Web Application Firewall - WAF

Estupiñan Ojeda Juan Evangelista
Universidad Piloto de Colombia
Bogotá D.C. – Colombia
Juaneo2002@hotmail.com

Abstract--- This article is focused on the use of WAF tool (Wireless Application Firewall), we will describe the history, function and importance of this segment has the security web applications development and protection systems must have the entities to maintain the confidentiality, the integrity and availability of information.

Index Terms--- WAF, Firewall, IPS, XSS, CSRF, SQL injection

Resumen--- El presente artículo está enfocado al uso de la herramienta WAF (Wireless Application Firewall), se van a describir los antecedentes, funcionamiento y la importancia que esta tiene como segmento de seguridad en el desarrollo aplicaciones web y en los sistemas de protección que deben tener las entidades para mantenerla confidencialidad, la integridad y disponibilidad de la información.

Índice de Términos--- WAF, Firewall, IPS, XSS, CSRF, SQL injection

I. INTRODUCCIÓN

Con el uso creciente de las aplicaciones vía Web en las organizaciones para soportar su operación y mantenerse en contacto con sus clientes, se hace crítico evaluar los esquemas de seguridad que las entidades poseen para garantizar la integridad y la disponibilidad de la información, usando el recurso de internet.

El acelerado crecimiento y la tendencia de las organizaciones a nivel mundial de contar con un sitio Web para mostrar su plataforma de productos y hacer procesos de actualización de datos por este medio, se convierte en un factor crítico de seguridad de la información transmitida desde y hacia internet.

Con el uso de herramientas de propósito general como lo son el firewall y el IPS, no es suficiente, se hace necesario el uso de una herramienta dedicada exclusivamente a la protección de tráfico WEB y es lo que se denomina WAF – Web Application Firewall y ese es el tema a tratar en este documento.

II. ANTECEDENTES

Ataques: En el proceso de ejecución y acceso a páginas web usando el protocolo HTTP y HTTPS que son actividades relacionadas con eventos asociados al puerto 80 y 443 surgen ataques como: SQL Injection, inyección de comandos, inyección LDAP, inyección de secuencias de comandos (skript injection), XPath Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), Manipulación de Campo oculto (Hidden Field Tampering), Manipulación de Parámetros (Parameter Tampering), Envenenamiento de cookies (Cookie Poisoning), Los ataques de desbordamiento de búfer, Browsing Forceful, El acceso no autorizado al servidor Web, y Remote File Inclusion (RFI). Este tipo de ataques no pueden ser detectados ni por un Firewall ni por un IPS.

Historia: El primer Ataque de hackers reportado a servidores web a gran escala, fue el PHF CGI en el año 1996 [1], Este evento conduce a la investigación sobre los modelos de seguridad para proteger las aplicaciones web. Este fue el comienzo de lo que se conoce actualmente como la familia de la tecnología de firewall de aplicaciones web (WAF). Los primeros participantes en el mercado comenzaron a aparecer en 1999, como Perfect Software AppShield, (quien más tarde cambió su nombre por el de Sanctum, y en 2004 fue adquirida por Watchfire (adquirida por IBM en 2007), que se centró principalmente en el mercado de comercio electrónico, protegiendo entradas de caracteres ilegales en páginas web. NetContinuum (adquirida por Barracuda Networks en 2007) se acercó al problema proporcionando "servidores de seguridad" preconfigurados. Estos pioneros enfrentaron problemas en derechos de propiedad y demasiados costos de desarrollo porque aún no se

veía latente la necesidad de seguridad y no tenían un buen patrocinio económico, sin embargo, la necesidad de este tipo de soluciones fue echando raíces.

En 2002, el proyecto de código abierto ModSecurity , dirigido por Thinking Stone y más tarde adquirido por Breach security en 2006, se formó con la misión de resolver estos obstáculos y hacer accesible la tecnología WAF para todas las empresas. Con el lanzamiento del conjunto de reglas principal, fue creado un único código abierto para proteger las aplicaciones Web, basado en OASIS Web Application Security Technical Committe's (WAS TC), el sistema tenía un modelo estandarizado, estable y bien documentado.

