CEQUENCE® SECURITY

# API Protection Report

## Shadow APIs and Automated Abuse Explode

First Half 2022

# Introduction

The term API economy carries a tinge of marketing hype, but the reality is we are truly living in it. Whether it's our favorite shopping, financial management, food delivery, ridesharing application or the new cars we drive, the tablets and mobile devices we use, or our home appliances, all of them are built on APIs. When viewed from this perspective, the API economy is real and can be measured in the many billions of dollars.

History has shown that the rapid adoption of a particular technology is followed by an equivalent growth in cyber threats along with solutions designed to address those threats. This is exactly what has transpired in the API protection space. API security related incidents like those at Peloton, ClubHouse, John Deere and more recently at Twitter where API keys were inadvertently exposed demonstrate the need for API protection. The market has responded with numerous new API security focused solutions to protect against the increased risks to these APIs and the attacks that exploit them.

CEQUENCE® SECURITY

# Contents


**The Intersection of APIs and Bots — 3**

Methodology — 4

**Key Findings — 5**

1. Shadow APIs are the #1 Attack Vector — 6
2. Mitigating ATOs Against APIs Saves $193 Million — 7
3. API10+: Perfectly Coded APIs Abused by Bots — 8
4. The Unholy Trinity: Credential Stuffing, Shadow APIs & Sensitive Data Exposure — 9

**Physical and Digital Worlds Collide — 10**

Blocking an Inventory API Enumeration Attack Saves $80,000 — 11

SIM Swapping – Stealing Your Phone Digitally — 12

OWASP API Security Top 10 to CWE Mapping — 13

**Ecosystem and 3rd-Party APIs Under Attack — 14**

Partner Ecosystem APIs: A Target Rich Environment for Bots — 15

Apple Pay API Abuse — 16

API10+: API Business Logic Abuse — 17

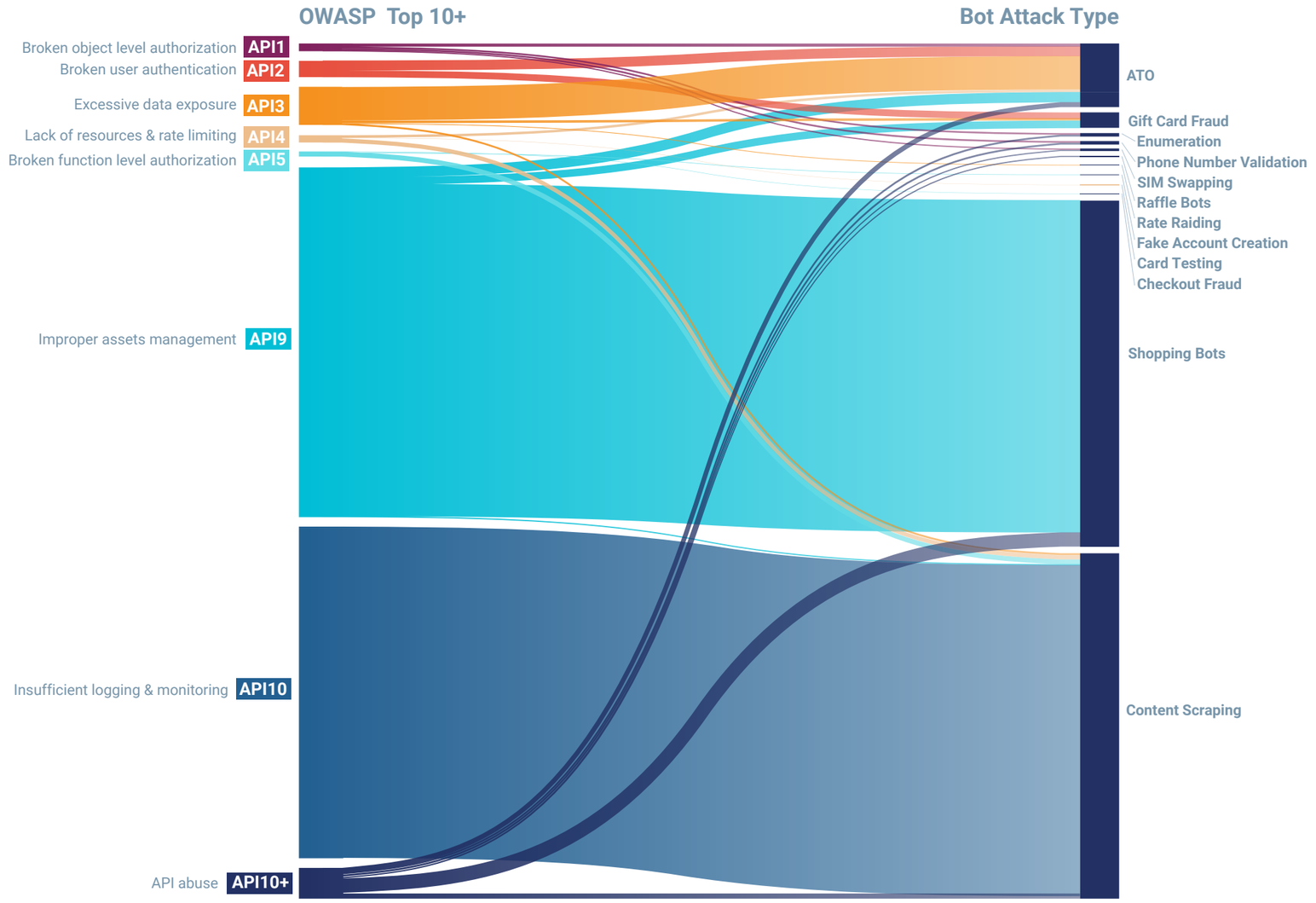
**Conclusion — 18**

# The Intersection of APIs and Bots

The rapid growth in API attacks and the corresponding security solutions market has led to a confusion that stopping bots and API security do not belong together. The truth is APIs and bots are inextricably connected. The same characteristics that developers love about APIs – flexibility, speed, ease of use – are also loved by attackers who either find coding errors to exploit, or use bots to attack perfectly coded APIs, or a combination of both.

