

**UNIVERSIDAD AUTONOMA DE MADRID**

**ESCUELA POLITECNICA SUPERIOR**



## **PROYECTO FIN DE CARRERA**

**Diseño, implementación y análisis de un sistema de detección y respuesta activa**

**Alejandro Huerta Molina**

**Marzo 2015**



# **Diseño, implementación y análisis de un sistema de detección y respuesta activa**

**AUTOR: Alejandro Huerta Molina**

**TUTOR: Daniel Aparicio Montero**

**PONENTE: Jorge E. López de Vergara Méndez**

**Dpto. Tecnología Electrónica y Comunicaciones**

**Escuela Politécnica Superior**

**Universidad Autónoma de Madrid**

**Marzo de 2015**



## ***Agradecimientos***

Quiero empezar agradeciendo a Daniel todo el interés mostrado y su tiempo dedicado durante los meses que hemos trabajado juntos para que este proyecto se hiciese realidad. Gracias.

Después de estos duros meses donde mi vida no ha sido para nada la de siempre, quiero agradecer a mi familia todo el apoyo mostrado y la preocupación que han tenido durante estos meses. Sí, he estado desaparecido pero esto se ha acabado. Julián, Pilar, Sara, gracias por aguantarme.

Tampoco puedo olvidarme de los compañeros de la oficina, Roro, César, Sabone y mis queridos becarios, habéis sido un gran apoyo en los momentos difíciles.

Gracias también a todos esos compañeros de la universidad, que aunque ya queda un poco lejos, hicieron mi paso por ella una experiencia increíble, tanto en los momentos malos como en los buenos. Esas partidas de mus, esas charlas en las escaleras, esas horas perdidas de biblioteca, esos cinquillos... Grandes todos.

Por último, y no por ello menos importante ni mucho menos, este proyecto también es obra de mis amigos. Enano, More, Capi, Ramiro, Luis, gracias por estar ahí siempre.



## INDICE DE CONTENIDOS

1	Introducción.....	1
1.1	Motivación .....	1
1.2	Objetivos .....	2
1.3	Trabajo realizado .....	3
1.4	Organización de la memoria .....	4
2	Entorno tecnológico .....	5
2.1	Elementos de seguridad .....	5
2.1.1	Firewall .....	5
2.1.1.1	Tecnología estudiada. <i>Endian Firewall</i> .....	6
2.1.2	Sistema de prevención de intrusiones.....	6
2.1.2.1	Tecnología estudiada. <i>Suricata</i> .....	7
2.1.3	Correladores de eventos .....	7
2.1.3.1	Tecnología estudiada. <i>Alienvault OSSIM</i> .....	8
2.1.3.2	Tecnología estudiada. <i>Logstash, Elasticsearch y Kibana</i> .....	9
2.2	Vulnerabilidades de servicios Web .....	9
2.2.1	Inyección .....	10
2.2.2	Pérdida de autenticación y gestión de sesiones .....	10
2.2.3	Secuencias de comandos en sitios cruzados (XSS) .....	11
2.2.4	Referencia directa insegura a objetos .....	11
2.2.5	Configuración de seguridad incorrecta.....	12
2.2.6	Exposición de datos sensibles .....	12
2.2.7	Inexistente control de acceso a nivel de funcionalidades .....	13
2.2.8	Falsificación de peticiones en sitios cruzados (CSRF) .....	13
2.2.9	Uso de componentes con vulnerabilidades conocidas.....	14
2.2.10	Redirecciones y reenvíos no válidos .....	14
2.3	Conclusiones .....	15
3	Diseño de la plataforma .....	17
3.1	Seguridad perimetral .....	18
3.1.1	Firewall .....	18
3.1.2	Sistema de prevención de intrusiones.....	19
3.2	Servicio de correlación.....	20
3.3	Máquina vulnerable .....	22
3.4	Esquema de la plataforma. ....	23
3.5	Conclusiones .....	24
4	Integración de la plataforma .....	25
4.1	Software base para la maqueta .....	25
4.2	Configuración de Red.....	25
4.2.1	Definición de las redes implicadas en la plataforma. ....	25
4.2.2	Direccionamiento de los dispositivos. ....	27
4.2.2.1	Firewall perimetral .....	27
4.2.2.2	Sistema de prevención de intrusiones .....	27
4.2.2.3	Máquina vulnerable .....	28
4.2.2.4	Correlador de eventos. ....	28
4.2.2.5	Conexión final de la infraestructura.....	28
4.3	Configuración de los dispositivos .....	29

4.3.1 Firewall .....	29
4.3.2 Sistema de prevención de intrusiones .....	30
4.3.3 Máquina vulnerable .....	31
4.3.4 Correlador de eventos .....	31
4.4 Conclusiones .....	31
5 Pruebas y resultados .....	33
5.1 Análisis de seguridad sin bloqueo de conexiones .....	33
5.1.1 Inyección SQL .....	34
5.1.1.1 Explotación de la vulnerabilidad .....	34
5.1.1.2 Análisis de la detección .....	36
5.1.2 Secuencias de comandos en sitios cruzados (XSS) .....	40
5.1.2.1 Explotación de la vulnerabilidad .....	40
5.1.2.2 Análisis de la detección .....	42
5.1.3 Inclusión de ficheros .....	46
5.1.3.1 Explotación de la vulnerabilidad .....	46
5.1.3.2 Análisis de la detección .....	47
5.1.4 Inyección de comandos.....	53
5.1.4.1 Explotación de la vulnerabilidad .....	53
5.1.4.2 Análisis de la detección .....	56
5.2 Análisis de seguridad con bloqueo de conexiones .....	57
5.2.1 Cambios en el sistema de prevención de intrusiones .....	57
5.2.2 Inyección SQL .....	58
5.2.3 Secuencias de comandos en sitios cruzados.....	60
5.2.4 Inclusión de ficheros .....	62
5.2.5 Inyección de comandos.....	64
5.3 Comparativa de los resultados .....	67
5.4 Conclusiones .....	68
6 Conclusiones y trabajo futuro .....	69
6.1 Conclusiones .....	69
6.2 Trabajo futuro .....	70
7 Referencias .....	71
8 Glosario.....	73
9 Anexos .....	75
9.1 Manual de configuración de Endian Firewall .....	75
9.2 Manual de configuración del correlador de eventos .....	79
10 Presupuesto.....	81
10.1 Presupuesto de ejecución material .....	81
10.1.1 Descomposición en tareas .....	81
10.1.2 Costes de mano de obra. ....	83
10.1.3 Costes de los recursos materiales.....	84
10.1.4 Coste total de los recursos.....	85
10.2 Gastos generales y beneficio industrial .....	86
10.3 Honorarios por redacción y dirección del proyecto .....	86
10.4 Presupuesto total.....	86



## **INDICE DE FIGURAS**

FIGURA 2-1: ESQUEMA DE FUNCIONAMIENTO DE UN CORRELADOR.....	8
FIGURA 2-2: OWASP TOP 10, INYECCIÓN [1].....	10
FIGURA 2-3: OWASP TOP 10, PÉRDIDA DE AUTENTICACIÓN Y GESTIÓN DE SESIONES [1].....	10
FIGURA 2-4: OWASP TOP 10, SECUENCIAS DE COMANDOS EN SITIOS CRUZADOS [1].....	11
FIGURA 2-5: OWASP TOP 10, REFERENCIA DIRECTA INSEGURA A OBJETOS [1].....	11
FIGURA 2-6: OWASP TOP 10, CONFIGURACIÓN DE SEGURIDAD INCORRECTA [1].....	12
FIGURA 2-7: OWASP TOP 10, EXPOSICIÓN DE DATOS SENSIBLES [1].....	12
FIGURA 2-8: OWASP TOP 10, INEXISTENTE CONTROL DE ACCESO A NIVEL DE FUNCIONALIDADES [1].....	13
FIGURA 2-9: OWASP TOP 10, FALSIFICACIÓN DE PETICIONES EN SITIOS CRUZADOS [1].....	13
FIGURA 2-10: OWASP TOP 10, USO DE COMPONENTES CON VULNERABILIDADES CONOCIDAS [1].....	14
FIGURA 2-11: OWASP TOP 10, REDIRECCIONES Y REENVÍOS NO VÁLIDOS [1].....	14
FIGURA 3-1: ESQUEMA GENERAL DE LA PLATAFORMA.....	23
FIGURA 4-1: ESQUEMA DETALLADO DE LA PLATAFORMA.....	28
FIGURA 5-1: FORMULARIO SQL INJECTION.....	34
FIGURA 5-2: USO FORMULARIO SQL INJECTION.....	34
FIGURA 5-3: CÓDIGO PÁGINA VULNERABLE A SQL INJECTION.....	35
FIGURA 5-4: EXPLOTACIÓN SQL INJECTION.....	36
FIGURA 5-5: EVENTOS GENERADOS POR SQL INJECTION.....	37
FIGURA 5-6: ALERTA DE SQL INJECTION.....	37
FIGURA 5-7: DETALLE DE ALERTA DE SQL INJECTION.....	38
FIGURA 5-8: ESQUEMA REGLA DE CORRELACIÓN DE SQL INJECTION.....	38
FIGURA 5-9: FORMULARIO XSS.....	40
FIGURA 5-10: USO FORMULARIO XSS.....	40
FIGURA 5-11: CÓDIGO DE LA PÁGINA VULNERABLE A XSS.....	41

FIGURA 5-12: EXPLOTACIÓN DE LA VULNERABILIDAD DE XSS .....	42
FIGURA 5-13: EVENTOS DETECTADOS DE XSS .....	43
FIGURA 5-14: ALERTA GENERADA DE XSS .....	43
FIGURA 5-15: DETALLE DE LA ALERTA DE XSS.....	44
FIGURA 5-16: PÁGINA VULNERABLE A LA INCLUSIÓN DE FICHEROS .....	46
FIGURA 5-17: EXPLOTACIÓN DE LA VULNERABILIDAD DE INCLUSIÓN DE FICHEROS.....	47
FIGURA 5-18: CÓDIGO DE LA PÁGINA VULNERABLE A LA INCLUSIÓN DE FICHEROS .....	47
FIGURA 5-19: ESQUEMA REGLA DE CORRELACIÓN DE INCLUSIÓN DE FICHEROS 1 .....	48
FIGURA 5-20: ESQUEMA REGLA DE CORRELACIÓN DE INCLUSIÓN DE FICHEROS 2 .....	49
FIGURA 5-21: EVENTOS DETECTADOS DE INCLUSIÓN DE FICHEROS 1.....	49
FIGURA 5-22: ALERTA GENERADA DE INCLUSIÓN DE FICHEROS 1.....	50
FIGURA 5-23: EVENTOS DETECTADOS DE INCLUSIÓN DE FICHEROS 2.....	50
FIGURA 5-24: ALERTA GENERADA DE INCLUSIÓN DE FICHEROS 2.....	51
FIGURA 5-25: DETALLE DE LA ALERTA DE INCLUSIÓN DE FICHEROS 2.....	51
FIGURA 5-26: FORMULARIO DE INYECCIÓN DE COMANDOS.....	53
FIGURA 5-27: USO DEL FORMULARIO DE INYECCIÓN DE COMANDOS .....	53
FIGURA 5-28: CÓDIGO PÁGINA DE INYECCIÓN DE COMANDOS.....	54
FIGURA 5-29: EXPLOTACIÓN DE LA VULNERABILIDAD DE INYECCIÓN DE COMANDOS.....	55
FIGURA 5-30: EVENTO DETECTADOS DE INYECCIÓN DE COMANDOS.....	56
FIGURA 5-31: FORMULARIO SQL INJECTION .....	58
FIGURA 5-32: USO FORMULARIO SQL INJECTION .....	58
FIGURA 5-33: INTENTO DE EXPLOTACIÓN SQL INJECTION.....	59
FIGURA 5-34: MENSAJE DE ERROR DESPUÉS DE SQL INJECTION .....	59
FIGURA 5-35: ALERTA DE SQL INJECTION .....	59
FIGURA 5-36: FORMULARIO XSS .....	60
FIGURA 5-37: USO FORMULARIO XSS .....	60

FIGURA 5-38: INTENTO DE EXPLOTACIÓN XSS.....	61
FIGURA 5-39: MENSAJE DE ERROR TRAS INTENTO DE XSS.....	61
FIGURA 5-40: ALERTA DE XSS.....	62
FIGURA 5-41: PÁGINA VULNERABLE A INCLUSIÓN DE FICHEROS.....	62
FIGURA 5-42: INTENTO DE EXPLOTACIÓN DE INCLUSIÓN DE FICHEROS.....	63
FIGURA 5-43: ALERTAS GENERADA DE INCLUSIÓN DE FICHEROS.....	63
FIGURA 5-44: FORMULARIO DE INYECCION DE COMANDOS.....	64
FIGURA 5-45: USO FORMULARIO DE INYECCION DE COMANDOS.....	64
FIGURA 5-46: INTENTO DE EXPLOTACIÓN DE INYECCION DE COMANDOS.....	65
FIGURA 5-47: PÁGINA DESPUÉS DE INTENTO DE INYECCION DE COMANDOS.....	65
FIGURA 9-1: ESTADO INTERFACES FIREWALL PERIMETRAL.....	75
FIGURA 9-2: NORMALIZADORES ACTIVADOS.....	79
FIGURA 9-3: EJEMPLO REGLA DE CORRELACIÓN.....	80
FIGURA 10-1: DIAGRAMA DE GANTT.....	83

## **INDICE DE TABLAS**

TABLA 5-1: COMPARATIVA DE RESULTADOS.....	67
TABLA 10-1: COSTES SALARIALES.....	84
TABLA 10-2: COSTES DE LA MANO DE OBRA.....	84
TABLA 10-3: GASTOS RECURSOS HARDWARE.....	84
TABLA 10-4: GASTOS RECURSOS SOFTWARE.....	85
TABLA 10-5: GASTOS RECURSOS MATERIALES.....	85
TABLA 10-6: PRESUPUESTO DE EJECUCIÓN MATERIAL.....	85
TABLA 10-7: PRESUPUESTO DE EJECUCIÓN POR CONTRATA.....	86
TABLA 10-8: PRESUPUESTO TOTAL.....	86



# **1 Introducción**

---

## **1.1 Motivación**

Durante los últimos años, la evolución de los contenidos servidos por Internet ha cambiado; el contenido es ahora dinámico y ensamblado usando servidores en la intranet, codificado usando diferentes aplicaciones y ofrecido mediante servidores web con una interfaz unificada. Con cada vez más contenido y con un número exponencial de usuarios, las infraestructuras hardware/software se han tenido que adaptar cubriendo las nuevas necesidades. Pero esta revolución no ha sido solamente tecnológica, al generalizarse el uso de Internet desde diferentes medios y situaciones, el usuario ha pasado de ser un mero consumidor de datos a proporcionarlos al resto de usuarios.

El intercambio de información e ideas ha aumentado gracias a la adopción de diferentes paradigmas y mediante el uso de diferentes tecnologías. Se ha pasado de una web 1.0 estática, técnica, poco integradora y no participativa donde al usuario se le contaba a base de clicks, a proporcionar feedback continuamente al creador del contenido, promoviendo discusiones y la inclusión o mejora del contenido aportado, convirtiéndose en un actor con una importancia mejor definida en la web.

Las tecnologías desarrolladas se han convertido en una herramienta generadora de cambios estructurales en la web. Se utiliza la web como plataforma, donde se ejecutan aplicaciones dentro del navegador web. Estos cambios permiten la elaboración de interfaces de comunicación más complejas aunque más intuitivas para el personal no técnico, facilitando la integración del conocimiento aportado por el creciente número de usuarios.

Sin embargo, con la complejidad creciente del extenso entorno tecnológico la seguridad de la información no siempre ha sido garantizada, y en demasiadas ocasiones, ni tan siquiera valuada.

En primer lugar se enumeran los ataques más comunes en los entornos web siguiendo la guía de referencia más importante para auditorías web (OWASP). A continuación, se exponen los mecanismos posibles habilitados actualmente de seguridad activa y pasiva. Una vez introducidos los conceptos, se justifica la elección de las herramientas para la elaboración de un laboratorio donde realizar la comparativa de la efectividad de las distintas soluciones a la hora de defender las aplicaciones.

De esta manera es muy importante, conociendo los riesgos que supone una fuga de información, el análisis de los diferentes tipos de defensa de los que se dispone en la actualidad, seguridad pasiva y seguridad activa.

La seguridad pasiva está compuesta por los dispositivos que brindan la primera capa de defensa en una red corporativa como son, por ejemplo los firewall perimetrales. Por otro lado, la seguridad activa es aquella que se encarga de dar una capa más de inteligencia a la seguridad pasiva, ya sea por el análisis en profundidad de protocolos o bien por la correlación de eventos de diferentes fuentes.

En este proyecto se llevan a cabo las tareas de diseñar, implementar y analizar un sistema completo de seguridad activa para la protección de una plataforma de publicaciones web, de esta forma evaluar los beneficios proporcionados por este tipo de seguridad frente a la seguridad pasiva.

Para llevar a cabo este proyecto, se ha procedido a la construcción de un laboratorio para la realización de las pruebas, en el que se han integrado elementos de seguridad pasiva y activa, como son sistemas de prevención de intrusiones, correladores de eventos, etc., en una estructura típica de doble capa dividida en Front-End y Back-End.

Este proyecto ha tenido una parte experimental significativa, haber montado una maqueta sin protección activa y otra con protección activa para mostrar las diferencias a nivel de seguridad.

## **1.2 Objetivos**

En el presente proyecto, se demuestran las notables diferencias entre una plataforma provista únicamente de seguridad pasiva y una plataforma con seguridad pasiva y activa. El objetivo final del proyecto es el diseño, implementación y análisis de una infraestructura de publicaciones web con dispositivos de seguridad pasiva y elementos para la seguridad activa capaces de detectar ataques sobre los servidores web publicados, intentos de intrusión sobre los servidores de Back-End, escaneos de vulnerabilidades sobre los equipos, etc.

Se procede al despliegue de la arquitectura mediante software libre, lo que reduce claramente los costes. Para ello, se lleva a cabo al análisis de diversos paquetes de software que nos permiten mediante máquinas Linux el despliegue de todos los elementos de la arquitectura como son los firewall, frontales web, servidores de datos, etc. Además se procede al análisis del software Suricata para actuar como sistema de detección de intrusos y de Alienvault OSSIM en su versión gratuita para llevar a cabo la correlación de eventos y generación de alertas de seguridad. Para poder demostrar la funcionalidad del sistema, se utiliza como servidor final una distribución

Linux vulnerable para poder ver los efectos de los ataques que se han utilizado para hacer las pruebas de contraste entre las dos plataformas.

Para la realización de estas pruebas, se simulan ataques de inyección SQL, secuencias de comandos en sitios cruzados, inyección de comandos e inyección de ficheros para hacer un seguimiento de los paquetes y ver las diferencias entre ambos entornos.

Con este entorno de laboratorio se facilita la simulación de ataques reales contra las aplicaciones web publicadas. Es posible entonces estudiar la efectividad de las contramedidas propuestas de forma sistemática.

### **1.3 Trabajo realizado**

Los trabajos realizados durante la realización del presente proyecto son los siguientes:

- Estudio de las necesidades de seguridad en las plataformas de publicaciones web.
- Análisis de las diferentes capas de seguridad que se añaden a este tipo de plataformas.
- Estudio de los diferentes productos disponibles para la implementación de cada una de las capas de seguridad.
- Diseño de la plataforma de hosting web que ha sido objeto de estudio.
- Implementación del laboratorio sobre la que se han realizado las pruebas de penetración.
- Comparativa de los resultados en función de los niveles de seguridad activos en cada una de las diferentes fases de prueba.
- Redacción de la memoria.

## **1.4 Organización de la memoria**

El resto de la memoria se estructura de la siguiente manera:

- En el capítulo 2, Entorno tecnológico, se lleva a cabo, de forma general, la exposición y estudio de los diferentes dispositivos que se ven implicados en el laboratorio en que se realizan las pruebas de seguridad.
- En el capítulo 3, Diseño de la plataforma, se definen las necesidades y elecciones que se han tomado para la correcta construcción del laboratorio de pruebas.
- En el capítulo 4, Integración de la plataforma, se detallan las configuraciones de los diferentes dispositivos existentes en el laboratorio y las interconexiones que existen entre ellos para cumplir con las especificaciones del diseño.
- En el capítulo 5, Pruebas y resultados, se exponen las pruebas realizadas en escenarios con configuraciones de seguridad diferentes para poder reflejar la comparativa realizada entre los distintos entornos.
- Finalmente, el capítulo 6 presenta las conclusiones del trabajo realizado y las líneas futuras de continuación



## **2 Entorno tecnológico**

---

En este capítulo, se estudia de una manera general los diferentes dispositivos que componen la seguridad perimetral, que podemos denominar seguridad pasiva, así como los elementos que componen la seguridad activa de la red, que son los encargados de realizar acciones en función de la información que recopilan de los sistemas de seguridad.

Como nos encontramos en un escenario de protección de una plataforma de publicaciones Web, nos centraremos en la protección de la misma y para ello estudiaremos tres elementos que están implicados en la seguridad de este tipo de implantaciones. Estos tres elementos son:

- Cortafuegos perimetral (Firewall)
- Sistema de prevención pasiva de intrusiones (IPS)
- Correlación de eventos.

Adicionalmente, en este capítulo, se estudian los diferentes tipos de ataques web que se han utilizado en las pruebas

### **2.1 Elementos de seguridad**

#### **2.1.1 Firewall**

Un cortafuegos o firewall es un sistema de defensa encargado de garantizar que todo el tráfico de entrada o salida a la red debe pasar obligatoriamente por un sistema de seguridad capaz de autorizar, denegar y almacenar las comunicaciones, de acuerdo con una política de control de acceso entre redes.

Controla tanto la comunicación desde el exterior como el tráfico generado desde la propia máquina o red interna. Actúa a base de normas que establece el administrador de seguridad o, en su defecto, el administrador de red o el usuario final. Dichas reglas definen las acciones correspondientes a llevar a cabo cuando se recibe un paquete que cumpla unas determinadas características.

En definitiva, se trata de cualquier sistema empleado para "separar" una máquina o una subred del resto, protegiéndola de servicios y protocolos que puedan suponer una amenaza a la seguridad desde el exterior. El espacio protegido por un firewall se denomina perímetro de seguridad, mientras que la red externa recibe el nombre de zona de riesgo.

### **2.1.1.1 Tecnología estudiada. Endian Firewall**

*Endian firewall* es una distribución ligera de Linux que es capaz de llevar a cabo todas las funcionalidades básicas de un Firewall. Esta distribución nos proporciona un *framework* sobre el módulo de Linux *iptables*.

Las funcionalidades básicas de este software son las siguientes:

- **Gestión de políticas de Firewall**, mediante las cuales somos capaces de definir los flujos de tráfico permitidos y los no permitidos. Estas políticas nos permiten hacer una segmentación de las redes que aparecen en el sistema así como llevar a cabo la publicación de servicios mediante reglas de traducción de direcciones de red (*NAT*).
- **Puerta de enlace**, que se utiliza por todos los dispositivos de las redes conectadas a este dispositivo para conseguir conectividad con Internet. Esto se consigue configurando este dispositivo en modo *Proxy*. También tiene la opción de habilitar la resolución de nombres (*DNS*).
- **Servidor de redes privadas virtuales (VPN)**, gracias a las cuales podemos establecer una conexión segura desde fuera de las redes privadas de la plataforma para así poder llevar a cabo la gestión de las máquinas en remoto.

### **2.1.2 Sistema de prevención de intrusiones**

La tecnología de Prevención de Intrusos (*IPS*) es considerada por algunos como una extensión de los Sistemas de Detección de Intrusos (*IDS*), pero en realidad es otro tipo de control de acceso, más cercano a las tecnologías de firewalls; incluso los complementan.

Los Sistemas de Prevención de Intrusos (*IPS*) tienen varias formas de detectar el tráfico malicioso:

1. Detección Basada en Firmas, como lo hace un antivirus,
2. Detección Basada en Anomalías: funciona con el patrón de comportamiento normal de tráfico (el cual se obtiene de mediciones reales de tráfico o es predeterminado por el administrador de la red), el cual es comparado permanentemente con el tráfico en línea, enviando una alarma cuando el tráfico real varía mucho con respecto del patrón normal, y

3. Detección Honey Pot: funciona usando un equipo que se configura como señuelo presentando distintos servicios vulnerables en un entorno controlado. De esta forma se consigue que los ataques vayan dirigidos sobre el equipo y se capturen evidencias de sus formas de acción y así posteriormente se pueden implementar políticas de seguridad.

