Universidad Europea
Propuesta de Proyecto de Fin de Grado

Reverse Proxy con capacidades de Firewall de aplicación web y aceleración TLS

Alumno:	Pedro Pozuelo Rodríguez
Directora:	Ana del Valle Corrales Paredes
Número de expediente:	217078491
Titulación:	Grado en Ingeniería Informática
Fecha:	30 de diciembre de 2018

Índice general

1 Identificación de la Propuesta

El presente documento recoge la propuesta asociada al Proyecto de Fin de Grado *Reverse Proxy con capacidades de Firewall de aplicación web y aceleración TLS* presentada por *Pedro Pozuelo Rodríguez* cuyo historial de versiones se detalla a continuación:

1.1 Control de versiones del documento

Versión	Fecha	Autor	Descripción
1.0	30/12/2018	Pedro Pozuelo	Primera versión de la propuesta
1.1	29/05/2019	Pedro Pozuelo	Añadida información de viabilidad y profesora asignada

2 Identificación del Proyecto

El título del proyecto propuesto es: *Reverse Proxy con capacidades de Firewall de aplicación web y aceleración TLS*.

El objetivo del proyecto es crear una solución de software libre de firewall de aplicación web perimetral con capacidades de aceleración TLS que permita proteger aplicaciones web con independencia de su arquitectura para lo que actuará como un proxy inverso sobre una tecnología de contenedores de software tipo Docker.

3 Resumen del proyecto

3.1 Versión en español

El objetivo del proyecto es construir una solución de software libre con capacidades de firewall de aplicación web (en adelante WAF, de sus siglas en inglés, *Web Application Firewall*) y aceleración SSL/TLS.

Actualmente la mayoría de los ataques se realizan contra aplicaciones web, con lo que es cada vez más importante contar con una solución que sea capaz de analizar el tráfico web y proteger las aplicaciones.

Por otro lado, en los últimos años existe una tendencia a publicar los servicios web sobre canales cifrados y el tráfico sin cifrar es cada vez menor. Este cambio tiene un impacto en el rendimiento de las plataformas sobre los que se ejecutan los servicios web y añade complejidad. Máxime cuando recientemente se han descubierto múltiples vulnerabilidades en los protocolos SSL/TLS que requieren realizar continuamente cambios y actualizaciones.

Actualmente existen soluciones propietarias que ofrecen funcionalidades WAF y aceleración SSL / TLS, pero son muy costosas y sólo son viables en proyectos con suficiente envergadura y presupuesto, quedando fuera del alcance en aplicaciones web con menos presupuesto o que no generen suficientes beneficios para justificar su inversión.

Por otro lado, existen soluciones WAF de software libre, que tradicionalmente funcionan como módulos adicionales al servidor web, como por ejemplo *modSecurity* como módulo del servidor web *Apache*. Este tipo de soluciones requiere por lo tanto un ejercicio de integración con las aplicaciones web y consumen recursos del servidor que pueden impactar en el rendimiento.

Adicionalmente, para la implementación y configuración adecuada de estos módulos se requiere de una figura con conocimientos de seguridad, y si se despliegan dentro del servicio web, se requiere que la figura responsable de la plataforma web configure unos componentes para los que carece de los conocimientos necesarios y se requiere que asuma el rol de administrador de servicios que desconoce.

Es por ello que se propone una solución que funcione en su propio contenedor o servidor, lo que permitirá desplegarla de manera independiente a la plataforma, con lo que no impactará a los recursos de la arquitectura web, y permitirá una administración basada en roles y que los cambios realizados en uno de los componentes no afecten a otros componentes.

3.2 English version

The goal of the project is to build a free software solution with WAF (Web Application Firewall) capabilities and SSL/TLS acceleration.

Nowadays, most of the attacks are run against web applications, hence it is more important than ever to have a mechanism able to analyze web traffic and protect web applications.

On the other hand, during the past few years it is more common to publish web services over encrypted channels instead of traditional decrypted ones. This trend impacts servers' performance where the web services are running and adds complexity. Especially, since recently several vulnerabilities in the SSL/TLS protocols have been published, requiring configuration changes and applying updates.

There are proprietary solutions that give us the WAF and SSL/TLS acceleration capabilities, but they are costly and they are only affordable for projects with enough magnitude and budget. So it is not worth it to deploy these type of solutions when the web applications don't have enough budget in order to justify the investment.

There are also free software WAFs, which run as a web server module, for instance *modSecurity* is a WAF module for Apache. These solutions require to be integrated as part of the web applications and they consume server resources that can impact on server's performance.

