WHITE PAPER

Best Practices for Web Application Firewall Management

Trustwave®

Smart security on demand

WHITE PAPER

Best Practices for Web Application Firewall Management

Introduction

While many organizations deploy a web application firewall (WAF) to meet a compliance requirement, for most, the top reason to deploy a WAF is to protect web servers and applications from being exploited via an application vulnerability. Most applications have vulnerabilities. In a recent study, the Trustwave SpiderLabs team identified at least one vulnerability in 100% of the applications they investigated. Most had more than one.

You might ask "why not fix the vulnerabilities and skip the WAF?" While there's a trend to incorporate secure development early in the application development process, the level of security expertise within development teams varies widely. Development practices can introduce vulnerabilities as well. Vulnerabilities can slip through in new applications as developers are pushed to deliver applications faster than ever. Using open source components and/or third-party code can lead to unknown bugs and flaws becoming part of applications. For the foreseeable future, organizations developing applications should expect that those applications will have vulnerabilities.

For this reason, a WAF is a necessary tool for protecting web servers and applications from attack. Like any security tool though, a WAF needs to be effectively deployed and managed to provide sustainable value.

To help you be more effective with your WAF, the top experts at Trustwave are sharing best practices ideas in this white paper.

We've divided the best practices into three topic areas:

- Deployment
- Technical Processes
- Management Processes

Our Trustwave experts have decades of WAF experience from supporting the ModSecurity Open Source WAF, building the Trustwave WAF, and other work in the industry. We hope you can use these ideas to make your WAF more effective.

Deployment Best Practices

The more time you spend planning for your WAF deployment, the more effective your WAF implementation will be over time. Work through these best practices before and as you deploy.

DOCUMENT YOUR SECURITY RISK TOLERANCE

Your organization's risk tolerance should impact how you set up your WAF policies. For example, if your organization is a large e-commerce operation, you might have a high tolerance for risk. You don't want any legitimate traffic to be blocked, as the revenue you get from your e-commerce business outweighs the risk of being successfully attacked. On the other hand, if you support a prestigious law firm, you might have a very low-risk tolerance. You're willing to let some legitimate user activity be blocked as a trade-off for avoiding an attack and the bad publicity that could come with it.

Most WAFs have options for monitoring and/or blocking web application traffic. Choose the monitoring and blocking options that best support your risk tolerance. For example, the law firm in the example above might:

- Set their WAF to block traffic for all web applications
- Set up custom rules, like the length of time a visitor is allowed access to your website after completing a challenge-response like a CAPTCHA test
- Create custom error pages, so that if a legitimate user is blocked, they'll see information on other ways to engage with the business

DOCUMENT APPLICATIONS AND OWNERS

Different applications serve different purposes, so your organization's overall risk tolerance might not apply to every application. Talk to each owner about what protection their application requires. Having the knowledge about applications and their purpose will help you as a security professional support your business, rather than impede it. It'll also help you quickly identify owners if something goes wrong.

Your organization might have want to apply a single security policy for all applications. Or you might be open to setting different policies for different applications as it makes sense. For example, you might set your WAF to block your web content management system because you expect it will have ongoing vulnerabilities, but for other applications, set the WAF to logging mode only.

Understanding applications can further help as you set WAF policies. For example:

- Deploy WAF policies that makes sense for your applications and technology stack. Your WAF vendor will likely provide a comprehensive set of policies for a variety of applications and technologies. Take the time to tune these. For example, if your website is developed in Joomla, turn off the Drupal policies. It'll save you time reviewing events, reduce complexity, and false positives. It'll also save computing resources by reducing the load on your WAF and improve responsiveness from a latency perspective.
- If you know the services that will be behind the WAF, extend the WAF policies to include all of them (or have some service traffic bypass the WAF). We frequently see WAF deployments that look for HTTP traffic but miss application traffic coming from iOS or Android applications.
- You can set most WAFs to block or ignore traffic using connection methods not used by your applications. Let's say your applications use GET, POST, and HEAD but not OPTIONS, PROPFIND, or PUT methods, block those ones that aren't used and avoid unwanted traffic. It'll also lock down your application so that methods used in development don't get pushed to production by mistake.

