IMPERVA®

10 Things Every Web Application Firewall Should Provide

WHITE PAPER

Introduction

Because they are easily accessible and often serve as an entry point to valuable data, web applications are now—and always will be—a prime target for attack. Indeed, it is not surprising that hacktivist attacks taking down corporate and government sites, DDoS attacks against major financial institutions, and massive Web breaches resulting in millions of compromised credit card numbers and personal data records are in the headlines practically every week. Hidden behind the front page headlines, too, lurk tens of thousands of unreported breaches—unexplained website outages, temporary website defacements, small-scale fraud incidents—that never make their way into news articles. This is because cybercriminals don't just target brand name companies; they are equal opportunists, constantly seeking out vulnerable sites to compromise, disable, or deface. And their weapons of choice are technical web attacks, business logic attacks, and fraud—which traditional network security defenses, like firewalls and intrusion prevention systems (IPSs), are largely incapable of preventing.

Technical Web Attacks

If hackers were surveyed about their favorite attack vectors, technical web attacks, like SQL injection and cross-site scripting (XSS), would undoubtedly top the list. This assumption is borne out by analyses of hacker forums and application attack traffic. In fact, SQL injection alone accounted for almost one fifth of all hacker forum discussions.[1] And, according to security research, nearly two thirds of organizations experienced one or more SQL injection attacks that evaded their firewall and other perimeter defenses in the past year (while the average time to detect these attacks was an astounding 140 days).[2]

To accelerate the rate of technical web attacks, cybercriminals have become "industrialized." They leverage a combination of off-the-shelf attack toolkits, infected 'bots,' and search engines to quickly find and exploit web application vulnerabilities. The industrialization of hacking has made technical attacks much more automated and dangerous.

Business Logic Threats

Hackers aren't stopping at traditional web attacks. Business logic attacks and fraud are also becoming increasingly popular techniques. Today, hackers exploit business logic flaws to post advertisements in online forums.

They scrape websites for valuable intellectual property. They perform repeated brute force attacks. They use wildcards in search fields to bring applications to a screeching halt. These attacks have left many organizations at wits' end, because application scanners cannot detect business logic flaws and secure development processes can't always mitigate them.

"Input validation is the single best defense against injection and XSS vulnerabilities."

BRENT HUSTON, STATE OF SECURITY

[1] "Monitoring Hacker Forums," HII Report #5, Imperva

[2] "The SQL Injection Threat Study," Ponemon Institute

Online Fraud

In addition, hackers have turned their sights to unsuspecting website visitors, infiltrating millions of computers with malware like Zeus and SpyEye. This malware steals user credentials and hijacks sessions by tracking keystrokes and manipulating website content. While the malware targets end users, the true victims are the website owners; often banks and ecommerce sites, which must pay fraud restitution costs.

Together, web application attacks, business logic attacks, and fraud can cost organizations millions or even billions of dollars. Breaches stemming from web-based attacks can result in brand damage, customer churn, lost revenue, fines, and lawsuits. Many victims of web application attacks have invested in customer notification and credit card monitoring services for their customers. In several instances, large-scale breaches have even driven companies out of business.

Why Network Security Fails

When businesses first connected to the Internet in the early 1990s, they encountered the precursor to modern day hackers: malicious users that probed computers for open ports and platform vulnerabilities. To prevent breaches, organizations deployed firewalls and intrusion prevention systems (IPSs). However, when these same organizations opened up access to their web applications, hackers quickly circumvented the firewalls, using evasion techniques like encoding and comments to evade IPS signature detection.

Next generation firewalls came onto the scene a few years later and they offered more capabilities; they could identify the type of application traffic—like HTTP or instant messaging. But, unfortunately, this application awareness provides zero benefit in terms of stopping web attacks. Next generation firewalls cannot block attacks that exploit custom web application vulnerabilities. They cannot detect cookie, session, or parameter tampering attacks. They cannot stop fraudulent devices or business logic attacks. Organizations that rely solely on network security solutions to protect their applications shouldn't be surprised if they suffer a web application breach.

