LINK11

UNSICHTBARE ANGRIFFE, SICHTBARE RISIKEN

Warum Webanwendungen und APIs ins Visier geraten und welche Sicherheitsarchitektur den Unterschied macht

www.link11.com

Executive Summary

Web Application and API Protection und die neue Ära automatisierter Bedrohungen

In den vergangenen Jahren hat das digitale Ökosystem einen leisen, aber folgenreichen Paradigmenwechsel erlebt: Das Web wird nicht mehr von Menschen, sondern von Maschinen dominiert. Bots, KI-Agenten und verteilte automatisierte Systeme erzeugen heute den Großteil des globalen HTTP-Traffics. Für Unternehmen bedeutet das: Die Bedrohungslandschaft ist nicht nur größer geworden, sondern hat auch ihr Gesicht grundlegend verändert. Klassische Schutzmechanismen greifen nicht mehr. Web Application & API Protection (WAAP) wird somit zur neuen Schlüsseldisziplin moderner Cyberresilienz.

Das Whitepaper zeigt, wie sich Angriffe weiterentwickelt haben: von simplen Volumenwellen zu hochgradig orchestrierten, mehrschichtigen Attacken. Web-DDoS-Kampagnen kombinieren Layer-3/4-Floods mit präzisen Layer-7-Angriffen, testen systematisch Abwehrmechanismen und rotieren ihre Angriffsvektoren dynamisch. Die vorgestellten Case Studies zeichnen ein klares Bild: Die Angreifer agieren global, adaptiv und mit einer Professionalität, die früher nur staatlichen Akteuren vorbehalten war.

Das wird besonders deutlich bei Angriffen, bei denen HTTP-Endpunkte mit zig Millionen Paketen pro Sekunde überflutet oder gleichzeitig Millionen HTTP-Anfragen und Sitzungen erzeugt werden, ohne dass auch nur ein Hauch verdächtiger Payloads hinterlassen wird. Moderne DDoS-Angriffe imitieren legitimes Nutzerverhalten so überzeugend, dass traditionelle Abwehrmechanismen blind bleiben.

Parallel dazu wächst eine zweite, mindestens genauso gefährliche Bedrohung: der Bot-Traffic. Heute stammt mehr als die Hälfte aller Webinteraktionen von automatisierten Clients. Viele davon verfolgen klare – mitunter kriminelle – Geschäftsmodelle wie Scraping, Account-Übernahme, Preismanipulation oder Betrug. Andere dienen der Vorbereitung von Angriffen, testen APIs aus, sondieren Schwachstellen oder erzeugen unauffälligen Druck auf Infrastruktur und Anwendungen. Die Case Studies zeigen: Es muss nicht immer laut sein. Manche Angriffe sind kaum messbar und dennoch gefährlich.

Insbesondere APIs geraten ins Visier. Sie treiben moderne Geschäftsmodelle – von Banking bis E-Commerce – voran und wachsen oft schneller als Sicherheitsarchitekturen mithalten können. Shadow-APIs, unkontrollierte Endpunkte oder ungesicherte Integrationen eröffnen Angreifern neue Wege. Oft bemerken Unternehmen Angriffe erst, wenn bereits Daten abgeflossen sind.

Die zentrale Botschaft ist klar und überzeugend: WAAP ist kein einzelnes Werkzeug, sondern ein strategisches Framework.
Eine moderne Abwehr muss DDoS-Mitigation, Bot-Management, verhaltensbasierte WAF und API-Sicherheit intelligent verbinden. Nur ein lernendes, kontextsensitives System kann heute zuverlässig zwischen legitimen Nutzern, harmlosen Bots, bösartigen Akteuren und ausgeklügelten Angriffsszenarien unterscheiden.

Die im Whitepaper analysierten Fälle zeigen, dass es bei Sicherheit heute weniger um das „Blockieren“ geht, sondern vielmehr um das Verstehen. Wer die Absicht hinter einer Anfrage erkennt, schützt nicht nur seine Systeme, sondern sein gesamtes digitales Geschäftsmodell.

Moderne WAAP-Plattformen wie die von Link11 setzen deshalb auf KI-gestützte Erkennung, forensische Analysemodelle, adaptive Mitigation und eine souveräne technische Infrastruktur, die auch unter extremer Last präzise arbeitet. Das Ergebnis sind weniger Fehlalarme, mehr Transparenz, eine höhere Verfügbarkeit und echte digitale Resilienz.

Die Evolution der Web-DDoS-Angriffe – vom Holzhammer zur chirurgischen Präzision

Die neue Ära des Web-DDoS: Präzision statt roher Gewalt

DDoS-Angriffe haben sich in den vergangenen Jahren radikal verändert. Was früher einfache Volumenattacken waren, ist heute ein komplexes, mehrschichtiges Angriffsfeld. Moderne Kampagnen kombinieren die Netzwerk-, Transport- und Anwendungsebene und nutzen hochautomatisierte und taktisch gesteuerte globale Botnetze.

Die Techniken reichen von klassischen Amplification-Angriffen (DNS, NTP, CLDAP) bis hin zu Slowloris-, API-Angriffen oder multifaktoriellen Attacken, die gleichzeitig auf mehreren OSI-Schichten operieren. Besonders prägnant ist die Fähigkeit moderner Angreifer, sich dynamisch an Gegenmaßnahmen anzupassen. On- und Off-Phasen, geänderte User Agents, rotierende IP-Pools und variable Payloads sind dabei nur einige Beispiele.

Web-DDoS ist somit keine reine Volumenfrage mehr. Die Bedrohung umfasst alles, was die Verfügbarkeit von Webdiensten stört – unabhängig davon, auf welcher Ebene der Angriff beginnt. Immer häufiger kommen hybride Angriffe zum Einsatz, die die Grenzen zwischen den Schichten bewusst verwischen.

Die folgenden Case Studies aus dem Link11-Netzwerk machen diese Entwicklung deutlich. Sie zeigen, wie unterschiedlich und zugleich komplex die Angriffsmuster sein können. Jeder einzelne Angriff liefert wertvolle Erkenntnisse für die Weiterentwicklung von Schutzstrategien und zeigt, dass klassische Abwehrmechanismen heute nicht mehr ausreichen.

Typologie von DDoS-Angriffen nach OSI-Schichten

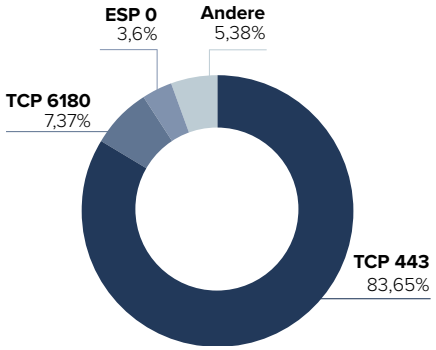
OSI-Schicht	Typische Angriffsformen	Ziel	Bezug zu Web-Diensten
Layer 3 (Netzwerk)	ICMP-Flood, IP-Flood	Netzwerküberlastung	Beeinträchtigt Routing und Erreichbarkeit von Webservern
Layer 4 (Transport)	TCP-SYN-Flood, UDP-Flood, TCP-Connection-Flood	Verbindungsaufbau blockieren, Ressourcen erschöpfen	Direkte Angriffe auf Webports (z. B. TCP 443 für HTTPS)
Layer 7 (Anwendung)	HTTP-Flood, Slowloris, API-Angriffe	Webserver und Applikationen lahmlegen	Zielt auf Webfunktionen, Formulare, APIs, Suchendpunkte
Hybrid-Angriffe	Kombination aus Layer 3/4/7	Mehrdimensionale Überlastung	Umgehen klassischer Schutzmechanismen durch Mischformen

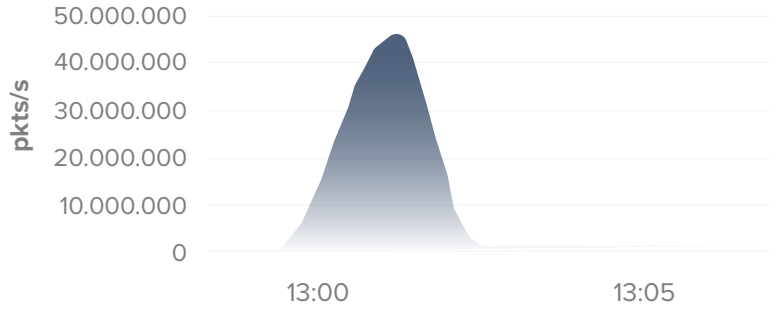
Case Study 1

Extreme TCP-Paketrate gegen Port 443 – ein Angriff mit doppelter Wirkung

In dieser Kampagne wurde ein HTTPS-Endpunkt mit bis zu **45 Millionen Paketen pro Sekunde** attackiert. Auffällig waren die ungewöhnlich großen Paketgrößen von etwa 1.400 Bytes – eine bewusste Kombination, um sowohl Bandbreite als auch Server-CPU gleichzeitig unter Druck zu setzen.

