Evaluaciones de Seguridad en entornos TIC:

- ✔ Fundamentos.
- ✔ Metodologías.
- ✔ Herramientas.

Autor: Diego Camacho Moreno

Consultor: D. Miquel Font Roselló

Trabajo Fin Carrera: Ingeniería Técnica de Telecomunicación

10/01/13

UOC
Universitat Oberta de Catalunya

Agradecimientos

Este Trabajo de Fin de Carrera marca, sin duda alguna, un hito en mi vida. Por ello, no puedo más que agradecer a todos aquellos que, de alguna manera u otra, han contribuido a la consecución de este logro. En primer lugar, a mis familiares y amigos que, de forma comprensiva, han sabido perdonar mis prolongadas ausencias. También quiero recordar a todos aquellos que no creyeron en la viabilidad de esta empresa. Con su falta de fe, redoblaron en mí mi motivación y mi empeño por alcanzar el objetivo. Sin embargo, si hay alguien de quien me acuerdo en este instante, es de mi mujer Inma y de mi hijo Diego. Ella, no sólo ha creído en mí en todo momento, sino que además me ha ayudado, soportado, animado y sustentado para que yo pudiese alcanzar este sueño. Él, con sólo una sonrisa suya, puede hacer desaparecer cualquier traza de dificultad o de desánimo.

Por último, pero no menos importante, quiero agradecer la inestimable ayuda de D. Miquel Font Roselló en el desarrollo de este Trabajo de Fin de Carrera. Su continua tutorización y directriz han contribuido significativamente a dirigir este proyecto a buen puerto.

"*Possunt quia posse videntur.*" (Pueden los que creen que pueden).- Publius Vergilius Maro (70 a.C. - 19 a.C.)

Palabras Clave

Hacking – Hacking ético – Criptografía – Seguridad – Seguridad de la Información – Seguridad de red – Seguridad informática – Test de Penetración - Evaluaciones de Seguridad – Auditoría de Seguridad – Análisis de Vulnerabilidades – Seguridad Redes Inalámbricas – Malware – Seguridad de Servidores Web – Seguridad de Aplicaciones.

Resumen

Con la globalización de la red de redes y la proliferación de servicios de todo tipo que sobre ésta se ofrecen, la evaluación de la seguridad de las aplicaciones, los sistemas, las redes, el software base, el middleware, etc.; toma una relevancia cada vez mayor. Además, debido a la relativamente reciente aparición de diferentes marcos legislativos en múltiples países, las evaluaciones de seguridad son cada vez más frecuentes. Llegando a ser, en algunos casos, de carácter obligatorio y con una periodicidad definida.

Al igual que los agentes policiales que tienen que aprender la jerga, las acciones y los comportamientos de los criminales para poder capturarlos, las evaluaciones de seguridad permiten encontrar y reparar las vulnerabilidades y los “agujeros” de seguridad en los sistemas y redes. Para ello, se tiene que pensar como un criminal y usar las mismas tácticas (i.e. metodologías), herramientas y procesos que ellos emplean.

El presente trabajo es una introducción a lo que se conoce como, entre otros términos, Evaluación de Seguridad TIC. En él se recogen los fundamentos teóricos y técnicos sobre las que se basan, la jerga y definiciones empleadas, las metodologías que se siguen y las herramientas que se utilizan.

Tabla de Contenidos

1 Introducción....1
1.1 Motivaciones....1
1.2 Objetivos....1
1.3 Enfoque y metodología....2
1.3.1 Entregables....2
1.3.2 Planificación....2
1.3.3 Laboratorio de pruebas....4
1.4 Esbozo de la memoria....14
2 Fundamentos de evaluaciones de seguridad....15
2.1 Elementos básicos de la Seguridad de la Información....15
2.2 Hackers Éticos: definición y tipos....16
2.3 Terminología Hacking y tipos de ataques....16
3 Metodologías para evaluaciones de seguridad....19
3.1 The Open Source Security Testing Methodology Manual (OSSTMM)....19
3.2 The Information Systems Security Assesment Framework (ISSAF)....20
3.3 Metodología del EC Council....21
4 Reconocimiento....23
4.1 Descripción general....23
4.2 Herramientas....26
4.3 Demostración....27
5 Escaneo y enumeración....28
5.1 Descripción general....28
5.2 Herramientas....33
5.3 Demostración....38
6 Obtención de acceso....39
6.1 Ataques a nivel de red: Sniffers y evasión....39
6.1.1 Descripción general....39
6.1.2 Herramientas....41
6.1.3 Demostración....42
6.2 Ataques a nivel de sistema....43
6.2.1 Descripción general....43
6.2.2 Herramientas....46
6.2.3 Demostración....48
6.3 Ataques de bajo perfil técnico: Ingeniería Social....49
6.3.1 Descripción general....49
6.4 Ataques a servidores y aplicaciones web....52
6.4.1 Descripción general....52
6.4.2 Herramientas....55
6.4.3 Demostración....57
6.5 Ataques a redes inalámbricas....58
6.5.1 Descripción general....58
6.5.2 Herramientas....63
6.5.3 Demostración....64
6.6 Ataques a través de Malware....65

6.6.1 Descripción general....65
6.6.2 Herramientas....70
6.6.3 Demostración....71
7 Conclusiones....72
7.1 Líneas de ampliación....72
8 Glosario....73
9 Bibliografía....84

Índice de Figuras

Figura 1: planificación proyecto a nivel alto.....2
Figura 2: planificación proyecto a nivel medio.....3
Figura 3: planificación proyecto a nivel detalle.....3
Figura 4: diagrama de red laboratorio.....4
Figura 5: fases de la metodología OSSTMM.....19
Figura 6: marco de trabajo ISSAF.....20
Figura 7: subfases hacking ético según el EC Council.....22

1 Introducción

1.1 Motivaciones

Gobiernos, cuerpos de seguridad, empresas, instituciones financieras, hospitales, empresas privadas, etc., amasan una gran cantidad de información confidencial sobre empleados, clientes, productos, investigaciones, estados financieros, etc. La mayor parte de esta información es recogida, procesada y almacenada en sistemas de información electrónicos y transmitida, a través de redes, a otros sistemas. Para las personas individuales, la seguridad de la información puede tener un efecto significativo sobre su privacidad.

Por tanto, en el escenario antes descrito, es un requisito, a veces ético y/o legal, el comprobar si los controles de seguridad aplicados por una organización están cumpliendo con la misión encomendada, si son suficientes, adecuados y óptimos. Para realizar dichas comprobaciones, existen las evaluaciones de seguridad, objeto principal del presente trabajo.

1.2 Objetivos

Los objetivos del presente trabajo son:

- ✔ Presentar una introducción a la Seguridad de la Información.
- ✔ Presentar una introducción a las evaluaciones de seguridad.
- ✔ Describir las diferentes metodologías estándares y estándares de facto que se usan en la actualidad para llevar a cabo las evaluaciones de seguridad.
- ✔ Describir las fases que componen la metodología más usada en la actualidad en la realización de evaluaciones de seguridad.
- ✔ Enumerar y describir las herramientas que componen el estado del arte en las evaluaciones de seguridad.
- ✔ Relacionar los conceptos teórico-técnicos sobre los que se basan las evaluaciones de seguridad con los conceptos aprendidos durante el desarrollo de la carrera.

- ✔ Reforzar los conocimientos teóricos con ejercicios prácticos realizados sobre un laboratorio virtual creado *ad hoc*.
- ✔ Utilizar técnicas de gestión de proyectos para garantizar la correcta finalización del trabajo dentro de los límites de tiempo, cumpliendo con los requisitos y restricciones marcados y con el grado de calidad requerido en el ámbito académico de un Trabajo de Fin de Carrera.

1.3 Enfoque y metodología

1.3.1 Entregables

Como entregables frutos de la realización del presente trabajo se obtendrán:

- ✔ La presente memoria.
- ✔ Un conjunto de vídeos con la grabación de los laboratorios prácticos llevados a cabo como demostración de algunas de las técnicas / herramientas utilizadas en las evaluaciones de seguridad en entornos TIC.

1.3.2 Planificación

A continuación se incluye la planificación temporal de este Trabajo de Fin de Carrera a diferentes niveles de detalle:

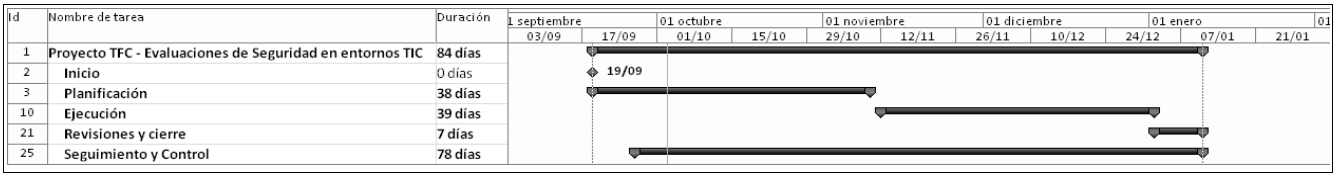
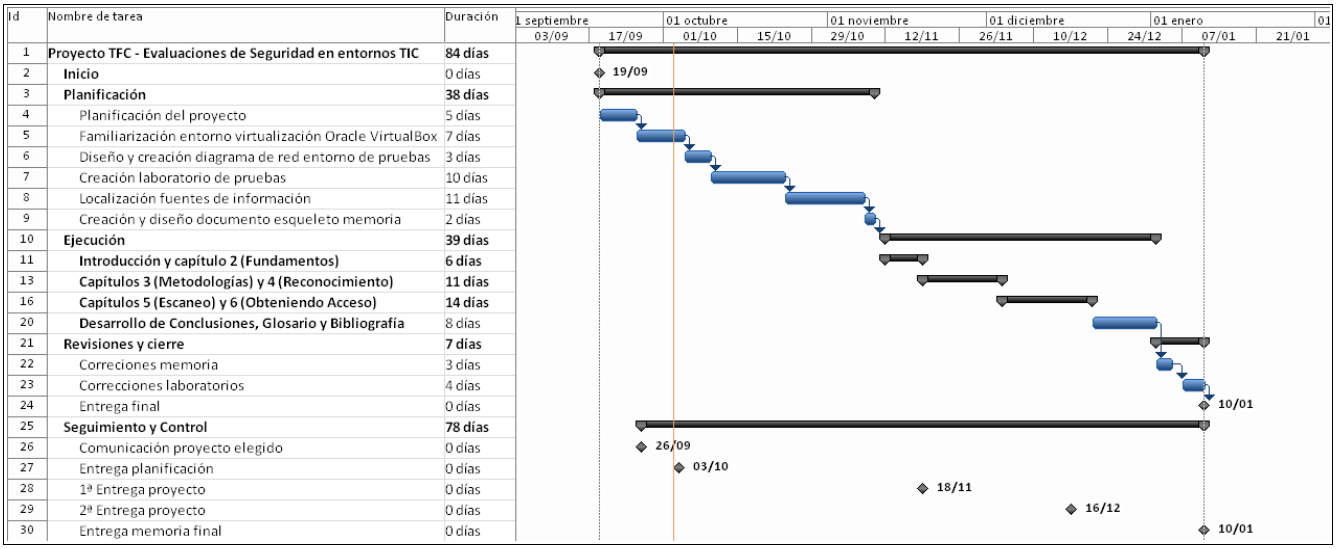
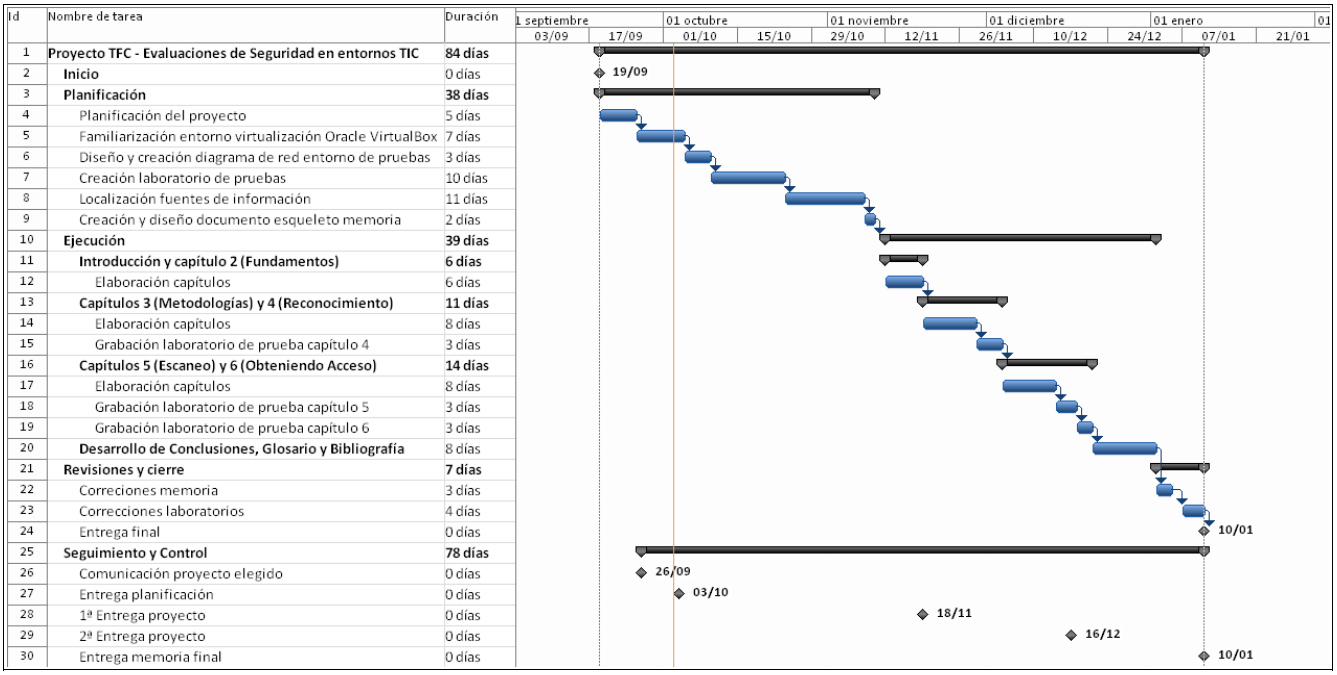


Figura 1: planificación proyecto a nivel alto.



Id	Nombre de tarea	Duración
1	**Proyecto TFC - Evaluaciones de Seguridad en entornos TIC**	**84 días**
2	**Inicio**	0 días
3	**Planificación**	**38 días**
4	Planificación del proyecto	5 días
5	Familiarización entorno virtualización Oracle VirtualBox	7 días
6	Diseño y creación diagrama de red entorno de pruebas	3 días
7	Creación laboratorio de pruebas	10 días
8	Localización fuentes de información	11 días
9	Creación y diseño documento esqueleto memoria	2 días
10	**Ejecución**	**39 días**
11	**Introducción y capítulo 2 (Fundamentos)**	**6 días**
13	**Capítulos 3 (Metodologías) y 4 (Reconocimiento)**	**11 días**
16	**Capítulos 5 (Escaneo) y 6 (Obteniendo Acceso)**	**14 días**
20	**Desarrollo de Conclusiones, Glosario y Bibliografía**	8 días
21	**Revisiones y cierre**	**7 días**
22	Correciones memoria	3 días
23	Correcciones laboratorios	4 días
24	Entrega final	0 días
25	**Seguimiento y Control**	**78 días**
26	Comunicación proyecto elegido	0 días
27	Entrega planificación	0 días
28	1ª Entrega proyecto	0 días
29	2ª Entrega proyecto	0 días
30	Entrega memoria final	0 días

Figura 2: planificación proyecto a nivel medio.



Id	Nombre de tarea	Duración
1	**Proyecto TFC - Evaluaciones de Seguridad en entornos TIC**	**84 días**
2	**Inicio**	0 días
3	**Planificación**	**38 días**
4	Planificación del proyecto	5 días
5	Familiarización entorno virtualización Oracle VirtualBox	7 días
6	Diseño y creación diagrama de red entorno de pruebas	3 días
7	Creación laboratorio de pruebas	10 días
8	Localización fuentes de información	11 días
9	Creación y diseño documento esqueleto memoria	2 días
10	**Ejecución**	**39 días**
11	**Introducción y capítulo 2 (Fundamentos)**	**6 días**
12	Elaboración capítulos	6 días
13	**Capítulos 3 (Metodologías) y 4 (Reconocimiento)**	**11 días**
14	Elaboración capítulos	8 días
15	Grabación laboratorio de prueba capítulo 4	3 días
16	**Capítulos 5 (Escaneo) y 6 (Obteniendo Acceso)**	**14 días**
17	Elaboración capítulos	8 días
18	Grabación laboratorio de prueba capítulo 5	3 días
19	Grabación laboratorio de prueba capítulo 6	3 días
20	**Desarrollo de Conclusiones, Glosario y Bibliografía**	8 días
21	**Revisiones y cierre**	**7 días**
22	Correciones memoria	3 días
23	Correcciones laboratorios	4 días
24	Entrega final	0 días
25	**Seguimiento y Control**	**78 días**
26	Comunicación proyecto elegido	0 días
27	Entrega planificación	0 días
28	1ª Entrega proyecto	0 días
29	2ª Entrega proyecto	0 días
30	Entrega memoria final	0 días

Figura 3: planificación proyecto a nivel detalle.

1.3.3 Laboratorio de pruebas

Para reforzar los conocimientos teóricos con ejercicios prácticos, se ha creado un laboratorio virtual que se describe con el siguiente diagrama:

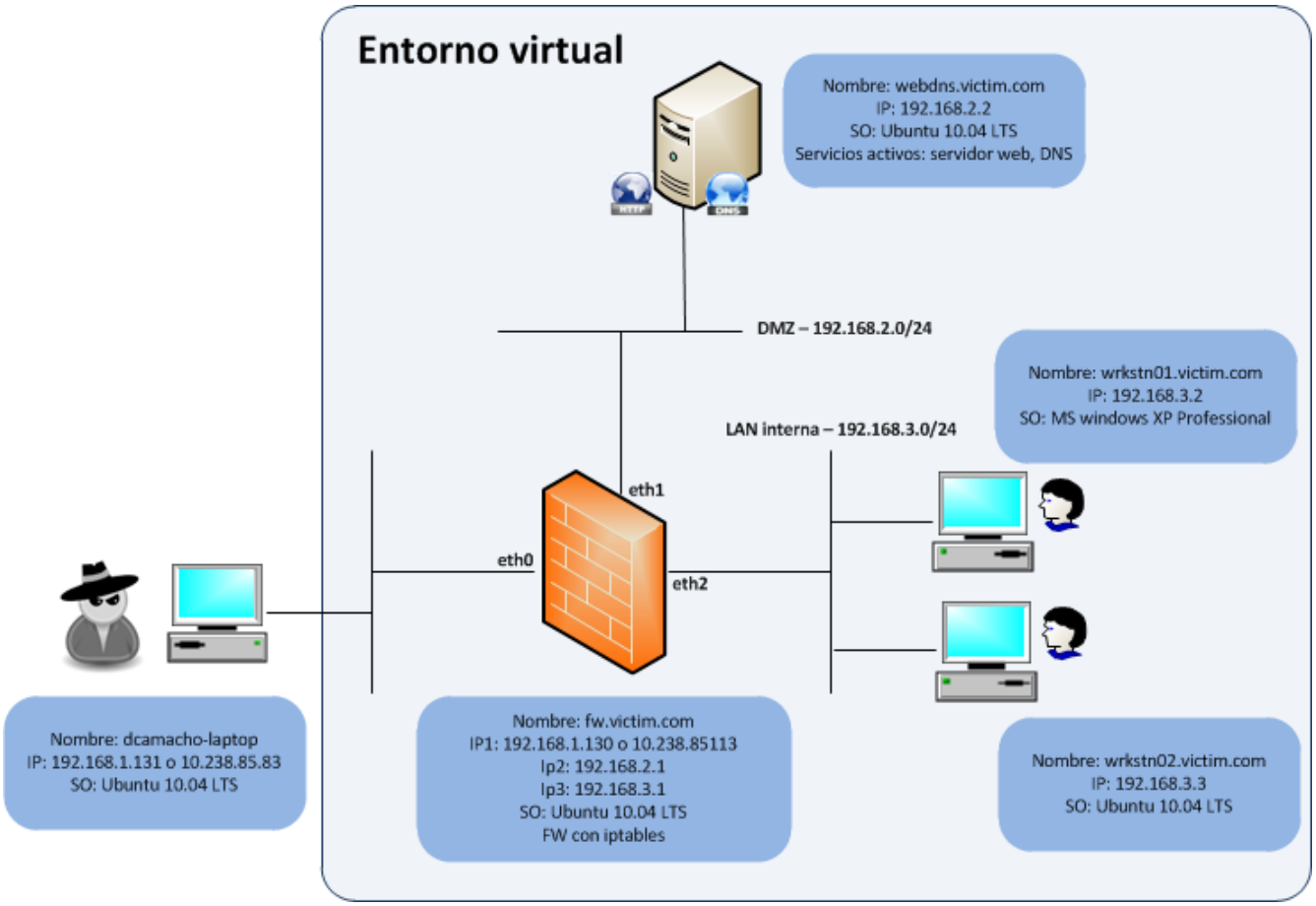


Figura 4: diagrama de red laboratorio.

