LINK11

INVISIBLE ATTACKS, VISIBLE RISKS

Why web applications and APIs are being targeted and which security architecture makes the difference

www.link11.com

Executive Summary

WAAP and the new era of automated threats

In the last 5 years, the digital ecosystem has undergone a quiet but momentous paradigm shift: the web is no longer dominated by humans, but by machines. Bots, AI agents, and distributed automated systems now generate the majority of global HTTP traffic. For businesses, this means that the threat landscape has not only grown larger, but has also fundamentally changed. Traditional protection mechanisms no longer work. Web Application & API Protection (WAAP) is thus becoming the new key discipline of modern cyber resilience.

This paper shows how attacks have evolved from simple volume waves to highly orchestrated, multi-layered attacks. Web DDoS campaigns combine Layer 3/4 floods with precise Layer 7 attacks, systematically test defense mechanisms, and dynamically rotate their attack vectors. The case studies presented below paint a clear picture: independent attackers operate globally, adaptively, and with a level of professionalism that was previously reserved for state actors.

This is particularly evident in attacks where HTTPS endpoints are flooded with tens of millions of packets per second or millions of HTTP requests and sessions are generated simultaneously without leaving even a hint of suspicious payloads. Modern DDoS attacks mimic legitimate user behavior so convincingly that traditional defense mechanisms remain blind.
At the same time, a second, equally dangerous threat is growing: bot traffic. Today, more than half of all web interactions come from automated clients. Many of them pursue clear—and sometimes criminal—business models such as scraping, account takeover, price manipulation, or fraud. Others serve to prepare attacks, test APIs, probe vulnerabilities, or generate inconspicuous pressure on infrastructure and applications. Our case studies show that it doesn‘t always have to be loud; some attacks are barely measurable and yet just as (or even more) dangerous.

APIs in particular are coming under scrutiny. They drive modern business models, from banking to e-commerce, and often grow faster than security architectures can keep up. Shadow APIs, uncontrolled endpoints, or unsecured integrations open up new avenues for attackers. Companies often only notice attacks once data has already been leaked.

The key message is clear and compelling: WAAP is not a single tool, but a strategic framework.

A modern defense strategy must intelligently combine DDoS mitigation, bot management, behavior-based WAF, and API security. Only a learning, context-sensitive system can reliably distinguish between legitimate users, harmless bots, malicious actors, and sophisticated attack scenarios.

The cases analyzed in the paper show that security today is less about “blocking” and more about understanding. Those who recognize the intent behind a request protect not only their systems, but also their entire digital business model.

Modern WAAP platforms such as Link11‘s therefore rely on AI-powered detection, forensic analysis models, adaptive mitigation, and a robust technical infrastructure that works accurately even under extreme load. The result is fewer false alarms, greater transparency, higher availability, and true digital resilience.

The evolution of web DDoS attacks – From a sledgehammer to surgical precision

The new era of web DDoS: precision instead of brute force

DDoS attacks have changed radically in recent years. What used to be simple volume attacks is now a complex, multi-layered field of attack. Modern campaigns typically combine network, transport, and application layers and use highly automated and tactically controlled global botnets.

The techniques range from classic amplification attacks (DNS, NTP, CLDAP) to Slowloris, API flood, or multifactorial attacks that operate simultaneously on multiple OSI layers. Particularly striking is the ability of modern attackers to dynamically adapt to countermeasures. Off and on phases, changed user agents, rotating IP pools, and variable payloads are just a few examples.

Web DDoS is therefore no longer purely a question of volume. The threat encompasses everything that disrupts the availability of web services—regardless of the level at which the attack begins. Hybrid attacks are increasingly used to deliberately blur the boundaries between layers.

The following case studies from the Link11 network illustrate this development. They show how varied and complex attack patterns can be. Each individual attack provides valuable insights for the further development of protection strategies and shows that classic defense mechanisms are no longer sufficient.

Typology of DDoS attacks according to OSI layers

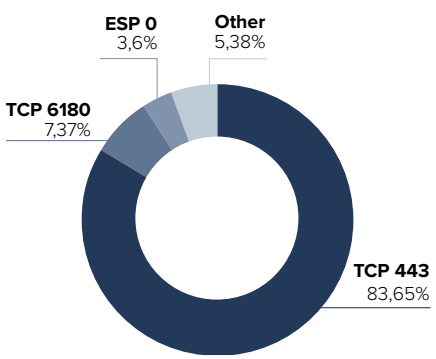
OSI layer	Typical forms of attack	Target	Relation to web services
Layer 3 (network)	ICMP-Flood, IP-Flood	Network overload	Impairs routing and accessibility of web servers
Layer 4 (transport)	TCP-SYN-Flood, UDP-Flood, TCP-Connection-Flood	Block connection establishment, exhaust resources	Direct attacks on web ports (e.g., TCP 443 for HTTPS)
Layer 7 (application)	HTTP flood, Slowloris, API attacks	Paralyze web servers and applications	Targets web functions, forms, APIs, search endpoints
Hybrid attacks	Combination of layers 3/4/7	Multidimensional overload	Bypasses traditional protection mechanisms through hybrid forms

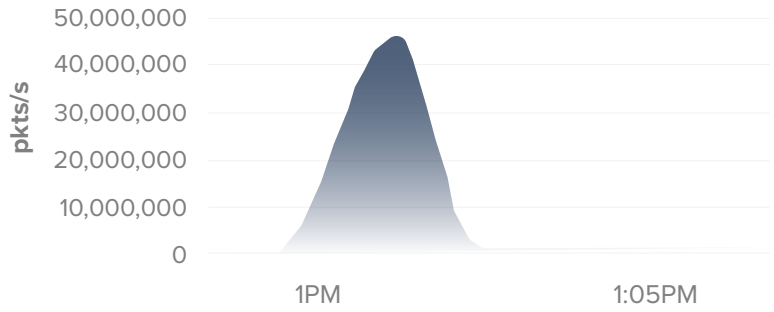
Case Study 1

Extreme TCP packet rate against port 443 – An attack with a double effect

In this campaign, a HTTPS endpoint was attacked with up to **45 million packets per second**. The unusually large packet sizes of around 1,400 bytes were striking—a deliberate combination to put pressure on both bandwidth and server CPU at the same time.

Add target ports of the attack and packet rate per second





Technical details:

- Globally distributed botnet with over 30,000 compromised devices (China, Russia, Vietnam, etc.).
- Packet fragmentation to bypass classic firewalls and IDS systems.
- Simultaneous flood on network resources and CPU-intensive TCP connections.

