ANÁLISIS COMPARATIVO DE UN FIREWALL DE APLICACIONES WEB COMERCIAL Y UN OPEN SOURCE FRENTE AL TOP 10 DE OWASP

ELKIN MAURICIO PIEDRAHITA VILLARRAGA

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA (UNAD)
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA
ESPECIALIZACIÓN EN SEGURIDAD INFORMÁTICA
BOGOTÁ
2016

ANÁLISIS COMPARATIVO DE UN FIREWALL DE APLICACIONES WEB COMERCIAL Y UN OPEN SOURCE FRENTE AL TOP 10 DE OWASP

ELKIN MAURICIO PIEDRAHITA VILLARRAGA

Trabajo de grado para optar el título de
ESPECIALISTA EN SEGURIDAD INFORMÁTICA

Director
Alexander Larrahondo Núñez

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA (UNAD)
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA
ESPECIALIZACIÓN EN SEGURIDAD INFORMÁTICA
BOGOTÁ
2016

Nota de aceptación

_____

_____

_____

_____

Firma del Presidente del Jurado

_____

Firma del Jurado

_____

Firma del Jurado

Bogotá julio de 2016

CONTENIDO

Pág.

INTRODUCCION 18

1. DEFINICION DEL PROBLEMA 19

1.1. FORMULACIÓN DEL PROBLEMA 20

2. JUSTIFICACION 21

3. OBJETIVOS 22

3.1. OBJETIVO GENERAL 22

3.2. OBJETIVOS ESPECÍFICOS 22

4. MARCO REFERENCIAL 23

4.1. ESTADO DEL ARTE 23

4.1.1. Cuadrante mágico de Gartner para WAF 2015 23

4.1.2. Antecedentes 31

4.2. MARCO TEÓRICO 34

4.2.1. Aspectos generales del WAF 34

4.2.2. Funcionamiento de una aplicación web 36

4.2.3. Vulnerabilidades y Vectores de ataques en aplicaciones web 37

4.2.4. Análisis del TOP 10 de OWASP frente al WAF 38

4.3. MARCO CONCEPTUAL 42

4.4. MARCO LEGAL 44

4.4.1. Ley De Delitos Informáticos En Colombia 44

5. DISEÑO METODOLÓGICO 48

5.1. TIPO DE ESTUDIO Y DISEÑO DE CONTRASTACIÓN DE HIPÓTESIS 48

5.2. POBLACIÓN, MUESTRA Y MUESTREO 48

5.3. MÉTODOS, TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS 48

5.4. PLAN DE PROCESAMIENTO Y ANÁLISIS A DATOS 49

5.5. METODOLOGÍA DE DESARROLLO 50

6. DISEÑO E IMPLEMENTACIÓN 51

6.1. PORTAL WEB DE PRUEBAS 51

6.2. FICHA TÉCNICA DE LOS FIREWALL DE APLICACIONE WEB 52

6.2.1. Información ModSecurity 52

6.2.2. Información F5 BIG-IP 53

6.3. ESQUEMA DE LA RED DE PRUEBAS 54

6.3.1. Configuración WAF ModSecurity en proxy reverso 54

6.3.2. Configuración WAF F5 en proxy reverso 57

7. ANALISIS Y RESULTADOS 62

7.1. PRUEBA OWASP A1: INYECCIÓN 62

7.2. PRUEBA OWASP A2: PERDIDA DE AUTENTICACIÓN 64

7.3. PRUEBA OWASP A3: XSS 67

7.4. PRUEBA OWASP A4: REFERENCIA DIRECTA INSEGURA A OBJETOS 69

7.5. PRUEBA OWASP A5: SEGURIDAD INCORRECTA 73

7.6. PRUEBA OWASP A6: EXPOSICION DE DATOS 77

7.7. PRUEBA OWASP A7: INEXISTENTE CONTROL DE ACCESO 79

7.8. PRUEBA OWASP A8: CSRF 83

7.9. PRUEBA OWASP A9: VULNERABILDADES CONOCIDAS 85

7.10. PRUEBA OWASP A10: REDIRECCIONES Y REENVÍOS 86

7.11. CUADRO COMPARATIVO DE LAS PRUEBAS EJECUTADAS 89

8. CONCLUSIONES 92

9. RESULTADOS E IMPACTOS DEL PROYECTO 93

10. DIVULGACIÓN DEL PROYECTO 94

BIBLIOGRAFIA 95

ANEXOS 96

LISTA DE TABLAS

Pág.

Tabla 1 – OWASP aplicado a un Firewall de aplicaciones Web 33

Tabla 2- Tabla comparativa de resultados 89

LISTA DE FIGURAS

Pág.

Figura 1- Cuadrante mágico de Gartner 2015 para WAF 24

Figura 2- Estructura de una aplicación web 36

Figura 3- Propiedades de seguridad en aplicaciones web 38

Figura 4- Portal de pruebas Mutillidae 51

Figura 5- Portal de pruebas DVWA 52

Figura 6- Esquema de red WAF 54

Figura 7- Configuración proxy reverso Apache 55

Figura 8- Fichero principal de ModSecurity 56

Figura 9- Archivo de configuración ModSecurity 56

Figura 10- Revisión logs ModSecurity 57

Figura 11- Configuración self IP F5 57

Figura 12- Configuración del Virtual Server en F5 58

Figura 13- Configuración del pool en F5 59

Figura 14- Configuración política de seguridad WAF F5 paso 1 59

Figura 15- Configuración política de seguridad WAF F5 paso 2 60

Figura 16- Configuración política de seguridad WAF F5 paso 3 61

Figura 17- Prueba A1 Inyección de SQL 62

Figura 18- Log WAF F5 Inyección SQL 63

Figura 19- Log WAF Modsecurity inyección SQL 63

Figura 20- Proxy con Burp Suite 64

Figura 21- Pruebas de autenticación de fuerza bruta 65
Figura 22- Captura de contraseña en ataque de fuerza bruta 65
Figura 23- Detección ataque fuerza bruta WAF F5 66
Figura 24- Detección ataque fuerza bruta WAF Modsecurity 66
Figura 25- Pruebas A3 Secuencia de comandos en sitios cruzados (XSS) 67
Figura 26- Resultado A3 Secuencia de comandos en sitios cruzados (XSS) 67
Figura 27- Log WAF F5 Secuencia de comandos en sitios cruzados (XSS) 68
Figura 28- Log WAF ModSecurity Secuencia de comandos en sitios cruzados (XSS) 69
Figura 29- A4 Referencia directa insegura a objetos en el portal mutillidae 69
Figura 30- Captura de la visualización de archivo de texto 70
Figura 31- Acceso a otros ficheros del portal mutillidae mediante Burp Suite 71
Figura 32- Fichero accedido mediante referencia insegura 71
Figura 33- Detección de referencia directa insegura a objetos en el WAF F5 72
Figura 34- Detección de referencia directa insegura a objetos en el WAF ModSecurity 72
Figura 35- A5 configuración de seguridad incorrecta 73
Figura 36- Identificación del formato URL del portal mutillidae 73
Figura 37- Captura del formato en el portal de login 74
Figura 38- Listado de páginas a explorar 74
Figura 39- Acceso a la página de configuración del servidor PHP 75
Figura 40- Detección de configuración de seguridad incorrecta WAF F5 76

Figura 41- Detección de configuración de seguridad incorrecta WAF ModSecurity 76
Figura 42- A6 Fingerprint con WAF marca F5 77
Figura 43- Identificación fingerprint en WAF F5 78
Figura 44- A6 Fingerprint con WAF marca ModSecurity 78
Figura 45- Identificación fingerprint en WAF ModSecurity 79
Figura 46- A7 Inexistente control de acceso 80
Figura 47- Identificación del directorio consultado 80
Figura 48- Bloqueo de URL en el WAF F5 81
Figura 49- Acceso al portal de administración con WAF f5 en bloqueo 81
Figura 50- Bloqueo de URL en el WAF ModSecurity 82
Figura 51- Acceso al portal de administración con WAF ModSecurity en bloqueo 82
Figura 52- A8 Falsificación de peticiones en sitios cruzados 83
Figura 53- Cambio de contraseña a través de CSRF 83
Figura 54- Detección CSRF en el WAF F5 84
Figura 55- Detección CSRF en el WAF ModSecurity 84
Figura 56- A9 Uso de componentes con vulnerabilidades conocidas 85
Figura 57- Detección de Shellshock en el WAF F5 86
Figura 58- Detección de Shellshock en el WAF ModSecurity 86
Figura 59- A10 Redirecciones y reenvíos no válidos 87
Figura 60- Identificación de la redirección 87
Figura 61- Redirección en la URL 88
Figura 62- Análisis de la redirección en el WAF F5 88

Figura 63- Análisis de la redirección en el WAF ModSecurity 89

Figura 64- Firmas Activadas por el WAF 90

Figura 65- Efectividad del WAF en la detección de ataques 91

LISTA DE ANEXOS

Pág.

ANEXO A. RESUMEN ANÁLITICO RAE 96

GLOSARIO

Abuso de Funcionalidad: Una técnica de ataque que usa las características y funcionalidades de un sitio web para con generar un alto consumo de recursos, realizar fraudes o eludir los controles de acceso.

Aplicación Web: Una aplicación de software, ejecutada por un servidor web, la cual responde a solicitudes dinámicas sobre la página web a través de HTTP.

Autenticación: El proceso de verificar la identidad o ubicación de un usuario, servicio o aplicación. La autenticación es desempeñada usando al menos uno de estos tres mecanismos: algo que usted tiene, algo que usted sabe o algo que usted es. La aplicación de autenticación podría proporcionar diferentes servicios basado en la ubicación, método de acceso, hora del día, entre otros.

Autenticación Insuficiente: Ocurre cuando un sitio web permite a un atacante accede contenido sensible o funcionalidades sin verificar su identidad.

Autorización: La decisión de que recursos un usuario, servicio o aplicación tiene permiso para acceder. Los recursos accesibles pueden ser URL's, archivos, directorios, bases de datos, rutas de ejecución, entre otros.

Cookie: Una pequeña cantidad de información enviada por el servidor web hacia el cliente, la cual puede ser almacenada o recuperada un tiempo después.

Denegación de Servicio: (DoS, en inglés) Es una técnica de ataque que consume todos los recursos disponibles de un sitio web al intentar de forma legítima establecer conexiones concurrentes en lapso corto de tiempo y en una cantidad que llegan a ser imposibles de procesar por el sistema.

Desbordamiento de búfer: Una técnica de explotación que altera el flujo de una aplicación al sobreescribir partes de memoria. Los desbordamientos de búfer son una causa común del mal funcionamiento del software.

Directorio Transversal: Una técnica usada para explotar sitios web al acceder a archivos y comandos más allá del directorio raíz del documento. La mayoría de sitios web restringen el acceso a usuario para una porciòn específica de los archivos del sistema, típicamente llamados el directorio raíz o CGI.

Divulgación de información: Cuando un sitio web revela información sensible, como son comentarios del desarrollador o errores de mensajes, lo cual ayuda a un atacante a explotar un sistema.

Fijación de sesión: Es una técnica de ataque que obliga a sesión de usuario o una sesión ID a utilizar un valor específico.

Firewall de Aplicación Web: Es un dispositivo intermedio, situado entre un cliente web y un servidor web que analiza mensajes sobre la capa 7 del modelo OSI identificando violaciones en las políticas de seguridad programas.

Fuerza Bruta: Es proceso automatización basada en la prueba y error para adivinar los secretos protegidos de un sistemas. Ejemplos de los secretos incluye: usuarios, contraseñas o claves criptográficas.

HyperText Transfer Protocol: (HTTP, en inglés) Un protocolo de esquema usado en la World Wide Web. HTTP describe la manera de como cliente web solicitud información y como un servidor web responde esa peticiones.

ID de Sesión: Es una cadena de datos proporcionados por un servidor web, normalmente almacenados dentro de una cookie o URL. Una sesión registra una valor de un usuario, o tal vez solo su sesión actual, y como él ha navegado a través del sitio web.

Inyección de Metacaracteres: Es una técnica de ataque usada para explotar sitios web al enviar metacaracteres, los cuales tienen un significado especial en los datos de entrada de una aplicación web.

Inyección de SQL: Es una técnica de ataque usada para explotar sitios web al alterar información de la base de datos SQL a través de la manipulación de las entradas de la aplicación.

Inyección Nula: Es una técnica de explotación usada para saltar los filtros de revisión de sanidad al adicionar caracteres codificados en la URL con el byte de Null a los datos suministrados por el usuario. Cuando los desarrolladores crean una aplicación web en una variedad de lenguajes de programación, esas aplicaciones web a menudo pasan los datos que subyacen a bajo nivel para procesamiento y funcionalidad adicional. Si un usuario suministra cadenas que contienen un carácter de null (\0), la aplicación web podría para de procesar la cadena hasta el punto del null.

Navegador Web: Es un programa que se utiliza para mostrar mensajes HTML enviados por un servidor web.

Manipulación de Cookie: Alteración o modificación de los valores de las cookies sobre el navegador del cliente, con el fin de explotar fallas de seguridad dentro de la aplicación web.

Manipulación Parámetros: Se le denomina a la alteración o modificación de los parámetros de usuario u otros valores dentro de la URL.

Manipulación de URL: Alterar o modificar los parámetros de una aplicación web a través de la información suministrada en la URL.

Manipulación del User-Agent: Es una técnica usada para salta restricciones de requerimiento que se tienen respecto al navegador al alterar el valor enviado dentro de los encabezados de HTTP.

Secuencia de Órdenes en Sitios Cruzados: (XSS, en inglés) Es una técnica de ataque que obliga a un sitio web a replicar la información suministrada por el cliente, la cual es ejecutada desde el navegador del usuario.

Servicio Web: Es una aplicación de software que usa mensajes en formato XML para comunicar sobre HTTP. Generalmente, este tipo de aplicación interactua con servicio web en lugar de usuarios convencionales.

Servidor Web: El propósito general del este dispositivo es manejar y responder las solicitudes HTTP.

Transport Layer Security: (TLS, en inglés) Es el successor más seguro de SSL. El protocolo TLS proporciona la privacidad de las comunicaciones sobre internet. El protocolo permite a la aplicación cliente/servidor comunicarse de forma adecuada previniendo espionaje, manipulación o falsificación de los mensajes.

Ubicación de Archivo Predecible: Es una técnica usada para acceder a un sitio web con contenido oculto o funcionalidades predecibles, ya sea por sus nombres o ubicación de sus archivos.

Universal Resource Locator: (URL, en inglés) Es una manera de especificar la ubicación de un objeto, normalmente una página web en internet.

RESUMEN

Desarrollar una aplicación web podría conllevar un gran número de riesgos informáticos inherentes, puesto que existen diferentes tipos de técnicas que han sido utilizadas para tomar provecho de este tipo de aplicaciones algunas de las más conocidas se pueden encontrar en el top OWASP 10, sin embargo día a día nuevas vulnerabilidades son encontradas y la posibilidad que un riesgo se materialicé es cada vez mayor. Por esta razón, las empresas que hacen uso de este tipo de aplicaciones deben tomar conciencia de las vulnerabilidades a las que pueden estar expuestos y establecer algún tipo de control que permita disminuir los riesgos.

Los controles que se pueden llevar a cabo en una aplicación web son dos, el primero es realizar una auditoria periódica donde se hace una revisión del código y se ejecutan pruebas de sombrero blanco con el fin de encontrar algún tipo de vulnerabilidad, la segunda es implementar un firewall de aplicación web. Cada control ofrece sus ventajas y desventajas, y en el mejor de los casos lo ideal sería disponer de ambos, si bien la auditoria permitiría generar código más robusto habría una brecha de seguridad en el lapso que una nueva vulnerabilidad sea encontrada hasta el momento en que la auditoria sea realizada. Así que disponer de un WAF que esté en todo momento revisando las peticiones de los usuarios podría incrementar el nivel de seguridad.

Ahora la pregunta sería, ¿Qué nivel de seguridad y confiabilidad ofrece un WAF?, históricamente los WAF empezaron a ganar popularidad luego que en el Consejo de Estándares de Seguridad (PCI-DSS) exigieran a las entidades emisoras de tarjetas de crédito realizar controles sobre las aplicaciones web bien sea por revisión del código o mediante un WAF. Hoy en día existen diferentes fabricantes dedicados al desarrollo de WAF, analizando lenguajes de programación tales como HTML, HTTPS, SOAP and XML-RPC, previniendo ataques como XSS, inyecciones de SQL, secuestro de sesión, desbordamiento de buffer, ataques de día cero, entre otros. Así que con base en la anterior los WAF hoy en día han tenido un gran desarrollo y están diseñados para detectar diferentes vectores de ataques, lo que brinda un buen nivel de seguridad.

Teniendo en cuenta que existen tantas marcas de WAF así como beneficios a nivel seguridad, este proyecto se enfocó en evaluar las diferencias que podrían existir entre un WAF comercial frente a uno de libre de distribución, esta comparación se hizo con base en el Top 10 de OWASP donde cada una de las vulnerabilidades fue probada en cada uno de los WAF. La implementación del esquema de pruebas requirió que el WAF operará en un modo de proxy reverso

de esta forma el servidor web no sufrió ningún tipo de alteración durante el desarrollo del proyecto y así garantizar unas pruebas ecuánimes.

Los resultados obtenidos de la prueba han permitido evidenciar que el WAF de marca F5 dispone una plantilla de gran cantidad lenguajes de programación web que permiten implementar reglas tan granulares tanto como se especifiquen por el administrador, además dispone un consola de administración web amigable que permite identificar de forma fácil el ataque o información anómala detectada, también se observa que la herramienta dispone de un esquema de aprendizaje el cual permite notificar al WAF eventos como falsos positivos y aceptar parámetros que en un principio fueron marcados como anómalos, lo cual se asemeja a un modelo de seguridad positivo que permite tener una mayor escalabilidad. Por su parte Modsecurity ofrece una gran versatilidad para el desarrollo de nuevas firmas de ataques, su consola de administración es a través de línea de comando, y el nivel de seguridad que se ofrece es muy similar al proporcionado por el WAF de marca F5 con base en las pruebas realizadas en este proyecto. Así que los niveles de seguridad brindados por el WAF comercial como el de libre distribución están iguales, sin embargo las diferencias radican en las funcionalidades que ofrece el WAF comercial que permite una mayor escalabilidad y un mejor modelo de implementación.

Palabras claves: Vulnerabilidades Web, Firewall de Aplicación Web, OWASP

INTRODUCCION

El firewall en el ámbito de la seguridad informática ha permitido establecer un punto de monitoreo y control en el acceso a la red que brindan las compañías a sus usuarios y/o empleados para el caso de los firewall de aplicación web el punto de supervisión está a nivel de aplicación validando las consultas que realiza el usuario así como las respuestas que entrega el servidor, de esta forma el WAF garantiza la integridad de la información que es intercambiada entre las partes.

En el mercado se pueden encontrar diferentes marcas y funcionalidades en servicios de WAF, pero antes de realizar una selección siempre es importante identificar las necesidades y requerimientos de cada aplicación. Para este proyecto se evalúa el cumplimiento que da un WAF, de una marca comercial comparada con uno de libre distribución, respecto al cumplimiento del Top 10 de vulnerabilidades de OWASP en 2013. Los elementos empleados en esta evaluación son: Application Security Manager como es conocido el producto WAF de F5, una de las marcas líderes en Gartner para 2015 y por otra parte se empleó ModSecurity un proyecto de libre distribución de la empresa Trustwave, un software conocido por ser uno de los antiguos y usado en proyectos de investigación.

Posteriormente, las dos marcas de WAF serán implementadas en un ambiente de pruebas, donde se tendrá un servidor web con fallas seguridad y del lado cliente se dispondrá de herramientas de pentesting para explotar las vulnerabilidades listadas en el Top 10 de OWASP. Como resultado se espera poder diferenciar las ventajas y desventajas que presenta cada una de las marcas.

1. DEFINICION DEL PROBLEMA

Con el auge de la era digital la mayoría de actividades cotidianas se pueden hacer en línea, solamente se requiere disponer de una conexión a internet, un equipo terminal como un PC o un teléfono inteligente y un navegador web, las ventajas para los usuarios son muchas, además de tener disponibilidad del servicio en todo momento sin restricción de horarios, ahorrando tiempo y dinero. Hoy en día existen diferentes servicios como son: banca en línea, comercio electrónico, trámites gubernamentales, educación, entre otros. Sin embargo, prestar estos servicios representa un reto en seguridad para las entidades que lo brindan puesto que deben garantizar la confiabilidad e integridad de la información que es intercambiada.

En Colombia, existen entidades encargadas de vigilar y hacer cumplir los estándares a nivel de seguridad y así proteger los datos de los usuarios, una de ellas es la Superintendencia Financiera de Colombia que a su vez se apoya en normativas a nivel internacional como son PCI-DSS, OWASP, entre otros. Este último, el Proyecto Abierto de Seguridad de Aplicaciones Web (OWASP por sus siglas en inglés) tiene el propósito de publicar e informar aquellas causas que hacen que las aplicaciones web sean inseguras, categorizándolas por criticidad e impacto que dependiendo los servicios prestados por la compañía llega a tener aplicabilidad.

Por su parte, dentro de la seguridad informática están los Firewall de Aplicaciones WEB (WAF por sus siglas en inglés) que surgen por la necesidad de mejorar la seguridad en los servicios web. Los WAF a diferencia de otros dispositivos tradicionales de seguridad como: firewall, Sistemas de Prevención o Detección de Intrusos (IPS/IDS por sus siglas en inglés) o antivirus; permiten un análisis más exhaustivo del contenido web, usual tipo de tráfico empleado en las operaciones en línea, ya que inspecciona: parámetros HTML, cabeceras de las peticiones, cookies, XML, javascript, entre otros, lo cual permite prevenir o detectar ataques como CrossSite Scripting (XSS), SQL Injection (SQL), Remote y Local File Inclusion (LFI), entre otras actividades que realizan los hackers para robar la información de los usuarios o violar la seguridad de los sistemas.

En definitiva, WAF y OWASP comparte un mismo propósito ofrecer estrategias para proteger las aplicaciones web en línea que en las empresas prestadoras del servicio se transforma en la protección de la información de los usuarios, uno de los bienes más importantes de este tipo de compañías.

1.1. FORMULACIÓN DEL PROBLEMA

¿Qué diferencia existe a nivel de seguridad y funcional entre un Firewall de aplicación WEB comercial y un Open Source frente al Top 10 de OWASP?

2. JUSTIFICACION

El Proyecto Abierto de Seguridad en Aplicaciones WEB (OWASP por sus siglas en inglés) es una comunidad abierta dedicada a facultar a las organizaciones a desarrollar, adquirir y mantener aplicaciones que pueden ser confiables[1]. En OWASP se desarrolló el proyecto TOP 10 el cual identifica los riesgos más críticos que enfrentan los desarrollos Web, estos riesgos son identificados por reconocidas empresas en el entorno de la seguridad informática tales como: MITRE o PCI DSS. Por su parte el WAF fue un dispositivo diseñado para proteger los sitios web de ataques que los tradicionales dispositivos de seguridad como son: los firewall, IDS/IPS o Antivirus no pueden prevenir.

Con base en lo anterior tanto el WAF como OWASP tienen como propósito de aumentar la seguridad de los sitios web, por lo que en este proyecto se busca analizar la relación que existe entre las recomendaciones dadas por el TOP 10 de OWASP y los mecanismos de seguridad que ofrece el WAF, de esta forma en primera instancia se puede identificarse los riesgos presentados en OWASP y cómo pueden mitigados por el WAF y en segunda medida se busca comparar las diferencias presentes entre un firewall comercial y uno de libre distribución.

[1] OWASP. OWASP Top 10 – 2013: Los diez riesgos más críticos en aplicaciones web [en linea].<https://www.owasp.org/images/5/5f/OWASP_Top_10_-_2013_Final_-_Espa%C3%B1ol.pdf> [citado en 16 de noviembre de 2015].

3. OBJETIVOS

3.1. OBJETIVO GENERAL

Realizar un análisis comparativo de los niveles de seguridad que brinda un firewall de aplicación web comercial y un Open Source frente al Top 10 de vulnerabilidades de OWASP.

3.2. OBJETIVOS ESPECÍFICOS

- Evaluar el estado de arte de los web aplications firewalls comerciales frente a los Open Source.

- Diseñar un esquema de pruebas en un ambiente virtual integrando un firewall de aplicación web, una aplicación web y una herramienta de pentester.

- Ejecutar pruebas de vulnerabilidades a una aplicación protegida por Firewall's de Aplicaciones Web tanto comerciales como OPEN SOURCE frente al Top 10 de OWASP.

- Presentar los resultados comparando los aspectos de seguridad de OWASP frente al firewall de aplicación web.

4. MARCO REFERENCIAL

4.1. ESTADO DEL ARTE

La seguridad en aplicaciones web ha tomado un gran valor en nuestros días, ya que una gran variedad de ataques se han desplegado en diferentes portales web y la seguridad ofrecida por el tradicional firewall es casi nula ante este tipo de eventos. Una de las principales razones detrás de todos estos ataques radica en la falta de conocimiento acerca de los temas de seguridad de los desarrolladores web, dando vía libre a los hacker para que puedan explotar sus vulnerabilidades y así irrumpir la seguridad del sitio web. Los problemas de seguridad en la web son tan graves que según un estudio realizado por la revista "SC Magazine" el 86% de los sitios web tienen al menos una vulnerabilidad de alto riesgo[2].

4.1.1. Cuadrante mágico de Gartner para WAF 2015 Gartner es una empresa dedicada a realizar consultoría e investigación de las tecnologías de la información, para su estudio toma como referencia un gran número de empresas en diferentes sectores, como gobierno, finanzas, telecomunicaciones, mercadeo, entre otras y tiene como finalidad realizar una análisis comparativo de las fortalezas y debilidades de las diferentes marcas en cada uno de los ámbitos de las tecnologías de la información.

En el estudio entregado por Gartner para Web Application Firewall se consideraron las marcas que activamente se están comercializando a organizaciones. Las tecnologías evaluadas son aquellas que brinden una protección de seguridad más allá de lo provisto por firewalls de nueva generación e IPS. Los WAF deben estar en la capacidad de soportar uno o múltiples servidores web, además en este estudio se incluyeron WAF que están en frente de la aplicación y no los que están integrados dentro de la aplicación web.

Otro aspecto que se evidencio en el estudio fue una mayor acogida por parte de las empresas para las marcas que permiten integrarse con otras tecnologías de seguridad, tal como pruebas de seguridad de aplicaciones, monitoreo de bases de datos, manejo de eventos e información de seguridad (en inglés, SIEM)[3].

---

[2] Study: 86 percent of websites contain at least one 'serious' vulnerability [enlínea]. <http://www.scmagazine.com/whitehat-security-release-website-security-statistics-report/article/416402/>[citado en 8 de noviembre de 2015].

[3] D'HOINNE Jeremy, HILS Adam, YOUNG Greg. Magic Quadrant for Web Application Firewalls. En revista: Gartner ID:G00271692. 2015.

En la Figura 1 se presentan las marcas evaluadas por Gartner, ubicando de ellas dentro de sus cuatro cuadrantes, de izquierda a derecha es ponderado su visión de negocio y de abajo a arriba se califica su facilidad de implementación.

Figura 1- Cuadrante mágico de Gartner 2015 para WAF

Fuente: Magic Quadrant for Web Application Firewalls[4]

A continuación se realiza una breve descripción de las fortalezas de cada una de las marcas presentes en WAF:

AdNovum

Esta es una empresa Suiza, opera en número limitado de países, tiene un mayor enfoque en dispositivos orientados a Gestión de Acceso Web (en inglés, WAM) y no es comúnmente referenciado como WAF.

Fortalezas: Incluyen fuertes mecanismos de autenticación y Single Sign-On (SSO). Cuenta con una fuerte experiencia en grandes empresas de Suiza, proporciona licencias demo para pruebas. Clientes están a gusto con las políticas de autoaprendizaje y funcionalidades de seguridad como, encripción de URL CSRF y firmas de formas.

[4] Ibid.

Debilidades: Tiene mayor enfoque hacia WAM que hacia WAF, lo cual limita su crecimiento para crecer en funcionalidad de seguridad. Tiene una limitada expansión hacia otros países fuera de Suiza. Las funcionalidades de inyección de SQL y XSS están enfocadas sobre ModSecurity herramienta libre y en firmas comerciales, y no disponen de otras fuentes de alimentación para investigación de amenazas e inteligencia de ataques.

Akamai

Es una empresa de Estados Unidos, su producto de WAF es llamado Kona Site Defender proporcionada servicios generando un pago mensual basado en los requerimientos y número de aplicaciones a proteger, además cuenta con servicios adicionales como protección a DDoS.

Fortalezas: Permite integrar WAF con protección de DDoS. Está bien diseñado para los casos en los que la prioridad es detener y alertar. Akamai se apoya en la información compartida en internet acerca de anomalías reportada y así genera ponderación de acuerdo a la reputación de las IP. Además, ha incrementado su cobertura en gran número de clientes alrededor del mundo.

Debilidades: Es un servicio que solo está disponible en servicios en la nube y no provee dispositivo en el centro de datos del cliente. Otra falencia detectada está en sus precios los cuales son altos para las pequeñas y medianas empresas (en inglés, SMBs) donde generalmente se prefieren los servicios en la nube. Respecto a temas de seguridad este WAF se base principalmente en firmas y puntuaciones de reputación y lo hace menos adecuado en el bloqueo de casos primarios o repuestas activa, además cuenta con un limitado nivel de autoaprendizaje y personalización de políticas.

Barracuda

Es una empresa de Estados Unidos cuenta con una gran cantidad de productos que son ampliamente aceptados en SMBs, ofrece dispositivos que son instalados en el predio del cliente.

Fortalezas: Tiene gran variedad de plataformas, y es el único que ofrece WAF sobre Azure y en plataformas con vCloud. Barracuda proporciona una fuerte reputación de IP, protección de cookie y capacidades firmado de información de cliente. Además proporciona un gran soporte y cubrimiento a nivel mundial.

Debilidades: Se reportan diferentes problemas referentes a la automatización de la seguridad. El resultado de escaneos de seguridad debe ser importado de forma manual, y las capacidades de aprendizaje son deshabilitados por defectos y

necesitan ser rehabilitados después de cada cambio. Por otra parte se reporta que no dispone de dispositivos para soportar tráfico mayor a 5Gbps.

Citrix

Es una empresa de Estado Unidos, proveedor de los soluciones de Controlador de Entrega de Aplicación (en inglés, ADC), infraestructura en la nube y un amplio portafolio de virtualización. El producto de WAF es llamado Netscaler, por más de una década fue vendido como software y a partir de 2014 se empezó a comercializar como un dispositivo.

Fortalezas: Netscaler incluye características maduras para seguridad web, y puede ser combinado con soluciones de VPN SSL para acceso remoto, también proporcionada una gran habilidad para grandes organizaciones cuando es requerido remover el SSL.

Debilidades: Como la mayoría de fabricantes de ADC, Citrix se enfoca en clientes con ADC y no enfoca sus esfuerzos principales sobre en los temas de seguridad. El fabricante no ofrece o colabora con servicios de protección de DDoS en la nube.

DBAPPSecurity

Es una empresa China, es un fabricante de WAF y soluciones de seguridad a bases de datos. Su producto de DAS-WAF fue liberado en el 2007. Su producto ofrece escaneo de vulnerabilidades en web y bases de datos. En 2014 liberaron su dispositivo hardware, su enfoque está en proporcionar servicios en China en SMBs.