En 2003, el trabajo de WAS TC se amplió y estandarizó en toda la industria a través de la lista Top 10 de OWASP Open Web Application Security Project. Este es el ranking anual de un esquema de clasificación para las vulnerabilidades de seguridad web, un modelo para proporcionar orientación para la amenaza inicial, el impacto, y una manera de describir las condiciones que pueden ser utilizados tanto por los instrumentos de evaluación y de protección, como un WAF. Esta lista comenzaría a convertirse en el referente de la industria durante muchos planes de cumplimiento.

En 2004, los grandes proveedores de gestión de tráfico y de seguridad, entraron en el mercado a través de una oleada de fusiones. Clave entre ellos fue a mediados de año con la empresa F5 para adquirir Magnifire WebSystems y la integración de la solución de software TrafficShield de este último con el sistema de gestión del tráfico Big-IP. Este mismo año, F5 adquirió AppShield y suspendió la tecnología. Una mayor consolidación se produjo en 2006 con la adquisición de Kavado por Protegrity y la compra de Teros Citrix Systems.

Hasta este punto, el mercado WAF fue dominado por Nicho Providers que se centraron en seguridad de la capa de aplicaciones web. Ahora el mercado está firmemente dirigido a la integración de productos WAF con las grandes tecnologías de red, balanceadores de carga, servidores de aplicaciones, firewalls, etc - y comenzó una oleada de cambio de marca, el cambio de nombre y el reposicionamiento de la WAF. Las opciones eran confusas y costosas.

En 2006, se formó el Web Application Security Consortium para ayudar a darle sentido al mercado WAF, se creó el Web Application Firewall Criteria Project (WAFEC), disponible para la comunidad de usuarios, proveedores, académicos y analistas independientes e investigadores el cual creó un criterio común de evaluación para su aprobación WAF que todavía se mantiene hoy en día.

En 2006 PCI Security Standards Council. Entidad que supervisa y tiene control sobre las principales empresas proveedoras de tarjetas de crédito tales como Amex, Visa, Mastercard, etc) forman la norma PCI como una forma de regular las prácticas de seguridad en toda la industria y reducir el fraude de tarjetas de crédito. En particular, esta norma estableció que todas las aplicaciones web deben estar seguras, ya sea a través del desarrollo seguro o el uso de un WAF (requisito 6.6) usando como base el Top 10 de OWASP.

Con la mayor atención a la virtualización y la computación en nube para maximizar los recursos existentes, la escala de la tecnología WAF ha convertido en el más reciente hito.

Para el 2010, el mercado WAF había madurado en el mercado a más de $ 200 Millones de dólares en tamaño de acuerdo a la revista Forrester. Se esperan fuertes incrementos en ventas por la adquisición de productos WAF en el comercio [2].

Actualmente la empresa Imperva es la líder en productos WAF y esto se corrobora con el cuadrante mágico de Gartner, la cual se encuentra en el cuadrante líder y es seguida de F5, Citrix, Barrauda Networks y Akamai en el cuadrante Challengers y por último 11 empresas más en el cuadrante Niche Players.

III. FUNCIONAMIENTO

Los WAF o Web Application Firewalls. Es un tipo de plataforma que se integra dentro de la arquitectura de red de la organización (generalmente en la DMZ) de manera que se altera la ruta del tráfico web (HTTP y/o HTTPS) para que atraviese este dispositivo. En general se trata de una máquina que simula ser, a ojos del cliente final que solicita un recurso web el servidor final. En realidad es lo que se denomina Proxy Inverso, puesto que la petición web del cliente es analizada por el software WAF y, si la considera "sana", genera una petición contra el servidor Web final que se hace en dos peticiones

diferentes, una desde el cliente hasta el WAF y una nueva desde el WAF hasta el servidor web final.

Un WAF trabaja como intermediario entre usuarios externos (ej. usuarios de Internet) y las aplicaciones web. Esto quiere decir que las peticiones y respuestas HTTP son analizadas por el WAF antes de que éstas lleguen a las aplicaciones web o a los usuarios de las aplicaciones.