Strategic insights:

- Traditional bandwidth filters are not sufficient: Layer 4 detection, packet reassembly, and deep packet inspection are necessary.
- Real-time analytics and automated response mechanisms are crucial for detecting fragmented and high-frequency traffic.
- Companies should have contingency plans in place with ISP and cloud scrubbing integration to enable rapid scaling in the event of packet floods.

Why Layer 4 attacks are part of web DDoS

Although Layer 4 attacks are technically “below” the application, they directly affect modern web systems.

- Port 443 is one of the lifelines of HTTPS communication.
- Many web servers collapse not because of the amount of data, but rather because of thousands of simultaneous TCP handshakes.
- High packet frequencies block network paths, preventing the server from even deciding whether the traffic is legitimate.

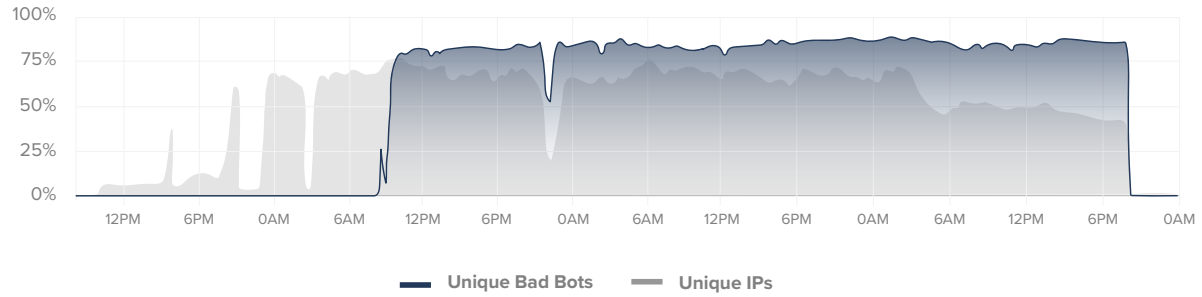
An attack on Layer 4 is therefore almost always an attack on the availability of a web application—regardless of whether Layer 7 is involved or not.

Case Study 2

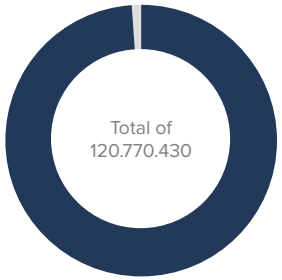
Hybrid attack – When all layers are under fire at the same time

Another campaign demonstrates the perfection of modern attacks: simultaneous Layer 3/4 floods and Layer 7 attacks with over **120 million HTTP requests** and up to 630 requests/second.

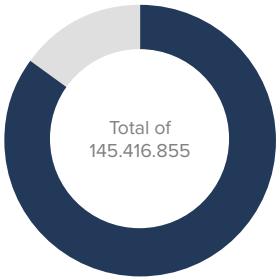
Bad bots cause high request numbers



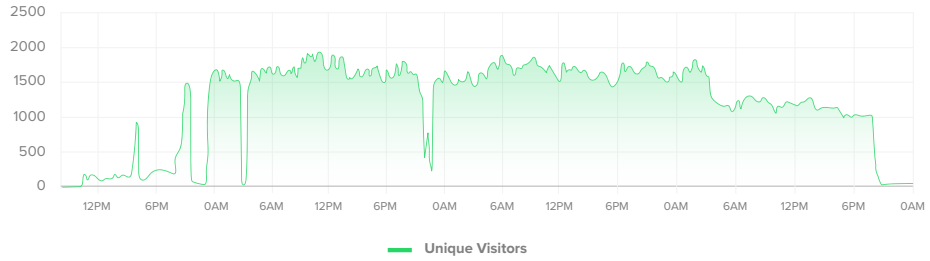
Bad Bot Requests



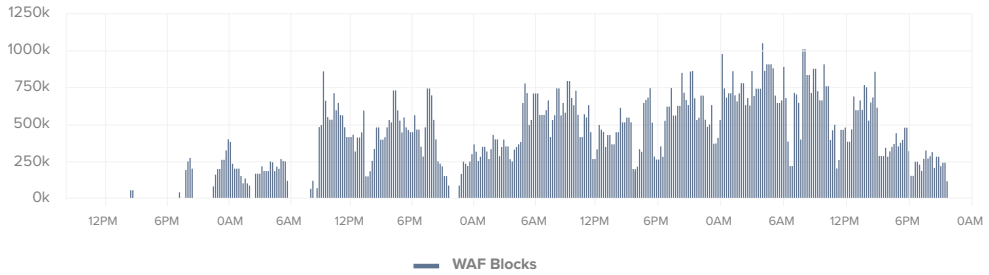
Distribution of requests



Unique IP addresses as an indication of a global botnet

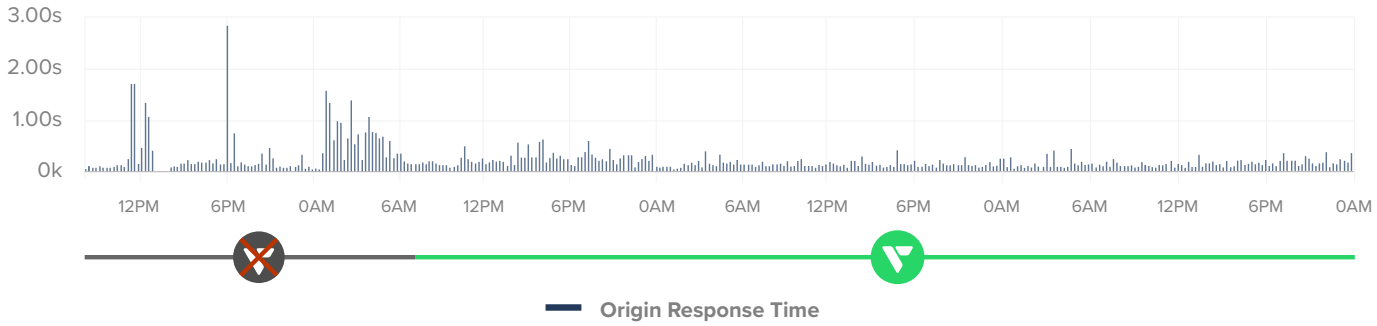


Several million WAF blocks during the attack



Case Study 2

Performance overview with and without protection.



Hybrid DDoS attacks are now increasingly part of attackers‘ toolkits. Holistic WAAP solutions that intelligently combine a web application firewall, bot management, DDoS mitigation, and API protection offer the necessary resilience against these complex threats.

