FireTail

The State of APIs and API security in 2023

PRESENTED BY FIRETAIL

firetail.io

TABLE OF CONTENTS

Table of Contents	**2**
Contributing authors	3
About this publication	3
Disclaimer	3
Document history	3
The rise and rise of APIs	**4**
What is the value of an API?	**6**
Key Statistics from Publicly Disclosed API Data Breaches	**8**
Hundreds of Millions of Records Breached	**9**
Analyzing API Attack Vectors	**9**
The OWASP API Top 10	10
The highest impact breach vectors	11
The failure of network defense for API security	13

CONTRIBUTING AUTHORS:

Jeremy Snyder, Riley Priddle, Ian Foster

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.

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 publicly disclosed information, which may not always include full details on a data breach.

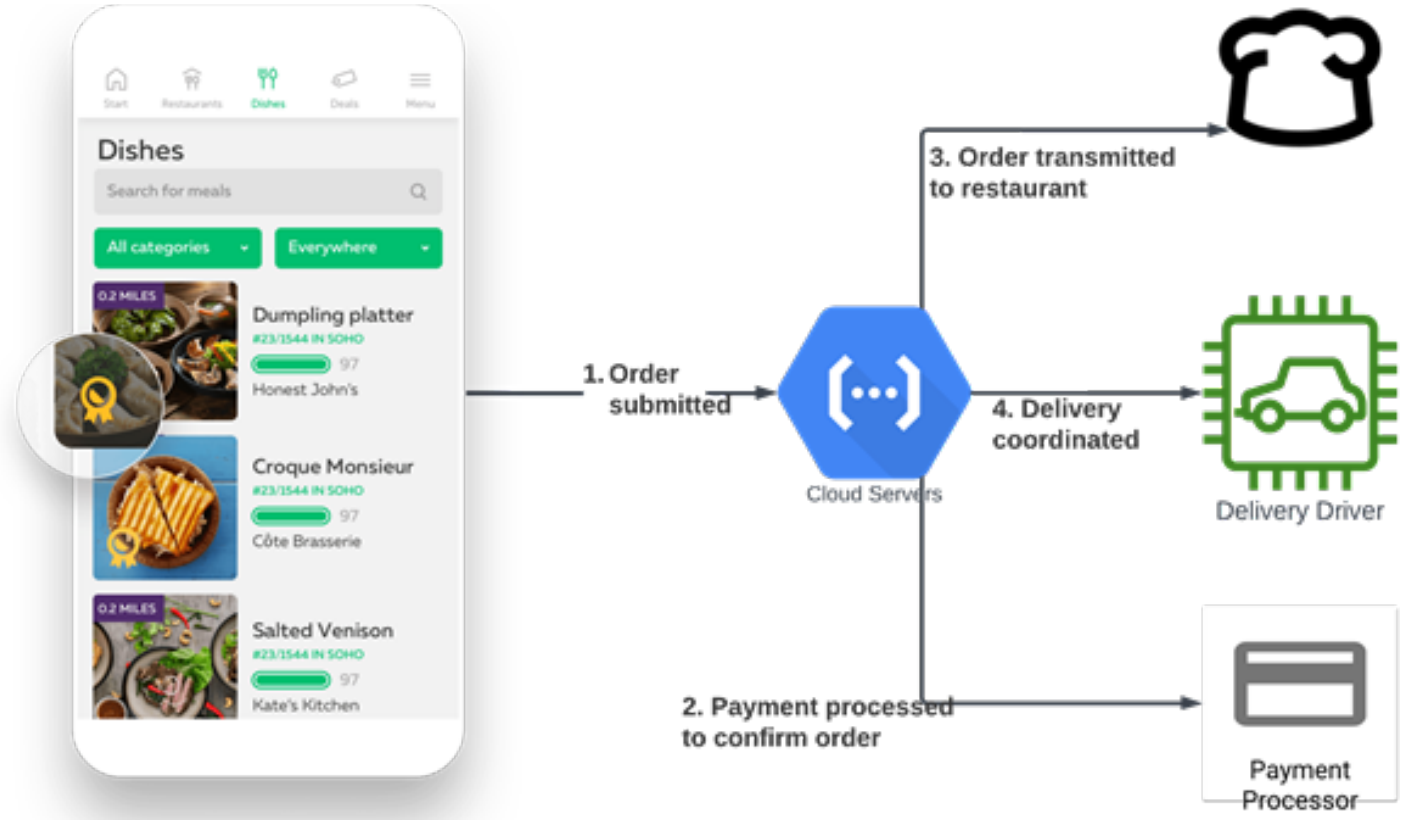
DOCUMENT HISTORY

The first version of this report was produced and published in April 2023.