Para la revisión del tráfico HTTP, el WAF aplica un conjunto de reglas (definidas con anterioridad) para llevar a cabo la detección de peticiones HTTP malformadas. Cuando el WAF detecta un ataque, intento de intrusión o fuga de información, entonces bloquea el tráfico web descartando la petición o respuesta HTTP evitando que los ataques afecten a la aplicación web o que información sensible sea enviada como respuesta a potenciales usuarios maliciosos.

De no detectarse peticiones web maliciosas o alguna anomalía, entonces las peticiones y respuestas HTTP fluyen con normalidad. Todo el proceso de análisis y protección ocurre de forma transparente para los usuarios, evitando así, interferir con las operaciones normales de las aplicaciones web.

Modos de implementación: El modo de implementación depende de la topología de red y de las necesidades de seguridad que se requieran para las aplicaciones web. A continuación se listan los modos de implementación más usados para un WAF.

WAF en modo Puente Transparente (Bridge): Se interconectan dos segmentos de red de forma transparente (sus interfaces de red no tienen dirección IP), de modo que no se requiere alterar la configuración de direcciones IP de los servidores web, ya que son estos mismos los que responden las peticiones web. No requiere de la reconfiguración de los registros DNS y permite proteger múltiples servidores de aplicaciones web, siempre y cuando estos se accedan mediante el canal que protege el WAF.

WAF en modo Proxy Inverso: Es un equipo que interconecta dos o más segmentos de red, pero éste si cuenta con dirección IP propia. Concentra, gestiona y analiza las peticiones y respuestas HTTP que circulan entre los usuarios y aplicaciones web. El WAF en modo de proxy inverso responde las peticiones web como si éste fuera el servidor web mismo, por lo tanto es de utilidad para ocultar a los servidores de aplicaciones web de la red exterior. Permite proteger múltiples servidores de aplicaciones web. Su implementación requiere modificar los registros DNS que ahora deben dirigirse a la dirección IP del WAF en modo proxy inverso en vez de a los servidores web.

WAF en modo embebido o plugin: El WAF se instala como un software de complemento o plugin en el servidor web a proteger. Para su operación hace uso de los recursos de hardware (procesador, RAM, disco duro) y software del servidor donde se ha instalado. Su instalación depende totalmente del tipo de servidor web y del sistema operativo subyacente. Afortunadamente existen WAFs para los entornos Windows, GNU/Linux y Unix, así como para los distintos servidores web más populares. Este modo de operación es el más sencillo pues no requiere configuraciones adicionales en la red.

Algunos WAF soportan características como:

Normalización del tráfico web: Usualmente los usuarios maliciosos usan técnicas para ocultar sus ataques web mediante codificación o cifrado. El WAF debe ser capaz de decodificar o descifrar el tráfico web para poder aplicar sus reglas de seguridad.

Aceleración SSL: Algunos WAF comerciales cuentan con hardware especializado para poder atender las peticiones web seguras (HTTPS) de forma rápida, pues el uso de cifrado en las transacciones web implica el uso adicional de procesador y memoria RAM de los servidores web. Emplear aceleradores SSL permite quitar carga de procesamiento a los servidores.

Tipos de solución WAF: En el mercado existen varias opciones de WAF a elegir que nos pueden ayudar a aumentar la seguridad sobre nuestros servidores de aplicaciones web de forma considerable. Entre las dos opciones open source más populares, encontramos las siguientes:

*ModSecurity:(*Desarrollado por Trustwave) ModSecurity funciona como un complemento que se instala en el servidor web. Actualmente soporta los servidores web Apache HTTPD, Microsoft IIS y NGinx. Provee protección contra las principales

amenazas del Top 10 de OWASP mediante su conjunto de reglas especializadas en detección y bloqueo de ataques. Es un proyecto con madurez de desarrollo y cuenta con una creciente comunidad de usuarios que lo han implementado.

