LINK11



WHITEPAPER

State of Bot Management

www.link11.com

Table of Content

Introduction	04
Threat Landscape	05
Traditional Detection Methods	14
Why Traditional Defenses Fail Against Modern Bots	15
What’s Working Today	18
UEBA: User and Entity Behavioral Analytics	19
Modern Bot Detection	20
Optimizing for Business Outcomes	21
Leveraging New Sources of Data	22
Conclusion	23

Introduction

Web Traffic Composition
Robust bot management is more critical than ever due to the rapid evolution of the threat landscape. In 2024, for the first time in over a decade, automated bot traffic surpassed human-generated traffic, accounting for 51% of all web activity.

The Role of Large Language Models in Accelerating Bot Attacks
The surge in malicious bot activity is directly associated with the proliferation of generative AI and Large Language Models (LLMs), such as ChatGPT, Gemini, Claude, and Perplexity AI. These tools have enabled even inexperienced attackers to deploy bots at scale and with high frequency. The barriers to entry fell even further with the emergence of the „Bots-as-a-Service“ ecosystem, which enables technically unskilled actors to launch automated attacks with minimal effort.

From a technical standpoint, attackers are now leveraging AI to analyze failed attacks and continuously refine their methods. This has led to the development of increasingly sophisticated, evasive bots that are more difficult for traditional security systems to detect.

Robust bot management is now essential for web security. On average, across different verticals, only 38 percent of incoming requests originate from human users. The remaining 62 percent have an automated source.

Not all bots are harmful, however. Some (such as search engine spiders) are often welcome, and some (such as content aggregators) are not overtly hostile. However, almost 40 percent of incoming requests come from malicious bots.

Consequences of Inadequate Protection
Threat actors use bots to wage a variety of web attacks. In fact, almost all attacks involve bots in one way or another. Hostile bots that are not identified and blocked can create a variety of problems for organizations with significant web assets (sites and web applications, microservices, and mobile/native API endpoints).

Some of the potential problems are:

Site downtime
when DDoS attacks exhaust app/server resources.

Data theft
from scraping.

Site breaches
via vulnerability discovery bots.

Loss of revenue
from inventory hoarding.

Bad business decisions
from skewed analytics.

Account theft
from credential stuffing.

Merchant account problems
from credit card testing.

Degraded customer experience
from loss of bandwidth & resources.

Threat Landscape

Different Industries, Different Bot Threats
Not all hostile bot traffic looks the same. Different industries attract different types of attacks. This is because attackers target what's valuable in a specific sector. For instance, an airline website may be inundated with bots attempting to scrape flight prices, while an e-commerce site may be overrun with bots hoarding limited inventory.

To better understand and respond to these threats, it helps to ask a simple but powerful question: Who stands to gain? This principle, shortened to WSTG, can guide organizations in identifying likely attackers and their motives. Just as detectives in crime shows "follow the money," cybersecurity professionals can often determine the intent of a bot attack by looking at who benefits from it.

The most common types of bot threats are listed below and will be explored in more detail in the following sections.

Distributed Denial of Service (DDoS):
Disruption at Scale

Credit Card Fraud:
Automation at the Core

Credential Attacks:
Gaining Access at Scale

Inventory Hoarding:
Denial of Access Through Bots

Scraping and Data Theft:
Silent, Scalable Extraction

Vulnerability Scanning:
The First Step Toward Exploitation

Advertising Abuse:
Click Fraud and Revenue Loss

Application Abuse:
Exploiting Functionality at Scale

Distributed Denial of Service (DDoS): Disruption at Scale

Distributed Denial of Service (DDoS) attacks are among the most aggressive and disruptive bot-based threats. Attackers use large networks of bots to overwhelm web applications or APIs with massive volumes of traffic, rendering them unavailable to legitimate users. If the attack traffic cannot be effectively filtered, the disruption can persist indefinitely, entirely on the attacker's terms.

Unlike other types of attacks that can be performed manually, such as vulnerability scanning, DDoS attacks are inherently automated and distributed. The bots involved vary in complexity, ranging from malware-infected PCs to hijacked IoT devices.

Timing and Strategy

DDoS attacks are often carefully timed. In online gaming, for example, they may occur just before a major event in an attempt to drive users toward competing platforms. In e-commerce, attackers often target peak shopping seasons and use downtime as leverage for extortion demands.

WSTG: Who Stands to Gain?

- **Criminals:** Use DDoS for ransom (DDoS-for-hire or extortion).
- **Competitors:** Aim to disrupt operations and divert users.
- **Hacktivists:** Launch attacks as retaliation or protest.
- **State actors:** Suppress dissent by targeting NGOs, the media, and opposition groups.

Consequences:

The impact goes beyond downtime. Businesses may suffer lost revenue, damaged reputation, and eroded customer trust. In severe or repeated cases, search engine rankings may decline, further affecting visibility and traffic over time.