WHY EXAMINE API SECURITY, AND WHY NOW?

The rise and rise of APIs

APIs (Application Programming Interfaces) are everywhere. They allow software systems to communicate with other software systems, both within companies and across enterprises / organizations. Already more than 83% of Internet traffic involves API communication[1] , not human-to-computer, and that percentage will continue to grow year over year. This is a little known fact, but one with major implications. Take the example of a food delivery app, probably very familiar after years of home-based shopping and ordering:



Sample mobile food ordering transaction, simplified

[1] https://apisecurity.io/issue-17-83-web-traffic-apis-query-params-bad-secrets/

It's easy to observe that this is a multi-party transaction, involving at least 4 third-party providers, and transmitting a high volume of sensitive data between the parties. For instance:

- The user sends an order, their home address, other personal identifiable information (PII), payment information and likely additional data shared from a social media profile and/or social login credentials.
- An ordering app in the cloud transmits the payment information to a payment processor before submitting the order to the restaurant. This is a multi-step process, involving transmission, confirmation, and call-back to the ordering app. A record is stored on the app upon the completion of the transaction, together with transaction ID.
- The ordering app then transmits the order to the restaurant and awaits confirmation. Once confirmed, the order is noted and recorded in the app, and the customer may be notified of order confirmation.
- A delivery notice is sent out to a fleet of potential logistics partners. Time, location, transportation mode, availability, other jobs in the queue and other factors may be consulted. Delivery is eventually confirmed, and the customer is notified.

Even in the simplified delivery diagram, at least 5 API calls made across 4 parties are demonstrated. In practice, a recent conversation with one of these app providers estimates that the actual number of API calls is closer to 25.[2] It's clear that the volume of API communication is ever-increasing at an exponential rate, as is the volume of sensitive data flowing over APIs. Forbes has hinted that the API economy may be the "next big thing"[3], but a recent headline from Google[4] summarized the current situation best:

SECURITY BLANKET: Our digital lives are only secure if our APIs are secure.

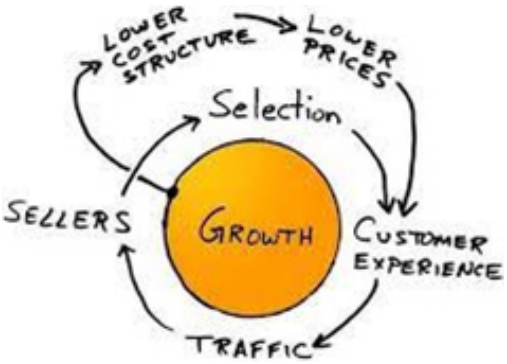
[2] Conversation with cloud-based food delivery app, AWS re:Invent 2022.
[3] https://www.forbes.com/sites/tomtaulli/2020/01/18/api-economy-is-it-the-next-big-thing/?sh=f251c4942ffd
[4] https://cloud.google.com/blog/transform/api-security-network-cybersecurity-digital-life-equinix/

WHAT IS THE VALUE OF AN API?

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

Why do companies create APIs? Why do developers create APIs?

Around 2002, reportedly, Amazon created an API mandate inside the company.[5] Reports at the time varied as to why Amazon did this, with reasons ranging from cutting down on internal cross-departmental meetings, to leveraging complementary data from diverse data sets. Whatever the original reason was, it led to a proliferation of APIs inside the company, and some that have become key partner-facing APIs over the years since.



The Amazon Flywheel
Source: Amazon

APIs open your organization to external value creation

Imagine that you run that 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. 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).[6]

Effectively, an API ecosystem can be one step towards creating a digital flywheel[7] of value creation.

[5] https://medium.com/api-university/the-api-mandate-install-api-thinking-at-your-company-4335433b7d0b

[6] Source: https://www.youtube.com/watch?v=nynBDfk8Fi4 Apidays Interface 2022 - How APIs Create Growth by Inverting the Firm By Marshall Van Asltyne, Questrom Chair Professor at Boston University and author of Platform Scale

[7] https://fourweekmba.com/amazon-flywheel/

APIs are the new battleground for application security.