Fortalezas: DAS-WAF incluye aprendizaje automático de políticas y cacheo de aplicaciones web, puede operar en proxy-reverso, proxy transparente o modo monitor. Puede integrar bien con fabricantes de escaneo de vulnerabilidades.

Debilidades: Con respecto a sus competidores tiene algunas falencias con respecto al role de gestión, reportes de actividad detallada y características de autenticación. Los clientes reportan que la gestión y las consolas de reportes requieren mejoras. Por otra parte tiene un mercado limitado fuera de China.

Denyall

Es una empresa Francesa y ha estado en el mercado de WAF desde 2001 su producto se llama rWeb, también ofrece servicios de WAM y Firewall de Servicio Web (en inglés, WSF) y administrador de vulnerabilidades web. Su producto está en versión de software y hardware. Denyall se enfoca en el mercado francés y europeo donde se enfoca en medianas y grandes empresas.

Fortalezas: Los clientes reportan una alta calidad en el soporte y capacidad de respuesta a las solicitudes. Deny all incluye gran variedad de técnicas avanzadas de protección, incluyendo protección y análisis de tráfico JSON, detección de debilidades de código y un agente de navegador ligero. Denyall gana también en su enfoque a la seguridad, ofrece gran variedad técnicas de anti-evasión y protecciones ante SQL y XSS.

Debilidades: El principal foco de esta marca está en Francia y Europa, y tiene una limitada visibilidad, y tiene una limitada cobertura en demás países, también indican que su crecimiento orgánico es lento comparado con sus competidores.

Ergon Informatik

Es una empresa Suiza, su producto de WAF es llamado Airlock. Es implementado en modo de proxy reverso y está disponible como hardware, software o dispositivo virtual, además cuenta varios módulos de autenticación.

Fortalezas: Brinda buena eficiencia de su soporte robustez en su solución de WAF cuando se implementa en soluciones productivas. Las más recientes actualizaciones proporcionan facilidad en la gestión de firmas y mejoras en la visibilidad sobre las funciones de seguridad. Asimismo, incluye técnicas extensas en parámetros de aplicaciones web, tales como cifrados de la URL, protección de cookies y verificación de integridad de parámetros.

Debilidades: Airlocks tiene algunas falencias de funcionalidades avanzadas de seguridad, como son aprendizaje automático de políticas e inyección de token en CSRF. Tampoco ofrece una gestión centralizada y tiene números bajos en implementaciones de WAF.

F5

Es una empresa de Estados Unidos enfocada en las aplicaciones de infraestructura con mayor enfoque en ADC. La solución de WAF se ofrece como un software llamado Application Security Manager (ASM), y a menudo se vende con demás funcionalidades dentro de un mismo hardware. Posee una gran cobertura a nivel mundial y tiene gran acogida por sus módulos de ADC.

Fortalezas: El equipo corporativo de F5 y sus proveedores de canales proporcionan capacidades de logística y soporte que tienen mayor cubrimiento que la gran mayoría de WAF. Además, tiene funcionalidades de scripting que permite la elaboración de reglas personalizadas mediante iRules y características de geolocalización para la ponderación de IP maliciosas.

Debilidades: Los clientes reportan dificultad en su configuración inicial y afinamiento de políticas, junto con problemas para mantener el histórico de los

reportes luego de un reinicio. Además, tal como otros WAF basados en ADC, la mayoría de los compradores de este producto perciben al WAF como un adicional acompañando a su ADC en el modo de proxy reverso, esto pone a ASM en desventaja frente a los WAF nativos.

Fortinet

Es una empresa de Estados Unidos, es fabricante importante en infraestructura de red y seguridad en redes. Ofrece el producto de WAF denominado Fortiweb, un ADC (FortiADC) y protección en bases de datos FortiDB. FortiWeb proporciona múltiples tipos de implementación tanto física como virtual y actúa como proxy transparente y reverso y no puede implementarse en línea.

Fortalezas: Los clientes mencionan la buena reputación de la marca, sus precios competitivos y la satisfacción con otros productos como la justificación de la compra de FortiWeb. Otra ventaja se basa en un sólido hardware con la desencripción del SSL, el uso del antivirus para la inspección de malware sobre servicios de archivos compartidos, un escaneo de vulnerabilidades, implementación fuera de banda (en inglés, OOB) y buen número de reportes predefinidos.

Debilidades: Fortinet es brindado como una línea secundaria comparado con otros productos como Fortigate (Firewall de nueva generación), además cuenta con dificultades para implementarse en ambientes de aplicaciones no triviales y gastan demasiado tiempo para su implementación. Por otro lado Fortiweb cuenta con poca capacidad de integración con respecto a los demás productos que ofrece la marca, tampoco proporcionada soluciones en servicios en la nube.

Imperva

Es una empresa de Estados Unidos, es un fabricante con un amplio legado de WAF bajo la marca SecureSphere. Esta marca cuenta con productos que están enfocados en la seguridad de los datos, incluyendo protección en bases de datos, monitoreo a la actividad de archivos, prevención de ataque de volumétricos de DDoS, seguridad de acceso en la nube.

Imperva se ha posicionado principalmente por su implementación de puente en modo transparente. Este modo de funcionamiento se alinea con los requerimientos de las empresas, debido a que sus implementaciones son más fáciles y evita introducir un segundo proxy. Existen otros servicios de seguridad como WAF en la nube o como servicio, que incluyen los DDoS.

El dispositivo en hardware tiene 7 modelos de soportadas, además cuenta con modelos de virtualización, y sus cajas de mayor capacidad pueden alcanzar un throughput de 10Gbps. Imperva es líder debido a que continuamente gana en

características de seguridad e innovación, además resiste la presión por el precio de sus competidores directos.

Fortalezas: Imperva constantemente ha obtenido altas calificaciones de sus clientes, con altas tasa de satisfacción referente a seguridad, reportes y protección. Además continuamente ha liderado en el mercado de WAF debido a sus nuevas funcionalidades que obligan a sus competidores a reaccionar, incluyendo muchas técnicas avanzadas para mejorar la eficiencia de la protección.

Debilidades: Imperva resulta ser usualmente muy avanzado para SMBs, o en proyectos donde el WAF está siendo desempeñado como una obligación para cumplir con algún requerimiento. Aunque Imperva tiene las mejores funcionalidades de seguridad, la mayoría de WAF ofrece ADC, y el servicio de WAF termina siendo una licencia que se adiciona a un dispositivo y no tiene que pagar por equipo adicional.

NSFOCUS

Es una empresa de China cuenta con servicios de Anti-DDoS, IPS y escaneo de vulnerabilidades y está enfocado a las SMBs.

Fortalezas: Clientes reportan un precio competitivo y buen desempeño, el dispositivo puede integrarse con la solución de Anti-DDoS en la nube. Además, el producto cuanta con certificaciones de calidad como ICSA.

Debilidades: Su mercado está muy limitado a solo China, por otro lado tiene una limitada consola de gestión, restricciones a la implementación de cluster activo/activo y aceleración del SSL. Tampoco tiene características para la autenticación e integración con SIEM para correlación de eventos.

Penta Security

Es una empresa Sur coreana dentro de sus productos se inluye la protección de bases de datos y autenticación con SSO. Su producto de WAF es conocido como Wapples y es ofrecido como hardware, sistema de virtualización o como servicio.

Fortalezas: Sus clientes los califican como una implementación fácil y tiene una baja carga operativa. Wapples se encuentra certificado con EAL4, e incluye características de seguridad como validación de parámetros y cookies.

Debilidades: Wapples dispone de limitadas funciones para crear políticas de seguridad automatizadas, además la robustez de su seguridad depende de su motor genérico y algunas técnicas complementarias en caso de no detectar los ataques.

Positive Technologies

Esta empresa tiene sus oficinas centrales en Moscu, Londres y Boston, su producto es llamado PT Application Firewall. Este producto está disponible en hardware y en una versión de software. La mayoría de sus clientes son agencias del gobierno y de finanzas.

Fortalezas: Los clientes seleccionan esta marca por la facilidad en su configuración a través de su autoaprendizaje y razones heurísticas, además indica que la respuesta a la atención de los incidentes se realiza de forma rápida.

Otras ventajas están en la detección del XSS reflejado al analizar las repuestas de HTTP y usa aprendizaje de máquina para la detección de anomalías.

Debilidades: Este producto joven y está soportado solo en pocas zonas geográficas, tiene algunas falencias para la integración con directorio activo y no soporta el retiro de SSL.

Radware

Es una empresa Israelí y brinda gran variedad de productos de seguridad, donde se incluyen varios tipos de DDoS y WAF (AppWall). AppWall puede ser instalado como máquina física o virtual, además ofrecer una gestión centralizada para la administración. Radware también incluyo en sus servicios el soporte fuera de banda integrándose con la mitigación de DDoS.

Beneficios: AppWall puede ser implementado en modo puente y proxy reverso, que junto con el manejo aprendizaje automático facilitan su implementación. Por otra parte Radware brinda unos precios acordes, además ofrece servicios de multi tenencia y se integra con módulos de autenticación como SSO.

Debilidades: Presenta debilidades para integrar con escáner de vulnerabilidades de otras marcas y soluciones de bases de datos. Otro aspecto desfavorable fue su lenta integración con el ADC lo cual le genero un desventaja competitiva contra sus competidores de ADC/WAF.

Trustwave

Es una empresa de Estados Unidos que proporciona un amplio portafolio de soluciones de seguridad, incluyendo su WAF, protección web, IPS, aplicaciones de seguridad y soluciones relacionadas con SIEM. La solución de WAF tiene compatibilidad con soluciones de su propia marca con escáner de vulnerabilidades y SIEM, además tiene soporte de la herramienta de libre distribución ModSecurity.

Fortalezas: Trustwave soporta ModSecurity que brinda un gran equipo de investigación de amenazas acceso retroalimentación de su comunidad, lo cual es útil para seguir mejorando su WAF. Trustwave también proporciona una configuración por defecto que está preparada para soportar PCI.

Debilidades: Existe incertidumbre del futuro de su empresa por la posible venta a Singtel, además de la posible fusión con Akamai. Por parte de los clientes mencionan un alta tasa de falsos positivos.

United Security Providers

Es una empresa de Suiza que proporciona soluciones de WAM que está integrada con WAF, servidor de autenticación y Gateway de XML. Proporciona diferentes servicios de seguridad gestionados que se integran con otros proveedores. El WAF es soportado en dispositivo físico, virtual, software y servicio en la nube.

Fortalezas: La integración con la solución de WAM ofrece flexibilidad para la autenticación y SSO, además el WAF proporciona avanzadas funcionalidades de seguridad como son encripción de URL, protección contra CSRF, seguridad con cookies y validación del cliente web.

Debilidades: United Security Providers incremento sus esfuerzos en la solución de WAF, sin embargo WAM sigue siendo su producto primario. Los clientes por su parte indica que la gestión y los reportes podrían ser mejorados. Otro falencia que se reporta es el no poder redireccionar tráfico al servicio de DDoS en la nube o integrar con servicios de DDoS de otras marcas.

4.1.2. Antecedentes Con la finalidad de identificar trabajos relacionados con la propuesta planteada, se tomaron algunos artículos y trabajos de grado que proponen temáticas similares a las del proyecto, como se citan a continuación.

En el artículo How to deploy a Web Application Firewall[5] – el autor plantea que el ciclo de vida de la seguridad es un aspecto importante durante la instalación del WAF: asegurar, monitorear, verificar y mejorar; este proceso continuo permite garantizar una implementación exitosa. Por otra parte, sugiere que antes de realizar la conexión a la red de cualquier dispositivo se debe ejecutar pruebas de hardening, junto con la instalación del sistema operativo y parches más recientes.

[5] ComputerWeekly.com. How to deploy a Web application firewall (WAF) [enlínea].<http://www.computerweekly.com/tip/How-to-deploy-a-Web-application-firewall-WAF>[citado en 8 de noviembre de 2015].

Ahora bien cuando se empieza la configuración WAF, un aspecto a revisar son las reglas del negocio definidas en la política de seguridad, tales como permitir el uso de determinados caracteres, longitud de contraseñas, entre otras, de esta forma la configuración del WAF puede ser creada a partir de estas definiciones. Probar cuidadosamente, es esencial especialmente si el sitio web a proteger hace uso de parámetros que no se alinean con los estándares web, lo que incluye la revisión de cabeceras, URL o cookies. Pruebas adicionales deben ser permitidas si usan versiones multilenguaje de una aplicación, puesto que estas manejan diferentes tipos de caracteres.

Las pruebas en los WAF deben ser lo más parecidas a los ambientes productivos, simulando situaciones de alta concurrencia y volumen de tráfico y de esta forma identificar los posibles cuellos de botella que se pueden llegar a generar a partir de la implementación de un WAF.

En el artículo Intrusion Detection FAQ: What is the difference betweenan IPS and a Web Application Firewall?[6]– se hace una revisión de las diferencias entre los IPS y los WAF, ambos tipos de dispositivos se asemeja en que generalmente son ubicados “in-line” y observan todo el tráfico que fluye a través de ellos además pueden ser dispositivos de red o aplicaciones cliente en los equipos terminales. Pero la diferencia entre los dispositivos está en la habilidad de analizar la lógica de los paquetes de las aplicaciones web, ya que los IPS interrogan el tráfico contra firmas y anomalías, mientras que el WAF evalúa el comportamiento de lo que es solicitado y retornado. Los WAF generalmente son implementados en modo de proxy justo por delante de la aplicación web, de esta forma ellos no observan todo el tráfico sobre nuestra red. Al monitorear el tráfico antes de que este llegue a la aplicación web, el WAF puede analizar la solicitud antes de pasarla. Esto representa una ventaja ante el IPS porque el IPS está diseñado para analizar todo el tráfico de la red y no pueden analizar la aplicación a fondo.

La ventaja más importante es que los WAF no solo detectan ataques ya conocidos, ellos también puede detectar ataques nuevos o desconocidos. Al detectar patrones inusuales o inesperados en el tráfico. Por ejemplo si un WAF detecta que la aplicación retorna más información de la que es esperada, el WAF puede bloquear y alertar la novedad.

[6]SANS. Intrusion Detection FAQ: What is the difference between an IPS and a Web Application Firewall?[enlínea]. <https://www.sans.org/security-resources/idfaq/ips-web-app-firewall.php>[citado en 8 de noviembre de 2015].

En el documento de Best Practices: Use of Web Application Firewalls[7] – Este documento hace una revisión de diferentes aspectos que se debe considerar al momento de usar un WAF, algunos de los más importantes y que tienen más relevancia con nuestro proyecto son:

- OWASP Top 10 usando WAF –En la tabla inferior el WAF es comparado con el Top10 de las vulnerabilidades de OWASP e indica el trabajo requerido para proteger la vulnerabilidad

Tabla 1 – OWASP aplicado a un Firewall de aplicaciones Web

TOP 10	Observaciones
A1. Cross-Site Scripting (XSS)	El WAF no permite la validación de salida en este caso, dado que el no reconoce el contexto de los datos. La validación debe ser llevada a cabo durante la fase de entrada y puede ser correlacionada con la salida.
A2. InjectionFlaws	WAF con lista negra: en principio el equipo solo podría buscar por caracteres específicos y prevenir su procesamiento. Pero, este modo tiene una desventaja cuando se usan ataques de evasión de filtrado y no se ha llevado a cabo una normalización de los parámetros de entrada. Aunque este modo trabaja bien con ataques conocidos puede tener limitación con protocolos o desarrollos no conocidos por el WAF. WAF con lista blanca: De este modo todas las entradas están contenidas en listado, además el WAF podría sugerir otras entradas adicionales en campos específicos durante la fase de aprendizaje. De esta forma la mayoría de los campos de entrada pueden estar protegidos contra los tipos de ataques por inyección de código.
A3. Malicious File Execution	Realizar una lista blanca de los parámetros permitidos de URL externas al sistema. Además, se puede incluir escáner externos a través del protocolo ICAP y análisis de respuesta para prevenir la publicación de datos críticos.
A4. Insecure Direct Object Reference	Proteger contra la manipulación de ID usando virtualización de ID o protegiendo parámetros ocultos.
A5. Cross-site Request Forgery (CSRF)	Puede ser prevenido usando token de página o cifrado de la URL.

[7] DERMANN Maximiliam, DZIADZKA Mirko, HEMKEMEIER Boris, HOFFMANN Achim MEISEL Alexander, ROHR Matthias, SCHREIBER Thomas. Best Practices: Use of Web Application Firewalls [enlínea]. <https://www.owasp.org/images/b/b0/Best_Practices_WAF_v105.en.pdf>[citado en 8 de noviembre de 2015].

A6.1 InformationLeakage	El WAF genera filtrado automático de la información. Generalmente los atacantes buscan archivos donde se encuentra las contraseñas.
A6.2 Improper Error Handling	Difícil para detectar
A7.1 BrokenAuthentication	Depende de la habilidad del WAF, ya que se puede llevar a cabo la autenticación que un ente independiente a la aplicación sin requerir cambios sobre la aplicación.
A7. 2 Session Management	Asegurar las sesiones inseguras, por ejemplo token de páginas.
A8. InsecureCryptographic Storage	Función no implementada en los WAF
A9. InsecureCommunitations	El WAF puede asegurar aplicación que usan HTTPS
A10. Failure to Restrict URL Access	El uso de token de página o cifrado de la URL puede ser usado para restringir a los usuarios de recibir aplicaciones como enlaces. El uso de listas blancas o negras y especificar el tipo de extensiones que puede ser permitidas.

Fuente: F5 Security on Owasp Top 10[8]

4.2. MARCO TEÓRICO

4.2.1. Aspectos generales del WAF Los WAF son dispositivos o aplicaciones de software que permiten analizar la comunicación a nivel de la capa de aplicación para servicios HTTP o HTTPS, examinando todos los datos enviados por el navegador del lado cliente hacia el servidor web. El WAF por lo tanto tiene la posibilidad de establecer políticas de seguridad con base en diferentes criterios que incluyen: firmas de ataques conocidos, estándares de los protocolos y anomalías en el tráfico de la aplicación. De esta forma en el momento que 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 lo contrario si la solicitud es legitima el WAF permitirá que la comunicación fluya[9].

[8] F5 Security on Owasp Top 10 [en línea].En <https://devcentral.f5.com/articles/f5-security-on-owasp-top-10>

9 REVISTASEGURIDADDEFENSA DIGITAL. Firewall de Aplicación Web – Parte II [en línea].En <http://revista.seguridad.unam.mx/numero-17/firewall-de-aplicaci%C3%B3n-web-parte-ii>[citado en 8 de noviembre de 2015].

En su modo de implementación los WAF pueden seguir un modelo de seguridad positiva o negativa, de acuerdo con las políticas de seguridad que se establezcan para cada aplicación. Donde un modelo positivo solo permite pasar el tráfico que es conocido y el resto es bloqueado, mientras que el modelo negativo permite todo el tráfico e intenta bloquear lo detectado como malicioso

- **Ubicación de un WAF en la red:** Los WAF implementados mediante dispositivo requieren ser ubicados en lugar donde todas las conexiones hacían los servidores web pasen a través de él, los montajes pueden ser hechos "in-line" donde todo el tráfico fluye a través del WAF o "out of band" que con el uso de un puerto en modo monitor puede copiar todo el tráfico dirigido a los servidores y redirigir al dispositivo WAF. Sin embargo,si la implementación no es hecha mediante un dispositivo, existen aplicaciones WAF las cuales son instaladas en los servidores o clientes y procesan toda la información antes que llegue en dispositivo final.

- **Modos de Operación de un WAF:** Pueden operar en diferentes modos dependiendo el fabricante del dispositivo, así que es necesario verificar las guías de configuración para validar si el modo de operación deseado es soportado en el equipo adquirido. Cada modo ofrece ventajas y desventajas lo cual requiere ser evaluado por cada organización para definir el modo más adecuado; en el documento de Beechy Jim se mencionan las cuatros formas que puede operar un WAF los cuales son:

    - Proxy reverso – Este modo es el más común y con el mayor beneficio en las implementaciones de WAF. Ubica el WAF "in-line" y expone una dirección IP donde todas las conexiones dirigidas al servidor terminan en el WAF, y luego el dispositivo reenvía la información al servidor web en nombre del navegador origen. Este modo presenta otro tipo de ventajas debido a su control sobre las conexiones TCP, lo cual es provechoso para otras funcionalidades como balanceo de carga, aceleración de SSL, compresión o caching.

    - Proxy transparente – Este modo ubica el WAF "in-line" tal como lo hace el proxy reverso, sin embargo el dispositivo no tiene una dirección IP asignada. A nivel de red no requiere grandes cambios, pero no puede proporcionar funcionalidades adicionales como el proxy reverso.

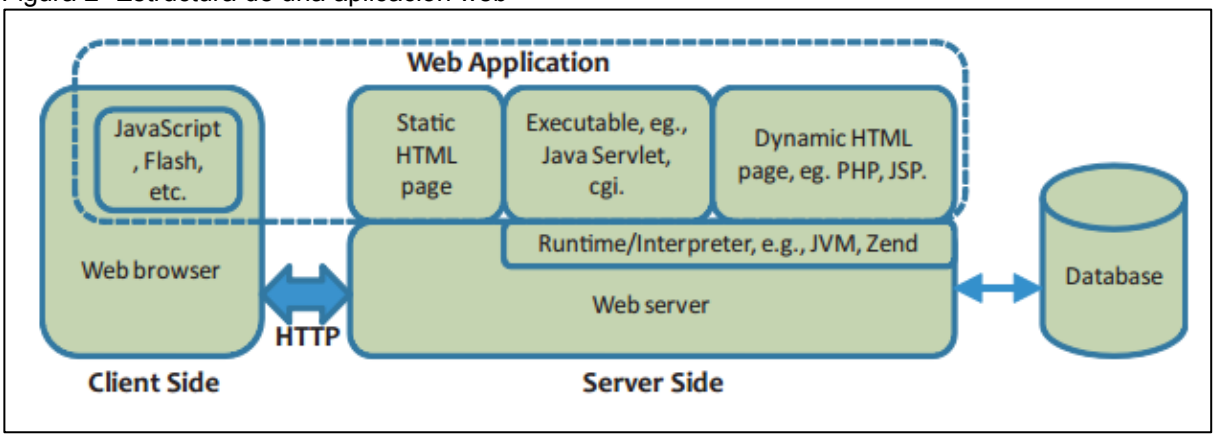
    - Monitoreo de red o fuera de banda –En este modo el WAF no está en línea y recibe el tráfico a través de un puerto en modo monitor configurado en un switch o conectado a hub. Este modo es ideal para probar un WAF sin generar impacto en los servicios. Si es

deseado el WAF podría realizar bloqueos de las conexiones al enviar un reinicio a la conexión TCP.

- Basado en servidor o cliente – En este modo los WAF son aplicaciones de software que se instalan en el servidor o cliente, no pueden proporcionar ventajas adicionales a la conexión, pero tienen como ventaja la posibilidad remover la funcionalidad de WAF sin requerir cambios que afecten la infraestructura. Aunque una desventaja de este tipo de soluciones son el incremente en el consumo de los recursos debido a análisis requerido en los equipos servidor o cliente[10].

4.2.2. Funcionamiento de una aplicación web El ecosistema de las páginas web es un mundo integral que habilita la posibilidad de tener entrega de servicios y contenido de forma dinámica. Una aplicación web consiste de un código sobre un cliente y un servidor. En el lado del servidor, la aplicación recibe las entradas del usuario vía solicitudes (Request HTTP) a través del navegador del cliente e interactúa con archivos locales del sistemas, base de datos u otros componentes para acceder a información. La respuesta del servidor es enviada a través de la página HTML por medio de un código de respuesta (HTTP Response). En el lado del cliente, las páginas HTML son presentadas junto con el código que se encuentre embebido y ejecutado por el navegador web. En la Figura 2 están los componentes presentes en una aplicación web.

Figura 2- Estructura de una aplicación web



Fuente: Artículo Web Page Access Model Study for Secure Execution and Prevention of Source Code Theft of Web Pages[11]

[10] Beechey Jim. Web Application Firewalls:Defense in Depth for Your Web Infrastructure[enlínea]. En <https://www.sans.edu/student-files/projects/200904_01.doc>[citado en 8 de noviembre de 2015].

[11] SHAH Ishit, SHAH Nisha. Web Page Access Model Study for Secure Execution and Prevention

- **Estado de sesión** HTTP es un protocolo no orientado al estado de una sesión (en inglés, stateless), donde cada solicitud es independiente de la otra. Pero, al implementar funcionalidades más complejas tener una sesión orientada al estado de la conexión (en inglés, statefull) es requerida, sin embargo esta debe ser implementada en una infraestructura ya definida como stateless. De esta forma, la abstracción de las sesiones web es adoptada para ayudar a la aplicación a identificar y correlacionar una serie de solicitudes web del mismo usuario durante un determinado lapso de tiempo. El estado de una sesión web graba el histórico de solicitudes y determina la futura ejecución de la aplicación web. Las sesiones puede ser mantenidas del lado cliente a través de: una cookie, una forma oculta o la reescritura de la URL o en el lado del servidor[12].

- **Lógica de la aplicación** La lógica del negocio traza la funcionalidad de la aplicación web y cada aplicación tiene una lógica particular. La funcionalidad se convierte en un control de la aplicación que significa que la aplicación debería ser ejecutada de la forma que fue diseñada por el desarrollador

4.2.3. Vulnerabilidades y Vectores de ataques en aplicaciones web Una aplicación web segura tiene que garantizar determinas propiedades de seguridad fijados en el modelo de amenazas (en inglés, Threat Model), en este modelo usualmente es considerado lo siguiente:

- Las aplicaciones web debe ser benignas e implementadas en una infraestructura confiable y endurecida, lo que incluye servidor web, sistema operativo, intérprete, base computacional, etc.
- El atacante es capaz de manipular los contenidos o las secuencia de las solicitudes enviadas a la aplicación web, pero no puede comprometer directamente la infraestructura o el código de la aplicación. Las vulnerabilidades dentro de la implementación de la aplicación web podrían violar el nivel de seguridad permitir una ataque exitoso.

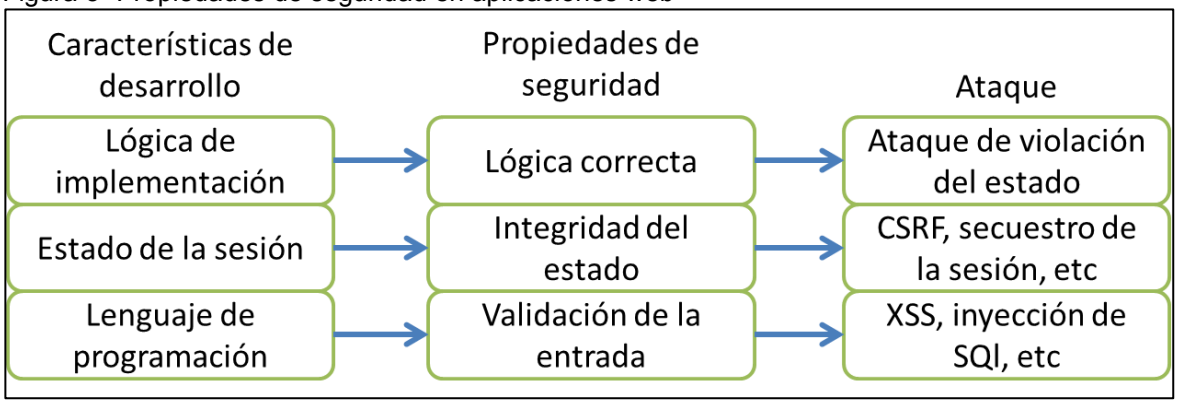
Cada aplicación web en particular debe cumplir con las propiedades de seguridad presentes en la Figura 3. La validación de la entrada significa que cada entrada hecha por el usuario debe ser validada antes que pueda ser utilizada por la aplicación web; la integridad del estado debe ser mantenida y no puede la información de la sesión no puede ser manipulada; además la aplicación debe ejecutarse en la lógica correcta.

---

of Source Code Theft of Web Pages. En: revista IOSR Journal of Computer Science (IOSR-JCE). (2014). e-ISSN: 2278-0661, p-ISSN: 2278-8727

[12] Ibid.

Figura 3- Propiedades de seguridad en aplicaciones web



Fuente: El autor

4.2.4. Análisis del TOP 10 de OWASP frente al WAF El objetivo del TOP 10 de OWASP permite catalogar las vulnerabilidades más presentes y críticas que están en las aplicaciones web, es un documento que se presenta cada 3 años y su versión más reciente data del 2013. OWASP evalúa las vulnerabilidades de acuerdo a su modo de operación, identifica la probabilidad de impacto en el negocio y la posibilidad que se materialice.

Con el fin de realizar un acercamiento de las funcionalidades del WAF frente OWASP, se realiza un revisión del TOP 10 de OWASP de 2013 donde se revisan los mecanismos que puede brindar un WAF para mitigar o remediar estas vulnerabilidades para el caso que aplique.

- **A1 Inyección** Descripción: Los ataques de inyecciones de SQL se presentan cuando bases de datos y otros sistemas son vulnerables a que se inyecten datos no confiables o maliciosos a través del sistema, estos datos pueden filtrarse a los usuarios y potencialmente infectarlos con malware. Las vulnerabilidades más comunes se encuentran relacionadas con:

  - Solicitudes de LDAP
  - Solicitudes de SQL
  - Solicitudes de XPath
  - Argumentos en los programas
  - Comandos de sistema operativo

  El impacto de este tipo de vulnerabilidades es severo ya que pueden presentarse pérdida o corrupción de la información y negación del acceso[13].

[13] DEFENCELY UNBREAKABLE. OWASP TOP 10 [en línea]. En <https://www.defencely.com/owasp>[citado en 13 de marzo de 2016].

Protección: El WAF realiza la inspección del tráfico y bloquea la inserción de scripts maliciosos. Lo hace mediante la detección de patrones de inyección, haciendo cumplir el uso los metacaracteres específicos dentro de la URI o en los parámetros de la aplicación[14].

- **A2 Pérdida de autenticación y gestión de sesiones** Este tipo de ataques buscan recuperar o robar las contraseñas, usuarios, detalles de una cuenta u otra información de sesiones ya establecidas. Generalmente se presentan cuando el atacante se hace pasar por alguien confiable mediante la información que ya han recaudo de forma previa, al realizar esto el atacante solicita información que puede permitir el robo de una cuenta, robo de información, entre otros.

  Algunas medidas que se pueden tener en cuenta para prevenir este tipo de ataques, incluyen la expiración de las sesiones, expiración de inicio de sesión y demás estrategias que protejan la información del usuario.

  Protección: El WAF previene a los atacantes de explotar una autenticación rota y manejo de sesiones. Además detiene ataques como secuestro de sesiones, sesiones fijas o manipulación de sesiones. Esto lo permite mediante, el manejo de tiempos de expiración para los portales de ingreso, monitoreo de los patrones empleados en las solicitudes, además de tener la posibilidad de firmar las cookies así las mismas no pueden ser manipuladas.

- **A3 Secuencia de comandos en sitios cruzados (XSS)** Esta vulnerabilidad toma ventaja de problemas de seguridad en los navegadores web y otro tipo de intérpretes en lugar de afectar el sitio mismo. Al igual que la inyección de SQL, es difícil rastrear un XSS a una única fuente, puesto que en cualquier momento una vulnerabilidad o hueco de seguridad se puede presentar.