Complete API protection will be illusive unless you have a complete understanding of how APIs – both correctly coded or those with errors – can be attacked. This includes how a risk is discovered, the tactics, tools, and procedures attackers use to exploit it, and how attackers will react to resistance. This means not only making sure that your APIs are not susceptible to the OWASP API Security Top Ten as a starting point, but also to look at what can be defined as API10+, a category that encompasses the many different ways that a perfectly coded API might be abused.

## MAPPING OWASP API TOP 10+ TO BOT ATTACKS OBSERVED



# Methodology

The Cequence CQ Prime Threat Research team pored over roughly 20 billion transactions from the first half of 2022 and highlighted trends we believe will help defenders understand this critical intersection. Our data focuses heavily on active API exploit attempts, delivered by bots, and natively mitigated. Some of these findings may appear new to the API security industry, yet these patterns have been in use consistently by attackers for years. These include findings like the explosion in attack attempts against shadow APIs, and the persistent threat that business logic abuse presents to an API ecosystem.

On the flip side, some of these discoveries may appear new to the bot management market, as these are new attack strategies used against APIs. Examples include exploiting Broken Object Level Authorization (BOLA) errors to execute Subscriber Identity Module (SIM) swapping attack campaigns, or attempted exploits of shadow APIs which simultaneously expose excessive data, a very attractive combination for an attacker. Some of these examples will attempt to expand upon the well-known OWASP API Security Top 10, highlighting how attackers turn the theoretical into the exploitable.

The rampant API abuse, the CQ Prime Research team coined as OWASP API 10+ because of its wide-ranging implications highlight the importance of using the OWASP API Security Top 10 as a starting point, not as the sole focus of an API protection initiative. API inventory and risk monitoring is an essential component of that strategy, yet at the end of the day it is only one piece of the Unified API Protection journey.

# Key Findings

KEY FINDING 1

# Shadow APIs are the #1 Attack Vector

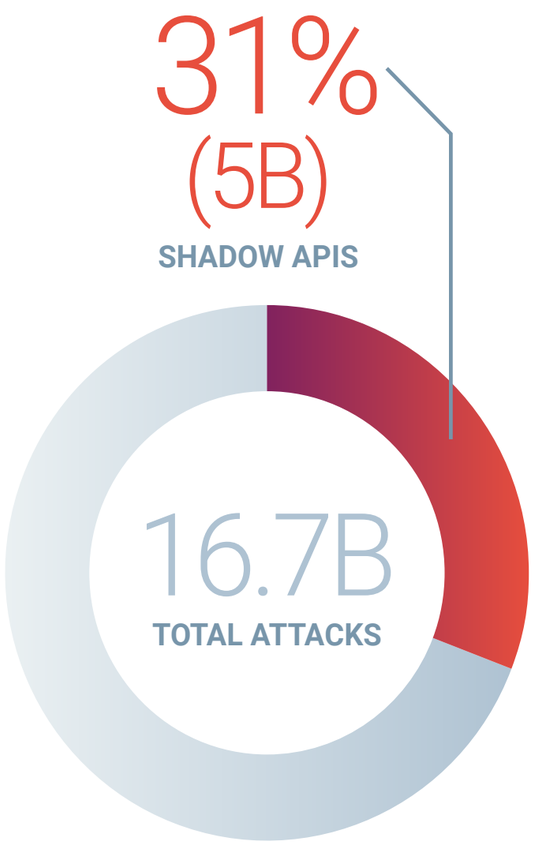
Roughly **31%, or 5 billion of the 16.7 billion malicious transactions** targeted unknown, unmanaged and unprotected APIs, commonly referred to as shadow APIs, making this the top API threat observed during the first half of 2022.

- Shadow APIs are a particularly pernicious threat that can be categorized as OWASP API9 (Improper Asset Management) abuse. Shadow APIs are a common problem in organizations that do not have proper inventory of their quality assurance/development API endpoints, or their versioning system, and attackers can easily discover API endpoints that will interact with production data. Shadow APIs can also appear when endpoints are coded to accept variables, or wildcard inputs either within the uniform resource identifier (URI) path or at the end.

- Attackers are able to easily find shadow APIs by analyzing a production API which may be well protected, then simply fuzz or modify the values, enumerating through other API endpoints on different versions, under different hostnames, or simply accepting random characters at the end of the URI. For attackers who are employing automation to monetize their attacks (e.g., shopping bots or credential stuffing), this strategy is akin to reading the manual on how an API works.

- Shadow API abuse was observed on a massive scale in first half 2022 with attacks spanning a wide range of use cases. From the highly volumetric sneaker bots attempting to cop the latest Dunks or Air Jordans, to stealthy attackers attempting a slow trickle of card testing fraud on stolen credit cards, to pure brute force credential stuffing campaigns. Driven by high volume content scraping as a precursor to shopping bot and gift card attacks, shadow API abuse surged in April 2022 and have continued to rise in volume throughout the year.

**TOTAL MALICIOUS API REQUESTS**



KEY FINDING 2

# Mitigating ATOs Against APIs Saves $193 Million

Highlighting the continued popularity of account takeovers, the CQ Prime Threat Research Team helped customers mitigate roughly 1.17 billion malicious account login requests  - all against APIs. The impact of an ATO on the business is significant, with each incident varying in cost from $290 (Juniper Research) and roughly 9 hours of investigative work to $311 (Federal Trade Commission). The mitigation efforts protected roughly 11.7 million accounts which equates to a savings of $193 million across all customers.