Technical features:

- Simultaneous overload of network and application.
- 2,000 rotating IPs from global botnets.
- Dynamic on-off phases, manipulated user agents, targeted WAF probing patterns.
- Over two million WAF blocks in just 30 minutes.

Special features:

- Use of outdated user agents to disguise automated bots.
- Originating from unusual networks such as the Viettel Group, China Telecom, and Yemen.net, indicating high resources and a professional infrastructure.
- Suspected motives: Hacktivism, DDoS-as-a-service, or targeted testing to optimize future attacks.

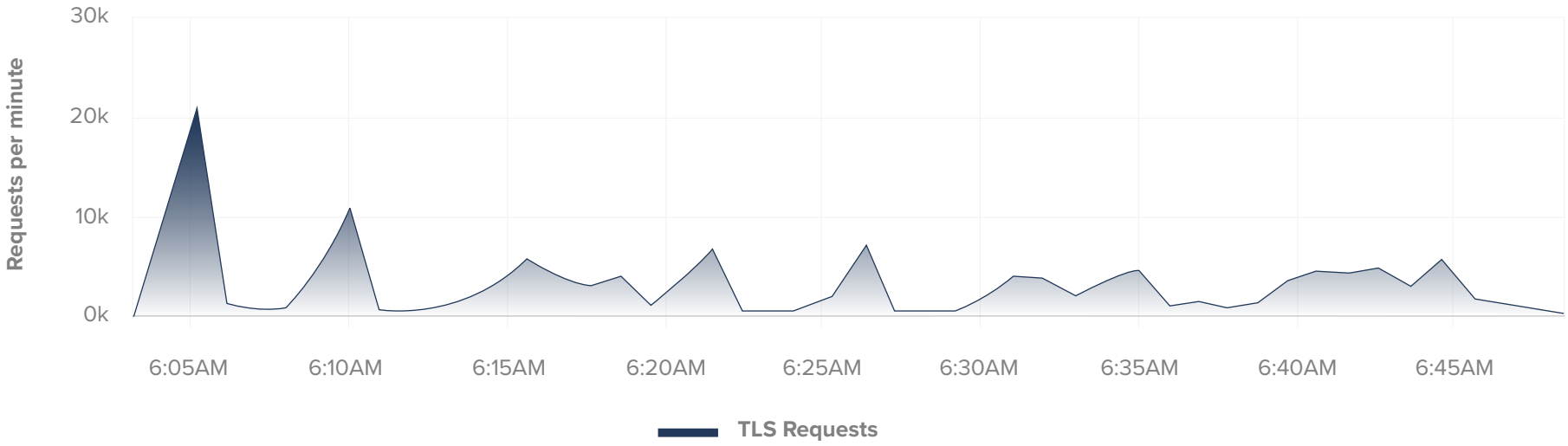
Strategic insights:

- WAAP systems must remain continuously active, as temporary deactivation during attack pauses is risky.
- Bot Management and WAFs must work closely together to detect and block adaptive attacks.
- API protection is essential, as APIs are increasingly the target of automated attacks.
- AI-supported detection helps to identify new attack patterns at an early stage and to respond with automated countermeasures.

Case Study 3

Geopolitical Layer 7 DDoS – Attack with a cloak of invisibility

Amid political tensions, pro-Russian hacktivists intensified their attacks on German institutions. Particularly striking was the seemingly legitimate traffic from German CDN and VPN nodes, which turned out to be structured Layer 7 DDoS.



Technical characteristics:

- Up to 20,000 requests per minute.
- Use of legitimate CDN/VPN infrastructure (Datacamp, M247).
- Automated camouflage through German edge nodes.

Detection and defense:

1. Blocking by modern protection systems due to inconsistent headers, missing referrers, and failed JS/CAPTCHA challenges.
2. The attack failed because human browsing behavior could not be fully simulated.

Strategic insights:

- Layer 7 attacks can cause significant damage with low volume.
- Camouflage via legitimate infrastructure is rapidly increasing.
- Behavioral detection is imperative.

Overarching patterns –
What characterizes modern DDoS attacks

1

Attackers rely on hybrid tactics

Multiple layers simultaneously, synchronously controlled.

2

Camouflage is becoming more important than volume

Technically inconspicuous requests, but with a significant impact.

3

Automation replaces manual control

Botnets react automatically to defensive measures.



For companies, this means:

- Behavior-based detection instead of purely signature-based filters.
- Automated mitigation across all layers.
- Proactive API protection mechanisms including schema validation and rate limits.
- Closely integrated WAF, bot management, and DDoS mitigation instead of isolated tools.
- Real-time analysis to immediately detect pattern changes.

Web DDoS is now a multidimensional problem that requires multidimensional defense. WAAP solutions are therefore not just a technical tool, but the foundation of modern cyber resilience for companies that want to protect their digital business models in the long term.

Bot Management – Man vs. Machine in the Modern Web

The silent takeover: bots as the new majority on the web

The internet is currently undergoing a quiet but profound revolution. For the first time in its history, the majority of traffic no longer comes from humans, but from machines.

More than half of global HTTP traffic is now generated by bots—from harmless crawlers to sophisticated AI-driven systems that simulate human behavior, probe API endpoints, or manipulate business processes.

This shift is particularly evident in live data: for many companies, the proportion of bots is permanently above 50 percent. Human traffic is becoming a minority phenomenon. This structural change forms the backdrop for everything that is happening today in the field of web, API, and DDoS security. This structural shift marks the beginning of a new era: one of automated actors.

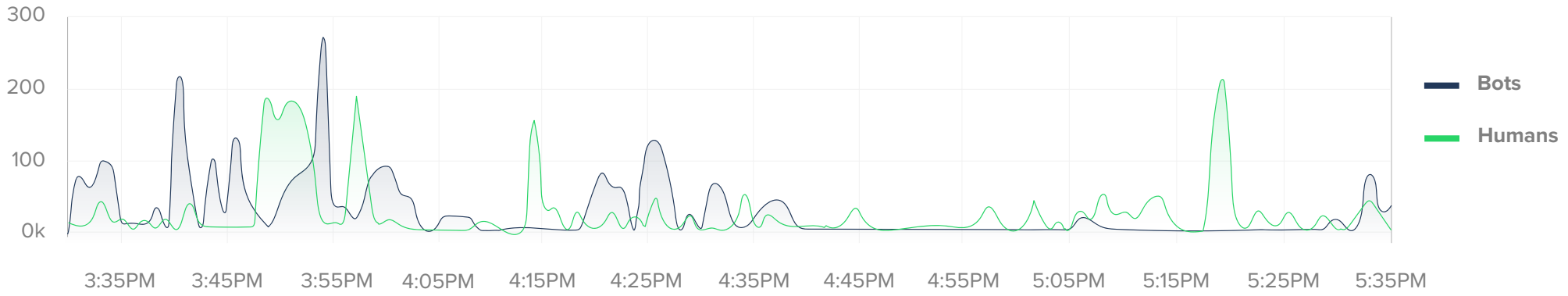
What really defines modern bots

Bots are no longer simple scripts that send requests at fixed intervals. Many of them operate with headless browsers, execute JavaScript, trigger cookies, rotate IPs, or analyze server responses using machine learning. Some imitate the behavior of legitimate users so convincingly that classic protective measures such as rate limits or CAPTCHAs are hardly effective anymore.