Additionally, the deployment and setup of these modules require security knowledge and, if they are deployed within the web server, it'd mean the person responsible for web administration, who may not have the proper knowledge, would need to set up the security components and would need to assume a role for tasks he/she is not qualified.

For all the reasons previously stated, I propose an autonomous solution running in its own container or server, which will be platform independent and will not impact on the web platform resources. It will also allow an administration based on roles (RBAC) and any configuration change would only affect its own component.

4 Antecedentes y estado actual del tema

Dentro de las soluciones de seguridad tradicionales, nos encontramos los firewall de red perimetrales que permiten proteger los servicios que no se quieren publicar en Internet, pero estas soluciones no permiten analizar o proteger la capa de aplicación de los servicios web que sí se publican, y permiten todo el tráfico dirigido a los servicios web, sea este legítimo o una amenaza de seguridad.

Para proteger estos servicios web, una de las claves es mejorar los patrones de desarrollo incluyendo principios y buenas prácticas de desarrollo seguro, pero estas medidas no son suficientes por diversas causas: se descubren nuevos ataques que no se conocían en el momento de realizar el desarrollo, los equipos de desarrollo no están adecuadamente formados, no existen los controles de validación adecuados como parte del ciclo de desarrollo, existen aplicaciones en producción que no se actualizan cuando se descubre una nueva vulnerabilidad, etc.

Por estos y otros motivos no se puede considerar que las aplicaciones web sean seguras y se deben desplegar controles de seguridad que ayuden a protegerlas frente a los ataques.

Las soluciones WAF nos permiten analizar este tipo de tráfico y proteger las aplicaciones web.

Dentro de las soluciones WAF, existen los siguientes tipos: Soluciones de tipo appliance, soluciones en la nube de tipo Software as a Service (en adelante SaaS) como parte de servicios de Red de distribución de contenidos (en adelante CDN, de sus siglas en inglés, *Content Delivery Network*) y soluciones de tipo software.

Las soluciones de tipo appliance o CDN son propietarias y muy costosas, por lo que sólo son viables en proyectos donde se justifique la inversión.

Las soluciones de tipo software requieren que se contemplen como parte del diseño de la aplicación web, por lo que requieren un ejercicio de integración con el servicio web y consumen recursos del servidor y puede afectar al rendimiento.

En materia de aceleración SSL/TLS, nos encontramos que en los últimos 5 años se ha producido un cambio significativo; el mercado ha apostado por utilizar canales cifrados HTTPS de forma masiva frente a la política previa en la que sólo se cifraban ciertas comunicaciones que se consideraban sensibles.

Paralelamente a este cambio, se han publicado múltiples ataques a los protocolos SSL y TLS que han supuesto que la práctica tradicional de habilitar cifrado por defecto y no cambiarlo ya no sea válida. Actualmente son habituales los cambios de configuración de los *ciphersuites*, certificados y configuración en general de los protocolos. Estos cambios no son triviales y deben realizarse sobre los terminadores del protocolo que están expuestos a Internet.

Es por ello que se propone una solución que pueda desplegarse y configurarse de forma indepen-

diente a la plataforma web, con lo que se asegure que un cambio en el componente de seguridad no afectará a la plataforma web y viceversa.

4.1 Bibliografía

A continuación se destacan las referencias que se han consultado para evaluar la necesidad del proyecto:

- *OWASP Top 10 Most Critical Web Application Security Risks [1]*
- *Modelo de amenazas SSL propuesto por* Qualys *[2]*
- *Majority of the world's top million websites now use HTTPS [3]*
- *HTTPS encryption on the web [4]*
- *ModSecurity [5]*
- *Fabricante de appliances WAF lider del mercado [6]*

Estas y otras referencias se recogen en la sección de *Referencias 12*.

5 Objetivos

5.1 Objetivo global

El objetivo del proyecto es crear una solución de software libre de firewall de aplicación web perimetral con capacidades de aceleración TLS que permita proteger aplicaciones web con independencia de su arquitectura para lo que actuará como un proxy inverso sobre una tecnología de contenedores de software tipo Docker.

5.2 Objetivos concretos

A la hora de abordar el proyecto se han identificado objetivos para el componente WAF y para el componente de TLS tal como se desglosa a continuación:

Objetivos del componente WAF:

WAF-1. La solución debe poder ejecutarse en un sistema operativo o máquina independiente de la plataforma de la aplicación web con el objetivo de garantizar independencia en las tareas de administración y permitir aplicar un modelo RBAC.