Se trata de un laboratorio virtual porque cuatro de las máquinas representadas en el diagrama son maquinas virtualizadas con el producto *Oracle Virtual Box*. Este producto forma parte de la familia de productos de virtualización de *Oracle Corporation*.

Virtual Box se instala sobre un sistema operativo *anfitrión* como una aplicación. Esta aplicación permite que sistemas operativos adicionales, conocidos como sistemas operativos *huéspedes*, puedan ser cargados y ejecutados cada uno con su propio entorno virtual. Entre los sistemas operativos de tipo anfitrión soportados por el producto *Virtual Box* tenemos los siguientes: *Linux*, *Mac OS X*, *Windows XP*, *Windows Vista*, *Windows 7*, *Windows 8*, *Solaris* y *OpenSolaris*. Entre los principales sistemas operativos de tipo huésped soportados por el producto *Virtual Box* tenemos los siguientes: *Linux*, *Windows* (diferentes versiones), *BSD*, *OS/2*, *Solaris*, etc.

El laboratorio trata de modelar la arquitectura de red que presentan muchas organizaciones con servicios y presencia en Internet. El laboratorio se compone de cinco máquinas, la máquina *host* (*dcamacho-laptop*) y cuatro máquinas *guest* (*fw.victim.com*, *webdns.victim.com*, *wrkstn01.victim.com* y *wrkstn02.victim.com*).

Desde la máquina con nombre *dcamacho-laptop*, se realizarán la mayoría de ataques, escáneres, intentos de intrusión, etc. Simulando, por tanto, ser una máquina bajo el control de un hacker malicioso intentando circunvalar los controles de seguridad de la organización ficticia *victim.com*.

La máquina con nombre *fw.victim.com*, representa la máquina que hace de pasarela entre las subredes DMZ, LAN interna de la organización e Internet. Realiza funciones de NAT (*Network Address Translation*) tanto de fuente como de destino para ocultar las direcciones IP de la subredes DMZ y LAN interna. Además, realiza funciones de filtrado de paquetes (i.e. de cortafuegos) a través de la implementación de un cortafuegos en arquitectura *multihomed* (i.e. un sistema con varias tarjetas de interfaz de red) con la herramienta *Iptables*. Un cortafuegos es un componente hardware y/o software que ayuda a mantener la seguridad de una red. Su principal objetivo es controlar el tráfico de entrada y de salida analizando los paquetes de datos y determinando si autoriza o no el tráfico conforme a un conjunto de reglas establecidas. Los cortafuegos suelen ser ubicados lógicamente de forma que se convierten en nodos de interconexión entre redes con diferentes perfiles de confianza. Existen diferentes generaciones de cortafuegos según las capacidades que éstos muestran, las cuáles son fruto de la evolución en el tiempo de los cortafuegos. En el laboratorio de pruebas, se va a utilizar un cortafuegos de segunda generación; es decir, un cortafuegos con estado (*statefull firewall*). Los cortafuegos con estado son aquellos que tienen la capacidad de almacenar suficientes paquetes hasta llegar a conocer el estado de una conexión. De esta forma, puede discernir si un paquete está iniciando una conexión, forma parte de una conexión establecida o no forma parte de ninguna conexión. A continuación, se incluye la configuración específica del cortafuegos implementado con la herramienta *iptables*:

Requerimiento 1	
¿Qué?	El cortafuegos debe iniciarse sin ninguna regla de filtrado establecida.
¿Cómo?	Se eliminarán todas las reglas del cortafuegos.
¿Cuándo?	Cada vez que se active el cortafuegos.
¿Por qué?	Evitar que diferentes configuraciones de cortafuegos se solapen.

Requerimiento 2	
¿Qué?	Cuando un paquete no coincida con ninguna regla debe ser descartado.
¿Cómo?	Se utilizarán las políticas por defecto.
¿Cuándo?	Mientras el cortafuegos esté activado.
¿Por qué?	Para evitar que paquetes que no han sido explícitamente activados o denegados pueden pasar a través del cortafuegos.

Requerimiento 3	
¿Qué?	Las direcciones de los equipos ubicados en las redes DMZ e interna deben quedar ocultas a externos. Así mismo, los puertos de los servicios publicados en el servidor bastión de la red DMZ deben ser diferentes a los puertos bien conocidos asociados a dichos servicios.
¿Cómo?	Se configurará NAT de fuente y de destino.
¿Cuándo?	Mientras el cortafuegos esté activado.
¿Por qué?	Para ahorrar direcciones IP públicas y por motivos de seguridad (ocultando las direcciones IP y puertos bien conocidos utilizados en las redes DMZ e interna al exterior).

Requerimiento 4	
¿Qué?	El cortafuegos recibirá y responderá a peticiones ICMP.
¿Cómo?	Se autorizará el tráfico ICMP de entrada de tipo 8 y se autorizará el tráfico ICMP de salida de tipo 0.
¿Cuándo?	Cuando expresamente así se solicite.
¿Por qué?	Para permitir operaciones de diagnóstico.

Requerimiento 5	
¿Qué?	El cortafuegos permitirá la navegación web por parte de los usuarios de la red interna.
¿Cómo?	Se autorizará el tráfico TCP con origen la red interna y con destino los puertos 80,443 y se autorizará el tráfico TCP establecido con origen los puertos 80,443 y con destino la red interna.
¿Cuándo?	Mientras el cortafuegos esté activado.
¿Por qué?	Los usuarios de la red interna necesitan la navegación web para realizar su desempeño.

Requerimiento 6	
¿Qué?	El cortafuegos permitirá el acceso universal al servicio web activo en el servidor bastión ubicado en la red DMZ.
¿Cómo?	Se autorizará el tráfico TCP con origen universal y con destino los puertos 8080,8443 del servidor bastión ubicado en la red DMZ y se autorizará el tráfico TCP establecido con origen los puertos 8080,8443 del servidor bastión ubicado en la red DMZ y con destino universal.
¿Cuándo?	Mientras el cortafuegos esté activado.
¿Por qué?	El modelo implica la prestación de servicios a través sitios web.

Requerimiento 7	
¿Qué?	El cortafuegos permitirá el acceso universal al servicio de resolución de nombres activo en el servidor bastión ubicado en la red DMZ.
¿Cómo?	Se autorizará el tráfico UDP con origen universal y con destino el puerto 53 del servidor bastión ubicado en la red DMZ y se autorizará el tráfico UDP establecido con origen el puerto 53 del servidor bastión ubicado en la red DMZ y con destino universal.
¿Cuándo?	Mientras el cortafuegos esté activado.
¿Por qué?	El modelo implica el control de la resolución de nombres de la zona *victim.com*.

Requerimiento 8	
¿Qué?	El cortafuegos permitirá el acceso desde la red interna al servicio de la herramienta de configuración WEBMIN activo en el servidor bastión ubicado en la red DMZ.
¿Cómo?	Se autorizará el tráfico TCP con origen la interna y con destino el puerto 10000 del servidor bastión ubicado en la red DMZ y se autorizará el tráfico TCP establecido con origen el puerto 10000 del servidor bastión ubicado en la red DMZ y con destino la red interna.
¿Cuándo?	Mientras el cortafuegos esté activado.
¿Por qué?	El modelo implica la utilización de la herramienta de configuración con interfaz basada en web WEBMIN.

Requerimiento 9	
¿Qué?	El cortafuegos permitirá el acceso desde el servidor bastión ubicado en la red DMZ al servicio de resolución de nombres universal.
¿Cómo?	Se autorizará el tráfico UDP con origen el servidor bastión ubicado en la red DMZ y con destino el puerto 53 universal y se autorizará el tráfico UDP establecido con origen el puerto 53 universal y con destino el servidor bastión ubicado en la red DMZ.
¿Cuándo?	Mientras el cortafuegos esté activado.
¿Por qué?	El servicio DNS ubicado en el servidor bastión de la red DMZ reenviará las consultas que no sean para la zona *victim.com* y que no estén almacenadas en memoria caché al sistema jerárquico DNS universal.