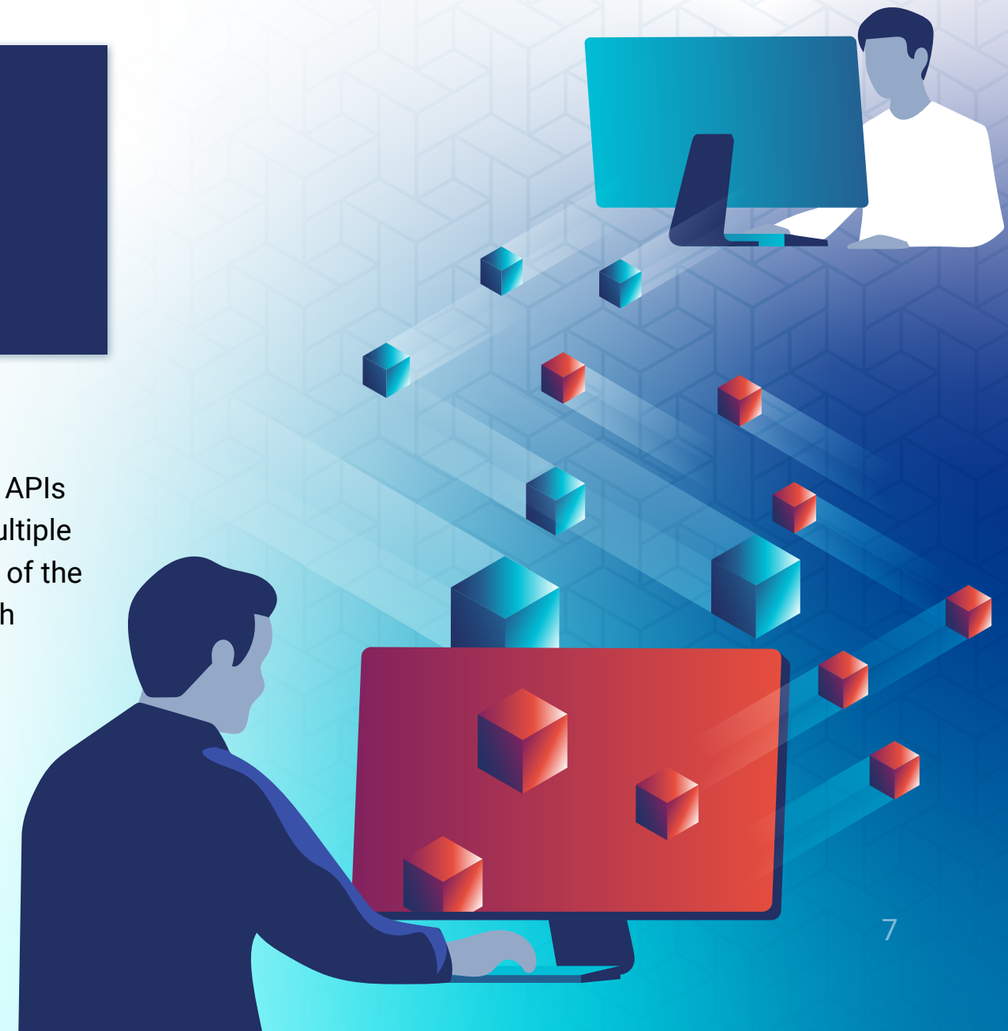
**ATOs Mitigated:**
1.17B

**Accounts Protected:**
11.7M

**Money Saved:**
$193M*

**Money saved: Average 1% ATO success rate times the $290 cost per account (Juniper Research).*

The popularity of ATOs can be tied directly to their versatility, which has been amplified by the adoption of APIs for account logins, and is shown throughout this report. ATOs, when successful, provide attackers with multiple end-goals ranging from theft of funds or loyalty points from a financial or travel account, to fraudulent use of the account to execute a purchase or validate gift cards. The other reason they remain popular is the ease with which they can be executed. Billions of stolen credentials are readily available. Commercially available ATO tools and infrastructure services have simplified the process of executing an ATO almost to the point where an attacker need only choose their target and launch the attack.

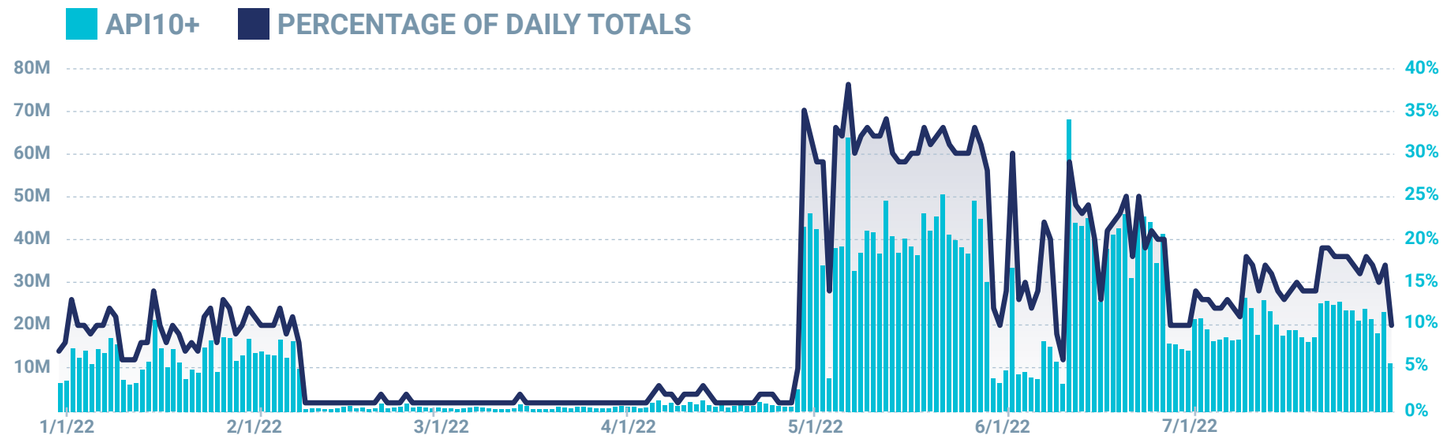


KEY FINDING 3

# API10+: Perfectly Coded APIs Abused by Bots

With 3.6 billion malicious requests blocked by the CQ Prime Threat Research team, the second largest API security threat mitigated during the first half of 2022 was API abuse, or what we call API10+. As an unofficial extension to the OWASP list, API10+ highlights how attackers target perfectly coded APIs that either do not cleanly fall into any of the OWASP API Security Top 10 threats or use a combination of them to achieve their goal. These attacks are targeted at APIs that are coded correctly and properly inventoried. Results from the malicious requests blocked include:

**TOTAL MALICIOUS API REQUESTS**




- **More than 3 billion** shopping bots targeting the latest hot sneakers or luxury goods.
- **Over 290 million** malicious gift card checking requests targeting well coded APIs with credential stuffing attacks looking to gain access to free money protected by a 4-digit PIN.
- **Roughly 237 million** fake account creation requests spanning a range of end goals from romance scams, to shopping bots, and other as-designed usage of the APIs.
- **More than 37 million** comment spam requests abusing correctly coded APIs that enable business critical engagement on customer sites.