Credit Card Fraud: Automation at the Core

Bots play a central role in modern credit card fraud. Criminals use them to steal, test, and exploit card numbers, resulting in financial losses, chargebacks, and reputational damage for merchants.

One common method is vulnerability scanning, in which bots search for weaknesses in payment-handling websites, such as e-commerce platforms or service portals. Once a vulnerability is identified, attackers breach the system and extract large volumes of card data, often stealing thousands of valid numbers in a single attack.

Stolen card numbers are then validated using bots, which submit the data to online forms to see if the number is accepted or rejected. Attackers also use a brute-force method called "card enumeration," in which bots cycle through number combinations until valid ones are found. Though crude, this method is effective and entirely automated.

A major driver of this trend is the rise of "card-not-present" (CNP) fraud. As EMV chip cards have made in-person fraud more difficult, cybercriminals have shifted their focus to the internet, where bots can be used on a large scale with minimal risk.

WSTG: Who Stands to Gain?

Cybercriminals: They profit by selling stolen cards on dark markets or using them for fraudulent purchases.

Consequences:

Merchants lose revenue when goods are shipped and fraudulent payments are reversed. Repeated fraud can result in financial penalties and, in severe cases, termination of merchant accounts by payment providers.

Year	Estimated losses (USD billion)
2022	17.5
2023	44.3
2024	48
2029	107

77%

of retailers in the US operate online.

77%

of internet users in the European Union shopped online.

84%

of the population in the Nordics shop online.

Credential Attacks:
Gaining Access at Scale

User credentials, such as logins and passwords, are prime targets for cybercriminals. Bots are routinely used in brute-force attacks, in which countless character combinations are tested until access is granted. Alternatively, attackers harvest credentials through large-scale data breaches and leaks.

Once attackers obtain valid credentials, they can be used for further attacks or sold on dark web marketplaces. One common tactic is credential stuffing, in which bots test stolen credentials across multiple websites, taking advantage of the fact that many users reuse the same login details. This often leads to multiple successful account takeovers from a single breach.

Another growing threat is the automated creation or takeover of accounts, where bots generate fake accounts for abuse (e.g. promotional fraud or spam) or take control of real user accounts for theft or exploitation.

WSTG: Who Stands to Gain?

Cybercriminals: They profit by reselling stolen credentials or using them to access sensitive systems and services.

Consequences:

Compromised accounts lead to fraud, data exposure, and reputational damage. Affected organizations may face public backlash, loss of customer trust, and, potentially, regulatory fines, especially in sectors that handle personal or financial data.

84%

of users worldwide **reuse their passwords** for more than one website or application.

Inventory Hoarding:
Denial of Access Through Bots

Web applications that facilitate purchases or reservations are prime targets for inventory hoarding, a bot-driven tactic that blocks legitimate customers from accessing available stock. Bots flood retail or booking platforms by adding products or reservations to carts without completing the purchase. This locks up inventory and creates artificial scarcity.

The travel industry is especially vulnerable to this method. Bots exploit the 15-minute „time-to-checkout“ window by continuously cycling through bookings without finalizing a purchase. This prevents real users from securing tickets or reservations.

The financial damage can extend beyond lost sales. Many travel sites rely on third-party aggregators for flight data. Each user search can trigger a data lookup fee, which is billed regardless of whether a purchase is made. Because bots generate high volumes of non-converting traffic, these fees can quickly become a major cost center with no revenue to offset them.

WSTG: Who Stands to Gain?

Competitors: They use bots to disrupt business operations and drain resources.

Consequences:

Inventory hoarding acts as an application-layer denial-of-service attack, can lead to:

- Lost revenue from blocked purchases.
- Unsold perishable inventory (e.g., event tickets, flights).
- Rising operational costs (e.g., aggregator fees).
- Damaged customer trust due to poor user experience.

59%

of e-commerce traffic comes from bots, a significant portion of which is used for inventory hoarding.

Scraping and Data Theft:
Silent, Scalable Extraction

Scraper bots are designed to extract valuable data from websites and web applications, often without being detected. They target industries that rely on proprietary data or content, such as aggregators, insurance platforms, and online retailers. For data-driven businesses, scraping poses a direct threat to their core revenue models.

In retail, for example, bots harvest product details, pricing, and inventory data. Competitors can then use this information to undercut prices, copy offerings, or replicate user-generated content, such as reviews, which damages brand differentiation and sales performance.

More advanced scraper bots use automated, high-frequency querying to reconstruct entire datasets that are not exposed through a single webpage or API call. For instance, insurance quote tools can be exploited by submitting thousands of quote requests with different parameters, gradually revealing full pricing models.

WSTG: Who Stands to Gain?

- **Competitors:** They use scraped data to gain unfair advantages, such as lower prices, cloned content, or faster time-to-market.
- **Criminals:** They sell stolen data to third parties or use it for fraudulent purposes.