Requerimiento 10	
¿Qué?	El cortafuegos permitirá el acceso a la red interna al servicio de resolución de nombres activo en el servidor bastión ubicado en la red DMZ.
¿Cómo?	Se autorizará el tráfico UDP con origen la red interna y con destino el puerto 53 del servidor bastión ubicado en la red DMZ y se autorizará el tráfico UDP establecido con origen el puerto 53 del servidor bastión ubicado en la red DMZ y con destino la red interna.
¿Cuándo?	Mientras el cortafuegos esté activado.
¿Por qué?	Dado que los usuarios de la red interna necesitan la navegación web para realizar su desempeño, esto implica que los usuarios de la red interna puedan resolver nombres de dominio.

```
#
# UOC
# Subject: End Degree Project.
# Student: Diego Camacho.
# Revision number: 20121027_01
#

#
# IPTABLES
#
# EXPLICACIÓN: IPTABLES es un mandato del sistemas operativo Linux que permite crear, mantener e inspeccionar el cortafuegos del kernel de Linux. El cortafuegos del kernel de Linux está formado por  tablas. Las cuáles, a su vez, están compuestas de cadenas. Las cadenas son listas de reglas que  indican características de filtrado que finalmente son las que determinan si un paquete es o no autorizado en su tránsito. En cada tabla existen una serie de cadenas definidas y, además, nuevas cadenas pueden ser definidas por el usuario. La tabla por defecto es la tabla FILTER, la cuál tiene definida las cadenas INPUT (para paquetes con sockets locales como destino), FORWARD (para paquetes siendo encaminados por el sistema) y OUTPUT (para paquetes generados localmente).

#
# Clear all. Implements requirement nº1.
#
# EXPLICACIÓN: Borramos todas las cadenas definidas por el usuario que pudiera haber (-X) y todas las cadenas predefinidas (-F) de la tabla por defecto (i.e. FILTER).

iptables -X
iptables -F

#
# Default policies. Implements requirement nº2.
#
#
# EXPLICACIÓN: Establecemos las políticas por defecto de las tres cadenas predefinidas (INPUT, FORWARD, OUTPUT) de la tabla por defecto FILTER. La política por defecto es rechazar. Es decir, cualquier paquete que no tenga una coincidencia con una regla será descartado.

iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT DROP
```

```
#
# Source NAT. Implements requirement nº3.
#
# EXPLICACIÓN: realizamos un NAT de fuente sobre todos los paquetes procedentes de la red DMZ y de la red interna (192.168.2.0/23) y que salgan por la interfaz eth0. Es decir, a todos los paquetes provenientes de las redes DMZ e interna se le cambiará la dirección IP origen por la dirección que dinámicamente se le haya asignado a la interfaz eth0.

iptables -t nat -A POSTROUTING -s 192.168.2.0/23 -o eth0 -j MASQUERADE

#
# Destination NAT. Implements requirement nº3.
#
# EXPLICACIÓN: realizamos un NAT de destino sobre todos los paquetes entrantes por la interfaz eth0 y con puerto destino 80 ó 443 por protocolo nivel 4 TCP, o con puerto destino 53 por protocolo nivel 4 UDP.  Esto es necesario por dos motivos: 1º) tenemos servicios publicados en nuestra red DMZ y queremos que puedan ser consumidos desde el exterior; 2º) La dirección IP visible desde el exterior es la dirección IP asignada a la interfaz eth0 de nuestro cortafuegos.

iptables -t nat -A PREROUTING -p tcp --dport 80 -i eth0 -j DNAT --to 192.168.2.2:8080
iptables -t nat -A PREROUTING -p tcp --dport 443 -i eth0 -j DNAT --to 192.168.2.2:8443
iptables -t nat -A PREROUTING -p udp --dport 53 -i eth0 -j DNAT --to 192.168.2.2

#
# Allow ping request to FW from ALL (commented out). Implements requirement nº4.
#
# EXPLICACIÓN: esta regla permite que el cortafuegos reciba peticiones ICMP de tipo ECHO REQUEST (--icmp-type 8). Para evitar ataques que utilizan el protocolo ICMP, sólo se habilitará esta opción con fines de diagnóstico bajo demanda.

##iptables -A INPUT -p icmp --icmp-type 8 -s 0/0 -m state --state NEW,ESTABLISHED,RELATED -j ##ACCEPT
```

```
#
# Allow ping responses to ALL from FW (commented out). Implements requirement nº4.
#
# EXPLICACIÓN: esta regla permite que el cortafuegos envíe respuestas ICMP de tipo ECHO REPLY (--icmp-type 0). Para evitar ataques que utilizan el protocolo ICMP, sólo se habilitará esta opción con fines de diagnóstico bajo demanda.

##iptables -A OUTPUT -p icmp --icmp-type 0 -d 0/0 -m state --state ESTABLISHED,RELATED -j ##ACCEPT

#
# Allow web answers to LAN from "Internet". Implements requirement nº5.
#

iptables -A FORWARD -i eth0 -p tcp -m state --state ESTABLISHED,RELATED -m multiport --source-port 80,443 --destination 192.168.3.0/24 -j ACCEPT

#
# Allow web requests to HTTP SERVER in DMZ from "INTERNET". Implements requirement nº6.
#
iptables -A FORWARD -i eth0 -p tcp --destination 192.168.2.2 -m multiport --destination-port 8080,8443 -j ACCEPT

#
# Allow dns requests to DNS SERVER in DMZ from "INTERNET".  Implements requirement nº7.
#
iptables -A FORWARD -i eth0 -p udp --destination 192.168.2.2 --destination-port 53 -j ACCEPT

#
# Allow dns answers to DNS SERVER in DMZ from EXTERNAL DNS SERVERS in "INTERNET". Implements requirement nº9.
#
iptables -A FORWARD -i eth0 -p udp -m state --state ESTABLISHED,RELATED --destination 192.168.2.2 --source-port 53 -j ACCEPT
```

```
#
# Allow web answers to ALL from WEB SERVER in DMZ. Implements requirement nº6.
#
iptables -A FORWARD -i eth1 -p tcp --source 192.168.2.2 -m state --state
ESTABLISHED,RELATED -m multiport --source-port 8080,8443 -j ACCEPT

#
# Allow web answers to LAN from WEBMIN in DMZ. Implements requirement nº8.
#
iptables -A FORWARD -i eth1 -p tcp --source 192.168.2.2 -m state --state
ESTABLISHED,RELATED -m multiport --source-port 10000 --destination 192.168.3.0/24 -j
ACCEPT

#
# Allow dns answers to ALL from DNS SERVER in DMZ. Implements requirement nº7.
#
iptables -A FORWARD -i eth1 -p udp --source 192.168.2.2 -m state --state
ESTABLISHED,RELATED --source-port 53 -j ACCEPT

#
# Allow dns requests to ALL from DNS SERVER in DMZ. Implements requirements nº9 and nº10.
#
iptables -A FORWARD -i eth1 -p udp --source 192.168.2.2 --destination-port 53 -j ACCEPT

#
# Allow web requests to WEB SERVERS in "INTERNET" from LAN. Implements requirement
nº5.
#
iptables -A FORWARD -i eth2 -p tcp --source 192.168.3.0/24 -m multiport --destination-port 80,443 -j
ACCEPT

#
# Allow web requests to WEB SERVER and WEBMIN in DMZ from LAN. Implements
requirement nº8.
#
iptables -A FORWARD -i eth2 -p tcp --source 192.168.3.0/24 -m multiport --destination-port
8080,8443,10000 --destination 192.168.2.2 -j ACCEPT
```

```
#
# Allow dns requests to DNS SERVER in DMZ from LAN. Implements requirement nº10.
#
iptables -A FORWARD -i eth2 -p udp --source 192.168.3.0/24 --destination 192.168.2.2 --destination-port 53 -j ACCEPT
```

La máquina con nombre *webdns.victim.com*, representa un servidor bastión (i.e. un servidor perteneciente a una organización y que es accesible desde Internet y que, normalmente, provee servicios) que presenta los servicios de contenido web (i.e. servidor web) y de resolución de nombres (i.e. servidor DNS). En relación al servidor web, se ha desplegado una aplicación Java que nos permitirá realizar evaluaciones de seguridad de aplicaciones web. Dicha aplicación es la aplicación *WebGoat*. Esta aplicación es una aplicación J2EE construida con vulnerabilidades de forma deliberada para impartir formación sobre seguridad en aplicaciones web. Por su parte, el servicio de resolución de nombres se ha implementado con el servidor de nombres de código abierto *BIND*. El servidor de nombres se ha configurado para que sea el servidor de nombres con autoridad sobre la zona *victim.com*.

Por último, se encuentran las máquina con nombres *wrkstn01.victim.com* y *wrkstn02.victim.com*. Estas máquinas modelan las estaciones de trabajo habituales de los usuarios finales ubicadas en la LAN interna de la organización. Es posible que se realicen algunas técnicas de ataque, o de otro tipo, desde estas máquinas, ya que, como demuestran las estadísticas, el 80% de los ataques telemáticos se llevan a cabo por personal interno de la organización.

En función del entorno en el que se encuentre el alumno, y tal y como se refleja en el diagrama anterior, las direcciones IP de la máquina *host* y la de la interfaz *eth0* del cortafuegos pueden variar. Esto hecho, en cualquier caso, quedará claramente reseñado en los vídeos que se grabarán por cada uno de los ejercicios prácticos incluidos en este proyecto.

1.4 Esbozo de la memoria

La presente memoria se estructura en los siguientes bloques:

- ✔ Punto 1: Introducción. Definición de objetivos, planificación, motivación, escenario de prueba, etc., del presente trabajo.
- ✔ Punto 2: Fundamentos de las evaluaciones de seguridad. En este punto se describirán los fundamentos teórico-técnicos en los que se basan las evaluaciones de seguridad en entornos TIC. Igualmente, se enumerarán y describirán la jerga y las definiciones utilizadas en las mismas.
- ✔ Punto 3: Metodologías para evaluaciones de seguridad. En este punto se describirán las metodologías estándares y estándares de facto que se utilizan en la actualidad para conducir las evaluaciones de seguridad.
- ✔ Punto 4: Reconocimiento. Como se detallará más adelante durante el desarrollo de esta memoria, una de las metodologías más usada a la hora de realizar evaluaciones de seguridad en entornos TIC es la metodología creada por el *EC Council*. En un primer nivel, está metodología se estructura en tres fases principales consecutivas en el tiempo: *preparación*, *evaluación* y *conclusión*. A su vez, la fase *evaluación* se estructura en cinco subfases consecutivas en el tiempo: *reconocimiento*, *escaneo y enumeración*, *obtención de acceso*, *manteniendo acceso* y *ocultación*. En este punto describiremos la subfase de *reconocimiento*, enumeraremos y describiremos las herramientas que se usan en la misma y haremos una demostración práctica de alguna de las herramientas más significativas.
- ✔ Punto 5: Escaneo y enumeración. En este punto describiremos la subfase de *escaneo y enumeración*, listaremos y describiremos las herramientas que se usan en la misma y haremos una demostración práctica de alguna de las herramientas más significativas.
- ✔ Punto 6: Obteniendo acceso. En este punto describiremos la subfase de *obteniendo acceso*, enumeraremos y describiremos las herramientas que se usan en la misma y haremos una demostración práctica de alguna de las herramientas más significativas.
- ✔ Punto 7: Conclusiones. En este punto listaremos aquellas conclusiones significativas que hayamos obtenido a lo largo de la ejecución del presente trabajo. Además, incluiremos las posibles líneas de ampliación en el estudio de las evaluaciones de seguridad en entornos TIC que se podrían seguir.

2 Fundamentos de evaluaciones de seguridad

2.1 Elementos básicos de la Seguridad de la Información

Los tres elementos básicos que conforman la base de la Seguridad de la Información son: confidencialidad, integridad, y disponibilidad. Es importante conocer qué significan cada uno de ellos y qué ataques se asocian con cada elemento.

- ✔ Confidencialidad: se refiere a las medidas necesarias para prevenir la revelación de información o datos a personas o sistemas no autorizados. La utilización de contraseñas es el medio más básico y común de asegurar la confidencialidad. Consecuentemente, los ataques contra las contraseñas son el tipo de ataque más habitual relacionado con la confidencialidad.
- ✔ Integridad: se refiere a los métodos y acciones llevadas a cabo para proteger la información de la alteración o revisión no autorizada. La integridad suele ser asegurada a través de un tipo de función llamada *hash*. Una función *hash* es un algoritmo matemático de un sólo sentido que aplicado a un argumento de entrada de longitud arbitraria, genera como salida un número de longitud fija. Con el cambio de un simple bit del argumento de entrada, la función *hash* genera un valor *hash* de salida diferente.
- ✔ Disponibilidad: se refiere a la capacidad de poder usar los sistemas y los datos por parte de los usuarios legítimos cuando éstos así lo necesitan. Los ataques relacionados con la disponibilidad se encuadran dentro de los ataques conocidos como de *Denegación de Servicio* (DOS - *Denial of Service attacks*). Los ataques DOS están diseñados para evitar que los usuarios legítimos puedan acceder a servicios o recursos alojados en sistemas.

