CYBEREDGE GROUP

THREAT BRIEF

Understanding API/Business Logic Abuse and Defense Requirements

WHAT IS API/BUSINESS LOGIC ABUSE?

The use of an API or business logic contrary to its intended use, typically for malicious purposes.

Businesses today operate in a hyper-connected, Internet-driven economy. And it's driving the need for an extended, API-linked application architecture. Internal Service Oriented Architecture (SOA), corporate BYOD initiatives, and the Internet of Things (IoT) are just a few of the use cases for APIs. Organizations also use APIs to facilitate interoperability between web applications to support and optimize critical business processes, and to improve customer engagement experiences via new features and capabilities.

Unfortunately, the proliferation of APIs and business logic in web and mobile applications also increases the organization's risk exposure. These technologies provide visibility into operations and allow access to data. Despite that, some consider APIs to be more secure and don't apply security on the backend. Still others find that applying consistent security measures is complex and can impact the user experience, so they neglect those efforts.

WHAT'S AT RISK

- Productivity of your security and fraud analysts, customer service, and IT
- Reduced app usage by legitimate customers
- Decline in revenue
- Brand reputation
- Financial fraud
- Customer and stakeholder trust
- Confidential and sensitive data

Attackers use APIs or business logic to steal data, or to commit server-based or application-level attacks.

HOW IT WORKS

API and business logic abuse are business-level attacks. Attackers leverage legitimate functionality against the business itself. In this case, attackers use APIs or business logic to steal data, or commit server-based or application-level attacks. APIs can also be abused by competitors for IP theft or scraping of business-critical information.

Before they can conduct an attack, attackers must download the application and reverse engineer the API or business logic to determine how it works. Once they identify how to abuse the API or business logic, they program bots to do so. The bots operate mobile APIs and send API requests. This activity can take place in a "low and slow" manner so as to blend in with legitimate traffic, or many requests can be sent simultaneously, resulting in disruption of services.

API/BUSINESS LOGIC ABUSE DEFENSE CHALLENGES

A lot of companies mistakenly believe that APIs are inherently secure because they are meant to be used by trusted partners or aggregators. However, attackers discover these APIs and use them when they are challenged on the web and mobile channel.

Once companies understand the need to defend against API and business logic abuse, they find it difficult to do so using legacy security tools. Attackers leverage legitimate access and input values, so Web Application Firewalls don't recognize that the traffic is malicious. Furthermore, it isn't always necessary for a user to log in to an application to gain access to APIs or business logic—so authentication mechanisms don't prevent these attacks. CAPTCHAs, while relatively easy to apply, are broken. Web application security defenses that only monitor the external perimeter don't see traffic on internal APIs. And bot detection is only effective if it can distinguish between good bots (like IoT devices) and bad bots.

Organizations may be able to detect API or business logic abuse after the fact by analyzing logs, but by then it may be too late.

REQUIREMENTS FOR API/BUSINESS LOGIC ABUSE DEFENSE

An effective solution for detecting and stopping API/business logic abuse must:

- Raise the effort required to abuse APIs/business logic (this serves as a deterrent)
- Block bad bots to prevent them from abusing APIs and business logic
- Understand the intent of application requests
- Deliver real-time protection against malicious bots
- Be a software-only solution; no need for expensive, purpose-built appliances
- Be deployable on-premises, in the cloud, across any number of locations
- Not rely on in line detection, which is limited and adds network latency
- Automatically discover all application assets across the organization
- Not require any application changes for detection
- Use an AI engine with machine learning to provide attack detection/analysis in real time
- Provide effective attack defense against all channels – web, mobile, and API apps
- Integrate with existing security tools, providing security teams with insight into attack details
- Stop attacks fast with customizable, automated mitigation policies
- Reduce infrastructure costs, financial loss, business disruption
- Allow legitimate bot or automation from partners or aggregators to access the API while blocking malicious bot activities

THREE KEY REQUIREMENTS FOR A BOT ATTACK:

1) Attack Tools

- Sentry MBA
- SNIPR
- Hitman
- Hydra
- Medusa
- Phantom JS
- CURL & WGET

2) Infrastructure Access

- Compromised devices
  - Business and personal computers
  - Home routers
  - IoT
- Underground VPNs
- Unscrupulous Internet Service Providers

3) Credentials (username–password pairs)

- Where are credentials acquired?
  A sampling of breaches in 2018 includes:
  - Under Armour, 150M records
  - MyHeritage, 92M records
  - Facebook, 50M records
  - British Airways, 380K records
  - T-Mobile, 2M records
- How are breached credentials sourced?
  - Dark Web supplier/seller e-commerce applications

Any company with a website or application is susceptible to API or business logic abuse. Companies must understand that they are at risk and that their traditional infrastructure security tools or first-generation bot defense solutions won't stop API or business logic abuse. These advanced attacks require innovative, advanced detection and mitigation technology.