  Protección: El WAF protege contra ataques de XSS al validar patrones de comportamiento, dando cumplimiento a uso preciso de determinados metacaracteres dentro de la URI y los nombres de parámetros. El WAF también permite inspeccionar valores de parámetros y hacer cumplir unos valores predefinidos, longitudes y uso de los metacaracteres apropiados.

- **A4 Referencia directa insegura a objetos** Este tipo de vulnerabilidad ocurre generalmente cuando los niveles de autenticación no son suficientemente verificados y el usuario puede ganar acceso hacia el

[14] F5. F5 Security on Owasp Top 10 [en línea]. En <https://devcentral.f5.com/articles/f5-security-on-owasp-top-10>[citado en 13 de marzo de 2016].

sistema. En la mayoría de los casos esto pasa al ingresar al sistema y tomar ventaja de vulnerabilidades para habilitar niveles de acceso incorrectos.

Protección: EL WAF perfila la aplicación para identificar y entender lo que es el comportamiento esperado para cada elemento. Por ejemplo, saber cuál es el tipo de entrada para permitir para cada campo de un formulario.

- **A5 Configuración de seguridad incorrecta** Los atacantes quienes quieren disfrazar su verdadera identidad u otra información personal a menudo se apoyan en errores de configuraciones de seguridad. Este involucra intentar accesos con cuentas por defecto, credenciales comunes, directorios web desprotegidos o páginas sin uso.

  Protección: El WAF protege contra configuraciones inadecuadas a través del perfilamiento y entendimiento de las entradas normales. Además, permite corregir problemas relacionados por errores del administrador con: el sistema operativo, servidor web o de aplicación, código de aplicación, ya que el dispositivo identifica patrones de vulnerabilidades conocidas en la aplicación.

- **A6 Exposición de datos sensibles** Este evento requiere un planeación extensa, inicialmente puede perpetrarse con un hombre en el medio (en inglés, MiTM) donde el atacante captura toda la información que dispone la página web, con base en esto por ejemplo podría capturar la información de la tarjeta de crédito de un usuario legitimo siempre y cuando la información que es intercambiada entre el cliente y el servidor no disponga de un cifrado seguro.

  Protección: El WAF mitiga la exposición de datos sensibles al inspeccionar el tráfico saliente, en especial datos tales como número de tarjetas y números de seguridad social, y al detectar un exposición de los datos el WAF podría bloquearlo antes que la respuesta llegue al usuario final.

- **A7 Inexistente control de acceso a nivel de funcionalidades** La mayoría de aplicaciones web verifican el nivel de acceso antes de hacer la funcionalidad accesible al usuario, sin embargo, si ese control de acceso no es verificado en el servidor, los atacantes van a ser capaces de penetrar la aplicación sin la respectiva autorización[15].

[15] TutorialPoints. Security Testing - Missing Function Level Access Control [en línea]. <En

Protección: El WAF aplica diferentes niveles de control para prevenir acceso no autorizado a las funciones de la aplicación. A nivel de aplicación, el WAF usa un perfilamiento dinámico que para aprender cuales URL o funciones requieren un identificador valido de la sesión tal como una cookie. Si una cookie no está presente o si el usuario la ha manipulado, entonces el WAF bloqueará la sesión.

El WAF también previene de forceful browsing que ocurre cuando un atacante intenta acceder a URI ocultas o protegidas al enumerar los diferentes archivos y directorios de la aplicación web. El WAF analiza la página web en cuestión y asegura que los archivos y funciones sean accedidos en el orden correcto. Otro control que ofrece el WAF es la restricción de ciertos archivos que solo pueden ser accedido por las IP registradas.

- **A8 Falsificación de peticiones en sitios cruzados (CSRF)** Un CSRF obliga a un usuario autenticado, victima, a enviar una solicitud falsificada, incluyendo la cookie de sesión de la víctima a una aplicación vulnerable lo cual permite al atacante forzar al navegador de la víctima a generar un solicitud tal que la página vulnerable perciba como legitima la solicitud de la víctima.

  Protección: el WAF mitiga este tipo de vulnerabilidad al adicionar un apéndice a cada URL, esta adición no puede ser adivinada por el atacante y por lo tanto disminuye el riesgo de este tipo de ataque. Adicionalmente, los WAF pueden manejar tiempos de expiración de las sesiones y así controlar el tiempo que las sesiones se mantienen establecidas.

- **A9 Uso de componentes con vulnerabilidades conocidas** Ningún paquete de software es perfecto, e incluso las organizaciones que distribuyen esos componentes trabajan para hacer buenos y seguros sus productos, sin embargo ellos son conscientes que los errores pasan. Cuando se usan librerías externas en los proyectos es importante tener en cuenta que se está adquiriendo una buena funcionalidad, pero también se está abriendo la puerta para potenciales bugs o vulnerabilidades de seguridad de dicho fabricante.

  Para prevenir este tipo de inconvenientes es importante tomarse un tiempo para analizar la documentación de los componentes que se agregan,

---

http://www.tutorialspoint.com/security_testing/missing_function_level_access_control.htm>[citado en 13 de marzo de 2016].

revisar foros y publicaciones acerca del componente. En la mayoría de los casos los proveedores solucionan bastante rápido las vulnerabilidades, pero también es requerido por parte de los administradores mantener al día los componentes[16].

Protección: Los WAF hacen uso de motores de correlación de eventos, donde se analizan eventos como inyecciones de SQL, XSS, detección de violaciones al protocolo HTTP e identifica ataques de día cero para explotar vulnerabilidades. Por otra parte, algunos fabricantes de WAF, una vez una nueva vulnerabilidad de es publicada se desarrolla una nueva firma o conjunto de políticas que permitan remediarla de esta forma se ayudan a minimizar los riesgos para las aplicaciones que no se les hagan las debidas actualizaciones.

- **A10 Redirecciones y reenvíos no válidos** La mayoría de aplicaciones web frecuentemente redirigen o reenvía a los usuarios a otras páginas o sitios externos, sin embargo, sin validar la credibilidad de la página los atacantes pueden redirigir a sus víctimas a sitios con malware o phishing, o usar envíos a páginas de acceso no autorizado.

  Protección: Este tipo de ataques operan como un ataque de XSS a través de un enlace en un correo o foro, los WAF previenen este tipo de ataque al validar el origen de la solicitud y determinar si es una fuente confiable o no. Otros mecanismos de detección incluyen la detección de violaciones a nivel del protocolo HTTP junto con inusuales caracteres como son; comas o diagonal adelante, de esta forma se puede relacionar alguna actividad maliciosa.

4.3. MARCO CONCEPTUAL

- **CGI:** La interfaz de entrada común (en inglés) permite a un cliente solicitar datos de un programa que se ejecuta en un servidor web. Esta funcionalidad ha permitido crear contenidos dinámicos en las páginas web.

- **Cookies**: Es información de las páginas web que se almacena de forma en local en los navegadores del usuario, de tal forma que dicha información

[16] CREDERA. Top 10 Web Security Risks: Using Components with Known Vulnerabilities (#9) [en línea]. <En https://www.credera.com/blog/technology-insights/java/top-10-web-security-risks-using-components-known-vulnerabilities-9/>[citado en 13 de marzo de 2016].

pueda ser consultada posteriormente el servidor. Generalmente, son usadas para almacenar información de autenticación, de esta forma los usuarios no deben no deben ingresar sus credenciales de manera recurrente durante una sesión de trabajo.

- **Cross-site Scripting**: Conocido también como XSS es una vulnerabilidad presente en las páginas web, se presenta cuando los hackers envían código malicioso a la aplicación web para obtener información de los usuarios.

- **HTML**: Es un lenguaje de programación empleado para páginas web, se basa en el uso de etiquetas que le permiten al navegador interpretar la información y de esta forma se diferencia el contenido que es presentado.

- **IPS**: Es un sistema de software que controla el acceso en la red ante abusos o ataques computacionales, mediante el análisis del comportamiento de las aplicaciones y las bases de firmas que dispone el software de forma local.

- **Javascript**: Es un lenguaje de programación utilizado mayormente del lado del cliente y permite mostrar contenidos dinámicos y con efectos al usuario, estas opciones son configuradas dentro del código HTML e interpretadas por el navegador localmente.

- **OWASP**: Es un documento acerca de los riesgos de seguridad más presentes en las aplicaciones web, la lista es actualizada cada tres años y la más reciente fue entregada en el 2013. Tiene como finalidad generar conciencia acerca de la seguridad en las aplicaciones a través de la identificación de los riesgos más críticos en las compañías.

- **PCI-DSS**: Es un estándar dirigido a las empresas de tarjetas de pago su objetivo es presentar una guía que ayude a las organizaciones a procesar, almacenar o transmitir la información de los clientes de forma segura y de esta forma minimizar los fraudes que se puedan generar por el robo o manipulación de los datos.

- **SQL Injection**: Es una vulnerabilidad informática a un atacante realizar consultas a una base de datos, se aprovecha de un filtrado inadecuado por parte del administrador de la información que puede ser ingresada por el usuario. Esta técnica permite modificar o robar información de las bases de datos de las aplicaciones web.

- **Vulnerabilidad informática**: Son puntos débiles de un sistema de información que le permiten a un atacante comprometer la integridad, disponibilidad o confidencialidad del sistema.

4.4. MARCO LEGAL

4.4.1. Ley De Delitos Informáticos En Colombia[17]

La Ley 1273 de 2009

Por medio de la cual se modifica el Código Penal, se crea un nuevo bien jurídico tutelado - denominado "de la protección de la información y de los datos"- y se preservan integralmente los sistemas que utilicen las tecnologías de la información y las comunicaciones, entre otras disposiciones.

EL CONGRESO DE COLOMBIA
DECRETA:

Artículo 1°. Adicionase el Código Penal con un Título VII BIS denominado "De la Protección de la información y de los datos", del siguiente tenor:

CAPITULO. I

De los atentados contra la confidencialidad, la integridad y la disponibilidad de los datos y de los sistemas informáticos

- Artículo 269A: ACCESO ABUSIVO A UN SISTEMA INFORMÁTICO. El que, sin autorización o por fuera de lo acordado, acceda en todo o en parte a un sistema informático protegido o no con una medida de seguridad, o se mantenga dentro del mismo en contra de la voluntad de quien tenga el legítimo derecho a excluirlo, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes.

- Artículo 269B: OBSTACULIZACIÓN ILEGÍTIMA DE SISTEMA INFORMÁTICO O RED DE TELECOMUNICACIÓN. El que, sin estar facultado para ello, impida u obstaculice el funcionamiento o el acceso normal a un sistema informático, a los datos informáticos allí contenidos, o a una red de telecomunicaciones, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes, siempre que la conducta no constituya delito sancionado con una pena mayor.

---

[17] Ley colombiana No. 1273 de 2009 [en línea]. <http://www.mintic.gov.co/portal/604/articles-3705_documento.pdf>

- Artículo 269C: INTERCEPTACIÓN DE DATOS INFORMÁTICOS. El que, sin orden judicial previa intercepte datos informáticos en su origen, destino o en el interior de un sistema informático, o las emisiones electromagnéticas provenientes de un sistema informático que los trasporte incurrirá en pena de prisión de treinta y seis (36) a setenta y dos (72) meses.

- Artículo 269D: DAÑO INFORMÁTICO. El que, sin estar facultado para ello, destruya, dañe, borre, deteriore, altere o suprima datos informáticos, o un sistema de tratamiento de información o sus partes o componentes lógicos, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes.

- Artículo 269F: VIOLACIÓN DE DATOS PERSONALES. El que, sin estar facultado para ello, con provecho propio o de un tercero, obtenga, compile, sustraiga, ofrezca, venda, intercambie, envíe, compre, intercepte, divulgue, modifique o emplee códigos personales, datos personales contenidos en ficheros, archivos, bases de datos o medios semejantes, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes.

Al respecto es importante aclarar que la Ley 1266 de 2008 definió el término dato personal como “cualquier pieza de información vinculada a una o varias personas determinadas o determinables o que puedan asociarse con una persona natural o jurídica”. Dicho artículo obliga a las empresas un especial cuidado en el manejo de los datos personales de sus empleados, toda vez que la ley obliga a quien “sustraiga” e “intercepte” dichos datos a pedir autorización al titular de los mismos.

- Artículo 269G: SUPLANTACIÓN DE SITIOS WEB PARA CAPTURAR DATOS PERSONALES. El que con objeto ilícito y sin estar facultado para ello, diseñe, desarrolle, trafique, venda, ejecute, programe o envíe páginas electrónicas, enlaces o ventanas emergentes, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes, siempre que la conducta no constituya delito sancionado con pena más grave.
En la misma sanción incurrirá el que modifique el sistema de resolución de nombres de dominio, de tal manera que haga entrar al usuario a una IP diferente en la creencia de que acceda a su banco o a otro sitio personal o de confianza, siempre que la conducta no constituya delito sancionado con pena más grave.
La pena señalada en los dos incisos anteriores se agravará de una tercera parte a la mitad, si para consumarlo el agente ha reclutado víctimas en la cadena del delito.
Artículo 269H: *Circunstancias de agravación punitiva*: Las penas imponibles de acuerdo con los artículos descritos en este título, se aumentarán de la mitad a las tres cuartas partes si la conducta se cometiere:

1. Sobre redes o sistemas informáticos o de comunicaciones estatales u oficiales o del sector financiero, nacionales o extranjeros.
2. Por servidor público en ejercicio de sus funciones.
3. Aprovechando la confianza depositada por el poseedor de la información o por quien tuviere un vínculo contractual con este.
4. Revelando o dando a conocer el contenido de la información en perjuicio de otro.
5. Obteniendo provecho para sí o para un tercero.
6. Con fines terroristas o generando riesgo para la seguridad o defensa nacional.
7. Utilizando como instrumento a un tercero de buena fe.
8. Si quien incurre en estas conductas es el responsable de la administración, manejo o control de dicha información, además se le impondrá hasta por tres años, la pena de inhabilitación para el ejercicio de profesión relacionada con sistemas de información procesada con equipos computacionales.

CAPITULO. II

De los atentados informáticos y otras infracciones

Artículo 269I: *Hurto por medios informáticos y semejantes*. El que, superando medidas de seguridad informáticas, realice la conducta señalada en el artículo 239 manipulando un sistema informático, una red de sistema electrónico, telemático u otro medio semejante, o suplantando a un usuario ante los sistemas de autenticación y de autorización establecidos, incurrirá en las penas señaladas en el artículo 240 de este Código.

Artículo 269J: *Transferencia no consentida de activos*. El que, con ánimo de lucro y valiéndose de alguna manipulación informática o artificio semejante, consiga la transferencia no consentida de cualquier activo en perjuicio de un tercero, siempre que la conducta no constituya delito sancionado con pena más grave, incurrirá en pena de prisión de cuarenta y ocho (48) a ciento veinte (120) meses y en multa de 200 a 1.500 salarios mínimos legales mensuales vigentes. La misma sanción se le impondrá a quien fabrique, introduzca, posea o facilite programa de computador destinado a la comisión del delito descrito en el inciso anterior, o de una estafa.

Si la conducta descrita en los dos incisos anteriores tuviere una cuantía superior a 200 salarios mínimos legales mensuales, la sanción allí señalada se incrementará en la mitad.

Artículo 2°. Adiciónese al artículo 58 del Código Penal con un numeral 17, así:

Artículo 58. *Circustancias de mayor punibilidad*. Son circunstancias de mayor punibilidad, siempre que no hayan sido previstas de otra manera:

(...)

17. Cuando para la realización de las conductas punibles se utilicen medios informáticos, electrónicos o telemáticos.

Artículo 3°. Adiciónese al artículo 37 del Código de Procedimiento Penal con un numeral 6, así:

Artículo 37. *De los Jueces Municipales*. Los jueces penales municipales conocen:
(...)

6. De los delitos contenidos en el título VII Bis.

Artículo 4°. La presente ley rige a partir de su promulgación y deroga todas las disposiciones que le sean contrarias, en especial el texto del artículo 195 del Código Penal.

5. DISEÑO METODOLÓGICO

5.1. TIPO DE ESTUDIO Y DISEÑO DE CONTRASTACIÓN DE HIPÓTESIS

Se realizó un tipo de estudio correlacional[18], ya que busca llevar a cabo análisis comparativo de un WAF de libre distribución ante uno comercial e identificar su efectividad para proteger a los servidores web de las vulnerabilidades identificadas en el TOP 10 de OWASP.

5.2. POBLACIÓN, MUESTRA Y MUESTREO

La población involucrada en este proyecto comprende cualquier tipo de aplicación web, independiente del propósito de la misma, ya sea un: portal bancario en línea, comercio electrónico, portal de estudio, servicios médicos, etc.

La muestra se tomará de las conexiones realizadas en la red prototipo.

El muestreo estará basado en los diferentes tipos de información que puede ser modificada por cliente.

5.3. MÉTODOS, TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS

El método empleado para el desarrollo del proyecto inicialmente se basará en la investigación de las funcionalidades disponibles en los WAF que cubran el top 10 de las vulnerabilidades web reportadas por OWASP, posteriormente con la información recolectada se realizará un diseño que permitan probar el funcionamiento del dispositivo.

Las herramientas utilizadas estarán basadas en el sistema operativo Kali Linux, como son:

- Nikto – El cual es un escáner que desempeña pruebas de ataques hacia servidores web, incluyendo potenciales archivos CGI, además permite revisar la presencia de múltiples archivos de index[19].

---

[18] [en línea]. <Tipos de studio http://www.tiposde.com/ciencia/estudio/tipos-de-estudio.html>
[19] CIRT.net. Nikto2. En < https://cirt.net/Nikto2>[citado en 28 de noviembre de 2015].

- Paros – Es un proxy HTTP/HTTPS basado en Java para realizar pruebas de vulnerabilidades web. Soporto editar y ver los mensajes HTTP on-the-fly. También permite escaneos inteligentes para XSS e inyecciones de SQL[20].
- LiveHTTPHeaders - Esta herramienta permite modificar las cabeceras HTTP al enviar las solicitudes o alterar sus respuestas[21].
- Webgoat – Este es un portal web con diferentes vulnerabilidades presentes y que permitirá emular los servicios que se prestan en una aplicación web[22].
- Wireshark - Esta herramienta permite el análisis del tráfico y permite identificar la estructura de la información[23].

La recolección de datos se hará tanto en los dispositivos WAF para evidenciar los eventos de seguridad generados como en los equipos cliente con el fin de validar el tipo de vector ataque empleado.

5.4. PLAN DE PROCESAMIENTO Y ANÁLISIS A DATOS

Para la validación de los datos se hará uso de: la herramienta de logs que dispone cada uno de los WAF de forma nativa y analizadores de tráfico para comparar los eventos detectados por el dispositivo versus los registros generados en las herramientas empleadas para los ataques, y así verificar la relación de los eventos generados y detectados. Posterior a la validación, se procederá con la codificación de la información mediante la clasificación de: los ataques, herramientas empleadas, acción realizada por el WAF y evento registrado en el dispositivo. Finalmente, se documentarán los resultados obtenidos en el WAF luego de ejecutar las pruebas.

---

[20] TestingSecurity.com. Paros Proxy. En <http://www.testingsecurity.com/paros_proxy> [citado en 28 de noviembre de 2015].

[21] TestingSecurity.com. LiveHTTPHeaders. En <http://www.testingsecurity.com/paros_proxy> [citado en 28 de noviembre de 2015].

[22] OWASP. Category: OWASP Webgoat Project. En <https://www.owasp.org/index.php/Category:OWASP_WebGoat_Project> [citado en 28 de noviembre de 2015].

[23] How to Geek. How to use wireshark to capture, filter and inspect packets. En <http://www.howtogeek.com/104278/how-to-use-wireshark-to-capture-filter-and-inspect-packets/> [citado en 28 de noviembre de 2015].

5.5. METODOLOGÍA DE DESARROLLO

El desarrollo del proyecto está orientado a cubrir los siguientes aspectos:

I. Estudio de las funcionalidades soportadas por los dispositivos WAF, esto con la finalidad de identificar las principales funciones que puede cumplir el WAF y de esta forma cubrir las vulnerabilidades de OWASP.
II. Revisión de las marcas y los servicios proporcionados por los diferentes tipos de WAF. Existen diferentes fabricantes dedicados a la producción de dispositivos de seguridad y más específicamente a los WAF, entre los más conocidos están: Imperva, F5, Citrix, Akamai, entre otros, y comúnmente el enfoque o la especialidad de estas marcas no es precisamente WAF, así que las funcionalidades o el modo de operación de sus WAF varían entre sí y el propositivo del estudio es identificar todas esas funcionalidades que puede ofrecer una mejora en la seguridad de los servicio web.
III. Diseño de un esquema de pruebas que permita evaluar las vulnerabilidades de OWASP, puesto que este tipo de trabajo tiene un enfoque práctico es necesario emular un escenario que permita incluir los 3 actores principales de este estudio como son: un servidor web vulnerable, un equipo cliente con herramientas para explotar vulnerabilidades y un WAF. Además, los dispositivos de ser puestos de forma que se permita realizar el cambio de los dos WAF (libre distribución y comercial) sin alterar las condiciones del servidor ni del cliente.
IV. Ejecución de los vectores de ataques sobre los WAF seleccionados, la ejecución de un ataque posee unas fases que para este proyecto no se cumplen en su totalidad, sin embargo, algunos pasos fueron considerados para el desarrollo del trabajo. Así que, en primera instancia es necesario realizar un reconocimiento de la vulnerabilidad a explotar, posteriormente con las herramientas de escaneo identificamos si es posible la ejecución del ataque y finalmente se lleva a cabo el vector de ataque seleccionado.
V. Realizar los análisis de resultados y conclusiones, considerando que luego de ejecutar los vectores de ataques sobre los dos tipos de WAF los resultados podrían diferir entre los tipos de dispositivos, por lo que llega a ser necesario identificar las ventajas y desventajas que presentan los WAF comerciales frente a los de libre distribución y validar si el WAF permitir minimizar el riesgo respecto a las vulnerabilidades reportadas en el TOP 10 de OWASP.

6. DISEÑO E IMPLEMENTACIÓN

6.1. PORTAL WEB DE PRUEBAS

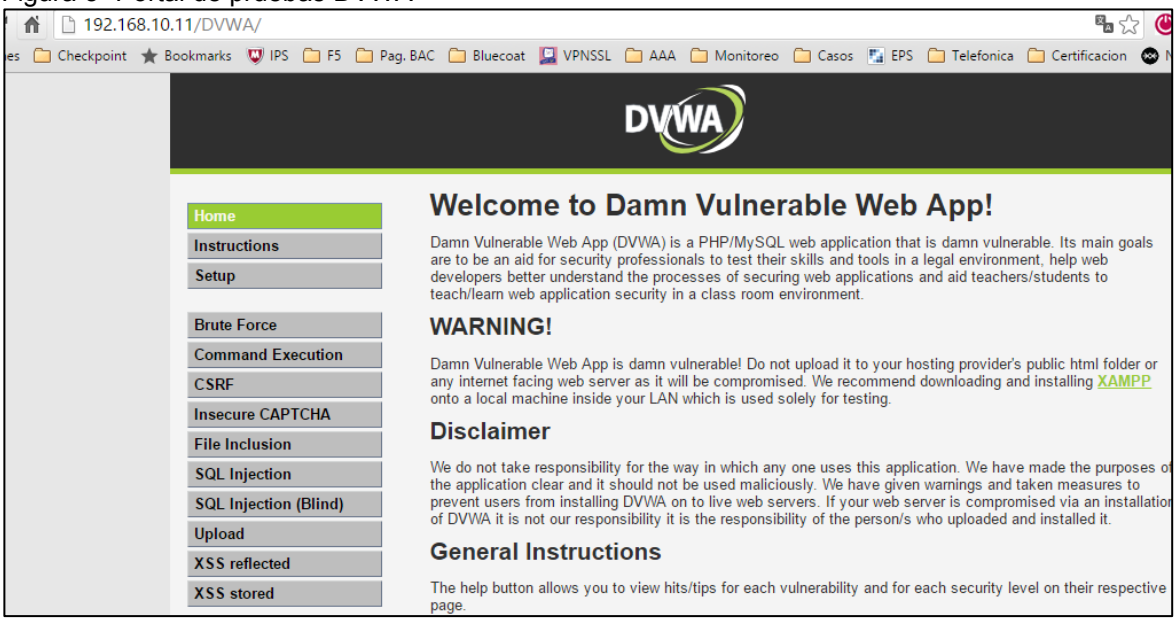
Mutillidae es un portal de libre distribución, deliberadamente vulnerable para ejecutar pruebas de aplicaciones web. Este software puede ser instalado en Linux o Windows usando LAMP, WMAP, XAMMP. Además, dispone de retos para explotar vulnerabilidades del top 10 de OWASP que incluyen la versión 2007, 2010 y 2013. En la Figura 4 se observa la página de inicio del portal.

Figura 4- Portal de pruebas Mutillidae

Fuente: El autor

El otro portal empleado para ejecutar las pruebas es Damn Vulnerable Web App (DVWA) el cual es una página web con vulnerabilidades implementada sobre PHP/MySQL. Su principal propósito es entrenar a los profesionales en seguridad a desarrollar sus habilidades y probar herramientas en un ambiente controlado sin incumplir con las normativas legales. Este portal también dispone de una serie de retos para el hallazgo de vulnerabilidades, además posee tres niveles de seguridad que exige ejecutar técnica más avanzadas con cada nivel. En la Figura 5 se muestra la página principal del portal.

Figura 5- Portal de pruebas DVWA



Fuente: El autor

6.2. FICHA TÉCNICA DE LOS FIREWALL DE APLICACIONE WEB

6.2.1. Información ModSecurity ModSecurity es una herramienta de libre distribución, que ofrece un WAF multiplataforma para Apache, IIS, Ngnix. Este software es desarrollado por los laboratorios Spider de la empresa Trustwave cuenta con un lenguaje de programación basado en eventos proporcionando protección frente a un rango de ataques contra aplicaciones web y permite monitorear, registrar y analizar en tiempo real el tráfico HTTP.

El lema de ModSecurity es la libertad para escoger lo que se quiere hacer que va muy relacionado con su naturaleza de libre distribución. ModSecurity es una herramienta con total acceso al código fuente de su aplicación, lo cual permite extender las habilidades y personalizar las reglas de acuerdo a las necesidades.

Los modos de implementación de ModSecurity son dos: embebido y proxy reverso, donde cada uno presenta sus ventajas y desventajas:

- La función de embebido es soportada como módulo de Apache para las versiones 2.0.x y 2.2.x, y no puede ser embebido con otro tipo de servidores web como IIS o Ngnix. Esta solución es idónea si no se desean realizar cambios en la infraestructura o adicionar nuevos puntos de falla, sin embargo la mayor desventaja es compartir los mismos recursos del servidor web con modsecurity o desplegar está solución en un gran número de servidores.

- En la función de proxy reverso, el WAF actúa como enrutador http diseñado para estar entre el servidor web y el cliente, cuando se despliega en este modo es requerido la instalación de un servidor web con Apache y con el módulo de ModSecurity. De esta forma, un solo dispositivo puede proteger cientos de servidores web. El modo embebido no incluyen nuevos puntos de falla, pero no es escale cuando se dispone de un número grande de servidores web.

Aunque Modsecurity es una buena herramienta, existen algunas funcionalidades que se debe ser mejoras, una de ellas es disponer de un método de aprendizaje para disminuir el número de falsos positivos y segundo la implementación del modo embebido en otros tipos de servidores web.

6.2.2. Información F5 BIG-IP F5 es una de las marcas líderes en el mercado de balanceo de tráfico lo cual trae beneficios con la integración de soluciones que disponen de varios servidores para ofrecer un servicio, tal como pasa en la mayoría de soluciones web. Mediante su producto BIG-IP, F5 ofrece diferentes tipos de servicios como lo es: gestor de tráfico global (GTM, en inglés), gestor de tráfico local (LTM, en inglés), gestor de firewall avanzado (AFM, en inglés), gestor de seguridad de aplicaciones (ASM, en inglés), entre otros. La licencia relacionada con WAF es la nombrada como ASM, la cual permite defensa ante ataque denegación de servicio a nivel de capa de aplicación, implementación de técnicas avanzadas de detección y mitigación, aprendizaje dinámico y visibilidad granular de ataques. Asimismo, tiene cumplimiento de normativas tales como FFIEC, HIPAA y PCI-DSS.

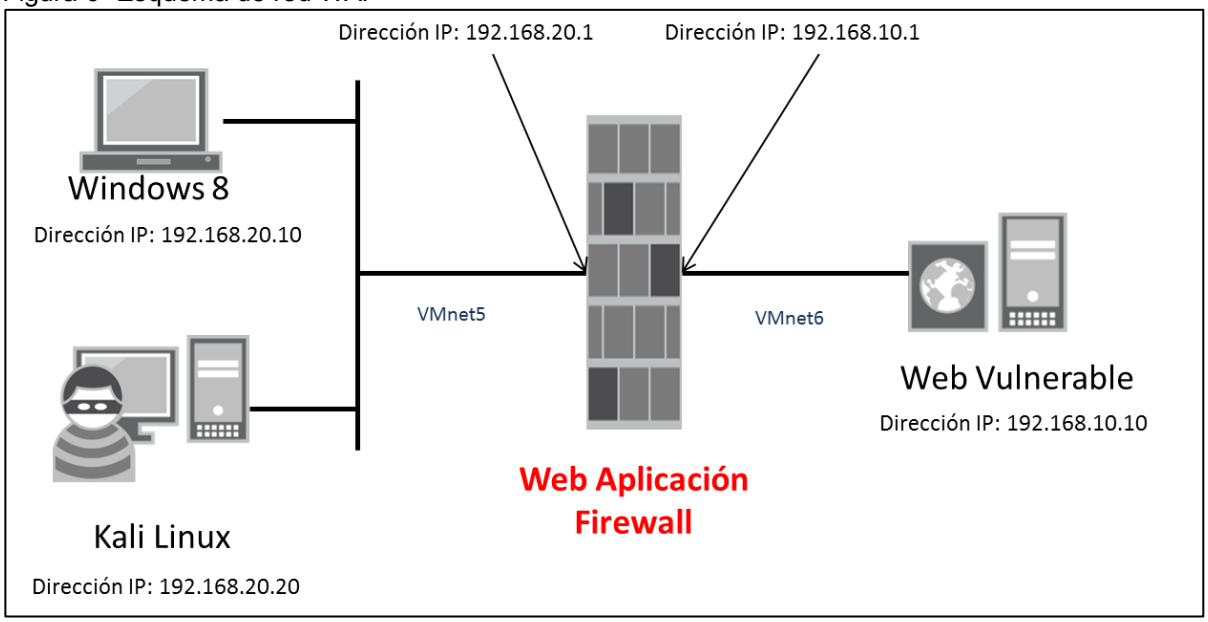
La implementación del WAF F5 BIG-IP en una edición virtual requiere llevar a cabo los siguientes pasos:

I. Obtener una licencia demo, la cual tiene un valor 95USD y brinda una vigencia de 90 días, se puede ordenar a través de internet o mediante un canal de F5.
II. Crear una cuenta de F5, para esto no es necesario comprar un producto de la marca.
III. Descargar la imagen de la edición virtual de BIG-IP, esto se realiza a través de la cuenta de F5 ingresando a la URL https://downloads.f5.com. Allí se deben descargar las versiones con extensión ZIP u OVA que son las compatibles con VM Ware.
IV. Realizar el montaje de la imagen en VM Ware
V. Licenciar el dispositivo, tan pronto como se inicia por primera vez la máquina virtual le permitirá al usuario ingresar las credencias que por defecto son root tanto para usuario y contraseña, sin embargo después de esto el dispositivo no permitirá realizar ninguna configuración adicional hasta que sea cargada la licencia.