But with opportunity, comes risk

One downside to the openness of this API-driven ecosystem approach, however, is that it comes with risk. Whenever an organization transmits data to a third-party, there is risk involved in everything from technical controls around encryption to contractual risks around data handling, privacy or data sovereignty. The same studies that showed a massive increase in organizational value, also show 13.5% increase in the risk and occurrence of API-based data breaches for companies that open an external API.[8] And in fact, this is playing out, with at least 40% of companies running in the cloud acknowledging some level of API-based data breach.[9]

Technical and industry analysis to quantify this risk has varied over the past several years, as the volume of API-based data breaches has grown, but estimates range from 40% to 90% of web-based applications' attack surface is represented by their APIs.[10] Some analyst firms have gone so far as to predict that 2022 would have been the year that APIs represented the number one attack vector for organizations.[11] While that may not have happened, the growth of API risk is clear, with attacks up well over 100% half-year over half-year.[12]

[8] See previous footnote from Apidays.

[9] Google: With cloud comes APIs and security headaches https://www.darkreading.com/cloud/google-cloud-apis-security-headaches

[10] https://www.forbes.com/sites/forbestechcouncil/2020/07/21/whats-under-the-hood-of-api-security

[11] https://www.infosecurity-magazine.com/webinars/apis-no1-enterprise-attack-vector/

[12] https://www.darkreading.com/attacks-breaches/cdnetworks-releases-state-of-web-security-h1-2022-attacks-against-api-services-surged-168-8-

REVIEWING A DECADE OF API SECURITY

The FireTail research team has compiled a list of publicly disclosed API-based data breaches. This list is available on the FireTail website as the FireTail API data breach tracker.[13] The first reported API security breach happened in June of 2011. While details of the event are no longer available in depth, it's notable that the first incident was reported only 3 years after a landmark study of web services was shared in 2008.[14] This is typical of new technologies - they are created, embraced initially by early adopters, and over a typical technology cycle of 2-3 years, start to get adopted into production environments where actual transactions and data are processed, inviting attackers in a similar timeframe.

Key Statistics from Publicly Disclosed API Data Breaches

40 Data Breach Events and Research Findings have been made Public

While the total number of breaches may not appear to be massive, it's interesting to see that the rate has increased dramatically in the past few years:

Over 500M records have been exposed or at-risk from APIs

Year	% breach acceleration	# breach events	# average records
2021	117%	7	11,167,142.86
2022	172%	12	1,347,045.67
2023[16]	227%	17	2,901,174.71

2023 is on track to be a record year, with 6 disclosures in the first 2 months of the year alone, with a potential impact of 49 million records.[15][16]

[13] The API breach tracker, hosted on the FireTail website at https://firetail.io/api-data-breach-tracker
[14] https://en.wikipedia.org/wiki/Web_service; WWW2008 Conference, Beijing
[15] https://firetail.io/api-data-breach-tracker
[16] 2023 predictions are extrapolated based on data to date and trend lines from the past 3 years

Hundreds of Millions of Records Breached

There were 12 publicly reported API data breaches in 2022, out of 4,100 total breaches that are known.[17] For comparison purposes, there were at least 335 ransomware attacks during the same period.[18] While the total number of breach events may strike some as posing less of a risk than other attack surfaces, it's important to understand that the nature of many API breaches leaves entire customer bases or entire datasets open to exposure, as has been demonstrated in both research and real-world scenarios. Recently, a telecommunications provider's API exposure allowed bad actors to exfiltrate PII of the entire customer base, well over 10 million people.[19] In fact, using the data available, the average (mean) size of API data breach exposure is over 10M records per incident. [20]

Leading research around the cost of a data breach, both direct (lost revenue, incident response, consumer credit monitoring and protection and similar services) and indirect (reputational damage) currently puts the business cost of a single breached record at $180 USD. This implies a cost of over $85B due to API data breaches, and nearly $2B in potential risk for any incident.[21]

Analyzing API Attack Vectors

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 common IT layers that these people have domain expertise in, such as network security, operating systems and related threats (malware, viruses, endpoint protection), and 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.