IDENTIFY WHAT TO RESTRICT AND ALLOW

Deployment is a good time to identify basic things you'll want to restrict and allow via your WAF. Geographic restrictions are an easy start. If your company operates in specific geographies, restrict access to HR portals to only countries where you have employees. It'll reduce the load on your WAF and web servers. Attackers around the world are constantly scanning web sites vulnerabilities to exploit. Restricting geographies where you don't operate can reduce the load on your web servers.

Similarly, but from a whitelisting perspective, configure your WAF to allow trusted traffic. One example is to allow your vulnerability scanner to interact completely with the back-end web applications and not be blocked by the WAF. Doing so let's you identify web application vulnerabilities that are currently mitigated by the WAF. From an application scanning perspective, you'll get greater visibility and identify vulnerabilities that might be good candidates for developers to address in future releases.

DEPLOY THE WAF IN-LINE

WAFs have different operating modes. There are in-line modes like reverse proxy and transparent bridge, where the WAF sits in-line between web requests and web servers. An in-line WAF can have a lot of control over web requests, such as blocking and/or masking traffic that doesn't meet policies (both incoming and outgoing). There's also out-of-line (sometimes called out-of-band) mode, where the WAF investigates a copy of the web traffic. In this mode, the WAF's ability to block traffic is limited. It can only send TCP-reset packets to interrupt traffic which means some malicious traffic may reach your web server before the TCP-reset happens.

As a best practice, choose an in-line mode. From a security perspective, the risk with an out-of-line WAF that malicious traffic will reach your web server and a subsequent successful reply to attackers is too great. Better to deploy an in-line mode WAF in a way that meets your security and application requirements than take on that risk. There's also an issue with being able to log traffic. An out-of-line WAF won't be able to decrypt Diffie-Hellman traffic which is the most common encryption method in use today. So, in addition to the heightened security risk, you'll be blind to much of your web traffic.

CREATE AN ACCOUNT FOR DEVELOPERS

Application developers can benefit from the information coming from the WAF, so it makes sense to set up an account (or accounts) for them. While a WAF is first and foremost a security control, most WAFs provide valuable application performance information. This information includes things like application change alerts, performance metrics, identification of sensitive data, and broken links.

Some error messages from the WAF will also be of interest to developers. Traffic that looks malicious is often created by non-malicious users who interact with applications in ways the developer didn't anticipate. WAF error messages that contain details like default WAF settings or directory listings of folders and files can indicate application issues that need to be addressed.

Technical Processes

The following are a few best practice ideas to help your WAF technically perform better.

LEVERAGE EXCESSIVE ACCESS RATE CONTROLS

In addition to legitimate traffic hitting your WAF, expect your WAF to also receive traffic from multiple automated sources, like vulnerability scanners constantly pinging your site to look for areas to exploit. Your WAF should be able to distinguish normal user vs automated behavior through the access rate. A normal user probably can't interact with your web applications a thousand times in a few minutes where an automated program can easily do so.

Within the Trustwave WAF, there's a capability to set a rule that allows only a certain number of hits from a source within a specified period of time. If a source exceeds the threshold set in the rule, the WAF can shut down access from that source/IP address or suspend access for some time. Setting this rule is a useful best practice as it can minimize the traffic load on your WAF and web servers.

It's worth nothing that as you implement this best practice, you'll need to identify within the WAF any IPs that might be generating traffic that could exceed the threshold that are serving a legitimate purpose, like check up services, application servers, or scanners.

USE DATA LOGGING AND MASKING

Your WAF, like other security devices, will generate a log (or logs) of data on system activity, WAF activity, web traffic, events, and more. As a technical best practice, you should log some WAF data while at the same time, mask data types that you don't need to keep in your WAF environment.

Much of the data the WAF generates is useful. For example, if you capture HTTP transactional logs when web applications generate error conditions, you can use that data to determine if the errors are generated by an actual attack or by communication or other issues within applications. If you have one in place, you can also send your WAF log data to your SIEM for additional correlation of events.

On the other hand, you should mask some data the WAF is capturing, like sensitive information such as passwords, user login details, and credit card numbers. From a security perspective, you might also want to mask data like security hashes being exchanged between parts of your applications.