Estos tres elementos forman lo que en el ámbito de la Seguridad de la Información se conoce como la triada de la seguridad o por el acrónimo CIA (*Confidentiality – Integrity – Availability*).

2.2 Hackers Éticos: definición y tipos

Un *Hacker Ético* es alguien que emplea las mismas herramientas y técnicas que usan los criminales, con el soporte y aprobación total por parte de un cliente, para ayudar a reforzar la seguridad de una red o de un sistema o sistemas. En contraposición, un *Cracker*, también conocido como un *Hacker malicioso*, usa las mismas capacidades, herramientas y técnicas para una ganancia personal o con propósitos destructivos. En términos más formales, un *Cracker* persigue un objetivo fuera del interés del propietario de la red o sistema.

Además de las definiciones dadas, la comunidad *hacking* se puede englobar en uno de los tres grupos siguientes:

- ✔ *Sombreros blancos* (*white hats*): considerados los "*chicos buenos*", se trata de los Hackers éticos. Aquellos contratados por un cliente para un objetivo específico de prueba y mejora de la seguridad.
- ✔ *Sombreros negros* (*black hats*): considerados los "*chicos malos*", se trata de los *Crakers*. Utilizan sus capacidades para una ganancia personal o con intenciones maliciosas.
- ✔ *Sombreros grises* (*gray hats*): el grupo más difícil de categorizar. Se trata de aquellos que tratan de demostrar, con o sin consentimiento, la existencia de debilidades o vulnerabilidades de seguridad en los sistemas o redes. En cualquier caso, realizar actividades de *hacking* sin permiso explícito es un delito.

2.3 Terminología Hacking y tipos de ataques

En primer lugar, se definirá qué es una Evaluación de Seguridad. Una *Evaluación de Seguridad* es cualquier prueba que es realizada para evaluar el nivel de seguridad de una red o un sistema. Las evaluaciones de seguridad pueden pertenecer a dos categorías: una *Auditoría de Seguridad* (también conocida como *Evaluación de Vulnerabilidades*) o un *Test de Penetración*.

Por su parte, una *Auditoría de Seguridad* escanea y prueba un sistema o red en busca de vulnerabilidades, pero no intenta intencionadamente explotar ninguna de ellas. Es decir, sólo señala las vulnerabilidades encontradas en beneficio de un cliente.

Por otro lado, una *Prueba de Penetración* (o *Prueba de Intrusión*) es una prueba claramente definida de los controles de seguridad de un sistema o red para identificar los riesgos y vulnerabilidades de seguridad, y se compone de tres fases principales:

- ✔ preparación: define el periodo de tiempo durante el cuál se desarrolla la prueba, el alcance de la misma, el tipo de ataque que están permitidos y las personas / perfiles que los llevarán a cabo.
- ✔ evaluación: es la fase donde se llevan a cabo los asaltos contra los controles de seguridad.
- ✔ conclusión: también conocida como post-evaluación, es la fase donde se preparan los informes finales para el cliente, detallando los resultados obtenidos y, habitualmente, las recomendaciones oportunas para mejorar la seguridad. En referencia a los informes finales entregables generados en esta fase, se citan a continuación algunos de los informes típicos que se suelen entregar como resultado de un *Test de Penetración*:
  - ✔ Un informe ejecutivo con la postura global de la organización con respecto a la seguridad.
  - ✔ Los nombres de todos los participantes y las fechas de todas las pruebas.
  - ✔ Un listado de todos los resultados, habitualmente clasificados por nivel de riesgo en orden descendente.
  - ✔ Un análisis de cada resultado y, si existen, pasos de resolución recomendados.
  - ✔ Ficheros de registro y cualquier otra prueba obtenida a partir del conjunto de herramientas utilizado.

Los *Test de Penetración* tratan de emular las actividades y metodologías empleadas por los criminales. Para ello, se clasifican en función del conocimiento inicial que se tiene del TOE (*Target of Evaluation*). Los elementos en los que se estructura la mencionada taxonomía se denominan *black box* (caja negra), *white box* (caja blanca) y *gray box* (caja gris):

- ✔ En las pruebas de tipo *black box*, el hacker ético no tiene conocimiento alguno sobre el TOE. Esta prueba está diseñada para simular un atacante externo y desconocido, es el tipo de prueba que más tiempo necesita y suele ser el tipo de prueba con un mayor coste económico.

- ✔ En la pruebas de tipo *white box*, el evaluador (i.e. hacker ético) tiene conocimiento completo de la red, sistema, y de la infraestructura que está evaluando. Este tipo de prueba se realiza de forma mucho más rápida, tiene un coste económico más bajo y trata de emular a una amenaza interna con un alto nivel de conocimientos sobre la infraestructura del objetivo.
- ✔ Las pruebas de tipo *gray box*, también conocidas como pruebas de conocimiento parcial, se diferencian de las pruebas de tipo *white box* en el nivel de privilegios del evaluador. En las pruebas de tipo *white box*, se asume que el nivel de privilegios del evaluador son los propios de un administrador de red. En las pruebas de tipo *gray box*, lo único que se asume es que el atacante es un interno. Este tipo de prueba está especialmente indicado para evaluar las posibilidades de conseguir una escalada de privilegios por parte de un usuario interno a la infraestructura.

En lo referente a los tipos de ataques que existen, se incluye a continuación la siguiente taxonomía de los mismos según el *EC Council*:

- ✔ Ataques a sistemas operativos: este tipo de ataques apuntan al error común que se suele cometer cuando se instala un sistema operativo, y se dejan en éste valores y configuraciones por defecto. También se incluyen en esta categoría, aquellos ataques que se aprovechan de las vulnerabilidades que puedan existir en un sistema operativo recién instalado y aún no parcheado.
- ✔ Ataques a nivel de aplicación: este tipo de ataques apuntan a las aplicaciones que se ejecutan por “encima” de los sistemas operativos. Muchas aplicaciones son desarrolladas sin seguir un ciclo seguro de desarrollo de software, lo que permite que sean liberadas con multitud de vulnerabilidades en ellas.
- ✔ Ataques a código “*empaquetado*”: este tipo de ataques apuntan al código estándar o estándar de facto que suele ir incorporado en muchas aplicaciones. Básicamente, se trata de atacar fragmentos de código que, por reutilización, suelen ir incorporados en muchas aplicaciones.
- ✔ Ataques a configuraciones erróneas: este tipo de ataque apuntan a sistemas que, por descuido o deliberadamente, no se encuentran configurados de forma segura. Este tipo de ataques toman ventaja de aquellos administradores que, simplemente, quieren hacer las cosas lo más fáciles posibles a los usuarios finales, generando en el empeño multitud de vulnerabilidades.

3 Metodologías para evaluaciones de seguridad

3.1 The Open Source Security Testing Methodology Manual (OSSTMM)

La OSSTMM es una metodología para la medida y la ejecución de pruebas / evaluaciones de seguridad. Los casos de prueba de la OSSTMM se dividen en canales que de forma colectiva prueban: controles sobre datos e información, niveles de concienciación del personal sobre la seguridad, controles contra la ingeniería social y el fraude, redes de telecomunicación y de ordenadores, dispositivos inalámbricos, dispositivos móviles, controles de acceso de seguridad física, procesos de seguridad, así como ubicaciones físicas como edificios, perímetros y bases militares.

La OSSTMM se focaliza en los detalles técnicos de exactamente qué elementos deben ser probados, qué hacer antes, durante y después de una prueba de seguridad; así como en cómo medir los resultados. Nuevas pruebas provenientes de mejores prácticas internacionales, leyes, regulaciones y asuntos éticos son regularmente añadidas y actualizadas.

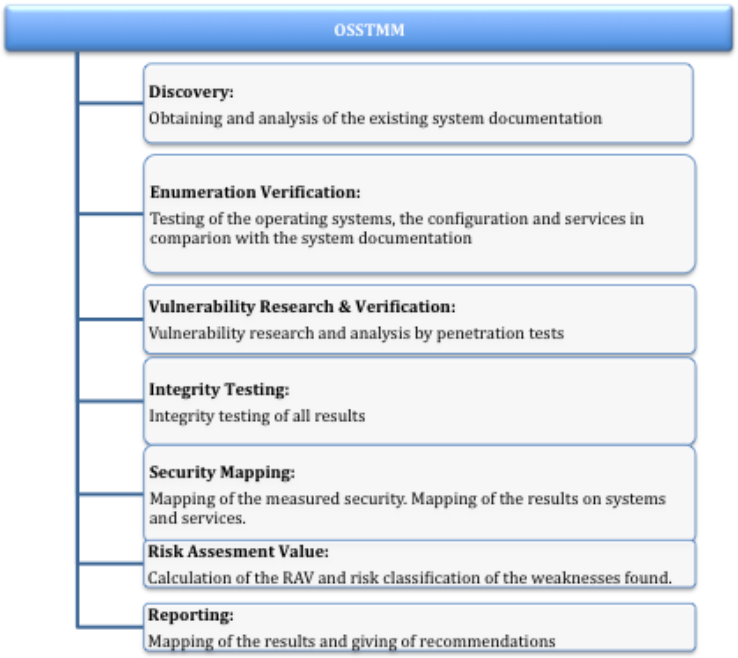


Figura 5: fases de la metodología OSSTMM.

3.2 The Information Systems Security Assesment Framework (ISSAF)

La ISSAF es un marco estructurado del *Open Information System Security Group* que categoriza las evaluaciones de seguridad de sistemas de información en varios dominios y detalla criterios de prueba específicos por cada uno de ellos. Este marco debe ser usado para cumplir con los requerimientos de evaluación de la seguridad de una organización, y puede opcionalmente ser usado como referencia para cumplir con otras necesidades dentro del campo de la Seguridad de la Información. Incluye la faceta crítica de los procesos de seguridad, su evaluación y reforzado para obtener una imagen global de las vulnerabilidades que pueden existir.