A WAF should "automatically receive and apply dynamic signature updates from a vendor or other source."

RECOMMENDED WAF CAPABILITY IN THE PCI DSS INFORMATION SUPPLEMENT: APPLICATION REVIEWS AND WEB APPLICATION FIREWALLS CLARIFIED

Web Application Firewalls Are Strategic

Web application firewalls have become the central platform for protecting applications against all online threats including technical web attacks, business logic attacks, and online fraud. Unlike traditional network security solutions, web application firewalls understand web usage and validate input to stop dangerous attacks like SQL injection, XSS, and directory traversal. They block scanners and virtually patch vulnerabilities. And they rapidly evolve to prevent new attacks and keep critical applications safe.

Because web application firewalls are strategic, organizations must carefully evaluate a products' security, management, and deployment capabilities. The remainder of this paper explains the top 10 features that every web application firewall should provide.

Web Application Firewalls Must:

1. Understand Web Applications

Challenge: Organizations face the growing specter of advanced, custom web attacks. The richness of JavaScript and SQL allows hackers to devise a virtually unlimited number of SQL injection and XSS attacks. Signatures can help detect web attacks, but they must either be written broadly to catch any potential threat—resulting in false positives—or they must define the exact syntax of the attack—resulting in false negatives. In addition, hackers can use encoding, comments, and obfuscation to evade signature detection. Using evasion methods or a little creativity, hackers can easily outwit traditional security solutions to compromise web applications.

Requirement: To accurately stop attacks, a web application firewall must understand the protected application, including URLs, parameters, and cookies. Understanding the protected application and validating input helps stop attacks like SQL injection, parameter tampering, and cookie poisoning. To validate input, a web application firewall must inspect parameter values for special characters like apostrophes and brackets and know whether these characters are expected or indicative of an attack. Dynamic profiling, or the ability to automatically build a baseline of acceptable user behaviors (i.e., requests and responses) is also important in this regard—not to mention critical to minimizing false positives. Since organizations frequently update applications, a web application firewall also needs to automatically learn application changes—without any manual intervention. Automated application learning makes managing a web application firewall manageable, while still providing the highest level of protection available.

"Web application firewalls must deliver more sophisticated control at the application layer through a variety of contextual rule sets and behavioral analysis."

CORE OF THE MATTER, SANDRA KAY MILLER, INFORMATION SECURITY MAGAZINE

2. Stay Ahead of Hackers

Challenge: Hackers constantly innovate. Whether it is because they are creating new attack tools, developing new ways to recruit volunteers, or honing existing techniques, application threats are always evolving. Nowhere is this more evident than on underground forums, where hackers continually unveil new attack vectors and vulnerabilities. Fraudsters are not standing still either; fraud malware developers have architected self-mutating files that can evade virus signature detection. Cybercriminals today can circumvent physical token, smartcard, and out-of-band authentication to perform fraud.

Keeping up with the latest application threats—including vulnerability exploits, malicious users, and fraud schemes—is perhaps the most difficult hurdle for application security solutions.

Requirement: A web application firewall must have up-to-date protection to defeat the latest web-borne threats. It should leverage live attack, reputation, and fraud data from around the world to identify both attacks and attackers. Security signatures, policies, reputation data, and fraud intelligence should be updated automatically without human intervention. Besides the frequency of security updates, it is important to look at the research organization that is producing security content. Is the research organization focused on web application security? Is it equipped to defeat the latest application attacks? If the answer to either of these questions is no, then businesses ought to move on and investigate alternative solutions.