[17] https://www.cshub.com/attacks/articles/the-biggest-data-breaches-and-leaks-of-2022

[18] https://www.securityinfowatch.com/cybersecurity/article/21292765/ransomware-attacks-declined-in-22-but-more-records-being-comprmised

[19] https://securityboulevard.com/2022/10/optus-data-breach-why-vulnerable-apis-are-to-blame/

[20] Specifically 11,925,582, as per FireTail research

[21] Assuming recoverability and implied consumer data.

The OWASP API Top 10

OWASP is one of the leading volunteer research consortiums for application security research. The OWASP Top 10 has outlined the top 10 risks for APIs as far back as 2019:[22]

ID:	Name:	Description:	Aligns to:
1	Broken Object Level Authorization	"APIs tend to expose endpoints that handle object identifiers, creating a wide attack surface Level Access Control issue. Object level authorization checks should be considered in every function that accesses a data source using an input from the user. "	Application-layer permissions and identity
2	Broken User Authentication	"Authentication mechanisms are often implemented incorrectly, allowing attackers to compromise authentication tokens or to exploit implementation flaws to assume other user's identities temporarily or permanently. Compromising a system's ability to identify the client/user, compromises API security overall."	Application-layer identity
3	Excessive Data Exposure	"Looking forward to generic implementations, developers tend to expose all object properties without considering their individual sensitivity, relying on clients to perform the data filtering before displaying it to the user."	Application-layer data handling
4	Lack of Resources & Rate Limiting	"Quite often, APIs do not impose any restrictions on the size or number of resources that can be requested by the client/user. Not only can this impact the API server performance, leading to Denial of Service (DoS), but also leaves the door open to authentication flaws such as brute force."	Network access
5	Broken Function Level Authorization	"Complex access control policies with different hierarchies, groups, and roles, and an unclear separation between administrative and regular functions, tend to lead to authorization flaws. By exploiting these issues, attackers gain access to other users' resources and/or administrative functions. "	Application-layer permissions and identity
6	Mass Assignment	"Binding client provided data (e.g., JSON) to data models, without proper properties filtering based on an allowlist, usually leads to Mass Assignment. Either guessing objects properties, exploring other API endpoints, reading the documentation, or providing additional object properties in request payloads, allows attackers to modify object properties they are not supposed to."	Application-layer data handling
7	Security Misconfiguration	"Security misconfiguration is commonly a result of unsecure default configurations, incomplete or ad-hoc configurations, open cloud storage, misconfigured HTTP headers, unnecessary HTTP methods, permissive Cross-Origin resource sharing (CORS), and verbose error messages containing sensitive information."	Network access[23]
8	Injection	"Injection flaws, such as SQL, NoSQL, Command Injection, etc., occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's malicious data can trick the interpreter into executing unintended commands or accessing data without proper authorization."	Application-layer data handling
9	Improper Assets Management	"APIs tend to expose more endpoints than traditional web applications, making proper and updated documentation highly important. Proper hosts and deployed API versions inventory also play an important role to mitigate issues such as deprecated API versions and exposed debug endpoints."	Security and IT management
10	Insufficient Logging and Monitoring	"Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain persistence, pivot to more systems to tamper with, extract, or destroy data. Most breach studies demonstrate the time to detect a breach is over 200 days, typically detected by external parties rather than internal processes or monitoring."	Security and IT management