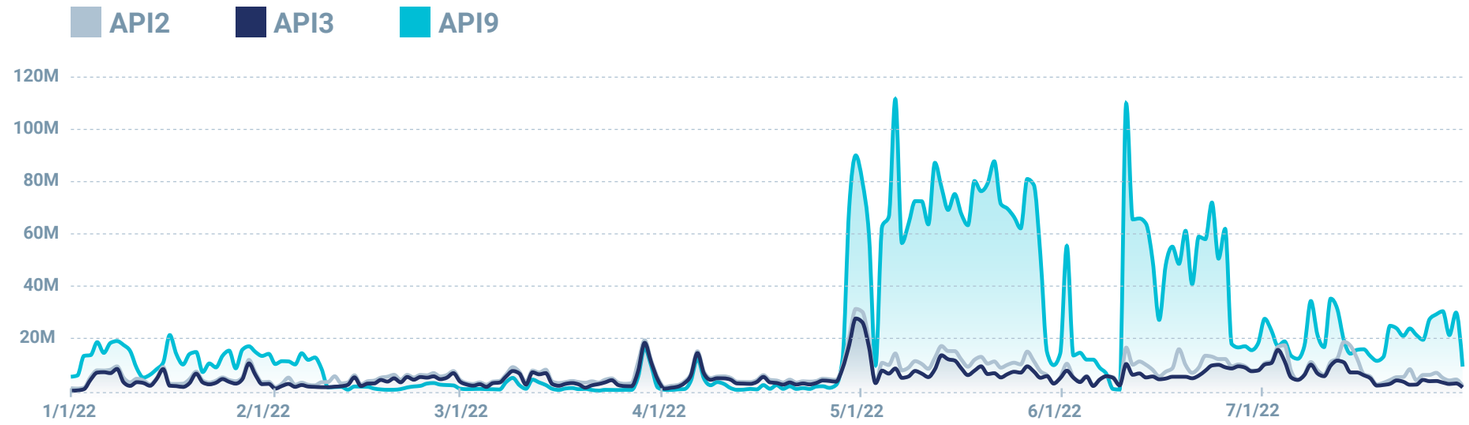
It is worth noting that detecting and mitigating API10+ attacks are exactly why API security and bot prevention are inextricably connected. An API only view would miss the behavioral abuse patterns that fall into the OWASP API10+ category.

KEY FINDING 4

# The Unholy Trinity: Credential Stuffing, Shadow APIs & Sensitive Data Exposure

While the volume of malicious requests compared to the other findings in this report is small at 100 million, the combined use of API2 (Broken User Authentication), API3 (Excessive Data Exposure) and API9 (Improper Assets management) signifies two things: attackers are performing detailed analysis of how each API works, how they interact with each other, and the expected outcome and developers need to stay ever vigilant in following API coding best practices.

**TOTAL MALICIOUS API REQUESTS USING API2, API3, API9**




## API2

**Credential Stuffing** is often associated with Broken User Authentication where attackers target the authentication mechanisms that protect user integrity. One of the most common behaviors that successful credential stuffing campaigns exhibit is a checker functionality that checks user confirmation APIs for sensitive customer data which can be stolen immediately after login.

## API3

**Excessive Data Exposure** occurs when the checker APIs return more data than necessary, as developers have a false sense of security because the user confirmation happens after authentication. The combination of APIs exposing too much (personal) data and those that are vulnerable to credential stuffing is a ripe target for API abuse.

## API9

**Shadow APIs** are a perfect example of Improper Assets Management making them susceptible to exactly the same API2 and API3 risks, but they are invisible to the security team. The lack of visibility means attackers can enumerate the victim's infrastructure using known API patterns to discover shadow APIs that are unprotected and vulnerable to credential stuffing and exposed excessive data.