3. Thwart Evasion Techniques

Challenge: Organizations need to block web attacks without blocking legitimate traffic. This may sound obvious, but the solution, unfortunately, is not. A web application firewall must validate input (see requirement #1), but it shouldn't block requests with accidental typos. If a hacker enters unusual characters like brackets and apostrophes—characters often used in web attacks—into a zip code field in an online form, a web application firewall should detect the unusual behavior. But what if a valid user inadvertently types five digits and a quote? How do you differentiate between a cybercriminal that executed an attack and a web user that accidentally submitted special characters in a form field? How do you construct attack signatures that detect SQL keywords, like "select" and "union," and "join," but that still allow legitimate requests with these same words? The answer: advanced analytics and correlation.

Requirement: A web application firewall must include an analytics engine that can examine multiple attack indicators to block attacks without false positives. This analytics engine must be able to evaluate factors such as attack keywords, special characters, protocol violations, and known attack strings simultaneously. It should identify violations and then perform additional analysis using risk scoring and regular expressions to differentiate between malicious requests and unusual, but harmless traffic. A web application firewall must also correlate requests over time to detect repetitive attacks, such as brute force login or Distributed Denial of Service

"When we talk about hackers, we are talking about a fully organized, well-oiled machine intent on gaining money. And hacking is most definitely a big industry.

SECURITY WEEK, "THE STRUCTURE OF A CYBERCRIME ORGANIZATION"

(DDoS). Only a flexible and intelligent correlation engine will enable a web application firewall to stop sophisticated hackers without blocking legitimate users.

4. Prevent Automated Attacks and Bots

Challenge: Cybercriminals have become industrialized, using automation to improve efficiency and scale—and, in turn, transforming hacking into a multi-billion dollar industry. Armed with web scanners and bots, cybercriminals today can quickly discover vulnerable sites. Then they can leverage off-the-shelf toolkits like the Havij SQL injection tool to extract sensitive data. On top of the threat posed by industrialized hackers, organizations today are inundated with other automated attacks—like competitors scraping web content, comment spammers injecting ads into online forums, and disgruntled individuals or groups launching site-crippling DDoS attacks.

Requirement: A web application firewall must be able to stop automated attacks like site scraping, comment spam, application DDoS, and vulnerability scans. Due to the explosion in automated attacks, stopping malicious users can be as important as stopping malicious requests. But correctly identifying the bad guys requires multiple defenses. First, a web application firewall should be empowered with real-time reputation intelligence that identifies known attacks sources, bots, phishing URLs, and anonymizing services and allows it to block malicious traffic before an attack can even be attempted. Secondly, a web application firewall should be able to recognize bots—the automated clients that are responsible for the lion's share of automated attacks. This can be achieved by transparently testing web users' browsers to determine if they are standard browsers or simple bots or scripts.

Other giveaways—indicative of application DDoS attacks in particular—include repeated downloads of large files or requests that generate long response times. To mitigate network-level DDoS attacks designed to saturate your Internet connection and prevent legitimate traffic from ever reaching your site, a web application firewall should also include integral support for a high-capacity, cloud-based DDoS protection service.

5. Recognize Malicious Sources

Challenge: Not all web visitors are good. Some actively try to uncover vulnerabilities, steal data, commit fraud, or take down websites. Many of these malicious visitors aren't human at all—they are bots that continuously attack one site after another. Human hackers are sneakier and more sophisticated than bots; they use anonymous proxies or Tor networks to cloak their identity. And fraudsters have their own unique attack vectors, such as stealing user credentials through phishing sites. The problem organizations face today is that they cannot identify malicious users or illicit sites until the damage is done—a bot has requested too many web pages, a hacker has conducted reconnaissance, or a phishing attack has succeeded.

"On average, 31% of Website visitors are intruders. These shady non-human visitors include hackers, scrapers, spammers, and spies of all sorts."