WAF-2. La solución debe disponer de un conjunto básico de políticas de auditoría o bloqueo que permitan proteger la aplicación web frente a los ataques más comunes.

WAF-3. La plataforma debe permitir implementar parches virtuales frente a ataques conocidos.

WAF-4. La solución debe permitir la elaboración de reglas personalizadas según las necesidades específicas de la plataforma del cliente.

WAF-5. La plataforma debe ser compatible con el modelo de licencias de *software libre[7]* tipo Licencia Pública General de GNU (en adelánte GPL, de sus siglas en inglés GNU General Public License[8, Licencia GPL]) o Licencia Pública General Reducida de GNU (en adelánte LGPL, de sus siglas en inglés GNU Lesser General Public License[9, Licencia LGPL]).

WAF-6. La plataforma debe generar logs de seguridad exportables a soluciones externas de gestión de información y eventos de seguridad (en adelante SIEM de sus siglas en inglés, Security information and event management).

A continuación se recogen los objetivos del componente de TLS:

TLS-1. La solución debe poder participar en la negociación TLS, presentando certificados confiables a los clientes.

TLS-2. La solución deber poder gestionar los certificados presentados a los clientes incluyendo soporte a la extensión SNI de TLS.

TLS-3. La plataforma debe soportar TLS versión 1.3, HTTP/2 y otros elementos incluidos en las *buenas prácticas de TLS[10]*.

TLS-4. Debe permitir aplicar soluciones de SSL offloading, entre el WAF y los frontales de la plataforma web, o permitir cifrado punto a punto.

5.3 Objetivos no contemplados

Queda fuera del alcance de la solución los siguientes objetivos:

- Auto-escalado o auto-aprovisionamiento de recursos.
- Aprendizaje automático de la estructura de las aplicaciones web.
- Sistema de detección de ataques basado en el número de peticiones o desviaciones estadísticas.
- Sistemas de aprovisionamiento en plataformas distintas a Docker.

Si bien estos objetivos no forman parte de la solución dentro del alcance del Proyecto de Fin de Grado, se utilizarán como referencia para evolucionar la solución posteriormente.

6 Compromisos y requisitos

6.1 Compromisos del cliente o usuario

El cliente debe cumplir con los compromisos impuestos por la licencia de software libre que se escoja finalmente. Si bien no se ha elegido la licencia concreta sobre la que se licenciará la solución, esta pertenecerá a la *familia de licencias de software libre [11]*.

6.2 Requisitos del sistema

La solución se construirá sobre un sistema operativo Debian GNU/Linux, el cual a su vez podrá ser desplegado sobre una plataforma de contenedores de software tipo Docker o instancias de la nube como AWS o Azure.

Se requeriría que las entradas DNS de las aplicaciones web puedan apuntar al servicio WAF o bien se modifique el enrutamiento de red de forma que el WAF éste en un punto de la red externo a la aplicación web.

La solución no implementará gestión de certificados o gestionará la arquitectura web que protege.

7 Plan de trabajo y objetivos del proyecto

7.1 Plan de trabajo

A la hora de planificar la ejecución del proyecto se han identificado los siguientes hitos:

- Análisis de las soluciones WAF actualmente disponibles y elaboración de un análisis del estado del arte actual.
- Definición de los requisitos y de los casos de uso.
- Construcción de un laboratorio en el que se evalúen las soluciones que cumplan los requisitos definidos.
- Construcción prototípica sobre la que se implementarán las funcionalidades elegidas.
- Ejecución de pruebas, detección de errores y afinamiento de la solución.
- Entrega de la solución.
- Elaboración de la presentación de la solución.
- Defensa del Proyecto de Fin de Grado.

La documentación del proyecto se generará a medida que se completen los hitos, por lo que no se definen hitos específicos.

A lo largo de la ejecución del proyecto se mantendrán reuniones regulares con la persona asignada con el fin de dar un seguimiento y evaluación continua de las decisiones tomadas y la evaluación de la solución.

Se seguirán las guías y buenas prácticas promovidas por el *Center for Internet Security* y *OWASP*

7.2 Presupuesto del proyecto

No se contempla que sea necesario un presupuesto económico.

8 Beneficios para el cliente

El protocolo HTTP es usado y atacado masivamente. Las soluciones actuales son o bien de pago y cerradas - tradicionalmente un modelo de appliance o CDN) no protegen adecuadamente las aplicaciones debido a la complejidad de integrarlas como parte de la aplicación web o tienen una alta complejidad de desplegar y mantener.