6.3. ESQUEMA DE LA RED DE PRUEBAS

Para el uso del WAF se utilizará un modo de operación en proxy reverso, actuando con una puerta de enlace para conectar la red de servidor o con la de los clientes. Para ejecutar las pruebas se empleará dos equipos clientes que incluyen un PC con Windows 8.1 y equipo con Kali Linux. En Figura 6 se detalla el esquema de red.

Figura 6- Esquema de red WAF



Fuente: El autor

6.3.1. Configuración WAF ModSecurity en proxy reverso Al configurar el servidor Apache en modo proxy reverso, será necesario disponer de dos servidores uno brindando la funcionalidad de WAF y el otro encargado de publicar la aplicación web. Para iniciar la configuración del WAF primero se deben habilitar la configuración del proxy reverso, para esto primero se cargan las dependencias necesarias mediante el siguiente comando:

apt-get install libapache2-mod-security2 –y

Luego se deben habilitar los modulos de proxy reverso:

a2enmod headers
a2enmod mod-security
a2enmod security2

Para validar que modulos se encuentran habilitados en el servicio de Apache se emplea el comando “apache2ctl –M”

Para finalizar la configuración del proxy reverso se debe editar el archivo "/etc/apache2/sites-enabled/000-default.conf" allí se define la IP del web server hacia donde se redireccionarán las solicitudes hechas por los usuarios para este caso será la 192.168.10.10 la cual corresponde al web server. En la Figura 7 se muestra la configuración realizada para el proxy reverso.

Figura 7- Configuración proxy reverso Apache

```
root@ubuntu:/home/tom# cat /etc/apache2/sites-enabled/000-default.conf
<VirtualHost *:80>
        ProxyPreserveHost On

        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf
        ProxyPass / http://192.168.10.10:80/
        ProxyPassReverse / http://192.168.10.10:80/


</VirtualHost>
```

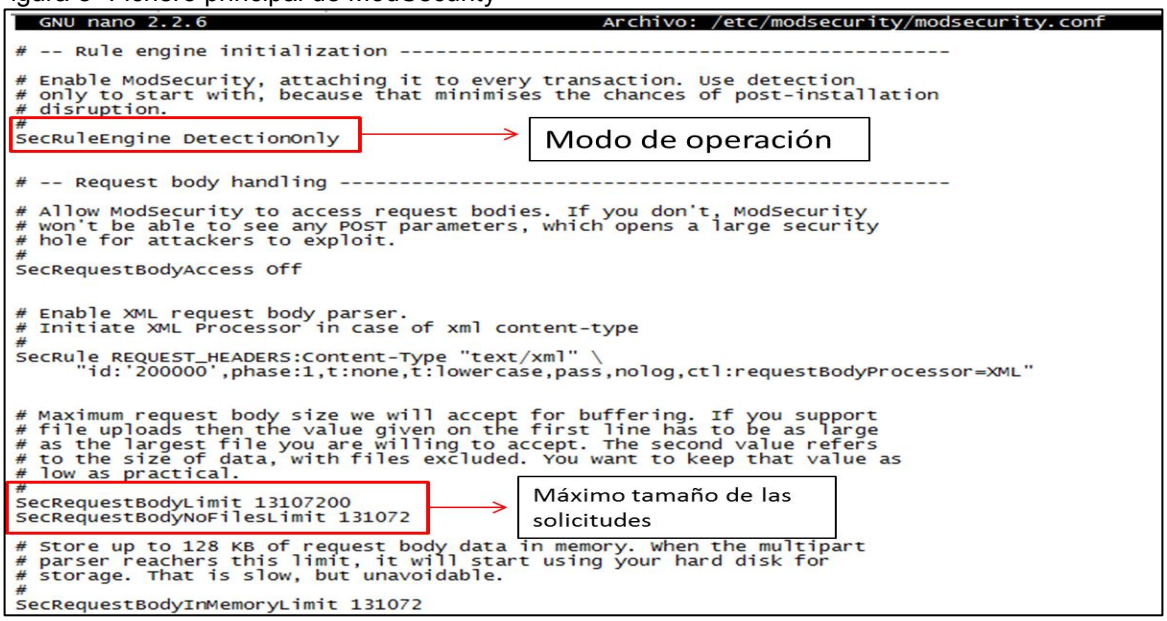
Fuente: El autor

Para instalar el Modsecurity en el servidor Apache será necesario agregar las siguientes dependencias:

sudo apt-get install libapache2-mod-security2 –y

Al terminar la instalación se ingresa al archivo "/etc/modsecurty/modsecurity.conf", el cual corresponde al fichero principal de la configuración del Modsecurity, allí se define el modo de operación del WAF si va a realizar bloqueos o solo detección, además se fijan los tamaños máximos que puede llegar a tener una solicitud HTTP. En Figura 8 se detalla la configuración del archivo.

Figura 8- Fichero principal de ModSecurity



Fuente: El autor

Luego se cargan las librerías y reglas preconfiguradas que se descargan desde https://github.com/SpiderLabs/owasp-modsecurity-crs/tarball/master, están reglas se incluyen en el archivo que se muestra en Figura 9.

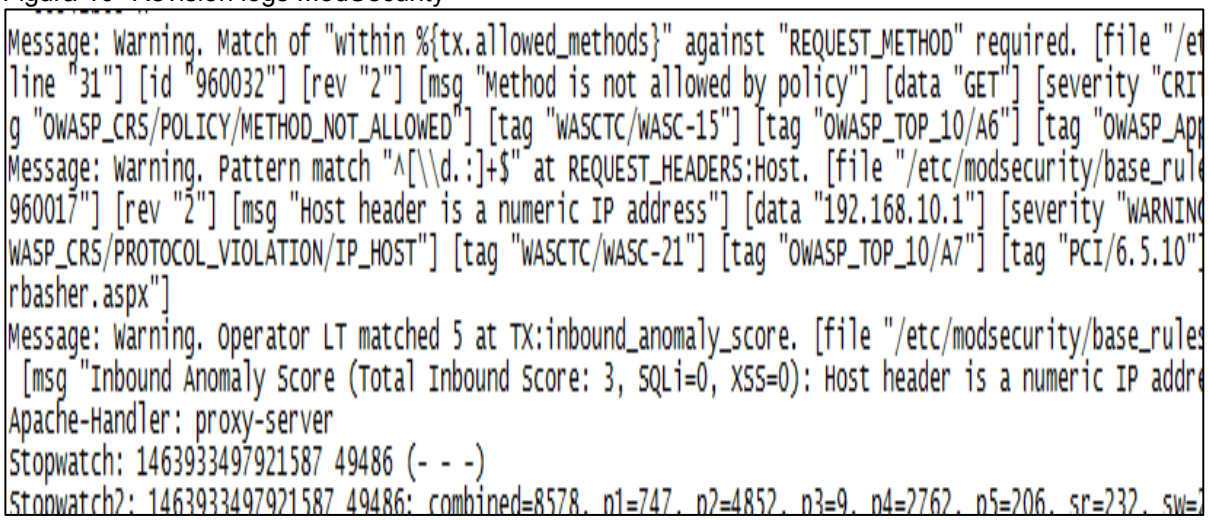
Figura 9- Archivo de configuración ModSecurity

```
root@ubuntu:/home/tom# cat  /etc/apache2/mods-enabled/security2.conf
<IfModule security2_module>
        # Default Debian dir for modsecurity's persistent data
        SecDataDir /var/cache/modsecurity

        # Include all the *.conf files in /etc/modsecurity.
        # Keeping your local configuration in that directory
        # will allow for an easy upgrade of THIS file and
        # make your life easier
        Include "/etc/modsecurity/activated_rules/*.conf"
        IncludeOptional /etc/modsecurity/*.conf
        # IncludeOptional "/usr/share/modsecurity-crs/*.conf"
        # IncludeOptional "/usr/share/modsecurity-crs/activated_rules/*.conf"
</IfModule>
```

Fuente: El autor

Finalmente, para validar si el tráfico está cursando por el WAF se debe validar el fichero “/var/log/apache2/error.log”, el cual mostrará si existe algún tipo de bloqueo o restricción en el WAF de ModSecurity, en la Figura 10 se detalla muestra de error detectado por el WAF.

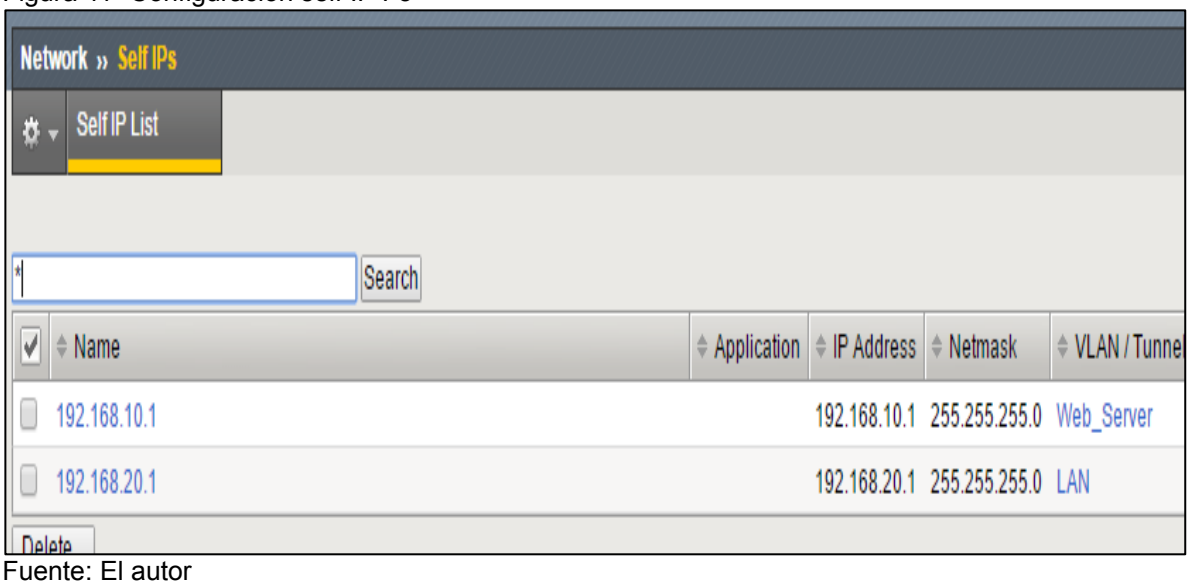
Figura 10- Revisión logs ModSecurity



Fuente: El autor

6.3.2. Configuración WAF F5 en proxy reverso La configuración del proxy reverso en F5 requiere configurar las interfaces de red en el dispositivo, para este caso se tendrán dos IP una para LAN y otra para la conexión con el servidor. La configuración del direccionamiento de red se realiza sobre el modulo network>selfip del dispositivo, tal como se observa en Figura 11.

Figura 11- Configuración self IP F5

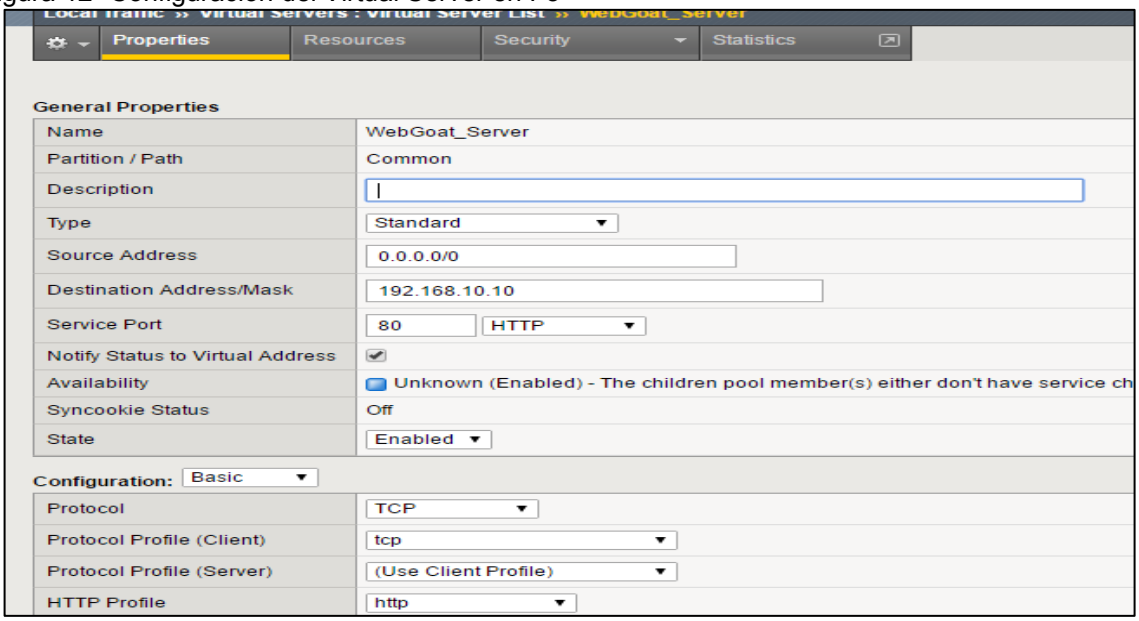


Fuente: El autor

El siguiente paso es crear el Virtual Server, el cual es un objeto en el equipo que representa una IP y un puerto de servicio, este objeto es usado para que los clientes externos hagan consultas al web server. En el caso del laboratorio el virtual server está asociado a la IP 192.168.10.10 y puerto 80 (http), este será el

objeto consultado para que el tráfico que ingrese por el f5. En la Figura 12 se detalla la configuración hecha para el virtual server.

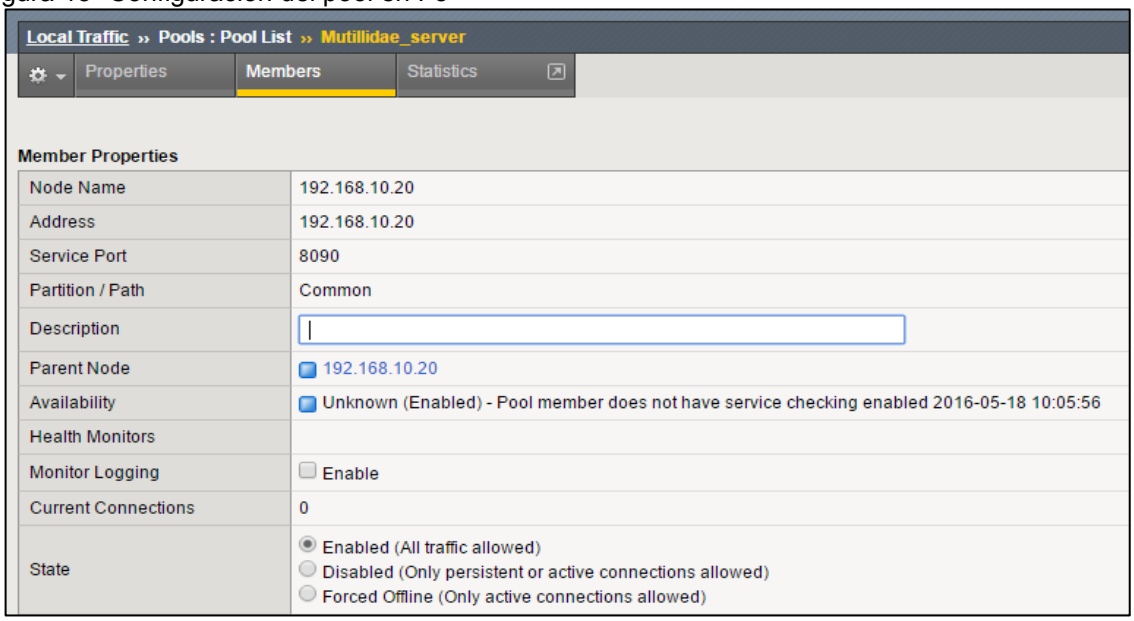
Figura 12- Configuración del Virtual Server en F5



Fuente: El autor

El flujo de tráfico en un F5 debe recibir las peticiones de los clientes externos a través del virtual server para posteriormente redireccionar la petición hacia el Pool asociado. Para el caso del laboratorio las consultas son hechas a la IP 192.168.10.10 y puerto 80 (http) y el F5 redireccionará el tráfico hacia la IP 192.168.10.20 la cual corresponde a la dirección real de nuestro servidor web. En Figura 13 se detalla la configuración del Pool.

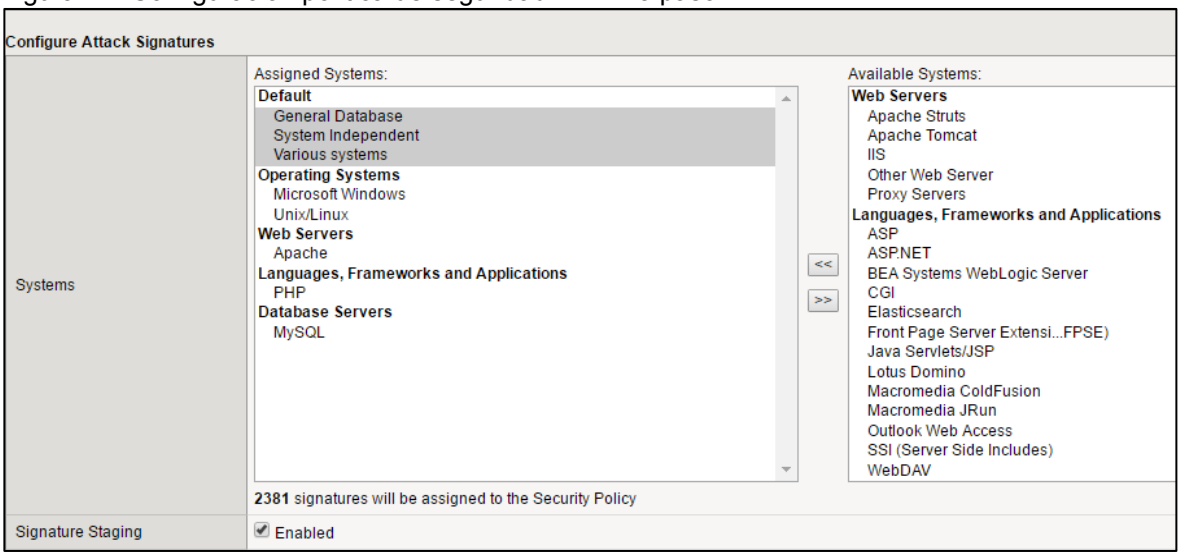
Figura 13- Configuración del pool en F5



Fuente: El autor

Posteriormente, se requiere realizar la configuración de una política de seguridad en ella se debe suministrar la información relacionada con el servidor web como: sistema operativo, lenguaje de programación, framework y base de datos. El suministrar esta información describe la política de seguridad de forma que solo alertará las firmas de ataques relacionadas con las características del servicio web instalado. En la Figura 14 se muestra la configuración hecha para el laboratorio ejecutado.

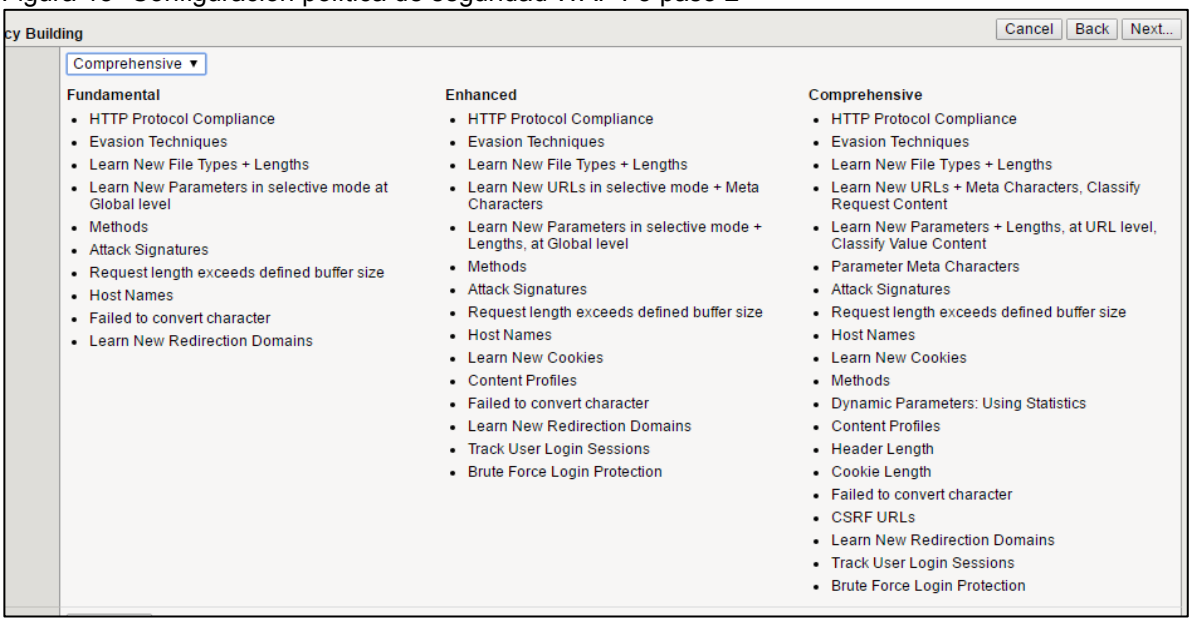
Figura 14- Configuración política de seguridad WAF F5 paso 1



Fuente: El autor

En el siguiente paso de la configuración se define lo complejidad que de aprendizaje que puede llegar a tener la política, va desde fundamental para la detección de los vulnerabilidades más sencillas hasta una política comprensiva que permite detectar comportamientos de las aplicaciones. Para las pruebas se utilizará la política de nivel comprensivo. En la Figura 15 se muestran los tres tipos de políticas.

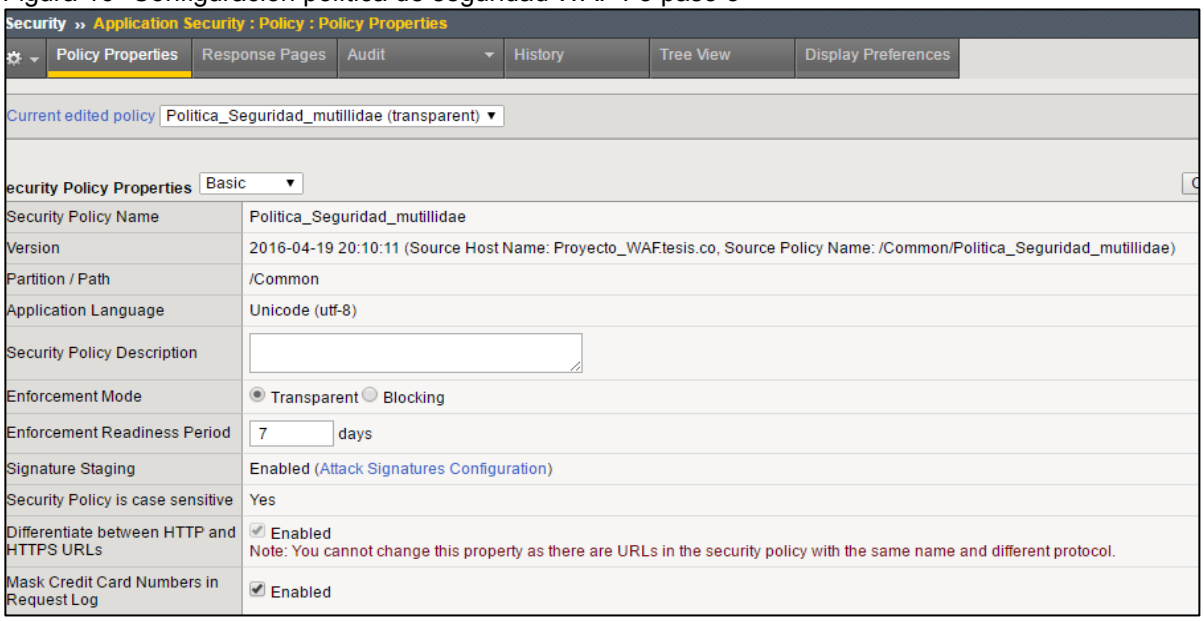
Figura 15- Configuración política de seguridad WAF F5 paso 2



Fuente: El autor

En el último paso de la configuración el equipo propone un periodo de aprendizaje de parámetros de la aplicación de 7 días, y permite configurar al dispositivo en modo de alerta o alerta y bloqueo, que se identifican como modo transparente o bloqueo. Para las pruebas de empleará un modo transparente. En la Figura 16 se muestran los detalles de la configuración.

Figura 16- Configuración política de seguridad WAF F5 paso 3



Fuente: El autor

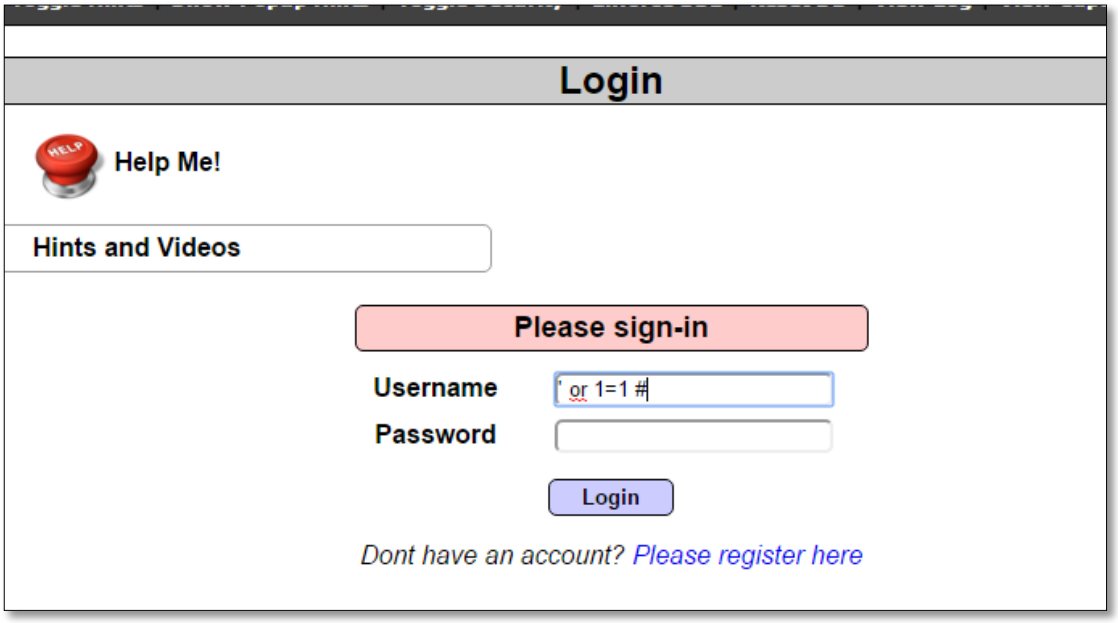
7. ANALISIS Y RESULTADOS

En el esquema de pruebas empleado se evaluaron los dos tipos de WAF tanto el comercial de la marca F5 y el de libre distribución ModSecurity, cada WAF fue sometido a las 10 vulnerabilidades una por cada ámbito del top 10 de OWASP de 2013 y los resultados evaluados se enfocan en la detección de los parámetros anómalos por parte del WAF.

7.1. PRUEBA OWASP A1: INYECCIÓN

Para la prueba de inyección de SQL se buscó saltar la autenticación del portal para esto se introdujo la variable ‘ or 1=1 # en el campo de usuario, al enviar esta solicitud una página vulnerable detectará esta afirmación como cierta evitará validar las credenciales, como se observa en la Figura 17.

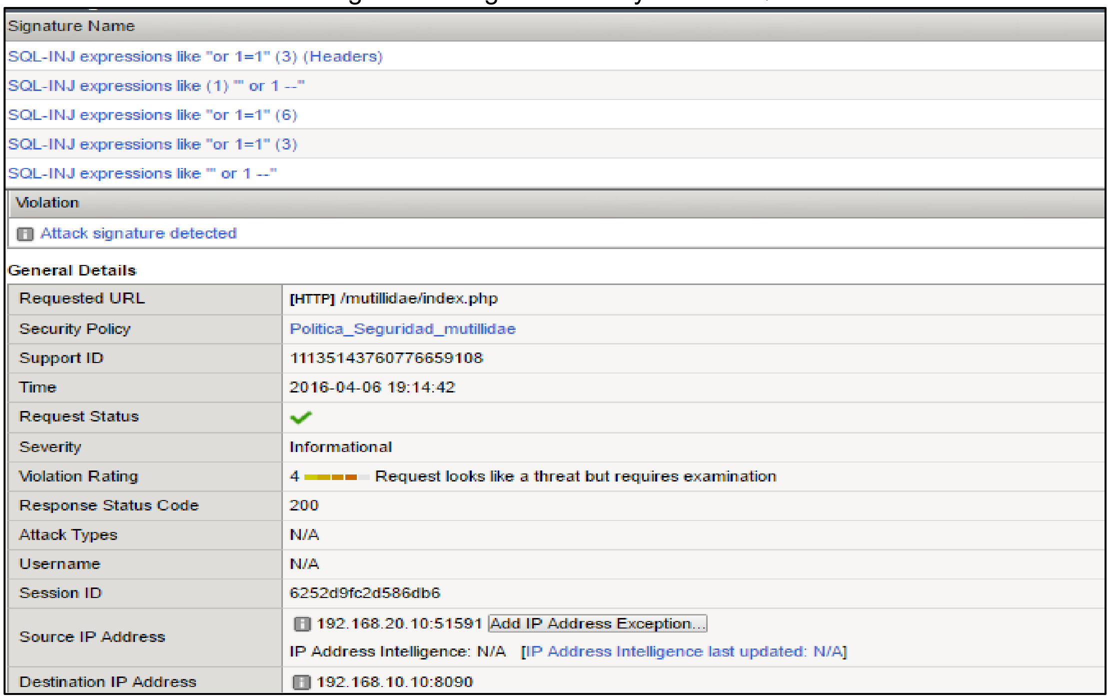
Figura 17- Prueba A1 Inyección de SQL



Fuente: El autor

En el equipo WAF de marca F5 luego de introducir la secuencia de inyección de SQL se detecta el ataque, donde la IP origen que generó el evento fue 192.168.20.10 y el nivel de violación fue calificado de nivel 4 en una escala de 1 a 5. En la Figura 18 mayor detalle del evento detectado por el WAF de F5.

