pace of attacks has accelerated, the nature of these attacks has remained largely constant.

As such, the fundamentals of protecting your APIs remain the same. Below is an overview of the 6 pillars of effective API security.



Discovery.

If you can't see it, you can't secure it. Effective API security begins with identifying all of your APIs. Build an accurate and up-to-date picture.



Visibility.

Once you've discovered all APIs, develop a deep inventory that provides visibility into the nature and behavior of those APIs.



Observability.

Monitor APIs for risky traffic, performance and errors. Use anomaly detection to identify deviations. Set custom thresholds for alerting.



Audit.

Create a full, centralized audit trail of all APIs with powerful search features and payload visibility. You need detailed logs for API incident response and reporting.



Assessment.

APIs should be continuously analyzed for misconfigurations and vulnerabilities. Automate your API security posture management.



Enforcement.

Ensure consistent policy and governance across the organization. Use runtime protection with API call validation, authentication checks & injection prevention.

Context is King.

When it comes to API security, many existing approaches just don't work. Successful API breaches look like normal requests. Attackers exploit business logic flaws to prod and probe your APIs for weaknesses. API gateways, WAFs and network traffic analysis just won't stop them.

Existing Approaches Lack Key Data.

API Call Log Visibility			
	VPC Flow Logs	WAF/API/GW	App Logs
Target	✓	✓	✓
Source	✓	✓	✓
URI	✗	✓	✓
Auth Header	✗	✓	✓
Args	✗	✗	✓
Req. Params	✗	✗	✓
Req. Payload	✗	✗	✓
Resp. Payload	✗	✗	✓

Real API security relies on context only found at the application layer. Payload visibility is critical towards ensuring a proper security posture is actively being implemented. Application layer visibility refers to the thorough monitoring and analysis of interaction with an API above and beyond simple network activity, providing a comprehensive understanding of how data is being interacted with on a call-by-call basis. If network visibility is seeing the numbers that were dialed and the length of the call, application visibility is tapping into the phone line and listening to the conversation.

This allows for precise examination of data content and behavior, aiding in the proactive detection of threats, effective policy enforcement, anomaly identification, and post-event investigations. This approach goes beyond the traditional network monitoring that is common in offerings such as WAFs, offering deeper insights into traffic and more actionable context.



Code to Cloud.

APIs are built, not born – they are the product of meticulous coding. It follows that their security must also be a fundamental part of the development cycle. The "Code to Cloud" approach embodies this principle, recognizing that APIs are at risk from the moment the first line of code is written.

Vulnerabilities can creep into APIs through insecure coding practices, misconfigurations within the codebase, and the use of compromised libraries or components. Left unchecked, these issues transform into attack vectors when an API is deployed in test, staging, or production environments. Shifting API security 'left' integrates checks early into the development process (using tools like SAST and SCA) and ensures remediation before potential attacks become real threats.

Additionally, a Code to Cloud approach emphasizes the need to tailor protection to the diverse environments where your APIs operate. Runtime API security in production requires a different toolkit than what you implement during coding. Runtime environments deserve a dedicated layer of protection (using tools like API gateways and WAAPs) for real-time threat detection and mitigation.

Effective API Security Summary

To achieve robust API security, organizations must adopt a multi-pronged and continuous approach that utilizes tools specifically designed for API protection:

- **6 Pillars:** Establish strong foundations through discovery, visibility, observability, auditing, assessments and enforcement.
- **Contextual Awareness:** Inspect payloads and deeply understand API behavior for accurate threat recognition.
- **Code to Cloud:** Embed security into development and adapt protection measures to each environment across the API's life cycle.



At FireTail, we're on a mission to secure the world's APIs with one API security platform that provides application layer visibility, real-time, inline inspection and blocking of malicious API calls.



FireTail

FireTail™ is a registered trademark

NORTH AMERICA

USA: FireTail, Inc. EIN 88-0823835
1775 Tysons Blvd, Suite 500, McLean,
VA 22102, USA | +1.703.828.5171

EUROPE

FireTail International Ltd. CRN 717465
2 Dublin Landings, North Wall Quay, Dublin 1,
D01V4A3, Ireland | +353.86.800.6111

info@firetail.io | www.firetail.io



FireTail