### **2.1.2.1 Tecnología estudiada. *Suricata***

Este paquete de software permite convertir cualquier máquina Linux que tenga al menos tres interfaces de red en un sistema de prevención de intrusiones. Las principales ventajas de *Suricata* son las siguientes:

- **Multihilo**, que permite el balanceo de carga de procesamiento entre cada procesador del equipo en el que haya sido configurado. Esta característica permite cómodamente lograr velocidades de hasta 10 Gbps de tráfico real sin sacrificar cobertura en el paquete de firmas.
- **Identificación de protocolo**. Los protocolos más comunes se reconocen automáticamente por *Suricata* cuando empieza la captura del tráfico. Esto permite generar reglas de detección adaptadas específicamente al protocolo sin importar el puerto por el que vengán dichas conexiones.
- **Identificación de ficheros**. Este software es capaz de identificar una gran cantidad de tipos de ficheros que viajan a través de la red. También tiene la capacidad de calcular el *checksum MD5* que permite detectar ficheros maliciosos conocidos.

### **2.1.3 Correladores de eventos**

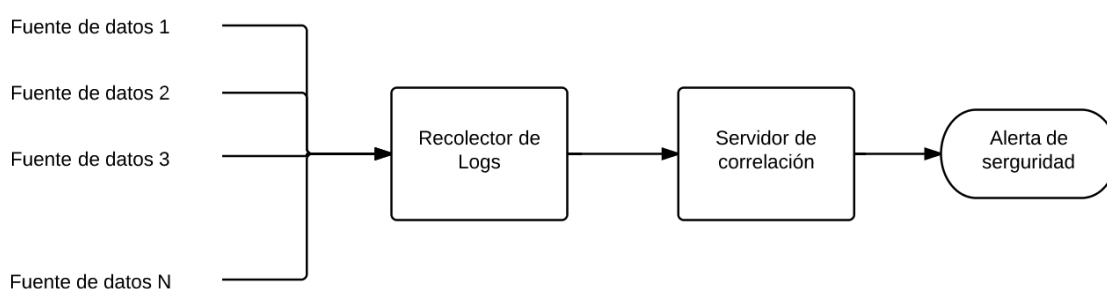
Esta tecnología se utiliza en los sistemas de seguridad para aglutinar la información producida por todos los dispositivos que hay en la red a fin de utilizarla para las siguientes tareas:

1. Normalización de la información proporcionada por los dispositivos presentes en la red.
2. Almacenamiento masivo de los logs producidos por los diferentes dispositivos.
3. Utilización de la información recolectada para generar alertas de seguridad de forma automática con los eventos que se han ido obteniendo de la plataforma.

Estos dispositivos tienen la capacidad mediante una correcta configuración de dar un nivel más de inteligencia a la información que se recoge de los elementos que conforman todo el sistema de seguridad de una plataforma.

El sistema de correlación es capaz de encontrar comportamientos de ataque en los logs almacenados, siendo capaz de generar una alerta ante estos comportamientos específicos pudiendo realizar labores de remediación y/o notificación.

El funcionamiento general de los correladores de eventos se puede definir mediante el siguiente esquema:



**Figura 2-1: Esquema de funcionamiento de un correlador**

### **2.1.3.1 Tecnología estudiada. *Alienvault OSSIM***

Para el desempeño de la correlación de eventos, en tiempo real, y con unos costes nulos gracias a que se trata de un producto open source, se ha llevado a cabo un análisis de del software *Alienvault OSSIM*.

Se trata de una distribución de Linux basada en Debian 6 que permite llevar a cabo las tareas explicadas en el apartado 2.1.3. Además tiene una integración nativa con *Suricata*, software utilizado para el sistema de prevención de intrusiones, lo que le hace estar un paso por delante de otras opciones.

Una de las características más importantes de este software es su capacidad de correlación en tiempo real de los eventos generados por los diferentes dispositivos de seguridad que se encuentran repartidos por la red. Además permite la construcción de una arquitectura en la que se separa, claramente, la recolección de eventos y la correlación de la información, lo que permite que las máquinas utilizadas para cada una de las funciones no se sobrecarguen.

Existen dos configuraciones posibles cuando se instala el paquete de software. Por un lado tenemos la opción de instalar un sistema completo de correlación que dispone de una base de

datos y del motor de correlación basado en reglas que permiten la generación de alertas de seguridad y la posterior generación de acciones relacionadas. Por otro lado, se puede montar el software sobre una máquina que únicamente se dedique a la recolección de información y la inspección del tráfico que pasa a través de ella mediante el software *Suricata*.

### **2.1.3.2 Tecnología estudiada. *Logstash, Elasticsearch y Kibana***

*Logstash* es un paquete de software nos permite llevar a cabo el almacenamiento y gestión de la información recogida de las fuentes de datos para poder efectuar búsquedas en ellos muy rápidamente gracias a la indexación que lleva a cabo sobre los datos recolectados.

No obstante, para poder llevar a cabo la correlación de eventos es necesario la agregación al sistema de almacenamiento, que nos ofrece *Logstash*, el paquete de software *Elasticsearch*. Este programa, nos da la posibilidad de llevar a cabo el análisis de los datos recogidos de las fuentes en tiempo real lo que provoca que tengamos la posibilidad de generar alertas de correlación relacionadas con dichos eventos. Esto se consigue mediante un sistema de reglas al igual que se consigue con el software de *Alienvault*.

Adicionalmente, con la inclusión del paquete de software *Kibana*, proveemos de una interfaz de usuario a *Elasticsearch*, lo que potencia mucho más sus capacidades de búsqueda y explotación de la información recogida por el sistema. Además, nos permite visualizar la información de una forma mucho más gráfica.

## **2.2 Vulnerabilidades de servicios Web**

Después del análisis realizado acerca de los diferentes dispositivos implicados en la seguridad de una red que se encarga de publicar servicios Web, es necesario llevar a cabo un análisis de algunas de las vulnerabilidades clásicas de este tipo sistemas.

En este punto se comentan los riesgos más significativos de aplicaciones Web, junto a una pequeña explicación, y una imagen con tablas que contienen información más detallada de estas vulnerabilidades. En las tablas se incluye quién puede sufrir esa vulnerabilidad, el grado de explotabilidad, si es muy común o no, la dificultad para detectarlo y los impactos técnicos e impactos al negocio.

En el año 2013, entre los diez riesgos más significativos, según el documento *OWASP Top 10 [1]*, se encontraban los siguientes:

### 2.2.1 Inyección

Las fallas de inyección, tales como SQL, OS y LDAP, ocurren cuando datos no confiables son enviados a un intérprete como parte de un comando o consulta. Los datos hostiles del atacante pueden engañar al intérprete y ejecutar comandos no intencionados o acceder a datos no autorizados.



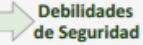

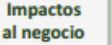
 Agentes de Amenaza	 Vectores de Ataque	 Debilidades de Seguridad		 Impactos Técnicos	 Impactos al negocio
<b>Específico de la Aplicación</b>	<b>Explotabilidad FÁCIL</b>	<b>Prevalencia COMÚN</b>	<b>Detección PROMEDIO</b>	<b>Impacto SEVERO</b>	<b>Específico de la aplicación/negocio</b>
Considere a cualquiera que pueda enviar información no confiable al sistema, incluyendo usuarios externos, usuarios internos y administradores.	El atacante envía ataques con cadenas simples de texto, los cuales explotan la sintaxis del intérprete a vulnerar. Casi cualquier fuente de datos puede ser un vector de inyección, incluyendo las fuentes internas.	Las <i>fallas de inyección</i> ocurren cuando una aplicación envía información no confiable a un intérprete. Estas fallas son muy comunes, particularmente en el código antiguo. Se encuentran, frecuentemente, en las consultas SQL, LDAP, Xpath o NoSQL; los comandos de SO, intérpretes de XML, encabezados de SMTP, argumentos de programas, etc. Estas fallas son fáciles de descubrir al examinar el código, pero difíciles de descubrir por medio de pruebas. Los analizadores y «fuzzers» pueden ayudar a los atacantes a encontrar fallas de inyección.		Una inyección puede causar pérdida o corrupción de datos, pérdida de responsabilidad, o negación de acceso. Algunas veces, una inyección puede llevar a el compromiso total de el servidor.	Considere el valor de negocio de los datos afectados y la plataforma sobre la que corre el intérprete. Todos los datos pueden ser robados, modificados o eliminados. ¿Podría ser dañada su reputación?

Figura 2-2: OWASP Top 10, Inyección [1]

### 2.2.2 Pérdida de autenticación y gestión de sesiones

Las funciones de la aplicación relacionadas con la autenticación y gestión de sesiones son frecuentemente implementadas incorrectamente, permitiendo a los atacantes comprometer contraseñas, claves, identificadores de sesiones, o explotar fallos en la implementación para asumir la identidad de otros usuarios.




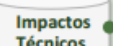
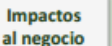
 Agentes de Amenaza	 Vectores de Ataque	 Debilidades de Seguridad		 Impactos Técnicos	 Impactos al negocio
<b>Específico de la Aplicación</b>	<b>Explotabilidad PROMEDIO</b>	<b>Prevalencia DIFUNDIDO</b>	<b>Detección PROMEDIO</b>	<b>Impacto SEVERO</b>	<b>Específico de la aplicación/negocio</b>
Considere atacantes anónimos externos, así como a usuarios con sus propias cuentas, que podrían intentar robar cuentas de otros. Considere también a trabajadores que quieran enmascarar sus acciones.	El atacante utiliza filtraciones o vulnerabilidades en las funciones de autenticación o gestión de las sesiones (ej. cuentas expuestas, contraseñas, identificadores de sesión) para suplantar otros usuarios.	Los desarrolladores a menudo crean esquemas propios de autenticación o gestión de las sesiones, pero construirlos en forma correcta es difícil. Por ello, a menudo estos esquemas propios contienen vulnerabilidades en el cierre de sesión, gestión de contraseñas, tiempo de desconexión (expiración), función de recordar contraseña, pregunta secreta, actualización de cuenta, etc. Encontrar estas vulnerabilidades puede ser difícil ya que cada implementación es única.		Estas vulnerabilidades pueden permitir que algunas o <i>todas</i> las cuentas sean atacadas. Una vez que el ataque resulte exitoso, el atacante podría realizar cualquier acción que la víctima pudiese. Las cuentas privilegiadas son objetivos prioritarios.	Considere el valor de negocio de los datos afectados o las funciones de la aplicación expuestas.  También considere el impacto en el negocio de la exposición pública de la vulnerabilidad.

Figura 2-3: OWASP Top 10, Pérdida de autenticación y gestión de sesiones [1]

### 2.2.3 Secuencias de comandos en sitios cruzados (XSS)

Las fallas XSS ocurren cada vez que una aplicación toma datos no confiables y los envía al navegador web sin una validación y codificación apropiada. XSS permite a los atacantes ejecutar secuencias de comandos en el navegador de la víctima los cuales pueden secuestrar sesiones de usuario, destruir sitios web o dirigir al usuario hacia un sitio malicioso.


					
<b>Específico de la Aplicación</b>	<b>Explotabilidad PROMEDIO</b>	<b>Prevalencia MUY DIFUNDIRA</b>	<b>Detección FÁCIL</b>	<b>Impacto MODERADO</b>	<b>Específico de la aplicación / negocio</b>
Considere cualquier persona que pueda enviar datos no confiables al sistema, incluyendo usuarios externos, internos y administradores.	El atacante envía cadenas de texto que son secuencias de comandos de ataque que explotan el intérprete del navegador. Casi cualquier fuente de datos puede ser un vector de ataque, incluyendo fuentes internas tales como datos de la base de datos.	XSS es la falla de seguridad predominante en aplicaciones web. Ocurren cuando una aplicación, en una página enviada a un navegador incluye datos suministrados por un usuario sin ser validados o codificados apropiadamente. Existen tres tipos de fallas conocidas XSS: 1) <u>Almacenadas</u> , 2) <u>Reflejadas</u> , y 3) <u>basadas en DOM</u> .  La mayoría de las fallas XSS son detectadas de forma relativamente fácil a través de pruebas o por medio del análisis del código.		El atacante puede ejecutar secuencias de comandos en el navegador de la víctima para secuestrar las sesiones de usuario, alterar la apariencia del sitio web, insertar código hostil, redirigir usuarios, secuestrar el navegador de la víctima utilizando malware, etc.	Considere el valor para el negocio del sistema afectado y de los datos que éste procesa.  También considere el impacto en el negocio la exposición pública de la vulnerabilidad.

Figura 2-4: OWASP Top 10, Secuencias de comandos en sitios cruzados [1]

### 2.2.4 Referencia directa insegura a objetos

Una referencia directa a objetos ocurre cuando un desarrollador expone una referencia a un objeto de implementación interno, tal como un fichero, directorio o base de datos. Sin un chequeo de control de acceso u otra protección, los atacantes pueden manipular estas referencias para acceder a datos no autorizados.






					
<b>Específico de la Aplicación</b>	<b>Explotabilidad FÁCIL</b>	<b>Prevalencia COMÚN</b>	<b>Detección FÁCIL</b>	<b>Impacto MODERADO</b>	<b>Específico de la aplicación/negocio</b>
Considere los tipos de usuarios en su sistema. ¿Existen usuarios que tengan únicamente acceso parcial a determinados tipos de datos del sistema?	Un atacante, como usuario autorizado en el sistema, simplemente modifica el valor de un parámetro que se refiere directamente a un objeto del sistema por otro objeto para el que el usuario no se encuentra autorizado. ¿Se concede el acceso?	Normalmente, las aplicaciones utilizan el nombre o clave actual de un objeto cuando se generan las páginas web. Las aplicaciones no siempre verifican que el usuario tiene autorización sobre el objetivo. Esto resulta en una vulnerabilidad de referencia de objetos directos inseguros. Los auditores pueden manipular fácilmente los valores del parámetro para detectar estas vulnerabilidades. Un análisis de código muestra rápidamente si la autorización se verifica correctamente.		Dichas vulnerabilidades pueden comprometer toda la información que pueda ser referida por parámetros. A menos que el espacio de nombres resulte escaso, para un atacante resulta sencillo acceder a todos los datos disponibles de ese tipo.	Considere el valor de negocio de los datos afectados o las funciones de la aplicación expuestas.  También considere el impacto en el negocio de la exposición pública de la vulnerabilidad.

Figura 2-5: OWASP Top 10, Referencia directa insegura a objetos [1]

### 2.2.5 Configuración de seguridad incorrecta

Una buena seguridad requiere tener definida e implementada una configuración segura de la aplicación, marcos de trabajo, servidor de aplicación, servidor web, base de datos y plataforma. Todas estas configuraciones por deben se definidas, implementadas y mantenidas ya que por lo general no son seguras por defecto. Esto incluye mantener el software actualizado, incluidas las librerías de código utilizadas por la aplicación.





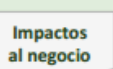
 Agentes de Amenaza	 Vectores de Ataque	 Debilidades de Seguridad		 Impactos Técnicos	 Impactos al negocio
<b>Específico de la Aplicación</b>	<b>Explotabilidad FÁCIL</b>	<b>Prevalencia COMÚN</b>	<b>Detección FÁCIL</b>	<b>Impacto MODERADO</b>	<b>Específico de la aplicación / negocio</b>
Considere atacantes anónimos externos así como usuarios con sus propias cuentas que pueden intentar comprometer el sistema. También considere personal interno buscando enmascarar sus acciones.	Un atacante accede a cuentas por defecto, páginas sin uso, fallas sin parchear, archivos y directorios sin protección, etc. para obtener acceso no autorizado o conocimiento del sistema.	Las configuraciones de seguridad incorrectas pueden ocurrir a cualquier nivel de la aplicación, incluyendo la plataforma, servidor web, servidor de aplicación, base de datos, framework, y código personalizado. Los desarrolladores y administradores de sistema necesitan trabajar juntos para asegurar que las distintas capas están configuradas apropiadamente. Las herramientas de detección automatizadas son útiles para detectar parches omitidos, fallos de configuración, uso de cuentas por defecto, servicios innecesarios, etc.		Estas vulnerabilidades frecuentemente dan a los atacantes acceso no autorizado a algunas funcionalidades o datos del sistema. Ocasionalmente provocan que el sistema se comprometa totalmente.	El sistema podría ser completamente comprometido sin su conocimiento. Todos sus datos podrían ser robados o modificados lentamente en el tiempo. Los costes de recuperación podrían ser altos.

Figura 2-6: OWASP Top 10, Configuración de seguridad incorrecta [1]

### 2.2.6 Exposición de datos sensibles

Muchas aplicaciones web no protegen adecuadamente sus datos sensibles tales como números de tarjetas de crédito o credenciales de autenticación. Los atacantes pueden robar o modificar tales datos para llevar a cabo fraudes, robos de identidad u otros delitos. Los datos sensibles requieren métodos de protección adicionales como el cifrado de datos, así como también de preocupaciones especiales en un intercambio de datos con el navegador.





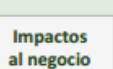
 Agentes de Amenaza	 Vectores de Ataque	 Debilidades de Seguridad		 Impactos Técnicos	 Impactos al negocio
<b>Específico de la Aplicación</b>	<b>Explotabilidad DIFÍCIL</b>	<b>Prevalencia NO COMÚN</b>	<b>Detección PROMEDIO</b>	<b>Impacto SEVERO</b>	<b>Específico de la Aplicación/Negocio</b>
Considere quién puede obtener acceso a sus datos sensibles y cualquier respaldo de éstos. Esto incluye los datos almacenados, en tránsito, e inclusive en el navegador del cliente. Incluye tanto amenazas internas y externas.	Los atacantes típicamente no quiebran la criptografía de forma directa, sino algo más como robar claves, realizar ataques "man in the middle", robar datos en texto claro del servidor, mientras se encuentran en tránsito, o del navegador del usuario.	La debilidad más común es simplemente no cifrar datos sensibles. Cuando se emplea cifrado, es común detectar generación y gestión débiles de claves, el uso de algoritmos débiles, y particularmente técnicas débiles de hashing de contraseñas. Las debilidades a nivel del navegador son muy comunes y fáciles de detectar, pero difíciles de explotar a gran escala. Atacantes externos encuentran dificultades detectando debilidades en a nivel de servidor dado el acceso limitado y que son usualmente difíciles de explotar.		Los fallos frecuentemente comprometen todos los datos que deberían estar protegidos. Típicamente, esta información incluye datos sensibles como ser registros médicos, credenciales, datos personales, tarjetas de crédito, etc.	Considere el valor de negocio de la pérdida de datos y el impacto a su reputación. ¿Cuál su responsabilidad legal si estos datos son expuestos? También considere el daño a la reputación.

Figura 2-7: OWASP Top 10, Exposición de datos sensibles [1]



### 2.2.7 Inexistente control de acceso a nivel de funcionalidades

La mayoría de las aplicaciones web verifican los derechos de accesos a nivel de función antes de hacer visible en la misma la interfaz de usuario. A pesar de esto, las aplicaciones necesitan verificar el control de acceso en el servidor cuando se accede a cada función. Si las solicitudes de acceso no se verifican, los atacantes podrían realizar peticiones sin la autorización apropiada.






					
<b>Específico de la Aplicación</b>	<b>Explotabilidad FÁCIL</b>	<b>Prevalencia COMÚN</b>	<b>Detección PROMEDIO</b>	<b>Impacto MODERADO</b>	<b>Específico de la aplicación/negocio</b>
Cualquiera con acceso a la red puede enviar una petición a su aplicación. ¿Un usuario anónimo podría acceder a una funcionalidad privada o un usuario normal acceder a una función que requiere privilegios?	El atacante, que es un usuario legítimo en el sistema, simplemente cambia la URL o un parámetro a una función con privilegios. ¿Se le concede acceso? Usuarios anónimos podrían acceder a funcionalidades privadas que no estén protegidas.	Las aplicaciones no siempre protegen las funcionalidades adecuadamente. En ocasiones la protección a nivel de funcionalidad se administra por medio de una configuración, y el sistema está mal configurado. Otras veces los programadores deben incluir un adecuado chequeo por código, y se olvidan. La detección de este tipo de vulnerabilidad es sencillo. La parte más compleja es identificar qué páginas (URLs) o funcionalidades atacables existen.		Estas vulnerabilidades permiten el acceso no autorizado de los atacantes a funciones del sistema. Las funciones administrativas son un objetivo clave de este tipo de ataques.	Considere el valor para su negocio de las funciones expuestas y los datos que éstas procesan. Además, considere el impacto a su reputación si esta vulnerabilidad se hiciera pública.

Figura 2-8: OWASP Top 10, Inexistente control de acceso a nivel de funcionalidades [1]

### 2.2.8 Falsificación de peticiones en sitios cruzados (CSRF)

Un ataque CSRF obliga al navegador de una víctima autenticada a enviar una petición HTTP falsificada, incluyendo la sesión del usuario y cualquier otra información de autenticación incluida automáticamente, a una aplicación web vulnerable. Esto permite al atacante forzar al navegador de la víctima para generar pedidos que la aplicación vulnerable piensa que son peticiones legítimas provenientes de la víctima.






					
<b>Específico de la Aplicación</b>	<b>Explotabilidad PROMEDIO</b>	<b>Prevalencia COMÚN</b>	<b>Detección FÁCIL</b>	<b>Impacto MODERADO</b>	<b>Específico de la aplicación/negocio</b>
Considere cualquier persona que pueda cargar contenido en los navegadores de los usuarios, y así obligarlos a presentar una solicitud para su sitio web. Cualquier sitio web o canal HTML que el usuario acceda puede realizar este tipo de ataque.	El atacante crea peticiones HTTP falsificadas y engaña a la víctima mediante el envío de etiquetas de imágenes, XSS u otras técnicas. <u>Si el usuario está autenticado</u> , el ataque tiene éxito.	CSRF aprovecha el hecho que la mayoría de las aplicaciones web permiten a los atacantes predecir todos los detalles de una acción en particular. Dado que los navegadores envían credenciales como cookies de sesión de forma automática, los atacantes pueden crear páginas web maliciosas que generan peticiones falsificadas que son indistinguibles de las legítimas. La detección de fallos de tipo CSRF es bastante fácil a través de pruebas de penetración o de análisis de código.		Los atacantes pueden cambiar cualquier dato que la víctima esté autorizada a cambiar, o a acceder a cualquier funcionalidad donde esté autorizada, incluyendo registro, cambios de estado o cierre de sesión.	Considerar el valor de negocio asociado a los datos o funciones afectados. Tener en cuenta lo que representa no estar seguro si los usuarios en realidad desean realizar dichas acciones. Considerar el impacto que tiene en la reputación de su negocio.

Figura 2-9: OWASP Top 10, Falsificación de peticiones en sitios cruzados [1]

## 2.2.9 Uso de componentes con vulnerabilidades conocidas

Algunos componentes tales como las librerías, los frameworks y otros módulos de software casi siempre funcionan con todos los privilegios. Si se ataca un componente vulnerable podría facilitar la intrusión en el servidor o una pérdida seria de datos. Las aplicaciones que utilicen componentes con vulnerabilidades conocidas debilitan las defensas de la aplicación y permiten ampliar el rango de posibles ataques e impactos.


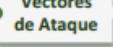
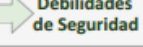

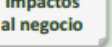
 Agentes de Amenaza	 Vectores de Ataque	 Debilidades de Seguridad		 Impactos Técnicos	 Impactos al negocio
<b>Específico de la Aplicación</b>	<b>Explotabilidad PROMEDIO</b>	<b>Prevalencia DIFUNDIDO</b>	<b>Detectabilidad DIFÍCIL</b>	<b>Impacto MODERADO</b>	<b>Específico de la aplicación / negocio</b>
Algunos componentes vulnerables (por ejemplo frameworks) pueden ser identificados y explotados con herramientas automatizadas, aumentando las opciones de la amenaza más allá del objetivo atacado.	El atacante identifica un componente débil a través de escaneos automáticos o análisis manuales. Ajusta el exploit como lo necesita y ejecuta el ataque. Se hace más difícil si el componente es ampliamente utilizado en la aplicación.	Virtualmente cualquier aplicación tiene este tipo de problema debido a que la mayoría de los equipos de desarrollo no se enfocan en asegurar que sus componentes / bibliotecas se encuentren actualizadas. En muchos casos, los desarrolladores no conocen todos los componentes que utilizan, y menos sus versiones. Dependencias entre componentes dificultan incluso más el problema.		El rango completo de debilidades incluye inyección, control de acceso roto, XSS, etc. El impacto puede ser desde mínimo hasta apoderamiento completo del equipo y compromiso de los datos.	Considere qué puede significar cada vulnerabilidad para el negocio controlado por la aplicación afectada. Puede ser trivial o puede significar compromiso completo.