Figura 18- Log WAF F5 Inyección SQL

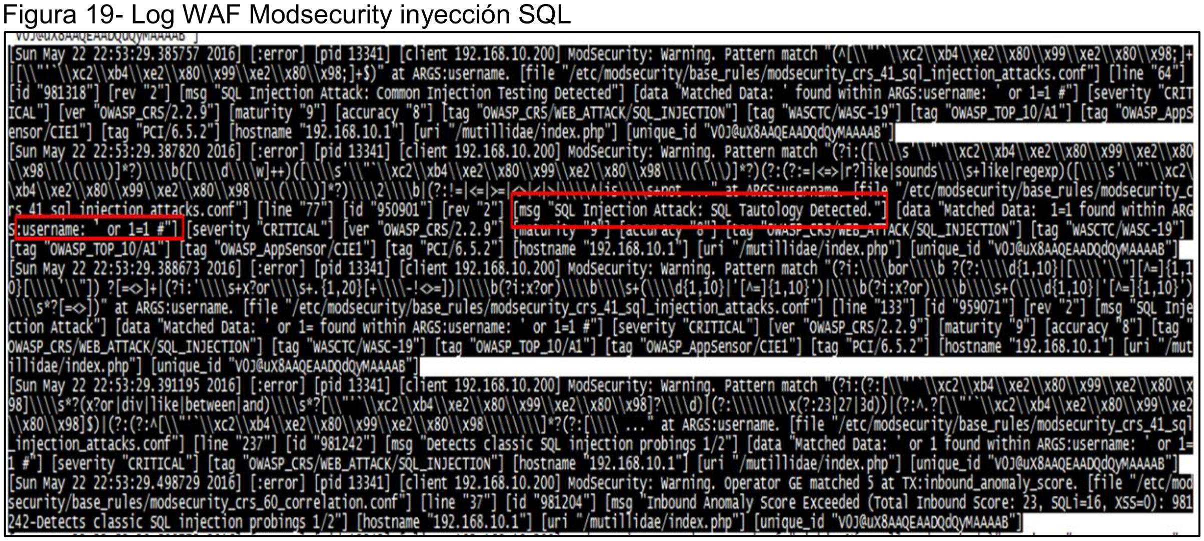


Signature Name	
SQL-INJ expressions like "or 1=1" (3) (Headers)	
SQL-INJ expressions like (1) "' or 1 --"	
SQL-INJ expressions like "or 1=1" (6)	
SQL-INJ expressions like "or 1=1" (3)	
SQL-INJ expressions like "' or 1 --"	
Violation	
Attack signature detected	
General Details	
Requested URL	[HTTP] /mutillidae/index.php
Security Policy	Politica_Seguridad_mutillidae
Support ID	11135143760776659108
Time	2016-04-06 19:14:42
Request Status	✔
Severity	Informational
Violation Rating	4 Request looks like a threat but requires examination
Response Status Code	200
Attack Types	N/A
Username	N/A
Session ID	6252d9fc2d586db6
Source IP Address	192.168.20.10:51591 Add IP Address Exception... IP Address Intelligence: N/A [IP Address Intelligence last updated: N/A]
Destination IP Address	192.168.10.10:8090

Fuente: El autor

Posteriormente, al realizar la inyección de SQL en el WAF de ModSecurity, se detecta en los logs que el campo de usuario se introdujo ‘or 1=1 #’ lo cual corresponde al patrón de comportamiento de una ataque de inyección. En Figura 19 se detalla la información revisada en el WAF de ModSecurity.

Figura 19- Log WAF Modsecurity inyección SQL

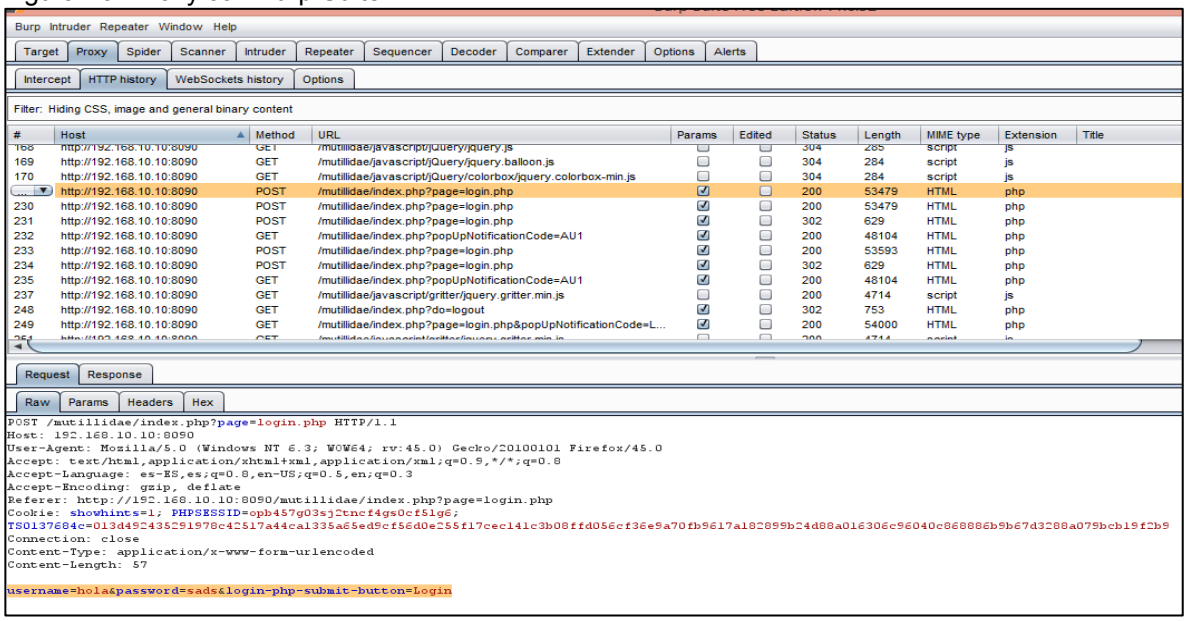


Fuente: El autor

7.2. PRUEBA OWASP A2: PERDIDA DE AUTENTICACIÓN

La pérdida de autenticación o gestión de sesiones mediante el portal Metillidae invita a realizar una prueba de fuerza bruta en el portal de autenticación de la página. Para esta prueba se hace uso de la herramienta Burp Suite, la cual permite interceptar el tráfico operando de modo proxy entre el navegador y el portal web, cuando la interceptación del tráfico se efectúa de forma adecuada se puede analizar toda la información ingresada por el cliente en el navegador. En el ataque de fuerza bruta la herramienta Burp Suite fue empleada para identificar la sintaxis de las solicitudes que se hace desde el navegador. En la Figura 20 se observa la información enviada al momento de la autenticación.

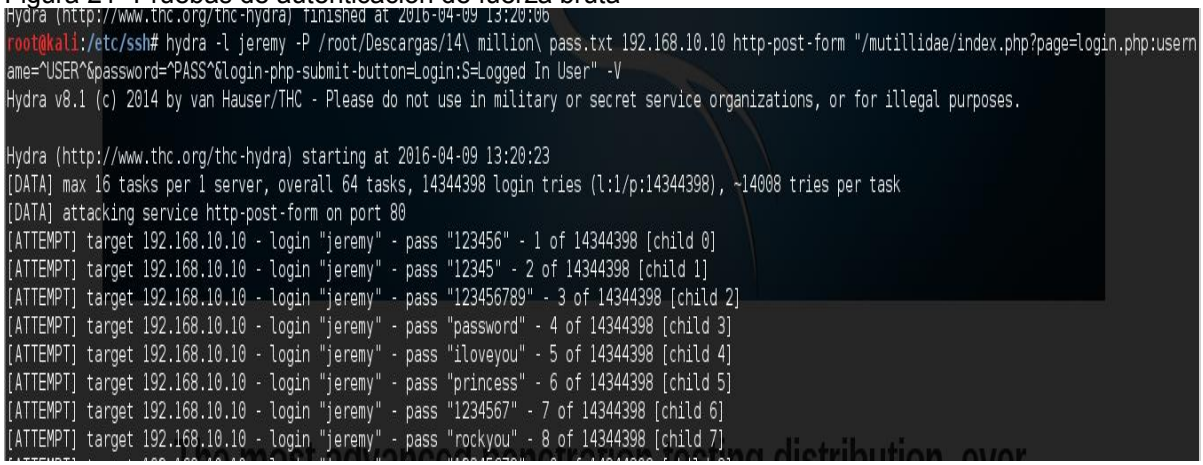
Figura 20- Proxy con Burp Suite



Fuente: El autor

Luego, con la ayuda de la herramienta Hydra se buscará encontrar la contraseña para el usuario Jeremy, para esto se debe suministrar la URL donde se realiza el login, además del formato del usuario y contraseña, información obtenida en el paso anterior. En la Figura 21 se muestra el comando ejecutado sobre la herramienta Hydra para realizar el ataque de fuerza bruta.

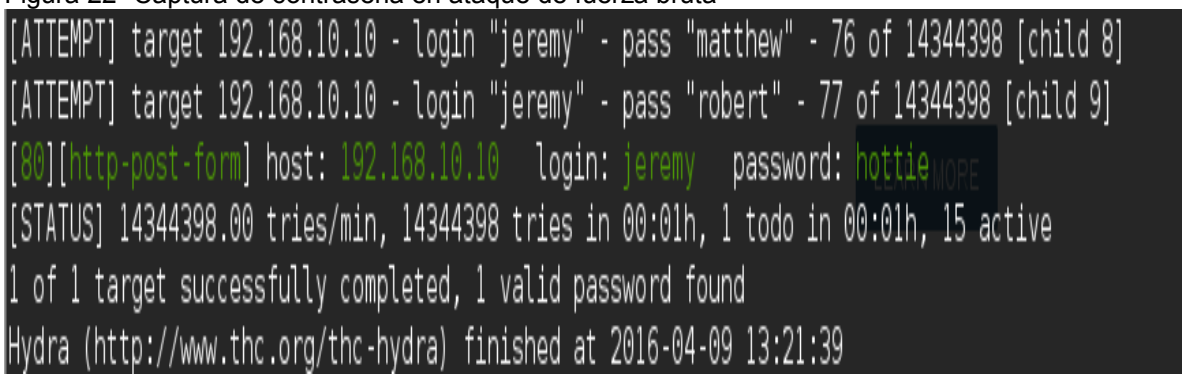
Figura 21- Pruebas de autenticación de fuerza bruta



Fuente: El autor

Los resultados del ataque de fuerza bruta se observan en la Figura 22, allí se indica que la contraseña encontrada para el usuario Jeremy fue hottie.

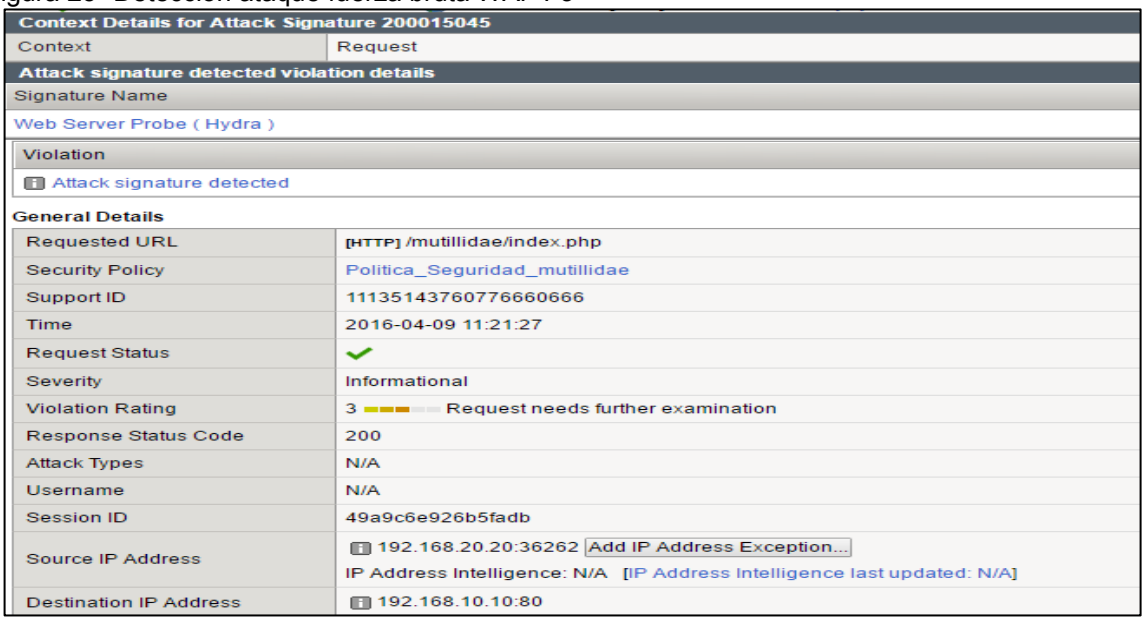
Figura 22- Captura de contraseña en ataque de fuerza bruta



Fuente: El autor

En el waf de marca F5 no se detectó que se estuviera presentado un ataque de fuerza bruta propiamente, sin embargo en sus firmas si se reporta que se está haciendo uso de la herramienta Hydra lo cual generaría un bloqueo en el WAF, ver Figura 23.

Figura 23- Detección ataque fuerza bruta WAF F5



Context Details for Attack Signature 200015045

Context	Request

Attack signature detected violation details

Signature Name

Web Server Probe (Hydra)

Violation

Attack signature detected

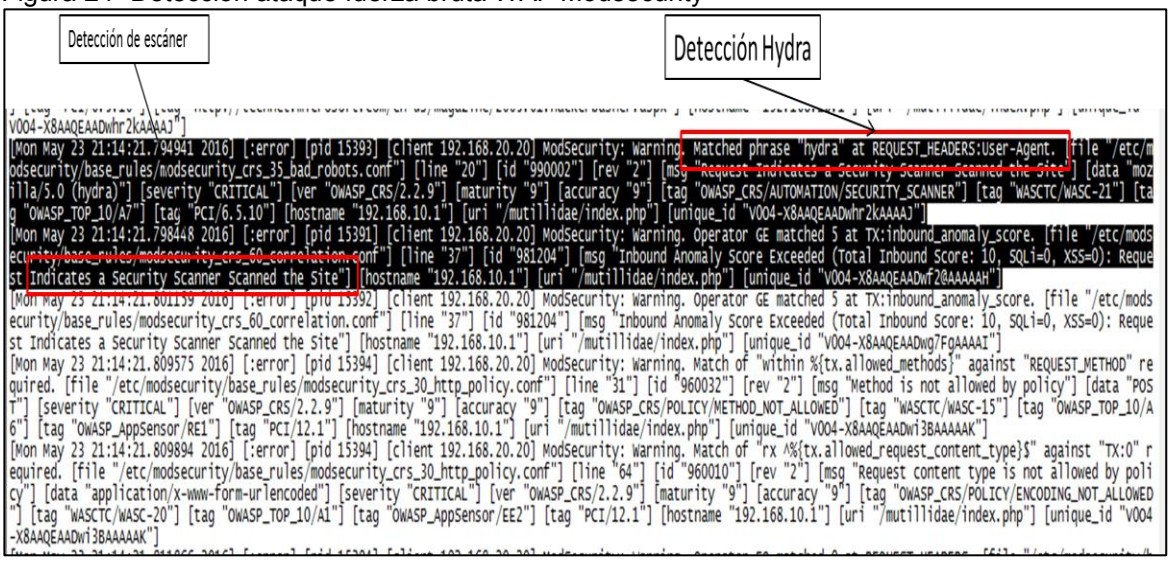
General Details

Requested URL	[HTTP] /mutillidae/index.php
Security Policy	Politica_Seguridad_mutillidae
Support ID	11135143760776660666
Time	2016-04-09 11:21:27
Request Status	✔
Severity	Informational
Violation Rating	3 Request needs further examination
Response Status Code	200
Attack Types	N/A
Username	N/A
Session ID	49a9c6e926b5fadb
Source IP Address	192.168.20.20:36262 Add IP Address Exception... IP Address Intelligence: N/A [IP Address Intelligence last updated: N/A]
Destination IP Address	192.168.10.10:80

Fuente: El autor

En el WAF Modsecurity, se detectan dos eventos durante el ataque de fuerza bruta, el primero indica que se está haciendo uso de Hydra reportado como un software anómalo, pero además se indica que se está realizando un escaneo sobre el sitio. En la Figura 24 más detalles de la información detectado por el WAF de ModSecurity.

Figura 24- Detección ataque fuerza bruta WAF Modsecurity

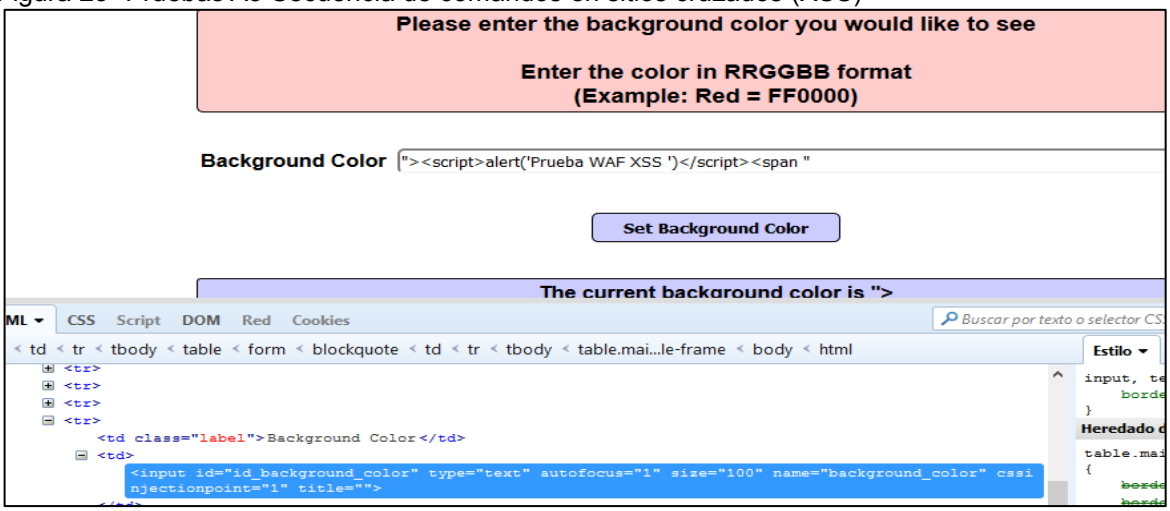


Fuente: El autor

7.3. PRUEBA OWASP A3: XSS

Para las pruebas de secuencia de comandos de sitios cruzados, en el portal Mutillidae se presenta un campo que permite ingresar un valor hexadecimal con el fin de cambiar el fondo de la pantalla del portal. Allí se ingresa el comando "><script>alert('Prueba WAF XSS')</script><span " el cual es un comando de XSS reflejado, ver figura Figura 25.

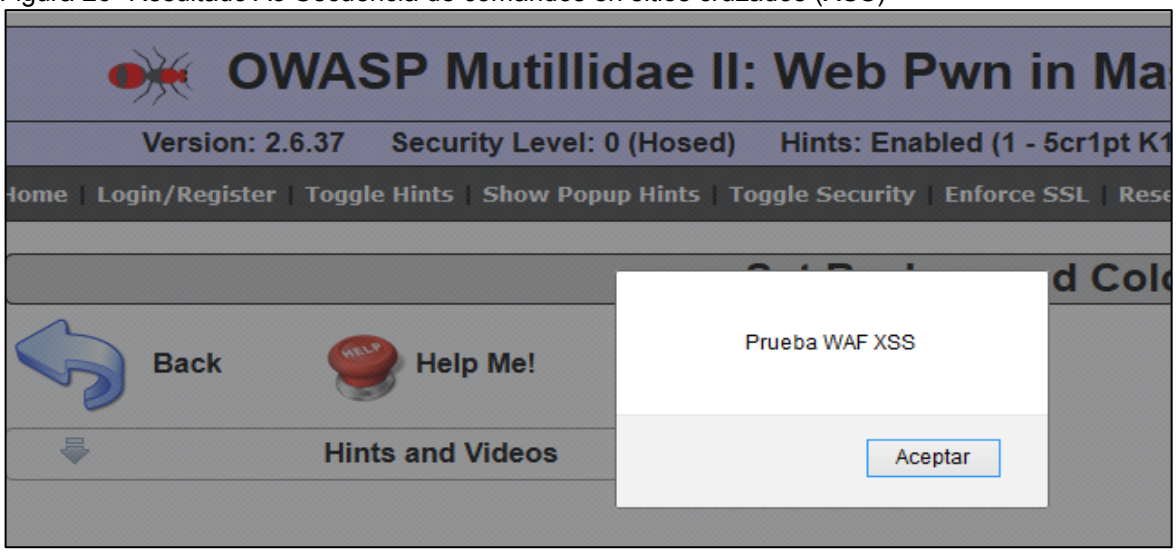
Figura 25- Pruebas A3 Secuencia de comandos en sitios cruzados (XSS)



Fuente: El autor

En Figura 26 se observa el resultado obtenido luego de ingresar el comando de XSS, lo cual corresponde a una ventana emergente en el navegador.

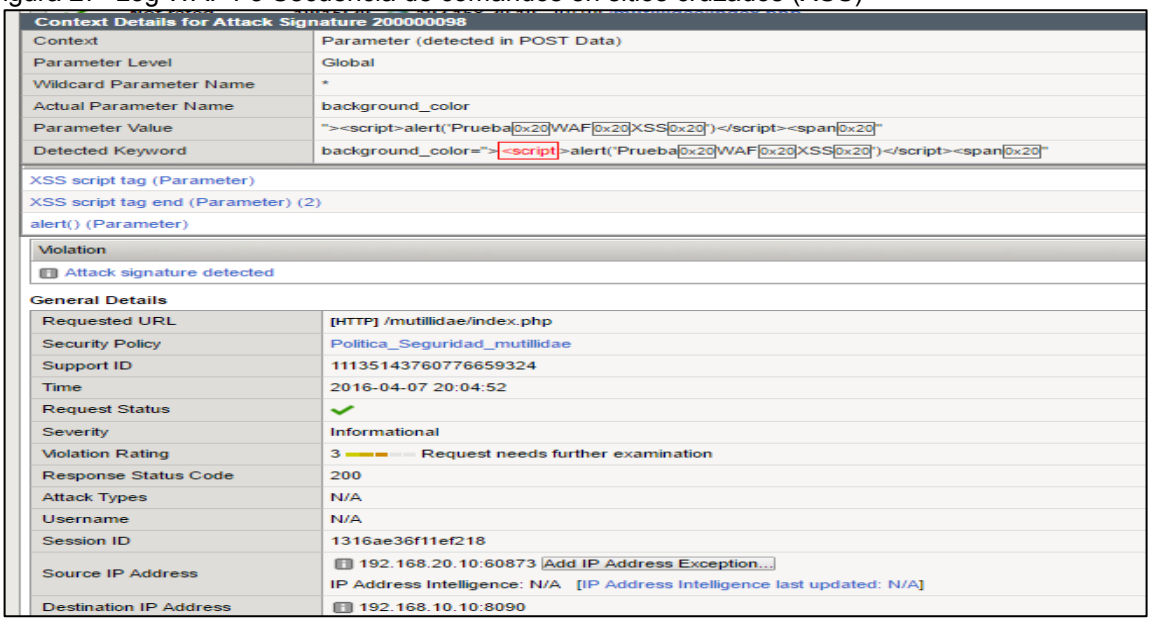
Figura 26- Resultado A3 Secuencia de comandos en sitios cruzados (XSS)



Fuente: El autor

Al revisar en el WAF de marca F5 se detecta 3 eventos, dos relacionados con XSS catalogados como nivel 3, y un último evento relacionado con el uso del parámetro alert(). En la Figura 27 se detalla la información detectada por el WAF de marca F5.

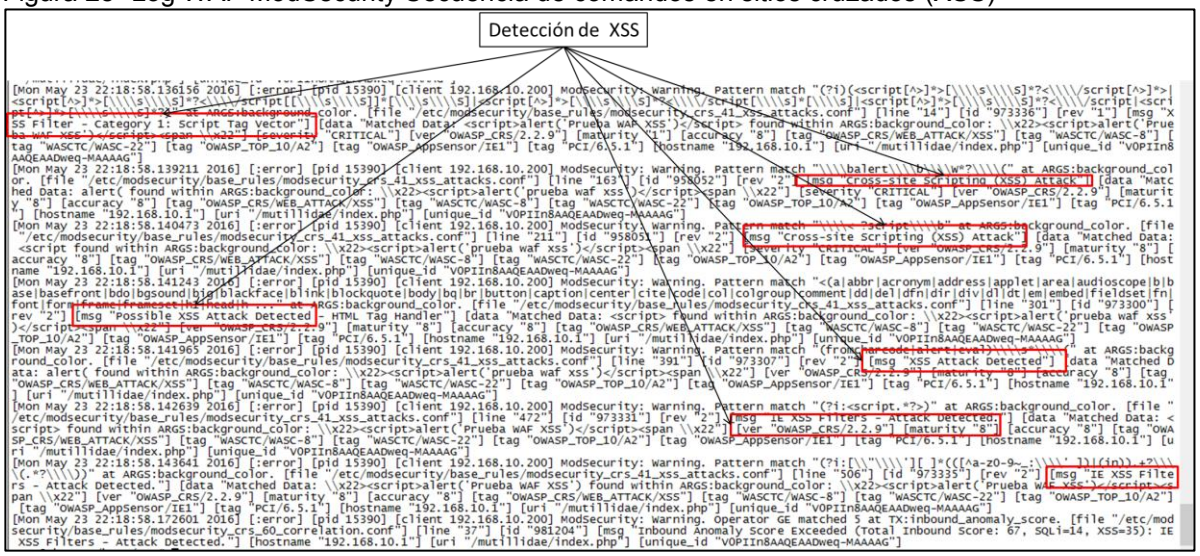
Figura 27- Log WAF F5 Secuencia de comandos en sitios cruzados (XSS)



Fuente: El autor

En la revisión hecha en el WAF Modsecurity se detectaron 7 eventos relacionados con XSS referente a la prueba ejecutada, se observó que se evalúan diferentes patrones para identificar este tipo de ataques como son la revisión de caracteres o palabras restringidas como es “alert” o “script”, además se identifica que se generó una ventana emergente con el navegador Internet Explorer (IE). En la Figura 28 mayor detalla del log de XSS en ModSecurity.

Figura 28- Log WAF ModSecurity Secuencia de comandos en sitios cruzados (XSS)

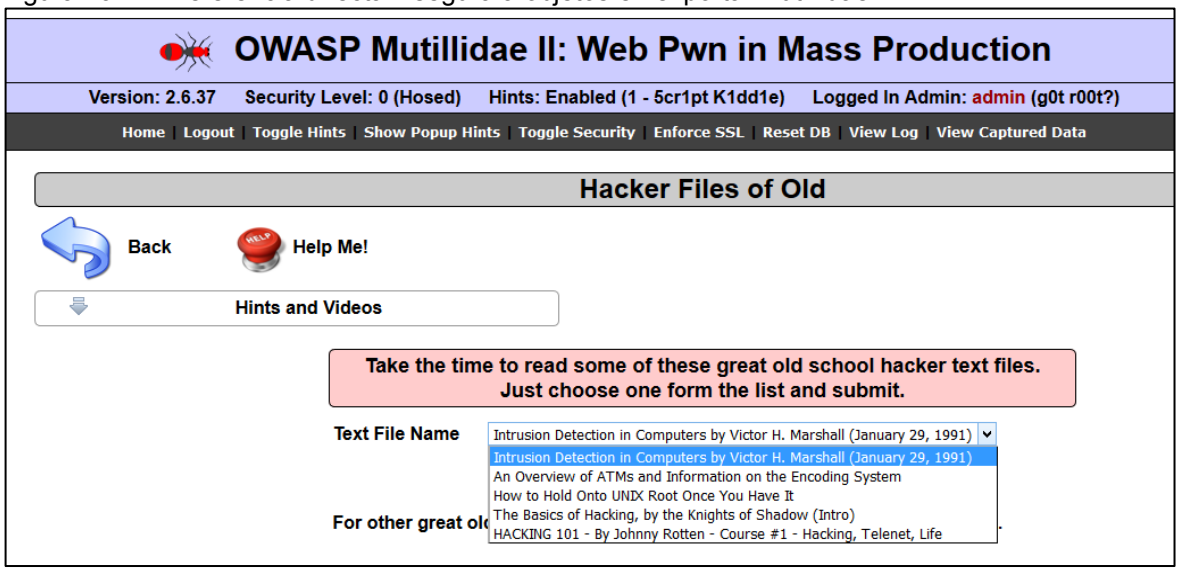


Fuente: El autor

7.4. PRUEBA OWASP A4: REFERENCIA DIRECTA INSEGURA A OBJETOS

En la referencia directa insegura a objetos en el portal de pruebas el escenario que se plantea es visualizar una lista de ficheros de texto, sin embargo mediante esta vulnerabilidad es posible visualizarse otros ficheros de configuración que no estén presentes en la lista, en la Figura 29 se detalla el contenido disponible para la prueba.

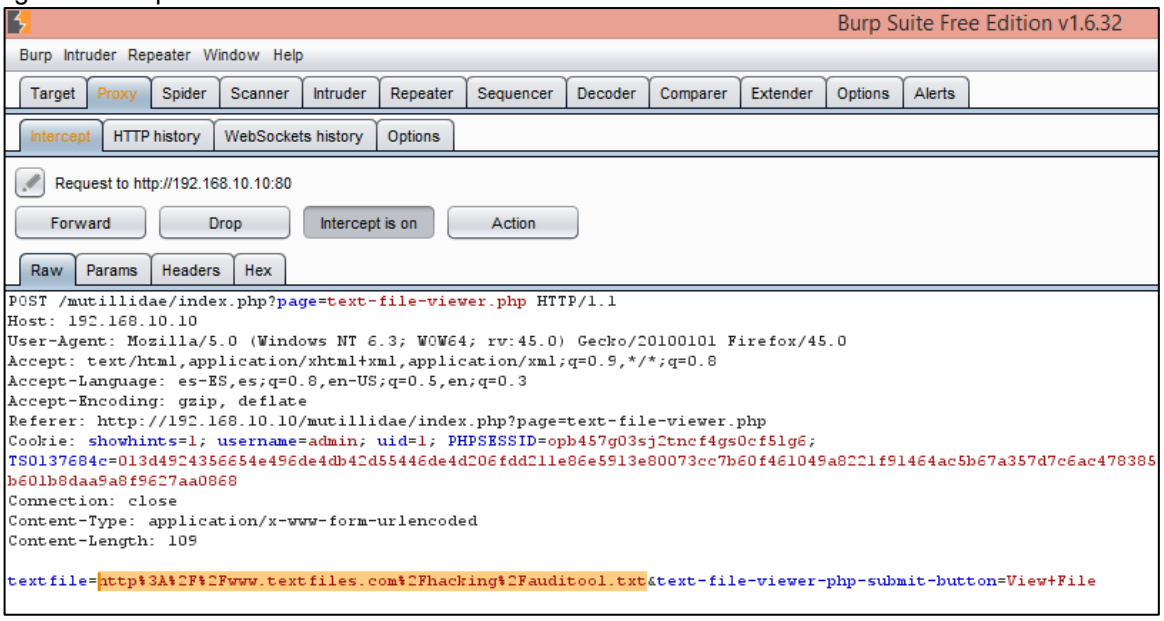
Figura 29- A4 Referencia directa insegura a objetos en el portal mutillidae



Fuente: El autor

En la Figura 30 mediante la herramienta Burp Suite se captura la solicitud realizada al tratar de visualizar los ficheros expuestos en la página, dentro de la solicitud viaja una variable denominada textfile en la cual se define el archivo que se desea visualizar y es allí donde se realiza la alteración al contenido de la página.