Zielports der Attacke und Paketrate pro Sekunde ergänzen





Technische Besonderheiten:

- Weltweit verteiltes Botnetz mit über 30.000 kompromittierten Geräten (China, Russland, Vietnam u. a.).
- Paketfragmentierung zur Umgehung klassischer Firewalls und IDS-Systeme.
- Gleichzeitige Überlastung auf Netzwerkressourcen und CPU-intensive TCP-Verbindungen.

Strategische Erkenntnisse:

- Klassische Bandbreitenfilter reichen nicht aus – notwendig sind Layer-4-Erkennung, Paket-Reassembly und Deep Packet Inspection.
- Echtzeit-Analytik und automatisierte Reaktionsmechanismen sind entscheidend, um fragmentierten und hochfrequenten Traffic zu erkennen.
- Unternehmen sollten Notfallpläne mit ISP- und Cloud-Scrubbing-Integration bereithalten, um bei Paketfluten schnell skalieren zu können.

Warum Layer-4-Angriffe zum Thema Web-DDoS gehören

Obwohl Layer-4-Angriffe technisch „unterhalb“ der Anwendung liegen, treffen sie moderne Websysteme direkt.

- Port 443 ist eine der Lebensadern der HTTPS-Kommunikation.
- Viele Webserver kollabieren nicht aufgrund der Datenmenge, sondern aufgrund von Tausenden gleichzeitiger TCP-Handshakes.
- Hohe Paketfrequenzen blockieren Netzwerkpfade, sodass der Server gar nicht erst entscheiden kann, ob der Traffic legitim ist.

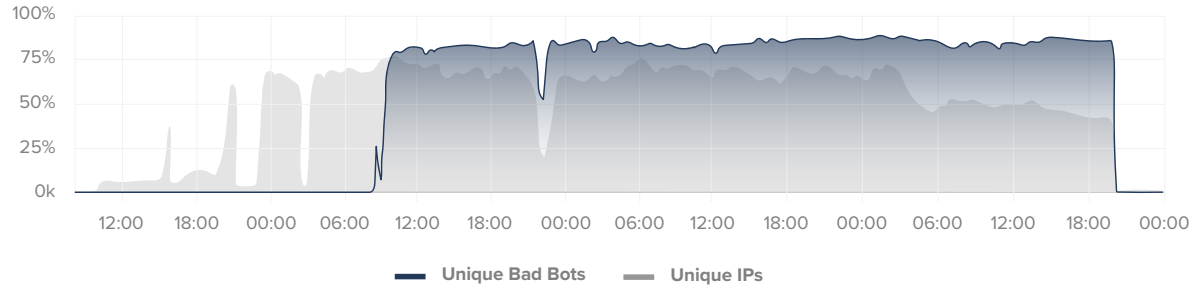
Ein Angriff auf Layer 4 ist somit fast immer ein Angriff auf die Verfügbarkeit einer Webanwendung – unabhängig davon, ob Layer 7 beteiligt ist oder nicht.

Case Study 2

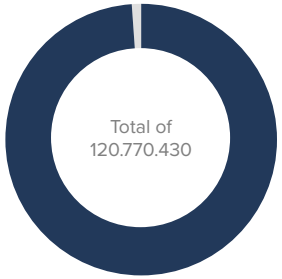
Hybrider Angriff – wenn alle Ebenen gleichzeitig unter Feuer stehen

Eine weitere Kampagne zeigt die Perfektion moderner Angriffe: Gleichzeitige Layer-3/4-Floods und Layer-7-Attacken mit über **120 Millionen HTTP-Requests** und bis zu 630 Requests/Sekunde.

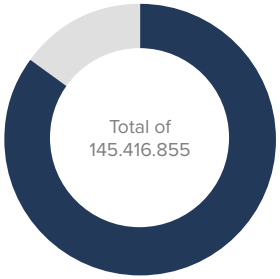
Bad Bots verursachen hohe Request-Zahlen



Bad Bot Anfragen



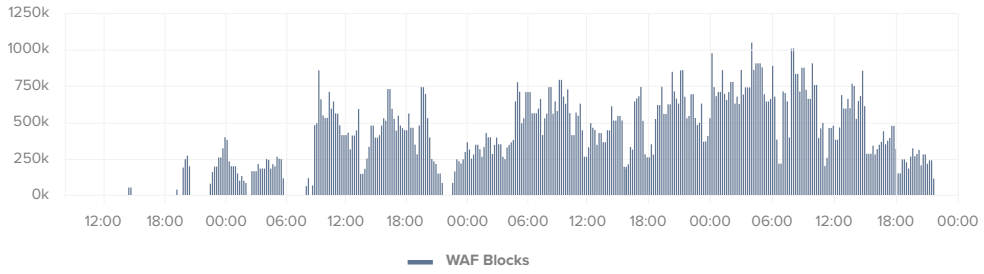
Verteilung der Anfragen



Eindeutige IP-Adressen als Hinweis auf ein globales Botnetz

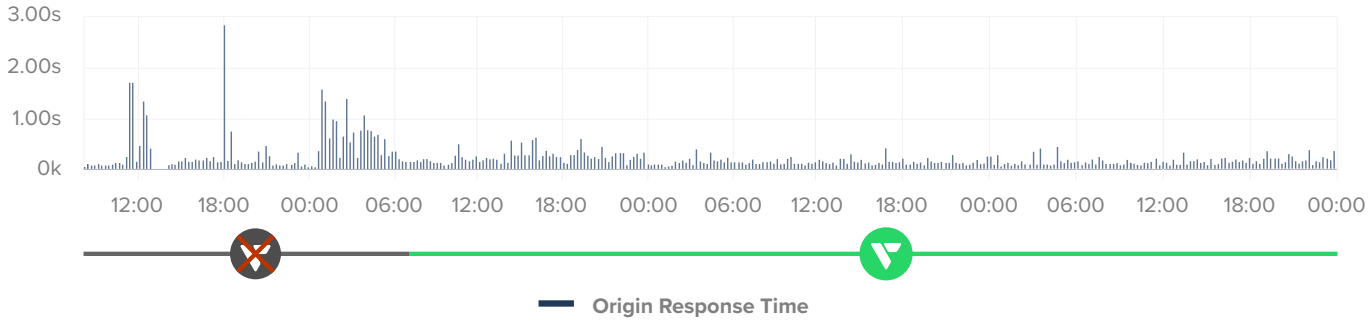


Mehrere Millionen WAF-Blocks im Angriffsverlauf



Case Study 2

Performance-Übersicht mit und ohne Schutz.



Hybride DDoS-Attacken gehören inzwischen verstärkt zum Repertoire der Angreifer. Besonders ganzheitliche WAAP-Lösungen, die eine Web Application Firewall, Bot-Management, DDoS-Mitigation und API-Schutz intelligent kombinieren, bieten die nötige Resilienz gegen diese komplexen Bedrohungen.

Technische Merkmale:

- Gleichzeitige Überlastung von Netzwerk und Applikation.
- 2.000 rotierende IPs aus globalen Botnetzen.
- Dynamische On-Off-Phasen, manipulierte User Agents, gezielte WAF-Probing-Muster.
- Über zwei Millionen WAF-Blocks in nur 30 Minuten.