Figura 2-10: OWASP Top 10, Uso de componentes con vulnerabilidades conocidas [1]

## 2.2.10 Redirecciones y reenvíos no válidos

Las aplicaciones web frecuentemente redirigen y reenvían a los usuarios hacia otras páginas o sitios web y utilizan datos no confiables para determinar la página de destino. Sin una validación apropiada, los atacantes pueden redirigir a las víctimas hacia sitios de phishing o malware o utilizar reenvíos para acceder a páginas no autorizadas.



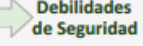
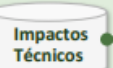
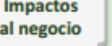
 Agentes de Amenaza	 Vectores de Ataque	 Debilidades de Seguridad		 Impactos Técnicos	 Impactos al negocio
<b>Específico de la Aplicación</b>	<b>Explotabilidad PROMEDIO</b>	<b>Prevalencia POCO COMÚN</b>	<b>Detección FÁCIL</b>	<b>Impacto MODERADO</b>	<b>Específico de la Aplicación / Negocio</b>
Considere la probabilidad de que alguien pueda engañar a los usuarios a enviar una petición a su aplicación web. Cualquier aplicación o código HTML al que acceden sus usuarios podría realizar este engaño	Un atacante crea enlaces a redirecciones no validadas y engaña a las víctimas para que hagan clic en dichos enlaces. Las víctimas son más propensas a hacer clic sobre ellos ya que el enlace lleva a una aplicación de confianza. El atacante tiene como objetivo los destinos inseguros para evadir los controles de seguridad.	Con frecuencia, las aplicaciones redirigen a los usuarios a otras páginas, o utilizan destinos internos de forma similar. Algunas veces la página de destino se especifica en un parámetro no validado, permitiendo a los atacantes elegir dicha página. Detectar redirecciones sin validar es fácil. Se trata de buscar redirecciones donde el usuario puede establecer la dirección URL completa. Verificar reenvíos sin validar resulta más complicado ya que apuntan a páginas internas.		Estas redirecciones pueden intentar instalar código malicioso o engañar a las víctimas para que revelen contraseñas u otra información sensible. El uso de reenvíos inseguros puede permitir evadir el control de acceso.	Considere el valor de negocio de conservar la confianza de sus usuarios. ¿Qué pasaría si sus usuarios son infectados con código malicioso? ¿Qué ocurriría si los atacantes pudieran acceder a funciones que sólo debieran estar disponibles de forma interna?

Figura 2-11: OWASP Top 10, Redirecciones y reenvíos no válidos [1]

## **2.3 Conclusiones**

Después del análisis realizado sobre los distintos elementos implicados en la protección de los servicios web publicados en una plataforma como la que nos encontramos y las principales vulnerabilidades que están presentes en los sistemas de este tipo, es necesario, que en los sistemas de este tipo, existan al menos las tres capas de seguridad expuestas en el apartado 2.1 para asegurarnos algún tipo de seguridad en los sistemas.

En los capítulos siguientes, se explica de forma mucho más detallada como se ha definido la plataforma y las implicaciones que esto supone a nivel de seguridad, se lleva a cabo la integración real de la plataforma en un laboratorio y se procede a someter dicho laboratorio a una batería de pruebas con diferentes tipos de configuraciones para demostrar las diferencias existentes.



## **3 Diseño de la plataforma**

---

Tras lo visto en los capítulos anteriores, se pasa a desarrollar más en profundidad las decisiones de diseño de la plataforma que se está presentando en este proyecto.

Una plataforma de seguridad para una publicación web contiene, por lo menos, las siguientes capas de seguridad perfectamente definidas y con funciones muy distintas:

- **Seguridad perimetral**

Esta sección habla acerca de las protecciones reales a nivel de red y aplicación. Se trata de protecciones activas y capaces de cortar comunicaciones en tiempo real. Se compone principalmente de dos elementos de red uno en la capa de transporte y otro en la capa de aplicación.

Dentro de la capa de transporte nos encontramos con los cortafuegos o firewall mientras que en la capa de aplicación tenemos diferentes opciones de sistemas de prevención de intrusiones.

- **Correlación de eventos**

Aquí se encuentran alojados los dispositivos que se encargan de relacionar la información recogida de todas las fuentes del sistema completo: trazas de los FW, eventos detectados en el IPS, peticiones realizadas a los servidores Web, etc. De esta manera, somos capaces de introducir un nivel más de inteligencia que permite detectar de forma automática máquinas que están llevando a cabo comportamientos maliciosos y que requieren algún tipo de acción adicional, como por ejemplo y bloqueo de las comunicaciones.

Estos dispositivos, también nos permiten detectar errores en las configuraciones de la electrónica de red de manera sistemática.

En las siguientes subsecciones se detallarán las necesidades de diseño y las decisiones en el mismo para cada una de las áreas anteriores.

### **3.1 Seguridad perimetral**

La seguridad perimetral es la primera capa de defensa frente a ataques de la que dispone una plataforma frente a ataques externos. En este apartado discutiremos la necesidad de la seguridad perimetral y sus funciones principales.

#### **3.1.1 Firewall**

Este dispositivo, que es la primera capa de seguridad, nos permite filtrar las conexiones entrantes a nuestra plataforma de modo que seamos capaces de definir políticas con respecto a qué tipo de protocolos pueden utilizarse para consultar nuestras aplicaciones publicadas.

Habitualmente en una plataforma dedicada a las publicaciones Web, estarán activadas las comunicaciones contra el puerto en el que esté publicado el servicio HTTP y, si además tenemos conexiones seguras, el puerto HTTPS.

De esta manera, somos capaces de restringir el acceso a nuestros equipos protegidos para que únicamente se redireccionen peticiones hacia nuestros equipos frontales cuando dichas comunicaciones sean, supuestamente, legítimas.

No obstante, estas comunicaciones, aunque sean conexiones lícitas para la plataforma, pueden contener códigos maliciosos que podrían suponer un riesgo para la plataforma si alguna de nuestras publicaciones sufre algún tipo de vulnerabilidad conocida.

El firewall realmente nos define un filtro que únicamente dejará pasar las comunicaciones que cumplan ciertos criterios (por ejemplo comunicaciones contra una dirección IP en concreto que vengan por los puertos permitidos), es decir define una pequeña ventana entre todos los servicios disponibles en Internet.

Después de ver la necesidad de este tipo de dispositivo como primera medida de seguridad, necesitamos definir las características de la máquina que nos proporcionará dichos servicios.

Debido a que el sistema que se va a presentar en este proyecto es un sistema completamente controlado y que no vamos a alojar más que un único frontal web, podemos utilizar una única máquina de pequeñas dimensiones, las cuales se presentan a continuación:

- 1 CPU
- 2 GB de RAM
- 10 GB de espacio en disco

El software que se utilizará para el desempeño de las funciones de Firewall es la distribución libre “Endian Firewall”. Este sistema está basado en el paquete IPTables de Linux que nos permite hacer funcionar un sistema Linux a modo de Firewall.

### **3.1.2 Sistema de prevención de intrusiones**

Este sistema, complementa la actividad del Firewall analizando en profundidad a nivel de aplicación los paquetes que han conseguido atravesar la barrera definida en el Firewall.

Gracias a estos equipos la plataforma es capaz de detectar ataques dirigidos contra los servicios publicados, es decir, consigue detectar ataques de tipo Inyección SQL, ejecución de comandos, etc. Este tipo de ataques, pasan completamente desapercibidos para el Firewall.

Este comportamiento, lo consiguen ya que disponen de un paquete de firmas que, son capaces de encontrar patrones de ataque conocidos en los paquetes que analizan. Estos patrones pueden ser de los siguientes tipos principalmente: volumetría anómala de paquetes, conjuntos de palabras sospechosas (que pueden estar en texto plano o codificadas según el estándar HTML), maliciosos conocidos, etc.

Este conjunto de patrones de detección, pueden estar configurados de diferentes maneras que nos permiten, entre otras cosas, generar eventos que más tarde serán correlados, e incluso con la configuración adecuada, y asegurándose que no se cortan comunicaciones legítimas derivadas de falsos positivos, impedir que los paquetes que sean sospechosos no lleguen al equipo que sirve los contenidos. De esta manera, se previene la integridad del sistema.

Para este desempeño se necesita una máquina con las siguientes características:

- 2 CPU
- 4 GB de RAM
- 10 GB de espacio en disco

Utilizaremos en la maqueta presentada una máquina Linux encargada de enrutar el tráfico desde el Firewall perimetral hasta el equipo que contiene el frontal web con el software libre “Suricata” configurado en su versión inline (basado en la tecnología Snort) para que actúe en modo IPS.

Una de las razones principales de por la que se ha elegido este software para la prevención de intrusiones es que se integra de manera muy sencilla con la plataforma de correlación que se presenta en el epígrafe 4.2. Por este motivo, y que la integración con el firewall IPTables que lleva

instalada la máquina donde se instalará el aplicativo, hace que sea la opción elegida para este desempeño.

### **3.2 Servicio de correlación**

Los correladores de eventos son un software que nos permite interrelacionar eventos de diferentes tecnologías para detectar comportamientos anómalos desde o sobre un equipo en concreto.

Esta tecnología, utiliza de manera reactiva toda la información que recibe de los equipos que componen la plataforma de seguridad, así como los equipos que sirven contenidos de una manera u otra para generar alertas que tengan asociada algún tipo de acción como respuesta a las mismas.

Además, como esta máquina actuará como dispositivo de almacenamiento de Logs, debido a que los demás dispositivos, tienen un espacio en disco reducido, será necesario que disponga de almacenamiento suficiente como para almacenar los eventos recogidos por los dispositivos un tiempo fijado que dependerá de las necesidades.

El software elegido para el servicio de correlación de la plataforma es la versión open source de la empresa Alienvault denominada OSSIM (Open Source SIEM) que cumple a la perfección con las necesidades requeridas para estas pruebas y además se integra de manera nativa con Suricata, opción elegida para el sistema de prevención de intrusiones.

La tecnología de correlación requiere mayores especificaciones que la mayoría de sistemas de red debido a la gran cantidad de operaciones que lleva a cabo con cada uno de los eventos de seguridad que llegan hasta este servidor. Estas acciones son las siguientes:

#### **1) Almacenamiento en texto de los eventos recibidos.**

Lo primero que hace este tipo de dispositivos es almacenar un histórico de los eventos recibidos de los dispositivos que se encuentran monitorizados a nivel de seguridad. Esto es necesario por si en algún momento es necesario realizar un forense de la información recibida.

#### **2) Normalización de los eventos recibidos de diferentes dispositivos.**

Debido a que cada fabricante genera unas trazas únicas de lo que ocurre en sus dispositivos, es necesario llevar a cabo un proceso de unificación de los eventos recolectados a fin de



tener la información en el mismo formato sea cual se sea la naturaleza de la fuente que nos envíe la información.

Este proceso, que en OSSIM se realiza mediante la aplicación de una serie de procesos compuestos por expresiones regulares permite transformar la información recibida a una versión estandarizada para la plataforma de modo que el motor de correlación sea capaz de trabajar con todos los eventos de la misma manera.

Además gracias a un sistema de diccionarios, el dispositivo es capaz de asignar la tecnología recibida así como la tipología de los eventos recolectados para poder generar reglas de correlación (las veremos más adelante) que los utilicen y poder generar alertas relacionadas.

### **3) Inserción en base de datos de los datos normalizados.**

Los eventos normalizados por el dispositivo son introducidos en la base de datos del sistema para poder ser consultados y analizados vía interfaz web.

Esto supone una mejora importante para el administrador del sistema ya que le permite disponer de la información de todos los elementos de la red en un único aplicativo, lo que implica una mejora en el tiempo de respuesta si nos encontramos ante un incidente de seguridad.

### **4) Correlación de los eventos recibidos para interrelacionar datos entre los dispositivos.**

Es posible en aplicativos de este tipo generar lo que se conoce como reglas de correlación que permite la generación de alertas automáticas. Estas reglas son capaces de encontrar patrones automáticamente en los eventos que recibe el correlador.

Un ejemplo de una regla de correlación sería:

- Se detecta un escaneo de puertos sobre una máquina determinada.
  - Acto seguido se detecta un intento de acceso a algún servicio publicado de la máquina atacada
    - Se genera una alerta de correlación ya podemos estar siendo víctimas de un intento de intrusión.

**5) Generación de alertas.**

En función de las reglas de correlación definidas el sistema es capaz de filtrar relacionar los eventos que le van llegando y decidir qué hacer con dicha información. La mayoría de las veces será únicamente generar una alerta para avisar al administrador de seguridad de alguna situación anómala pero hay ocasiones, para comportamientos muy localizados, en las que es necesario aplicar alguna acción en alguno de los dispositivos de red para prevenir futuros daños.

**6) Remediaciones. Aplicación de actuaciones automáticas.**

La actuación automática más habitual en los sistemas es la aplicación automática de un bloqueo de las comunicaciones a nivel de FW de un origen sospechoso.

**7) Generación de informes automáticos.**

Podemos generar informes del estado de la seguridad en un periodo de tiempo determinado, presentando, el número de alertas generadas, los orígenes de alertas más activos, los activos protegidos que reciben el mayor número de alertas, etc.

Así para llevar a cabo todas estas funciones, se necesita un equipo lo suficientemente potente y con la capacidad de almacenamiento suficiente. Para la realización de estas pruebas, debido a que nos encontramos en un entorno totalmente controlado y que no nos encontramos expuestos a Internet, con una máquina con las siguientes especificaciones es suficiente:

- 4 CPU
- 8 GB de RAM
- 100 GB de espacio en disco

### **3.3 Máquina vulnerable**

Se utilizará para las pruebas, un equipo Linux con un portal web vulnerable a diferentes tipos de ataques sobre el que se realizarán las pruebas.

La aplicación web elegida es la Damn Vulnerable Web Application que principalmente está destinada a la demostración de los comportamientos de los ataques web clásicos con diferentes niveles de seguridad a nivel software.

Gracias a este equipo, será posible ver cómo sin la utilización de algunos de los elementos de detección que estamos incluyendo en el sistema, es fácil la extracción de información sensible como puede ser usuarios y contraseñas, acceder a archivos críticos del sistema o generar modificaciones en el código de la página.

Esta máquina tendrá las siguientes especificaciones:

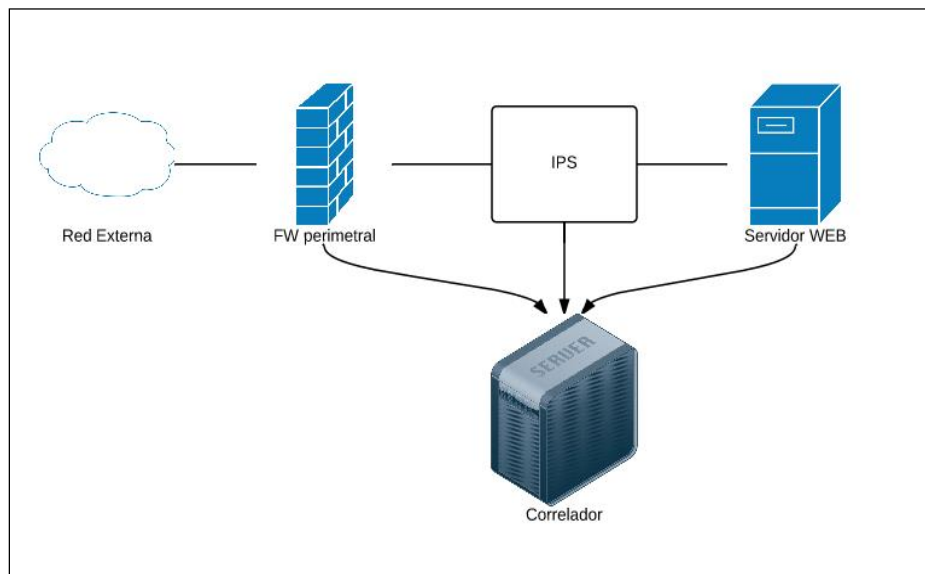
- 1 CPU
- 512 MB de RAM
- 5 GB de espacio en disco

Después de ver las necesidades físicas de cada uno de los dispositivos integrados en la plataforma y debido a que toda la maqueta será montada en un entorno virtual, las características de la máquina que necesitamos son las siguientes:

- 10 CPU
- 24 GB de RAM
- 200 GB de espacio en disco

### **3.4 Esquema de la plataforma.**

En figura 3-1 podemos ver la disposición que tendrán las máquinas implicadas en la maqueta de forma general:



**Figura 3-1: Esquema general de la plataforma**

### **3.5 Conclusiones**

En este capítulo, se han visto las necesidades en el diseño de la plataforma de publicaciones web a nivel funcional y de seguridad. Con las capas de seguridad propuestas es posible generar un entorno con un mínimo de seguridad para mantener a salvo la información contenida en la publicación web a nivel de Back-End.

Tras el análisis del diseño de la plataforma, se pasa en el siguiente capítulo a la explicación del montaje de la misma en un entorno virtualizado con lo que se consigue un entorno completamente funcional para realizar las pruebas de intrusión en la plataforma.

## **4 Integración de la plataforma**

---

Tras el análisis sobre el diseño de la plataforma, se desarrolla el proceso de creación del laboratorio para su completo funcionamiento así como las configuraciones establecidas en las diferentes máquinas del entorno. Para la realización de esta integración es necesario definir los siguientes puntos que son, a nivel de integración, los mínimos necesarios:

- Software base para la maqueta
- Configuración de red
- Configuración de los dispositivos

### **4.1 Software base para la maqueta**

Para la virtualización de la plataforma el Software ESXi de vmware que nos permite la generación de máquinas virtuales y creación de VLANs estancas dentro de una misma máquina.

Este software, en su versión gratuita, nos ofrece todas las funcionalidades necesarias para la realización de nuestra maqueta de una manera muy sencilla.

### **4.2 Configuración de Red**

En esta sección se presentan las necesidades de red a nivel de red de la plataforma así como las configuraciones de cada uno de los dispositivos desplegados que se han fijado en la fase de diseño.

Para ello, primero se analizan las necesidades de redes aisladas que son necesarias para el correcto funcionamiento de la plataforma y después se definen los direccionamientos de cada una de las máquinas dentro de esas redes en función de sus necesidades de conexión.

#### **4.2.1 Definición de las redes implicadas en la plataforma.**

A continuación se definirán las redes necesarias para el montaje de la plataforma de seguridad en el escenario presentado en la figura 3-1. Según las características de la plataforma, y al ser ésta virtual, se encuentran las siguientes necesidades de interconexión:

- Es necesario crear un canal de comunicaciones seguro y aislado entre la red externa y el equipo que se encarga de servir contenidos web. Debido a esto, se necesita crear una red que nos permita esta interconexión.

## *Diseño, implementación y análisis de un sistema de detección y respuesta activa*

- Por otro lado, se necesita intercalar entre el entre la red externa y el servidor web el IPS y como la plataforma se encuentra alojada en un entorno virtual en lugar de en un entorno físico en el que se podría pasar el tráfico a través del IPS de una forma física, necesitamos generar dos redes independientes a ambos lados del mismo. Por este motivo, se presenta la necesidad de dividir el canal seguro en dos.
- Así mismo, para la gestión de todos los dispositivos se hace necesario disponer de una red independiente a la que presta servicio que denominaremos red de gestión y en la que se encuentran conectados en paralelo todos los dispositivos de la red.

Por lo expuesto anteriormente se requiere la creación de tres redes internas que nos permitan llevar a cabo todas las comunicaciones expuestas anteriormente:

- Red de *servicio*

Esta red será la destinada a la conexión del IPS con el firewall perimetral y le asignamos el siguiente direccionamiento: 10.147.105.0/25.

- Red de *Front-end*

En este caso nos encontramos con la red a la que se conectarán el IPS y la máquina de publicaciones y le asignamos el siguiente direccionamiento: 10.147.106.0/25.

- Red de *Gestión*

En esta red se conectarán las interfaces de administración de todos los dispositivos para que sean accesibles para ser gestionados por el equipo de administración. Le asignamos el siguiente direccionamiento: 10.147.104.0/25. Se asigna un direccionamiento más grande para la red de gestión ya que tiene que albergar equipos que se encuentran tanto en la red de Servicio, en la red de Front-end y equipos que únicamente se encuentran en la red de gestión.

## **4.2.2 Direccionamiento de los dispositivos.**

### **4.2.2.1 Firewall perimetral**

Este dispositivo necesita tres interfaces de red para la conexión con las siguientes redes:

- Conexión externa

Conexión con el exterior de la plataforma: 172.18.169.11.

- Red de servicio

Conexión con la red en la que se encuentra el IPS: 10.147.105.1.

- Red de gestión

Conexión de administración del equipo y comunicación con el correlador de eventos:  
10.147.104.2.

### **4.2.2.2 Sistema de prevención de intrusiones**

Este equipo necesita un total de tres interfaces de red para permitir las siguientes conexiones:

- Red de servicio

Conexión con la red en la que se encuentra el Firewall perimetral: 10.147.105.11.

- Red de Front-end

Conexión con la red en la que se encuentra el equipo que tiene la publicación web:  
10.147.106.11.

- Red de gestión

Conexión para la administración del equipo y comunicación con el correlador de eventos: 10.147.104.11.

#### 4.2.2.3 Máquina vulnerable

Este equipo necesita un mínimo de dos interfaces de red para ser capaz de tener las siguientes comunicaciones:

- Red de Front-end

Conexión con la red en la que se encuentra el IPS: 10.147.106.20.

- Red de gestión

Conexión de administración del equipo y comunicación con el correlador de eventos: 10.147.104.20.

#### 4.2.2.4 Correlador de eventos.

Este elemento se encuentra aislado de las redes de servicio con lo que únicamente necesita una única interfaz de red para conectarse con la siguiente red:

- Red de gestión

Conexión de administración del equipo y conexión para la recolección de información de los demás dispositivos de la red: 10.147.104.10.

#### 4.2.2.5 Conexión final de la infraestructura

En el figura 4-1 se muestra como queda configurada a nivel de red la infraestructura.

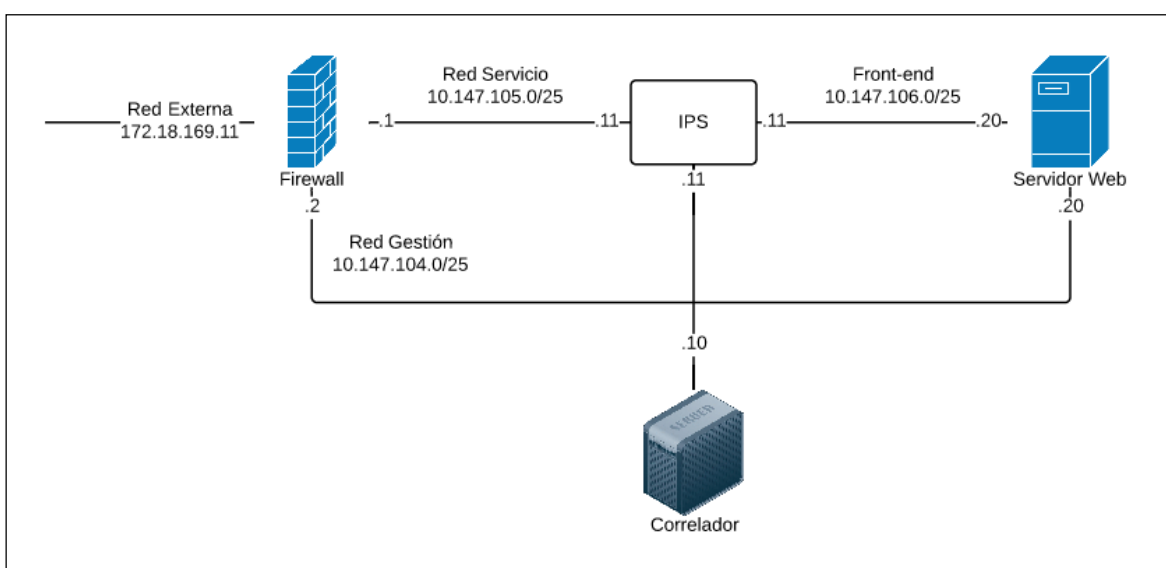


Figura 4-1: Esquema detallado de la plataforma



### **4.3 Configuración de los dispositivos**

Una vez vistos los requerimientos de conectividad de la plataforma, se analizan las diferentes configuraciones de los dispositivos implicados para el correcto funcionamiento de la misma.