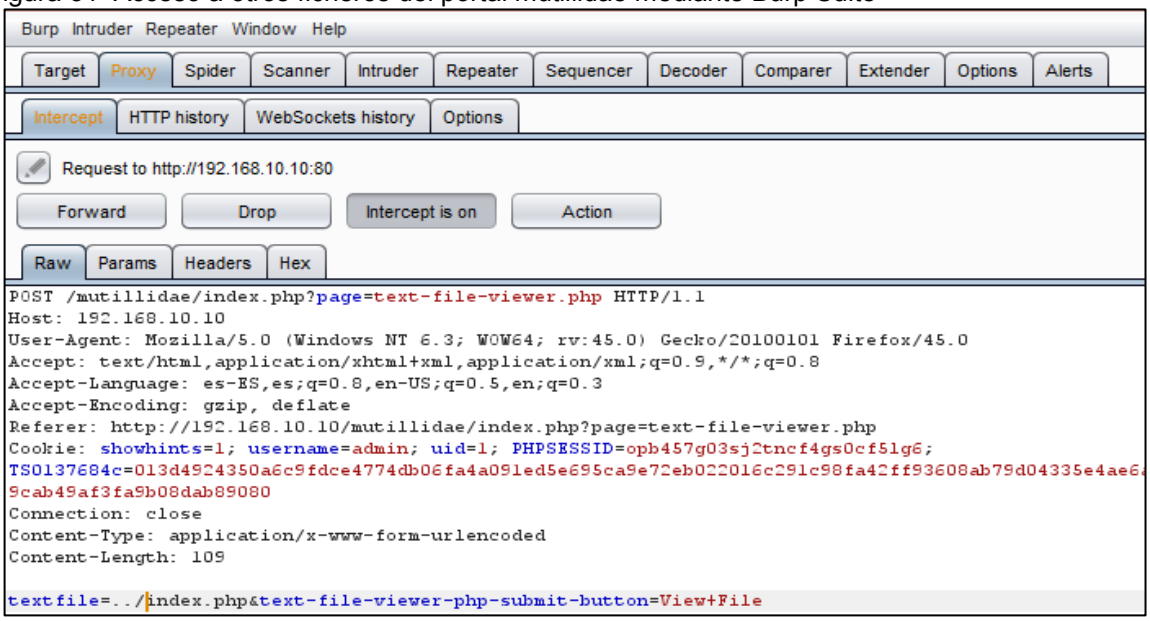
Figura 30- Captura de la visualización de archivo de texto



Fuente: El autor

En Figura 31 se intenta explorar otra carpeta de configuración dentro del servidor al dar la instrucción "../" e ingresar al fichero llamado index.php, el cual no está dentro de los listado en el menú inicial presentado por la página.

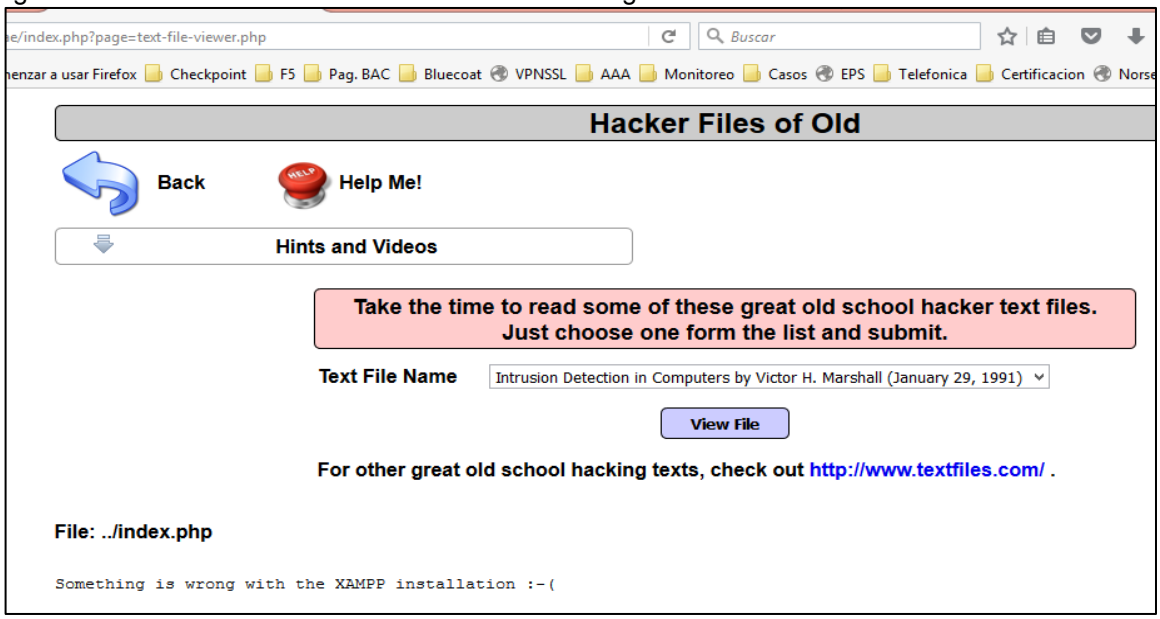
Figura 31- Acceso a otros ficheros del portal mutillidae mediante Burp Suite



Fuente: El autor

Al acceder a la ruta ../index.php desde la herramienta Burp Suite la página muestra el contenido que tiene dicho fichero. Ver figura Figura 32.

Figura 32- Fichero accedido mediante referencia insegura

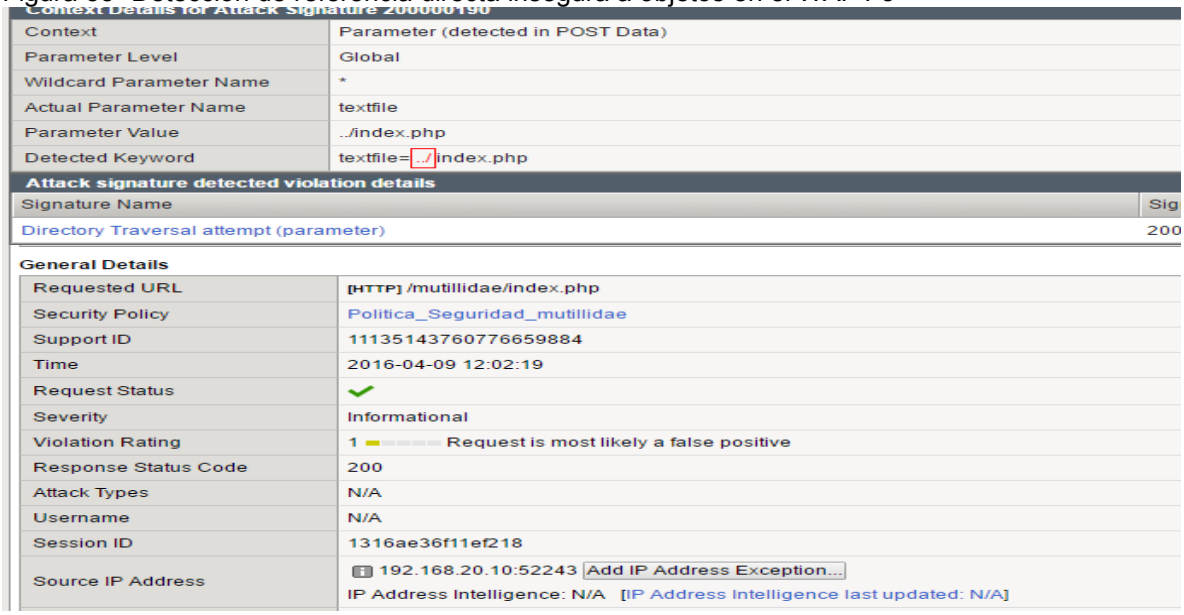


Fuente: El autor

Al revisar en el WAF F5 se detecta que la solicitud hecha por el usuario conincide con el valor de "../" lo cual es catalogado por el WAF como un intento de directorio

transversal y se considera como una violación con una severidad de evento informativo.

Figura 33- Detección de referencia directa insegura a objetos en el WAF F5



Context	Parameter (detected in POST Data)
Parameter Level	Global
Wildcard Parameter Name	*
Actual Parameter Name	textfile
Parameter Value	../index.php
Detected Keyword	textfile=../index.php

Attack signature detected violation details

Signature Name	Sig
Directory Traversal attempt (parameter)	200

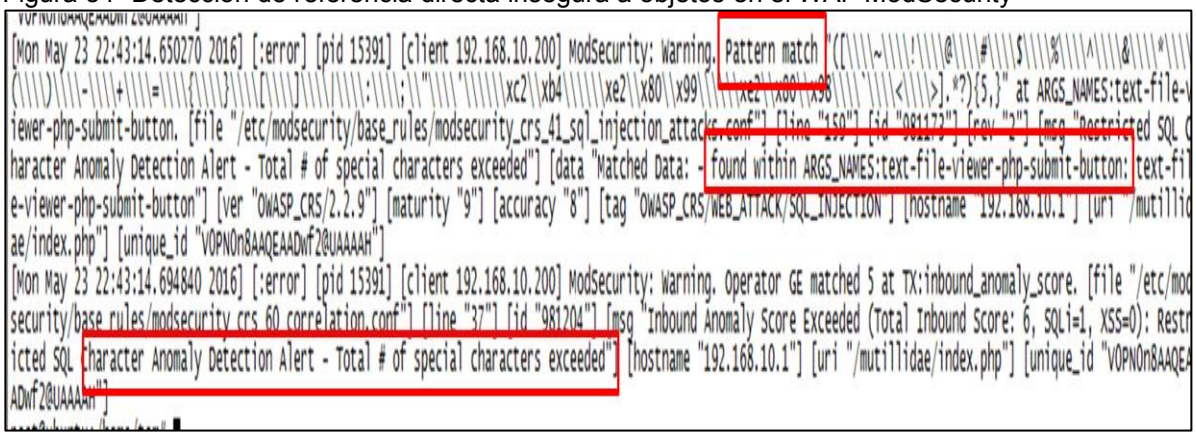
General Details

Requested URL	[HTTP] /mutillidae/index.php
Security Policy	Politica_Seguridad_mutillidae
Support ID	11135143760776659884
Time	2016-04-09 12:02:19
Request Status	✓
Severity	Informational
Violation Rating	1 Request is most likely a false positive
Response Status Code	200
Attack Types	N/A
Username	N/A
Session ID	1316ae36f11ef218
Source IP Address	192.168.20.10:52243 Add IP Address Exception... IP Address Intelligence: N/A [IP Address Intelligence last updated: N/A]

Fuente: El autor

Al analizar el evento detectado por el WAF ModSecurity, se detectan 2 eventos el primero hace referencia a una inyección de código debido a los caracteres ingresados, la segunda detecta que se supera el límite de caracteres permitidos, sin embargo para está vulnerabilidad no se relaciona el intento de acceder a otras carpetas con si se observo con el evento reportado por el WAF de marca F5. En la Figura 34 se detalla el log.

Figura 34- Detección de referencia directa insegura a objetos en el WAF ModSecurity



```
[Mon May 23 22:43:14.650270 2016] [:error] [pid 15391] [client 192.168.10.200] ModSecurity: Warning. Pattern match "([\\~\\!\\@\\#\\$\\%\\^\\&\\*\\
(\\)\\-\\+\\=\\{\\}\\[\\]\\|\\:\\;\"\\'\\xc2\\xb4\\xe2\\x80\\x99\\xe2\\x80\\x98\\`\\<\\>].*?){5,}" at ARGS_NAMES:text-file-v
iewer-php-submit-button. [file "/etc/modsecurity/base_rules/modsecurity_crs_41_sql_injection_attacks.conf"] [line "150"] [id "981173"] [rev "2"] [msg "Restricted SQL C
haracter Anomaly Detection Alert - Total # of special characters exceeded"] [data "Matched Data: - found within ARGS_NAMES:text-file-viewer-php-submit-button: text-fil
e-viewer-php-submit-button"] [ver "OWASP_CRS/2.2.9"] [maturity "9"] [accuracy "8"] [tag "OWASP_CRS/WEB_ATTACK/SQL_INJECTION"] [hostname "192.168.10.1"] [uri "/mutillid
ae/index.php"] [unique_id "V0PNOn8AAQEAADmf2@UAAAAH"]
[Mon May 23 22:43:14.694840 2016] [:error] [pid 15391] [client 192.168.10.200] ModSecurity: Warning. Operator GE matched 5 at TX:inbound_anomaly_score. [file "/etc/mod
security/base_rules/modsecurity_crs_60_correlation.conf"] [line "37"] [id "981204"] [msg "Inbound Anomaly Score Exceeded (Total Inbound Score: 6, SQLi=1, XSS=0): Restr
icted SQL Character Anomaly Detection Alert - Total # of special characters exceeded"] [hostname "192.168.10.1"] [uri "/mutillidae/index.php"] [unique_id "V0PNOn8AAQEA
ADmf2@UAAAAH"]
```

Fuente: El autor

7.5. PRUEBA OWASP A5: SEGURIDAD INCORRECTA

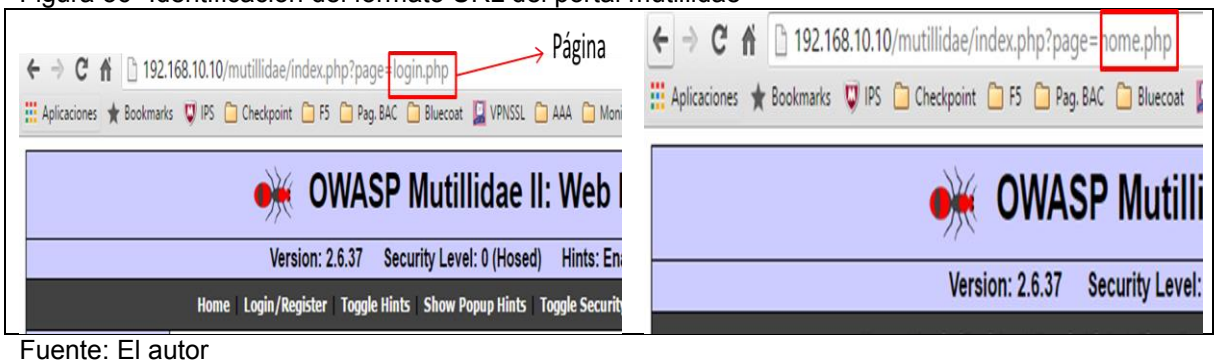
En la vulnerabilidad A5 de OWASP se consideran esas configuraciones de seguridad incorrectas que permiten tener acceso a los portales de administración o información sensible del portal. El reto en la página es encontrar los portales de administración secretos, tal como se observa en la Figura 35.

Figura 35- A5 configuración de seguridad incorrecta

Fuente: El autor

Antes de iniciar el descubrimiento del portal oculto en este tipo de vulnerabilidades es necesario capturar el formato que tiene la URL esto con el fin de identificar la variable que pueden ser modificadas de forma dinámica, como se observa en Figura 36 la páginas siempre terminan con un signo de = y finalizan con un .php.

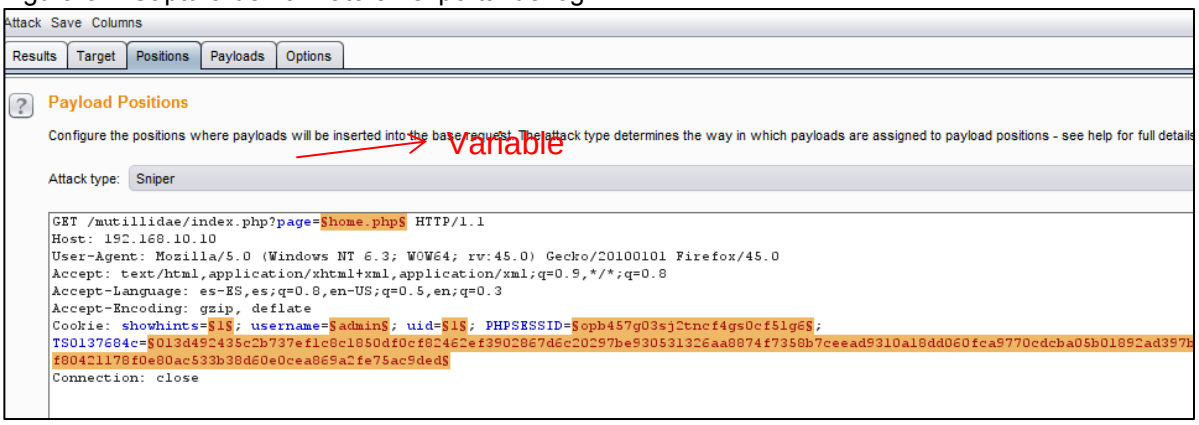
Figura 36- Identificación del formato URL del portal mutillidae



Fuente: El autor

Para realizar el descubrimiento del portal oculto se hace uso de la herramienta Burp Suite, la cual permite generar consultas de forma automatizada modificando los parámetros en la URL acorde con un listado de rutas que comunmente se encuentran en los portales, para este caso se define la variable a modificar en medio del signo $, tal como se observa en la Figura 37.

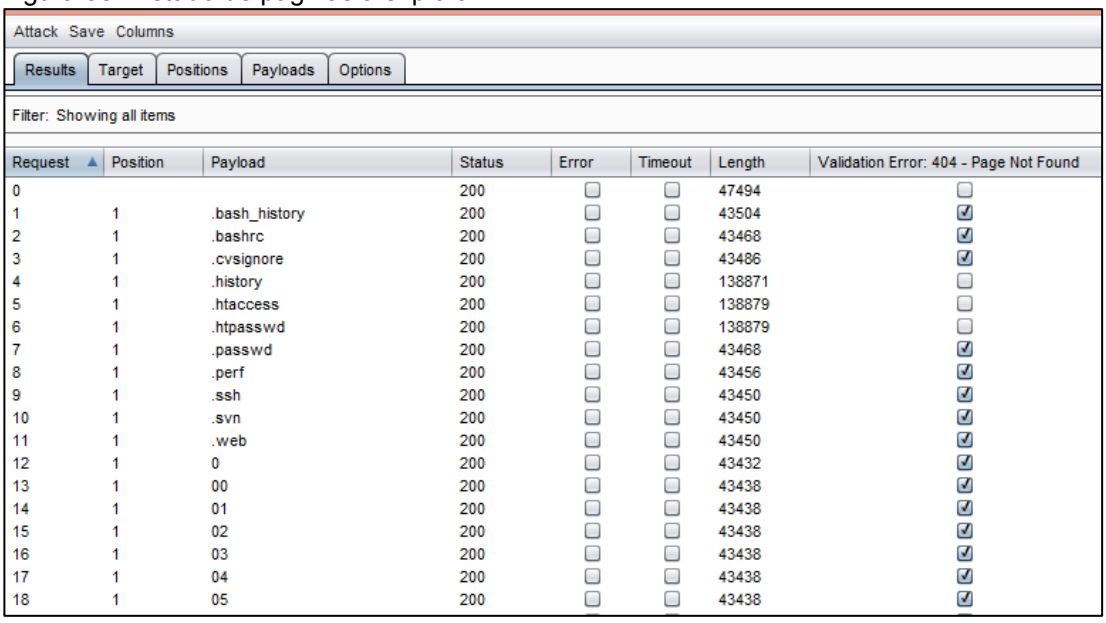
Figura 37- Captura del formato en el portal de login



Fuente: El autor

En la Figura 38 se tiene el listado de las rutas a consultadas, además se validar las respuestas dadas por la página luego de la consulta, en este caso un mensaje de “404 Page Not Found” nos indicará que la URL consultada no existe. Así se observa que existen las páginas: history, htacccess y htpasswtd.

Figura 38- Listado de páginas a explorar

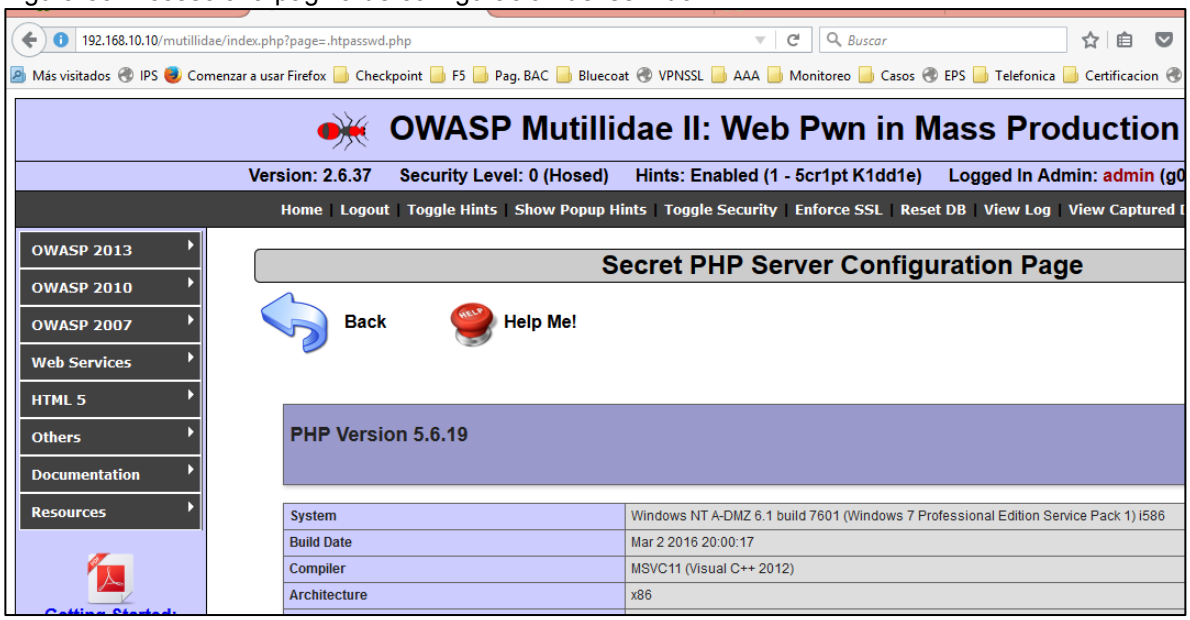


Request	Position	Payload	Status	Error	Timeout	Length	Validation Error: 404 - Page Not Found
0			200	☐	☐	47494	☐
1	1	.bash_history	200	☐	☐	43504	☑
2	1	.bashrc	200	☐	☐	43468	☑
3	1	.cvsignore	200	☐	☐	43486	☑
4	1	.history	200	☐	☐	138871	☐
5	1	.htaccess	200	☐	☐	138879	☐
6	1	.htpasswd	200	☐	☐	138879	☐
7	1	.passwd	200	☐	☐	43468	☑
8	1	.perf	200	☐	☐	43456	☑
9	1	.ssh	200	☐	☐	43450	☑
10	1	.svn	200	☐	☐	43450	☑
11	1	.web	200	☐	☐	43450	☑
12	1	0	200	☐	☐	43432	☑
13	1	00	200	☐	☐	43438	☑
14	1	01	200	☐	☐	43438	☑
15	1	02	200	☐	☐	43438	☑
16	1	03	200	☐	☐	43438	☑
17	1	04	200	☐	☐	43438	☑
18	1	05	200	☐	☐	43438	☑

Fuente: El autor

Se procede a validar las páginas encontradas, como es la URL htpasswd.php, la cual redirige al portal de configuración de la página, tal como se observa en Figura 39. Este tipo de divulgación de información representa un riesgo para el administrador del portal ya que facilita el levantamiento de información a un hacker, por ende el administrador del portal debe restringir el acceso a este tipo de rutas.

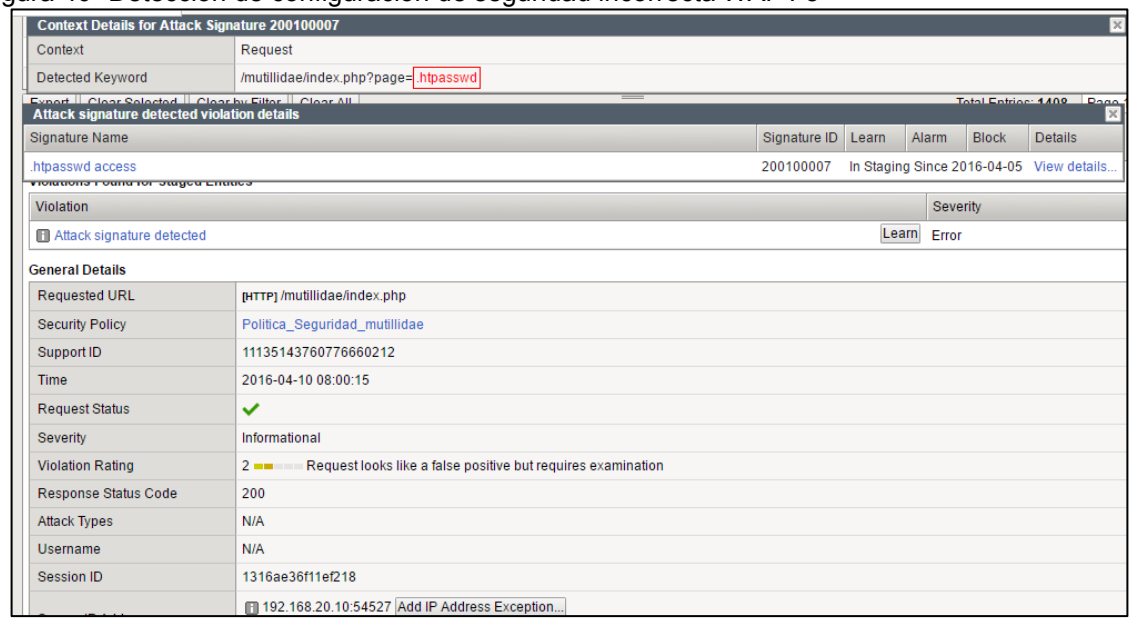
Figura 39- Acceso a la página de configuración del servidor PHP



Fuente: El autor

En este caso para el WAF F5 no se detectó que se estuviera realizando un barrido para identificar las páginas disponibles en el portal, sin embargo al intentar acceder a la página htpasswd, la misma se califica como una URL que no debe ser accedida y genera una alerta, ver Figura 40.

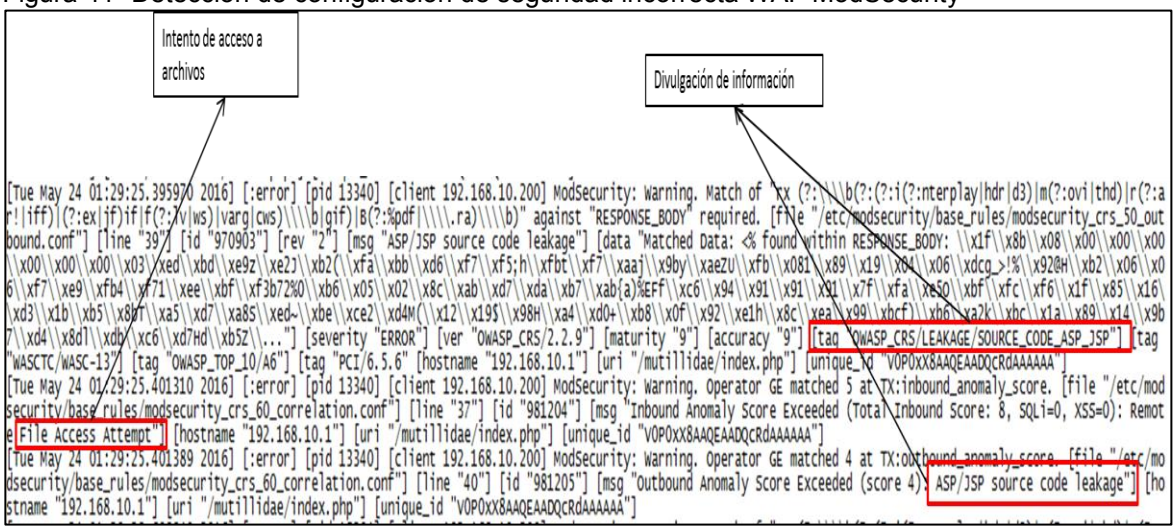
Figura 40- Detección de configuración de seguridad incorrecta WAF F5



Fuente: El autor

Al revisar en el WAF de ModSecurity, se detectan dos eventos que indican que existe información del código fuente que está siendo divulgado, además existe otro evento que indica que se están presentando intentos de acceso archivos del portal. Ver Figura 41 para mayor detalle.

Figura 41- Detección de configuración de seguridad incorrecta WAF ModSecurity

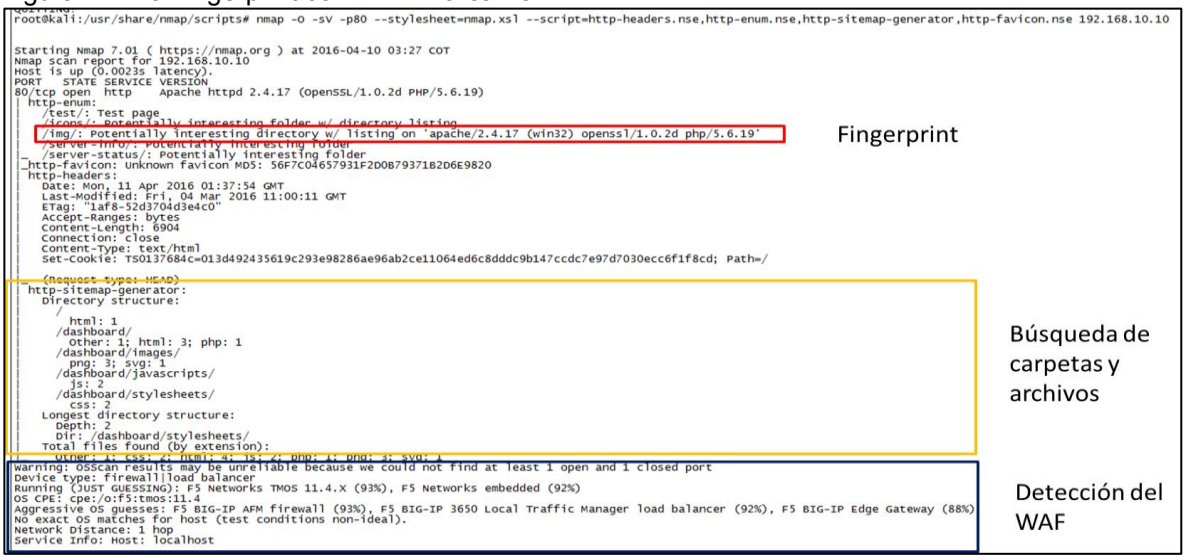


Fuente: El autor

7.6. PRUEBA OWASP A6: EXPOSICION DE DATOS

En la vulnerabilidad A6 de OWASP se tiene la exposición de datos sensibles que pueden permitir al atacante identificar el tipo de sistema operativo o tipo de servidor web instalado. Para esta prueba se hace uso de la herramienta Nmap la cual dispone de script que permiten realizar fingerprint de las páginas web, además de otras funcionalidades que permiten identificar los puertos y servicios activos de los que dispone un servidor.

