IMPERVA®

Mitigating OWASP Automated Threats

Learn how Imperva SecureSphere Web Application Firewall and ThreatRadar combine to stop automated threats in their tracks

WHITEPAPER

Executive Summary

Automated threats are a major challenge for IT security teams defending the plethora of web applications upon which modern organizations have come to depend. These threats are trivial to replicate once they are developed and focus primarily on abusing inherent functionality rather than exploiting conventional software vulnerabilities, in web applications. They are not only pervasive but also inherently difficult for traditional countermeasures to detect.

To bring greater attention and clarity to this situation, the Open Web Application Security Project (OWASP) recently published its Automated Threat Handbook for web applications.[1] This groundbreaking research explains the challenges with the automated threat problem and provides a common language for the industry to use when coming up with mitigation and control strategies.

This paper builds on the foundation the OWASP Handbook puts in place. Using the classification scheme from the Handbook as a framework, it demonstrates how Imperva SecureSphere Web Application Firewall and Imperva ThreatRadar intelligence services combine to provide modern organizations with precisely what they need: effective protection against automated threats, both now and in the future.

The Rise of Automated Threats

Defining automated threats as they pertain to web applications is best accomplished by outlining their key characteristics. Automated threats:

- Execute a series of actions that an unassisted human could not practically accomplish – for example, conducting a high volume of authentication attempts in the case of credential stuffing, or having the precise timing needed to perform a sniping attack on auction sites.
- Focus primarily on abuse of valid application functionality, rather than on seeking to exploit conventional software vulnerabilities, or implementation bugs – for example, where malicious spammers insert rogue URLs/links to skew search engine results (also known as a "spamdexing" attack)
- Are closely – but not exclusively – tied to bots, which also run automated tasks over the Internet

In fact, because of its relationship to automated threats, bot activity is an excellent proxy for understanding the scope of the automated threat problem. According to the 2015 Incapsula Global Bot Traffic Report[2]:

- Bots account for approximately half of all website traffic
- Malicious bots account for close to 30 percent of total bot traffic
- Impersonator bots – the most sophisticated and malevolent of all bot types – continue to grow, accounting for 25 percent of total bot traffic

[1] OWASP Automated Threat Handbook, October 2015

[2] 2015 Bot Traffic Report, Incapsula (an Imperva Company), December 2015

Regardless of whether bots are involved, a key point to understand is that once an automated threat is created, re-using it for future attacks requires little more than picking a target. It is no surprise, therefore, that automated threats have become such a pervasive problem.

Adding to the level of concern for today's enterprises is the potential impact of automated threats. Consider first the fundamental nature of these threats. Because they are designed to abuse valid application functionality, they are able to evade popular countermeasures that focus on detecting exploitation of traditional software vulnerabilities.

Next, consider the damage they can do. Potential downsides range from performance degradation and application downtime to hijacked accounts and data theft. The cost of an attack can easily exceed hundreds of thousands of dollars in lost revenue and remediation efforts – not to mention reputation damage and potential blacklisting if your site is converted into a watering hole or used to wage spam attacks.

OWASP – Building a Foundation for Thwarting Automated Threats

Having recognized not only the rise of automated threats but also the lack of clarity in attempts to address them – due in part to inadequate visibility and inconsistent naming practices – OWASP set out to correct these deficiencies. Intended as a first step, the Automated Threat Handbook establishes a detailed ontology, lingua franca, and classification scheme for disparate parties to use when discussing and treating the problem. The Handbook also signals OWASP's intent to publish related materials to further identify applicable symptoms and deliver guidance on potential mitigations and recommended control strategies.

"The focus for the project is the abuse of functionality - misuse of inherent functionality and related design flaws, some of which are also referred to as business logic flaws. There is no coverage of implementation bugs … The threat events are scenarios which are seen commonly by real operating web applications, and are multi-step and/or highly iterative and/or multiple weaknesses involved, and not primarily about events that relate to the tool-based exploitation of single-issue vulnerabilities of individual web applications."

OWASP AUTOMATED THREAT HANDBOOK

Table 1 below shows how the OWASP Automated Threats Events can be mapped to specific OWASP Attack Categories.