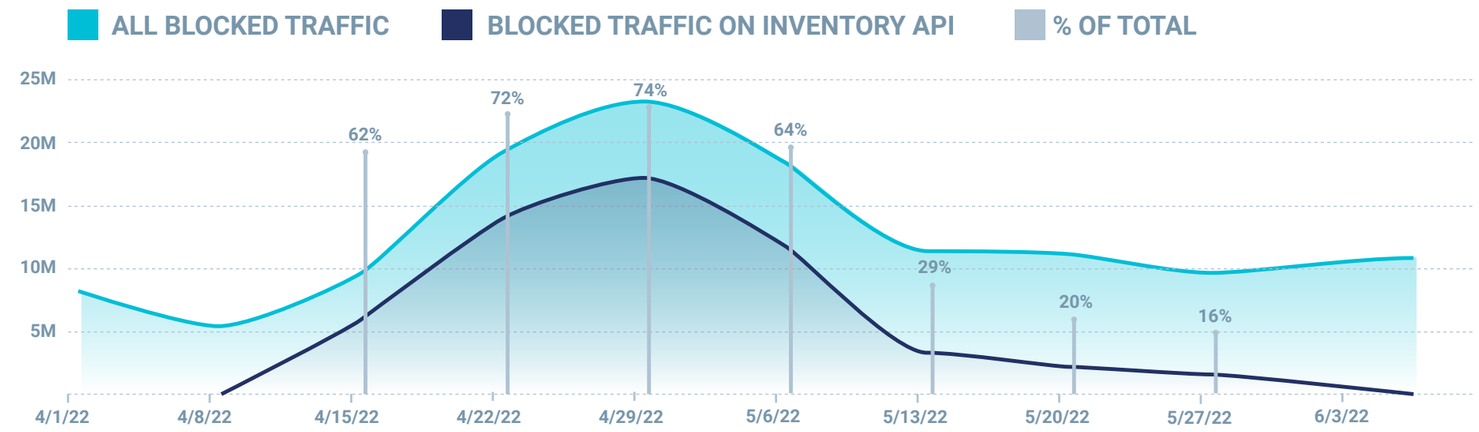
# Physical and Digital Worlds Collide

In the first half of 2022, the CQ Prime Threat Research team saw two compelling examples of physical and digital worlds colliding. In one example, a third-party location-based inventory API used to help Ulta Beauty customers find the desired products for purchase at the store or via curbside pickup was hit with a high volume, globally distributed content scraping attack. In the second example, one of the largest telecom providers in the world was hit with an API enumeration and account takeover attack (ATO) with the end goal of SIM swapping.

# Blocking an Inventory API Enumeration Attack Saves $80,000

A local-inventory search API at Ulta Beauty was hit with traffic volume 700X larger than average load. It originated from high-quality residential proxy IP addresses rotating through more than 153,000 unique product and SKU combinations while scraping 61,000 ZIP codes and 33,000 products. These proxies rendered traditional web application firewall (WAF) and CDN mitigation efforts ineffective. With the attack volume increasing, the inventory search API supplier notified the Ulta Beauty security team of the sudden traffic surge, requesting help to stop the attack or to renegotiate their financial terms. The investigation mapped the attack to OWASP API4 (Lack of Resources and Rate Limiting) and API5 (Broken Function Level Authorization).

**ENUMERATION ATTACK ON LOCAL INVENTORY CHECK API**




The CQ Prime Threat Research team worked closely with the Ulta Beauty security team to put policies in place to block 85.9 million total requests resulting in $80,000 saved in infrastructure and loss prevention. At the height of the attack, policies were blocking upwards of 17 million requests as shown in the chart above. The attack exhibited the following behaviors:

**Aggressive enumeration:** The attack cycled through ZIP codes to find areas with a greater concentration of products with higher retail values impacting resources and infrastructure.

**Web-to-mobile shift:** Initially, attackers targeted the web API, but quickly pivoted to the analogous mobile API which provides similar information.

**Direct-to-API:** The attack was designed to target the inventory API directly, without hitting any other app or web function. Normal behavior would show the user traversing multiple APIs.

**High volumetric threshold:** The attacker used enumeration to rotate through the inventory at such a volumetric rate that it represented 90% of ALL the customer traffic at the time.

**Outdated browser used:** The attack was built to use very outdated or anomalous versions of Chrome.

**Single cookie generation:** Each attack generated a single cookie whereas normal users would generate upwards of 40-50 cookies as they browsed the inventory.

# SIM Swapping – Stealing Your Phone Digitally

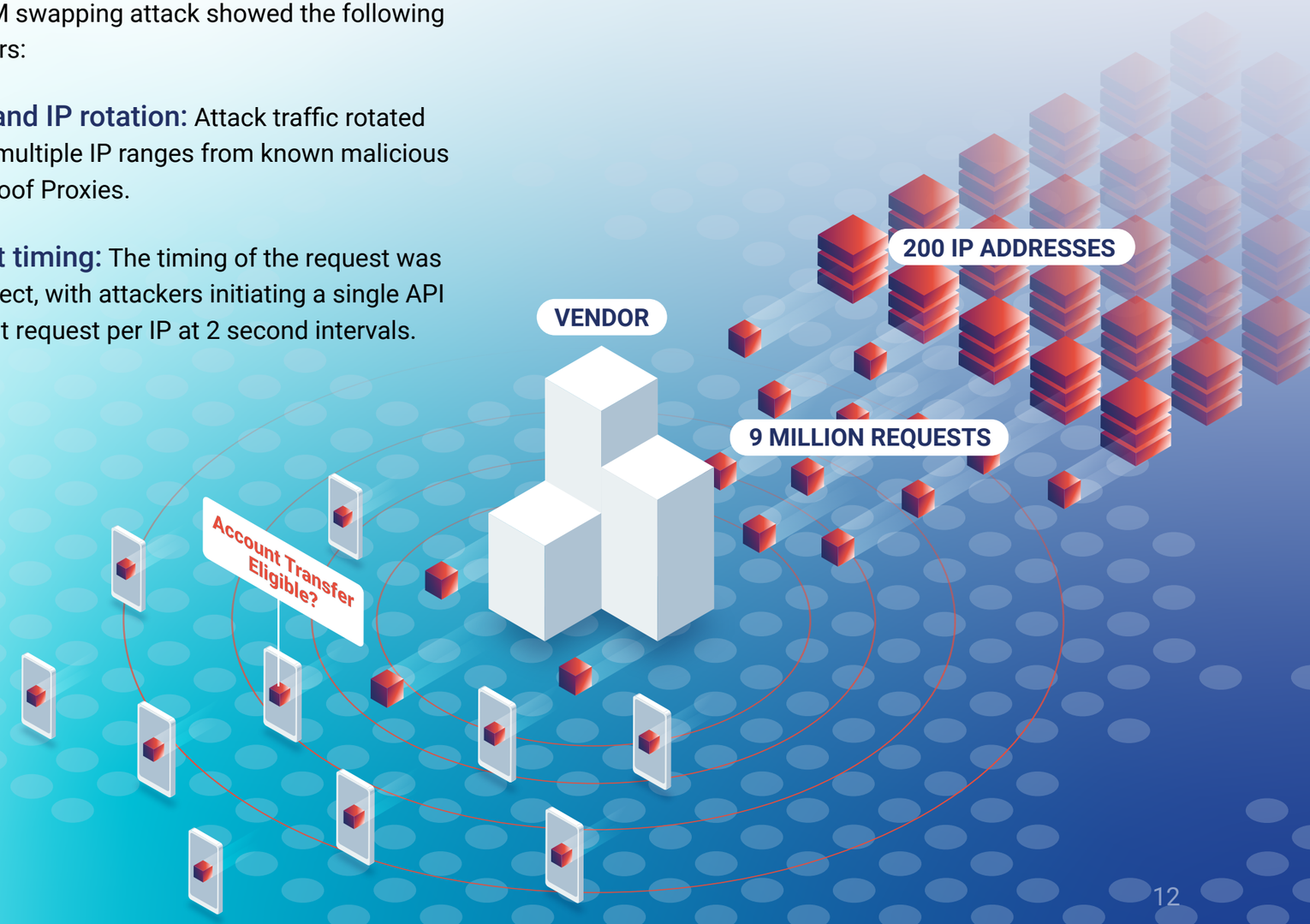
SIM swapping attacks are a new and growing class of threats where a telecom provider's 'port-a-phone' functionality is compromised to intercept sensitive information like one-time passwords (OTP) or telephone calls meant for the victim. OTPs are the keys to the kingdom when it comes to banking, payments, social and all things digitally stored in mobile devices, making a successful SIM swapping attack very lucrative for an attacker. According to an **IC3 report**, the number of SIM swapping complaints skyrocketed to 1,611 in 2021 from roughly 106 the previous year – a 15X increase. The same report noted that the financial impact measures more than $68 million in annual adjusted losses, averaging roughly $42,000 per incident.