Consequences:

The damage is multifaceted:

- Loss of competitive edge through pricing and content theft
- Reduced search engine visibility due to duplicate content
- Brand erosion occurs when users encounter stale, copied, or misused information elsewhere

AI-powered scraping tools will be standard by 2025: They will recognize and circumvent anti-bot measures, solve CAPTCHAs, and adapt to dynamic websites.

Nine out of ten websites were targeted by bot attacks in 2024, including scraping.

Vulnerability Scanning:
The First Step Toward Exploitation

Cybercriminals use bots to scan large numbers of systems for known vulnerabilities. These automated scans often serve as the starting point for more severe attacks. Once a vulnerability is identified, attackers take action, either immediately or by laying the groundwork for future breaches.

The follow-up tactics depend on the vulnerability. In some cases, bots trigger data theft, malware delivery, or ransomware encryption within minutes. In other cases, attackers install backdoors to maintain long-term access, enabling deeper intrusion over time. This approach is common when targeting complex environments with high-value data.

One infamous example is the Equifax breach, which exploited an unpatched Apache Struts vulnerability. The attack began with an automated scan and ultimately led to a massive data breach that affected millions.

WSTG: Who Stands to Gain?

- **Cybercriminals:** They use scanning as a gateway to broader attacks, often as part of larger campaigns.

Consequences:

Immediate risks:

- Data exfiltration
- Malware infections
- Ransomware deployment

Long-term risks:

- persistent backdoors, network compromise, and reputational damage

Regulatory fallout may include investigations, penalties, and compliance failures in the wake of large-scale breaches.

60% of data breaches are caused by unpatched vulnerabilities.

In large companies, **45.4%** of vulnerabilities remain unpatched for more than 12 months, especially in the network area.

Advertising Abuse:
Click Fraud and Revenue Loss

Click fraud is one of the most underestimated forms of bot-driven abuse. Bots generate fake clicks on ads, which inflates traffic metrics and sabotages ad campaigns. Although click fraud may seem less aggressive than other types of attacks, it can have a substantial financial impact.

Advertisers are the immediate victims, wasting their budgets on fake engagement and receiving poor returns. Publishers (the sites hosting the ads) are affected as well. When bot traffic goes undetected, ad networks often blacklist the site, cutting off its future monetization.

Ad networks closely monitor for invalid traffic. If they detect excessive fraudulent clicks, they reverse previously paid earnings, resulting in direct income loss for the publisher. Even worse, these impressions can't be monetized again because the traffic is gone and the opportunity is lost.



WSTG: Who Stands to Gain?

- **Criminals:** Earn affiliate revenue through fraudulent clicks on their own ad-serving pages.
- **Competitors:** Use bots to drain rival ad budgets or damage their reputation with networks.



Consequences:

- Revenue loss from payment reversals and penalties for invalid traffic.
- Long-term damage as ad networks refuse to provide future inventory.
- Poor campaign performance for advertisers, which undermines business or political goals.

According to the Ad Fraud Report 2025, nearly **12%** **of all clicks on paid ads are invalid** - meaning they come from bots, competitors, or fraudulent sources.

Application Abuse:
Exploiting Functionality at Scale

This category covers a wide range of bot-driven attacks that exploit the unique features of a web application or its APIs. Unlike other threat types, this category isn't tied to a single tactic: it encompasses everything from SMS spam via telecom APIs to unauthorized access to application logic.

Attackers increasingly target API endpoints because they expose core functionality and data. Many of the aforementioned attacks—such as credential stuffing, scraping, and inventory hoarding—can be executed directly through APIs. Some bots are designed to specifically reverse-engineer undocumented APIs, identifying hidden methods or weak points that can be exploited on a large scale.

Mitigating application abuse requires a layered defense:

- Preventing API discovery and reverse engineering
- Enforcing schema validation and proper rate limiting
- Detecting abnormal usage patterns unique to each app or service

This makes application abuse one of the most dynamic and difficult bot threats to manage.



WSTG: Who Stands to Gain?

- **Criminals:** Exploit exposed functionality for fraud, spam, or resource theft.
- **Competitors:** Interfere with services or extract proprietary data.



Consequences:

Application abuse can lead to a widerange of issues, including:

- DDoS attacks on APIs
- Credential-based intrusions
- Inventory hoarding or denial
- Scraping and data theft
- Fraudulent activity or mass spamming

The likelihood of each depends on how the app or API is exposed and how well it's protected.

According to Statista, global losses due to cybercrime will amount to around US$15.6 trillion in 2029 - an **increase of almost 70%** compared to 2024.

Traditional Detection Methods

Historically, bot detection has relied on a few core techniques.

Rate limiting:
Blocking users who exceed defined request thresholds.

IP blacklisting:
Denying traffic from known malicious or suspicious IP addresses.

Environment detection:
Using JavaScript challenges or cookie tests to spot non-browser environments.

Signature matching:
Identifies known bot patterns in request headers or behavior.

CAPTCHA:
Presenting puzzles to differentiate humans from bots.