OWASP ATTACK CATEGORY	OWASP AUTOMATED THREAT EVENTS
Abuse of Functionality, General	Account Creation, Ad Fraud, CAPTCHA Bypass, Cashing Out, Expediting, Scalping, Skewing, Spamming, and Sniping
Abuse of Functionality, Brute-force	Carding, Card Cracking, Credential Cracking, Credential Stuffing, and Token Cracking
Abuse of functionality, Lift Sensitive Data	Account Aggregation and Scraping
Component and Vulnerability Scanning	Fingerprinting, Footprinting, and Vulnerability Scanning
Denial of Service	Denial of Service

Table 1: Threat Classification of OWASP Automated Threat Events

A comprehensive definition for each OWASP Automated Threat (OAT) can be found in the Handbook. Other details the Handbook provides include the industry sectors commonly being targeted, the parties (or classes of users) commonly being affected, and the types of data commonly being misused or at risk.

Imperva – Delivering Real-time Protection from Automated Threats Today

Although the OWASP Handbook provides a critically important starting point for addressing automated threats targeting web applications, it leaves the treatment of controls and mitigation strategies to be covered in a future installment. For organizations looking to do something now, there is the Imperva SecureSphere Web Application Firewall and Imperva ThreatRadar intelligence services. Combining core web application firewall capabilities, crowd-sourced threat intelligence, and the ability to create correlated, compound detection and mitigation policies, the Imperva solution delivers effective protection from automated threats, with minimal occurrence of false positives and no need to modify your organization's web application in any way.

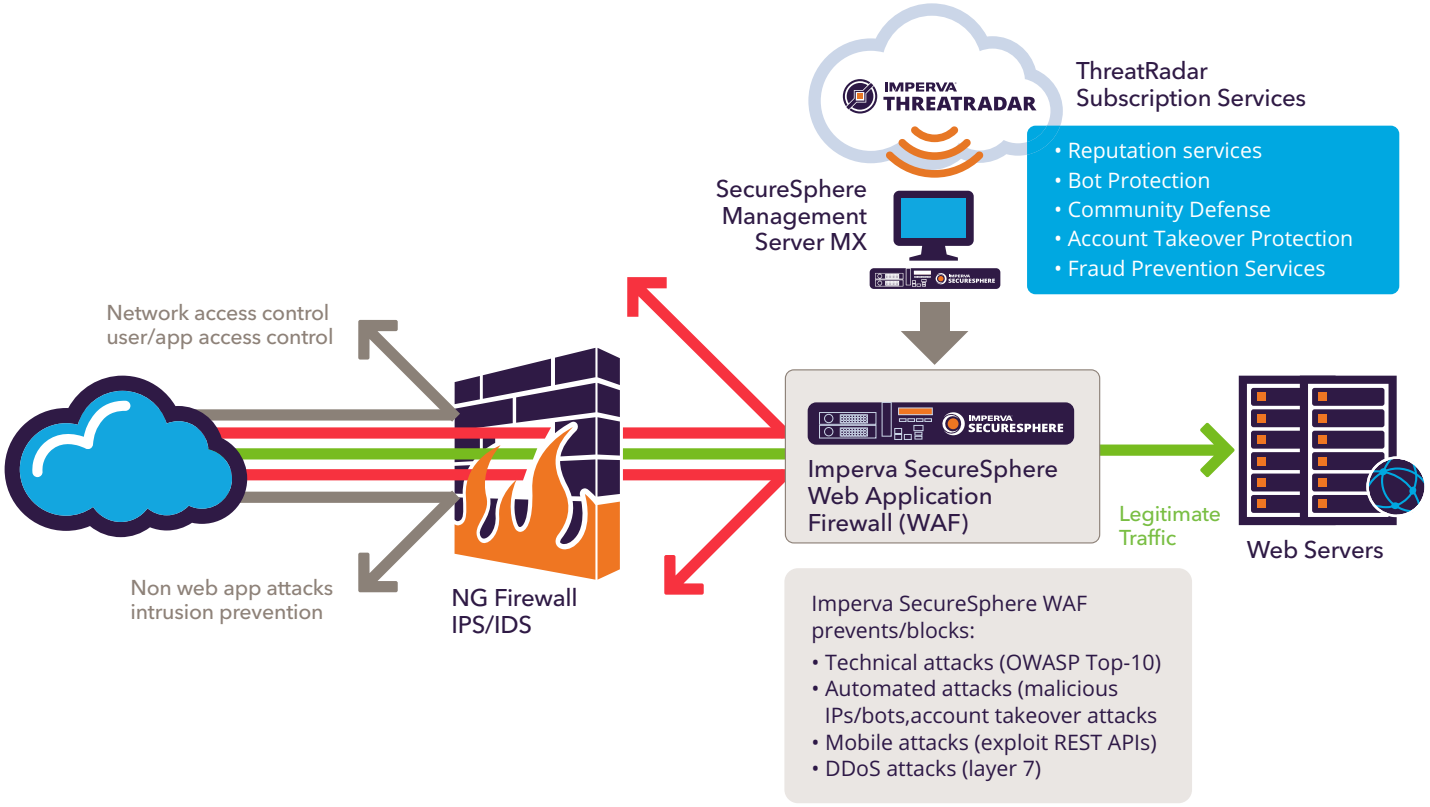


Figure 1: Imperva SecureSphere Web Application Firewall with ThreatRadar

Core web application firewall features.