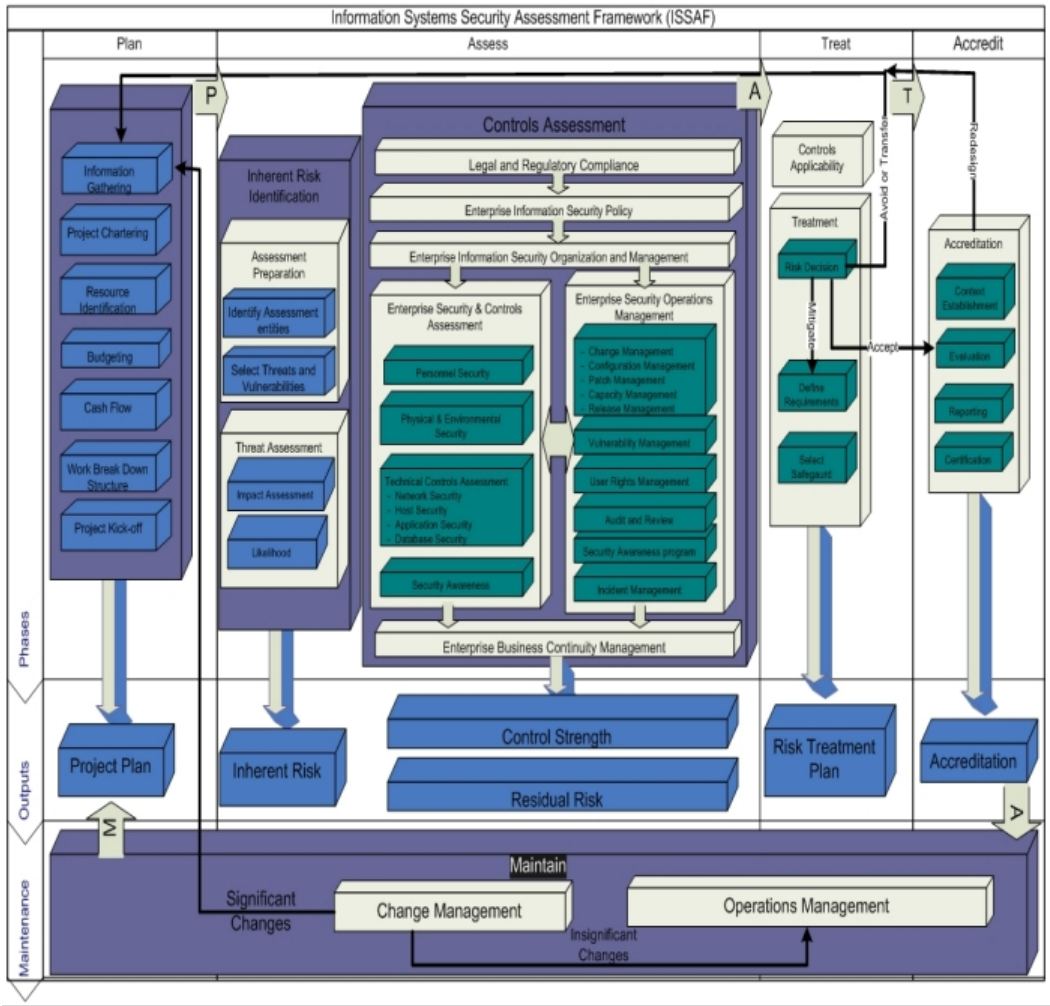


Figura 6: marco de trabajo ISSAF.

3.3 Metodología del EC Council

Como se comentó anteriormente en esta memoria, un *Test de Penetración* se estructura en tres fases principales: *preparación*, *evaluación* y *conclusión*. El *EC Council*, a su vez, definió cinco subfases dentro de la fase principal de *evaluación*: *reconocimiento*, *escaneo y enumeración*, *obtención de acceso*, *manteniendo acceso* y *ocultación*.

La subfase de *reconocimiento* no es nada más que los pasos necesarios para obtener pruebas e información de los objetivos a los que se desea atacar. Puede ser de naturaleza pasiva o activa. El reconocimiento pasivo conlleva obtener información sobre el objetivo sin interactuar directamente o indirectamente con él, mientras que el reconocimiento activo usa herramientas y técnicas que podrían ser o no descubiertas por el objetivo (ya que se produce una interacción con éste).

En la subfase de *escaneo y enumeración*, los evaluadores toman la información obtenida durante la fase de *reconocimiento* y activamente aplican herramientas y técnicas para obtener una información de mayor profundidad sobre los objetivos. Esto puede involucrar algo tan simple como ejecutar un barrido de pings (*ping sweep*) o un mapeador de red para ver qué sistemas están activos en la red; o tan complejo como ejecutar un escáner de vulnerabilidades para determinar qué vulnerabilidades existen en un sistema.

En la fase de *obtención de acceso* se realizan ataques reales sobre los sistemas previamente enumerados en la fase dos. Estos ataques pueden ser tan simples como acceder a un punto de acceso inalámbrico abierto y no asegurado, o tan complejos como el desarrollo y entrega de un *desbordamiento de buffer* (*buffer overflow*).

En la fase de *manteniendo acceso*, los hackers maliciosos tratan de asegurarse que disponen de una forma de volver a entrar en un sistema que ha sido exitosamente atacado. Por ejemplo, el atacante podría dejar puertas traseras (*backdoors*) para un uso futuro.

En la fase de *ocultación*, los hackers maliciosos tratan de ocultar su éxito y de evadir la detección por parte de los profesionales de la seguridad. Ejemplos de actividades que se llevan dentro de este apartado son la eliminación o alteración de ficheros de registro (*log files*), o la utilización de ficheros o directorios ocultos.

Figura 7: subfases hacking ético según el EC Council.

Esta memoria se centra en las tres primeras fases descritas, ya que son éstas las que se llevan a cabo en una Evaluación de Seguridad. Aunque también se mencionarán, de forma más resumida, técnicas y herramientas de las dos últimas fases.

4 Reconocimiento

4.1 Descripción general

Es habitual en las evaluaciones de seguridad en entornos TIC usar el termino *reconocimiento* y el término *footprinting* para hacer referencia a lo mismo. En otros ámbitos, sin embargo, se denomina al *footprinting* como una parte específica dentro de la fase de reconocimiento. En cualquier caso, *footprinting* engloba las técnicas a partir de las cuáles se puede obtener una mayor nivel de conocimiento sobre un objetivo. Existen dos tipos de *footprinting*:

- ✔ *Footprinting* activo: implica que el atacante o evaluador "toque" el dispositivo, sistema o red objetivos.
- ✔ *Footprinting* pasivo: puede ser llevado a cabo sin llegar a comunicarse con las máquinas o redes objetivos.

Por tanto, podríamos definir formalmente el *footprinting* como el proceso para obtener información sobre sistemas de computación y redes. Siendo éste el primer paso en la obtención de información y obteniendo como resultado un borrador a alto nivel del sistema o red objetivos. Se trata de obtener la mayor cantidad de información disponible — siendo ésta habitualmente fácil de obtener y de dominio público.

El proceso de *footprinting* se lleva a cabo siguiendo un flujo lógico: investigación de los recursos web del objetivo, determinación de los rangos de red, extracción de información desde bases de datos de tipo *Whois* y DNS y, finalizando, con ingeniería social, rastreo de correos electrónicos y *Google Hacking*.

Dentro de este ámbito, nos encontramos con el término Inteligencia Competitiva (*Competitive Intelligence*). La Inteligencia Competitiva se refiere a la información obtenida por una entidad de negocio sobre los clientes, productos y marketing de sus competidores. La mayor parte de esta información está públicamente disponible y es perfectamente legal visualizarla y obtenerla. Herramientas que permiten realizar Inteligencia Competitiva son: *Google Alerts*, *Yahoo Site Explorer*, *SEO for Firefox*, *SpyFu*, *Quarkbase* y el sitio web *DomainTools.com*.

El sistema DNS (*Domain Name System*) es un gran recurso a utilizar en el proceso de *footprinting*. El sistema DNS se compone de múltiples servidores a lo largo y ancho del mundo, con cada servidor almacenando y gestionando los recursos propios de su espacio de nombres. Las búsquedas DNS se realizan utilizando el protocolo de la capa de transporte UDP por el puerto bien conocido 53.

En un mismo espacio de nombres, es habitual que haya un servidor primario del espacio y uno o varios servidores secundarios. El servidor primario es el que tiene la copia maestra de los recursos de la zona, mientras que los servidores secundarios mantienen un respaldo de la copia maestra. Las copias respaldo de los servidores secundarios es actualizada a partir de la copia maestra del servidor primario a partir de un mecanismo llamado *Transferencia de Zona*. Este mecanismo utiliza el protocolo de transporte TCP por el puerto bien conocido 53.

Los diferentes registros del sistema DNS aportan información específica sobre diferentes tipos de recursos:

- ✔ SRV (*Service*): especifica el nombre del sistema y el número de puerto de los servidores que proveen servicios específicos, como por ejemplo un servidor de Servicios de Directorio.
- ✔ SOA (*Start of Authority*): identifica el nombre del servidor primario de la zona o espacio de nombres. Este registro contiene el nombre del sistema responsable de todos los registros dentro del espacio de nombres, así como características propias del dominio.
- ✔ PTR (*Pointer*): relaciona una dirección IP a un nombre de un servidor (permitiendo la ejecución de búsquedas DNS inversas).
- ✔ NS (*Name Server*): identifica los servidores de nombres dentro del dominio.
- ✔ MX (*Mail Exchange*): identifica los servidores de correo electrónico dentro del dominio.
- ✔ CNAME (*Canonical Name*): provee de alias dentro de la zona.
- ✔ A (*Address*): relaciona un nombre de un servidor a una dirección IP (permitiendo la ejecución de búsquedas DNS directas).

El registro SOA, por sí mismo, provee información sobre el nombre del servidor de nombres primario del dominio, la dirección de correo electrónico de la persona responsable de la zona, el número de serie (i.e. número de revisión del fichero de la zona), el tiempo de refresco (el número de segundos que un servidor de nombres secundario esperará antes de solicitar actualizarse), el tiempo de nuevo intento (el número de segundos que un servidor de nombres secundario esperará antes de volver a intentar un transferencia de zona fallida), el tiempo de expiración (el número máximo de segundos en los que la información almacenada de la zona se considera válida por parte de un servidor de nombres secundario) y, por último, el TTL (i.e. *Time to Live*), que indica el número de segundos máximo que por defecto los registros DNS pueden ser almacenados en memorias intermedias (i.e cachés) antes de que la fuente de la información necesite ser consultada de nuevo.