Their goals are diverse: they index content, extract data, automate transactions, take over accounts, or test the resilience of APIs. AI-based crawlers, which can semantically capture content and make autonomous decisions, are undergoing particularly dynamic development.

This increasingly blurs the line between useful and harmful, and organizations quickly lose control over who uses their data and for what purpose.

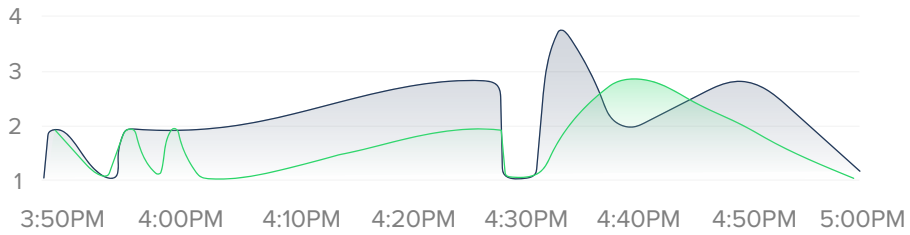
Comparison between human and automated traffic for a Link11 customer



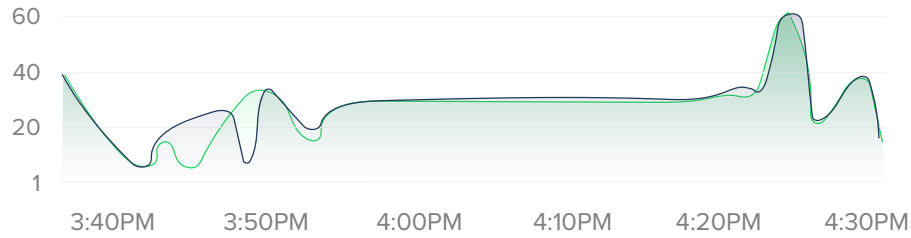
How bots work today –
and why traditional defense mechanisms fail

Modern bots are programmed to remain inconspicuous. While human users interact at irregular intervals, with pauses, page changes, and scrolling movements, bots act at precise intervals and with perfectly reproducible patterns. However, they can also deliberately conceal this behavior; for example, through variable timing strategies or adaptive rebalancing as soon as an endpoint responds to defensive measures.

Request frequency per user IP address



Bot request frequency



In addition, there are global proxy networks that enable attackers to use short-lived IP addresses. This tactic effectively undermines IP-based filters. At the same time, headless browsers have the ability to pass challenge mechanisms that were previously considered secure. Even CAPTCHAs can now be solved automatically.

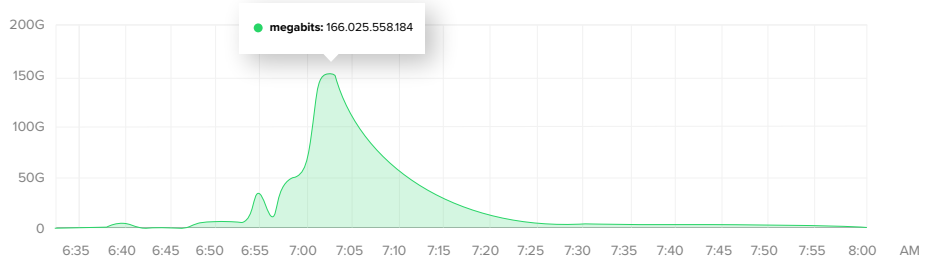
The effect is always the same: systems based on static rules or simple thresholds are becoming increasingly blind.

Case Study 4

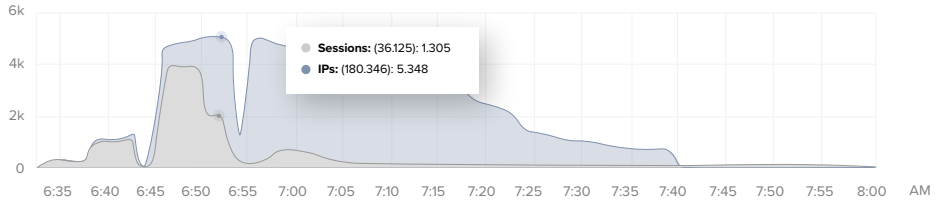
When a global botnet “gently” overloads a platform

In a typical attack scenario, traffic on a business platform initially increased only by a small amount. As an indicator of a pattern, this is often underestimated in the early stages.
But behind the inconspicuous peaks were 16 million requests per minute generated by several thousand IP addresses. Almost all requests carried the same user agent, indicating an automated origin.

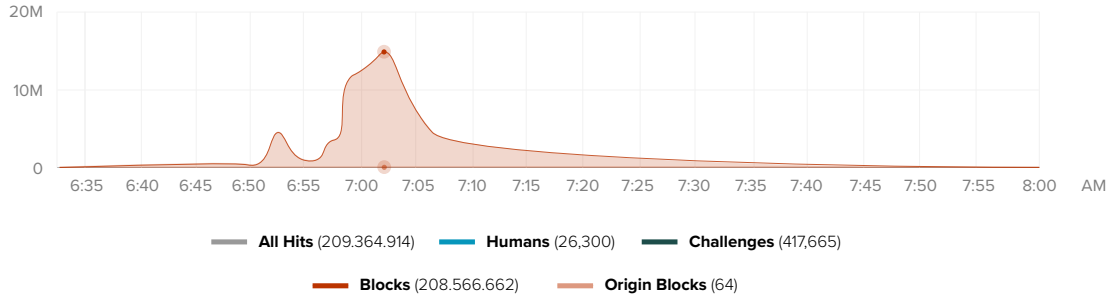
Total bandwidth



One-time sessions & IP



Passed vs. blocked



It is interesting to note how the bots behaved: after the first IPs were blocked, the botnet adjusted its distribution, shifting traffic to other regions or increasing the frequency there. The attack was not loud, but it was persistent.