*IronBee: (*Desarrollado por: Qualys) IronBee es un WAF desarrollado y mantenido por el equipo que diseñó y desarrolló a ModSecurity en sus inicios. Este proyecto apunta a producir un WAF que sea aún más seguro, de alto rendimiento, portable y libremente disponible, incluso para el uso comercial. El enfoque de este WAF va dirigido a perfilar el comportamiento de la aplicación web y sus usuarios, de esta forma se pueden establecer controles de seguridad basados en la forma de uso de las aplicaciones web, así como los convencionales contra ataques web comunes.

WAFs en el código de las aplicaciones web: Dentro de la amplia gama de herramientas de seguridad para aplicaciones web, existen otro tipo de opciones que están muy ligadas con las acciones que realiza un WAF. Estas herramientas también pueden ser de utilidad cuando se quiere proporcionar seguridad adicional a las aplicaciones web a nivel de código.

Este conjunto de herramientas se implementan directamente en el código de la aplicación, para hacer uso de ellas, se debe tener acceso directo al código fuente de la aplicación web, lo que implica conocer el lenguaje de programación en que se desarrolló y, con base en ello, se pueda comenzar con la modificación del código y acoplarlo según las necesidades.

En muchos casos no se tiene acceso al código fuente de la aplicación web debido a que la aplicación misma podría ser solo un archivo ejecutable. Es ahí cuando el uso de este tipo de herramientas se tiene que descartar. Si este no es el caso, se puede hacer uso de herramientas como las siguientes:

ESAPI WAF (Enterprise Security API Web Application Firewall): Son bibliotecas que se incluyen directamente en el código y, una vez que se implementan en la aplicación web, proporcionarán seguridad de una forma más directa mediante la validación de los datos de entrada, permitiendo así filtrar ataques web. El equipo de desarrollo debe implementar las bibliotecas de esta API sobre el código de las aplicaciones web que se deseen asegurar.

PHPIDS Web Application Security 2.0 (PHP-Intrusion Detection System): Es un software utilizado para reforzar la seguridad en la capa de aplicación a nivel web (directamente en la aplicación). Al igual que los WAF convencionales, se basa en el análisis de transacciones HTTP mediante un conjunto de reglas que se encargan de filtrar anomalías y ataques web. Su modo de operación se lleva a cabo mediante puntajes, donde a un ataque o anomalía se le asigna un puntaje numérico mediante el cual se decide qué tipo de acción defensiva debe seguirse.

Existen varias alternativas que ayudan a complementar la seguridad de aplicaciones web, para hacer uso de ellas. Se debe indagar sobre las que más interesen, ya que solo de esta manera se podrán descartar algunas o bien, acoplarlas según las necesidades.

La seguridad de un sistema es tan fuerte como lo sea su eslabón más débil. Hoy en día la seguridad web es un punto crítico que no puede echarse en saco roto, pues las afectaciones por ataques exitosos, intrusiones y fugas de información sensitiva, pueden traer repercusiones sociales (pérdida de reputación), monetarias e incluso legales. A veces, una simple entrada de datos no asegurada puede acarrearnos grandes problemas.

Los WAF son herramientas que han evolucionado y han demostrado su efectividad a tal grado que dentro de poco tiempo serán considerados como una capa de seguridad obligatoria en cualquier entorno de aplicaciones web seguro.

Implementación de un WAF propio: Las guías de instalación contemplan la implementación del WAF ModSecurity en un servidor web apache en modo embebido. Se debe tener en cuenta que la instalación de un WAF puede resultar sencilla, pero el proceso de aprendizaje para su administración y afinación de políticas requiere de entendimiento en su forma de operación y en las configuraciones específicas [3].

Una regla importante para todos los WAF en general, es que siempre se deben de implementar en modo de solo monitoreo por un periodo de tiempo. Esto con la finalidad de que se observen las alertas que se muestran al analizar el tráfico web dirigido a la aplicación web y así se pueda determinar si hay reglas del WAF que puedan interferir con el funcionamiento de la misma. Tenga en cuenta que como otras tecnologías de monitoreo es susceptible a falsos positivos.