[22] https://owasp.org/www-project-api-security/

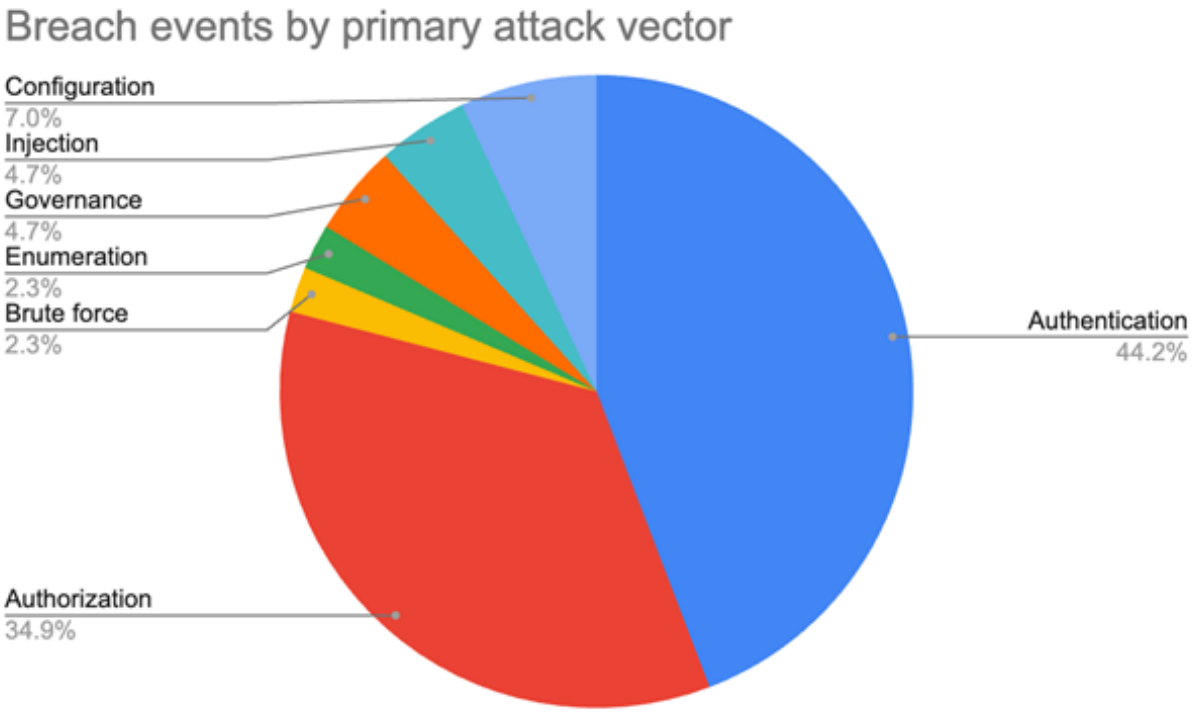
[23] Also implies cloud infrastructure / Infrastructure-as-a-service configuration security controls

One of the most interesting observations of this list is that only two out of 10 items on the list align with traditional network security controls, and yet for the past 10 years, many approaches to API management and security have relied on API Gateways and web application firewalls (WAFs). Additionally, three align to application-layer identity, three more align to application-layer data handling and the final two are general IT management and information security core principles.

None of these are covered by the historical TTPs of cybersecurity described previously. Endpoint protection on a server does not have application-layer visibility for authentication, authorization or data handling. Similarly, traditional network security built on blocking ports and protocols would fail to capture attacks that are built around application logic - after all, those will look like normal network traffic to most network security devices or software.

The highest impact breach vectors

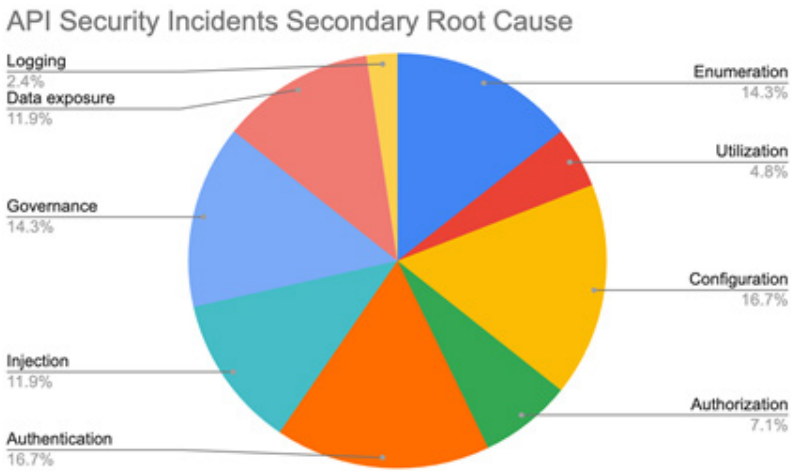
In analyzing the data from the breaches (dates, number of records, attack vectors, breach mechanisms), the top two categories of data breach are authorization (135M records, 28% of all records breached) and authentication (105M records, 22% of all records breached)[24] . Both of these fall under the 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.



[24] Categories are not mutually exclusive; breaches may be in multiple categories. Totals will exceed 100%.