Besonderheiten:

- Verwendung veralteter User Agents zur Tarnung automatisierter Bots.
- Herkunft aus ungewöhnlichen Netzwerken wie der Viettel Group, China Telekom und Yemen. net, was auf hohe Ressourcen und eine professionelle Infrastruktur hindeutet.
- Vermutete Motive: Hacktivismus, DDoS-as-a-Service oder gezielte Tests zur Optimierung zukünftiger Angriffe.

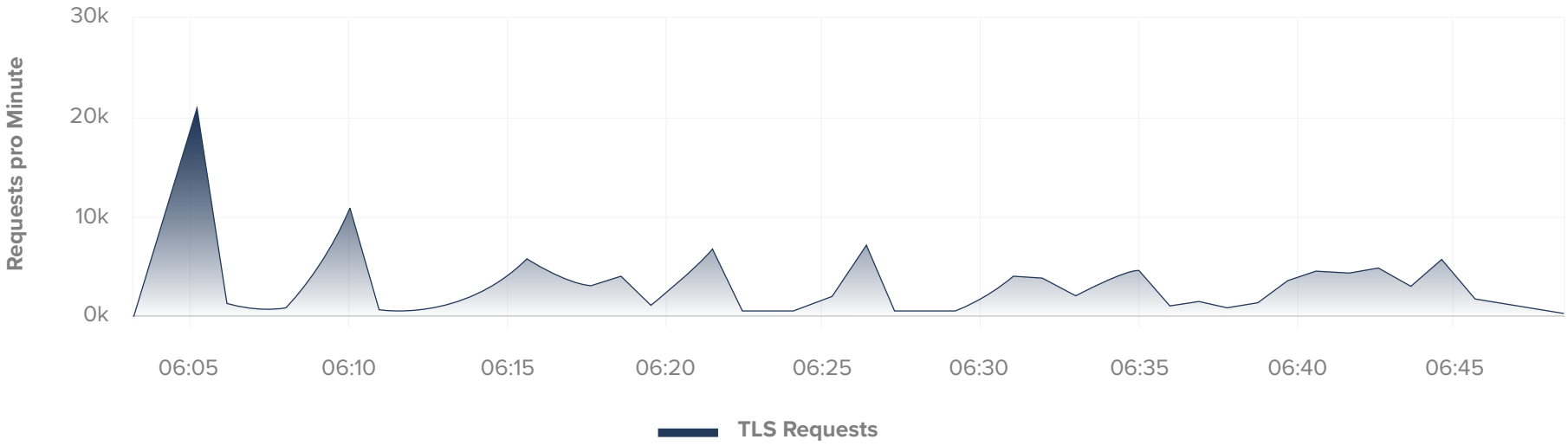
Strategische Erkenntnisse:

- WAAP-Systeme müssen kontinuierlich aktiv bleiben, da eine temporäre Deaktivierung bei Angriffspausen riskant ist.
- Bot-Management und WAFs müssen eng verzahnt arbeiten, um adaptive Angriffe zu erkennen und zu blockieren.
- API-Schutz ist essenziell, da APIs zunehmend Ziel automatisierter Angriffe sind.
- KI-gestützte Erkennung hilft dabei, neue Angriffsmuster frühzeitig zu identifizieren und automatisiert zu reagieren .

Case Study 3

Geopolitischer Layer-7-DDoS – Angriff mit Tarnkappe

Im Umfeld politischer Spannungen intensivierten prorussische Hacktivisten ihre Angriffe auf deutsche Institutionen. Besonders auffällig war dabei der scheinbar legitime Traffic aus deutschen CDN- und VPN-Knoten, der sich jedoch als strukturierter Layer-7-DDoS entpuppte.



Technische Merkmale:

- Bis zu 20.000 Requests pro Minute.
- Nutzung legitimer CDN/VPN-Infrastruktur (Datacamp, M247).
- Automatisierte Tarnung durch deutsche Edge-Nodes.

Erkennung und Abwehr:

1. Blockierung durch moderne Schutzsysteme aufgrund inkonsistenter Header, fehlender Referrer und nicht bestandener JS-/CAPTCHA-Challenges.
2. Der Angriff scheiterte daran, dass das menschliche Browsing-Verhalten nicht vollständig simuliert werden konnte.

Strategische Erkenntnisse:

- Layer-7-Angriffe können mit geringem Volumen hohen Schaden anrichten.
- Tarnung über legitime Infrastruktur nimmt rapide zu.
- Behavioral Detection ist zwingend erforderlich.

Übergreifende Muster –
was moderne DDoS-Angriffe auszeichnet

Alle analysierten Angriffe zeigen 3 gemeinsame Muster:

1

Angreifer setzen auf hybride Taktiken

Mehrere Layer gleichzeitig, synchron gesteuert.

2

Tarnung wird wichtiger als Volumen

Technisch unauffällige Anfragen, aber deutlicher Impact.

3

Automatisierung ersetzt manuelle Steuerung

Botnetze reagieren selbsttätig auf Abwehrmaßnahmen.



Für Unternehmen bedeutet das:

- Verhaltensbasierte Erkennung statt rein signaturbasierter Filter.
- Automatisierte Mitigation über alle Layer hinweg.
- Proaktive API-Schutzmechanismen inklusive Schema-Validierung und Rate-Limits.
- Eng verzahnte WAF, Bot-Management und DDoS-Mitigation statt isolierter Tools.
- Echtzeit-Analysen, um Musterwechsel sofort zu erkennen.

Web-DDoS ist heute ein multidimensionales Problem und erfordert multidimensionale Verteidigung. WAAP-Lösungen sind somit nicht nur ein technisches Werkzeug, sondern für Unternehmen, die ihre digitalen Geschäftsmodelle nachhaltig schützen wollen, die Grundlage moderner Cyberresilienz.

Bot Management – Mensch vs. Maschine im modernen Web

Die stille Machtübernahme: Bots als neue Mehrheit im Web

Das Internet erlebt derzeit eine leise, aber tiefgreifende Revolution. Zum ersten Mal in seiner Geschichte stammt der überwiegende Teil des Traffics nicht mehr von Menschen, sondern von Maschinen.

Mehr als die Hälfte des globalen HTTP-Verkehrs wird inzwischen von Bots erzeugt – von harmlosen Crawlern bis hin zu hochentwickelten, KI-gesteuerten Systemen, die menschliches Verhalten simulieren, API-Endpunkte sondieren oder Geschäftsprozesse manipulieren.

Besonders eindrucksvoll wird diese Verschiebung in Live-Daten sichtbar: Bei vielen Unternehmen liegt der Bot-Anteil dauerhaft über 50 Prozent. Menschlicher Traffic wird zum Minderheitenphänomen. Diese strukturelle Veränderung bildet den Hintergrund für alles, was im Bereich Web-, API- und DDoS-Security heute passiert. Diese strukturelle Verschiebung markiert den Beginn einer neuen Ära: der Ära der automatisierten Akteure.