La solución propuesta es gratuita, software libre y se despliega en su propio contenedor o servidor, lo que permitirá ahorrar costes, adaptarla a las necesidades del cliente y desplegarla de manera independiente a la plataforma web.

9 Experiencia previa en el tema

Desde hace más de 10 años he trabajado en múltiples proyectos en los que se he desplegado y administrado diversas tecnologías propietarias WAF, destacando las soluciones tipo appliances, como por ejemplo Secure Sphere Imperva, y soluciones CDN, como Akamai Kona.

Antes de dedicarme a la seguridad era administrador de sistemas y siempre he sido un entusiasta del software libre y un gran defensor de sus bondades.

Por lo tanto tengo experiencia en las tecnologías WAF privativas tradicionales así como en las plataformas Linux y creo que este proyecto puede cubrir una carencia que actualmente existe.

10 Viabilidad y plan de recursos

10.1 Estudio de viabilidad técnica

Durante la preparación de la presente propuesta se han evaluado las distintas soluciones que actualmente hay disponibles en el mercado, tanto soluciones privativas, como soluciones de software libre.

Las soluciones privativas se caracterizan por una mayor potencia y su elevado coste, lo que las descarta como opción válida en entornos en los que no es posible o rentable realizar esta inversión.

Las soluciones de software libre que se han evaluado por su parte tienen una fuerte dependencia con el software de la plataforma web, por lo que la operación y mantenimiento son complejos. Este es una de las principales causas de su escasa adopción en el mercado.

Tras realizar un análisis no exhaustivo de los distintos componentes involucrados, se considera que se puede implementar una solución atómica que permita su implementación con un bajo impacto y una independencia de la plataforma web. Lo que facilitaría su adopción y mantenimiento.

10.2 Plan de recursos

El proyecto se realizará individualmente.

10.3 Estudio de viabilidad económica

No se contempla que el proyecto requiera una inversión económica.

11 Comentarios

Conozco bien las soluciones privativas del mercado y, si bien tecnológicamente son soluciones potentes, no pueden ser modificadas y carecen de la adaptabilidad del software libre.

Por otro lado, estas soluciones tienen unos precios muy elevados, lo que hace que no sean viables para aplicaciones web que no generan unos beneficios significados; y, por lo tanto, tradicionalmente estas aplicaciones no son protegidas adecuadamente.

Con la solución propuesta el objetivo es crear un WAF con capacidad de aceleración SSL/TLS que permita proteger aplicaciones web sin requerir una inversión significativa y sin añadir complejidad a la arquitectura web existente.

12 Aceptación del proyecto

Pendiente de aceptación.

Acrónimos

GPL GNU General Public License[8, Licencia GPL]. 6

LGPL GNU Lesser General Public License[9, Licencia LGPL]. 6

SIEM Security information and event management. 7

SNI Server Name Indication[12, Wikipedia]. 7

Referencias

[1] *OWASP Top 10 Most Critical Web Application Security Risks:*
. URL: https://www.owasp.org/index.php/Category:OWASP%5C_Top%5C_Ten%5C_Project.

[2] *Modelo de amenazas SSL propuesto por* Qualys*:*
. URL: https://www.ssllabs.com/projects/ssl-threat-model/index.html.

[3] *Majority of the world's top million websites now use HTTPS:*
. URL: https://www.welivesecurity.com/2018/09/03/majority-worlds-top-websites-https/.

[4] *HTTPS encryption on the web:*
. URL: https://transparencyreport.google.com/https/overview?hl=en.

[5] *ModSecurity:*
. URL: http://www.modsecurity.org/.

[6] *Fabricante de appliances WAF lider del mercado:*
. URL: https://www.imperva.com/products/web-application-firewall-waf/.

[7] *¿Qué es el software libre?*
. URL: https://www.gnu.org/philosophy/free-sw.es.html.

[8] *GNU General Public License*
. URL: https://www.gnu.org/licenses/gpl-3.0.html.

[9] *GNU Lesser General Public License*
. URL: https://www.gnu.org/licenses/lgpl-3.0.html.

[10] *SSL and TLS Deployment Best Practices*
. URL: https://github.com/ssllabs/research/wiki/SSL-and-TLS-Deployment-Best-Practices.

[11] *GPL-Compatible Free Software Licenses*
. URL: https://www.gnu.org/licenses/license-list.html#SoftwareLicenses.

[12] Wikipedia. *Server Name Indication*
. URL: https://es.wikipedia.org/wiki/Server_Name_Indication.