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,[25] including the risk of those secrets leaving your organization when an employee leaves. Another authentication challenge is related to another hot topic in cybersecurity - supply chain security. Some common authentication mechanisms may actually introduce vulnerabilities into APIs.[26] 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.

Another interesting observation around these events is that many events involve two or more problems. These may be authorization problems after authentication has been checked, or this may be enumeration that leads to an unrestricted query endpoint.

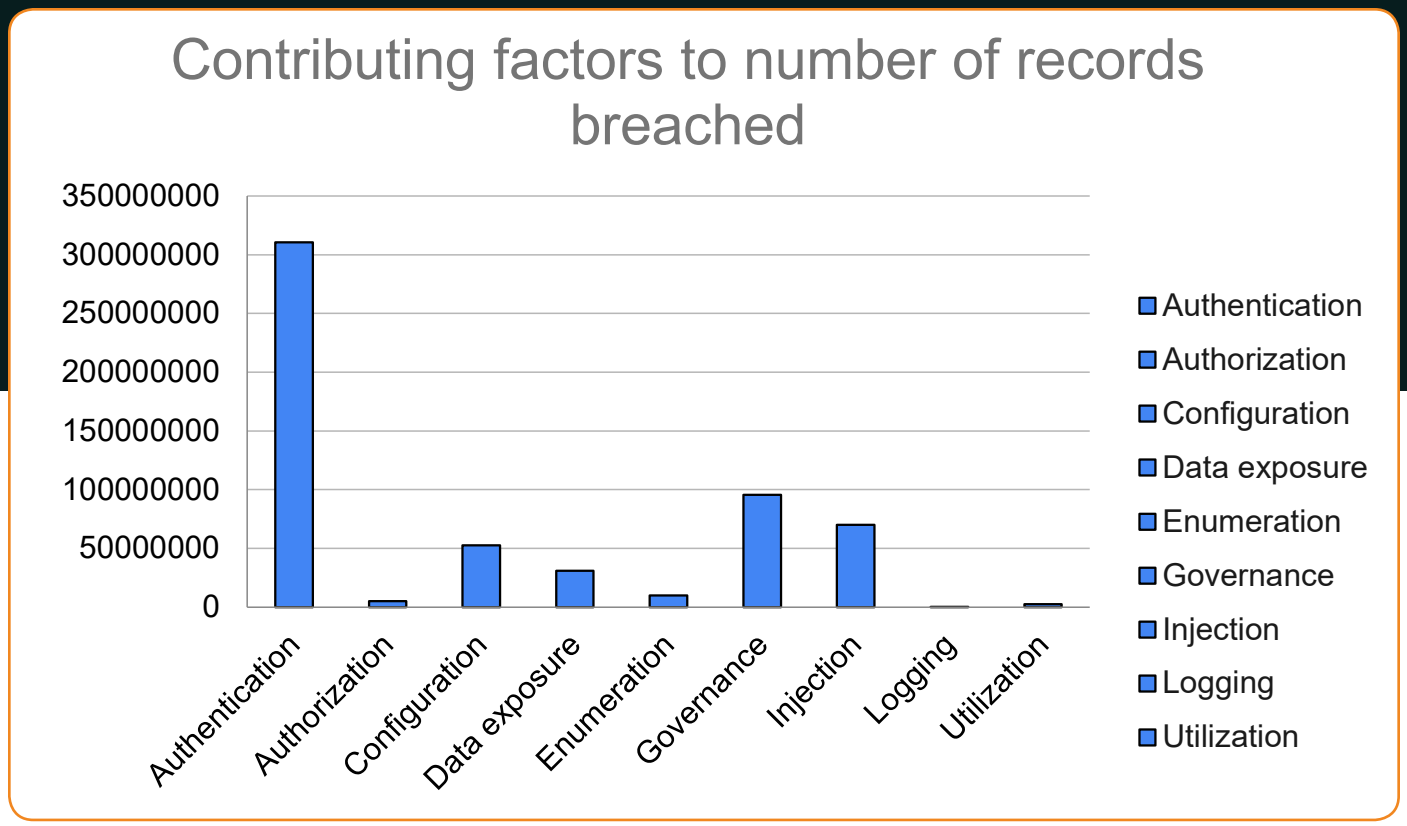


Other notable breach vectors

OWASP top 10 items 3) Excessive Data Exposure and 6) Mass Assignment have been massive breach vectors. While establishing primary versus secondary root cause is tricky in many of those cases, as authentication or authorization flaws allowed the queries, these two application logic vulnerabilities have been at least partially to blame for a whopping 65% of the data records exposed. 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.