En las siguientes subsecciones, se hace referencia a los paquetes software instalados en cada uno de los dispositivos así como las configuraciones aplicadas en cada uno de ellos para proporcionar el correcto funcionamiento de la plataforma.

#### **4.3.1 Firewall**

Este dispositivo se encuentra entre en la frontera de la red interna y la red externa con lo a todos los efectos, es la conexión con el exterior de todos los dispositivos implicados en la plataforma.

El software escogido para este desempeño es la distribución de software libre “Endian Firewall” que nos permite llevar a cabo todas las configuraciones necesarias para llevar a cabo la publicación de servicios así como proveer de conectividad a todos los dispositivos de la plataforma.

Los servicios principales que nos proporciona este dispositivo son los siguientes:

- Firewall

Este servicio se encarga de proporcionarnos la primera capa de seguridad en la plataforma permitiendo únicamente las conexiones entrantes que van dirigidas contra la publicación Web que se sirve por la plataforma. Para esto, se genera una traducción de direcciones de red (regla de NAT) desde la dirección 172.18.169.11 por el puerto 8081 (servicio publicado) hacia la IP 10.147.106.20 puerto 80 que es donde se encuentra realmente el frontal web publicado.

- Proxy y Servidor de nombre (DNS)

Esta dupla de servicios se encarga de proporcionar navegación a Internet a las máquinas de la red interna a través de la red de gestión. Estos servicios son necesarios para llevar a cabo las actualizaciones en los paquetes de software instalados en los diferentes dispositivos.

- Servidor de tiempo (NTP)

Debido a que incluimos en la plataforma el servicio de correlación, es necesario que todas las máquinas estén correctamente sincronizadas para poder llevar a cabo el proceso de correlación. De esta manera, todos los equipos sincronizarán su hora con el Firewall que a su vez se sincroniza periódicamente con servidores públicos de NTP.

- Servidor de red privada virtual (VPN)

Este dispositivo nos da la opción también de proporcionarnos conectividad vía VPN con seguridad de doble factor de autenticación para la conexión de los administradores a la red de gestión.

### **4.3.2 Sistema de prevención de intrusiones**

Para el despliegue del IPS se ha montado una máquina con el paquete de Alienvault OSSIM tipo Sensor que nos permite las siguientes funcionalidades:

- Al tratarse de una distribución Linux basada en Debian, nos permite utilizar las funciones básicas de este sistema operativo. De este modo podemos utilizar el equipo para generar el enrutamiento necesario entre la red de servicio y la red de Front-end modificando el parámetro del Kernel que permite realizar el reenvío de paquetes entre interfaces para hacerlo funcionar como un router.
- Integración nativa con Suricata el cual se ha configurado en modo en línea (*Inline*). Esto permite, de una manera bastante sencilla, poder cortar tráfico de manera pasiva en función de la condición de los paquetes a nivel de aplicación y recolección de los eventos generados por Suricata para su posterior correlación con los eventos generados con el resto de dispositivos.
- Es necesario para el correcto funcionamiento de Suricata en modo *Inline*, configurar el Firewall de la máquina (IPTables) para que el tráfico entre las interfaces eth1 (red de servicio) y eth2 (red de Front-end) pase por Suricata antes de llevar a cabo el reenvío de los paquetes y así poder ejecutar acciones en ellos.

### **4.3.3 Máquina vulnerable**

Esta máquina se ha montado con el sistema operativo Debian 7 y se ha instalado el paquete LAMP que incluye Apache, MySQL y PHP para hacer correr el aplicativo “Damn Vulnerable Web Application”. De esta manera se consigue una máquina capaz de servir una página web la cual es vulnerable a ciertos ataques.

Gracias a estas configuraciones, se pueden analizar las peticiones hacia el servidor y las respuestas desde el mismo para ver qué ocurre a nivel de paquetes cuando se realizan ataques ya que al ser vulnerable el aplicativo en algunas secciones, va a emitir respuestas relacionadas con estas vulnerabilidades.

### **4.3.4 Correlador de eventos**

Este equipo ha sido provisionado con la versión todo en uno de Alienvault en su versión gratuita (OSSIM). Se han configurado las alertas de correlación necesarias para llevar a cabo la correlación de las diferentes fuentes de información.

Por otro lado, este equipo también se ha configurado a nivel de Syslog, mediante el aplicativo “Rsyslog” que se utiliza para la recepción de los logs de los diferentes dispositivos, para actuar como equipo concentrador de Logs. Esta funcionalidad nos permite guardar un histórico en texto no indizado de los eventos generados por los distintos dispositivos implicados en la seguridad y servicio de la red.

También se ha configurado en el equipo el módulo de Alienvault que se encarga de la normalización de todos los eventos recibidos de los diferentes dispositivos para su posterior inyección en la base datos del correlador de eventos.

## **4.4 Conclusiones**

En este capítulo hemos visto como llevar a cabo la integración real del sistema analizado y diseñado en los capítulos anteriores. Se han analizado las necesidades reales de conectividad así como el software necesario y sus configuraciones para establecer un sistema completamente funcional que nos permita realizar la batería de pruebas que se llevan a cabo en el capítulo 5.



## **5 Pruebas y resultados**

---

En este capítulo se hace uso de la maqueta montada con las tecnologías descritas durante todo el desarrollo del proyecto para demostrar las diferencias de seguridad activando y desactivando diferentes capas de seguridad de las que se dispone en la plataforma.

Por un lado, se presentan los resultados obtenidos en las pruebas de intrusión realizadas sobre el portal web publicado teniendo únicamente en modo detección el sistema de prevención de intrusiones y posteriormente, después de haber realizado el análisis correspondiente a de las alertas detectadas, se activará la función de bloqueo en el sistema de prevención de intrusiones para observar los niveles de seguridad que se pueden alcanzar con un dispositivo de seguridad de este tipo que haya sido configurado convenientemente.

Los ataques que se han utilizado para analizar los niveles de detección y mitigación son los siguientes: Inyección SQL, Inyección de comandos, Inclusión de ficheros y Secuencias de comandos en sitios cruzados. Para analizar el funcionamiento de los diferentes ataques nos apoyamos en la aplicación web vulnerable que se ha instalado en la maqueta.

### **5.1 Análisis de seguridad sin bloqueo de conexiones**

En este apartado se procede al análisis de los diferentes tipos de ataques para analizar su comportamiento y presentar los niveles de detección que nos brinda la plataforma de seguridad que se ha implementado a lo largo de este documento.

Para la demostración de los comportamientos de los diferentes tipos de ataque, se utiliza el portal web vulnerable. Este portal nos brinda la posibilidad de analizar el código fuente de la página de sus secciones vulnerables con lo que es relativamente sencillo explotar dichas vulnerabilidades y ver las implicaciones que ello supone.

Por otro lado, para las pruebas de detección nos apoyamos en el análisis de los eventos generados por el sistema de prevención de intrusiones, en modo análisis, y los eventos que nos proporciona el propio servidor web. Gracias a estas dos fuentes de información es posible llevar a cabo la correlación de los mismos y así detectar comportamientos no deseados sobre nuestra plataforma.

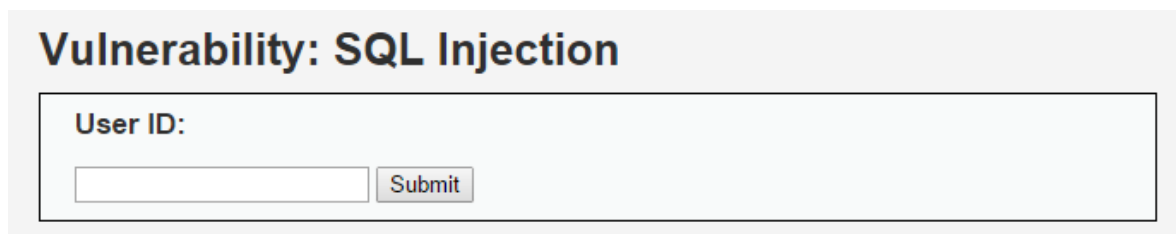
Es importante apuntar que, con esta configuración, no se procede a la mitigación de los análisis de forma activa con lo que si queremos detener un ataque, la única herramienta de la que disponemos es realizar un bloqueo de las comunicaciones desde la máquina atacante a nivel de firewall de manera reactiva.

### 5.1.1 Inyección SQL

Este tipo de ataques se basa en la explotación de una vulnerabilidad que exista en algún campo de la página web que estemos atacando. Esta vulnerabilidad normalmente consiste en no tener una correcta validación de los datos introducidos para llevar a cabo una consulta a una base de datos de la parte del Back-end de la aplicación. De este modo se puede modificar la consulta que se va a realizar a la base de datos y nos permite extraer información sensible que de otra forma no sería accesible.

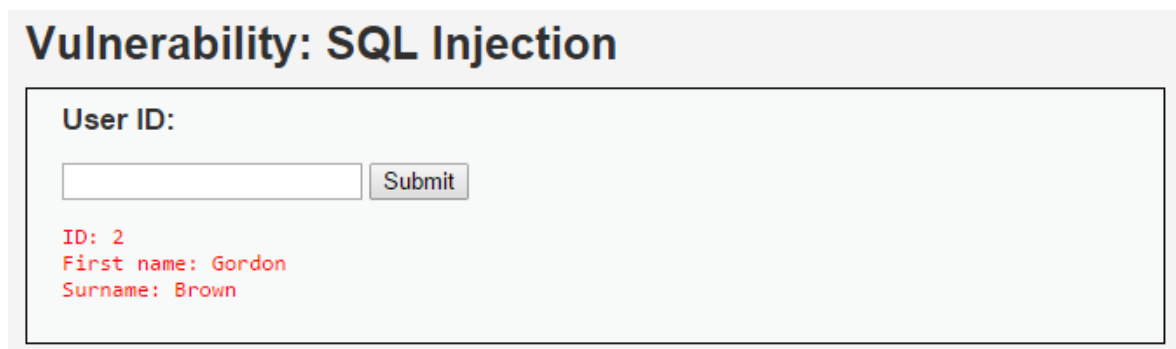
#### 5.1.1.1 Explotación de la vulnerabilidad

En la aplicación web vulnerable en la que nos encontramos, uno de los campos que hay disponibles para introducir información y esperar resultados de una búsqueda a la base de datos es vulnerable.



**Figura 5-1: Formulario SQL injection**

Este campo introduciendo el identificador de un usuario, nos devuelve el nombre y los apellidos del mismo como se puede apreciar en la siguiente figura.



**Figura 5-2: Uso Formulario SQL injection**

Si se analiza el código de la página, se puede ver que la información que se introduce en el campo "User ID" se pasa automáticamente, sin validar de manera alguna, a la consulta que se hace a la base de datos, lo que hace que este campo sea vulnerable a la Inyección SQL.

## SQL Injection Source

```
<?php
if(isset($_GET['Submit'])){
    // Retrieve data
    $id = $_GET['id'];
    $getid = "SELECT first_name, last_name FROM users WHERE user_id = '$id'";
    $result = mysql_query($getid) or die('<pre>' . mysql_error() . '</pre>');
    $num = mysql_numrows($result);
    $i = 0;
    while ($i < $num) {
        $first = mysql_result($result,$i,"first_name");
        $last = mysql_result($result,$i,"last_name");
        echo '<pre>';
        echo 'ID: ' . $id . '<br>First name: ' . $first . '<br>Surname: ' . $last;
        echo '</pre>';
        $i++;
    }
}
?>
```

Figura 5-3: Código página vulnerable a SQL injection

Debido a esta vulnerabilidad, podemos extraer los nombres de usuario y sus contraseñas de una manera sencilla. Para extraer dicha información, es suficiente introducir la siguiente sentencia en el campo "User ID": ***' union select first\_name, password from users #***. Como se observa en la figura 5-4 la información que nos muestra la página ya no son los nombres y apellidos de los usuarios, si no los nombres y las contraseñas.

## Vulnerability: SQL Injection

User ID:

```
ID: ' union select first_name, password from users #
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: ' union select first_name, password from users #
First name: Gordon
Surname: e99a18c428cb38d5f260853678922e03

ID: ' union select first_name, password from users #
First name: Hack
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: ' union select first_name, password from users #
First name: Pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: ' union select first_name, password from users #
First name: Bob
Surname: 5f4dcc3b5aa765d61d8327deb882cf99
```

Figura 5-4: Explotación SQL injection

### 5.1.1.2 Análisis de la detección

Este tipo de ataques en el entorno en el que nos encontramos, son relativamente sencillos de detectar debido a que la petición que se hace al servidor es de tipo GET tal y como podemos ver en el siguiente registro del servidor Apache:

```
Feb 17 18:42:15 vulnerable apache: 172.24.121.134 -- [17/Feb/2015:18:42:15 +0100] "GET
/vulnerabilities/sqli/?id=%27+union+select+first_name%2C+password+from+users+%23&Submit=Submit HTTP/1.1"
200 5354 "-" "Mozilla/5.0 (Windows NT 6.3; WOW64; Trident/7.0; rv:11.0) like Gecko"
```

De esta manera, gracias al paquete de firmas de snort podemos detectar los siguientes patrones sospechosos en esta petición:

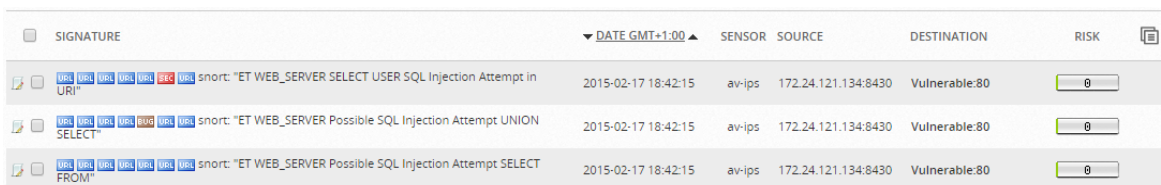
- Que contenga la palabra **union** y la palabra **select**
- Que contenga la palabra **select** y la palabra **from**
- Que contenga la palabra **select** y la palabra **user**

Si se cumple cualquiera de estas condiciones, el dispositivo de detección de intrusiones cataloga el paquete como un posible intento de inyección SQL, que puede ser exitoso o no. No obstante a nosotros, como encargados de la gestión de la seguridad en la plataforma, no nos



importa si el ataque tiene éxito o no si no detectar el propio intento de explotación de una vulnerabilidad.

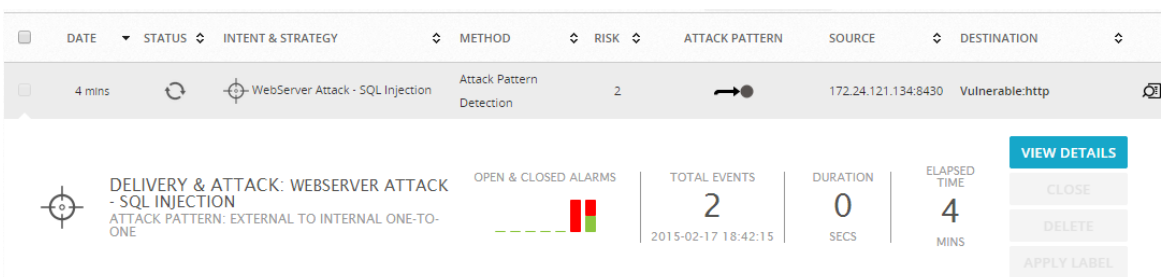
Ahora analizando la información proporcionada por el sistema de detección de intrusiones vemos que esta petición ha generado 3 eventos relacionados con posibles intentos de ataque de inyección SQL que se ven en la figura 6-x:



SIGNATURE	DATE GMT+1:00	SENSOR	SOURCE	DESTINATION	RISK
snort: "ET WEB_SERVER SELECT USER SQL Injection Attempt in URI"	2015-02-17 18:42:15	av-ips	172.24.121.134:8430	Vulnerable:80	0
snort: "ET WEB_SERVER Possible SQL Injection Attempt UNION SELECT"	2015-02-17 18:42:15	av-ips	172.24.121.134:8430	Vulnerable:80	0
snort: "ET WEB_SERVER Possible SQL Injection Attempt SELECT FROM"	2015-02-17 18:42:15	av-ips	172.24.121.134:8430	Vulnerable:80	0

Figura 5-5: Eventos generados por SQL injection

Estos eventos detectados han generado una alerta que nos avisa de un ataque de tipo inyección SQL sobre el equipo vulnerable.



4 mins

WebServer Attack - SQL Injection

Attack Pattern Detection

RISK: 2

SOURCE: 172.24.121.134:8430

DESTINATION: Vulnerable:http

DELIVERY & ATTACK: WEBSERVER ATTACK - SQL INJECTION

ATTACK PATTERN: EXTERNAL TO INTERNAL ONE-TO-ONE

OPEN & CLOSED ALARMS: 1 Open, 1 Closed

TOTAL EVENTS: 2 (2015-02-17 18:42:15)

DURATION: 0 SECS

ELAPSED TIME: 4 MINS

VIEW DETAILS, CLOSE, DELETE, APPLY LABEL

Figura 5-6: Alerta de SQL injection

Si analizamos en detalle la alerta podemos observar, como se ha generado a partir de los eventos detectados en el módulo de detección de intrusiones.

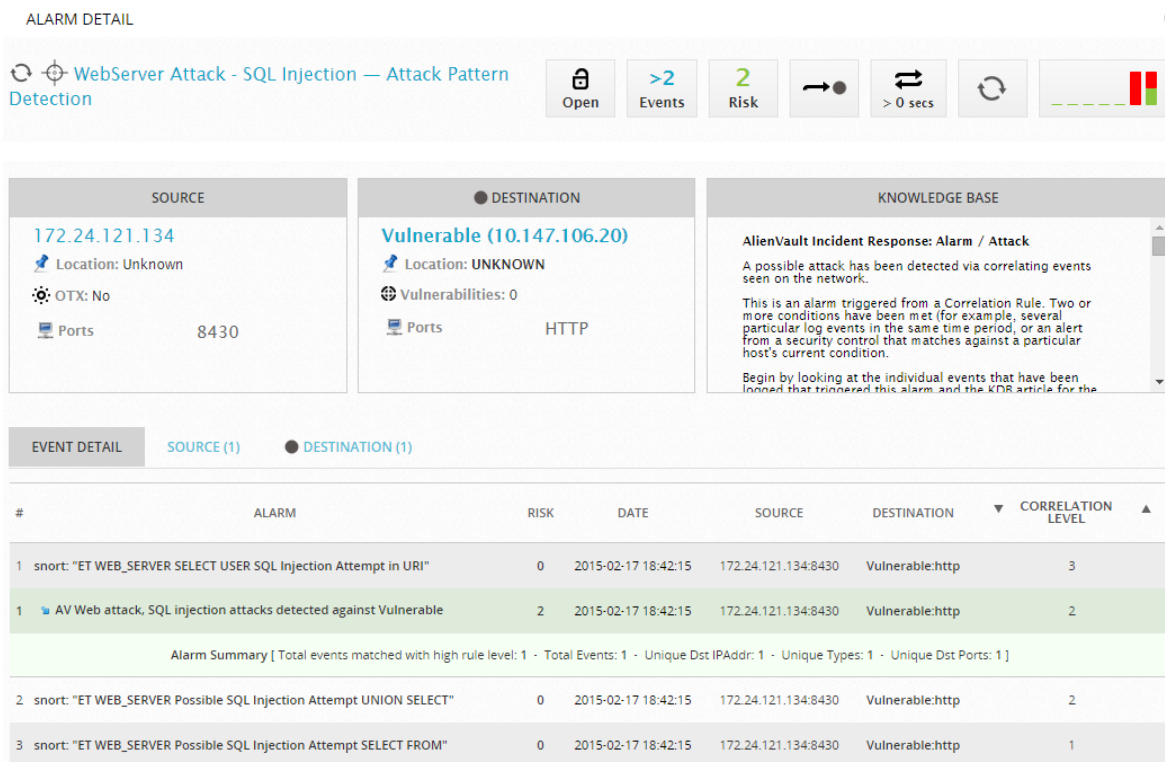


Figura 5-7: Detalle de alerta de SQL injection

Como se puede observar en la figura anterior los eventos generados por el módulo de detección de intrusiones han generado la alerta de correlación encargada de detectar este tipo de ataques de inyección SQL. La regla de correlación es una regla compuesta por tres niveles que se encargan de lo siguiente:

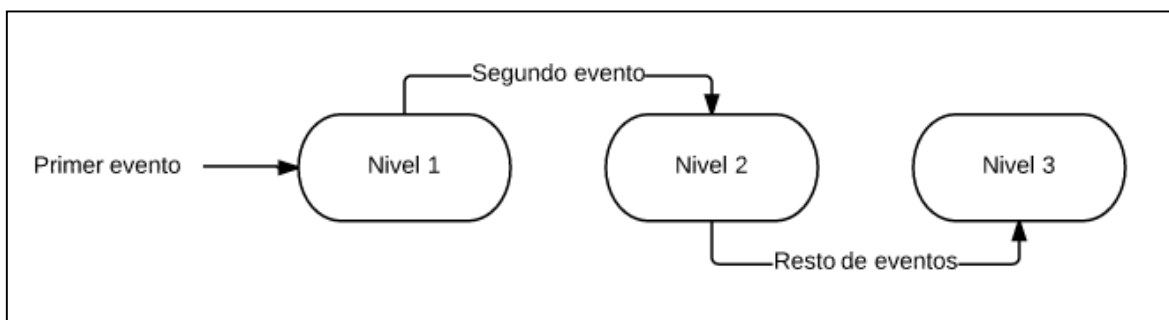


Figura 5-8: Esquema regla de correlación de SQL injection

- El primer nivel detecta un evento catalogado como un posible intento de inyección SQL
- El segundo nivel detecta un segundo evento de posible inyección SQL entre el mismo origen y el mismo destino que fueron detectados en el primer nivel para confirmar el ataque
- El tercer nivel de correlación es un nivel acumulativo que se encarga de recoger todos los demás eventos de este tipo en un tiempo determinado, una hora en este caso, con la finalidad de no generar un número demasiado elevado de alertas cuando este tipo de ataques son generados por herramientas de explotación automáticas.

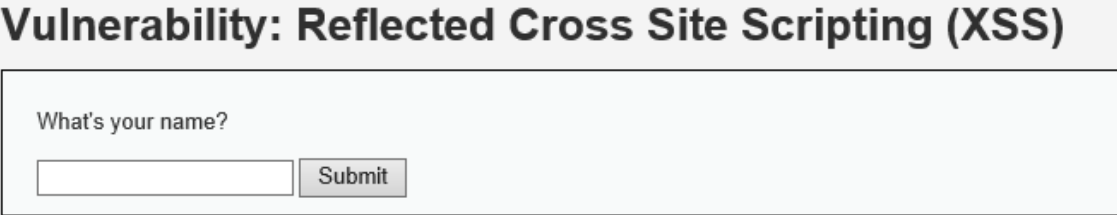
Después de este análisis, podemos concluir que este tipo de ataques de inyección SQL son detectados **exitosamente** por la plataforma y gracias a la generación de esta alerta de correlación se podría, de manera reactiva, generar algún tipo de acción manual o automática para protegernos de futuros ataques desde dicho origen. No obstante, como hemos podido observar en la explicación de la vulnerabilidad, el ataque ha sido satisfactorio para el atacante y ha conseguido extraer la información antes de que hayamos podido hacer nada.

## 5.1.2 Secuencias de comandos en sitios cruzados (XSS)

Este tipo de ataque web se basa en la explotación de una vulnerabilidad sobre algunos de los parámetros de una petición web. Estos parámetros vulnerables son utilizados para la construcción de la página en cuestión y sin una correcta comprobación de los datos introducidos pueden suponer una brecha en la seguridad del sitio web.

### 5.1.2.1 Explotación de la vulnerabilidad

Para llevar a cabo la explotación de este tipo de vulnerabilidades, es necesario introducir una secuencia de comandos en alguno de los parámetros de la página web que se vaya a reutilizar para la composición de la misma más adelante. En la aplicación web vulnerable en la que nos encontramos, existe uno de estos parámetros que se presenta en la figura siguiente.




**Vulnerability: Reflected Cross Site Scripting (XSS)**

What's your name?

**Figura 5-9: Formulario XSS**

Si introducimos un valor en el campo y procedemos a hacer click en el botón la página nos contesta saludandonos.