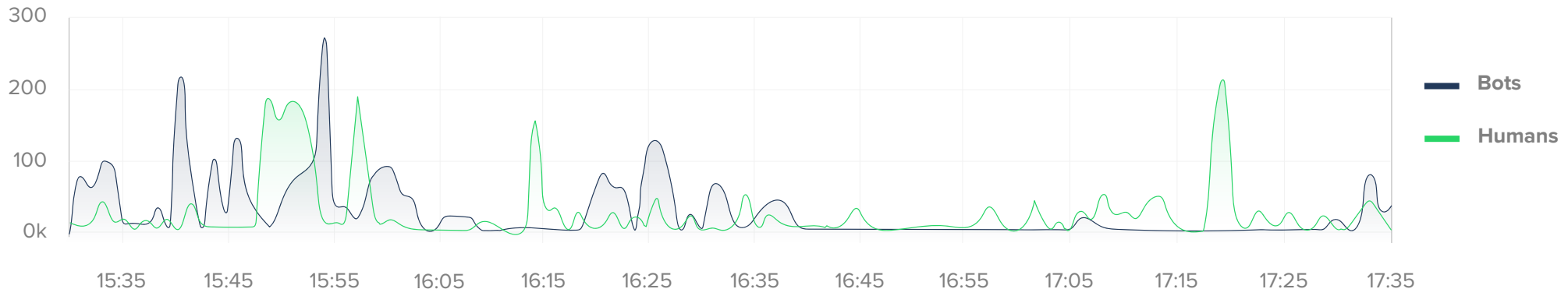
Was moderne Bots wirklich ausmacht

Bots sind schon lange keine einfachen Skripte mehr, die in festen Intervallen Anfragen senden. Viele von ihnen operieren mit Headless-Browsern, führen JavaScript aus, lösen Cookies aus, rotieren IPs oder analysieren Serverantworten mithilfe von Machine Learning. Einige imitieren das Verhalten legitimer Nutzer so überzeugend, dass klassische Schutzmaßnahmen wie Rate-Limits oder CAPTCHAs kaum noch greifen.

Ihre Ziele sind vielfältig: Sie indexieren Inhalte, extrahieren Daten, automatisieren Transaktionen, übernehmen Accounts oder testen die Belastbarkeit von APIs. Eine besonders dynamische Entwicklung nehmen KI-basierte Crawler, die Inhalte semantisch erfassen und autonom Entscheidungen treffen können.

Damit verschwimmt die Grenze zwischen nützlich und schädlich zunehmend – und Organisationen verlieren schnell die Kontrolle darüber, wer ihre Daten nutzt und zu welchem Zweck.

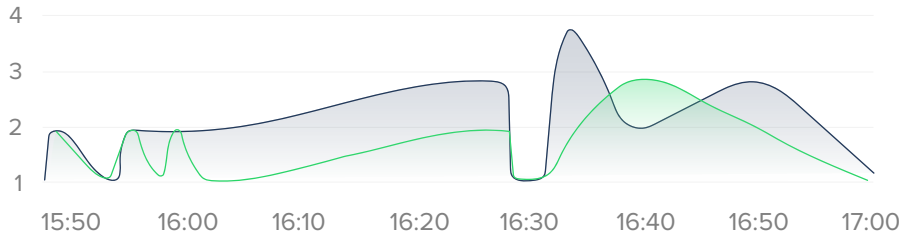
Vergleich zwischen menschlichem und automatisiertem Traffic bei einem Link11-Kunden



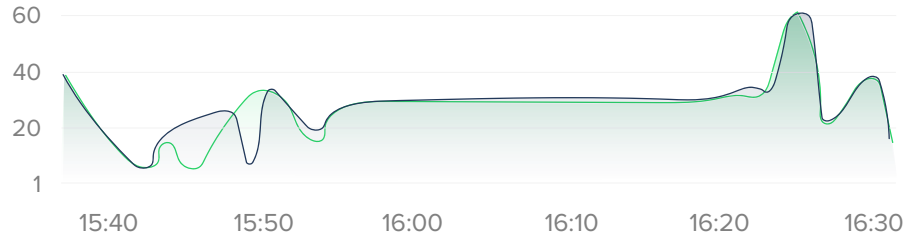
Wie Bots heute arbeiten – und warum klassische Abwehrmechanismen scheitern

Moderne Bots sind so programmiert, dass sie unauffällig bleiben. Während menschliche Nutzer in unregelmäßigen Abständen interagieren, mit Pausen, Seitenwechseln und Scrollbewegungen, agieren Bots in präzisen Intervallen und mit perfekt reproduzierbaren Mustern. Sie können dieses Verhalten jedoch auch gezielt verschleiern, etwa durch variable Timing-Strategien oder adaptives Rebalancing, sobald ein Endpunkt auf Abwehrmaßnahmen reagiert.

Anfragefrequenz pro IP-Adresse eines Nutzers



Bot-Anfragefrequenz



Hinzu kommen globale Proxy-Netzwerke, die es Angreifern ermöglichen, tausende kurzlebige IP-Adressen einzusetzen. Diese Taktik unterwandert IP-basierte Filter regelrecht. Gleichzeitig verfügen Headless-Browser über die Fähigkeit, Challenge-Mechanismen zu bestehen, die früher als sicher galten. Selbst CAPTCHAs können inzwischen automatisiert gelöst werden.

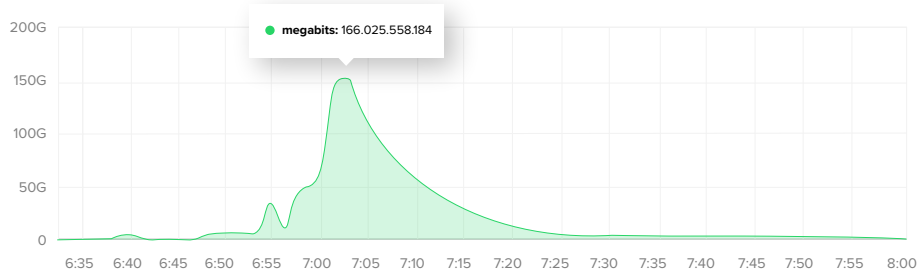
Der Effekt ist immer derselbe: Systeme, die auf statischen Regeln oder simplen Schwellenwerten basieren, werden zunehmend blind.

Case Study 4

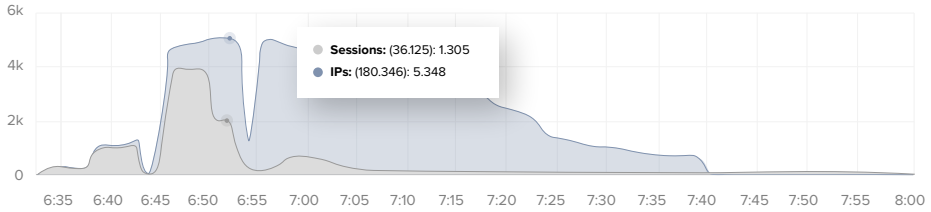
Wenn ein globales Botnetz eine Plattform „sanft“ überlastet

In einem typischen Angriffsszenario stieg der Traffic auf einer Business-Plattform zunächst nur leicht an. Das ist ein Muster, das in der Anfangsphase häufig unterschätzt wird. Doch hinter den unscheinbaren Peaks verbargen sich 16 Millionen Requests pro Minute, die von mehreren tausend IP-Adressen erzeugt wurden. Fast alle Anfragen trugen denselben User-Agent, was auf eine automatisierte Herkunft hindeutet.

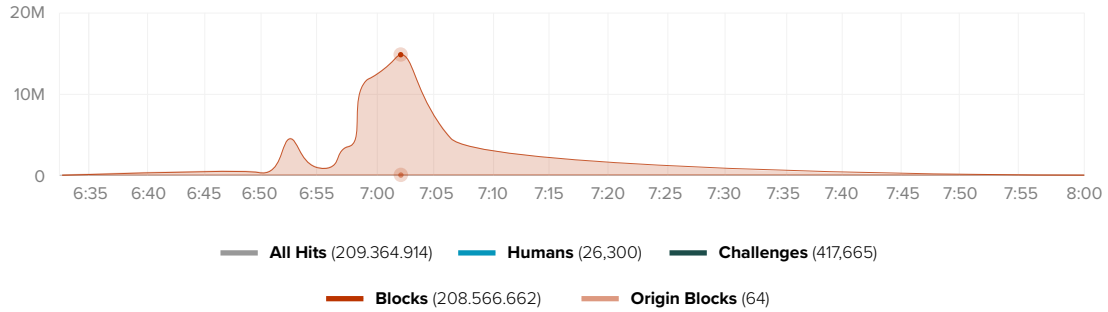
Gesamte Bandbreite



Einmalige Sitzungen & IP



Durchgelassen vs. blockiert



Spannend ist, wie sich die Bots verhielten: Nachdem die ersten IPs blockiert worden waren, passte das Botnetz seine Verteilung an, schob Traffic auf andere Regionen oder erhöhte dort die Frequenz. Der Angriff war nicht laut, aber penetrant.