Finally, 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.

[25] https://www.techtarget.com/searchsecurity/tip/How-to-manage-and-reduce-secret-sprawl
[26] https://www.darkreading.com/vulnerabilities-threats/jsonwebtoken-security-bug-opens-servers-rce



The failure of network defense for API security

There are two network controls in the OWASP Top 10 - rate limiting and security misconfiguration. The analysis of the API data breaches shows that in all the cases where "security misconfiguration" was flagged 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 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.

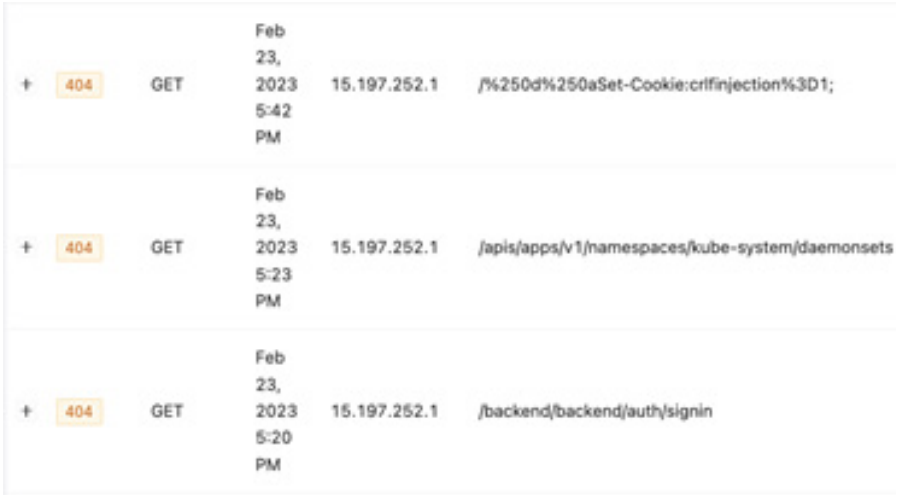
Finally, it has now been proven that other network security tools, like WAFs can be bypassed with simple techniques.[27] [28]

[27] https://thehackernews.com/2022/12/researchers-detail-new-attack-method-to.html

[28] https://www.darkreading.com/attacks-breaches/researcher-finds-akamai-waf-bypass-to-trigger-rce

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 up to 99.8% of all traffic received by APIs is looking for credentials, secrets, access points or ways to query data from an API.



+	404	GET	Feb 23, 2023 5:42 PM	15.197.252.1	/%250d%250aSet-Cookie:crlfinjection%3D1;
+	404	GET	Feb 23, 2023 5:23 PM	15.197.252.1	/apis/apps/v1/namespaces/kube-system/daemonsets
+	404	GET	Feb 23, 2023 5:20 PM	15.197.252.1	/backend/backend/auth/signin

Even leading multinational companies have found themselves guilty of putting APIs online that have or have had these risks.[29] The risk of these types of breaches is on the rise with the accelerated pace of cloud adoption, more frequent application deployment, and the lack of visibility for security teams that comes with those patterns.[30]

The Impact of Cloud APIs

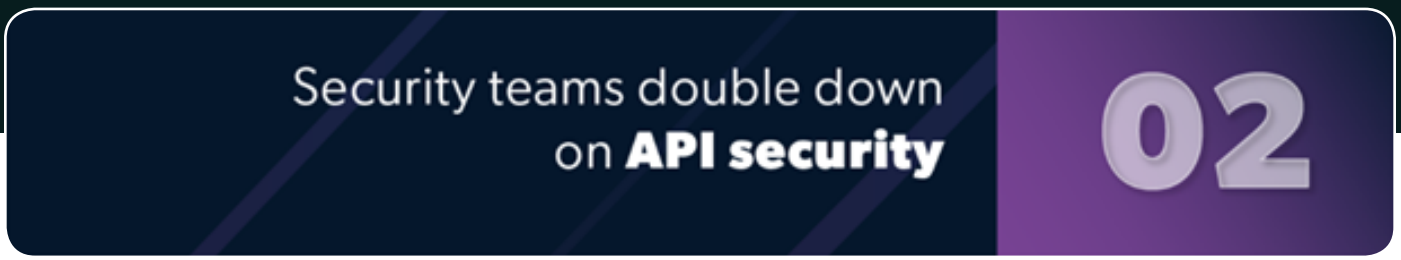
One other macro trend very closely related to API security is the rise of cloud adoption.[31] Just as APIs can enable broader ecosystem positive impact, interaction with cloud platforms in any programmatic way involves APIs. This can introduce third-party risk, as cloud platforms have experienced their own set of security challenges for the APIs and related services that they provide, ranging from potential access to another customer's cloud environment[32] to newly released services having undocumented APIs, or APIs that do not provide the centralized audit trails that customer security teams rely on.[33]