A Fortune 500 telecom customer saw 9 million malicious requests from 200 residential proxy IP's that were flagged and blocked over a 10-hour period. The goal of this attack was to exploit OWASP API1 (Broken Object Level Authorization) to obtain information about a customer account by enumerating whether cell phone numbers could be transferred to the provider's network. Once the attacker discovered transferable phone numbers, they attempted to impersonate the true account owner to manipulate an employee into transferring the cell number onto a SIM card in the attacker's possession. From there, the attacker can then take control of the victim's sensitive accounts by completing SMS based two factor authentication. This SIM swapping attack showed the following behaviors:

**Proxy and IP rotation:** Attack traffic rotated across multiple IP ranges from known malicious Bulletproof Proxies.

**Perfect timing:** The timing of the request was too perfect, with attackers initiating a single API endpoint request per IP at 2 second intervals.

**High IP to request ratio:** Each phone number was identified to have been attempted from over 200 IP addresses.



# OWASP API Security Top 10 to CWE Mapping

Ultimately, all OWASP API Security Top Ten threats map to a set of Common Weakness Enumeration (CWE) vectors depending on the attack. APIs are simply a new channel through which adversaries attempt to exploit weaknesses in enterprise applications, and oftentimes the new tricks look a lot like old nemeses. As defenders, using a common taxonomy to put these threats into context will help evangelize API Security to executives. In the case of this SIM Swapping campaign, a detailed analysis identified the following OWASP API Security Top Ten and Common Weakness Enumeration (CWE) vectors.

## OWASP API1

**Broken Object Level Authorization**
Broken Object Level Authorization - Insufficient validation of an object access request allows an attacker to perform an unauthorized action by reusing an access token. **Learn More**

### CWE-284

**Improper Access Control**
A failure to restrict or incorrectly restricts access to a resource from an unauthorized actor.
**Learn More**

### CWE-1230

**Exposure of Sensitive Information Through Metadata**
Direct access to a resource containing sensitive information is blocked, but it does not sufficiently limit access to metadata that is derived from the original, sensitive information.
**Learn More**