Dieses Beispiel zeigt, dass große Botnetze nicht zwingend große Datenmengen benötigen. Schon ein geringes Volumen, das jedoch perfekt getaktet und hoch parallelisiert ist, kann kritische Ressourcen verbrauchen.

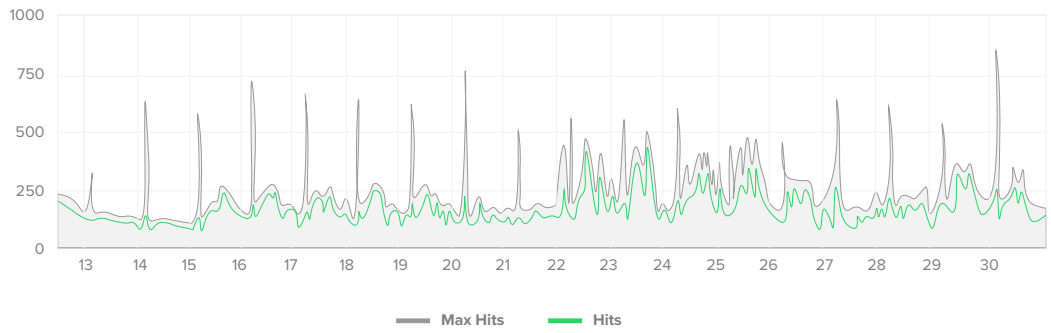
Ein funktionierendes Bot-Management trennt legitimen Traffic von automatisiertem, bevor Ressourcen erschöpft werden.

Case Study 5

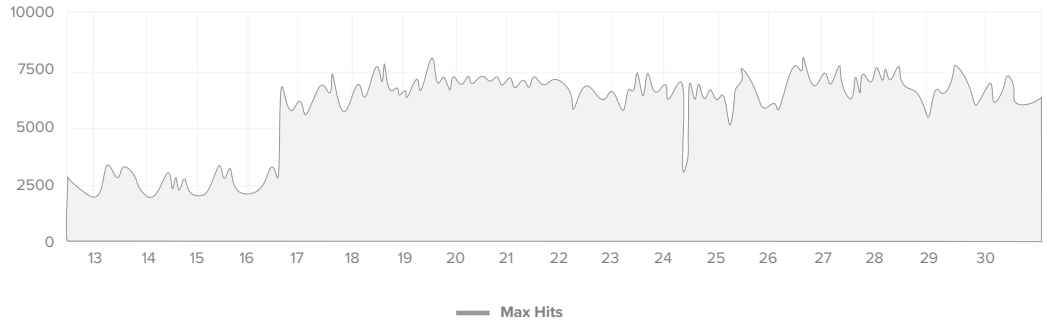
Der rätselhafte Anstieg der Unique IPs

Ein weiteres Beispiel verdeutlicht, wie verdeckt moderne Bot-Kampagnen operieren. So verdoppelte sich die Zahl der Unique IPs von rund 2.000 auf bis zu 8.000, während die Bandbreite und das Request-Volumen nur moderat anstiegen. Auf den ersten Blick wirkt das harmlos, doch die Serverlatenz stieg und es traten immer wieder 500er-Fehler auf.

Requests pro Tag



Eindeutige IPs pro Tag



Strategische Erkenntnisse:

- Klassische Monitoring-Metriken (Bandbreite, Request-Rate) können täuschen.
- Ohne Bot-Analyse bleiben solche Anomalien unsichtbar. Es ist ein „blinder Fleck“ im Netzwerk.

Der Grund dafür ist, dass viele kleine, kurzlebige Verbindungen eine Last erzeugen, die weder im Traffic-Volumen noch in klassischen Metriken sichtbar ist. Zusätzlich stieg der Anteil der POST-Anfragen signifikant, was auf Formular- und Suchfunktionsmissbrauch hindeutet. Auch die Herkunft des Traffics aus spezifischen Cloud-Providern passte zum typischen Muster eines automatisierten Netzes.

Diese Kombination zeigt ein typisches Botnetz-Muster: Viele kurzlebige Verbindungen, kaum Daten, aber eine hohe Serverlast. Echte Browser halten Sessions offen und bündeln Anfragen, während Bots ständig neue TCP-Verbindungen ohne Keep-Alive öffnen.

Warum traditionelle Bot-Abwehr versagt

Viele gängige Schutzverfahren wie IP-Blocklisten, einfache Rate-Limits oder CAPTCHAs stammen aus einer Zeit, in der Bots noch sehr einfach aufgebaut waren. Heute sind Angreifer jedoch in der Lage, all diese Mechanismen zu umgehen. CAPTCHAs werden automatisiert gelöst, IPs rotieren im Sekundentakt, Header lassen sich beliebig fälschen und Headless-Browser simulieren überzeugend menschliche Interaktionen.

Zwar reduzieren auch reCAPTCHA v3 oder ähnliche unsichtbare Systeme die Reibung für Nutzer, sie führen jedoch zu höheren Kosten und Datenschutzrisiken, während fortgeschrittene Bots trotzdem durchschlüpfen. Die Konsequenz ist klar: Signatur- oder regelbasierte Systeme reichen schlicht nicht mehr aus. Eine zeitgemäße Bot-Abwehr muss daher verhaltensbasiert, adaptiv und kontextsensitiv arbeiten.



Die neue Generation der Bot-Erkennung: Von Regeln zu lernenden Systemen

Moderne Bot-Management-Lösungen setzen auf ein mehrschichtiges Analysemodell. An erster Stelle steht die Verhaltensanalyse, bei der auf Muster wie Tippgeschwindigkeit, Scroll-Verhalten, Session-Rhythmik oder Clientkonsistenz geachtet wird. Zusätzlich kommen clientseitige Zertifikate, SDK-basierte Integritätsprüfungen und serverseitige Validierungsprozesse zum Einsatz.

Für APIs gelten nochmals eigene Anforderungen. Ohne Browserfingerprinting, Replay-Erkennung oder Payload-Validierung bleibt die API ein offenes Einfallstor. Die beste Abwehr ist jedoch kein perfekter Filter, sondern ein lernendes System, das False Positives minimiert und gleichzeitig bösartige Bots zuverlässig erkennt.

Dieses Zusammenspiel aus Analyse, Automatisierung und kontinuierlichem Feedback bildet die Grundlage moderner Bot-Governance und ist somit ein wesentlicher Baustein der Cyber-Resilienz.



Bot-Governance als strategischer Faktor

Bot-Management ist schon lange keine rein technische Aufgabe mehr. Die zentrale Frage lautet heute: „Wer darf automatisiert auf meine digitalen Systeme zugreifen und unter welchen Bedingungen?"

Um dies zu klären, müssen Unternehmen zunächst vollständige Transparenz über ihre automatisierten Interaktionen schaffen. Auf dieser Grundlage müssen sie eine klare Policy für erlaubte und unerlaubte Bots definieren, technische Schutzschichten einrichten und deren Wirksamkeit kontinuierlich messen.

WAAP-Lösungen mit integriertem Bot-Management bilden das technische Fundament dafür, doch entscheidend ist die Kombination aus Technologie, Expertise und Governance. Nur so lässt sich die Kontrolle über automatisierten Traffic zurückgewinnen – und damit über einen zentralen Teil der digitalen Infrastruktur.

Moderne WAF: Von statischen Regeln zur intelligenten Analyseinstanz

Warum klassische WAFs nicht mehr ausreichen

Lange Zeit waren Web Application Firewalls das zentrale Werkzeug, um Webanwendungen vor bekannten Angriffen zu schützen. Doch die Angriffslandschaft hat sich schneller weiterentwickelt als diese Systeme. Während traditionelle WAFs auf Signaturen, feste Regeln und statische Heuristiken setzen, agieren moderne Angreifer flexibel, adaptiv und hochautomatisiert.