**Vulnerability: Reflected Cross Site Scripting (XSS)**

What's your name?

Hello Alejandro

**Figura 5-10: Uso Formulario XSS**

Hasta aquí el funcionamiento es correcto, pero veamos el código fuente de la página para ver las implicaciones que tiene esta funcionalidad.

## Reflected XSS Source

```
<?php
if(!array_key_exists ("name", $_GET) || $_GET['name'] == NULL || $_GET
['name'] == ''){
    $isempty = true;
} else {
    echo '<pre>';
    echo 'Hello ' . $_GET['name'];
    echo '</pre>';
}
?>
```

Figura 5-11: Código de la página vulnerable a XSS

Se puede observar que se utiliza el valor del parámetro “name” directamente en para generar la página siempre y cuando contenga algún valor. Este hecho es una mala praxis en la programación web ya que hace nuestra página completamente vulnerable a este tipo de ataques. Si en lugar de introducir nuestro nombre, en el campo de entrada, introducimos una secuencia de comandos, como puede ser `<script>alert('prueba')</script>`, podemos comprobar que en lugar de saludarnos, con la cadena introducida, la página ejecuta el código que acabamos de introducir como puede apreciarse en la siguiente figura.

## Vulnerability: Reflected Cross Site Scripting (XSS)

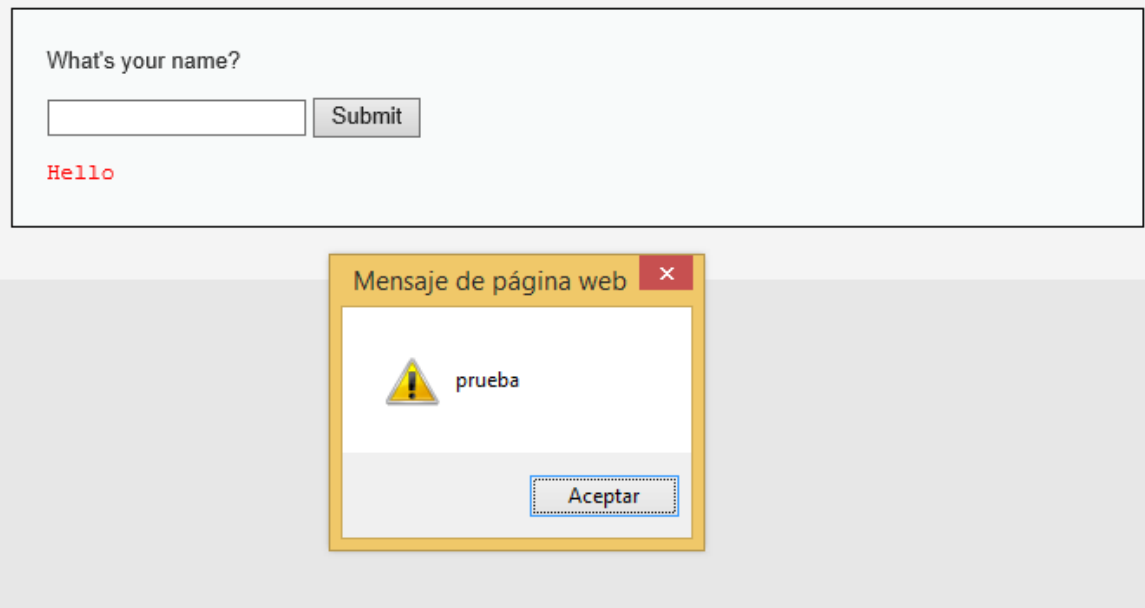


Figura 5-12: Explotación de la vulnerabilidad de XSS

Esta prueba nos confirma que este campo es vulnerable a Cross-Site Scripting con lo que supone una brecha en la seguridad de la página y la posibilidad de que alguien malintencionado pueda ejecutar código malicioso en nuestra página.

### 5.1.2.2 Análisis de la detección

Para la detección de este tipo de ataques, se procede al análisis de los paquetes que atraviesan el sistema de prevención de intrusiones y las peticiones que se hacen al servidor web, que son las dos fuentes de información que nos proporcionan visibilidad contra estos ataques. A continuación se muestra la petición que ha sido recibida por el servidor web:

```
Feb 18 21:18:35 vulnerable apache: 172.24.121.134 - - [18/Feb/2015:21:18:35 +0100] "GET /vulnerabilities/xss_r/?name=%3Cscript%3Ealert%28%27prueba%27%29%3C%2Fscript%3E HTTP/1.1" 200 4502 "http://172.18.169.11/vulnerabilities/xss_r/?name=Alejandro" "Mozilla/5.0 (Windows NT 6.3; WOW64; Trident/7.0; rv:11.0) like Gecko"
```

De esta forma, gracias al paquete de firmas de snort del que dispone el sistema de prevención de intrusiones, es sencillo detectar el patrón `</script>`, que nos indica el final de una secuencia de comandos en un fichero HTML, señal inequívoca de que se ha intentado la ejecución de comandos de manera no legítima en la página.

Se han realizado un total de tres peticiones iguales a la página y analizando la información proporcionada por el sistema de prevención de intrusiones vemos que se han generado los siguientes eventos en el sistema de correlación:

SIGNATURE	DATE GMT+1:00	SENSOR	SOURCE	DESTINATION	RISK
snort: "ET WEB_SERVER Script tag in URI, Possible Cross Site Scripting Attempt"	2015-02-18 21:18:35	av-ips	172.24.121.134:14016	Vulnerable:80	0
snort: "ET WEB_SERVER Script tag in URI, Possible Cross Site Scripting Attempt"	2015-02-18 21:16:32	av-ips	172.24.121.134:14075	Vulnerable:80	0
snort: "ET WEB_SERVER Script tag in URI, Possible Cross Site Scripting Attempt"	2015-02-18 21:14:02	av-ips	172.24.121.134:13978	Vulnerable:80	0

Figura 5-13: Eventos detectados de XSS

Estos eventos generados en el sistema de prevención de intrusiones catalogados como posible intento de Cross-Site Scripting, han generado la una alerta en el sistema que nos advierte de un ataque de tipo XSS sobre la máquina vulnerable.

8 mins

WebServer Attack

XSS

2

172.24.121.134:14075

Vulnerable:http

DELIVERY & ATTACK:  
WEBSERVER ATTACK  
ATTACK PATTERN: EXTERNAL TO  
INTERNAL ONE-TO-ONE

OPEN & CLOSED  
ALARMS

TOTAL EVENTS  
2  
2015-02-18  
21:16:32

DURATION  
5  
MINS

ELAPSED  
TIME  
9  
MINS

VIEW DETAILS

CLOSE

DELETE

APPLY LABEL

Figura 5-14: Alerta generada de XSS

Analizando la alerta en profundidad, podemos observar como los eventos detectados han generado la alerta con sus diferentes niveles de riesgo.

ALARM DETAIL

WebServer Attack — XSS

Open Events Risk > 5 mins

SOURCE	DESTINATION	KNOWLEDGE BASE
172.24.121.134 Location: Unknown OTX: No Ports: 13978 14075	Vulnerable (10.147.106.20) Location: UNKNOWN Vulnerabilities: 0 Ports: HTTP	AlienVault Incident Response: Alarm / Attack A possible attack has been detected via correlating events seen on the network. This is an alarm triggered from a Correlation Rule. Two or more conditions have been met (for example, several particular log events in the same time period, or an alert from a security control that matches against a particular host's current condition. Begin by looking at the individual events that have been logged that triggered this alarm, and the KDR article for the

EVENT DETAIL SOURCE (1) DESTINATION (1)

#	ALARM	RISK	DATE	SOURCE	DESTINATION	CORRELATION LEVEL
1	snort: "ET WEB_SERVER Script tag in URI, Possible Cross Site Scripting Attempt"	0	2015-02-18 21:18:35	172.24.121.134:14016	Vulnerable:http	3
1	AV Web attack, XSS attacks detected against Vulnerable	2	2015-02-18 21:16:32	172.24.121.134:14075	Vulnerable:http	2
Alarm Summary [ Total events matched with high rule level: 1 - Total Events: 1 - Unique Dst: IPAddr: 1 - Unique Types: 1 - Unique Dst Ports: 1 ]						
2	snort: "ET WEB_SERVER Script tag in URI, Possible Cross Site Scripting Attempt"	0	2015-02-18 21:16:32	172.24.121.134:14075	Vulnerable:http	2
3	snort: "ET WEB_SERVER Script tag in URI, Possible Cross Site Scripting Attempt"	0	2015-02-18 21:14:02	172.24.121.134:13978	Vulnerable:http	1

Figura 5-15: Detalle de la alerta de XSS

A la vista de la figura 5-15, vemos como tras dos posibles intentos de ataque de Cross-Site Scripting, el sistema genera una alerta de ataque real. Asimismo, al igual que en los ataque de Inyección SQL, nos encontramos ante una regla de correlación de 3 niveles con una funcionalidad diferente cada uno de ellos:

- El primer nivel detecta un evento catalogado como un posible intento de Cross-Site Scripting
- El segundo nivel detecta un segundo evento de posible intento de Cross-Site Scripting entre el mismo origen y el mismo destino que fueron detectados en el primer nivel para confirmar el ataque
- El tercer nivel de correlación es un nivel acumulativo que se encarga de recoger todos los demás eventos de este tipo en un tiempo determinado, una hora en este caso, con la finalidad de no generar un número demasiado elevado de alertas cuando este tipo de ataques son generados por herramientas de explotación automáticas



Después de este análisis, podemos concluir que este tipo de ataques de ejecución de comandos en sitios cruzados son detectados **exitosamente** por la plataforma y gracias a la generación de esta alerta de correlación se podría, de manera reactiva, generar algún tipo de acción manual o automática para protegernos de futuros ataques desde dicho origen. No obstante, como hemos podido observar en la explicación de la vulnerabilidad, el ataque ha sido satisfactorio para el atacante y ha conseguido modificar el contenido del documento antes de que hayamos podido hacer nada.

### 5.1.3 Inclusión de ficheros

Este tipo de ataques son posibles cuando se utiliza para la construcción del sitio web una página externa que se define a través de algún parámetro en la petición web. De este modo, si la página es vulnerable a este tipo de ataques, podemos incluir en la página mostrada información sensible del sistema como puede ser, archivos sensibles del sistema, algún tipo de fichero previamente modificado que nos permita obtener datos sensibles e incluso poder modificar completamente el contenido de la página haciendo referencia a un enlace externo a la propia máquina que nos está sirviendo el contenido.

#### 5.1.3.1 Explotación de la vulnerabilidad

Para poder explotar esta vulnerabilidad es necesario que para la construcción de la página, que nos va a servir el servidor web, se haga referencia a algún fichero a través de la petición web. En la aplicación web que nos encontramos, en la sección de File Inclusion, existe una página con estas características.

En la figura siguiente se puede apreciar la página con estas características:

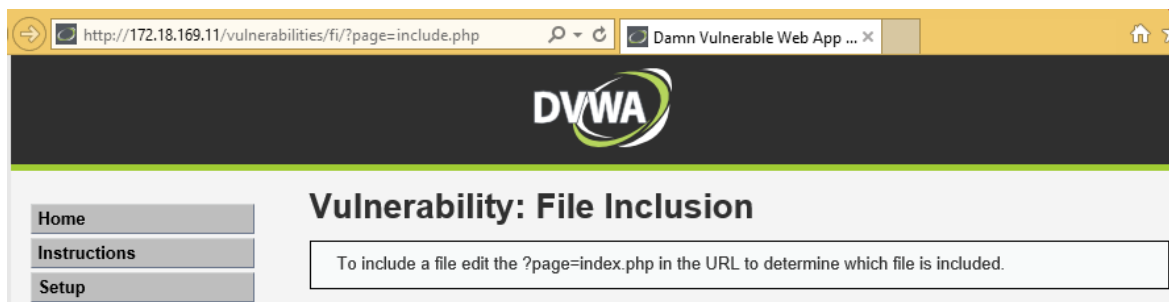


Figura 5-16: Página vulnerable a la inclusión de ficheros

Si se modifica la petición cambiando el fichero *include.php* por el fichero */etc/passwd*, vemos claramente como esta página web es vulnerable ya que nos muestra sin ningún tipo de problema el contenido del fichero de usuarios del sistema.



Figura 5-17: Explotación de la vulnerabilidad de inclusión de ficheros

Que ocurra esto se debe a que no se hace ningún tipo de comprobación sobre la página que se va a incluir como contenido tal y como se puede apreciar en el código de la página.



Figura 5-18: Código de la página vulnerable a la inclusión de ficheros

Este tipo de vulnerabilidad es crítica ya que, si somos capaces de levantar una consola inversa sobre la máquina destino, ya disponemos de los usuarios del sistema y sería tan fácil comprometer el servidor como llevar a cabo un ataque por fuerza bruta sobre los usuarios que ya hemos sido capaces de extraer.

### 5.1.3.2 Análisis de la detección

Para este tipo de vulnerabilidades, disponemos de la información proporcionada tanto por los logs del servidor web como por los paquetes analizados por el sistema de prevención de intrusiones. Con los primeros, podemos detectar intentos de acceso a ficheros sensibles, sean exitosos o no, y con los segundos, al analizar la petición y la respuesta del servidor, es posible detectar si el ataque ha tenido éxito o no.

Veamos que ocurre con los logs del servidor web cuando llevamos a cabo una petición que contenga el archivo */etc/passwd*.

Feb 19 12:27:42 vulnerable apache: 172.24.121.134 - - [19/Feb/2015:12:27:42 +0100] "GET /vulnerabilities/fi/?page=/etc/passwd HTTP/1.1" 200 4648 "-" "Mozilla/5.0 (Windows NT 6.3; WOW64; Trident/7.0; rv:11.0) like Gecko"

En la petición web vemos claramente el intento la cadena */etc/passwd*, con lo que a través de una expresión regular, podemos detectar este tipo de inclusiones de una manera muy sencilla. Así si encontramos varias peticiones web con peticiones a archivos críticos del sistema es muy fácil configurar una regla de correlación que nos advierta de un posible intento de acceso a información sensible del sistema.

El esquema de la regla de correlación sería:

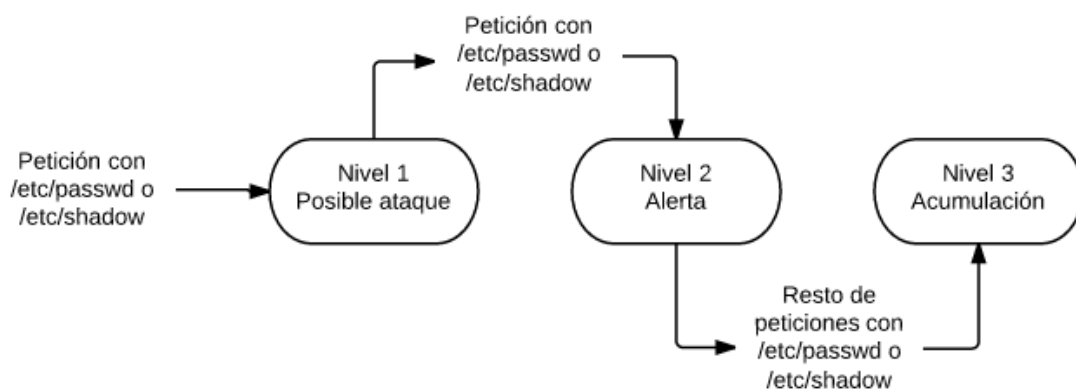


Figura 5-19: Esquema regla de correlación de inclusión de ficheros 1

Esta alerta generada requeriría una análisis posterior para comprobar la fiabilidad de la misma y si realmente el ataque ha tenido éxito o no. En el caso de que el intento de ataque sea real, haya tenido éxito o no, se podría llevar a cabo de manera reactiva un bloqueo de las conexiones provenientes del origen para protegernos frente a posibles nuevos ataques.

Por otro lado, gracias al análisis de los paquetes llevado a cabo por el sistema de prevención de intrusiones podemos llevar a cabo la correlación de la petición al servidor web y la respuesta del mismo para generar una alerta de correlación adicional en el caso de que el ataque haya tenido éxito. El esquema de esta regla de correlación es el siguiente:

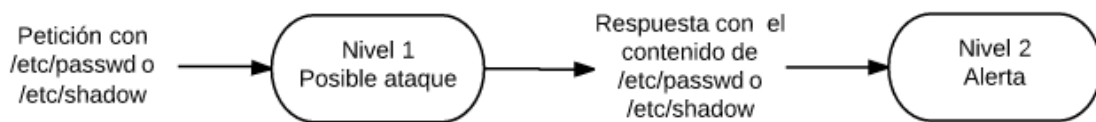


Figura 5-20: Esquema regla de correlación de inclusión de ficheros 2

En esta caso nos encontramos ante un ataque real y que además ha tenido éxito con lo que podríamos activar un bloqueo del atacante automáticamente en el Firewall perimetral para evitar futuros ataques desde este origen.

Como hemos visto en esta sección este tipo de ataques de inclusión fácilmente detectables con la información de la que disponemos en el sistema de correlación, ahora veamos si el sistema de correlación correctamente configurado es capaz de detectarlos automáticamente. Para ello, veamos primero que eventos se han generado en el correlador cuando se ha generado una petición de este tipo.

Primero analizamos los eventos generados por el servidor web gracias al paquete de firmas PHPIDS:

SIGNATURE	DATE GMT+1:00	SENSOR	SOURCE	DESTINATION	RISK
RTLA: Detects etc/passwd inclusion attempts	2015-02-19 12:51:24	av-allinone	172.24.121.134	Vulnerable	0
RTLA: Detects etc/passwd inclusion attempts	2015-02-19 12:27:42	av-allinone	172.24.121.134	Vulnerable	0
RTLA: Detects etc/passwd inclusion attempts	2015-02-19 11:59:06	av-allinone	172.24.121.134	Vulnerable	0

Figura 5-21: Eventos detectados de inclusión de ficheros 1

Provocado por estos tres eventos desde el mismo origen al mismo destino el sistema de correlación ha generado la siguiente alerta relacionada.

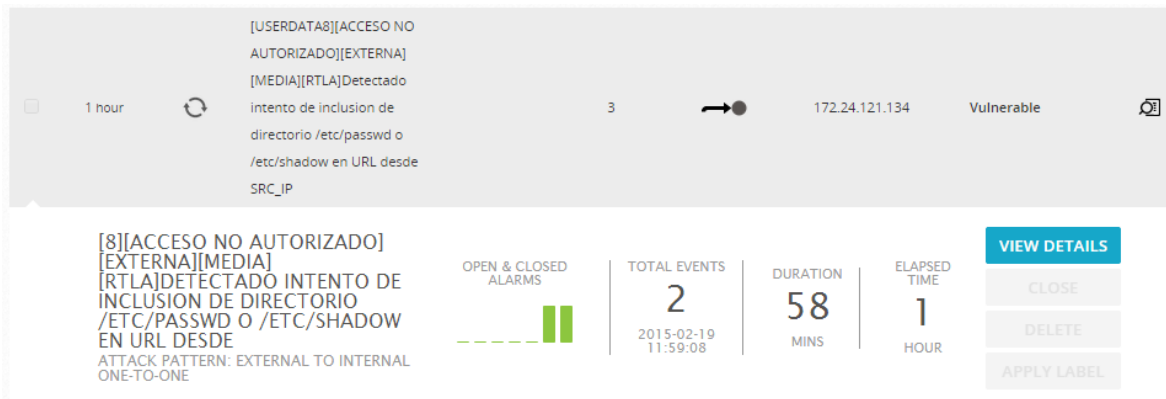


Figura 5-22: Alerta generada de inclusión de ficheros 1

Se puede afirmar que se detectan **exitosamente** los intentos recursivos de intentos de inclusión de ficheros /etc/passwd.

Ahora se procede al análisis de los eventos del sistema de prevención de intrusiones, cuyos eventos generados con una única petición son los siguientes.

SIGNATURE	DATE GMT+1:00	SENSOR	SOURCE	DESTINATION	RISK
snort: "GPL WEB_SERVER /etc/passwd"	2015-02-19 12:58:50	av-ips	172.24.121.134:63312	Vulnerable:80	0
snort: "GPL WEB_SERVER /etc/passwd"	2015-02-19 12:58:50	av-ips	172.24.121.134:63312	Vulnerable:80	0
snort: "GPL WEB_SERVER /etc/passwd"	2015-02-19 12:58:45	av-ips	172.24.121.134:63312	Vulnerable:80	0
snort: "GPL WEB_SERVER /etc/passwd"	2015-02-19 12:58:45	av-ips	172.24.121.134:63312	Vulnerable:80	0
snort: "GPL WEB_SERVER /etc/passwd"	2015-02-19 12:58:45	av-ips	172.24.121.134:63312	Vulnerable:80	0
snort: "GPL WEB_SERVER /etc/passwd"	2015-02-19 12:58:45	av-ips	172.24.121.134:63312	Vulnerable:80	0
URL snort: "ET ATTACK RESPONSE Possible /etc/passwd via HTTP (linux style)"	2015-02-19 12:58:45	av-ips	Vulnerable:80	172.24.121.134:63312	0
snort: "GPL WEB_SERVER /etc/passwd"	2015-02-19 12:58:45	av-ips	172.24.121.134:63312	Vulnerable:80	0

Figura 5-23: Eventos detectados de inclusión de ficheros 2

Se puede observar que se han categorizados paquetes como peticiones al servidor web que contienen el contenido /etc/passwd y además se ha detectado una respuesta del servidor con el fichero /etc/passwd.

Ahora sólo queda comprobar si estos eventos han generado alerta.

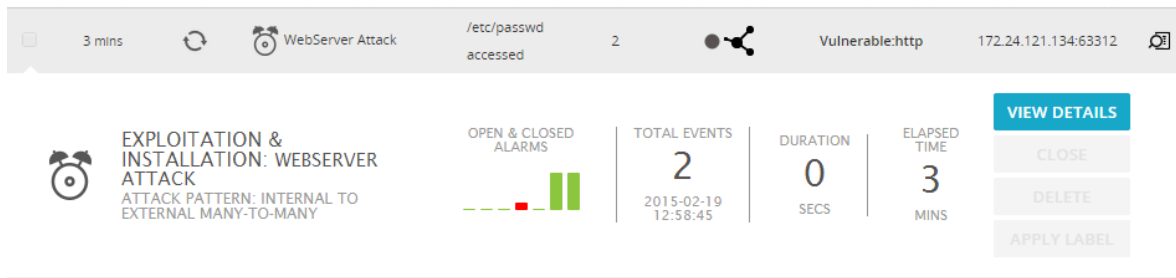


Figura 5-24: Alerta generada de inclusión de ficheros 2

Se observa claramente como la alerta ha sido generada y la detección de este tipo de ataque ha sido un **éxito**.

Si analizamos los eventos implicados en la generación de la alerta, vemos que se tratan de la petición y la respuesta del servidor.

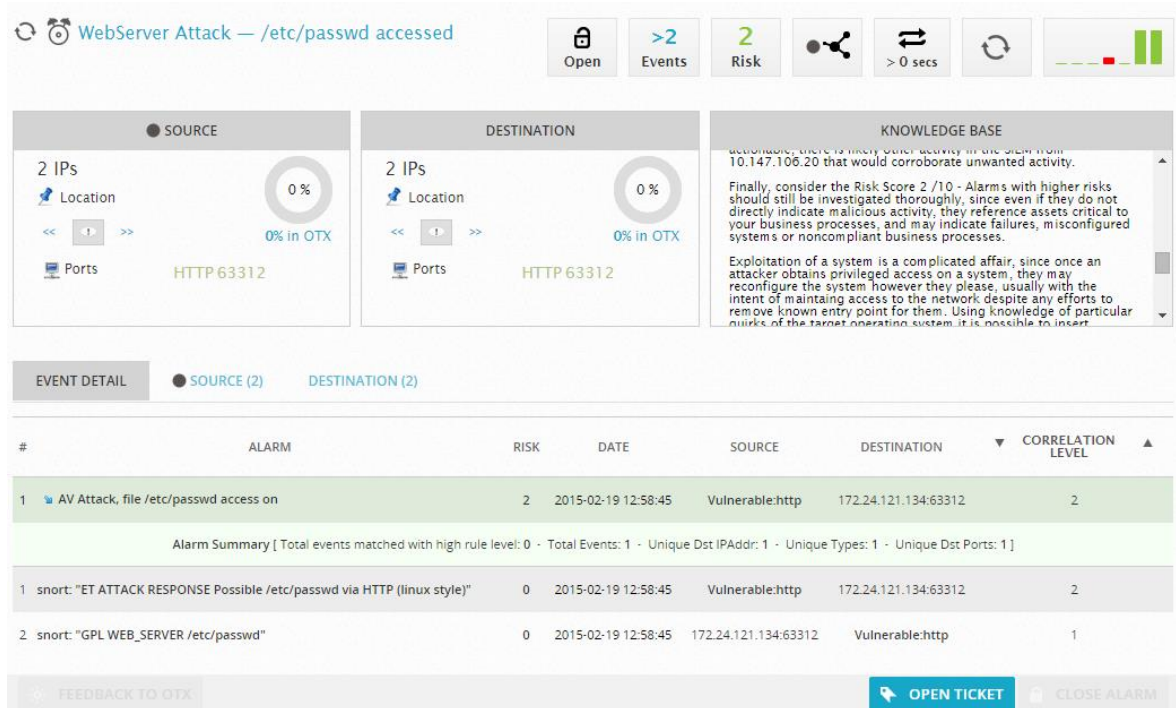


Figura 5-25: Detalle de la alerta de inclusión de ficheros 2

Si bien este ataque ha sido detectado por ambas fuentes y se han generado las alertas correspondientes, no se ha impedido que se extrajese la información del sistema, únicamente podríamos realizar una acción reactiva para prevenir futuros ataques.