INCAPSULA RESEARCH

Requirement: A web application firewall must recognize known malicious sources and sites. It should identify users that are actively attacking other websites and stop them instantly, before they can inflict more damage. Because hackers often use anonymizing services, a web application firewall should detect access from anonymous proxies and Tor networks. To combat phishing, it should recognize users referred from a phishing site. Geographic location provides additional context about web users; a web application firewall should be able to restrict access by location both to eliminate unwanted traffic and to thwart DDoS attacks originating from a specific country. Since web application firewalls are the most effective solutions at detecting web-based threats, then web application firewalls ought to collect and share information about attacks and attack sources among one another. A cloud-based community defense should deliver accurate, live information about hackers and bots and fraudsters; such intelligence-based solutions are the future of application security.

6. Virtually Patch Vulnerabilities

Challenge: Despite the best efforts of application developers and IT security teams, most applications have vulnerabilities. In fact, according to one report, more than three quarters of scanned sites were found to have at least one vulnerability[3]. In addition, 1 in 8 were found to have a "critical" vulnerability—one that would make it trivial for a hacker to access sensitive data or alter the site's content. Worst of all, the average length of time to fix discovered vulnerabilities is 38 days or more, leaving applications exposed to attack for long periods. Besides the cost and the time required to fix vulnerabilities, organizations must consider additional hurdles like vulnerabilities in legacy applications—which may have been untouched for years—and in packaged applications—which may entail obtaining (if available) and implementing patches from application vendors.

Requirement: A web application firewall must prevent attempts to exploit application vulnerabilities. Defenses such as input validation, HTTP protocol validation, and attack signatures must be able to block most vulnerability exploits out-of-the-box. However, organizations need granular control to ensure strict security measures are applied to known application vulnerabilities. To achieve this capability, a web application firewall can integrate with application scanners and build custom policies to virtually patch vulnerabilities discovered by the scanners.

7. Stop Malware

Challenge: Fraud malware has become enemy #1 for financial institutions. Cybercriminals are now leveraging their success with online banks to branch out into other applications like ecommerce and bill payment. So, how do cybercriminals carry out malware-based fraud? First, they infect machines with malware such as the Zeus or SpyEye Trojans. Then, when infected

"Web Application Firewalls genuinely raise the bar on application security… they 'virtually' patch the application faster than code fixes can be implemented."

ADRIAN LANE, SECUROSIS

[3] "Internet Security Threat Report, Volume 19,"Symantec

users log into a targeted web applications such as online banking sites, the malware modifies web pages, performs unauthorized transactions, or steals login credentials. While fraud malware targets website users, the ultimate victims are the website owners–typically banks and e-tailers–who are forced to reimburse customers for fraudulent transactions.

Requirement: A web application firewall must be able to mitigate the growing scourge of fraud malware. Positioned between web users and applications and with full visibility into layer 7 transactions, web application firewalls can analyze end user attributes and web traffic patterns for the tell-tale signs of malware infection and block malware-infected devices. They can also perform a number of actions, such as monitoring the user for a specified period of time, generating an alert, or integrating with a fraud management solution to open an investigation case. A web application firewall must be able to provide this capability without requiring any changes to the protected web application.

8. Eliminate Payment and Account Origination Fraud

Challenge: Just like fraud malware, online payment fraud costs organizations millions of dollars. E-tailers must contend with expensive chargeback fees, notification costs, and unhappy customers due to a range of Internet fraud schemes. So how can organizations fortify their applications against fraudulent users? And how can they roll out these fraud defenses quickly, without requiring expensive and protracted application development projects?

Requirement: A web application firewall must be able to mitigate payment and new account fraud without requiring application changes. A web application firewall must be able integrate with cloud-based fraud security solutions to extract and analyze a number of user and transaction attributes , including browser irregularities, known fraudulent devices, and suspicious payment information. The web application firewall should correlate fraud risk data with web attack and user information to accurately identify and stop fraud.

9. Support on Premise and Cloud Deployment

Challenge: Application architectures are as diverse and rapidly evolving as application threats. Consider the different configurations and security needs of organizations today. Some businesses host their applications on-premises; others host their applications in the cloud. Some organizations need a transparent security solution with zero impact on applications; others wish to change application content by rewriting URLs and encrypting cookies. Some companies require a high- performance hardware appliance; others desire a flexible virtual appliance. And large enterprises need it all: a variety of deployment options to support distinct applications hosted by different divisions in different locations.