IT and application security teams can use several native features of the SecureSphere Web Application Firewall to help stop automated threats. These include:

- IP list – block designated networks/sources from gaining access to specific applications and services (or, vice versa, to only allow access from designated networks)
- Rate limiting – identify when page requests are being made at an abnormal (or inhuman) rate
- Time of day – combine with rate limiting to identify abnormal traffic volume for a specific timeframe, which may be related to scraping
- Cookie enforcement – distinguish site scraping agents from true web browsers
- Forceful browsing – distinguish real users from automated threats by detecting when web pages are being accessed in an unexpected order

- Anti-scraping – identify when an excessive number of unique pages or resources are being accessed from a single source
- Automated site access/reconnaissance – detect a high volume of requests from a source that is returning suspicious response codes
- Vulnerability scanning – detect a high volume of requests from a single source that violates the dynamically learned application profile
- Cookie tampering – detect requests with altered cookies when the application did not expect them to be altered

ThreatRadar services.

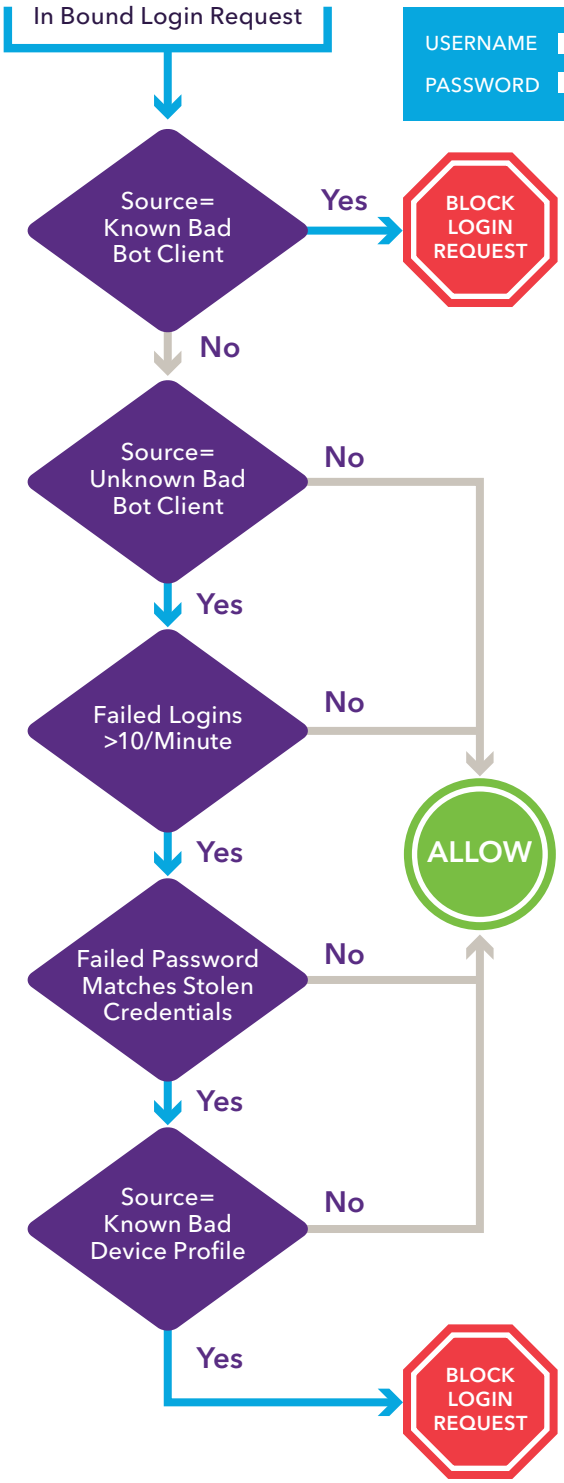
Crowd-sourced threat intelligence curated by the application research organization in the Imperva Defense Center provides a second set of capabilities for countering automated threats. Relevant subscriptions that work in conjunction with the SecureSphere Web Application Firewall include:

- ThreatRadar Reputation Services – identifies known malicious and otherwise troublesome source IPs, such as those associated with anonymous proxies, TOR networks, phishing attacks, and comment spammers
- ThreatRadar Community Defense – identifies live attacks seen by other Imperva customers, who have opted-in to share related data and indicators of compromise, including spamdexing URLs and source IP addresses associated with malicious scanning and SQL injection and remote file inclusion attacks
- ThreatRadar Bot Protection – distinguishes between human and bot sources of incoming traffic, good and bad types of bots, and "imitation" browsers used by bots to fool detection mechanisms into concluding they are human users
- ThreatRadar Account Takeover Protection – leverages a combination of both credential intelligence and over 2.5 billion device intelligence entries to detect attempts to gain unauthorized access to provisioned web application and privileged accounts

In most cases, engaging these capabilities is simply a matter of selecting the pre-defined detection policies to apply for each site, server group, application, or service the SecureSphere Web Application Firewall is protecting.

The power of correlation.

Each SecureSphere capability and ThreatRadar intelligence feed works directly to defend against automated threats. However, the combined solution delivers even greater protection – not to mention a reduction in the frequency of false positives – by enabling correlation of numerous events and conditions to establish a more informed picture of what is really happening in a given situation.



Comprehensive Protection Against OWASP Automated Attacks

BENEFITS

1) Power of correlating multiple attack conditions

2) Seemless interaction with realtime threat intelligence

3) Reduced false positives and false negatives

Figure 2: Sample SecureSphere security policy represented in a decision tree

In addition to taking advantage of numerous pre-defined correlation policies, administrators can craft their own custom policies to account for other pieces of data available through SecureSphere, such as irregularities in how HTTP or XML services are being used. The net result is another powerful mechanism that is available to help thwart the rising tide of automated attacks threatening an organization's essential web applications.

For easy reference, **Table 2** (at the end of this document) summarizes the SecureSphere with the corresponding ThreatRadar services that apply to each category of automated threats, and the sample policies security teams can configure to protect specific types of "sample resources."

Real-world Protection for OWASP Automated Threats

SecureSphere Web Application Firewall dynamically profiles protected web applications, discovering all exposed application interfaces and automatically learning acceptable navigation paths and user responses, such as legitimate form field values. Layering in ThreatRadar enhances these foundational defenses by eliminating known sources of malicious activity and enabling correlation of web application firewall events with extensive threat, device, and credential intelligence to significantly reduce the occurrence of both false positives and negatives.

The following real-world use cases further demonstrate the effectiveness of this powerful combination, specifically for addressing the growing problem of automated threats.

Vacation Rental Site

For one of the largest vacation rental sites in the world – over one million listings spanning 190 countries – critical security concerns include protecting property owner information and website integrity. Among the specific threats they were struggling to address were comment spamming and site scraping.

- Comment spamming – Cyber attackers were using bots to submit inquiry forms on specific vacation rental properties at a higher than normal rate (e.g., ten/minute). This was creating unnecessary backend manual work for the customer.
- Site-scraping (with a side of phishing) – Bots were leveraging site-scraping techniques to harvest property owner's email addresses. These addresses would then be leveraged in a phishing campaign to obtain associated credential information. With valid account credentials in hand, attackers would then proceed to rent properties without the corresponding owner's knowledge.

Implementing SecureSphere Web Application Firewall in conjunction with ThreatRadar mitigated both of these threats. Application requests originating from known malicious bots were dropped "out of hand," while the use of custom correlation policies – such as 'block unknown bots making POST requests at a rate that exceeds X' and 'block unknown bots using known scraping techniques' – weeded out the remaining threat traffic. The net result was not only the savings of the one million dollars they had been losing each month to such attacks, but also the restoration of their customer's confidence and satisfaction in the service being provided.

SaaS-based Health Insurance Management

For this cloud-based healthcare management provider, normal operations consist of its customer's employees logging in to manage and view their healthcare benefits, approved doctors, medical claims and other related information. Ongoing threats faced by this multi-tenant, SaaS solution included credential stuffing and account creation attacks.

- Credential Stuffing – Bots regularly conduct brute-force, credential reconnaissance attacks on the service's login pages, where positive outcomes lead to subsequent account takeover attacks (including exposure of sensitive personal/healthcare information, not to mention unauthorized changes to an employee's healthcare coverage and services).
- Account Creation – Attackers were regularly targeting registration pages in an effort to create fake accounts, which could then be used in a variety of malicious/unauthorized ways (e.g., for comment generating comment spam, spreading malware, or otherwise degrading the service's reputation).