This example shows that large botnets do not necessarily require large amounts of data. Even a small volume, if perfectly timed and highly parallelized, can consume critical resources.

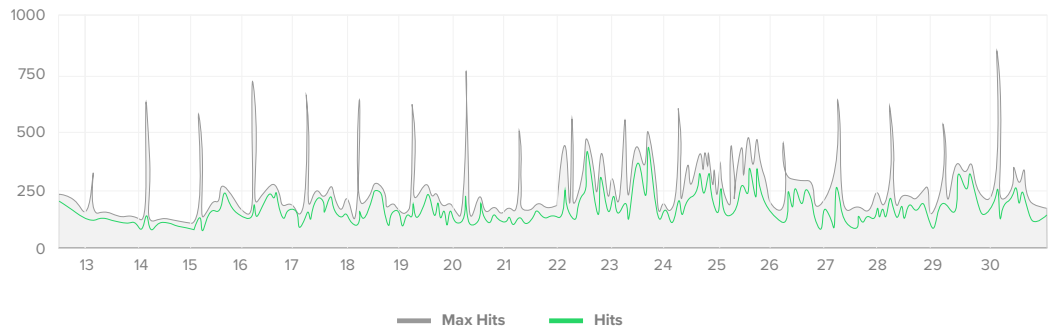
Effective bot management separates legitimate traffic from automated traffic before resources are exhausted.

Case Study 5

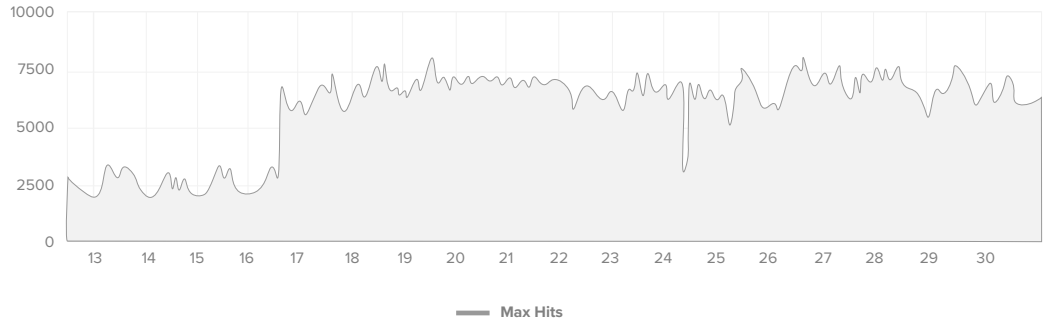
The puzzling rise of “unique” IPs

Another example illustrates how covertly modern bot campaigns operate. The number of unique IPs doubled from around 2,000 to 8,000,
while bandwidth and request volume increased only moderately. At first glance, this seems harmless, but server latency increased and 500 errors repeatedly occurred on.

Requests per day



Unique IPs per day



Strategic insights:

- Classic monitoring metrics (bandwidth, request rate) can be misleading.
- Without bot analysis, such anomalies remain invisible. They are “blind spots” in the network.

The reason for this is that many small, short-lived connections create a load that is not visible in traffic volume or classic metrics. In addition, the proportion of POST requests increased significantly, indicating form and search function abuse. The origin of the traffic from specific cloud providers also matched the typical pattern of an automated network.

This combination shows a typical botnet pattern: many short-lived connections, hardly any data, but a high server load. Real browsers keep sessions open and bundle requests, whereas bots constantly open new TCP connections without keep-alive.

Why traditional bot defense fails

Many common protection methods such as IP block lists, simple rate limits, or CAPTCHAs date back to a time when bots were still very simple in design. Today, however, attackers are able to circumvent all of these mechanisms. CAPTCHAs are solved automatically, IPs rotate every second, headers can be falsified at will, and headless browsers convincingly simulate human interactions.

Although reCAPTCHA v3 and similar invisible systems reduce friction for users, they lead to higher costs and data protection risks, and advanced bots still slip through. The consequence is clear: signature- or rule-based systems are simply no longer sufficient. Modern bot defense must therefore be behavior-based, adaptive, and context-sensitive.



The new generation of bot detection: From rules to learning systems

Modern bot management solutions rely on a multi-layered analysis model. The first step is behavioral analysis, which looks at patterns such as typing speed, scrolling behavior, session rhythm, and client consistency. In addition, client-side certificates, SDK-based integrity checks, and server-side validation processes are used.

APIs are subject to additional requirements. Without browser fingerprinting, replay detection, or payload validation, the API remains an open gateway. However, the best defense is not a perfect filter, but rather a learning system that minimizes false positives while reliably detecting malicious bots.

This interplay of analysis, automation, and continuous feedback forms the basis of modern bot governance and is therefore an essential component of cyber resilience.



Bot governance as a strategic factor

Bot management has long since ceased to be a purely technical task. Today, the central question is: “Who is allowed to access my digital systems automatically, and under what conditions?”

To clarify this, companies must first create complete transparency about their automated interactions. On this basis, they must define a clear policy for permitted and unauthorized bots, set up technical layers of protection, and continuously measure their effectiveness.

WAAP solutions with integrated bot management form the technical foundation for this, but the combination of technology, expertise, and governance is crucial. This is the only way to regain control over automated traffic—and thus over a central part of the digital infrastructure.

Modern WAF: From static rules to intelligent analysis instance

Why classic WAFs are no longer sufficient

For a long time, web application firewalls were the central tool for protecting web applications from known attacks. But the attack landscape has evolved faster than these systems. While traditional WAFs rely on signatures, fixed rules, and static heuristics, modern attackers are flexible, adaptive, and highly automated.

APIs have become the backbone of digital business models and now generate more than 80 percent of web traffic. At the same time, bots and AI agents dominate the internet. Many attacks are no longer loud and conspicuous, but subtle and carefully disguised. They are able to test protective mechanisms and circumvent them in a targeted manner.

Companies are confronted with three structural challenges as a result:

1. The volume of automated traffic is so high that rule-based systems can hardly distinguish between what is legitimate and what is harmful.
2. Complex microservice architectures and fast-paced DevOps cycles change APIs and endpoints at ever shorter intervals. Static rules become obsolete within a few days.
3. Operational teams are under pressure. False alarms, unclear blocking reasons, and manually maintained rulesets put a strain on SOCs, especially during peak times—similar to an air traffic control desk that becomes confusing during a storm.