Figura 42- A6 Fingerprint con WAF marca F5

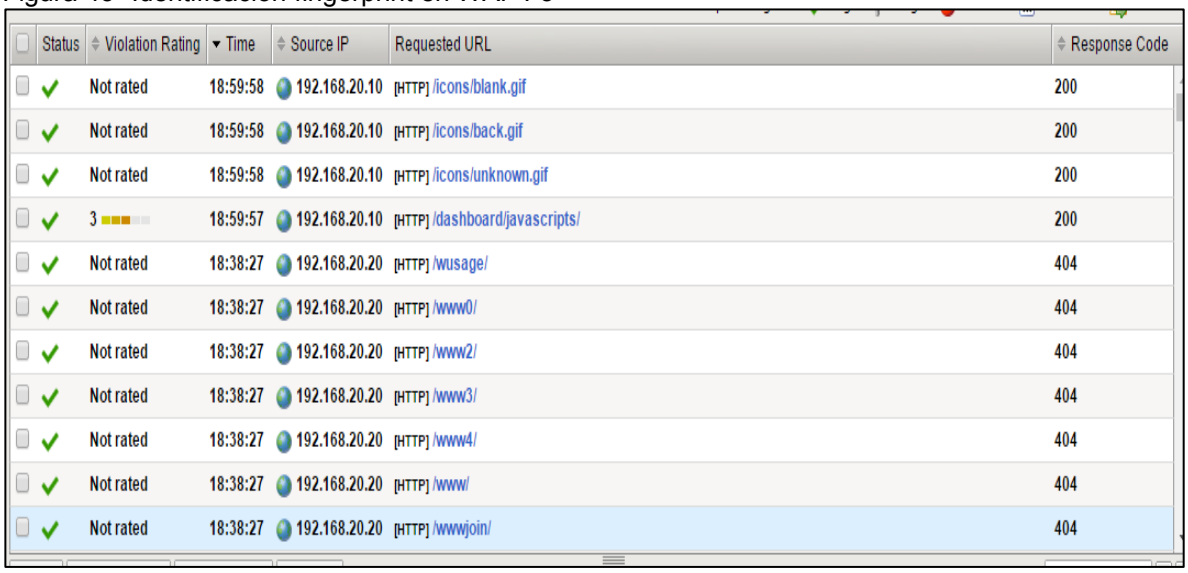


Fuente: El autor

En la Figura 42 se realiza un fingerprint del servidor HTTP en el cual está instalada la página de mutillidae empleando el WAF de marca F5. Como resultado se detecta que el servicio web corresponde a Apache y el lenguaje intérprete es PHP, además se muestra que se tiene un WAF de marca F5 en medio del camino. El comando empleado desde la consola fue “nmap -O -sV -p80 --stylesheet=nmap.xsl --script=http-headers.nse,http-enum.nse,http-sitemap-generator,http-favicon.nse 192.168.10.10”

Durante las pruebas de fingerprint en el WAF F5 no se detectó ninguna actividad anómala considerable, tan solo la revisión de diferentes tipos de carpetas en la página pero no categoriza la actividad como maliciosa, excepto el intento de acceso a la carpeta /dashboar/javascript lo cual es catalogada como una ruta de información sensible. Este tipo de comportamiento no es el esperado en un WAF puesto que la herramienta Nmap intenta realizar diferente intentos de conexión y debería generar una alerta relacionada con un escaneo. En la Figura 43 se detallan los logs revisados en el WAF de F5.

Figura 43- Identificación fingerprint en WAF F5

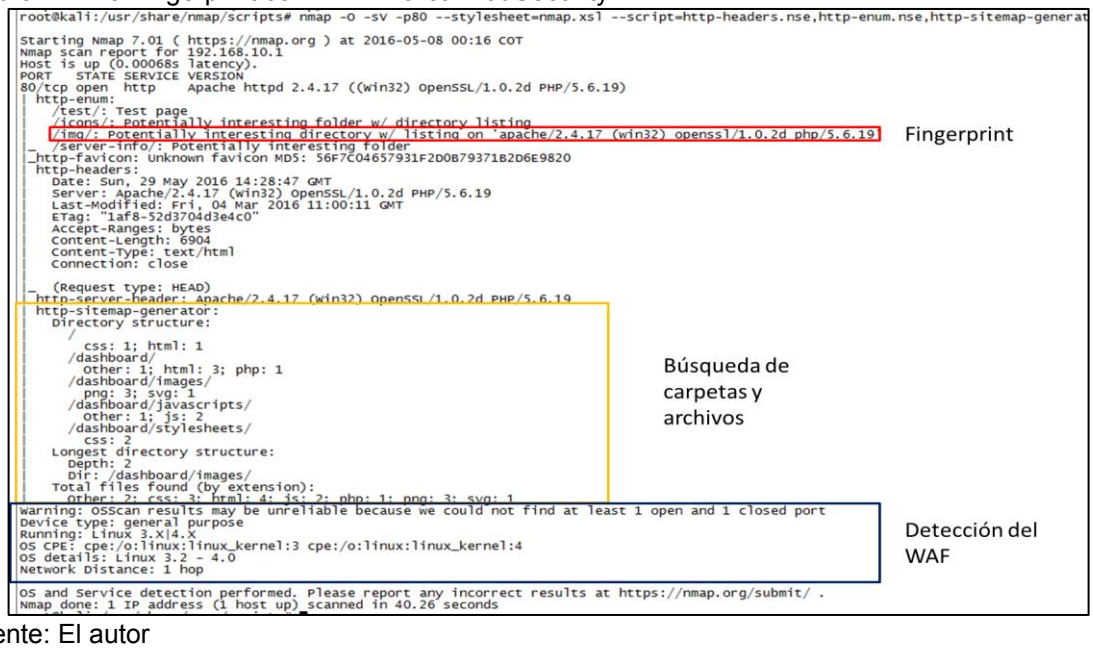


Status	Violation Rating	Time	Source IP	Requested URL	Response Code
✓	Not rated	18:59:58	192.168.20.10	[HTTP] /icons/blank.gif	200
✓	Not rated	18:59:58	192.168.20.10	[HTTP] /icons/back.gif	200
✓	Not rated	18:59:58	192.168.20.10	[HTTP] /icons/unknown.gif	200
✓	3	18:59:57	192.168.20.10	[HTTP] /dashboard/javascripts/	200
✓	Not rated	18:38:27	192.168.20.20	[HTTP] /wusage/	404
✓	Not rated	18:38:27	192.168.20.20	[HTTP] /www0/	404
✓	Not rated	18:38:27	192.168.20.20	[HTTP] /www2/	404
✓	Not rated	18:38:27	192.168.20.20	[HTTP] /www3/	404
✓	Not rated	18:38:27	192.168.20.20	[HTTP] /www4/	404
✓	Not rated	18:38:27	192.168.20.20	[HTTP] /www/	404
✓	Not rated	18:38:27	192.168.20.20	[HTTP] /wwwjoin/	404

Fuente: El autor

Al realizar las pruebas de Fingerprint a través del WAF ModSecurity, se obtuvieron resultados similares al del WAF F5, esto debido a que ambos dispositivos fueron configurados en modo escucha y permitieron el escaneo. Además, se observa que mediante la herramienta Nmap fue posible detectar en ambas ocasiones que se tenía un WAF de por medio, como se observa en la Figura 44, donde el web server está sobre un sistema operativo Windows y se detecta un servidor Linux actuando como CPE.

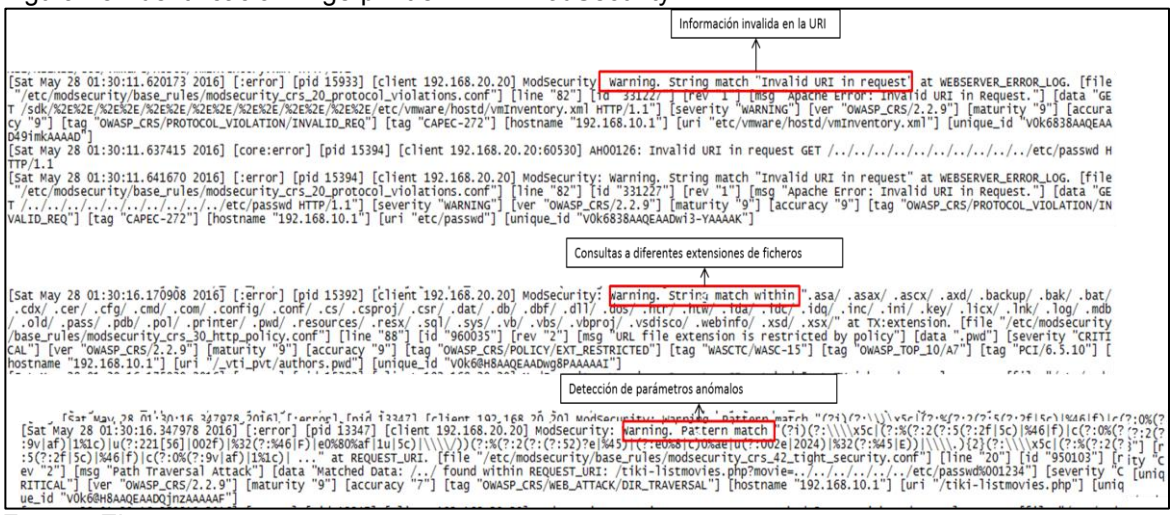
Figura 44- A6 Fingerprint con WAF marca ModSecurity



Fuente: El autor

Al revisar los eventos detectados por el WAF ModSecurity luego de ejecutar el fingerprint, en los logs del dispositivo diferentes tipos de alertas fueron generadas, en la Figura 45 se relacionan los más importantes como son: solicitudes invalidas en la URI relacionadas con directorios transversales, consultas a diferentes tipos de extensiones de archivos como .asa. asax.ascx, entre otros; y detección de parámetros anómalos o caracteres no válidos. Esto indica que el WAF de ModSecurity es más sensible a la ejecución de escaneos con base en lo arrojado por los logs.

Figura 45- Identificación fingerprint en WAF ModSecurity

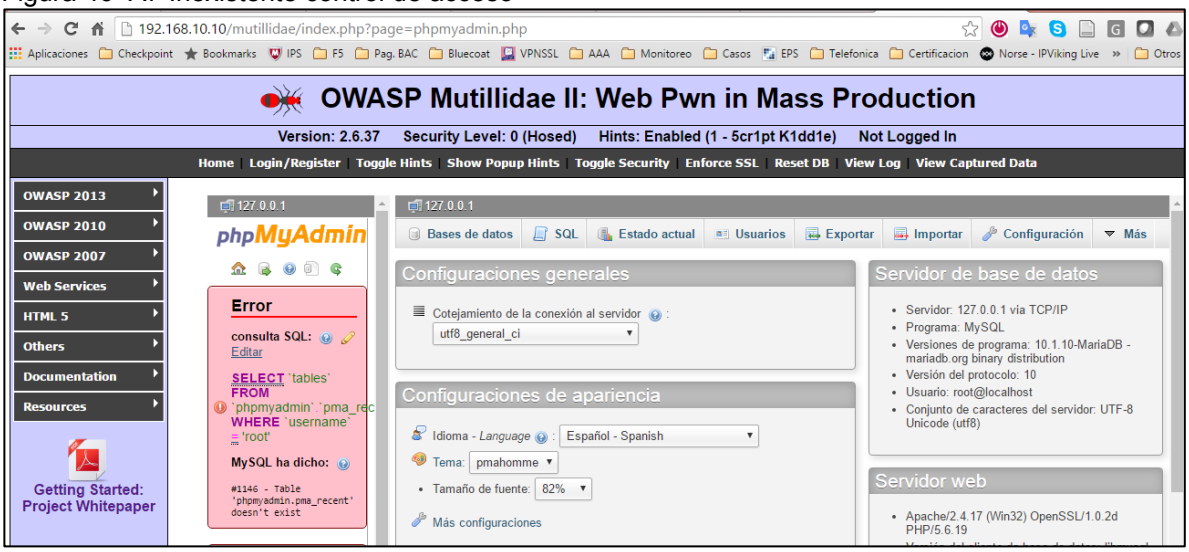


Fuente: El autor

7.7. PRUEBA OWASP A7: INEXISTENTE CONTROL DE ACCESO

En la vulnerabilidad A7 de OWASP se tiene el inexistente control de acceso, para esta prueba se analizó el portal de administración de la página el cual se conoce como phpadmin. A través de esta página los administradores realizan la configuración o modificación tanto de las propiedades del portal como de la base de datos y permitir un acceso sin restricciones a esta página podría permitir a un atacante realizar alteraciones a una página. En la Figura 46 se observa la página de administración del portal de pruebas el cual fue accedido sin ningún tipo de restricción u contraseña.

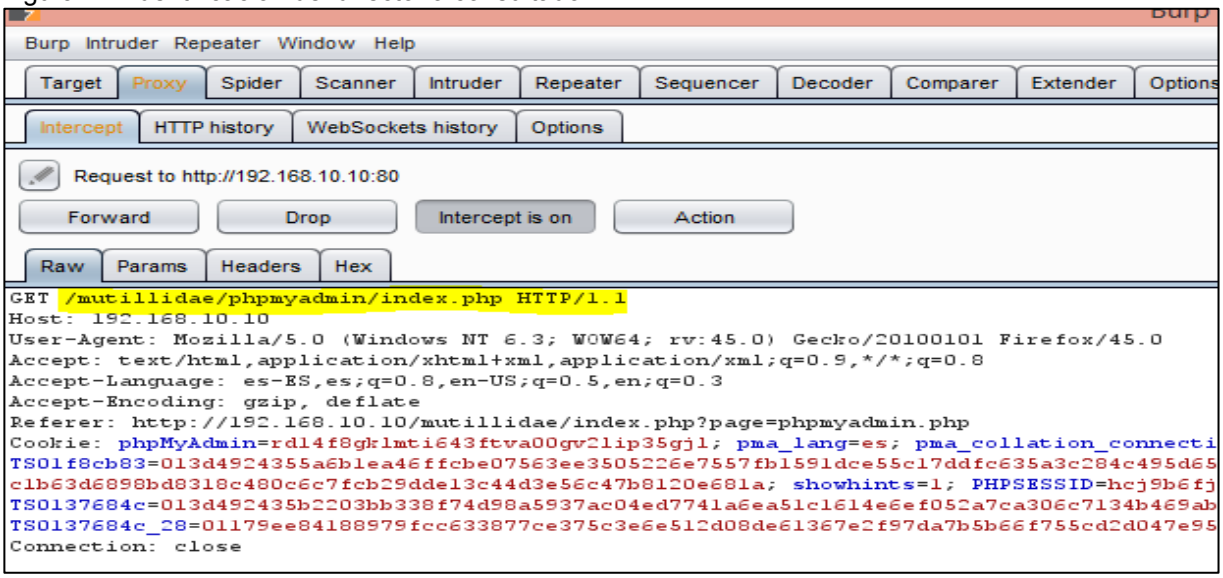
Figura 46- A7 Inexistente control de acceso



Fuente: El autor

Si bien la página de restricción debería contar con una mecanismo de autenticación, este control no sería suficiente puesto que un ataque de fuerza bruta podría encontrar la contraseña, para esto otro control propuesto es el bloqueo de este portal mediante el WAF y esta forma solo permitiriamos el acceso a un conjunto de IP que se definan como confiables. Para activar el bloqueo en el WAF se debe primero identificar el directorio donde se encuentra el portal, para ello a través de la herramienta Burp suite se realiza la captura de un GET hacia la página de administración, tal como se observa en la Figura 47.

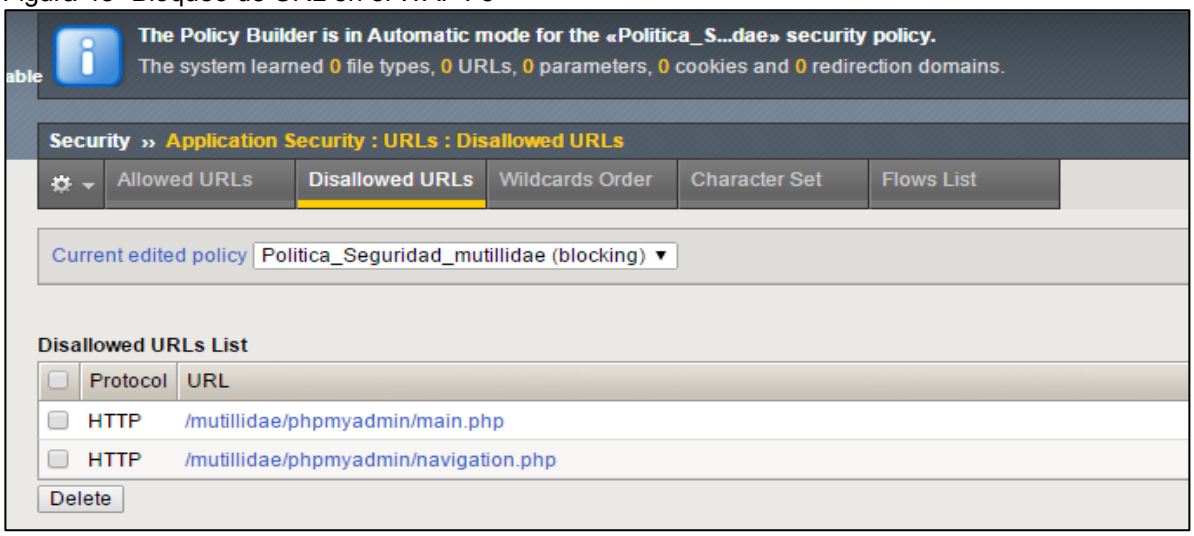
Figura 47- Identificación del directorio consultado



Fuente: El autor

Luego de identificar la URL consultada al ingresar al portal de administración, se procede a ingresar las rutas en el WAF dentro del listado de las URL no permitidas, como se muestra en la figura Figura 48. Además, se debe cambiar el modo de operación del WAF de transparente a modo bloqueo de esta forma el WAF no permitirá la conexión la URL definidas.

Figura 48- Bloqueo de URL en el WAF F5



Fuente: El autor

Como resultado en Figura 49 se observa como el WAF de marca F5 genera el bloqueo hacia el portal de administración, generando un tiquete de servicio e indicando de forma explicita que la comunicación ha sido bloqueada por un WAF.

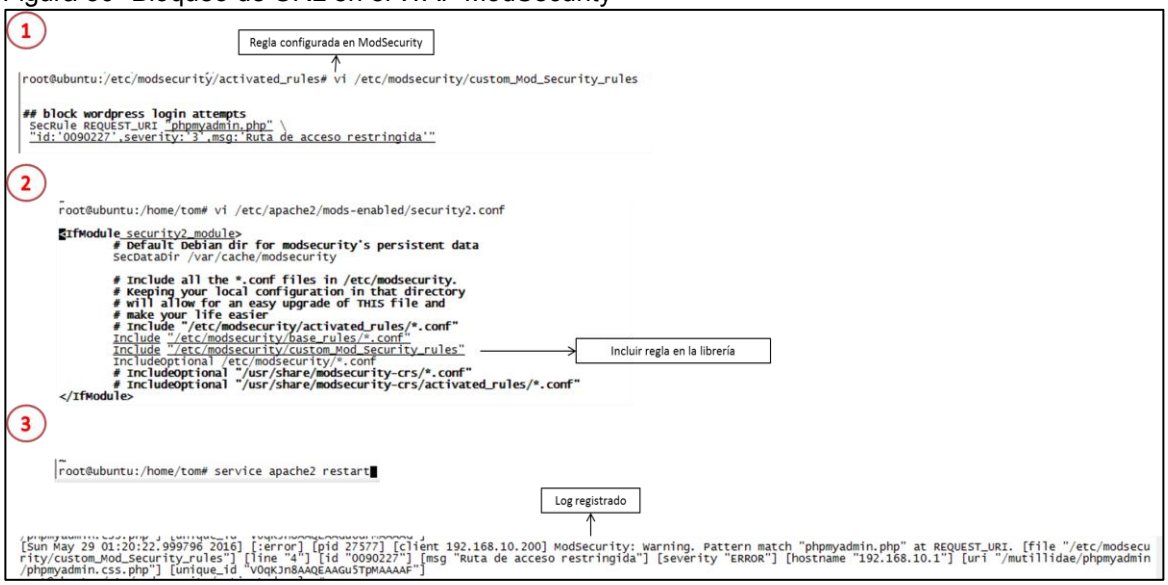
Figura 49- Acceso al portal de administración con WAF f5 en bloqueo

Fuente: El autor

Para realizar una regla personalizada por primera vez en el WAF de ModSecurity, se requiere realizar tres pasos tal como se observa en la Figura 50, primero se

debe crear una nueva regla de configuración la ruta seleccionada es “/etc/modsecurity/custom_Mod_Security_rules” allí se define el patrón con el cual se va a generar la coincidencia, el cual es una URI con el valor “phpmyadmin.php”, luego se incluye la nueva regla dentro de las librería dentro del fichero “/etc/apache2/mods-enabled/security2.conf” y finalmente se debe reiniciar el servicio para ejecutar los cambios. Al revisar los logs se observa que la URI tuvo coincidencia con el valor configurado y se muestra el mensaje personalizado.

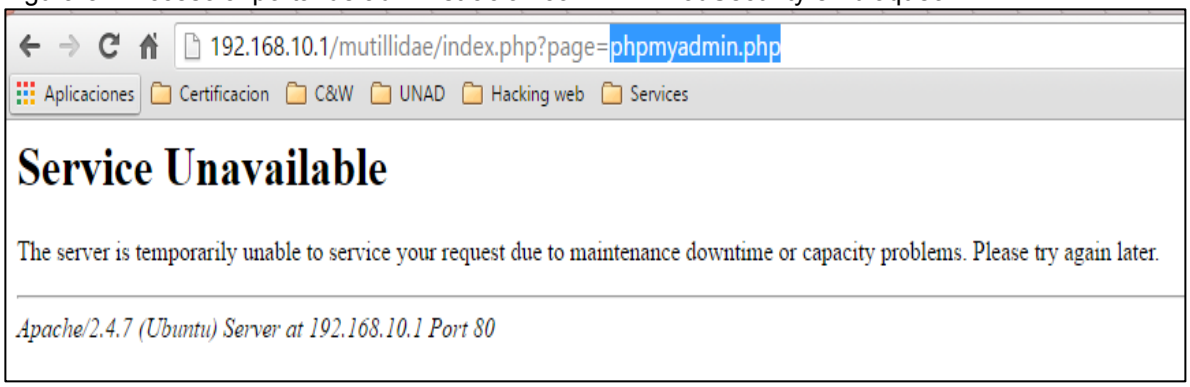
Figura 50- Bloqueo de URL en el WAF ModSecurity



Fuente: El autor

Al revisar desde el navegador el bloqueo, se observará un error como el que se muestra en la Figura 51, indicando que el servicio no está disponibile.

Figura 51- Acceso al portal de administración con WAF ModSecurity en bloqueo

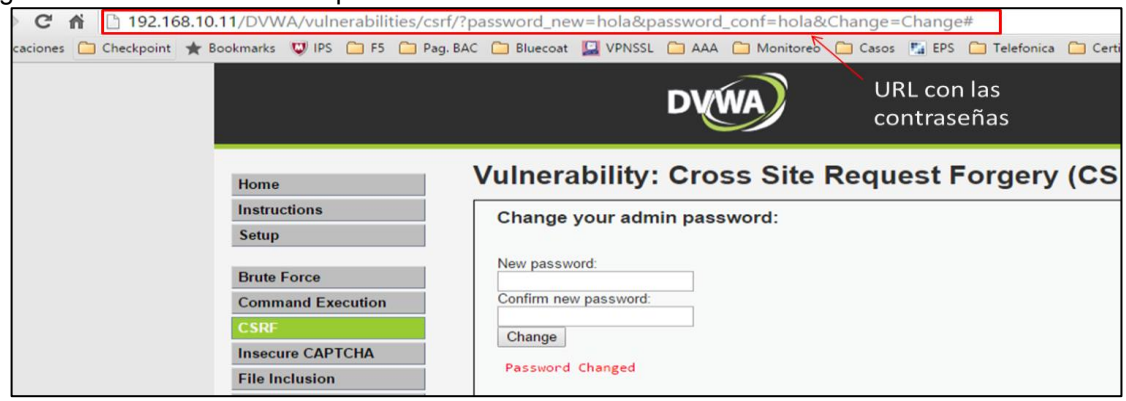


Fuente: El autor

7.8. PRUEBA OWASP A8: CSRF

La vulnerabilidad A8 de OWASP falsificación de peticiones en sitios cruzados, permite a un atacante ejecutar una acción de un contexto dentro de otro contexto diferente en la misma página o desde otro portal. En el ejemplo de la Figura 52 se tiene un contexto donde se realiza el cambio de contraseña para un usuario, y en Figura 53 se tiene un contexto para enviar mensajes. Para este caso el ataque consiste en realizar el cambio de la contraseña desde el contexto creado para enviar mensajes. Este tipo de ataque podría ser usado para evadir al mecanismo de control que se tenga dentro de un contexto.

Figura 52- A8 Falsificación de peticiones en sitios cruzados



Fuente: El autor

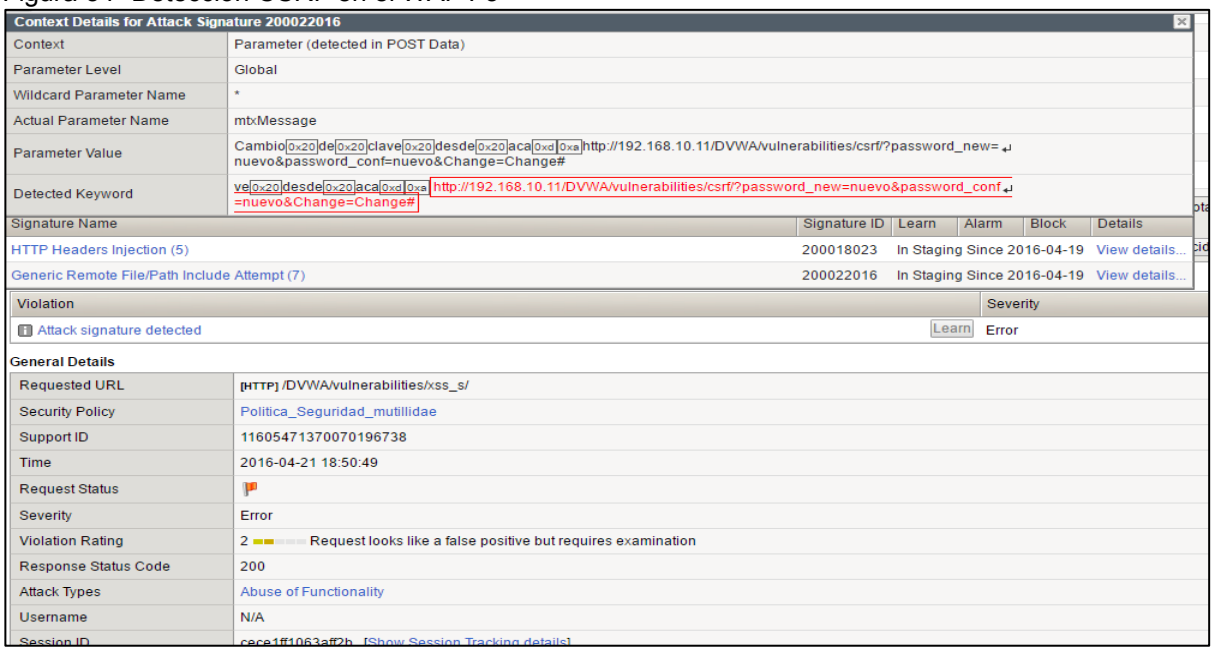
Figura 53- Cambio de contraseña a través de CSRF

Fuente: El autor

Al analizar el evento con el WAF de marca F5 se identificaron dos eventos, el primero correspondiente a una inyección en las cabeceras del encabezado de HTTP y segundo un intento de acceder a un archivo o ruta remota, tal como se

aprecia en Figura 54. Lo cual indica que el equipo no permitiría ejecutar este tipo de acción.

Figura 54- Detección CSRF en el WAF F5



Fuente: El autor

La revisar en el WAF de ModSecurity, se observan 3 eventos el primero correspondiente a un inclusión de archivos remota, el segundo relacionado con una inyección de SQL detectada por el uso de la palabra “Change=Change” dentro de la solicitud y el último evento reportado es debido a que se exceden el número de máximo de caracteres dentro de la solicitud. En la Figura 55 se detallan los logs del ModSecurity.

Figura 55- Detección CSRF en el WAF ModSecurity

Fuente: El autor

7.9. PRUEBA OWASP A9: VULNERABILDADES CONOCIDAS

El punto A9 de OWASP conocida como uso de componentes con vulnerabilidades, hace referencia a esa fallas de seguridad con las que cuenta bien sea: el sistema operativo, base de datos, lenguaje de programación o servidor web, de las cuales el atacante puede sacar ventaja si llegan a ser explotadas con éxito. Para la prueba se realiza la ejecución de vulnerabilidad llamada ShellShock, mediante la herramienta metaexploit en KaliLinux y el objetivo es identificar si el vector de este ataque es identificado por el WAF. En la Figura 56 se muestran los comandos empleados desde el equipo kalilinux para ejecutar el ShellShock con metaexploit.

ShellShock, es una vulnerabilidad conocida como GNU Bash Remote Code Execution Vulnerability (CVE-2014-6271), es catalogada como grave y al tener una ejecución exitosa de esta vulnerabilidad se obtiene control remoto de un ordenador. El problema con la vulnerabilidad se debe a que Bash permite declarar funciones pero estas no se validan de forma correcta al momento de almacenar las variables[24].

Figura 56- A9 Uso de componentes con vulnerabilidades conocidas

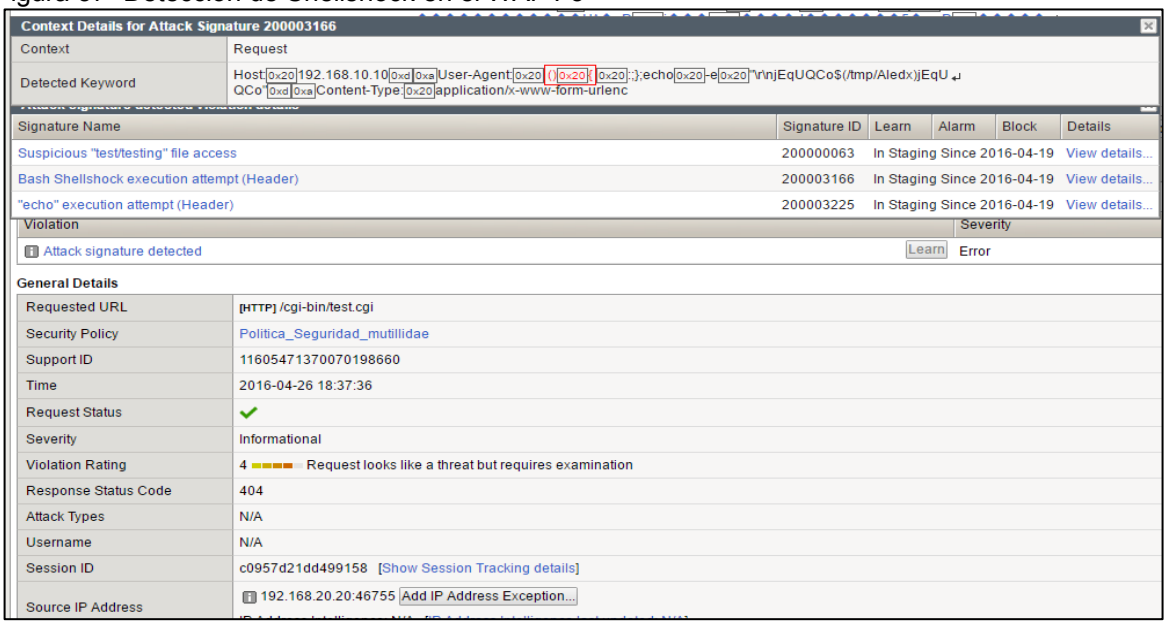
```
msf > use exploit/multi/http/apache_mod_cgi_bash_env_exec
msf exploit(apache_mod_cgi_bash_env_exec) > set rhost 192.168.10.10
rhost => 192.168.10.10
msf exploit(apache_mod_cgi_bash_env_exec) > set lhost 192.168.20.20
lhost => 192.168.20.20
msf exploit(apache_mod_cgi_bash_env_exec) > set rport 80
rport => 80
msf exploit(apache_mod_cgi_bash_env_exec) > set PAYLOAD linux/x86/shell/reverse_tcp
PAYLOAD => linux/x86/shell/reverse_tcp
msf exploit(apache_mod_cgi_bash_env_exec) > set CMD_MAX_LENGTH 2048
CMD_MAX_LENGTH => 2048
msf exploit(apache_mod_cgi_bash_env_exec) > set HEADER User-Agent
HEADER => User-Agent
msf exploit(apache_mod_cgi_bash_env_exec) > set METHOD GET
METHOD => GET
msf exploit(apache_mod_cgi_bash_env_exec) > set RPATH /bin
RPATH => /bin
msf exploit(apache_mod_cgi_bash_env_exec) > set TARGETURI /cgi-bin/test.cgi
TARGETURI => /cgi-bin/test.cgi
msf exploit(apache_mod_cgi_bash_env_exec) > exploit

[*] Started reverse TCP handler on 192.168.20.20:65000
[*] Command Stager progress - 100.60% done (837/832 bytes)
[*] Exploit completed, but no session was created.
```

Fuente: El autor

Al realizar la inspección del evento sobre el WAF de marca F5, en la Figura 57 se puede observar que varias firmas de ataques se han detectado por el WAF incluyendo la nombrada como Shellshock, lo cual nos permite identificar que este dispositivo cuenta con firmas preconfiguradas de vulnerabilidades conocidas y de esta forma se previenen este tipo de ataques.