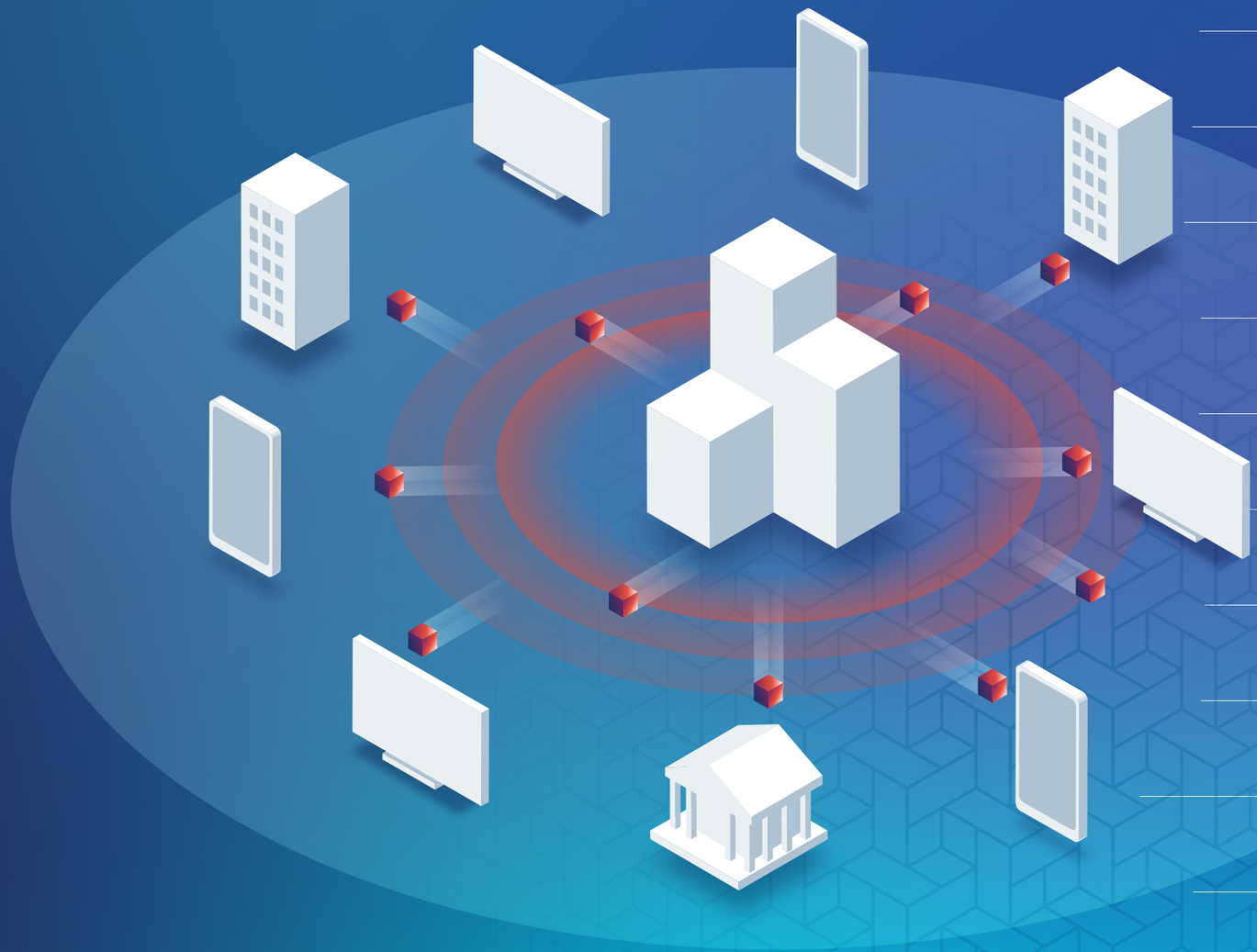
### CWE-202

**Exposure of Sensitive Information Through Data Queries**
When trying to keep information confidential, an attacker can often infer some of the information by using statistics.
**Learn More**

# Ecosystem and 3rd-Party APIs Under Attack

Long before APIs exploded into mainstream use, organizations used them to support and grow their partner ecosystem. Third-party APIs enable app-to-app connectivity for functions such as log aggregation and analysis, and popular consumer banking app integrations to enable cross-account visibility. They tie customer relationship management (CRM) and marketing automation together and provide a wide range of services across many industries.

The downside of third-party APIs is that they are often brought into the organization without security teams being aware of them. In other cases, their implementation may have subtle flaws that can be exploited. Even perfectly coded and implemented third-party APIs can be a target for threat actors hiding in plain sight. Furthermore, as certain third-party ecosystems succeed and grow, by nature they become a valuable single point in the digital supply chain where an attacker could use an API to target many victims.



# Partner Ecosystem APIs: A Target Rich Environment for Bots

The CQ Prime Threat Research team helped a financial services customer mitigate a coordinated credential stuffing campaign where attackers were abusing a third-party API endpoint to simultaneously execute credential stuffing attacks against multiple financial institutions. The targeted third-party API enabled a handful of consumer-friendly functionalities such as cross-account visibility and tracking, retirement planning, and net-worth tracking. Attackers were aware of this one-to-many connection in the institutions' supply chain and they knew that these API calls likely came from allowing listed infrastructure in the eyes of their ultimate target – the banks themselves. Therefore, instead of attacking the bank directly, the attackers targeted the API ecosystem and by proxy, the partner banks themselves. The attack was identified and blocked based on the following behaviors:

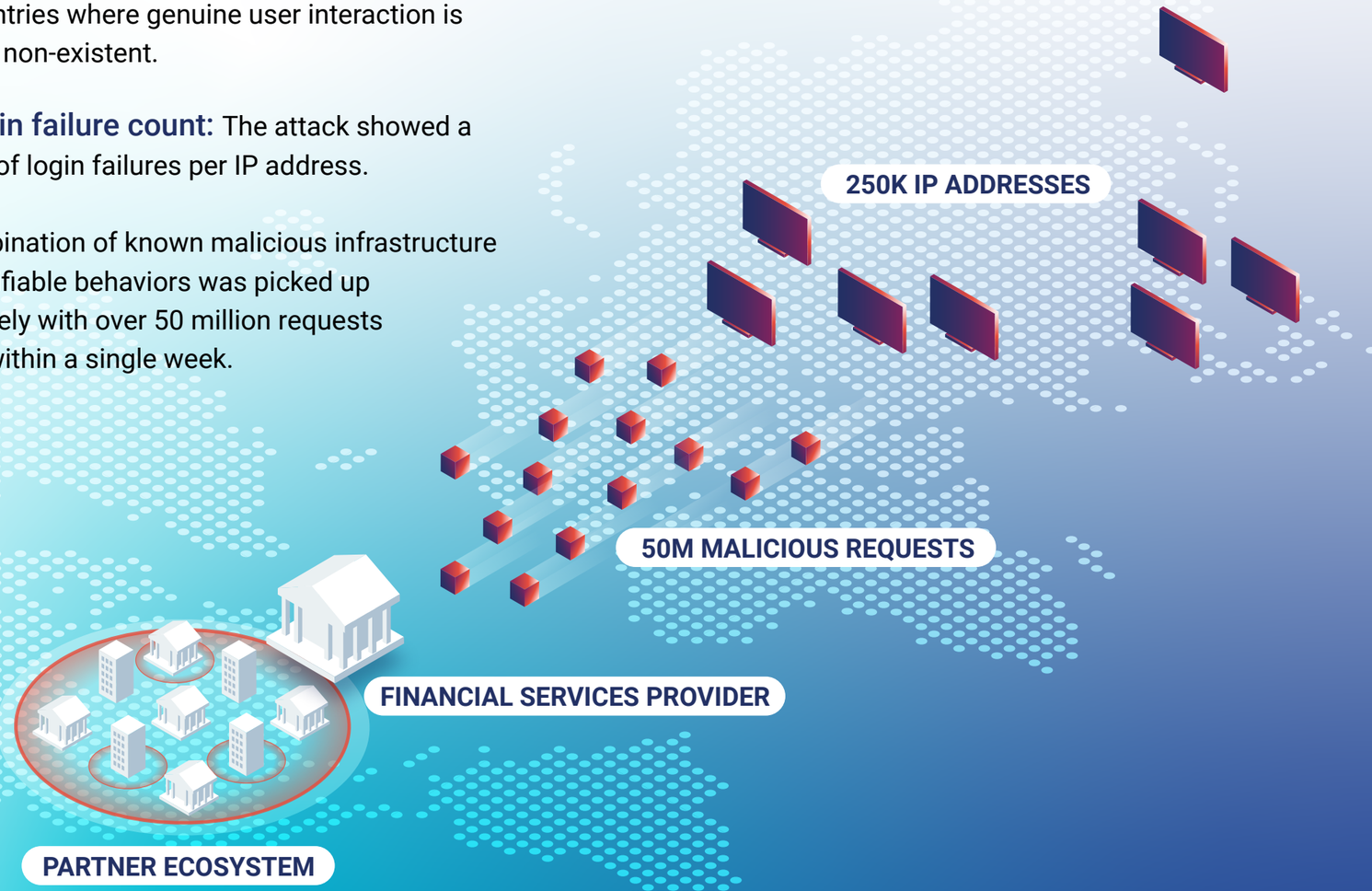
**Known malicious infrastructure:** Traffic originated from 250,000 geo-distributed residential proxy IP's belonging to Bulletproof proxy providers in the Middle East, North Africa, Korea, Russia, the Philippines and Indonesia.