While these methods are effective against basic or outdated bots, they are ineffective against today's advanced threats. Many modern bots, especially those powered by AI or built to mimic human behavior, can bypass these defenses entirely.

The problem isn't the methods themselves but rather relying solely on them for detection. Unfortunately, many intrusion detection (IDS) and bot mitigation systems still just offer basic services, leaving organizations exposed to evolving, increasingly sophisticated bot attacks.

Why Traditional Defenses Fail Against Modern Bots

Legacy bot detection techniques are no longer sufficient. As attackers become more sophisticated and AI continues to lower the barrier to entry, modern bots are designed to specifically evade traditional defenses.

Here's why these methods fall short:

- **Rate limiting** is bypassed by rotating IP addresses and throttling requests to stay within acceptable limits.
- **IP blacklisting** is ineffective against large botnets and proxy networks cycle through countless IP addresses.
- **Signature recognition** is easily defeated by spoofing headers, such as user-agent strings, to mimic legitimate traffic.
- **Headless browsers** have become more advanced and harder to distinguish from real users, especially since they are widely used for legitimate automation and testing.
- **Environment detection** cannot reliably stop headless browsers, which now support JavaScript execution, cookie handling, and human-like interaction patterns.

These advancements have rendered traditional detection techniques largely obsolete. Effective bot mitigation today requires a more adaptive, behavioral approach—one that can distinguish real users from sophisticated automation in real time.

CAPTCHA: Balancing Bot Defense and User Experience

CAPTCHA is a common method for distinguishing human users from bots. These challenges are presented to suspicious traffic and are designed to be solvable by humans but difficult for automated systems.

CAPTCHA systems are widely used to distinguish between human users and automated bots. These systems introduce challenge-response tests ranging from simple checkbox prompts to image recognition tasks that aim to block malicious traffic without burdening legitimate users too heavily.

However, over time, the bot landscape has evolved significantly. Many AI-powered bots, headless browsers, or CAPTCHA-solving services are capable of bypassing traditional challenges. This has led to an ongoing cycle of increasingly complex CAPTCHA mechanisms that can degrade the user experience.

Solutions like reCAPTCHA v3 attempt to shift away from visible challenges by analyzing user behavior behind the scenes. While this reduces friction, it also raises questions about detection accuracy, privacy implications due to deep integration with browser cookies, and cost structure for high-volume sites.

Despite these advances, no CAPTCHA solution fully eliminates the risk of bot abuse without some trade-offs.

- AI-powered bots can solve text and image CAPTCHAs accurately.
- Automated CAPTCHA-solving services enable large-scale circumvention.
- Traditional CAPTCHAs block basic bots but are less effective against advanced threats.
- Behavioral CAPTCHAs detect bots unobtrusively, improving user experience.

Consequently, many organizations are exploring complementary or alternative approaches to bot mitigation, such as behavioral analytics, fingerprinting, and server-side validation, to strengthen protection without relying solely on challenge-based defenses.

While CAPTCHA remains a widely deployed security layer, it is most effective when combined with broader bot management strategies that address today's adaptive and stealthy threats.

On average, a person needs **32 seconds** to complete a CAPTCHA challenge.

The increasing irrelevance of IP and Geolocation

Traditional attacker tracking methods rely on IP addresses. But today, intrusion detection systems that depend heavily on IP tracking are largely ineffective.

Relying on IP addresses to identify attackers is no longer effective. Modern users—both legitimate and malicious—constantly change IP addresses as they switch between mobile networks, virtual private networks (VPNs), and public Wi-Fi networks. A single device may cycle through multiple addresses in a single session, and a public hotspot may serve thousands of users in a day. Blocking an IP address because of one malicious user could lock out many legitimate customers.

Attackers can exploit this fluidity. Bot operators rotate IP addresses with every request, abuse mobile gateways, and hijack Internet of Things (IoT) devices or browser extensions to hide behind vast pools of addresses. Large-scale attacks routinely arrive as tens of thousands of requests per minute, each from a unique IP address scattered across the globe. No single address appears twice, rendering blacklisting ineffective.

Geolocation is equally unreliable. With IPs shifting so quickly, location data changes from request to request and offers little insight into intent.

IP and geolocation still have some value against unsophisticated threats that don't rotate addresses; however, they now detect only a shrinking percentage of hostile traffic. An effective defense requires deeper, behavior-based signals rather than perimeter assumptions tied to "good" or "bad" IPs.

By 2025, over **60% of bots will use rotating proxies** and IP switching to bypass blocks.

The challenge of API protection

Traditional bot detection methods were primarily designed for web traffic. However, mobile apps and microservices now account for a significant portion of network activity for many organizations. Threat actors often reverse-engineer application programming interfaces (APIs) by analyzing mobile app communications, allowing them to program bots that mimic legitimate application behavior. These bots target vulnerable functions, including credit card validation, credential stuffing, brute force login attempts, and gift card and coupon code abuse. Since any action that a legitimate API user can perform can also be automated by bots, the potential monthly financial damage can reach millions of dollars.