Modelo de Seguridad: existen dos modelos de seguridad en los WAF:

Modelo de Seguridad Positiva: Los WAF que siguen el modelo de seguridad positiva deniegan por defecto todas las transacciones y solamente acepta las que identifica como seguras o válidas. Para determinar si una transacción es segura, consulta una serie de reglas que se definen previamente, ya sea, por el auto-aprendizaje de la aplicación o configuradas manualmente.

A simple vista, parece ser la solución idónea, pero si nuestra aplicación está sujeta a cambios en el diseño o funcionamiento este modelo de seguridad puede volverse difícil de mantener. Otras ventajas de este modelo es que no dependen de ningún tipo de actualizaciones y nos protegen de ataques desconocidos, como desventaja, son más propensos a detectar falsos positivos y necesitan un proceso de aprendizaje, para saber cómo funciona la aplicación.

Modelo de Seguridad Negativa: En este modelo de seguridad, el WAF acepta todas las transacciones y solamente deniega las que detecta como una posible amenaza o un ataque. Contrastando con el anterior, no es muy preciso y depende de actualizaciones y bases de firmas de posibles ataques.

A pesar de las desventajas arriba citadas, los WAF que siguen este modelo no precisan de muchos ajustes y suelen ser fáciles de administrar.

Ataques que puede bloquear: Este aspecto puede variar dependiendo del fabricante del equipo, aunque, la mayoría detectan los ataques más comunes.

Puede detectar un posible buffer overflow analizando las variables que lleguen por GET o POST. Por ejemplo, si el valor de una variable es superior a 150 caracteres, el WAF podría detectar la transacción como maliciosa y denegarla.

Para los ataques de Cross Site Scripting y SQL Injection, el WAF vigila que los valores pasados tanto por GET como por POST, no contengan valores como "SELECT FROM, UNION, CONCAT, <script >, %, etc.

El factor negativo, es que, si nuestra aplicación utiliza caracteres que el WAF tenga en su "lista negra", denegará las peticiones de los usuarios, en este caso hay dos soluciones: rediseñar el funcionamiento de la aplicación o configurar el WAF para que ignore dichos caracteres.

Algunos WAF también monitorizan las respuestas del servidor, por ejemplo, si en una respuesta, que el servidor web envía al usuario se detectan cadenas que pueden ser identificadas como cuentas bancarias, el WAF lo puede detectar como un posible ataque y denegar la respuesta.

Proceso de Aprendizaje: Supongamos que tenemos una aplicación que muestra los productos de un catálogo y para mostrar un producto, se realiza la siguiente petición:

GET http://sitepath.com/show_article.php?id=15

La aplicación de por sí sola, creará una regla que especifique que el valor de esa variable es numérica. Si se realiza la siguiente petición:

GET http://sitepath.com/show_article.php?id=15' or 1=1 --

El WAF detectará una anomalía y tomará las medidas necesarias, que suelen ser, la denegación de la petición o el redireccionamiento a una página previamente configurada, como un error 404.

Riesgos que implica emplear un WAF: Si no está configurados correctamente, pueden detectar muchos falsos positivos, por tanto, muchas transacciones denegadas y pérdida de capital por parte de la empresa.

La mayoría necesitan adaptación y configuración ante nuevos cambios en el funcionamiento de la aplicación.

Pueden introducir un cierto retardo en las transferencias, por tanto, clientes insatisfechos y jefes molestos. Para atenuar este factor negativo, se pueden implementar aceleradores SSL mencionados anteriormente [4].

IV. TOP 10 DE OWASP AÑO 2013

El top 10 de OWASP sobre vulnerabilidades en aplicaciones web se basa en información sobre riegos provenientes de 8 firmas especializadas en seguridad de aplicaciones. Se tienen en cuenta alrededor de 500,000 vulnerabilidades alrededor de cientos de organizaciones y miles de aplicaciones. Asimismo, estas vulnerabilidades son priorizadas de acuerdo al nivel de explotación, detección e impacto estimado.

De acuerdo al ranking, el top 10 de este año quedó de la siguiente manera:

A1 Injection: Corresponde a las inyección de código, siendo las inyecciones SQL una de las más comunes.