Adding to this is an increasingly stringent compliance environment. Today, companies must not only defend against attacks, but also be able to prove why certain accesses were blocked. Auditability, transparency, and traceability are mandatory.

All of this leads to one clear conclusion: the era of purely rule-based WAFs is over.

How modern WAF systems work – and why they need to be smarter

Modern WAFs no longer function like filters that recognize patterns, but rather like analytical entities that interpret behavior. They combine various levels of detection. Instead of simply checking the content of a request, WAFs include new strategies:

Pattern recognition to distinguish bots from legitimate users: Timing, session rhythm, header consistency, deviations in the client profile, or unusual navigation patterns. This approach allows automated bots that attempt to imitate browser logic to be identified.

Each API endpoint has its own context and path logic: A login behaves differently from a search form or an order process. Modern WAFs recognize whether a request fits into the expected context, whether it is semantically coherent, or whether it represents an abuse attempt.

Machine Learning: ML models recognize connections that humans or static rules cannot detect. For example, clusters of anomalies in user agents, atypical correlations in TLS fingerprints, or repetition patterns that only become visible over millions of requests.

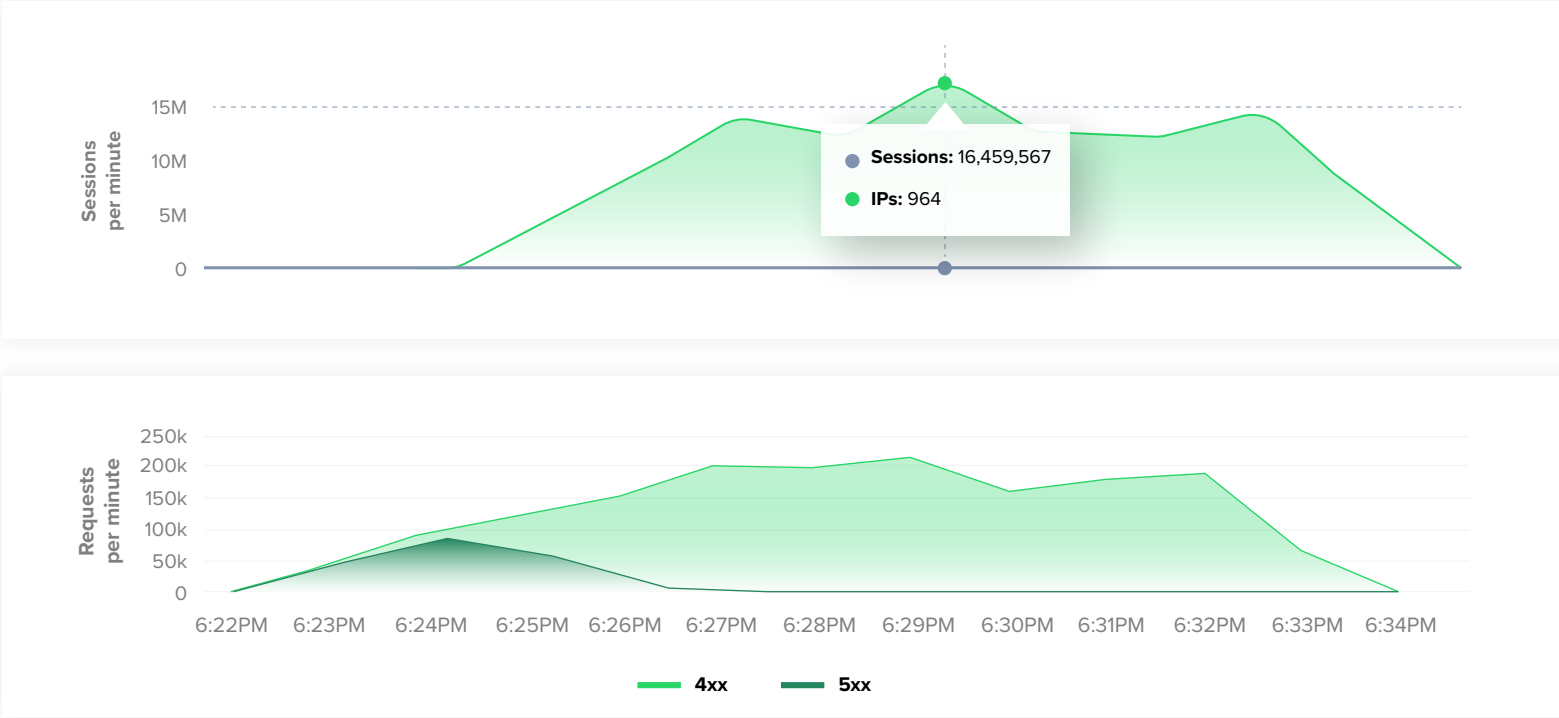
These levels complement each other. The WAF does not make decisions based on a single characteristic, but rather on the basis of a probability calculated from many signals. This makes blocking decisions more precise and comprehensible, which is a decisive advantage for security and compliance teams.

Case Study 6

Silent Layer 7 DDoS through session flooding

An attack on a European platform exemplifies how dangerous “quiet” traffic can be. The attack did not rely on large amounts of data or complex payloads, but on an enormous number of sessions: **16 million new sessions per minute,** a total of over **580 million HTTP requests,** and around **1,000 IP addresses** that dynamically adjusted their distribution. Each individual request was technically inconspicuous—simple GET requests to the root domain.

However, the sheer volume of sessions overwhelmed the backend: the CPU, RAM, and session handling nearly collapsed. Such a scenario would have been difficult to detect with traditional DDoS sensors, as the traffic volume was not exceptionally high.



Detection and defense measures:

- The WAF identified unusual access patterns, high session density, and non-human behavior.
- Suspicious IPs were automatically isolated and additionally blocked using Layer 3 protection.
- A quarantine mechanism prevented repeat attempts by already detected sources.

Special features of the attack:

- The botnet acted adaptively: as soon as IPs were blocked, new ones were reloaded.
- The attackers simulated legitimate traffic, thereby circumventing classic filter mechanisms.

This case study shows how dangerous attacks that occur below the usual alarm thresholds have become as well as why a modern WAF must understand not only content but also behavior.

WAF as a strategic component of modern security

Modern WAFs are more than just a technical protective wall. They are becoming strategic components of WAAP architecture because they explain decisions, make risks measurable, and act close to the application.