Making matters worse, many conventional bot detection techniques rely on verifying browser-based environments, which APIs lack. This renders traditional methods largely ineffective for securing API endpoints against sophisticated automated attacks.

Unfortunately, many of the traditional methods for bot detection are not useful for API protection. For example, an API user has no web browser environment that could be verified.

In the last two years, **57%** of companies have experienced at least one API-related security incident.

Conclusion

As older bots still represent a significant portion of automated traffic and can be effectively detected using traditional methods, these approaches remain a core part of many bot management solutions due to their efficiency and low computational cost. However, as increasingly sophisticated bots target both web applications and APIs, relying solely on legacy detection techniques is insufficient.

Modern, advanced detection methods are essential to effectively combat these sophisticated threats.

What's Working Today

Modern bot detection solutions improve upon traditional methods by incorporating advanced, proactive techniques. While classic approaches often rely on passive metrics, such as monitoring resource consumption rates and observing user behavior, more effective solutions actively verify the authenticity and intent of each request. These techniques complement each other and should be combined to maximize detection accuracy. Leading platforms like Reblaze employ these methods alongside additional proprietary technologies to provide comprehensive protection.

Client Certification

With the widespread availability of third-party server certificates, clients can verify that their connection is to a legitimate endpoint. The next step is mutual authentication, in which clients present certificates to confirm their identity. While this is common in industries like FinTech, client certification elsewhere remains rare. Link11 offers a client certification feature and encourages its adoption to strengthen security.

Other Client Authentication Methods

In addition to certificates, clients can be authenticated through native or mobile apps using built-in verification mechanisms. Link11 provides an SDK for Android and iOS that customers can integrate into their own apps. The SDK signs the application, authenticates the device, and verifies the user's identity. All communication is secured via TLS and includes an HMAC signature, which is unique cryptographic proof per session or request based on multiple factors, such as time, location, and environment. This approach ensures that traffic originates from legitimate users and not emulators or bots. Additionally, apps using the Link11 SDK benefit from advanced authentication methods, such as biometric behavioral profiling.

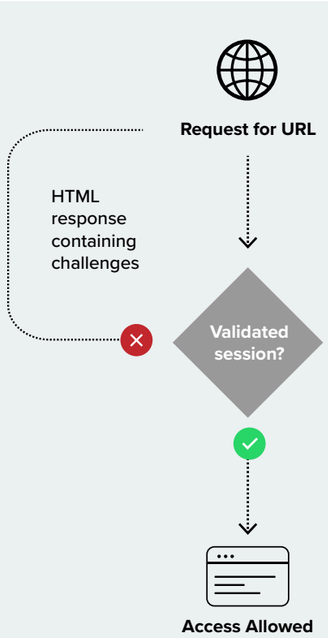
Advanced Browser Verification

Traditional methods of verifying browser environments have become ineffective because modern headless browsers fully support JavaScript execution. Simple JavaScript checks no longer reliably distinguish bots from real users.

Advanced security solutions address this issue by applying more rigorous challenges. For instance, a web application firewall (WAF) might issue a JavaScript-based arithmetic test, flagging browsers that fail to solve it as headless. Other techniques include verifying the browser's ability to render audio or images.

Link11 enhances this approach with a suite of proprietary environmental detection methods that combine JavaScript-based and alternative techniques to provide robust browser verification.

As we have seen, traditional methods of verifying browser environments have become obsolete. The first generation of headless browsers, for example, could not execute JavaScript. Today, however, most automated environments have the ability to execute JavaScript. Therefore, simple JavaScript verification is no longer effective.



UEBA: User and Entity Behavioral Analytics

UEBA: User and Entity Behavior Analytics

UEBA is a process in which data on user behavior is fed back into intrusion detection systems (IDS) in order to establish baselines for normal activity. Subsequent user behavior is then compared to these baselines to identify potential threats. UEBA is a broad approach rather than a single technique, and its implementation varies widely across security solutions—from absent to basic to highly sophisticated.

1. Basic UEBA

At its most basic level, UEBA monitors simple metrics such as typing speed and interaction patterns. For example, human typing varies in speed and rhythm, whereas bots tend to rapidly and mechanically enter long strings. Similarly, clicking the same screen pixel multiple times without typical mouse movements might signal automation.

2. Biometric Behavioral Profiling and Machine Learning

Modern bots can closely mimic human behavior, evading basic UEBA. To counter these advanced threats, more sophisticated UEBA uses machine learning to create detailed behavioral profiles based on biometric data, or measurable human characteristics. This approach analyzes comprehensive datasets from each protected API, including:

- Device and software details (e.g., hardware, screen resolution, battery level, extensions, emulator detection)
- User interactions (e.g., mouse movements, clicks, scrolls, typing speed).
- Session metrics (request patterns, IP usage, timing)
- Consumption analytics (pages viewed and time spent)
- Application-specific events, and more

By processing billions of requests daily, ML identifies subtle patterns and relationships that human analysts might miss. Rather than using rigid rules, the system assigns weighted behavioral scores to each requestor. When these scores exceed thresholds, the requester is flagged and blocked.