APIs: Key to Cloud Transformation

[29] https://samcurry.net/hacking-starbucks/
[30] https://securityboulevard.com/2022/09/api-security-incidents-rise-despite-confidence-in-protection/
[31] https://www.darkreading.com/cloud/google-cloud-apis-security-headaches
[32] https://securitylabs.datadoghq.com/articles/appsync-vulnerability-disclosure/
[33] https://securitylabs.datadoghq.com/articles/iamadmin-cloudtrail-bypass/

EFFECTIVE API SECURITY STRATEGIES

Some analysts are predicting that 2023 will be the year that organizations start to make investments into API security,[34] or at least engage in meaningful research to find and implement strategies.



An axiom of cybersecurity is that if you can't see it, you can't protect it. A CISO Mag survey captured this as the number one priority for CISOs[35] looking to engage in API security programs:

1. Lack of API inventory
2. Enforcing perimeter security
3. End-to-end tracing of code to API
4. Number of required security configs per API
5. API change management, security implications
6. Gap between developers and security teams

Lack of inventory, getting communication between developers and security teams and change management are all symptomatic of the visibility problem. Tools that can discover APIs are a starting point. However, it's critical these discovery activities are ongoing and aligned to an organization's infrastructure and application technology stacks.

Once visibility is established, organizations can move forward with other strategies to counter the other attack vectors, either from the perspective of prioritization related to the volume of breaches (number of events or records), or relative to the risks that they believe that they face in their API estate. For instance, those with public-facing APIs for mobile apps may wish to focus on authentication, authorization and data exposure; while other organizations with only internal or partner-facing APIs may wish to focus efforts on security configuration, combatting potential public exposure and preventing enumeration.

As organizations get deeper into eliminating the risk of the various attack vectors, one very important consideration will be at which layer to implement security controls. Different inspection points across the technology or network stack provide access to different data elements.

[34] https://aite-novarica.com/blogs/tari-schreider/2023-cybersecurity-trends-you-need-know-about
[35] https://cisomag.eccouncil.org/api-security/ (Page archived)

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

If organizations feel that their risks are primarily around data exposure (API 3: Excessive Data Exposure and API 6: Mass Assignment) and from the call parameters or data responses to API calls, only the application layer can provide that visibility.

Similarly, if organizations are primarily concerned with post-authentication authorization checks (API 1: Broken Object Level Authorization and API 5: Broken Function Level Authorization), these authorization controls are typically implemented in application code, or by third-party services elsewhere in an organization's network. External-facing network controls, or NTA / NDR will not be able to see the result of an authorization calculation.

FireTail

About FireTail

FireTail engineered a hybrid approach to API security: an open source library that protects programmable interfaces with inline API call evaluation and blocking, cloud-based API security posture management, centralized audit trail, and detection and response capabilities. FireTail is the only company offering these capabilities together, ultimately helping organizations eliminate API vulnerabilities from their applications and providing runtime API protection.

FireTail is headquartered in Washington, DC, with additional offices in Dublin, Ireland and Helsinki, Finland.

FireTail is backed by leading investors, including Paladin Capital, Zscaler, General Advance and SecureOctane.

FireTail. API Security.
Import. Setup. Done

USA: FireTail Inc, EIN 88-0823835

IE: FireTail Int'l Ltd, Co. reg. nr. 717465

1775 Tysons Blvd, Suite 500, McLean,
VA 22102, USA I +1.703.828.5171

2 Dublin Landings, North Wall Quay, Dublin 1,
D01V4A3, IRELAND I +353.86.800.6111

info@firetail.io I www.firetail.io