Conclusion

Automated threats that target essential web applications are a major challenge for today's organizations, especially given their ability to evade commonly deployed countermeasures. The OWASP Automated Threat Handbook is an invaluable resource that brings much needed attention and clarity to this growing problem.

For organizations looking to take the next step – that is, to mitigate and avoid the costly impact of such threats – Imperva SecureSphere Web Application Firewall and ThreatRadar intelligence services are also invaluable resources. With this tightly integrated, two-pronged solution, IT security and application teams obtain comprehensive protection that accounts for all twenty types of automated threats identified by OWASP, with minimal risk of false positives and no need to modify their organization's web application in any way.

To learn more about SecureSphere, ThreatRadar, and other Imperva solutions for protecting your organization's data, applications, and reputation, please visit www.imperva.com.

THREAT CATEGORIES	OWASP AUTOMATED THREATS	CORE WAF	TR REP[1]	TR CD[1]	TR BOT[1]	TR ATO[1]	SAMPLE POLICIES
Abuse of Functionality, General	Account Creation, Ad-Fraud, CAPTCHA Bypass, Expediting, Skewing, and Spamming	✖	✖	✖	✖		Sample policies to apply: • Block malicious bots – block IPs with known bad reputation and tools commonly used to perform massive registrations • Block unknown bots – for bots with no specific reputation, leverage mechanisms such as IP reputation OR geo-location OR velocity-checks OR source-location OR indicators of forceful-browsing OR CAPTCHA challenge to deliver granular protection • Whitelist known good clients – allow known good IP addresses, user agents, match signatures Sample resources to apply to (for threat X): • Registration pages (for Account Creation) • Check-out pages (for Cashing Out) • User-feedback pages (for Skewing, Spamming)
	Cashing Out	✖	✖	✖	✖	✖	
Abuse of Functionality, Brute-force	Carding, Card Cracking, Credential Cracking, Credential Stuffing, Token Cracking	✖	✖		✖	✖	Sample policies to apply: • Block unknown bots – for bots with no specific reputation, utilize mechanisms such as IP reputation OR geo-location OR velocity-checks OR source-location OR indicators of forceful-browsing OR CAPTCHA challenge to deliver granular protection • Block clients – block access when there are repeated login failures AND credential matches OR a successful login after repeated failures from same device. • Block clients – when using stolen credentials OR weak credentials from dictionary based attacks • Whitelist known good clients – allow known good IP addresses, user agents, match signatures Sample resources to apply to (for threat X): • Login pages (for Credential Cracking/Stuffing) • Check-out pages (for Carding, Card Cracking) • Coupon-entry pages (for Token Cracking)
Abuse of functionality, Lift Sensitive Data	Account Aggregation, Scraping	✖	✖		✖		Sample policies to apply: • Block malicious bots – block IPs with known bad reputation and tools commonly used for site scanning • Block unknown bots/clients – for bots/clients with no specific reputation, utilize velocity rules per IP/session/user OR CAPTCHA challenge to deliver granular protection Sample resources to apply to (for threat X): • Entire site (for Vulnerability Scanning, etc.)
Component and Vulnerability Scanning	Fingerprinting, Footprinting, Vulnerability Scanning	✖	✖	✖	✖		Sample policies to apply: • Block malicious bots – block IPs with known bad reputation and tools commonly used to perform scraping attacks • Block unknown bots/clients – for bots/clients with no specific reputation, utilize velocity rules per IP/session/user OR CAPTCHA challenge to deliver granular protection Sample resources to apply to (for threat X): • Product/catalog pages (for Scraping)
Denial of Service	Application Denial of Service, Scalping, Sniping	✖	✖		✖		Sample policies to apply: • Block malicious bots – block IPs with known bad reputation and tools commonly used for denial of service attacks • Block unknown bots/clients – for bots/clients with no specific reputation, utilize velocity rules per IP/session/user OR CAPTCHA challenge to deliver granular protection Sample resources to apply to: • Sensitive data pages (e.g., login, registration, password reset) • Resource-intensive pages (e.g., videos, galleries, complex business logic or back-end operations)

Table 2: Imperva capabilities and sample policies for mitigating OWASP automated threats

[1] ThreatRadar (TR) Services: TR Rep = Reputation; TR CD = Community Defense; TR Bot = Bot Protection; TR ATO = Account Take Over