APIs sind zum Rückgrat digitaler Geschäftsmodelle geworden und erzeugen inzwischen mehr als 80 Prozent des Webverkehrs. Gleichzeitig dominieren Bots und KI-Agenten das Internet. Viele Angriffe sind nicht mehr laut und auffällig, sondern subtil und sorgfältig getarnt. Sie sind in der Lage, Schutzmechanismen auszutesten und gezielt zu umgehen.

Unternehmen sind daher mit drei strukturellen Herausforderungen konfrontiert:

1. Das Volumen an automatisiertem Traffic ist so hoch, dass regelbasierte Systeme kaum unterscheiden können, was legitim ist und was schädlich.
2. Komplexe Microservice-Architekturen und schnelllebige DevOps-Zyklen verändern APIs und Endpunkte in immer kürzeren Abständen. Statische Regeln veralten innerhalb weniger Tage.
3. Betriebliche Teams stehen unter Druck. Fehlalarme, unklare Blockgründe und manuell gepflegte Regelwerke belasten die SOCs gerade in Spitzenzeiten – ähnlich wie ein Flugsteuerpult, das ausgerechnet bei Sturm unübersichtlich wird.

Hinzukommt ein zunehmend strenger werdendes Compliance-Umfeld. Unternehmen müssen heute nicht nur Angriffe abwehren, sondern auch nachweisen können, warum bestimmte Zugriffe blockiert wurden. Auditfähigkeit, Transparenz und Nachvollziehbarkeit sind Pflicht.

All das führt zu einer klaren Erkenntnis: Die Zeit rein regelbasierter WAFs ist vorbei.

Wie moderne WAF-Systeme funktionieren – und warum sie intelligenter arbeiten müssen

Die moderne WAF funktioniert nicht mehr wie ein Filter, der Muster erkennt, sondern wie eine Analyseinstanz, die Verhalten interpretiert. Sie kombiniert verschiedene Erkennungsebenen. Statt nur den Inhalt einer Anfrage zu prüfen, erkennt die WAF das Wie:

Timing, Session-Rhythmik, Headerkonsistenz, Abweichungen im Clientprofil oder ungewöhnliche Navigationsmuster. Mit diesem Ansatz lassen sich automatisierte Bots identifizieren, die versuchen, Browserlogik zu imitieren

Jeder API-Endpunkt hat seine eigene Kontext- und Pfadlogik: Ein Login verhält sich anders als ein Suchformular oder ein Bestellprozess. Moderne WAFs erkennen, ob eine Anfrage in den erwarteten Kontext passt, ob sie semantisch schlüssig ist oder ob sie sich wie ein Missbrauchsversuch darstellt.

Machine Learning: ML-Modelle erkennen Zusammenhänge, die Menschen oder statische Regeln nicht erfassen können: Beispielsweise Cluster von Anomalien in User Agents, untypische Korrelationen in TLS-Fingerprints oder Wiederholungsmuster, die erst über Millionen Requests sichtbar werden.

Diese Ebenen ergänzen sich gegenseitig. Die WAF trifft nicht auf Basis eines einzelnen Merkmals Entscheidungen, sondern auf Grundlage einer Wahrscheinlichkeit, die aus vielen Signalen berechnet wird. Dadurch werden Blockentscheidungen präziser und nachvollziehbarer, was einen entscheidenden Vorteil für Sicherheits- und Compliance-Teams darstellt.

Case Study 6

Leiser Layer-7-DDoS durch Session-Flut

Ein Angriff auf eine europäische Plattform zeigt exemplarisch, wie gefährlich „leiser“ Verkehr sein kann. Der Angriff setzte nicht auf große Datenmengen oder komplexe Payloads, sondern auf eine enorme Zahl von Sessions: **16 Millionen neue Sitzungen pro Minute**, insgesamt über **580 Millionen HTTP-Anfragen** und rund **1.000 IP-Adressen**, die ihre Verteilung dynamisch anpassten. Jede einzelne Anfrage war technisch unauffällig – einfache GET-Requests auf die Root-Domain.

Die Masse an Sessions überforderte jedoch das Backend: CPU, Arbeitsspeicher und das Session Handling kollabierten nahezu. Ein solches Szenario wäre mit klassischen DDoS-Sensoren kaum zu erkennen gewesen, da das Traffic-Volumen nicht außergewöhnlich hoch war.



Erkennungs- und Abwehrmaßnahmen:

- Die WAF identifizierte ungewöhnliche Zugriffsmuster, eine hohe Session-Dichte sowie nichtmenschliches Verhalten.
- Verdächtige IPs wurden automatisch isoliert und zusätzlich mithilfe eines Layer-3-Schutzes blockiert.
- Ein Quarantäne-Mechanismus verhinderte Wiederholungsversuche durch bereits erkannte Quellen.

Besonderheit des Angriffs:

- Das Botnetz agierte adaptiv: Sobald andere blockiert wurden, wurden neue IPs nachgeladen.
- Die Angreifer simulierten legitimen Traffic, wodurch klassische Filtermechanismen umgangen wurden.

Diese Case Study zeigt, wie gefährlich die Angriffe geworden sind, die unterhalb der üblichen Alarmgrenzen stattfinden – und warum eine moderne WAF nicht nur Inhalte, sondern auch das Verhalten verstehen muss.

Die WAF als strategischer Baustein moderner Sicherheit

Moderne WAFs sind mehr als nur ein technischer Schutzwall. Sie werden zu strategischen Komponenten einer WAAP-Architektur, da sie Entscheidungen erklären, Risiken messbar machen und applikationsnah agieren.

Die Plattformarchitektur von Link11 ist dafür beispielhaft: Sie basiert nicht auf einer Multi-Tenant-Umgebung, sondern auf isolierten Instanzen. Dadurch bleibt die Latenz planbar und regulatorische Anforderungen wie ISO 27001, PCI-DSS, SOC 2 Type II und die KRITIS-Qualifizierung des BSI werden erfüllt. Zwei Analyseebenen sorgen für Präzision und Geschwindigkeit: Ein globaler Filter bewertet eingehende Requests sofort, während eine kontextbasierte, tiefere Analyse Pfadlogik, Headerkonsistenz und ML-basierte Anomalien berücksichtigt.

Für Unternehmen bedeutet das weniger Fehlalarme, weniger manuelle Regelpflege und eine erhöhte Transparenz. Besonders in API-getriebenen Umgebungen, in denen der Traffic dynamisch, volatil und schwer einzuordnen ist, wird eine moderne WAF zum unverzichtbaren Steuerungsinstrument.

WAF im Zusammenspiel und warum Integration entscheidend ist

Eine WAF allein kann die heutige Bedrohungslage nicht abdecken. Erst im Zusammenspiel mit DDoS-Mitigation, Bot-Management und API-Schutz entsteht die erforderliche Widerstandsfähigkeit für moderne digitale Geschäftsmodelle.

Dieses Zusammenspiel bietet drei Vorteile:

1. Es entsteht ein durchgängiges Verständnis für den Kontext einer Anfrage – vom Netzwerk über die Anwendung bis zur API.

2. Entscheidungen werden konsistenter, da alle Systeme auf dieselben Daten und Risikoscores zugreifen.

3. Der operative Aufwand reduziert sich erheblich, da weniger manuelle Eingriffe erforderlich sind und Fehlalarme deutlich reduziert werden.

Somit wird die WAF zum zentralen Knotenpunkt der Web- und API-Sicherheit.

API Security: Die unsichtbare Angriffsfläche moderner digitaler Geschäftsmodelle

APIs als neue Lebensadern und neue Schwachstellen

APIs sind die stillen Motoren der digitalen Wertschöpfung. Sie verbinden mobile Apps mit Backends, orchestrieren Microservices, ermöglichen Zahlungsprozesse, authentifizieren Nutzer und machen komplexe digitale Ökosysteme überhaupt erst möglich.

Doch gerade, weil APIs allgegenwärtig sind, entwickeln sie sich zur größten und oft unterschätzten Angriffsfläche moderner IT. Gartner prognostiziert, dass 80 Prozent des modernen Traffics über APIs laufen, 40 Prozent der Unternehmen bis 2027 fortgeschrittenes Bot-Management benötigen und 70 Prozent der API-Produkte in WAAP-Systeme übergehen werden.