UEBA's focus on anomaly detection contrasts with traditional, static, rule-based WAFs, which is why many legacy security products lack this capability.

3. Granular Profiling:

Effective behavioral profiling should be granular, analyzing not just entire apps, but also individual pages or screens. For instance, if most users zoom in on a map when they first open a mobile app, users who never zoom in could be flagged. Similarly, retail visitors who skip typical page interactions, such as scrolling to view details before making a purchase, could be considered suspicious.

These behaviors are weighted factors, rather than strict rules. They allow the system to accumulate indicators and reach highly accurate bot detection. Granular profiling increases robustness against evasion by attackers because it relies on private analytics patterns that threat actors cannot easily replicate or reverse-engineer.

Furthermore, machine learning (ML) uncovers complex, non-obvious behavior patterns, enhancing detection beyond what human analysis alone could achieve.

This layered, data-driven approach makes UEBA one of the most effective tools for distinguishing bots from legitimate users in today's complex threat landscape.

The global UEBA market is growing from US$2.39 billion (2024) to US$3.21 billion (2025) - corresponding to a compound annual growth rate (CAGR) of 34.3%.

Modern Bot Detection

Traditional methods of tracking requestors by IP address and geolocation are no longer reliable. Now, the core basis for detection must be the identity and—more importantly—the behavior of the requestor. Behavior analysis is essential because even verified human users may act maliciously. However, all hostile actors, whether bots or humans, eventually deviate from legitimate behavioral patterns. Behavioral profiling is designed to detect these deviations.

Frontiers of Bot Detection

As bots become more sophisticated, security solutions must continually advance. Key areas for ongoing improvement include:

- Performance Optimization
- Business outcome optimization
- Integration of new data sources
- Optimizing performance

Bot detection methods have matured significantly and generally fall into four categories:

1. Static Rules
2. Statistical Approaches
3. Machine Learning (ML)
4. Hybrid approaches combine ML and statistical analysis

This spectrum ranges from fast, lightweight, and less precise techniques, like static rules, to more complex and resource-intensive methods, like ML. For effective bot detection, an IDS should blend these approaches, using simpler methods for rapid preliminary detection and advanced analytics to ensure high accuracy and reduce false positives.

1. Static Rules

Static rulesets are the simplest form of bot detection. They include access control lists (ACLs), limits on request size or timing, IP blacklists, and similar measures. Because they require minimal processing, static rules are fast and resource-efficient. However, modern bots can easily evade these controls by rotating IP addresses and slowing request rates, among other tactics.

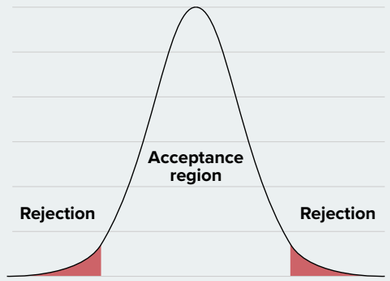
Another drawback is that overly strict static rules often generate false positives and inadvertently block legitimate users. For example, limiting the number of requests per session may incorrectly flag users with longer or more active sessions.

Despite these limitations, static rules remain valuable. Moderately strict rulesets can catch a significant number of hostile bots while maintaining low false positive rates. Using static rules as a first line of defense reduces the need for heavy computation on easily detectable threats.

2. Statistical Approaches

Statistical analysis of traffic requires more processing than static rules but is still relatively straightforward. Common statistical tests, such as the Student's t-test, the Z-test, and analysis of variance (ANOVA), help distinguish normal behavior from abnormal behavior by calculating standard deviations and setting thresholds.

These analyses are typically performed on aggregated data from traffic logs rather than on individual requests in real time. The results can be used to generate „local" or customized rules that an IDS enforces in real time alongside static rules. Ideally, the IDS updates these rules frequently to adapt to evolving traffic patterns with minimal delay.



Statistical methods such as the Z-test provide clear, wellunderstood analytical methods to determine anomalies within a dataset.

3. Machine Learning (ML):

As bots become more sophisticated, applying ML to bot detection is increasingly necessary. Although some solutions have yet to adopt it, ML is essential for detecting bots that mimic legitimate human behavior and evade static or statistical detection.

Earlier ML models focused on identifying anomalies and outliers using methods like Random Sample Consensus (RANSAC), which estimates model parameters while ignoring outliers. However, modern bots are more subtle and require more complex behavioral analysis.

Like statistical methods, ML uses aggregated data to develop detection rules that are applied in real time. However, ML requires larger datasets—approximately ten times more observations per feature—and is more difficult to configure. Selecting the right features and understanding data distribution requires specialized expertise, which can pose a challenge for some users.