### 5.1.4 Inyección de comandos

Este tipo de ataque se aprovecha de una vulnerabilidad de los sitios web que mandan comandos al propio sistema operativo del servidor con el fin de ejecutar alguna acción, requerida por las funcionalidades de la página, pero no hacen una correcta comprobación de los parámetros de entrada que proporciona el usuario. Esto supone un gran riesgo para el equipo ya que permite al usuario realizar diferentes operaciones sobre el propio sistema operativo de la máquina destino.

#### 5.1.4.1 Explotación de la vulnerabilidad

En la aplicación web que tenemos integrada en nuestra maqueta de pruebas, una de las secciones es vulnerable a este tipo de ataque. En esta ocasión la página nos pide que introduzcamos una IP para realizar una prueba de conectividad mediante el comando **ping** desde el equipo hasta la máquina introducida como vemos en la siguiente figura:



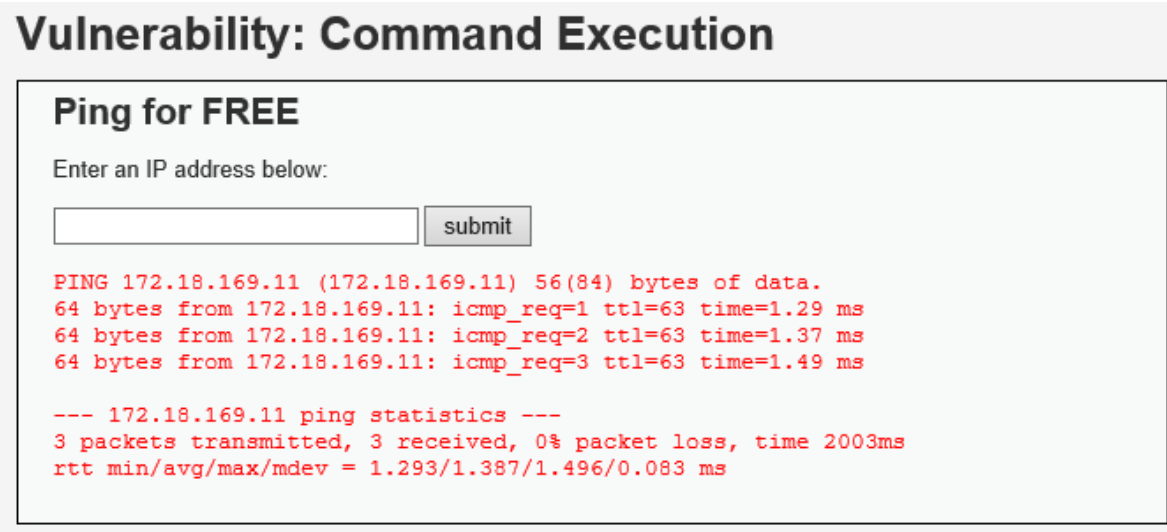
**Vulnerability: Command Execution**

**Ping for FREE**

Enter an IP address below:

Figura 5-26: Formulario de inyección de comandos

Comprobamos su correcto funcionamiento ejecutando un ping a la máquina 172.18.169.11 que es la IP pública del Firewall perimetral.



**Vulnerability: Command Execution**

**Ping for FREE**

Enter an IP address below:

```
PING 172.18.169.11 (172.18.169.11) 56(84) bytes of data.  
64 bytes from 172.18.169.11: icmp_req=1 ttl=63 time=1.29 ms  
64 bytes from 172.18.169.11: icmp_req=2 ttl=63 time=1.37 ms  
64 bytes from 172.18.169.11: icmp_req=3 ttl=63 time=1.49 ms  
  
--- 172.18.169.11 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2003ms  
rtt min/avg/max/mdev = 1.293/1.387/1.496/0.083 ms
```

Figura 5-27: Uso del formulario de inyección de comandos

Como podemos comprobar la página nos devuelve la ejecución del comando **ping**, el cual en este caso ha sido exitoso y tenemos conectividad con la máquina consultada.

Pasemos ahora a analizar el código del lado del servidor de esta página y veamos por qué es vulnerable:

### Command Execution Source

```
<?php
if( isset( $_POST[ 'submit' ] ) ) {
    $target = $_REQUEST[ 'ip' ];

    // Determine OS and execute the ping command.
    if (strstr(php_uname('s'), 'Windows NT')) {

        $cmd = shell_exec( 'ping ' . $target );
        echo '<pre>'.$cmd.'</pre>';

    } else {

        $cmd = shell_exec( 'ping -c 3 ' . $target );
        echo '<pre>'.$cmd.'</pre>';

    }

}
?>
```

Figura 5-28: Código página de inyección de comandos

Como se ve en el fragmento de código anterior esta página es vulnerable a la inyección de comandos ya que no se hace ningún tipo de validación de los datos introducidos por el usuario en el campo IP. De esta manera, nada nos impide ejecutar comandos después de que el equipo lleve a cabo la acción del **ping**. Debido a esto, es relativamente sencillo leer el archivo `/etc/passwd` que contiene los usuarios del sistema introduciendo en el campo IP la siguiente sentencia: **172.18.169.11; cat /etc/passwd**.

A continuación se muestra el resultado de hacer esta inyección de código:

## Vulnerability: Command Execution

### Ping for FREE

Enter an IP address below:

```
PING 172.18.169.11 (172.18.169.11) 56(84) bytes of data.  
64 bytes from 172.18.169.11: icmp_req=1 ttl=63 time=1.21 ms  
64 bytes from 172.18.169.11: icmp_req=2 ttl=63 time=1.28 ms  
64 bytes from 172.18.169.11: icmp_req=3 ttl=63 time=1.05 ms  
  
--- 172.18.169.11 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2002ms  
rtt min/avg/max/mdev = 1.057/1.185/1.285/0.095 ms  
root:x:0:0:root:/root:/bin/bash  
daemon:x:1:1:daemon:/usr/sbin:/bin/sh  
bin:x:2:2:bin:/bin:/bin/sh  
sys:x:3:3:sys:/dev:/bin/sh  
sync:x:4:65534:sync:/bin:/bin/sync  
games:x:5:60:games:/usr/games:/bin/sh  
man:x:6:12:man:/var/cache/man:/bin/sh  
lp:x:7:7:lp:/var/spool/lpd:/bin/sh  
mail:x:8:8:mail:/var/mail:/bin/sh  
news:x:9:9:news:/var/spool/news:/bin/sh  
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh  
proxy:x:13:13:proxy:/bin:/bin/sh  
www-data:x:33:33:www-data:/var/www:/bin/sh  
backup:x:34:34:backup:/var/backups:/bin/sh  
list:x:38:38:Mailing List Manager:/var/list:/bin/sh  
irc:x:39:39:ircd:/var/run/ircd:/bin/sh  
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh  
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh  
libuuid:x:100:101::/var/lib/libuuid:/bin/sh  
Debian-exim:x:101:103::/var/spool/exim4:/bin/false  
statd:x:102:65534::/var/lib/nfs:/bin/false  
ahuerta:x:1000:1000:Alejandro Huerta Molina,,,:/home/ahuerta:/bin/bash  
sshd:x:103:65534::/var/run/sshd:/usr/sbin/nologin  
mysql:x:999:999::/home/mysql:/bin/sh
```

Figura 5-29: Explotación de la vulnerabilidad de inyección de comandos

Como podemos observar claramente en la figura anterior, vemos que tras la ejecución del comando *ping*, se ha llevado a cabo la ejecución del comando *cat* a uno de los archivos críticos del sistema sin ningún problema.

### 5.1.4.2 Análisis de la detección

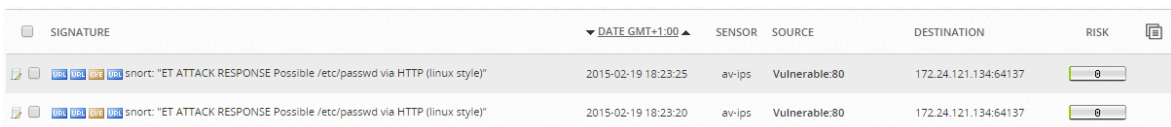
Vamos a analizar el nivel de detección que tenemos ante este tipo de ataque web con las herramientas que están a nuestra disposición.

Debido a que esta petición se hace a través del método POST, no podemos obtener información de los logs del servidor web, pero si podemos analizar los paquetes que pasan a través del sistema de prevención de intrusiones. La petición que registra el servidor web es la siguiente:

```
Feb 19 18:23:17 vulnerable apache: 172.24.121.134 - - [19/Feb/2015:18:23:15 +0100] "POST /vulnerabilities/exec/HTTP/1.1" 200 5881 "http://172.18.169.11/vulnerabilities/exec/" "Mozilla/5.0 (Windows NT 6.3; WOW64; Trident/7.0; rv:11.0) like Gecko"
```

Como podemos observar en la petición anterior, no tenemos la posibilidad de encontrar patrones sospechosos en ella ya que los parámetros en el método post se mandan como parte de la cabecera y no aparecen en la URI.

Por otro lado, en el sistema de prevención de intrusiones sí que aparecen eventos relacionados con estas peticiones al archivo */etc/passwd* detectando la repuesta del servidor web como podemos ver en la siguiente figura.



SIGNATURE	DATE GMT+1:00	SENSOR	SOURCE	DESTINATION	RISK
snort: "ET ATTACK RESPONSE Possible /etc/passwd via HTTP (linux style)"	2015-02-19 18:23:25	av-ips	Vulnerable:80	172.24.121.134:64137	0
snort: "ET ATTACK RESPONSE Possible /etc/passwd via HTTP (linux style)"	2015-02-19 18:23:20	av-ips	Vulnerable:80	172.24.121.134:64137	0

Figura 5-30: Evento detectados de inyección de comandos

No obstante, como no tenemos visibilidad de la petición en ninguno de los caso las no se ha generado alerta de correlación relacionada con estas peticiones y el sistema no nos habría avisado de este tipo de ataque.

Según lo expuesto, el sistema, con las configuraciones actuales, carece de visibilidad sobre este tipo de ataques. Sería necesario la generación de alguna firma o regla de correlación nueva para la detección de este tipo de comportamientos. No obstante, también existen otros tipos de dispositivos que se encargan del análisis profundo del protocolo HTTP, como son los firewall de aplicación web (Web Application Firewall), que serían de gran ayuda en la detección de este tipo ataque de acceso no autorizado.

## **5.2 Análisis de seguridad con bloqueo de conexiones**

Tras el análisis de las vulnerabilidades y las capacidades de detección que se han expuesto en el apartado 6.1 pasamos al desarrollo de las acciones mitigantes que podemos ejecutar para prevenir estas fugas de información en una plataforma vulnerable como la que nos encontramos.

Principalmente tenemos dos opciones para preservar la información o evitar intentos de intrusión en los equipos protegidos. Por un lado tenemos la respuesta reactiva a la que se ha hecho mención en el apartado 6.1 derivada de una alerta de correlación y por otro, gracias a que disponemos de un sistema de prevención de intrusiones, podemos configurarlo en modo en línea con la capacidad de cortar conexiones cuando detecta eventos que están catalogados como maliciosos.

En este apartado, nos centramos en los cambios realizados al sistema de prevención de intrusiones y a la activación de bloqueo por firmas derivado del análisis de los ataques que se han visto en el apartado 6.1.

### **5.2.1 Cambios en el sistema de prevención de intrusiones**

Para llevar a cabo los bloqueos en las conexiones maliciosas detectadas en el sistema de prevención de intrusiones, es necesario que las firmas que nosotros queramos pasen a estar en bloqueo. Esto se consigue cambiando el primer parámetro de las reglas de *alert* a *drop*. Así, de esta manera, el sistema estará protegido frente a los ataques de una manera completamente activa pudiendo cortar las comunicaciones antes de que se produzca la fuga de información.

Para ello tras el análisis de las firmas detectadas en los diferentes ataques del apartado 6.1 procedemos a poner en bloqueo las siguientes firmas:

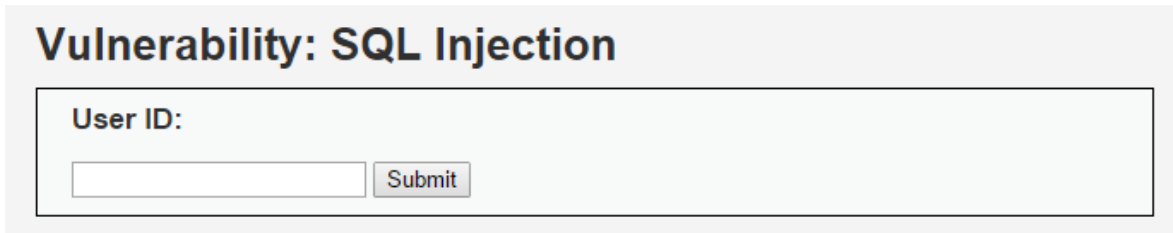
- snort: "ET WEB\_SERVER Script tag in URI, Possible Cross Site Scripting Attempt"
- snort: "ET ATTACK RESPONSE Possible /etc/passwd via HTTP (linux style)"
- snort: "ET WEB\_SERVER Possible SQL Injection Attempt UNION SELECT"

En los siguientes apartados, se procede a repetir las pruebas de la explotación de las vulnerabilidades para ver las diferencias de comportamiento de la plataforma gracias a la activación de los bloqueos en las firmas mencionadas anteriormente.

## 5.2.2 Inyección SQL

Repetimos las pruebas de explotación de la vulnerabilidad de inyección SQL atacando la página con los mismos parámetros que se utilizaron en el apartado anterior, es decir, comprobamos que sigue funcionando la página con los parámetros no maliciosos y vemos el comportamiento con la inyección SQL.

Nos encontramos de nuevo en la siguiente página:

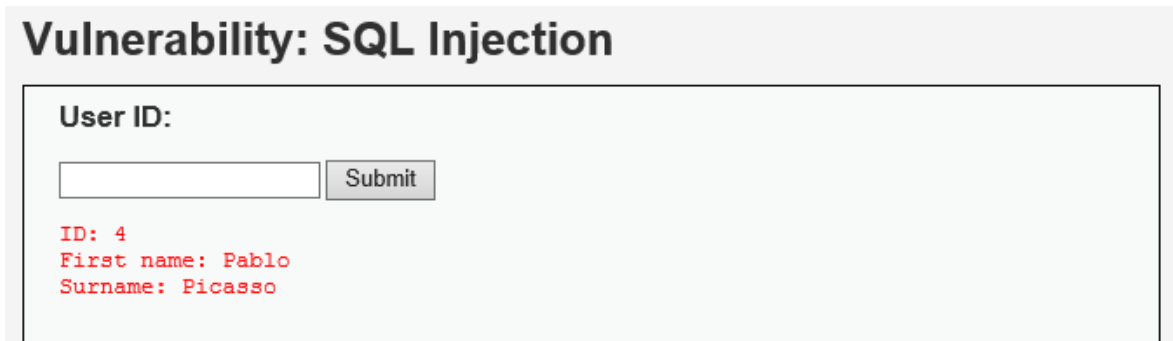


Vulnerability: SQL Injection

User ID:

Figura 5-31: Formulario SQL injection

Si repetimos la consulta a un usuario vemos como el sistema sigue funcionando correctamente:



Vulnerability: SQL Injection

User ID:

ID: 4  
First name: Pablo  
Surname: Picasso

Figura 5-32: Uso Formulario SQL injection

Comprobamos por tanto que los cambios ejecutados en el sistema de prevención de intrusiones no afectan al correcto funcionamiento de la página.

Por otro lado vamos a ver cuales son las implicaciones del cambio en las políticas del sistema de prevención de intrusiones cuando intentamos hacer una inyección SQL para extraer los usuarios y las contraseñas de la base de datos con la siguiente sentencia: *' union select first\_name, password from users #*.

## Vulnerability: SQL Injection

User ID:

Figura 5-33: Intento de explotación SQL injection

Después de pulsar en el botón Submit para llevar a cabo la petición al servidor web con la inyección SQL esta es la respuesta del mismo.

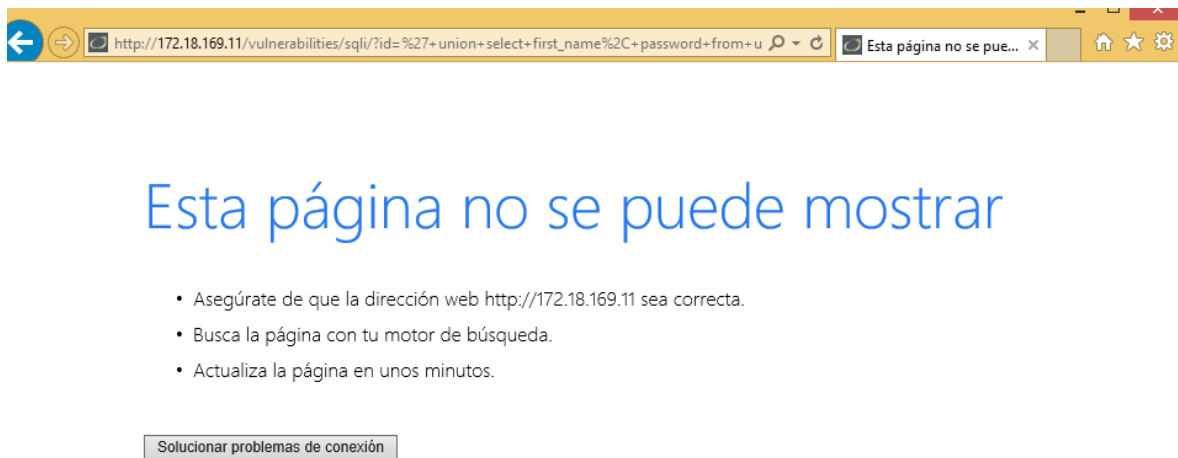


Figura 5-34: Mensaje de error después de SQL injection

Como podemos observar el sistema nos muestra un error a la hora de mostrar la página y esto se debe a que el sistema de prevención de intrusiones ha bloqueado esta conexión maliciosa.

Por otro lado, el correlador de eventos también ha registrado los eventos, de la misma manera que ocurría cuando teníamos el sistema en solo detección, y ha generado una alerta de correlación relacionada a estos eventos que nos avisa del ataque de inyección SQL.

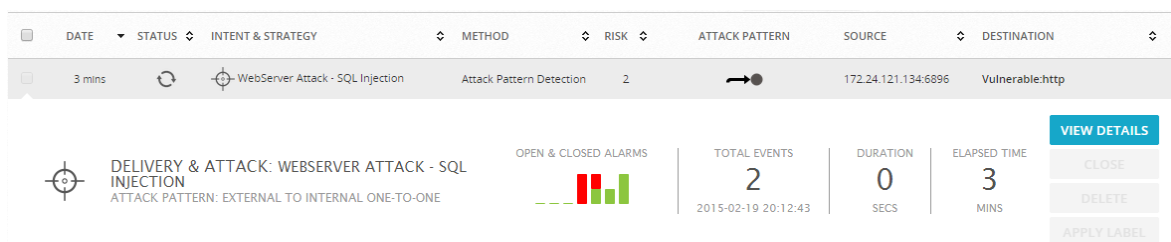


Figura 5-35: Alerta de SQL injection

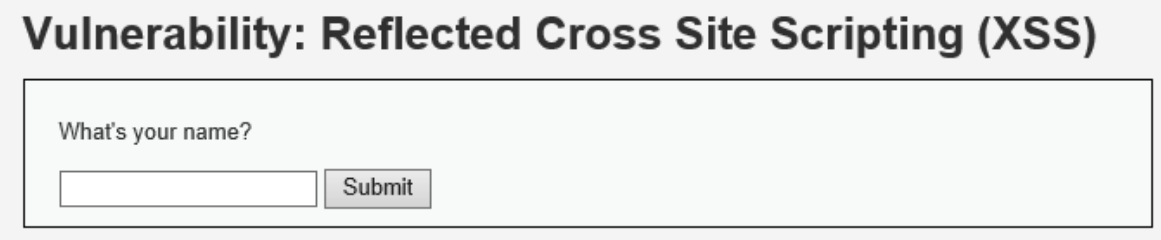
Gracias a esta alerta se podría, además de la respuesta activa proporcionada por el sistema de prevención de intrusiones, generar de manera reactiva un bloqueo para protegernos de futuros ataques de la IP de origen.

Tras este análisis, vemos que el ataque de inyección SQL no ha sido satisfactorio para el atacante y éste no ha podido extraer la información que solicitaba. Así podemos afirmar que nos encontramos protegidos ante este tipo de ataques de acceso no autorizados.

### 5.2.3 Secuencias de comandos en sitios cruzados

Repetimos las pruebas de explotación de la vulnerabilidad de Cross-Site Scripting atacando la página con los mismos parámetros que se utilizaron en el apartado anterior, es decir, comprobamos que sigue funcionando la página con los parámetros no maliciosos y vemos el comportamiento con la introducción en el parámetro la secuencia de comandos a ejecutar.

Nos encontramos de nuevo en la siguiente página:

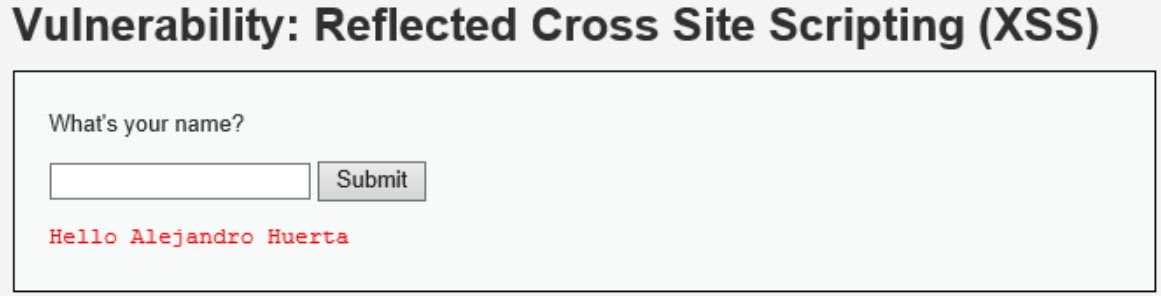


**Vulnerability: Reflected Cross Site Scripting (XSS)**

What's your name?

**Figura 5-36: Formulario XSS**

Ejecutamos la página con la entrada esperada por parte de la página web sin ningún tipo de código malicioso.



**Vulnerability: Reflected Cross Site Scripting (XSS)**

What's your name?

  
**Hello Alejandro Huerta**

**Figura 5-37: Uso Formulario XSS**



Como podemos comprobar la página sigue comportándose como se espera cuando los valores introducidos en el campo “name” no contienen ningún intento de explotación de la vulnerabilidad.

Ahora comprobemos el funcionamiento de la página si introducimos la siguiente entrada en el parámetro: `<script>alert('Prueba con bloqueo')</script>`. Esta prueba nos permite ver si esta página es vulnerable a XSS.