Como se ha comentado anteriormente, *Google Hacking* es otra técnica utilizada en esta fase de reconocimiento. *Google Hacking* se refiere a la manipulación de una cadena de búsqueda con operadores específicos para la búsqueda de vulnerabilidades. En este sentido, los operadores más buscados son:

- ✔ *Cache*: este operador indica que la búsqueda se realice sobre la copia almacenada por Google, permitiendo buscar en una versión anterior de un sitio web.
- ✔ *Filetype*: este operador permite realizar búsquedas de tipos específicos de archivos (.doc, .xls, etc.)
- ✔ *Intitle*: este operador permite buscar páginas que contienen una cadena específica en el título. Para buscar varias cadenas, se utiliza el operador *allintitle*.
- ✔ *Inurl*: este operador permite visualizar páginas con una cadena específica en la URL. Para buscar varias cadenas, se utiliza el operador *allinurl*.
- ✔ *Link*: este operador visualiza páginas enlazadas en base a un término de búsqueda.
- ✔ *Site*: este operador visualiza páginas de un sitio web determinado o dominio donde se almacena un término de búsqueda.

La ingeniería Social, el rastreo de correos electrónicos y la utilización de "arañas web" son también herramientas y técnicas de *footprinting*. La Ingeniería Social conlleva técnicas de hacking de bajo perfil técnico que se basan en la interacción con humanos para la consecución de información (*phising*, llamadas telefónicas, etc.) El rastreo de correos electrónicos se lleva a cabo a través de aplicaciones para obtener datos sobre la ubicación física dónde se dirigen dichos correos. Las "arañas web" por su parte son programas que recorren sitios web en busca de información y de los que, al menos teóricamente, se puede permanecer al margen creando un fichero con nombre "*robots.txt*" ubicado en el directorio raíz del sitio web.

4.2 Herramientas

Netcraft	
Subcategoría:	Herramientas para investigar sitios web.
Descripción:	Netcraft provee análisis de cuota de mercado de los servicios de hosting y de los servidores web, incluyendo detección de sistemas operativos y servidores web. Dependiendo del sistema operativo consultado, esta herramienta puede aportar información del tiempo que una máquina lleva disponible.
URL:	http://news.netcraft.com/

Webmaster Link Extractor	
Subcategoría:	Herramientas para investigar sitios web.
Descripción:	Esta herramienta nos permite obtener los enlaces existentes dentro de una página web perteneciente a un sitio web arbitrario.
URL:	http://www.webmaster-a.com/link-extractor-internal.php

Archive	
Subcategoría:	Herramientas para investigar sitios web.
Descripción:	Esta herramienta nos permite obtener versiones históricas de sitios web y de recursos multimedia (imágenes, audio, etc.)
URL:	http://archive.org/index.php

Nslookup	
Subcategoría:	Herramientas de DNS y Whois.
Descripción:	Se trata de una herramienta de línea de mandatos, incorporada en múltiples sistemas operativos, cuyo objetivo es poder consultar el Sistema de Nombres de Dominio (DNS). Permite obtener relaciones entre nombres de dominio y direcciones IP, así como otros registros DNS.
URL:	

DNSstuff	
Subcategoría:	Herramientas de DNS y Whois.
Descripción:	Conjunto de herramientas y utilidades que permiten obtener información DNS y Whois a través de Internet consultando para ello varias fuentes.
URL:	http://www.dnsstuff.com/

Wget	
Subcategoría:	Herramientas de copia de sitios web.
Descripción:	Se trata de una herramienta libre que permite la descarga de contenidos desde servidores web de una forma simple.
URL:	http://www.gnu.org/software/wget/

SpyPig	
Subcategoría:	Herramientas para verificar la lectura de correos electrónicos.
Descripción:	Se trata de una herramienta gratuita que permite saber si un correo electrónico ha sido leído por el destinatario.
URL:	http://www.spypig.com/

4.3 Demostración

A continuación se incluye un enlace a un vídeo donde se demuestra como se realiza la técnica de enumeración DNS a partir de las herramientas *Dnslookup* y *Dig*:

Enumeración DNS (Internet)

5 Escaneo y enumeración

5.1 Descripción general

El Escaneo y Enumeración son pasos importantes en la obtención de información para el hacker ético. El Escaneo es el proceso por el cuál se descubren qué sistemas hay en la red, qué puertos abiertos tienen y qué aplicaciones / servicios se están ejecutando. Las pasos de una metodología de escaneo genérica son: identificar los sistemas activos, descubrir los puertos abiertos, identificar los sistemas operativos, los servicios y la busca de vulnerabilidades.

Un *barrido de ping* es el método más fácil de identificar máquinas activas en la red. Un mensaje ICMP de Petición de Eco (tipo 8) es enviado a cada dirección de la subred. Aquellas máquinas que están activas (y que no filtran el protocolo ICMP) responderán con un mensaje ICMP de Repuesta Eco (tipo 0).

El escaneo de puertos es el método por el cuál se consulta a los sistemas de una subred para ver qué puertos se encuentran en estado de escucha (i.e. el puerto está abierto). Un número de puerto, en los protocolos de capa de trasporte TCP y UDP, identifica qué protocolo de la capa de aplicación debe recibir la información. Los números de puerto en origen son establecidos de forma dinámica usando cualquier número por encima de 1.023; mientras que el puerto en destino es habitualmente un puerto perteneciente al rango de puertos bien conocidos. Los números de puerto van desde el 0 hasta el 65.535 y se clasifican en tres grupos diferentes:

- ✔ Puertos bien conocidos: desde el 0 hasta el 1.023.
- ✔ Puertos registrados: desde el 1.024 hasta el 49.151.
- ✔ Puertos dinámicos: desde el 49.152 hasta el 65.535.

Algunos de los protocolos y sus respectivos puertos bien conocidos más importantes son:

- ✔ FTP. Puertos 20 y 21.
- ✔ Telnet. Puerto 23.

- ✔ SMTP. Puerto 25.
- ✔ DNS. Puerto 53.
- ✔ POP3. Puerto 110.
- ✔ NetBIOS. Puertos 137 y 139.
- ✔ SNMP. Puertos 161 y 162.

En la capa de transporte, la comunicación no orientada a comunicación es llevada a cabo con el protocolo UDP. Por su parte, TCP permite el establecimiento de comunicaciones orientadas a conexión. Para establecer esta conexión, el cliente, iniciando una apertura activa, enviará un segmento TCP con el indicador lógico SYN activado, indicando el deseo de sincronizar una sesión de comunicación. Este segmento también contiene un número de secuencia: un número aleatorio que ayuda a mantener la legitimidad y la unicidad de la sesión.

Cuando el servidor recibe este segmento, responde con otro segmento TCP con los indicadores lógicos SYN y ACK activados, y reconoce el número de secuencia del cliente incrementándolo en uno. Además, este segundo segmento lleva un número de secuencia generado por el servidor. Cuando este segmento es recibido por el cliente, éste genera un tercer segmento TCP para finalizar la sincronización. En este segmento, el indicador lógico ACK está activado y el número de secuencia del servidor es reconocido.

Al final de este "*apretón de manos*" en tres fases, existe un canal de comunicación (o circuito virtual establecido), hay números de secuencia establecidos en ambos extremos de la comunicación y la transferencia de datos puede comenzar.

Una de las herramientas más importantes para realizar escáneres es *nmap*, la cuál puede realizar desde la identificación de sistemas activos hasta el escaneo y enumeración de puertos. En *nmap* se puede configurar la velocidad a la que se quieren realizar las pruebas. En general, mientras la velocidad de escaneo sea menor, menor será la posibilidad de ser detectado. Existen multitud de opciones disponibles en *nmap*, los parámetros de tipo "*s*" permiten indicar el tipo de escaneo a realizar, los parámetros de tipo "*P*" configuran las opciones referentes a barridos de ping y los parámetros de tipo "*o*" tienen que ver con la salida producida por la herramienta. Los parámetros de tipo "*T*" tiene que ver con la velocidad y el sigilo, siendo las ejecuciones en serie las más sigilosas pero las que llevan más tiempo. Por su parte, los métodos paralelos son mucho más rápidos porque ejecutan varios escáneres de forma simultánea, pero son mucho más "ruidosos".

Los tipos de escáner más habituales con la herramienta *nmap* son:

Tipo Escaneo	Indicadores activados	Respuesta si puerto abierto	Respuesta si puerto cerrado	Observaciones
FULL (-sT)	SYN	SYN/ACK	RST	El más "ruidoso" pero el más fiable.
Half Open (-sS)	SYN	SYN/ACK	RST	No se finaliza el "*apretón de manos*" TCP. Sigiloso pero puede ser detectado por Sistemas de Detección de Intrusos (IDS).
XMAS (-sX)	FIN/URG/PSH	Sin respuesta	RST/ACK	No funciona con sistemas con Microsoft Windows como sistema operativo.
FIN (-sF)	FIN	Sin respuesta	RST/ACK	No funciona con sistemas con Microsoft Windows como sistema operativo.
NULL (-sN)	Sin indicadores activados	Sin respuesta	RST/ACK	No funciona con sistemas con Microsoft Windows como sistema operativo.
ACK (-sA)	ACK	RST	Sin respuesta	Usado para averiguar los filtros activos de un cortafuegos.

Las salida de la herramienta *nmap* puede ser personalizada. Por defecto, la salida se realiza en modo interactivo, lo que significa que la salida es enviada a la salida estándar (i.e. a la pantalla). También se puede indicar que la salida sea en un archivo XML (el cuál puede ser analizado sintácticamente por interfaces gráficas o importado a una bases de datos).

El proceso conocido como *War Dialing* es aquel proceso mediante el cuál un atacante llama a un grupo específico de números de teléfono con intención de encontrar un módem abierto. Los módem, habitualmente, están diseñados para responder llamadas y pueden aportar un acceso de *puerta trasera* a sistemas muy seguros contra ataques en otros vectores. Por su parte, el proceso conocido como *War Driving* se refiere al proceso de conducir con un coche buscando puntos de acceso inalámbricos abiertos.

A la hora de ocultar las actividades llevadas a cabo por un hacker, malicioso o no, de la vigilancia ejercida por parte de los profesionales de la Seguridad de la información, existen varias posibilidades: el uso de un *proxy*, el falseo de la dirección IP origen, el uso del encaminamiento de fuente (*source routing*) o en el uso de redes o sistemas que confieren anonimato (*anonymizers*).

Un servidor *proxy* no es más que un servidor configurado para actuar como intermediario entre el hacker y los objetivos de éste. El hacker envía mandatos y peticiones al *proxy* y éste, a su vez, las reenvía a los objetivos. Cualquiera que supervise la subred del objetivo verá que es el servidor *proxy* el que lanza los ataques y no el equipo del hacker.

En la falsificación de la dirección IP de origen (*spoofing source IP address*), el atacante utiliza alguna herramienta para la manipulación de paquetes para ocultar la verdadera dirección IP desde la que proceden los paquetes.