4. Combining ML and Statistical Techniques

No single detection method is perfect or universally applicable. Even theoretically ideal techniques would quickly be circumvented by attackers developing new evasion strategies.

In practice, the most effective bot detection solutions combine static rules, statistical analysis, and ML. Simple bots can be quickly blocked using lightweight methods, while more advanced threats are identified through deeper analysis.

The key challenge for IDS providers is to deliver flexible platforms that seamlessly integrate multiple detection methods with user-friendly interfaces that minimize the need for in-depth ML expertise.

Optimizing for Business Outcomes

When evaluating a bot detection solution, accuracy is usually the first metric that comes to mind. Accuracy is easy to calculate: the number of correctly identified requests divided by the total number of requests. However, while useful, accuracy alone doesn't tell the whole story.

It's tempting to pursue a "perfect" detection algorithm—one that delivers the highest possible accuracy in all scenarios. In reality, however, no such universal algorithm exists. Bot detection is highly context-specific. An approach tailored for one use case may perform poorly in another. What works well for an e-commerce platform may not be ideal for a banking API, and vice versa.

Moreover, accuracy alone can be misleading. A high overall accuracy rate can mask critical failures. Two types of errors are especially important to consider: false positives (FPs) and false negatives (FNs). Thus, a complete measurement of accuracy becomes:

$$Accuracy = \frac{(TP + TN)}{(TP + FP + FN + TN)}$$

(Where TP and TN represent True Positive and True Negative events, respectively.)

- FPs occur when legitimate users are incorrectly blocked.
- FNs occur when malicious requests are incorrectly allowed.

Both types of errors are problematic, but not equally so. Their impact depends on the business context. In retail, for example, FPs can directly result in lost revenue if customers are prevented from completing purchases. In contrast, an FN in a highly regulated sector, such as finance or healthcare, could lead to a data breach, resulting in fines, lawsuits, or reputational damage.

Therefore, the most effective bot detection system is not necessarily the one with the highest accuracy but rather the one that best aligns with the organization's risk tolerance and business priorities. Sometimes, slightly lower accuracy with fewer FNs is preferable, even if it means accepting a higher rate of FPs, depending on which error carries the greater cost.

In order to optimize outcomes, IDS must be adaptable. They should enable organizations to:

- Prioritize minimizing FPs or FNs based on their specific needs.
- Tune detection strategies per application or endpoint; balance performance and protection dynamically.

Unfortunately, most legacy web security products offer limi-

ted customization. Fine-tuning detection typically necessitates revealing internal detection parameters and presenting performance feedback in a user-friendly manner that does not overwhelm users with technical complexity. Building this kind of interface is challenging but necessary. As bot tactics evolve and regulatory and commercial pressures increase, IDS customization will become essential.

This brings us to a practical challenge: How can users know if their IDS is making FP or FN errors? After all, the IDS itself can't always distinguish between correct and incorrect decisions. This question leads us to our next topic.

Leveraging New Sources of Data

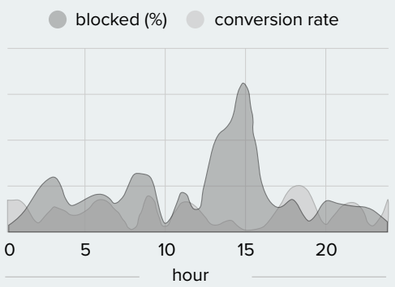
Traditionally, bot detection has focused on analyzing incoming traffic by looking at request headers, IP addresses, behavior, and other technical signals. However, traffic data isn't the only source of valuable insight. Forward-thinking security strategies are increasingly incorporating application-layer metrics, especially conversion rate (CR), as part of their detection and optimization process.

CR refers to the percentage of visitors or requests that result in a defined goal, such as making a purchase, submitting a form, downloading a file, or completing another key action within the application.

Bots generally do not complete these goal-oriented behaviors. Therefore, when IDS correctly identifies and filters out bots, conversion rates should remain steady because those visitors were unlikely to convert in the first place.

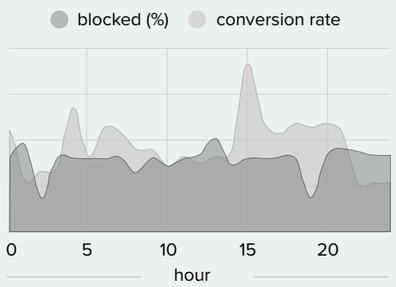
However, if the IDS begins blocking legitimate users (i.e., generating false positives), those users can no longer perform the intended actions. This causes a drop in the conversion rate, which may signal that the detection rules are too aggressive.

Blocked requests vs. Conversion Rate



Comparison of blocked requests to conversion rate. For hour 15, as more requests were blocked, conversion rate fell to near-zero. This indicates a probable high rate of FPs being generated by the IDS.