A2 Broken Authentication and Session Management: Corresponde al mal manejo de las sesiones en aquellas aplicaciones que utilizan autenticación.

A3 Cross-Site Scripting (XSS): Ocurre cuando existe validación pobre de la información ingresada por el atacante.

A4 Insecure Direct Object References: Puede derivar en un acceso no autorizado a información crítica debido a errores en el diseño o desarrollo.

A5 Security Misconfiguration: Corresponde a configuraciones no adecuadas que pueden impactar en la seguridad de la propia aplicación.

A6 Sensitive Data Exposure: Se refiere a la protección incorrecta de datos críticos tales como, por ejemplo, números de tarjetas de crédito, contraseñas, entre otros.

A7 Missing Function Level Access Control: Corresponde a la falta de controles desde el servidor, permitiendo a un posible atacante acceder a funciones a las que no debería.

A8 Cross-Site Request Forgery (CSRF): Permite a un atacante generar peticiones sobre una aplicación vulnerable a partir de la sesión de la víctima.

A9 Using Known Vulnerable Components: Corresponde a la explotación de librerías, frameworks y otros componentes vulnerables por parte de un atacante con el fin de obtener acceso o combinar con otros ataques.

A10 Unvalidated Redirects and Forwards: Los atacantes aprovechan el uso de redirecciones de sitios web a otros sitios utilizando información no confiable (untrusted) para redirigir a las víctimas a sitios de phishing o que contienen malware [5].

V. CONCLUSIONES

Está comprobado que toda organización debido al entorno de los negocios y a los avances de la tecnología, use el recurso de páginas web como medio de comunicación y actualización de su información, con el uso de Firewall e IPS no se garantiza la seguridad de estos accesos. El WAF se creó como solución exclusiva para poder publicar contenidos web garantizando la confidencialidad, integridad y disponibilidad de los datos generados, transmitidos y mostrados por medio de este recurso.

El WAF trata de un dispositivo físico que analiza el tráfico web entre el servidor web y la WAN, los datos recibidos por parte del usuario son evaluados de acuerdo a los diferentes tipos de ataques, si el dato es anómalo el paquete es rechazado y se niega el servicio, pero, sí el dato es lícito se le entrega al servidor web para ser procesado.

La tendencia a que se creen nuevos ataques por el surgimiento de nuevas vulnerabilidades es evidente, los grandes productores de software siempre están a la expectativa de cualquier evento, razón por la cual se han creado corporaciones de seguridad dedicada a evaluar este tipo de incidentes y con base en ello se crea anualmente el TOP 10 de OWASP. Ellos se basan de esta información para mantener y actualizar los huecos de seguridad que vayan surgiendo para aplicar al WAF.

Históricamente se logra ver como los huecos de seguridad han ido evolucionando a tal escala que algunas entidades como la PCI Security Standards Council exige como medio de seguridad y certificación para la administración de tarjetas de crédito el uso de WAF en sus corporaciones.

Existen diversas formas de configuración de un WAF, se puede desarrollar para cualquier plataforma de sistema operativo, también de acuerdo a su complejidad y rendimiento se puede suministrar en modo appliance o virtualizados, incluso en el mismo código del programador de la página se pueden crear procesos de desarrollo para garantizar el control de acceso a páginas web.

REFERENCIAS

[1] Rapid7.com. (1996. Mar 19). PHF CGI Remote Command Execution. http://www.rapid7.com/db/vulnerabilities/http-cgi-phf-command-execution

[2] Wikipedia.org. (2014. Ago 11). Application firewall. http://en.wikipedia.org/wiki/Application_firewall

[3] Revista Seguridad Universidad Autónoma de México. (2013, May 2). Firewall de Aplicación WEB – parte II. http://revista.seguridad.unam.mx/numero-17/firewall-de-aplicaci%C3%B3n-web-parte-ii

[4] Elhacker.net. Introducción a los Web Applications Firewall (WAF). http://wiki.elhacker.net/seguridad/web/introduccion-a-los-web-application-firewalls-waf

[5] OWASP. (2014, Jul 4). Top 10 2013. https://www.owasp.org/index.php/Top_10_2013