Gleichzeitig zeigen Untersuchungen, dass fast alle Unternehmen in den letzten zwölf Monaten API-bezogene Sicherheitsprobleme hatten. Viele davon waren nicht spektakulär, aber riskant: Ungeschützte Endpunkte, übersehene Parameter, falsch konfigurierte Gateways oder schlicht APIs, von denen niemand mehr wusste, dass sie existieren.

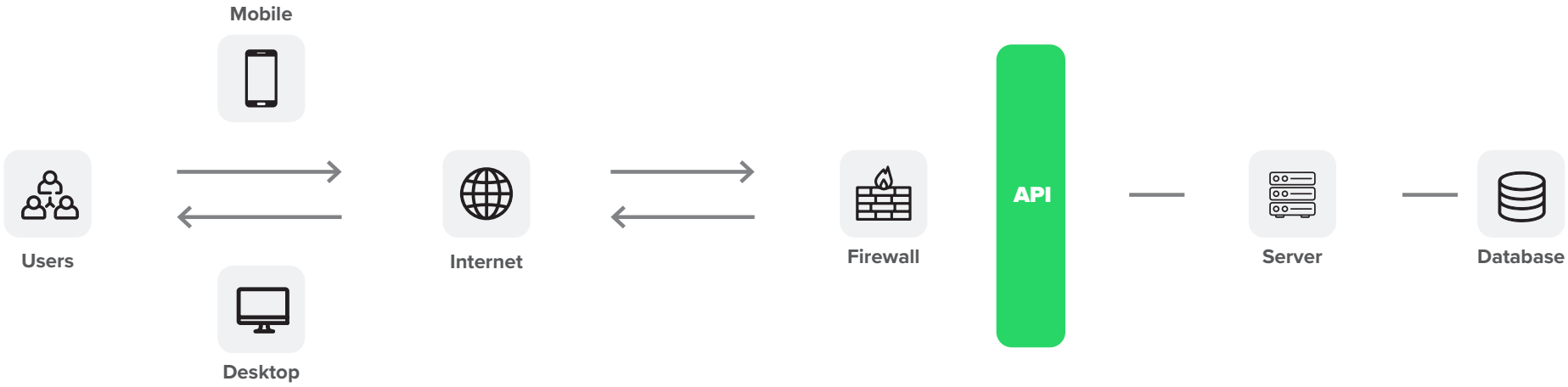
Die wachsende API-Landschaft ist somit nicht nur ein technologischer Fortschritt, sondern birgt auch ein strukturelles Risiko. Zudem ist sie schneller gewachsen, als Sicherheitsarchitekturen Schritt halten konnten.

Die Unsichtbarkeit der Shadow-APIs – ein systemisches Problem

Kaum eine Organisation verfügt heute über ein vollständiges Inventar ihrer APIs. In dynamischen DevOps-Umgebungen entstehen ständig neue Endpunkte für Tests, interne Integrationen, Feature-Flows oder kurzfristige Projekte. Viele davon verschwinden nie wieder, sondern bleiben im Hintergrund aktiv – ohne Governance, ohne Monitoring, ohne Schutz.

Diese sogenannten Shadow-APIs sind einer der Hauptgründe für API-Sicherheitsvorfälle. Sie erweitern die Angriffsfläche, ohne im Blickfeld des Security-Teams zu stehen. Besonders gefährlich sind interne oder „semi-private“ APIs, die eigentlich nur für bestimmte Systeme vorgesehen sind, aber versehentlich öffentlich erreichbar bleiben.

Die Folge: Angreifer müssen nicht mehr nach klassischen Schwachstellen suchen, da sie oft einfach ungeschützte oder vergessene Zugänge finden, die direkt zu sensiblen Daten führen.



Warum API-Sicherheit anders ist als klassische Websicherheit

APIs lassen sich nicht wie Websites analysieren. Es gibt keine Mausbewegungen, keine Scroll-Ereignisse und kein Navigationsverhalten, die Hinweise auf eine menschliche Nutzung geben könnten. Technisch gesehen sieht jeder Request wie der eines Bots aus, weshalb die Unterscheidung zwischen legitimen und bösartigen automatisierten Clients eine Kernaufgabe darstellt.

Zudem führen moderne Softwarearchitekturen zu hochgradig fragmentierten Systemen. Microservices, Container, Serverless Functions und API-Gateways erzeugen hunderte Endpunkte, die sich ständig ändern. Wo früher ein Monolith mit klaren URL-Strukturen existierte, gibt es heute eine Vielzahl dynamischer, versionierter Schnittstellen.

Die Konsequenz: APIs sind von Natur aus maschinendominiert – und Angriffe auf APIs sind oft nicht volumetrisch, sondern subtil.

Es kommt nicht auf die Menge der Anfragen an, sondern auf ihre Bedeutung: Ein einziger schädlicher API-Aufruf kann ausreichen, um Daten abzugreifen, Identitäten zu kompromittieren oder Prozesse zu manipulieren.

Kriterium	API-Sicherheit	Web-Sicherheit
Perimeterschutz	Viele Öffnungen	Klar definierter Perimeter
Client-Verifizierung	Keine Browser-Verifizierung	Browserbasierte Sicherheit
Angriffserkennung	Legitime Anfragen	Anomalien in Anfragen
Protokolländerungen	Häufige Änderungen	Statische Protokolle

Vergleich zwischen API Security und traditioneller Web Security

Die zentralen API-Bedrohungen: Präzise, leise und geschäftsrelevant

Angriffe auf APIs haben ein spezifisches Profil. Sie zielen nicht primär darauf, Systeme zum Absturz zu bringen, sondern deren Geschäftslogik auszunutzen. Zu den typischsten Angriffstypen gehören:

DDoS auf API-Endpunkte

Hohe Frequenzen harmlos wirkender Requests überlasten Backends, die auf Sessions, Authentifizierung oder Datenbankzugriffe angewiesen sind.

Datenscraping und Informationsdiebstahl

Automatisierte Systeme extrahieren Inventare, Preise oder Inhalte – oft kontinuierlich und unauffällig.

Account Takeovers (ATOs)

Login-APIs sind das bevorzugte Ziel von Credential Stuffing, Brute-Force-Angriffen und Passwortlisten

Inventory Denial & Business Manipulation

Bots reservieren Produkte, blockieren Tickets oder manipulieren Kaufprozesse – wirtschaftlich ein enormes Risiko.

Missbrauch interner APIs nach Kompromittierung

Hat ein Angreifer einmal Zugang, nutzt er interne API-Kommunikation für seitliche Bewegungen oder Datenexfiltration.

Diese Formen sind nicht spektakulär – aber effektiv. Sie treffen Organisationen dort, wo es am meisten schmerzt: bei Prozessen, Transaktionen und ihrer Geschäftslogik.

Die drei Säulen moderner API-Sicherheit

Eine moderne API-Sicherheitsarchitektur basiert auf drei Grundprinzipien: Discovery, Protection und Governance.
Gemeinsam bilden sie ein ganzheitliches System, das nicht nur schützt, sondern auch Transparenz und Kontrolle ermöglicht.

1

Discovery – Sichtbarkeit über jede API, auch die unbekannte

Der erste Schritt zu sichereren APIs ist Transparenz. Discovery bedeutet, alle aktiven Schnittstellen zu erkennen – unabhängig davon, ob sie intern genutzt werden, dokumentiert, veraltet oder in Vergessenheit geraten sind.

Ein vollständiges Inventar ist die Grundlage für eine korrekte Risikobewertung: Welche APIs sind öffentlich zugänglich? Welche werden noch genutzt? Welche Versionen existieren? Welche Endpunkte enthalten sensible Logik?

Ohne Discovery bleibt die Sicherheitsstrategie unvollständig – und damit gefährlich blind.