Link11's platform architecture is a prime example of this: it is not based on a multi-tenant environment, but on isolated instances. This keeps latency predictable and ensures compliance with regulatory requirements such as ISO 27001, PCI-DSS, SOC 2 Type II, and the BSI's KRITIS qualification. Two levels of analysis ensure precision and speed: a global filter evaluates incoming requests immediately, while a context-based, deeper analysis takes path logic, header consistency, and ML-based anomalies into account.

For companies, this means fewer false alarms, less manual rule maintenance, and increased transparency. Especially in API-driven environments, where traffic is dynamic, volatile, and difficult to classify, a modern WAF becomes an indispensable control tool.

WAF in interaction and why integration is crucial

A WAF alone cannot cover today's threat landscape. Only in combination with DDoS mitigation, bot management, and API protection can the necessary resilience for modern digital business models be achieved.

This interaction offers three advantages:

1. It creates a consistent understanding of the context of a request, from the network to the application to the API.

2. Decisions become more consistent because all systems access the same data and risk scores.

3. Operational overhead is significantly reduced because less manual intervention is required and false alarms are significantly reduced.

This makes a WAF the central hub of web and API security.

API security: The invisible attack surface of modern digital business models

APIs as new lifelines and new vulnerabilities

APIs are the silent engines of digital value creation. They connect mobile apps to backends, orchestrate microservices, enable payment processes, authenticate users, and make complex digital ecosystems possible in the first place.

But it is precisely because APIs are ubiquitous that they are becoming the largest and often most underestimated attack surface of modern IT. Gartner predicts that 80 percent of modern traffic will run through APIs, 40 percent of companies will need advanced bot management by 2027, and 70 percent of API products will transition to WAAP systems.

At the same time, research shows that almost all companies have had API-related security issues in the last twelve months. Many of these were not spectacular, but they were risky: Unprotected endpoints, overlooked parameters, misconfigured gateways, or simply APIs that no one knew existed anymore.

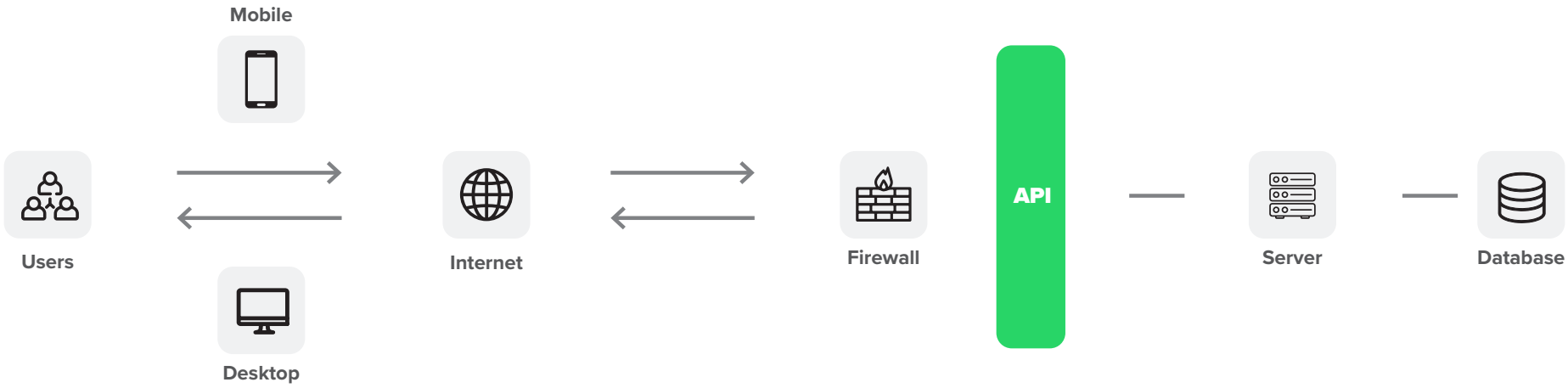
The growing API landscape is therefore not only a technological advance, but also a structural risk. Moreover, it has grown faster than security architectures have been able to keep pace.

The invisibility of shadow APIs – a systemic problem

Hardly any organization today has a complete inventory of its APIs. In dynamic DevOps environments, new endpoints are constantly being created for testing, internal integrations, feature-flows, or short-term projects. Many of these never disappear, but remain active in the background—without governance, without monitoring, without protection.

These so-called “shadow APIs” are one of the main reasons for API security incidents. They increase the attack surface without being visible to the security team. Internal or “semi-private” APIs, which are actually only intended for certain systems but accidentally remain publicly accessible, are also particularly dangerous.

As a result, attackers no longer have to search for classic vulnerabilities; they can often simply find unprotected or forgotten access points that lead directly to sensitive data.



Why API security is so fundamentally different from traditional web security

APIs cannot be analyzed like websites. There are no mouse movements, no scroll events, and no navigation behavior that could indicate human use. Technically speaking, every request looks like that of a bot, which is why distinguishing between legitimate and malicious automated clients is a core task.

In addition, modern software architectures lead to highly fragmented systems. Microservices, containers, serverless functions, and API gateways generate hundreds of endpoints that are constantly changing. Where once there was a monolith with clear URL structures, today there are a multitude of dynamic, versioned interfaces.

The consequence: APIs are inherently machine-dominated, and attacks on APIs are often not volumetric, but subtle.

It is not the number of requests that matters, but their significance: a single malicious API call can be enough to capture data, compromise identities, or manipulate processes.

Criteria	API Security	Web Security
Perimeter Protection	Many openings	Well-defined perimeter
Client Verification	No browser verification	Browser-based security
Attack Identification	Legitimate requests	Anomalies in requests
Protocol Changes	Frequent changes	Static protocols

Comparison between API Security and traditional or Web Security

The key API threats: precise, silent, and business-critical

Attacks on APIs have a specific profile. Their primary goal is not to crash systems, but rather to exploit their business logic. The most typical types of attacks include:

DDoS on API endpoints

High frequencies of seemingly harmless requests overload backends that rely on sessions, authentication, or database access.

Data scraping and information theft

Automated systems extract inventories, prices, or content—often continuously and inconspicuously.

Account takeovers (ATOs)

Login APIs are the preferred target of credential stuffing, brute force attacks, and password lists.

Inventory denial & business manipulation

Bots reserve products, block tickets, or manipulate purchasing processes, all of which are enormous economic risks.

Misuse of internal APIs after compromise

Once an attacker has gained access, they use internal API communication for lateral movement or data exfiltration.

These forms are not spectacular, but they are effective. They hit organizations where it hurts most: in their processes, transactions, and business logic.