**High session rotation:** Each IP address showed high session rotation.

**No business dealings:** Traffic was originating from countries where genuine user interaction is limited or non-existent.

**High login failure count:** The attack showed a high rate of login failures per IP address.

The combination of known malicious infrastructure and identifiable behaviors was picked up immediately with over 50 million requests blocked within a single week.



# Apple Pay API Abuse

Margins in the resale markets have tightened along with a general decrease in asset prices across stocks, crypto currencies, and trading cards. While the tighter margins have weeded out some of the attackers, it has also made those remaining in the game much more dedicated to finding a competitive edge against others. To that end APIs such as Apple Pay are a perfect target for attackers, as they are designed for a frictionless purchasing experience for humans, which bots will readily abuse.

Certain bot tools and cook groups aim to limit their membership to a small group of users. This ensures that they do not tip their hand on second-rate products and gives defenders an insight into their strategy. They effectively save their bullets for the most important launches. During a 3-hour launch for a highly anticipated sneaker, a large footwear and apparel retailer detected and mitigated a bot attack that was 50X normal, with 200 million API requests coming from roughly 6 million unique IP addresses.

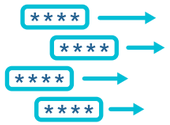
The attack did not end there with the next phase incorporating one of the aforementioned attack secrets. One cook group had discovered the existence of a shadow API which invoked the Apple Pay functionality on the retailers' platform. To avoid detection for as long as possible, the attackers held onto this trick until the last minute, and as soon as the launch began, the (shadow) Apple Pay API was hit with more than 100 million malicious API requests, all from high-quality residential proxies. While the attack was successfully mitigated using shadow API specific ML models and policies, it highlights how third-party payment APIs (e.g., Apple Pay, Google Pay, PayPal, etc.), typically managed by the business groups can fall outside of the security team's view. These APIs are coded correctly, but their shadow classification, and lack of protection makes them more susceptible to an attack.

# API10+: API Business Logic Abuse

The CQ Prime Threat Research Team worked to mitigate a financially motivated attacker targeting e-commerce platforms abusing OWASP API5 (Broken Function Level Authorization) which the attackers used to automate the purchase of items with stolen credit card data. The API abuse characteristics showed that the attackers were methodical in their efforts.

### Vulnerability Scanning

The attackers began by mapping the entire site using commonly known vulnerability scanning tools from a single IP address. This included some basic behaviors like SQL injection, command injection (OWASP API8), directory traversal, and searching for exposed sensitive files. When basic recon did not yield any low-hanging fruit, the attacker moved toward mapping the API ecosystem.




### Attack Probes

The attackers then began using existing attack configurations from well-known bot automation tools like OpenBullet to perform basic credential stuffing and account creation attacks. During a 24-hour period, attackers initiated more than 1.5 million requests from 130,000 IP addresses, all of which were mitigated by more than 1,000 different behavioral fingerprints.


### Continued Reconnaissance

The attack continued even as it was mitigated, leading to the discovery that this was ultimately a head-fake from the attackers and was not the goal. During the following attacks, the reconnaissance behavior returned, this time focusing on account creation and checkout APIs.

### Vulnerability Discovered

Attackers discovered that upon creation of a brand-new account, and before email verification had taken place, that the checkout APIs (particularly those to add a payment method) could be invoked by the user. This is an example of broken function level authorization, where an API functionality is intended to be used only by users who have both authenticated and are authorized.

### Theft

The focus of the attack shifted to account creation, and attackers immediately began stuffing new (fake) accounts with stolen payment info, targeting retail products for purchase. They did not care their credential stuffing campaign was failing, they were simply watching which of the new accounts they created would be able to successfully access payment APIs, iterating through stolen credit cards until they found one eligible to continue with the purchase.

1 2 3 4 5

# Conclusion

APIs have been in use for many years; however only recently have their use cases emerged from deep within IT to being the cornerstone of a business. This shift makes API protection a key initiative for the business, not just the business unit or security and development teams. As shown in this report, APIs are under attack from many different vectors and these attacks have a direct impact on a company's bottom line driven by lost customers, brand damage, IT infrastructure cost overruns, compliance violations and more.

The view that API protection can be addressed by a shift left, development focused effort with the OWASP API Security Top 10 list is a start, but as the report shows, threat actors do not adhere to a top 10 list, and perfectly coded APIs are susceptible to attacks. Alternatively, protecting APIs is not solely the responsibility of the security team. Here too, the high volume of attacks on shadow APIs highlights the obvious – you cannot protect what you cannot see. API protection needs to be treated holistically, with a uniform approach that begins with discovering, identifying, and inventorying your API footprint. Once the API estate is known, continuous risk analysis can be performed to uncover and remediate sensitive data, authentication or specification non-conformance related coding errors for production and non-production APIs. This middle phase of the API protection journey also incorporates runtime attack detection. Employing countermeasures such as real-time blocking or deception without the need for added third-party data security tools, combined with ongoing testing to ensure risky APIs do not go live make up the last phase.

Get a free API security assessment at **cequence.ai/assessment**.