[24] #Shellshock, la grave vulnerabilidad en Bash -y todo lo que debes saber [en linea].< http://www.welivesecurity.com/la-es/2014/09/26/shellshock-grave-vulnerabilidad-bash/ > [citado en 29 de mayo de 2016].

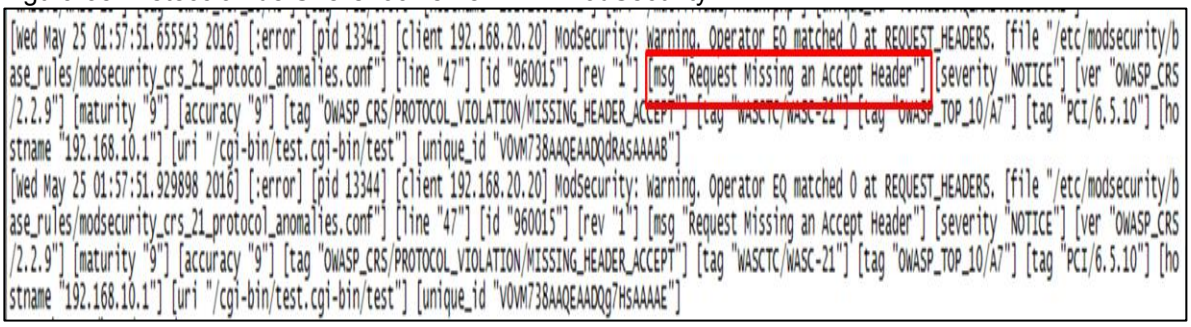
Figura 57- Detección de Shellshock en el WAF F5



Fuente: El autor

Al realizar las pruebas del vector de ataque ShellShock sobre el WAF de ModSecurity, solo se detecta un evento relacionado con la falta de un “accept” en el encabezado http, sin embargo no se observa nada relacionado con la firma de ShellShock.

Figura 58- Detección de Shellshock en el WAF ModSecurity

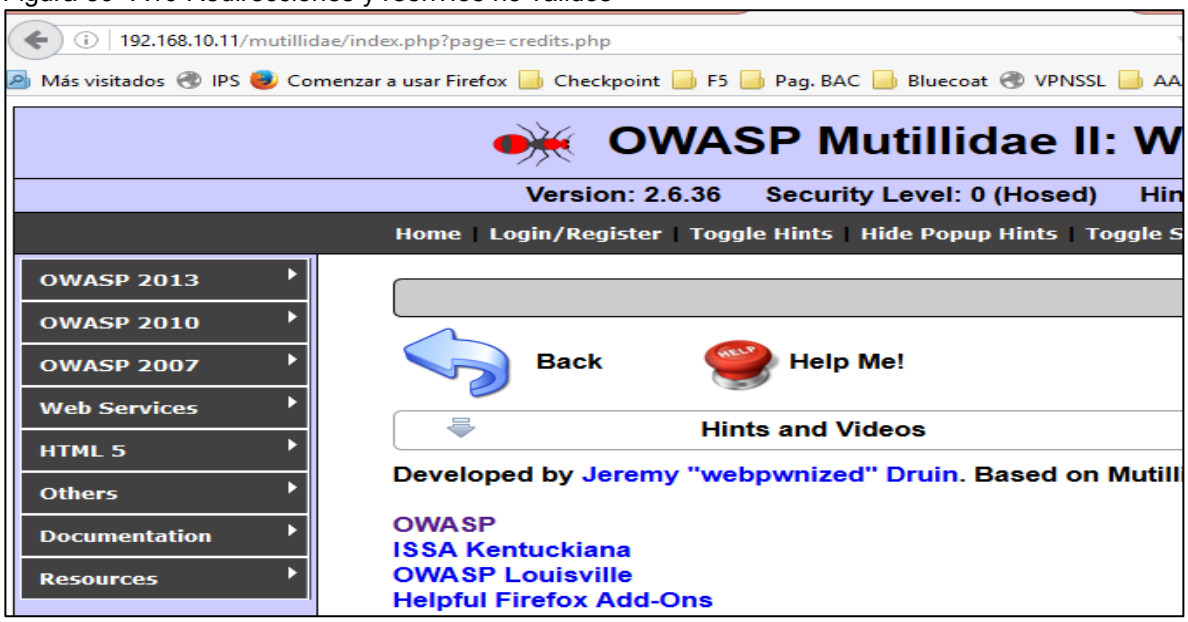


Fuente: El autor

7.10. PRUEBA OWASP A10: REDIRECCIONES Y REENVÍOS

La vulnerabilidad A10 de OWASP trata el tema de las redirecciones y reenvíos no válidos, generalmente empleado en temas de phishing donde un usuario es redireccionado de un portal no confiable a uno confiable esto para hacer creer al usuario en la etapa final de ataque que se encontraba en el portal confiable. En el portal de Mutillidae el reto es modificar las redirecciones establecidas en el portal, en la Figura 59 se muestra la página definida para este reto.

Figura 59- A10 Redirecciones y reenvíos no válidos



Fuente: El autor

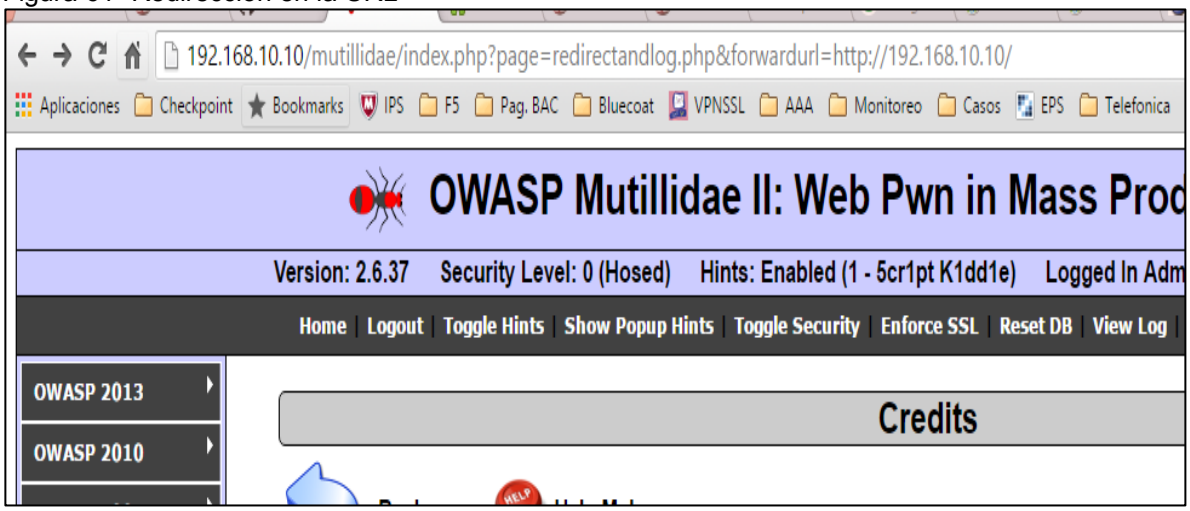
Mediante la ayuda del complemento de firebug del navegador Firefox se permite analizar el código html de un sitio, para este caso en la Figura 60 se resalta la configuración hecha para redireccionar el sitio de Mutillidae hacia el portal de OWASP (http://www.owasp.org) mediante la instrucción href, con esta información se podrá configurar la redirección hacia el que se desee.

Figura 60- Identificación de la redirección

Fuente: El autor

Finalmente, para ejecutar la redirección se hará uso de la sintaxis identificada mediante el firebug y se copia esta información sobre la URL, típicamente las redirecciones son realizadas sobre el código html sin embargo en esta ocasión no se modificará este código y si la información que ingresamos sobre la URL. En la Figura 61 se detalla la redirección hecha hacia la IP 192.168.10.10.

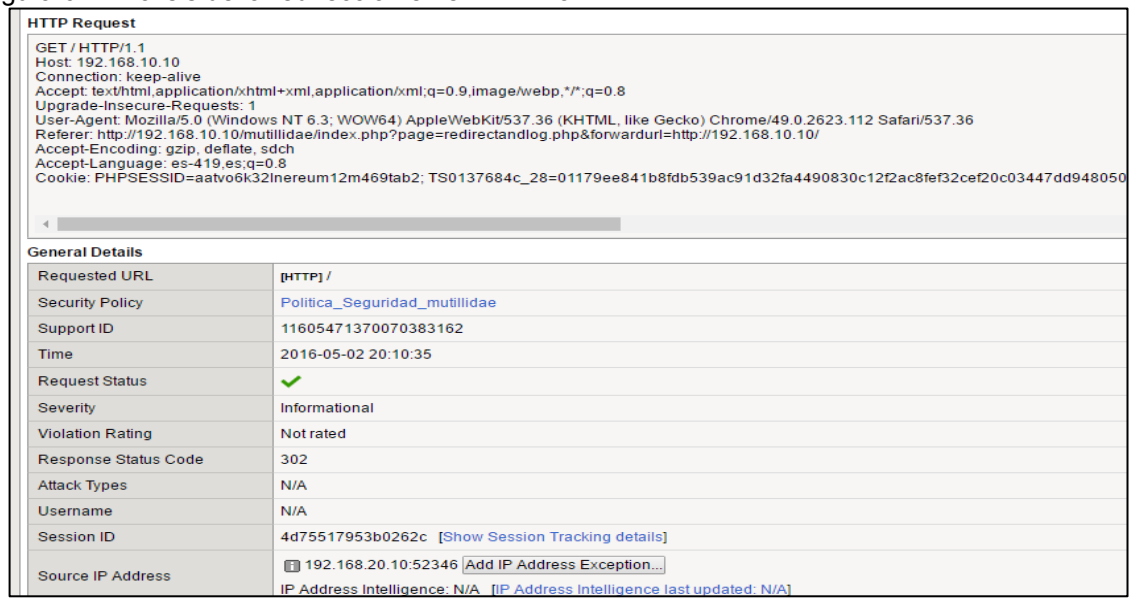
Figura 61- Redirección en la URL



Fuente: El autor

Al realizar la revisión de la redirección en el WAF de marca F5, se observa que el evento no generó una alerta en el dispositivo, sin embargo el WAF F5 permite analizar todas las solicitudes que son hechas por los usuarios y en la Figura 62 se puede observar como la redirección paso a través del WAF.

Figura 62- Análisis de la redirección en el WAF F5



Fuente: El autor

Al analizar la misma redirección con el WAF ModSecurity, como se observa en la Figura 63 se evidencia que el evento es detectado como una posible inclusión remota de archivos con una referenciación fuera del dominio. Indicando este

evento que cada vez que se presente redirecciones a un dominio diferente al dominio del servidor una alerta será generada.

Figura 63- Análisis de la redirección en el WAF ModSecurity

Fuente: El autor

7.11. CUADRO COMPARATIVO DE LAS PRUEBAS EJECUTADAS

A continuación se presenta cuadro resumen con las pruebas ejecutadas sobre el WAF de la marca F5 y ModSecurity frente al Top 10 de OWASP.

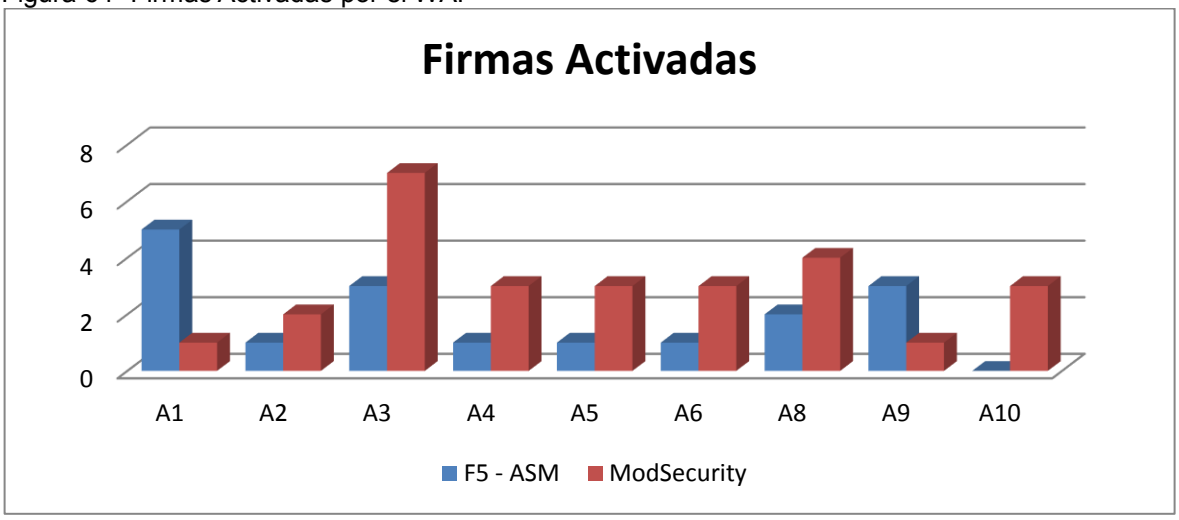
Tabla 2- Tabla comparativa de resultados

PRUEBA OWASP	WAF marca F5	WAF marca ModSecurity
A1	Se detecta la inyección de SQL.	Se detecta la inyección de SQL
A2	Se detecta uso de aplicación de la herramienta Hydra.	Se detecta la ejecución de intento de fuerza bruta y del uso herramienta Hydra.
A3	Se detecta XSS	Se detecta XSS
A4	Se detecta ataque de directorio transversal.	Se detectan caracteres anómalos, pero no relaciona el vector de ataque empleado.
A5	Se detecta acceso a rutas reservadas, no se detecta el barrido en la URL.	Se detecta intento de acceso a archivos y divulgación de información.
A6	Se detecta Fingerpinrt	Se detecta Fingerpinrt
A7	El WAF permite controlar acceso a la páginas de configuración	El WAF permite controlar acceso a la páginas de configuración
A8	Se detecta un acceso a un archivo remoto	Se detecta una acceso a un archivo remoto
A9	Se detecta el vector de ataque de Shellshock	No fue detectado el vector de ataque de ShellShock
A10	La redirección sobre la URL no fue catalogado como un evento anómalo	Se detecta una redirección fuera del dominio de la página.

Fuente: El autor

El WAF es un dispositivo que basa su funcionamiento en firmas para la detección de ataques, esto teniendo en cuenta la información que es enviada y/o recibida por el servidor web. Al realizar los ataques hacia el servidor web se evidenciaba que diferentes firmas eran activadas por el WAF, lo cual indicaba que un evento anómalo estaba sucediendo. En la Figura 64, se muestran el número de firmas que fueron activadas posteriormente a cada ataque, y se observa que el WAF de ModSecurity registraba más alertas comparado con F5. Esto se puede explicar en que los umbrales que se establecieron para F5 durante la configuración inicial estaban enfocados a generar un menor número de falsos positivos; mientras que las configuraciones por defecto que trae ModSecurity no permiten ese tipo de personalización en su parametrización, haciendo que este WAF sea más sensible y los ajustes de umbrales queden a cargo del administrador.

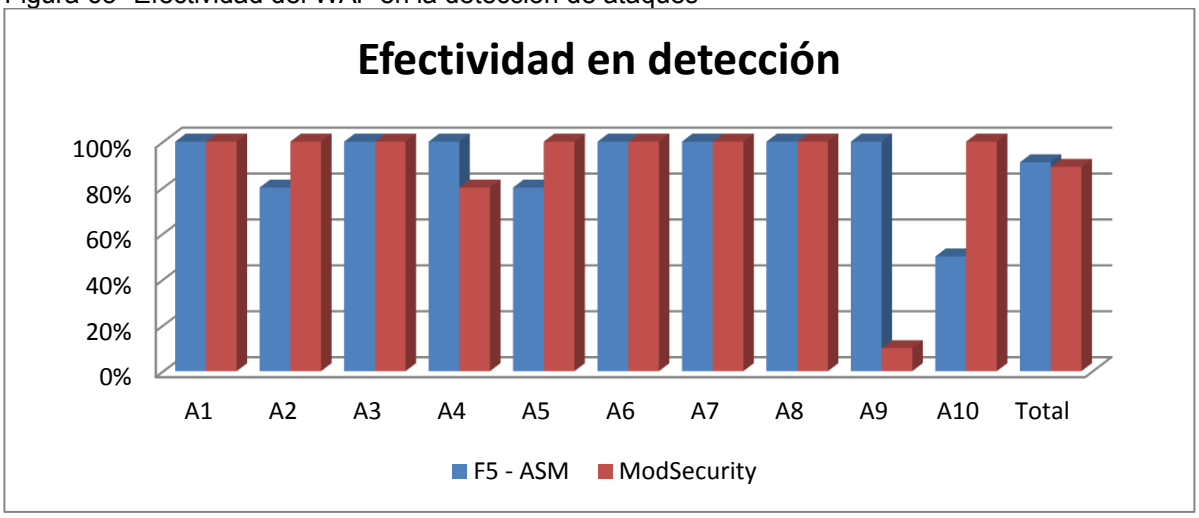
Figura 64- Firmas Activadas por el WAF



Fuente: El autor

La efectividad en la detección del tipo de ataque que se estaba ejecutado sobre el WAF fue otro aspecto evaluado durante el proyecto, esto dado que en ocasiones los eventos y alertas que generaba cada uno de los WAF permitían tener una mayor claridad del evento que estaba ocurriendo. En la Figura 65, se evaluó la efectividad en la detección de cada una de las pruebas realizadas en los dispositivos, donde se observar que la tasa de efectividad para ambas marcas fue alta, con un resultado promedio de 89% para ModSecurity y 91% para F5.

Figura 65- Efectividad del WAF en la detección de ataques



Fuente: El autor

8. CONCLUSIONES

Las funcionalidades de seguridad soportadas por ambos dispositivos presentan un alto nivel de confiabilidad, sin embargo el firewall comercial de la marca F5 presenta un mejor desempeño en el reconocimiento de nuevos vectores de ataques tal como pasó con la firma de ShellShock, pero ModSecurity presentó un mayor índice de alertas frente a las pruebas realizadas en laboratorio lo cual permitió identificar un mayor número de eventos anómalos.

Referente a la consola de administración el WAF de marca F5 presenta un mejor de desempeño al permitir analizar los eventos de una forma más intuitiva a través de su portal web, por su parte el WAF ModSecuriity exige un mayor trabajo para el análisis de los eventos puesto que su administración es mediante una interfaz de línea de comando.

La versatilidad de ModSecurity para el desarrollo de nuevas reglas de seguridad es una ventaja frente al WAF F5 y esto es ligado a su carácter de ser un código de programación abierto, sin embargo F5 también permite generar diferentes tipos de restricción de acuerdo a la información que pueda ingresar el usuario como: URL, parámetros, IP, entre otros, pero no permitiría desarrollar una firma para un ataque de día cero.

Una desventaja de ModSecurity frente a F5 y a la mayoría de firewall comerciales, es la imposibilidad de manejar un método de aprendizaje automático, lo cual en un ambiente productivo permite reducir el número de falsos positivos puesto que el WAF identificará esos anómalos que son reiterativos y que son hechos por diferentes tipos de usuarios para clasificarlos como confiables o sugerir al administrador un revisión de los eventos.

El esquema de proxy reverso se sugiere como la implementación con mayor escalabilidad en cuanto se disponga de un gran número de servidores web para proteger, además este tipo de solución se acopla bien con soluciones que requieren realizar balanceo de carga.

9. RESULTADOS E IMPACTOS DEL PROYECTO

Las pruebas ejecutadas dentro de este trabajo de grado evaluaron el comportamiento del WAF con respecto a las 10 principales vulnerabilidades reportadas por OWASP en el 2013 lo que permitió identificar que el WAF presenta una tasa mayor del 70% de detección de los ataques ejecutados, mientras que el 30% restante pudo ser mitigado mediante la parametrización de nuevas variables en el WAF.

Otro de los aspectos importantes del estudio fue la capacidad de reacción que tenían los WAF frente a nuevas vulnerabilidades o ataques de día cero, y referente a este aspecto se encontró que el WAF de libre distribución no presentó el comportamiento adecuado a diferencia del WAF de marca F5 que sí pudo identificar el vector de ataque, y esta diferencia puede estar involucrada con el hecho de que las firmas de ataques de ModSecurity no tienen una base de datos que le permita estar ejecutando actualizaciones.

Este tipo de proyectos tiene impacto en todas las organizaciones o personas que tienen portal web y que a partir de éste generan movimientos transaccionales y/o custodian información sensible, ya que como es de demostración en este proyecto se evidencia que las fallas de seguridad de un portal web podrían permitir a un atacante: realizar acceso no autorizados, o modificar o alterar la información que allí está almacenada. Es por este motivo que un WAF es un dispositivo que debe estar presente dentro de una infraestructura tecnológica donde se quiere brindar un mayor nivel de seguridad y se deseen minimizar los riesgos ante un ataque informático.

10. DIVULGACIÓN DEL PROYECTO

La divulgación del proyecto se hará mediante la publicación del trabajo de grado en la biblioteca de la UNAD.

BIBLIOGRAFIA

Varón, A. Á. R., Muñoz, C. A. B., Sánchez, R. M., Moreno, Á. G., Nava, J. M. G., & TOME, A. G. (2014). Hacking y seguridad de páginas Web. Paracuellos de Jarama, Madrid: RA-MA S.A. Editorial y Publicaciones

OWASP. OWASP Top 10 – 2013: Los diez riesgos más críticos en aplicaciones web [en linea]. <https://www.owasp.org/images/5/5f/OWASP_Top_10_-_2013_Final_-_Espa%C3%B1ol.pdf> [citado en 16 de noviembre de 2015].

RAZZAQ Abdul, HUR Ali, SHAHBAZ Sidra, MASOOD Muddassar, AHMAD H Farooq. Critical Analysis on Web Application Firewall Solutions. En: 2013 IEEE Eleventh International Symposiumon Autonomous DecentralizedSystems (ISADS). IEEE, Mar. 2013. ISBN 978-1-4673-5070-9.

HASTY Atashzar, ATEFEH Torkaman, MARJAN Bahrololum, MOHAMMAD H. Tadayon. (2011). A survey on web application vulnerabilities and countermeasures. En 2011 6th International Conference on Computer Sciences and Convergence Information Technology (ICCIT) (pp. 647-652).

BAU Jason, BURSZTEIN Elie, GUPTA Divij, MITCHELL John. (2010). State of the Art: Automated Black-Box Web Application Vulnerability Testing. En 2010 IEEE Symposium on Security and Privacy (pp. 332-345). http://doi.org/10.1109/SP.2010.27

BLERIM Rexha, ARBNOR Halili, KORAB Rrmoku, DREN Imeraj. (2015). Impact of secure programming on web application vulnerabilities. En 2015 IEEE International Conference on Computer Graphics, Vision and Information Security (CGVIS) (pp. 61-66).

ZIBORDI DE PAIVA Oscar, RUGGIERO Wilson Vicente. (2015). A survey on Information Flow Control mechanisms in web applications. En 2015 International Conference on High Performance Computing Simulation (HPCS) (pp. 211-220).

DERMANN Maximiliam, DZIADZKA Mirko, HEMKEMEIER Boris, HOFFMANN Achim MEISEL Alexander, ROHR Matthias, SCHREIBER Thomas. Best Practices: Use of Web Application Firewalls [enlínea]. <https://www.owasp.org/images/b/b0/Best_Practices_WAF_v105.en.pdf> [citadoen8 de noviembre de 2015].

REVISTASEGURIDADDEFENSA DIGITAL. Firewall de Aplicación Web – Parte II [en línea].En <http://revista.seguridad.unam.mx/numero-17/firewall-de-aplicaci%C3%B3n-web-parte-ii>[citado en8 de noviembre de 2015].

ANEXOS

ANEXO A. RESUMEN ANÁLITICO RAE

Título de Documento.	ANÁLISIS COMPARATIVO DE UN FIREWALL DE APLICACIONES WEB COMERCIAL Y UN OPEN SOURCE FRENTE AL TOP 10 DE OWASP
Autor	PIEDRAHITA VILLARRAGA Elkin Mauricio
Palabras Claves	Vulnerabilidades Web, Firewall de aplicación web, OWASP
Descripción Hace referencia al tipo de documento es una monografía, investigación, proyecto aplicado. Igualmente con qué objetivo se realizó el documento.	
Fuentes Bibliográficas	Varón, A. Á. R., Muñoz, C. A. B., Sánchez, R. M., Moreno, Á. G., Nava, J. M. G., & TOME, A. G. (2014). Hacking y seguridad de páginas Web. Paracuellos de Jarama, Madrid: RA-MA S.A. Editorial y Publicaciones
Desarrollar una aplicación web podría conllevar un gran número de riesgos informáticos inherentes, existen diferentes tipos de técnicas que han sido utilizadas para tomar provecho de este tipo de aplicaciones algunas de las más conocidas se pueden encontrar en el top OWASP 10, sin embargo día a día nuevas vulnerabilidades son encontradas y la posibilidad que un riesgo se materialicé es cada vez mayor. Por esta razón, las empresas que hacen uso de este tipo de aplicaciones deben tomar conciencia de las vulnerabilidades a las que pueden estar expuestos y establecer algún tipo de control que permita disminuir los riesgos. Los controles que se pueden llevar a cabo en una aplicación web son dos, el	

primero es realizar una auditoria periódica donde se hace una revisión del código y se ejecutan pruebas de sombrero blanco con el fin de encontrar algún tipo de vulnerabilidad, la segunda es implementar un firewall de aplicación web. Cada control ofrece sus ventajas y desventajas, y en el mejor de los casos lo ideal sería disponer de ambos, si bien la auditoria permitiría generar código más robusto habría una brecha de seguridad en el lapso que una nueva vulnerabilidad sea encontrada hasta el momento en que la auditoria sea realizada. Así que disponer de un WAF que esté en todo momento revisando las peticiones de los usuarios podría incrementar el nivel de seguridad.

Ahora la pregunta sería, ¿Qué nivel de seguridad y confiabilidad ofrece un WAF?, históricamente los WAF empezaron a ganar popularidad luego que en el Consejo de Estándares de Seguridad (PCI-DSS) exigieran a las entidades emisoras de tarjetas de crédito realizar controles sobre las aplicaciones web bien sea por revisión del código o mediante un WAF. Hoy en día existen diferentes fabricantes dedicados al desarrollo de WAF y analizando lenguajes de programación tales como HTML, HTTPS, SOAP and XML-RPC, además permiten prevenir ataques como XSS, inyecciones de SQL, secuestro de sesión, desbordamiento de buffer, ataques de día cero entre otros. Así que con base en la anterior los WAF hoy en día tienen una gran reputación y ofrecen un alto nivel de seguridad.

Teniendo en cuenta que existen tantas marcas de WAF así como beneficios a nivel seguridad, este proyecto se enfocó en evaluar las diferencias que podrían existir entre un WAF comercial frente a uno de libre de distribución, esta comparación se hizo con base en el Top 10 de OWASP donde cada una de las vulnerabilidades fue probada en cada uno de los WAF. La implementación del esquema de pruebas requirió que el WAF operará en un modo de proxy reverso de esta forma el servidor web no sufrió ningún tipo de alteración durante el desarrollo del proyecto y así garantizar unas pruebas ecuánimes.

Los resultados obtenidos de la prueba han permitido evidenciar que el WAF de marca F5 dispone una plantilla de gran cantidad lenguajes de programación web que permiten implementar reglas tan granulares tanto como se especifiquen por el administrador, además dispone un consola de administración web amigable que permite identificar de forma fácil el ataque o información anómala detectada, también se observa que la herramienta dispone de esquema de aprendizaje el cual permite notificar al WAF eventos como falso positivos y aceptar parámetros que en un principio son marcados como anómalos y lo cual es modelo de seguridad positivo permitiendo tener una mayor escalabilidad. Por su parte Modsecurity ofrece una gran versatilidad para el desarrollo de nuevas firmas de ataques, su consola de administración es a través de línea de comando, y el nivel

de seguridad que se ofrece es igual al proporcionado por WAF de marca F5 con base en las pruebas realizadas en este proyecto, las cuales no involucrando técnicas avanzadas de ataques web. Así que los niveles de seguridad brindados por el WAF comercial como el de libre distribución están iguales, sin embargo las diferencias radican en las funcionalidades que ofrece el WAF comercial que permite una mayor escalabilidad y mejor modelo de implementación.

Metodología

Inicialmente, se realizó un análisis teórico de las funcionalidades que brinda un WAF, así como su modo de operación. Luego fue necesario identificar las marcas de los WAF que se utilizarían para realizar las pruebas. Con base en lo anterior se planteó una topología pruebas la cual debería ser igual para ambos dispositivos tanto el comercial como el de libre distribución.

Posteriormente, fue necesario identificar un portal web con vulnerabilidades y revisar cómo y mediante que herramientas se podían llevar a cabo la explotación de las mismas. Para esto se encontró el portal de pruebas llamado Mutillidae y mediante el sistema operativo Kali Linux se dispuso de las herramientas necesarias realizar la explotación de las vulnerabilidades.

Finalmente, se realiza un revisión en cada uno de los WAF y se analizan los logs obtenidos en cada dispositivo luego de generar cada uno de los ataques, y de esta forma se puede evidenciar si el dispositivo cumplió o no con la protección de los portales web.

Conclusiones

A nivel de seguridad y con base en las pruebas realizadas en este proyecto tanto el firewall de aplicaciones web comercial como el de libre de distribución son idóneos para ser implementados en un esquema productivo.

Recomendaciones.

Existen diferentes tipos de ataques Web con complejidades bajas y altas, y una sola de las vulnerabilidades web listadas en el Top 10 de OWASP desprende una gran variedad de pruebas generando un documento tan extenso como pruebas se quieran. Por lo que si se desea realizar un análisis de seguridad más exhaustivo sería necesario enfocarse en un tipo de vulnerabilidad y aplicar técnicas más avanzadas de ataques que las empleadas en este proyecto.