The three pillars of modern API security

A modern API security architecture is based on three basic principles: discovery, protection, and governance. Together, they form a holistic system that both protects and enables transparency and control.

1

Discovery – Visibility across every API, even the unknown ones

The first step toward more secure APIs is transparency. Discovery means identifying all active interfaces—regardless of whether they are used internally, documented, obsolete, or forgotten.

A complete inventory is the basis for a correct risk assessment: Which APIs are publicly accessible? Which ones are still in use? Which versions exist? Which endpoints contain sensitive logic?

Without discovery, the security strategy remains incomplete – and thus dangerously blind.

2

Protection – Intelligent protection of APIs during operation

API protection mechanisms must work much more finely than classic web security. They check not only the technical structure, but also the semantics of a request. Essential components are:

- Strong authentication: MFA, mTLS, OAuth tokens, or HMAC signatures.
- Precise rate limiting, adapted to endpoint, risk, and user role.
- Schema and payload validation: Each request is checked against the expected formats.
- Replay detection and fingerprinting to expose reused or fake sessions.
- Integrity checks on mobile devices, e.g., root/jailbreak detection or app signature validation.
- Behavior-based bot detection: This is particularly important because API traffic is almost exclusively machine-generated.

This layer of protection is dynamic. It continuously learns from traffic patterns and adjusts its decisions accordingly.

3

Governance – Control over the entire lifecycle

An API does not end with its publication; that is when really begins. Governance means systematically defining and enforcing guidelines:

- API lifecycle management,
- Clear guidelines on who can access what and when
- Audit-proof logging of every transaction
- Continuous risk assessment
- Monitoring and telemetry ensure real-time transparency

Governance ensures that security works and remains traceable. This is a crucial point for compliance, forensics, and internal accountability.

Case Study 7

Best practices for implementing API security

Successful API security follows proven basic principles. These include strong authentication, end-to-end encryption, and consistent input validation. Rate limiting protects particularly sensitive endpoints such as the login and search functions, while the integrity of sessions and devices (e.g., in mobile apps) creates additional layers of security.

It is also important to minimize sensitive client data. The less information stored there, the lower the risk in the event of device loss or a compromised end device. Finally, continuous audits are necessary to ensure that all APIs comply with the original security requirements.

The most effective protection is provided by a WAAP platform that does not view API security in isolation, but in the context of DDoS, WAF, and bot management.

API security in banking

A typical use case illustrates the urgency:

- A bank operates a mobile app that allows users to perform transactions and retrieve account information.
- Attackers attempt to gain access via credential stuffing, particularly to the login API endpoint.
- The bank implements MFA and rate limiting for login attempts, detects compromised devices (e.g., jailbreaks), and validates each payload together with the API gateway rule.
- With the help of monitoring, anomalies are detected in real time and attacks are immediately repelled.

The result is a significant reduction in ATOs, greater security for API access, and more robust protection for sensitive financial data.

The mobile factor – Why smartphones increase the attack surface

Mobile devices have fundamentally changed the API landscape. Not only do they generate the majority of API traffic, but they also represent a direct point of attack. Users spend more than three hours a day in apps, but only about 30 of an average 80 installed apps are used regularly.

This discrepancy creates an ideal breeding ground for automated clients: old, unused, or poorly maintained apps continue to access APIs, which expands the attack surface. In addition, attackers specifically compromise mobile clients to extract API tokens or manipulate requests.

API security is therefore always mobile security.

Why API security is becoming a basic requirement for modern resilience

APIs are the nervous system of digital companies and, at the same time, one of their most vulnerable areas. They are growing rapidly, are difficult to keep track of, and are being targeted by increasingly automated and tactical attackers.

Those who fail to master API security lose control over data flows, business logic, and, in the worst case, key revenue streams.

A modern protection concept is based on:

- Complete visibility
- Intelligent runtime protection
- Clear governance

Combined with an integrated WAAP platform, this creates a security architecture that not only defends against attacks but also permanently protects a company's digital sovereignty.

Technical excellence that delivers tangible protection

The Link11 WAAP platform combines AI-powered protection mechanisms with a European, sovereign infrastructure. This reduces complexity, attack surface, and failure risks in one step. Link11 delivers precise, autonomous defense that stops attacks in milliseconds while allowing legitimate traffic to pass through without delay—even under extreme load peaks or in complex multi-cloud environments.

With a European private cloud, its own global backbone, and over 40 points of presence, Link11 delivers performance that remains stable even under extreme load. The platform combines WAF, bot management, API security, and web DDoS protection under a single SLA that is managed entirely by a team of experts. This ensures precise and explainable defense around the clock. The result is maximum security, maximum visibility, and significantly less operational effort for IT teams.

For companies striving for digital resilience, European sovereignty, and true future-proofing, Link11 is more than just a protective shield. The platform combines technical strength, operational relief, and compliance security, delivering what digital companies urgently need today: true resilience.

The 5 most important advantages at a glance

- **Zero-tuning security:** No manual rules, no complexity. ML-based detection, behavioral analytics, and API-native detection stop web DDoS, bots, and API attacks in real time.
- **100% EU sovereignty:** Private cloud, ISO & KRITIS compliant, without US hyperscaler dependency.
- **Unified protection:** WAF, bot management, API security, and DDoS—all on one platform, one SLA.
- **Forensic attack detection:** ML-powered threat profiling engine with minimal false positives.
- **High availability & speed:** : 40+ PoPs, 13 scrubbing centers, latency-optimized global backbone.

Your contact

Michael Scheffler
Vice President Sales

+49 69 58004926-306
m.scheffler@link11.com



Sources

https://www.fastly.com/blog/ai-bots-q2-2025-trends-fastlys-threat-insights-report

https://thunderbit.com/blog/web-crawling-stats-and-industry-benchmarks

https://www.brightspot.com/cms-resources/technology-insights/brightspot-automated-bot-traffic-report-and-recommendations

https://www.cybersecurity-insiders.com/2025-global-state-of-api-security-report-new-data-shows-api-breaches-continue-to-rise-due-to-fraud-bot-attacks-and-genai-risks/

https://mobian.studio/de/app-statistics/

https://salt.security/api-security-trends

https://www.traceable.ai/2025-state-of-api-security

https://www.uscsinstitute.org/cybersecurity-insights/blog/shadow-apis-explained-risks-detection-and-protection

https://content.salt.security/state-api-report.html

https://digitalike.de/ai-crawler-steuern/

https://www.gartner.com/en/documents/3903064#:~:text=Summary,will%20improve%20their%20security%20posture



www.link11.com