MONITOR FOR WEBSITE CLONING

One type of event to look for in your WAF logs is anything suggesting your website is being cloned. Leeching content and scripts from a website to create a malicious clone is a popular way to set a trap that looks and feels legitimate to website visitors but isn't. Hackers might create a malicious clone to simply damage an organizations operations or brand, or to trick users into sharing information like user name, passwords or other Personally Identifiable Information (PII).

While there might be a legitimate reason to clone a website, it's worth paying attention if/when it happens in case it's not a valid situation.

Management Processes

A WAF, like any other product you install, needs to be managed. Someone needs to be responsible for updates and making sure the WAF is up-and-running. With new releases, you'll want to have someone see what new capabilities your vendor has added and decide if they are ones your organization can benefit from. There might also come a time when you decide you need to increase the number of WAF devices you have in place or decommission them.

You also need to make decisions on how your WAF will fit into the rest of your business processes. For example, who responds to alerts, what's your incident response workflow, how do you deal with false positives and your exception process.

In addition to the ideas above, below are several specific best practices to add to your WAF management processes.

DEVELOP A HABIT OF MONITORING WEB TRAFFIC

Whether you can get web traffic information from your WAF or other sources like your network team, get into the habit of checking web traffic on a regular basis. Depending on the nature of your website and web applications, you'll soon see the trend for how much traffic you typically receive and when.

Variations from the norm will quickly highlight potential issues. When a new CVE comes out for a serious new web application vulnerability, you might see spikes in web traffic from scanners looking for a specific port or service that could lead them to it. The quick visibility into changing web traffic behavior can give you a warning to look for and fix or mitigate the vulnerability if it applies to you.

While a reduction in traffic could indicate a known event, like a holiday, it could also point out a web server or application issue. For example, if a web or database server isn't available and can't respond to requests, you'll likely see less traffic. Same if administrator accidentally deletes files and makes some URLs now available. Monitoring web traffic for these types of variations from the norm will give you visibility into potential issues and outright problems.

INVEST IN USER EDUCATION

Earlier in this white paper, we discussed how finding people with the application development and security skills needed for effective WAF management. Even if you're fortunate and find or have the perfect team, a key management best practice is to invest in user training. Ensure the WAF team is trained on application security best practices, web attacks, and incident response, as well as the specifics of the WAF technology they have deployed.

If you are budget-constrained, look at the Additional Resources section of this document for free training sources. Ideally you can carve out time each week for team members to invest in their own education and training.

Additional Resources

There are great resources available to help you better understand how to make your WAF more effective.
These are a few free ones that we recommend.

- Trustwave SpiderLabs blog. Trustwave's team of ethical hackers, forensic investigators and researchers offer blog articles on a wide range of security topics.
  https://www.trustwave.com/Resources/SpiderLabs-Blog/
- OWASP, the Open Web Application Security Project, has many useful resources about web application security and WAFs. One piece we recommend is "OWASP Best Practices: Use of Web Application Firewalls."
  https://www.owasp.org
- Center for Internet Security (CIS) offers more than 100 configuration guidelines for various technology groups
  https://cisecurity.org
- Cybrary. Free online and on-demand training on a variety of security topics.
  https://cybrary.it

ABOUT TRUSTWAVE

Trustwave helps businesses fight cybercrime, protect data and reduce security risk. With cloud and managed security services, integrated technologies and a team of security experts, ethical hackers and researchers, Trustwave enables businesses to transform the way they manage their information security and compliance programs. More than three million businesses are enrolled in the Trustwave TrustKeeper® cloud platform, through which Trustwave delivers automated, efficient and cost-effective threat, vulnerability and compliance management. Trustwave is headquartered in Chicago, with customers in 96 countries.

Trustwave offers an enterprise WAF solutions and contributes to the ModSecurity Open Source WAF project.

- **Trustwave WAF:** Continuously monitor application traffic, enforce policies to block threats, and address compliance requirements, including the PCI DSS. Available as an enterprise solution or as a Managed WAF service.
- **ModSecurity Rules & Support:** Trustwave provides a commercial certified rule set for ModSecurity®2.9.X that protects against known attacks that target vulnerabilities in public software.

For more information about Trustwave, visit https://www.trustwave.com

Trustwave®

Smart security on demand

TRUSTWAVE.COM