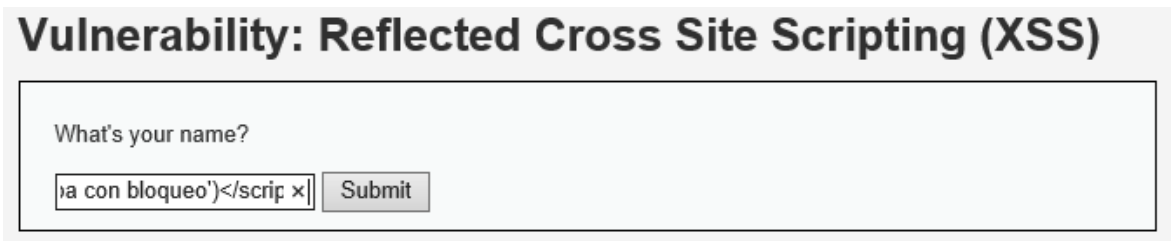


Figura 5-38: Intento de explotación XSS

Tras pulsar en el botón Submit, el servidor nos devuelve un mensaje de error como el siguiente.

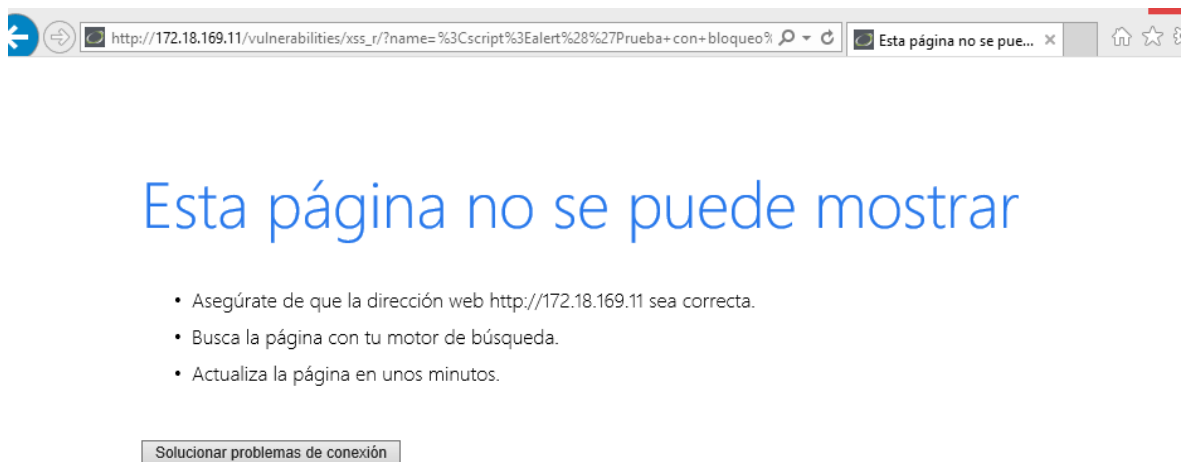


Figura 5-39: Mensaje de error tras intento de XSS

De esta manera, podemos comprobar como el sistema de prevención de intrusiones ha bloqueado el intento de explotación de la vulnerabilidad de XSS presente en la página y nos mantiene protegidos frente a los ataques de este tipo. Además, tal y como vemos a continuación, se ha generado una alerta de correlación en el correlador que nos permitiría aplicar una acción de bloqueo automático del origen para prevenir ataques desde el mismo origen.

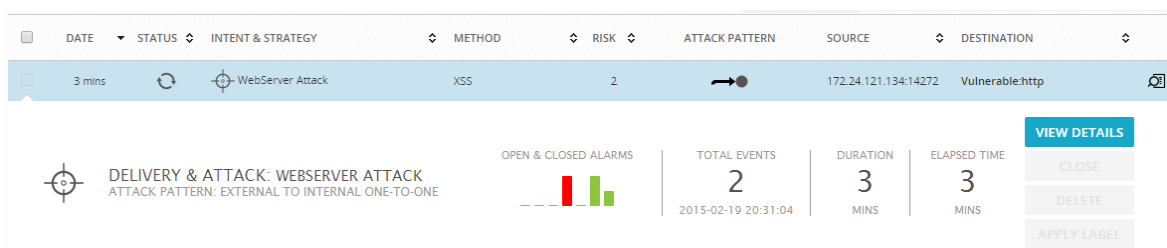


Figura 5-40: Alerta de XSS

Como se ha podido comprobar, gracias al bloqueo de estas conexiones, el sistema vulnerable se encuentra protegido frente a los ataques de XSS de este tipo, lo que supone una gran mejora a nivel de seguridad ya que estamos parando el ataque de forma activa y no tenemos que pararlo de una manera reactiva.

### 5.2.4 Inclusión de ficheros

Repetimos las pruebas de la inclusión del fichero `/etc/passwd` de la misma manera que en el apartado 6.1.3 para comprobar la diferencia de comportamiento con las firmas del sistema de prevención de intrusiones activadas en modo bloqueo de conexiones.

Cuando accedemos a la página y llevamos a cabo la inclusión de la página permitida, es decir, el funcionamiento normal de la misma, el sistema sigue funcionando correctamente tal y como podemos observar en la siguiente figura.

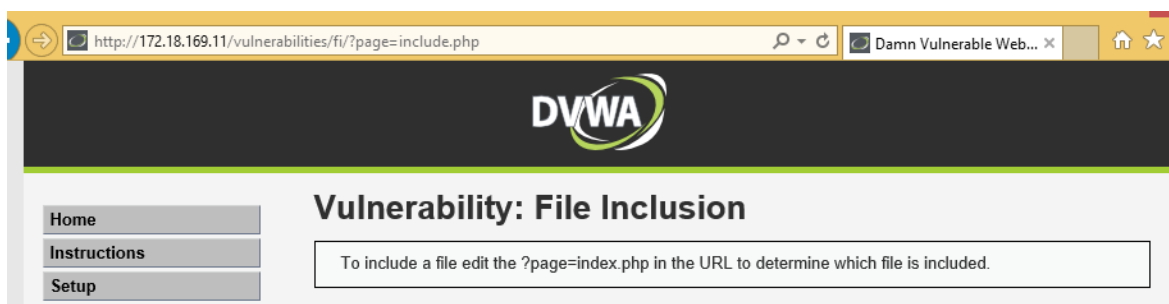


Figura 5-41: Página vulnerable a inclusión de ficheros

Ahora pasamos de nuevo a editar el fichero que vamos a incluir para la generación de la página por el fichero `/etc/passwd` y veamos cual es el resultado.

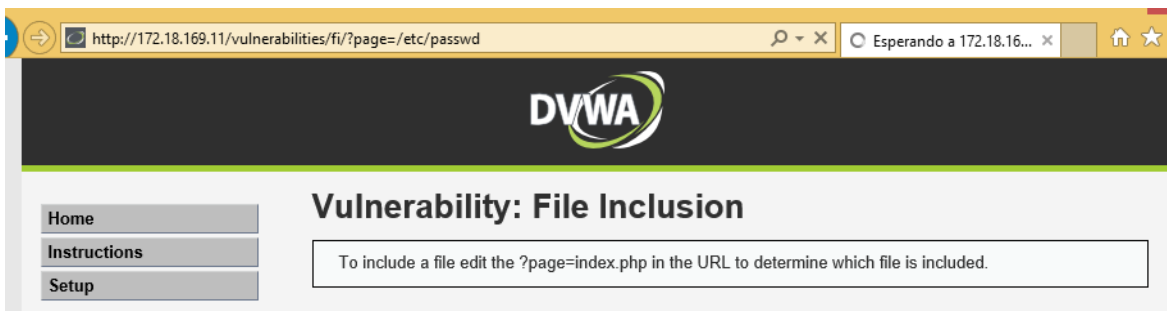


Figura 5-42: Intento de explotación de inclusión de ficheros

En este caso, la página no nos muestra un mensaje de error ya que la conexión que se ha bloqueado es la repuesta del servidor pero podemos observar en la figura como se mantiene esperando la respuesta, una respuesta que nunca llega ya que ha sido bloqueada por el sistema de prevención de intrusiones.

No obstante, los eventos han sido generados tanto en el servidor web como en el sistema de prevención de intrusiones y como se observa en la figura siguiente, se han generado las alertas de correlación correspondientes como se puede ver en la siguiente figura.

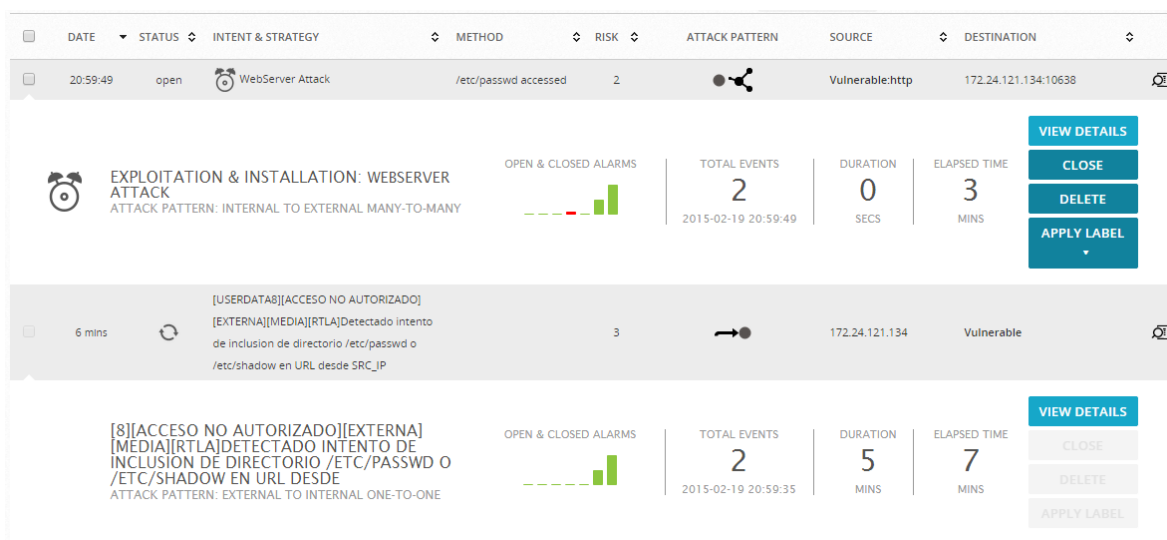


Figura 5-43: Alertas generada de inclusión de ficheros

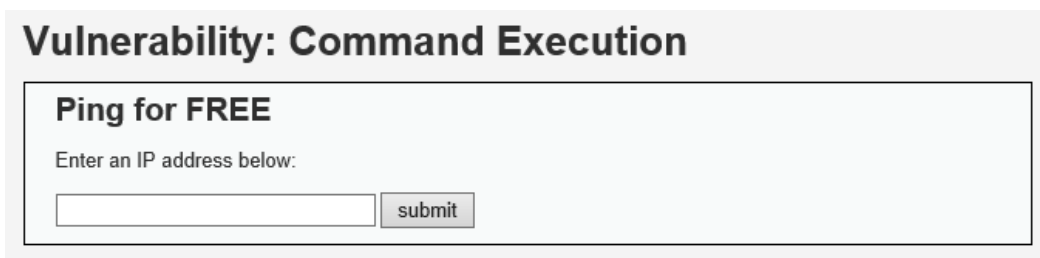
Esto supone un gran avance en la seguridad de la página ya que aunque los intentos de explotación están siempre presentes, de esta manera podemos proteger la página de esta vulnerabilidad y evitar la fuga de información de una manera activa.

Asímismo, como estas alertas se han generado de manera satisfactoria, se podría configurar el correlador para que se llevase a cabo la acción reactiva de bloquear al atacante en el firewall perimetral para prevenir nuevos ataques desde el mismo origen.

### 5.2.5 Inyección de comandos

Pasamos al análisis de seguridad con los cambios establecidos en el sistema de prevención de intrusiones para este tipo de ataques, que como vimos en la sección 6.1.4.2 pasaban desapercibidos a la detección del correlador.

Volvemos de nuevo a la página que es vulnerable a este tipo de ataques.



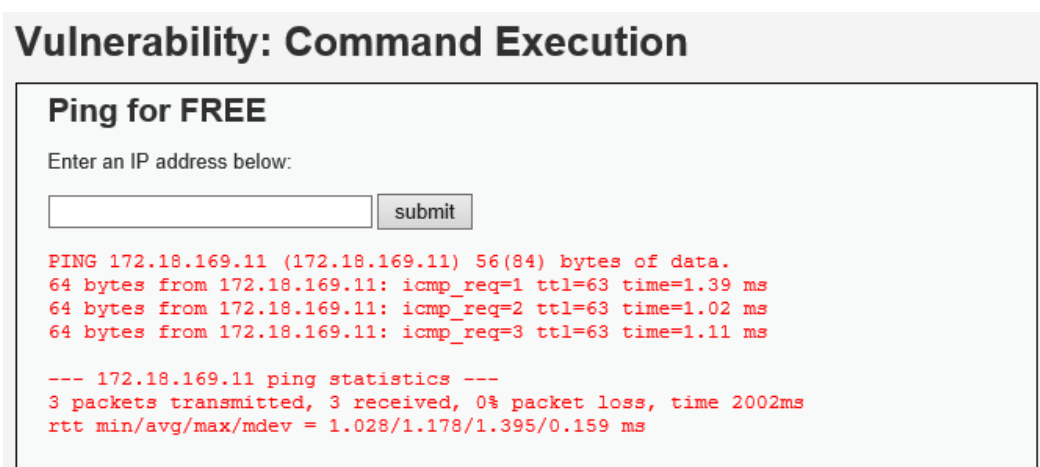
**Vulnerability: Command Execution**

**Ping for FREE**

Enter an IP address below:

**Figura 5-44: Formulario de inyeccion de comandos**

Primero comprobamos el correcto funcionamiento de la página en las condiciones habituales para comprobar que no cortamos las comunicaciones legítimas.



**Vulnerability: Command Execution**

**Ping for FREE**

Enter an IP address below:

```
PING 172.18.169.11 (172.18.169.11) 56(84) bytes of data.  
64 bytes from 172.18.169.11: icmp_req=1 ttl=63 time=1.39 ms  
64 bytes from 172.18.169.11: icmp_req=2 ttl=63 time=1.02 ms  
64 bytes from 172.18.169.11: icmp_req=3 ttl=63 time=1.11 ms  
  
--- 172.18.169.11 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2002ms  
rtt min/avg/max/mdev = 1.028/1.178/1.395/0.159 ms
```

**Figura 5-45: Uso formulario de inyeccion de comandos**

Como podemos observar, la página sigue comportándose de la forma correcta cuando le pasamos una IP a la cual lanzar un ping.

Ahora procedemos, al igual que en la sección 6.1.4.1 a modificar la cadena de entrada para, además de hacer un ping a la IP proporcionada, proceda a leer la información del fichero /etc/passwd con el siguiente formato: **172.18.169.11; cat /etc/passwd**.

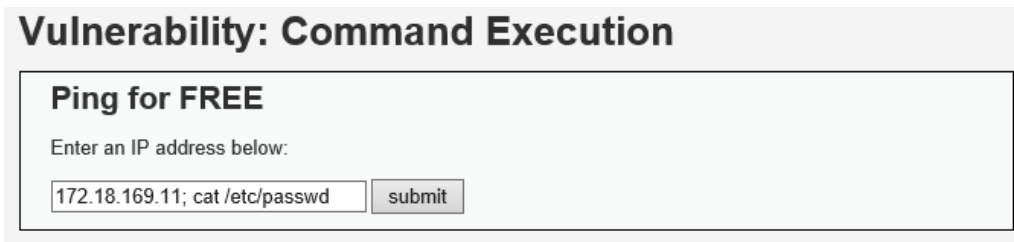


Figura 5-46: Intento de explotación de inyección de comandos

Después de hacer click en el botón submit, la página nos devuelve la siguiente salida.

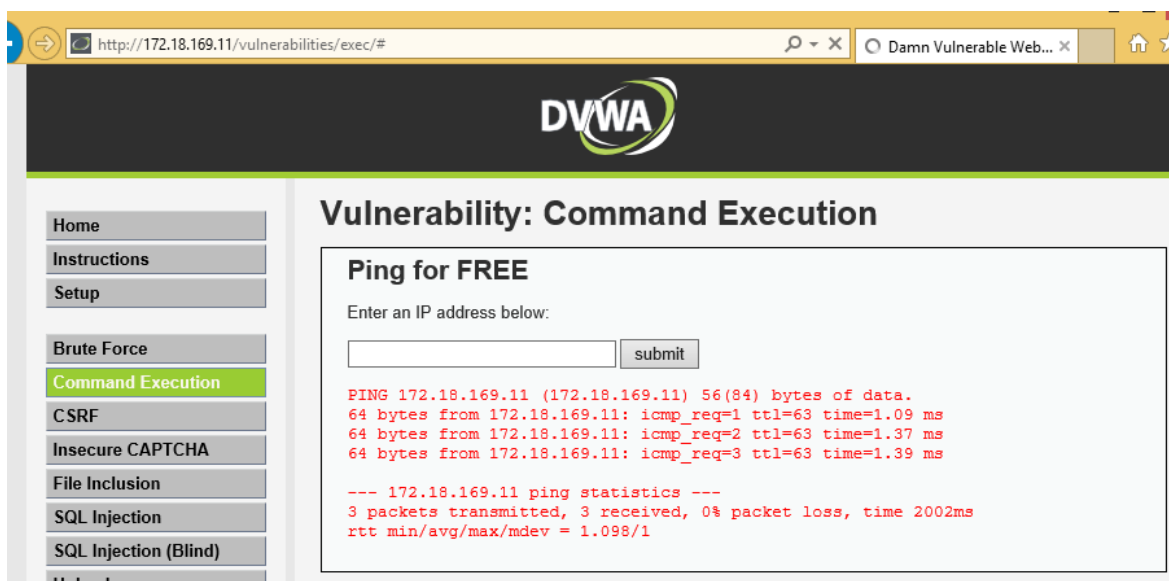


Figura 5-47: Página después de intento de inyección de comandos

Analizando la figura, vemos que la página nos devuelve la información del comando ping mientras que la información contenida en la segunda parte del comando, acceso al fichero /etc/passwd, nunca llega al cliente ya que debido a la misma casuística que en el apartado 6.2.4 la

respuesta del servidor con la información contenida en el fichero `/etc/passwd` es bloqueada por el sistema de prevención de intrusiones.

Este caso es el que mejor representa la mejoría con respecto al modo de detección ya que, aunque este tipo de ataque no lo detectamos de ninguna de las maneras, al menos cortamos las comunicaciones entre el origen y el destino para los paquetes que puedan suponer un riesgo para la seguridad de la plataforma.

### 5.3 Comparativa de los resultados

Tras los resultados obtenidos en las pruebas llevadas a cabo en los apartados 6.1 y 6.2 a continuación se muestra una comparativa entre las dos configuraciones propuestas para la detección y mitigación de los diferentes tipos de ataque.

Ataques	Modo no bloqueo		Modo bloqueo	
	Detección	Mitigación	Detección	Mitigación
Inyección SQL	SI	Reactiva	SI	Tiempo real y reactiva
Cross-Site Scripting	SI	Reactiva	SI	Tiempo real y reactiva
Inclusión de ficheros	SI	Reactiva	SI	Tiempo real y reactiva
Inyección de comandos	NO	Ninguna	NO	Tiempo real

**Tabla 5-1: Comparativa de resultados**

Como se puede apreciar en la tabla anterior, nos encontramos ante una mejora sustancial cuando activamos el modo bloqueo en el sistema de prevención de intrusiones. Aunque la detección de los ataques se encuentra en los mismos umbrales, la mitigación de los mismos mejora de una manera muy sensible protegiendo la página web vulnerable. Estos bloqueos de conexiones en tiempo real, provoca que, aunque el portal web sea vulnerable a ciertos ataques, la información no sea servida al atacante y por lo tanto el ataque no tenga éxito.

Así mismo en ambos casos cuando la detección de los ataques es exitosa, se pueden llevar a cabo de manera reactiva (a posteriori) ciertas acciones en los dispositivos de red repartidos por la plataforma con la función de aumentar la seguridad de la misma.

Con los dispositivos que tenemos desplegados en el laboratorio, la única opción disponible a la hora de mejorar los bloqueos en tiempo real, del sistema de prevención de intrusiones, es el firewall perimetral. Así se podrían ejecutar bloqueos automáticos en el firewall derivados de las alertas detectadas en el correlador, que mejorarían ostensiblemente la seguridad de la red ya que,

una vez identificado el atacante se bloquearían, durante un periodo de tiempo, todas las comunicaciones desde este origen que ya ha sido catalogado como no confiable.

## **5.4 Conclusiones**

Después del desarrollo de este capítulo, quedan evidenciadas las diferencias a nivel de seguridad con las dos configuraciones propuestas en el sistema de prevención de intrusiones. Queda demostrado que, con la correcta configuración de este dispositivo, se puede obtener una protección pasiva y activa suficiente para la defensa de las publicaciones web vulnerables que se encuentran, a día de hoy, expuestas en internet.

También vemos que gracias a la incorporación de un correlador de eventos en la plataforma, tenemos un alto grado de detección de los diferentes tipos de ataque, lo que nos ofrece la posibilidad de ejecutar acciones a posteriori para mejorar la protección, por ejemplo llevando a cabo un bloqueo temporal de las comunicaciones de la IP atacante.



## 6 Conclusiones y trabajo futuro

---

A la vista de todo lo presentado a largo de este proyecto, pasamos a reflejar un resumen con las conclusiones y posibilidades de trabajo futuro.

### 6.1 Conclusiones

Como se ha desarrollado a lo largo de las páginas anteriores, se tiene la posibilidad de crear un sistema dedicado a dar servicio de publicaciones Web de una forma suficientemente segura, mediante el despliegue de una plataforma como la que se describe en este proyecto.

Para ello, se han evaluado diferentes opciones a nivel de dispositivos de red que nos permiten dar este nivel de seguridad requerido a la plataforma como son los firewall, sistemas de prevención de intrusiones y servidores de correlación. Además, a fin de reducir al máximo los costes, se ha decidido la utilización de software *open source* en todas las capas de seguridad, así como el montaje de la plataforma en un entorno virtualizado para, así, necesitar un único servidor con un sistema de gestión de máquinas virtuales gratuito como se ESXi.

Las diferentes opciones de software *open source* utilizadas para el montaje del laboratorio son:

- **Firewall:** Endian Firewall
- **IPS:** Suricata
- **Correlador:** Alienvault OSSIM
- **Máquina vulnerable:** Debian 7 con LAMP y la aplicación web vulnerable Damn Vulnerable Web Application

Viendo las diferencias en los niveles de seguridad, que se han probado con la batería de pruebas propuesta, se puede ver concluir que las aplicaciones que estén publicadas en la plataforma, aunque sean vulnerables a ataques de diferentes naturalezas, se pueden proteger mediante la adición de cierta clase de dispositivos que nos ayudan a detectar y mitigar, de una manera relativamente sencilla y sin un alto coste, los intentos de explotación sufridos.

En resumen, se puede afirmar que mediante la inclusión en la red de un sistema de prevención de intrusiones y un servicio de correlación de la información, es posible llevar a cabo la mitigación de ataques tanto pasiva como activamente. La respuesta pasiva del sistema son los bloqueos ejecutados en tiempo real por el sistema de prevención de intrusiones, mientras que la

respuesta activa de la plataforma es la desencadenada por la detección de alertas de seguridad en el servidor de correlación derivadas de la recolección y análisis de los eventos generados en los diferentes dispositivos de la red (firewall, IPS, servidor web, etc.)

## **6.2 Trabajo futuro**

En este apartado se proponen diferentes formas de ampliar la información contenida en este proyecto que estaba dedicado al análisis de la seguridad en publicaciones web.

- Análisis de este tipo de plataformas para la protección de redes de usuarios.
- Ampliación de los dispositivos integrados en el sistema para configurar un servicio de publicaciones web de doble capa de firewall que separen tanto Front-End como Back-end.
- Generación de bloqueos automáticos a nivel de firewall perimetral para el baneo temporal de los atacantes.