"A layered fraud prevention approach provides defense in depth, and it is the best policy for preventing and containing losses that result from today's and tomorrow's threats."

AVIVAH LITAN, GARTNER

Requirement: A web application firewall must provide flexible deployment and configuration options to satisfy every organization's unique requirements. As many businesses transition their application infrastructure to the cloud, web application firewalls must adapt, supporting virtual appliance solutions for private clouds and cloud-based security services to protect hosted web applications.

Organizations that host their applications on premise have at least as many demands as their cloud brethren. Many of them require a high performance solution with no changes to existing applications or network devices. Others need a web application firewall that can modify content, sign cookies, rewrite HTML. And others yet require non-inline deployment, enabling the IT security team to ease into inline deployment over time. During evaluation, organizations should verify that potential solutions can support their on-premise and cloud requirements now and in the future.

10. Automate and Scale Operations

Challenge: Web application attacks can be complicated. Stopping those attacks shouldn't be. Security administrators ought to be able to create custom security policies without learning a scripting language. Another challenge for many organizations is that they operate dozens—and even hundreds—of web servers, and these servers may be located in different data centers or even in different countries. Organizations must be able to centrally manage application security policies and monitor events at a global level. Lastly, organizations must be able to investigate security incidents. So, they need detailed security alerts and customizable reports for monitoring and forensics.

Requirement: A web application firewall must deliver point-and-click security policies. Simple, but flexible policy configuration not only eases initial configuration, but it also makes it easier for administrators to review security policies developed by their peers. In addition to custom policies, web application firewalls must also support centralized management so that businesses can synchronize policies and application profiles across all of their web application firewalls, even if those devices are located in separate data centers or separate continents. Lastly, web application firewalls must provide detailed, actionable security event information. Armed with this data, administrators can understand how hackers are attempting to undermine their applications. Robust monitoring and reporting, along with centralized management and flexible policy configuration, provide organizations the capabilities they need to successfully manage, monitor, and secure their web applications.

"Cloud-based security services offer an easy and effective way to make websites faster and protect websites against hackers and bots."

LAWRENCE PINGREE, GARTNER

Conclusion

Hackers have become industrialized, using automated tools to steal data. Hacktivists have emerged from almost nowhere to become a major threat, compromising and disabling hundreds of wellknown websites. Fraudsters have ratcheted up their schemes to perpetrate online fraud while evading detection. And, unfortunately, network security products like firewalls and intrusion prevention systems are unable to stop these growing risks. Web applications drive businesses more today than at any other time in history. To adequately protect these business-critical resources, organizations need a web application firewall. However, not just any web application firewall will do. Establishing strong, thorough protection depends on selecting a web application firewall that fully meets the requirements and supports the essential capabilities identified in this paper.

About Imperva

Imperva® (NYSE: IMPV), is a leading provider of cyber security solutions that protect business-critical data and applications. The company's SecureSphere™, Incapsula™ and Skyfence™ product lines enable organizations to discover assets and vulnerabilities, protect information wherever it lives—onpremises and in the cloud—and comply with regulations. The Imperva Application Defense Center, a research team comprised of some of the world's leading experts in data and application security, continually enhances Imperva products with up-to-the minute threat intelligence, and publish reports that provide insight and guidance on the latest threats and how to mitigate them. Imperva is headquartered in Redwood Shores, California.

"When buying an application firewall, be sure to understand what is needed to protect, both now and in the foreseeable future. If the web applications touch internal resources, configuration is a critical aspect of deployment... Finally, be sure that there are adequate reporting and analysis functions so that admins can not only analyze attack attempts, but meet applicable regulatory requirements."

SC MAGAZINE REVIEWS, APPLICATION SECURITY