2

Protection – Intelligente Absicherung von APIs im laufenden Betrieb

API-Schutzmechanismen müssen deutlich feiner arbeiten als klassische Websicherheit. Sie prüfen nicht nur die technische Struktur, sondern auch die Semantik einer Anfrage. Wesentliche Bestandteile sind:

- Starke Authentifizierung: MFA, mTLS, OAuth-Token oder HMAC-Signaturen.
- Präzises Rate Limiting, das an Endpunkt, Risiko und Benutzerrolle angepasst ist.
- Schema- und Payload-Validierung: Jede Anfrage wird gegen die erwarteten Formate geprüft.
- Replay-Detection und Fingerprinting, um wiederverwendete oder gefälschte Sessions zu entlarven.
- Integritätsprüfungen auf mobilen Endgeräten, z. B. Root-/Jailbreak-Erkennung oder App-Signaturvalidierung.
- Verhaltensbasierte Bot-Erkennung: Dies ist besonders wichtig, da API-Traffic fast ausschließlich maschinell ist.

Diese Schutzschicht ist dynamisch. Sie lernt fortlaufend aus Traffic-Mustern und passt ihre Entscheidungen entsprechend an.

3

Governance – Kontrolle über den gesamten Lebenszyklus

Eine API endet nicht mit ihrer Veröffentlichung –sie beginnt dann erst. Governance bedeutet, Richtlinien systematisch zu definieren und durchzusetzen:

- Lebenszyklusmanagement von APIs,
- klare Richtlinien, wer wann wie zugreifen darf,
- auditfähige Protokollierung jeder Transaktion,
- kontinuierliche Risikobewertung,
- Monitoring und Telemetrie sorgen für Echtzeit-Transparenz.

Governance sorgt dafür, dass Sicherheit nicht nur funktioniert, sondern auch nachvollziehbar bleibt. Das ist ein entscheidender Punkt für Compliance, Forensik und interne Verantwortlichkeit.

Case Study 7

Best Practices oder wie Organisationen API-Sicherheit in der Praxis umsetzen

Erfolgreiche API-Security folgt bewährten Grundprinzipien. Dazu gehören eine starke Authentifizierung, eine durchgängige Verschlüsselung und eine konsequente Eingabevalidierung. Rate Limiting schützt besonders sensible Endpunkte wie die Login- und Suchfunktion, während die Integrität von Sitzungen und Geräten (etwa bei mobilen Apps) zusätzliche Sicherheitsebenen schafft.

Wichtig ist auch, sensible Daten auf Clients zu minimieren. Je weniger Informationen dort abgelegt werden, desto geringer ist das Risiko im Falle eines Geräteverlusts oder eines kompromittierten Endgeräts. Schließlich sind kontinuierliche Audits erforderlich, um sicherzustellen, dass alle APIs den ursprünglichen Sicherheitsvorgaben entsprechen.

Den effektivsten Schutz bietet eine WAAP-Plattform, die API-Sicherheit nicht isoliert, sondern im Kontext von DDoS-, WAF- und Bot-Management betrachtet.

API-Sicherheit im Banking

Ein typischer Use Case verdeutlicht die Dringlichkeit:

- Eine Bank betreibt eine mobile App, über die Nutzer Transaktionen durchführen und Kontoinformationen abrufen können.
- Angreifer versuchen, über Credential Stuffing Zugriff zu erlangen, insbesondere auf den Login-API-Endpunkt.
- Die Bank implementiert MFA und Rate Limiting für Log-in-Versuche, erkennt kompromittierte Geräte (z. B. Jailbreaks) und validiert jeden Payload gemeinsam mit der API-Gateway-Regel.
- Mit der Hilfe von Monitoring werden Anomalien in Echtzeit erkannt und Angriffe unmittelbar abgewehrt.

Das Ergebnis ist eine deutliche Reduktion von ATOs, eine höhere Sicherheit bei API-Zugriffen und ein robusterer Schutz sensibler Finanzdaten.

Der mobile Faktor – warum Smartphones die Angriffsfläche vergrößern

Mobile Geräte haben die API-Landschaft grundlegend verändert. Sie erzeugen nicht nur den Großteil des API-Verkehrs, sondern stellen auch einen direkten Angriffspunkt dar. Nutzer verbringen mehr als drei Stunden täglich in Apps, doch nur etwa 30 der durchschnittlich installierten 80 Apps werden regelmäßig verwendet.

Diese Diskrepanz schafft einen idealen Nährboden für automatisierte Clients: Alte, ungenutzte oder schlecht gewartete Apps greifen weiterhin auf APIs zu und erweitern so die Angriffsfläche. Hinzu kommt, dass Angreifer mobile Clients gezielt kompromittieren, um API-Tokens zu extrahieren oder Anfragen zu manipulieren.

API Security ist deshalb immer auch Mobile Security.

Warum API Security zur Grundvoraussetzung moderner Resilienz wird

APIs sind das Nervensystem digitaler Unternehmen und gleichzeitig einer ihrer anfälligsten Bereiche. Sie wachsen schnell, sind schwer zu überblicken und werden von einer zunehmend automatisierten und taktisch agierenden Angreifern ins Visier genommen.

Wer die Sicherheit von APIs nicht beherrscht, verliert die Kontrolle über Datenflüsse, Geschäftslogik und im schlimmsten Fall über zentrale Umsatzströme.

Ein modernes Schutzkonzept basiert auf:

- vollständiger Sichtbarkeit,
- intelligenter Laufzeitabsicherung,
- klarer Governance.

In Kombination mit einer integrierten WAAP-Plattform entsteht eine Sicherheitsarchitektur, die nicht nur Angriffe abwehrt, sondern auch die digitale Souveränität eines Unternehmens dauerhaft schützt.

Technische Exzellenz, die spürbaren Schutz liefert

Die Link11 WAAP-Plattform vereint KI-gestützte Schutzmechanismen mit einer europäischen, souveränen Infrastruktur. Damit werden Komplexität, Angriffsfläche und Ausfallrisiken in einem Schritt reduziert. Link11 liefert eine präzise, autonome Abwehr, die Angriffe in Millisekunden stoppt und gleichzeitig legitimen Traffic unverzögert passieren lässt – selbst unter extremen Lastspitzen oder in komplexen Multi-Cloud-Umgebungen.

Mit einer europäischen Private Cloud, einem eigenen globalen Backbone und über 40 Points of Presence liefert Link11 eine Performance, die selbst unter extremer Last stabil bleibt. Die Plattform vereint WAF, Bot-Management, API-Security und Web-DDoS-Schutz unter einem SLA und wird vollständig von einem Expertenteam verwaltet. Dieses stellt rund um die Uhr eine präzise und erklärbare Abwehr sicher. Das Ergebnis ist maximale Sicherheit, höchste Sichtbarkeit und deutlich weniger operativer Aufwand für IT-Teams.

Für Unternehmen, die digitale Resilienz, europäische Souveränität und echte Zukunftssicherheit anstreben, ist Link11 mehr als nur ein Schutzschild. Die Plattform verbindet technische Stärke, operative Entlastung und Compliance-Sicherheit und liefert damit das, was digitale Unternehmen heute zwingend benötigen: echte Resilienz.

Die 5 wichtigsten Vorteile auf einen Blick

- **Zero-Tuning Security:** Keine manuellen Regeln, keine Komplexität. ML-basierte Erkennung, behavioral Analytics und API-native Detection stoppen Web-DDoS, Bots und API-Angriffe in Echtzeit.
- **100 % EU-Souveränität:** Private Cloud, ISO & KRITIS-konform, ohne US-Hyperscaler-Abhängigkeit.
- **Unified Protection:** WAF, Bot-Management, API Security & DDoS – alles auf einer Plattform, einem SLA.
- **Forensische Angriffserkennung:** ML-gestützte Threat Profiling Engine mit minimalen False Positives.
- **Hochverfügbarkeit & Geschwindigkeit:** 40+ PoPs, 13 Scrubbing Center, latenzoptimierter globaler Backbone.

Ihr Ansprechpartner

Michael Scheffler
Vice President Sales

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



Nachweise

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



Hauptsitz

Link11
Lindleystr. 12
60314 Frankfurt