Blocked requests vs. Conversion Rate



Comparison of blocked requests to conversion rate. There appears to be little correlation between blocked/passed requests and CR. The fluctuations in CR are thus attributable to other factors (e.g., normal variations in buyer behavior throughout the day), rather than being the result of FPs and FNs.

This creates an opportunity. The conversion rate can serve as a feedback mechanism for adjusting the IDS. For example:

- If new detection rules or algorithm updates lead to a decrease in CR, it may indicate an increase in false positives. An immediate review is warranted.
- Conversely, if more traffic is blocked but the CR remains steady (or improves), it suggests that primarily non-converting bots were filtered and the changes were effective.

Note that this assumes the CR is calculated using all incoming requests, including those flagged as bots later on. Some systems calculate CR only from visitors who pass bot screening. In that case, false negatives (bots misidentified as humans) will impact CR, and the feedback loop becomes more complex. Either way, conversion tracking still provides actionable insight.

Beyond conversion rate, other application analytics can also be useful. Integrating the IDS with analytics platforms like Google Analytics allows for a deeper correlation between detection events and user behavior patterns. This, in turn, can help estimate FP and FN rates more accurately, which the IDS alone cannot always do.

Summary:

Network traffic logs are no longer the sole source of intelligence for bot detection. By combining behavioral profiling, conversion metrics, and broader analytics, organizations can better evaluate the effectiveness of their IDS, minimize false positives, and optimize for both security and business performance.

Conclusion

Protecting against malicious bots is no longer optional; it's a strategic necessity for any organization with a significant web presence. On average, nearly 40% of incoming web traffic consists of hostile automated activity. While the specific threats vary by industry, the outcomes are consistent: degraded performance, compromised data, skewed analytics, and lost revenue.

Despite the scale and sophistication of today's threats, many IDS rely on outdated techniques, such as blacklists, static rate limits, signature-based rules, and traditional CAPTCHAs. These legacy methods are increasingly inadequate. Modern bots are engineered to evade them with ease.

To meet current and future challenges, detection strategies must evolve. This is where modern technologies like UEBA (user and entity behavioral analytics), machine learning, and biometric behavioral profiling come into play. These approaches analyze user behavior at a granular level, distinguishing legitimate users from highly sophisticated bots. This behavioral intelligence, especially when applied to individual pages, sessions, or actions, provides a far more robust and resilient defense.

While these advanced techniques are not yet widespread in the industry, they are quickly becoming essential for reliable bot mitigation. Importantly, effective bot detection must be adaptable and tuned to specific business use cases, risk tolerances, and evolving attack methods.

As outlined in the Frontiers section, the future of bot detection lies in the deeper integration of performance analytics, business outcomes, and real-world feedback loops, such as conversion rates. Leveraging broader data sources, such as network logs, application behavior, and business KPIs, allows organizations to fine-tune their IDS performance, improving both security posture and operational results.

Link11 is at the forefront of this evolution, incorporating these next-generation capabilities into its platform while exploring new frontiers in detection accuracy, performance, and resilience.

Sources

2025 Bad Bot Report | Resource Library
https://cropink.com/ecommerce-fraud-statistics
https://ec.europa.eu/eurostat/web/products-eurostat-news/w/ddn-20250220-3
European e-commerce: Report + 2025 insights - E-commerce Germany News
World Password Day - Global Survey 2025 | Bitwarden
https://www.mind-verse.de/post/ki-fuer-bot-detection-zukunft-cybersicherheit
https://latesthackingnews.com/2025/01/03/data-scraping-in-2025-trends-tools-and-best-practices/
https://hackernoon.com/web-scraping-in-2025-staying-on-track-with-new-rules
https://www.indusface.com/blog/key-cybersecurity-statistics/
https://www.getastra.com/blog/security-audit/cyber-security-vulnerability-statistics/
https://www.helpnetsecurity.com/2025/04/30/edgescan-2025-vulnerability-statistics/
https://www.edgescan.com/stats-report/
https://www.kim-weinand.de/ad-fraud-report-2025/
https://www.statista.com/forecasts/1280009/cost-cybercrime-worldwide
https://prosopo.io/blog/what-is-the-future-of-captcha-and-online-privacy/
Botting in 2025: Why CAPTCHA-Solving Services Are No Longer Optional – Blog
https://clickpatrol.com/does-captcha-stop-bots-the-effectiveness-and-fut/
https://securityboulevard.com/2025/03/captchas-demise-multi-modal-ai-is-breaking-traditional-bot-management/
reCAPTCHA v2 vs v3: Effective Bot Protection? [2025 Update]
http://azapi.ai/blog/best-captcha-solvers/
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://www.thebusinessresearchcompany.com/market-insights/user-and-entity-behavior-analytics-market-overview-2025
https://www.thebusinessresearchcompany.com/report/user-and-entity-behavior-analytics-global-market-report

LINK11

Headquarter

Link11
Lindleystr. 12
60314 Frankfurt