## **7 Referencias**

---

- [1] OWASP Top 10, [https://www.owasp.org/images/5/5f/OWASP\\_Top\\_10\\_-\\_2013\\_Final\\_-\\_Espa%C3%B1ol.pdf](https://www.owasp.org/images/5/5f/OWASP_Top_10_-_2013_Final_-_Espa%C3%B1ol.pdf), Junio 2013.
- [2] Alienvault OSSIM, <https://www.alienvault.com/open-threat-exchange/projects>.
- [3] Suricata, <http://suricata-ids.org/>
- [4] Endian Firewall, <http://www.endian.com/>
- [5] Damn Vulnerable Web Application, <http://www.dvwa.co.uk/>
- [6] Logstash, <http://logstash.net/>
- [7] Elasticsearch, <http://www.elasticsearch.org>
- [8] Kibana, <http://www.elasticsearch.org/overview/kibana/>
- [9] Event Correlation Systems - The New Threat Frontline, <http://www.giac.org/paper/gsec/2607/event-correlation-systems-threat-frontline/104476>, 2003
- [10] Understanding IPS and IDS: Using IPS and IDS together for Defense in Depth, <https://www.sans.org/reading-room/whitepapers/detection/understanding-ips-ids-ips-ids-defense-in-depth-1381>, 2004
- [11] Logging and Monitoring to Detect Network Intrusions and Compliance Violations in the Environment, <http://www.sans.org/reading-room/whitepapers/detection/logging-monitoring-detect-network-intrusions-compliance-violations-environment-33985>, 2012



## **8 Glosario**

---

API	Application Programming Interface
SIEM	Security information and event management
OSSIM	Open Source SIEM
XSS	Cross-Site Scripting
CSRF	Cross Site Request Forgery
SQL	Structured Query Language
OWASP	Open Web Application Security Project
IPS	Intrusion Prevention System
IDS	Intrusion Detection System
NAT	Network Address Translation
DNS	Domain Name Server
VPN	Virtual Private Network
IP	Internet Protocol
CPU	Central Processing Unit
RAM	Random Access Memory
HTML	HyperText Markup Language
VLAN	Virtual Local Area Network
NTP	Network Time Protocol
LAMP	Linux Apache MySql PHP
PHP	HyperText Preprocessor



## 9 Anexos

### 9.1 Manual de configuración de Endian Firewall

A continuación se hace un resumen de las configuraciones que se han establecido en el dispositivo que actúa como firewall perimetral de la plataforma.

La primera parte de la configuración de este dispositivo es configurar las interfaces del mismo con sus direcciones IP correspondientes:

- Interfaz de gestión (VERDE): 10.147.104.2
- Interfaz de servicio (NARANJA): 10.147.105.1
- Interfaz externa (ROJA): 172.18.169.11

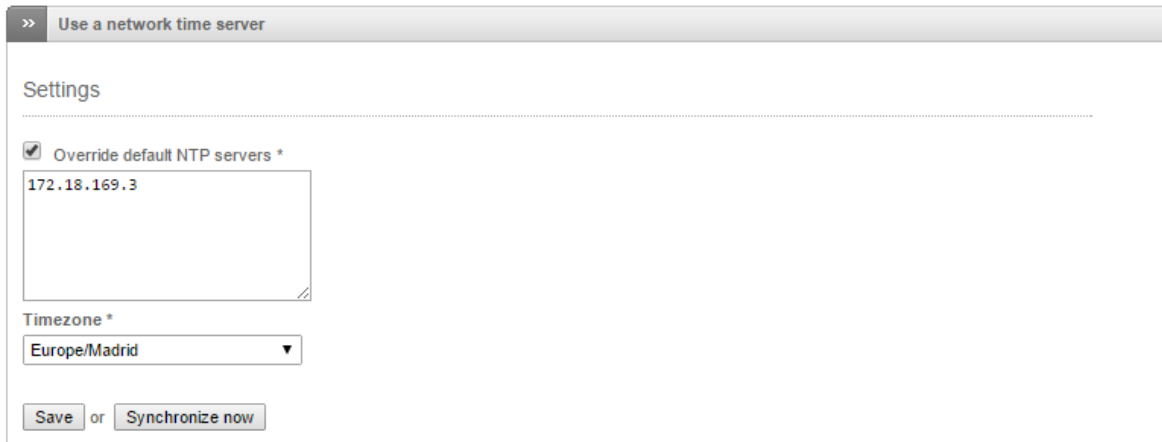
En la siguiente figura se puede observar esta configuración en el dispositivo:

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
  link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
  inet 127.0.0.1/8 brd 127.255.255.255 scope host lo
  inet6 ::1/128 scope host
    valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
  link/ether 00:0c:29:d2:37:8c brd ff:ff:ff:ff:ff:ff
  inet 172.18.169.11/27 brd 172.18.169.31 scope global eth0
  inet6 fe80::20c:29ff:fed2:378c/64 scope link
    valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
  link/ether 00:0c:29:d2:37:96 brd ff:ff:ff:ff:ff:ff
  inet6 fe80::20c:29ff:fed2:3796/64 scope link
    valid_lft forever preferred_lft forever
4: eth2: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
  link/ether 00:0c:29:d2:37:a0 brd ff:ff:ff:ff:ff:ff
  inet6 fe80::20c:29ff:fed2:37a0/64 scope link
    valid_lft forever preferred_lft forever
6: br1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN
  link/ether 00:0c:29:d2:37:a0 brd ff:ff:ff:ff:ff:ff
  inet 10.147.105.1/25 brd 10.147.105.127 scope global br1
  inet6 fe80::20c:29ff:fed2:37a0/64 scope link
    valid_lft forever preferred_lft forever
7: br0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN
  link/ether 00:0c:29:d2:37:96 brd ff:ff:ff:ff:ff:ff
  inet 10.147.104.2/25 brd 10.147.104.127 scope global br0
  inet6 fe80::20c:29ff:fed2:3796/64 scope link
    valid_lft forever preferred_lft forever
8: tap0: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN qlen 100
  link/ether 9e:f4:ca:4f:ae:63 brd ff:ff:ff:ff:ff:ff
  inet 10.100.100.1/24 brd 10.100.100.255 scope global tap0
  inet6 fe80::9cf4:c4ff:fe4f:ae63/64 scope link
    valid_lft forever preferred_lft forever
```

Figura 9-1: Estado interfaces firewall perimetral

Una vez tenemos las interfaces configuradas, se procede a la configuración de los servicios básicos del Firewall.

- Configuración del servicio NTP.



Use a network time server

Settings

Override default NTP servers \*

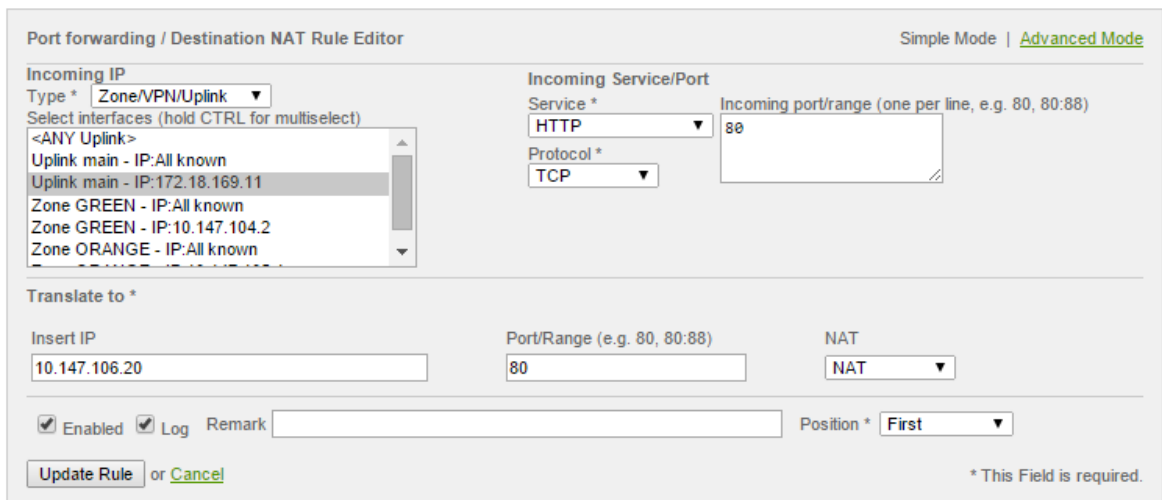
172.18.169.3

Timezone \*

Europe/Madrid

Save or Synchronize now

- Configuración de la regla de NAT para la publicación del servicio web.



Port forwarding / Destination NAT Rule Editor

Simple Mode | [Advanced Mode](#)

Incoming IP

Type \* Zone/VPN/Uplink

Select interfaces (hold CTRL for multiselect)

<ANY Uplink>

Uplink main - IP:All known

Uplink main - IP:172.18.169.11

Zone GREEN - IP:All known

Zone GREEN - IP:10.147.104.2

Zone ORANGE - IP:All known

Incoming Service/Port

Service \* HTTP

Incoming port/range (one per line, e.g. 80, 80:88)

80

Protocol \* TCP

Translate to \*

Insert IP 10.147.106.20

Port/Range (e.g. 80, 80:88) 80

NAT NAT

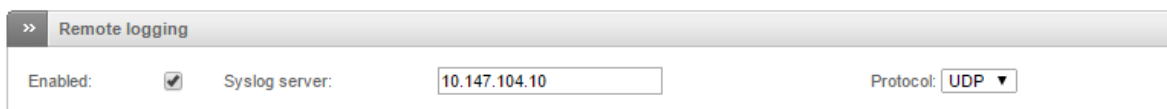
Enabled  Log Remark

Position \* First

Update Rule or Cancel

\* This Field is required.

- Configuración del envío de eventos vía SYSLOG al servidor de correlación.



Remote logging

Enabled:  Syslog server: 10.147.104.10 Protocol: UDP



- Configuración del servicio de Proxy.

## HTTP proxy: Configuration

>> Configuration | Access Policy | Authentication | Web Filter | AD join | HTTPS Proxy

Enable HTTP Proxy

GREEN  
transparent

ORANGE  
not transparent

Proxy settings ?

Port used by proxy \*  
8080

Error Language \*  
English

Visible Hostname used by proxy

Email used for notification (cache admin)

Maximum download size (incoming in KB) \*  
0

Maximum upload size (outgoing in KB) \*  
0

Allowed ports and ssl ports ?

Log settings ?

Bypass transparent proxy ?

Cache management ?

Upstream proxy ?

Upstream proxy  
 Use upstream proxy

Upstream server \*  
172.18.204.79

Upstream port \*  
12345

Upstream username

Upstream password

Client username forwarding  
 Forward username to upstream proxy

Client ip forwarding  
 Forward ipaddress to upstream proxy

Save

\* This Field is required.

- Configuración de la VPN de gestión.

## OpenVPN - Virtual Private Networking

>> Server configuration

Enable OpenVPN server

OpenVPN settings

Authentication type  
PSK (username/password)

Server certificate  
Certificate configuration \* 172.18.169.11  
Use selected certificate [View details](#)

Certificate Authority  
ca  
[Download certificate](#)

\* This Field is required.

OpenVPN server configuration

Bind only to  Port \* 1194

Network options

Device type TAP Protocol TCP

Bridged  VPN Subnet 10.100.100.0/24

Advanced options

Allow multiple connections from one account

Block DHCP responses coming from tunnel  Don't block traffic between clients

Push options

Push these nameservers  Push these networks

Nameservers

Networks 10.147.104.0/25

Push this domain  Domain

or [Cancel](#) \* This Field is required.

Con estas configuraciones, el firewall perimetral queda perfectamente configurado.

## 9.2 Manual de configuración del correlador de eventos

Para llevar a cabo la configuración del correlador de eventos, es necesaria la activación de varios módulos del sistema.

Lo primero de todo es configurar el módulo denominado “ossim-agent” que es el encargado de recolectar la información de las diferentes fuentes y llevar a cabo la normalización de los datos. Este paso es muy importante ya que de esta manera, todos los eventos recogidos tienen el mismo formato lo que hace que su almacenamiento y análisis sea el mismo, sea cual sea el origen. Para ello necesitamos activar en el siguiente fichero de configuración las fuentes de datos de las que recibimos información.

```
[plugins]
iptables=/etc/ossim/agent/plugins/iptables.cfg
rtla=/etc/ossim/agent/plugins/rtla.cfg
suricata=/etc/ossim/agent/plugins/suricata.cfg
```

**Figura 9-2: Normalizadores activados**

En este caso, únicamente disponemos de 3 fuentes de datos integradas con el correlador de eventos. Estos ficheros cargados, están compuestos por un conjunto de expresiones regulares que se encargan de recolectar la información contenida en los ficheros de log que llegan al equipo.

Después de la configuración del normalizador de logs, es necesario llevar a cabo la configuración del servidor de correlación. Para ello, lo primero que necesitamos hacer es definir las redes que vamos a proteger para que el dispositivo sepa distinguir entre lo que es interno y lo que es externo, que en este caso definiremos como red interna la red 10.147.106.0/25 que queremos proteger.

Por último quedaría la configuración de las reglas de correlación para la detección de alertas de seguridad. A continuación se muestra un ejemplo de regla de correlación.

```
<directive id="45186" name="AV Web attack, SQL injection attacks detected against DST_IP" priority="3">
  <rule type="detector" name="SQL injection attempt detected" reliability="6" occurrence="1" from="ANY" to="ANY" port_from="ANY" port_to="ANY" plugin_id="7060" plugin_sid="31103" protocol="TCP">
    <rules>
      <rule type="detector" name="SQL injection attempt detected" plugin_id="7060" plugin_sid="31103" reliability="6" occurrence="1" from="1:SRC_IP" to="1:DST_IP" port_from="ANY" port_to="ANY" time_out="360" protocol="TCP">
        <rules>
          <rule type="detector" name="SQL injection attempt detected" plugin_id="7060" plugin_sid="31103" reliability="6" occurrence="10000" from="1:SRC_IP" to="1:DST_IP" port_from="ANY" port_to="ANY" time_out="43200" protocol="TCP"/>
        </rules>
      </rule>
    </rules>
  </rule>
</directive>
```

Figura 9-3: Ejemplo regla de correlación

Se puede observar en la figura anterior la estructura por niveles que se utiliza para la configuración de las alertas de seguridad. Veamos qué significado tienen los campos presentes en la configuración de las alertas

- Priority: Fija la importancia de la alerta
- Reliability: Se trata del grado de confianza que hay en ese nivel de que nos encontremos ante un incidente real
- From y To: Origen y destino del evento que vamos a recoger.
- Plugin\_id: Tecnología que detecta el evento
- Plugin\_sid: Tipo de evento dentro de la tecnología.
- Occurrence: Número de eventos necesarios para que ese cumpla ese nivel.
- Time\_out: Tiempo máximo que nos mantenemos a la espera en un nivel concreto a la espera de eventos.

Es importante la siguiente fórmula, que es la que nos fija si tenemos alerta de seguridad o no en Alienvault:

$$\text{Riesgo} = (\text{Valor del activo} * \text{Prioridad} * \text{Fiabilidad}) / 25$$

Si este riesgo es mayor o igual que 1, entonces el correlador generará una alerta de seguridad.

## **10 Presupuesto**

---

El presupuesto se compone de las siguientes partes:

- Presupuesto de ejecución material.
- Gastos generales y Beneficio industrial.
- Honorarios por la redacción y dirección del proyecto.
- Presupuesto total.

El Presupuesto de ejecución material y los Gastos generales y beneficio industrial constituyen el Presupuesto por ejecución por contrata que, junto con los Honorarios por la redacción y dirección del proyecto, integran el Presupuesto total.

Todas las cantidades aparecen expresadas en euros.

### **10.1 Presupuesto de ejecución material**

El Presupuesto de ejecución material consta de Costes de mano de obra y Costes de recursos materiales. No se incluirán los Honorarios de dirección del proyecto que serán considerados aparte.

#### **10.1.1 Descomposición en tareas**

- **Tarea 1:**

*Objetivo:* Estudio de las tecnologías implicadas en las plataformas de publicaciones web a nivel de funcionalidad y seguridad.

*Duración:* 1 mes.

*Esfuerzo:* Ingeniero Superior, 1 personas-mes.

- **Tarea 2:**

*Objetivo:* Diseño de la plataforma teniendo en cuenta las necesidades de seguridad analizadas y elección de los diferentes paquetes de software utilizados para el despliegue del laboratorio de pruebas.

*Duración:* 2 meses.

*Diseño, implementación y análisis de un sistema de detección y respuesta activa*

*Esfuerzo:* Ingeniero superior, 1 personas-mes.

- **Tarea 3:**

*Objetivo:* Análisis de las necesidades de interconexión de los dispositivos, configuración de la máquina utilizada para la creación del entorno virtual y creación de las máquinas virtuales necesarias así como el despliegue y configuración de los diferentes dispositivos.

*Duración:* 2.5 meses.

*Esfuerzo:* Ingeniero superior, 1 personas-mes.

- **Tarea 4:**

*Objetivo:* Pruebas funcionales de la plataforma para comprobar que el funcionamiento se corresponde con el diseño.

*Duración:* 0.5 meses.

*Esfuerzo:* Ingeniero superior, 0.25 personas-mes.

- **Tarea 5:**

*Objetivo:* Reconfiguración de las diferentes máquinas para afinar los niveles de seguridad.

*Duración:* 1 mes.

*Esfuerzo:* Ingeniero superior, 1.5 personas-mes.

- **Tarea 6:**

*Objetivo:* Realización de las pruebas de intrusión en los distintos escenarios propuestos.

*Duración:* 1 mes.

*Esfuerzo:* Ingeniero superior, 1 personas-mes.

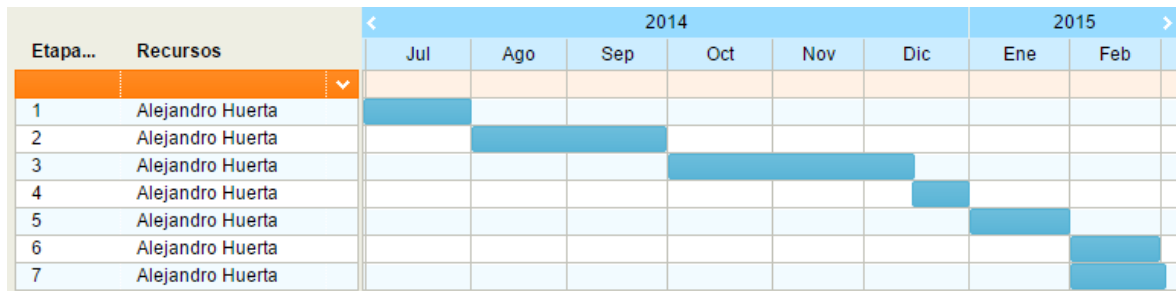
- **Tarea 7:**

*Objetivo:* Redacción de la documentación asociada al Proyecto Final de Carrera.

*Duración:* 1 meses.

*Esfuerzo:* Ingeniero superior, 0.75 personas-mes, Administrativo, 0.5 personas-mes.

La figura 10-1 muestra un diagrama de Gantt con las relaciones de dependencia entre las distintas tareas.



**Figura 10-1: Diagrama de Gantt**

Con esto, se obtiene un plazo de ejecución del proyecto de 8 meses.

### **10.1.2 Costes de mano de obra.**

Para la realización del proyecto se requieren los siguientes perfiles profesionales:

- Un Ingeniero Superior en Telecomunicaciones, encargado del planteamiento, desarrollo e implementación del trabajo técnico.
- Un Administrativo, encargado de la redacción, presentación y encuadernación del proyecto.

La estimación de los costes se realiza en base a los siguientes datos:

- Cotizaciones según el Régimen General de la Seguridad Social. El ingeniero pertenece al grupo 1 y el administrativo pertenece al grupo 7.
- Jornada laboral de 8 horas/día y 21 días laborables/mes.

Con estos datos y la distribución del trabajo mostrado en el apartado 10.1.1, se obtiene lo siguiente:

<b>Costes Salariales</b>		
<b>Concepto</b>	<b>Grupo 1</b>	<b>Grupo 7</b>
Base cotizable máxima anual	43.272,00 €	43.272,00 €
Contingencias comunes (23,6%)	10.212,19 €	3.776,00 €
Desempleo F.G.S. y Formación profesional (7,5%)	3.245,40 €	1.200,00 €
Accidentes de trabajo y enfermedades profesionales	3.606,00 €	1.333,33 €
Coste de la seguridad social	17.063,59 €	6.309,33 €
Salario bruto anual	45.000,00 €	16.000,00 €
Coste salarial anual	62.063,59 €	22.309,33 €
Coste salarial por hora	30,79 €	11,07 €
Número de horas	1.407,00 €	84,00 €
<b>Coste total</b>	<b>43.315,22 €</b>	<b>929,56 €</b>

**Tabla 10-1: Costes Salariales**

<b>Costes de la mano de obra</b>	
<b>Concepto</b>	<b>Coste</b>
Ing. Superior en Telecomunicación	43.315,22 €
Administrativo	929,56 €
<b>Coste total</b>	<b>44.244,77 €</b>

**Tabla 10-2: Costes de la mano de obra**

### 10.1.3 Costes de los recursos materiales

En la siguiente tabla constan los costes de los recursos materiales empleados, considerando un periodo de amortización para el hardware y el software de 3 años.

En primer lugar se indican los costes totales y después se imputarán las cuantías correspondientes a la amortización de los recursos durante su periodo de utilización en el desarrollo del proyecto.

<b>Recursos Hardware</b>			
<b>Concepto</b>	<b>Coste total</b>	<b>Meses</b>	<b>Coste real</b>
Equipo de desarrollo	1.200,00 €	8	266,67 €
Servidor	6.000,00 €	8	1.333,33 €
Equipo generación de documentación	600,00 €	2	33,33 €
<b>Total recursos Hardware</b>			<b>1.633,33 €</b>

**Tabla 10-3: Gastos recursos Hardware**



Recursos Software			
Concepto	Coste total	Meses	Coste real
Sistema operativo Debian 7	- €	8	- €
Endian Firewall	- €	8	- €
Alienvault OSSIM (x2)	- €	8	- €
ESXi v5.5	- €	8	- €
Microsoft Office 2013	269,00 €	2	14,94 €
Microsoft Windows 8.1	119,00 €	8	26,44 €
Sistema operativo Ubuntu 14.04	- €	8	- €
<b>Total recursos Software</b>			<b>41,39 €</b>

Tabla 10-4: Gastos recursos Software

Recursos materiales	
Concepto	Coste
Recursos Hardware	1.633,33 €
Recursos Software	41,39 €
Consumibles, material fungible y de oficina	200,00 €
<b>Total recursos materiales</b>	<b>1.874,72 €</b>

Tabla 10-5: Gastos recursos materiales

#### 10.1.4 Coste total de los recursos

La suma de los costes por mano de obra y de los costes por recursos materiales es lo que constituye el *Presupuesto de Ejecución Material (P.E.M.)*.

Presupuesto de ejecución material	
Concepto	Coste
Coste mano de obra	44.244,77 €
Coste recursos materiales	1.874,72 €
<b>Total</b>	<b>46.119,49 €</b>

Tabla 10-6: Presupuesto de ejecución material

## 10.2 Gastos generales y beneficio industrial

Bajo Gastos generales se incluyen todos aquellos gastos derivados de la utilización de instalaciones, cargas fiduciarias, amortizaciones, gastos fiscales, etc. Con esto, el Presupuesto de Ejecución por contrata queda como sigue:

<b>Presupuesto de ejecución por contrata</b>	
<b>Concepto</b>	<b>Coste</b>
Presupuesto de ejecución material	46.119,49 €
Gastos generales (16% del PEM)	7.379,12 €
Beneficio industrial (6% del PEM)	2.767,17 €
<b>Total</b>	<b>56.265,78 €</b>

**Tabla 10-7: Presupuesto de ejecución por contrata**

## 10.3 Honorarios por redacción y dirección del proyecto

Los Honorarios que recomienda aplicar el Colegio Oficial de Ingenieros de Telecomunicación, tanto para la redacción como para la dirección del proyecto son los asociados a Trabajos tarifados por tiempo empleado, con un valor de un 5.6%.

## 10.4 Presupuesto total

Para finalizar, sumando todas las cantidades anteriores y aplicando el 21% de IVA, se obtiene el presupuesto total.

<b>Presupuesto total</b>	
<b>Concepto</b>	<b>Coste</b>
Presupuesto de ejecución material	56.265,78 €
Honorarios por dirección	3.150,88 €
Honorarios por dirección	176,45 €
Subtotal	59.593,11 €
I.V.A. (21%)	3.375,95 €
<b>Total</b>	<b>62.969,06 €</b>

**Tabla 10-8: Presupuesto total**

El presupuesto total del proyecto asciende a SESENTA Y DOS MIL NOVECIENTOS SESENTA Y NUEVE Euros CON SEIS céntimos.

Madrid, Febrero de 2015

El Ingeniero Jefe de Proyecto

## **PLIEGO DE CONDICIONES**

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un Diseño, implementación y análisis de un sistema de detección y respuesta activa. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

### **Condiciones generales**

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partida alzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella.

Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

### **Condiciones particulares**

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.