El encaminamiento de fuente (*source routing*) ofrece otra posibilidad de ocultar la identidad en una red. Fue originalmente diseñado para permitir a las aplicaciones especificar la ruta que un paquete debía coger para llegar a su destino, independientemente de lo que las tablas de encaminamiento entre los dos sistemas extremos indicaran. El atacante puede usar la dirección IP de otro sistema en la subred como dirección origen de los paquetes y, sin embargo, seguir obteniendo todo el tráfico de vuelta a través de él.

Finalmente, dentro del escaneo, está la opción de utilizar sistemas que confieren anonimato (*anonymizers*). Estos sistemas son servicios en Internet que hacen uso de un servidor web *proxy* para ocultar la identidad.

Seguidamente al escaneo, tenemos la Enumeración. La enumeración consiste en el listado de elementos dentro de un objetivo específico. En este caso, queremos encontrar directorios compartidos abiertos e información sobre cuentas de usuario fácil de obtener. En este momento, haremos una descripción más específica sobre el sistema operativo Microsoft Windows, dado que este sistema operativo es el que en la actualidad tiene una mayor cuota de mercado y es sobre el cuál se aplican la mayoría de las técnicas de enumeración.

Todo lo que se ejecuta en una máquina con sistema operativo Windows, lo hace dentro del contexto de una cuenta. Una cuenta puede pertenecer a un usuario, ejecutándose los procesos bajo esta cuenta en lo que se denomina *modo usuario*, o puede ser la cuenta del sistema, ejecutándose los procesos bajo esta cuenta en lo que se denomina *modo kernel*. Acciones y aplicaciones que se ejecutan en *modo usuario* son fáciles de detectar y contener. Aquellas que se ejecutan en *modo kernel*, por el contrario, pueden estar ocultas y ejecutarse con autoridad absoluta.

Los derechos de los usuarios son otorgados a través de la pertenencia de una cuenta a un grupo de usuarios y determina las tareas / acciones que dicha cuenta puede realizar. Los permisos se usan para determinar sobre qué recursos una cuenta tiene acceso. El método a través del cuál Windows mantiene registro de qué cuentas tienen qué derechos y permisos se basa en el uso de los *Secure Identifiers* (SID) y de los *Relative Identifiers* (*RID*). Un SID identifica de forma unívoca a un usuario o a un grupo de usuarios. Un RID describe la relación entre el usuario o grupo de usuarios identificado por el SID y la autoridad que lo creó. El RID forma parte del SID. Los SID se componen de una "*S*", seguidos de un número de revisión, un valor de autoridad, un indicador de ordenador local o dominio y el RID. Cuando el RID, parte final del SID, tiene un valor de 500 indica la cuenta de *administrador*. La siguiente cuenta en el sistema, *invitado* (g*uest)*, tiene el RID 501. A partir de ahí, todas los cuentas creadas en el sistema tendrán un valor RID consecutivo empezando a partir del valor 1.000.

Las cuentas se identifican de forma única por su SID, pero es evidente que las contraseñas deben estar almacenadas en algún sitio. En los sistemas Windows, las contraseñas se almacenan en el fichero *c:\WINDOWS\system32\config\SAM*. La base de datos SAM almacena las versiones cifradas de todas las contraseñas locales de la máquina. Para aquellas máquinas que forman parte de un dominio, las contraseñas son almacenadas y gestionas por el controlador de dominio.

Una sesión nula se produce cuando se inicia sesión en un sistema si ningún usuario o contraseña. En versiones antiguas de Windows se podía iniciar una sesión nula con el mandato:

```
net use \\maquina objetivo\IPC$ "" /u:""
```

Una sesión nula requiere que estén abiertos los puertos 135, 137, 139 y 145.

Por su parte, los sistemas Unix / Linux usan los atributos *User ID* (UID) y *Group ID* (GID) de una forma muy similar a como Windows utiliza los SID y RID.

Banner Grabbing es otra técnica de enumeración. Conlleva el envío de peticiones no solicitadas a un puerto abierto para ver qué error devuelve. Dependiendo de la versión de la aplicación que se esté ejecutando en el puerto, el mensaje de error devuelto puede indicar una vulnerabilidad potencial. Un método común de llevar a cabo *Banner Grabbing* es a través del uso de *Telnet* especificando un puerto específico. Otra herramienta con la que se puede llevar a cabo *Banner Grabbing* es *netcat*. *Netcat* es una herramienta de línea de mandatos que lee y escribe datos a través de conexiones de red usando TCP/IP. También permite crear túneles, escanear, etc.

Por último, la enumeración SNMP (*Simple Network Management Protocol*) puede ser muy poderosa. SNMP usa cadenas de comunidad a modo de contraseñas. La versión de sólo lectura de la cadena de comunidad permite que un solicitante pueda leer virtualmente cualquier cosa de un dispositivo gestionado. La versión de escritura y lectura es usada para controlar el acceso de peticiones SNMP *SET*, las cuáles pueden cambiar valores de un dispositivo gestionado. Los valores por defecto para ambas cadenas de comunidad son *public* para la versión de sólo lectura, y *private* para la versión de lectura-escritura. Suponiendo que un administrador dejó SNMP habilitado y que no cambió los valores por defecto, la enumeración con SNMP es muy sencilla. Ejemplos de herramientas para realizar enumeración SNMP son: *SNMPUtil* e *IP Network Browser*.

5.2 Herramientas

Angry IP Scanner	
Subcategoría:	Barridos de ping.
Descripción:	Se trata de una herramienta para hacer barridos de ping de código abierto, gratuito y multiplataforma que permite escanear direcciones IP y puertos, así como otras características.
URL:	http://www.angryip.org/w/Home

Nmap	
Subcategoría:	Herramientas de escaneo.
Descripción:	Se trata de una utilidad para realizar auditorías de seguridad y escanear una red.
URL:	http://nmap.org/

Hping	
Subcategoría:	Herramientas de escaneo.
Descripción:	Se trata de un escáner de red multiplataforma que permite llevar a cabo multitud de tipos de escáneres de red.
URL:	http://www.hping.org/

THC-Scan	
Subcategoría:	War Dialing.
Descripción:	Se trata de una herramienta multiplataforma que permite realizar llamadas a un rango de números telefónicos en busca de módem abiertos. Necesita que haya un módem conectado por el puerto serie.
URL:	http://www.thc.org/thc-scan/index.html

Telnet	
Subcategoría:	Banner Grabbing.
Descripción:	Se trata de un protocolo de red (implementado en un programa servidor y cliente) incorporado en la mayoría de los sistemas operativos. Puede ser utilizado para obtener información sobre sistemas en red y los servicios que se ejecutan sobre sus puertos abiertos.
URL:	

Netcat	
Subcategoría:	Banner Grabbing.
Descripción:	Se trata de una utilidad de red que lee y escribe datos a través de conexiones de red usando el protocolo TCP/IP. Entre otros usos, se puede utilizar para obtener información sobre sistemas en red y los servicios que se ejecutan sobre sus puertos abiertos.
URL:	http://netcat.sourceforge.net/

Nessus	
Subcategoría:	Escáner de vulnerabilidades.
Descripción:	Se trata de un escáner de vulnerabilidades propietario. Su objetivo es el de detectar potenciales vulnerabilidades sobre los sistemas sobre los que se aplica. Su uso en entornos no empresariales es gratuito.
URL:	http://www.tenable.com/products/nessus

SAINT	
Subcategoría:	Escáner de vulnerabilidades.
Descripción:	Se trata de un software usado para el escaneo de redes de ordenadores en busca de potenciales vulnerabilidades. Incorpora un modulo de explotación de las vulnerabilidades encontradas, con lo que también entra en el ámbito del test de penetración.
URL:	http://www.saintcorporation.com/

GFI LanGuard	
Subcategoría:	Escáner de vulnerabilidades.
Descripción:	Solución que permite escanear, detectar, evaluar y remediar cualquier vulnerabilidad de seguridad de red.
URL:	http://www.gfi.com/network-security-vulnerability-scanner/

Retina	
Subcategoría:	Escáner de vulnerabilidades.
Descripción:	Solución que encuentra vulnerabilidades, arregla rápidamente las más críticas y que puede proteger contra ataques futuros.
URL:	http://www.eeye.com/

MBSA	
Subcategoría:	Escáner de vulnerabilidades.
Descripción:	Es una herramienta gratuita de Microsoft para la evaluación de la seguridad y vulnerabilidades en entornos Microsoft.
URL:	http://technet.microsoft.com/es-es/security/cc184924.aspx

Nikto	
Subcategoría:	Escáner de vulnerabilidades.
Descripción:	Se trata de escáner de servidores web de código abierto que permite probar diferentes elementos en servidores web, como: ficheros / CGI peligrosos, versiones caducadas, problemas específicos de versiones, problemas de configuración, etc.
URL:	http://cirt.net/nikto2

Tor	
Subcategoría:	Ocultación de identidad.
Descripción:	Se trata de un sistema creado para otorgar anonimato. El programa cliente de esta solución, redirige el tráfico a través de una red mundial de servidores voluntarios, para ocultar la localización y las acciones del origen a cualquiera que esté espiando o haciendo análisis de tráfico en la red.
URL:	https://www.torproject.org/

ProxyChains	
Subcategoría:	Ocultación de identidad.
Descripción:	Se trata de una herramienta que te permite ejecutar cualquier aplicación a través de una lista personalizable de servidores proxys de tipo HTTP o SOCKS.
URL:	http://proxychains.sourceforge.net/

Anonymouse	
Subcategoría:	Ocultación de identidad.
Descripción:	Se trata de un servicio en línea y gratuito que permite navegar por Internet sin revelar ninguna información personal.
URL:	http://anonymouse.org/

P0f	
Subcategoría:	Enumeración.
Descripción:	Esta herramienta permite identificar los actores de cualquier conexión TCP/IP sin alterar la conexión de ninguna forma (i.e. de forma pasiva).
URL:	http://lcamtuf.coredump.cx/p0f3/

User2SID y SID2User	
Subcategoría:	Enumeración.
Descripción:	Estas herramientas permiten obtener un usuario a partir de su SID y viceversa en determinados sistemas operativos de la familia de Microsoft Windows. Son herramientas utilizadas en la enumeración de las cuentas de un sistema a partir de una sesión nula.
URL:	http://www.svrops.com/svrops/dwnldutil.htm

SNMP Scanner	
Subcategoría:	Enumeración.
Descripción:	Esta herramienta usa la base de datos SNMP MIB y las notificaciones asíncronas de SNMP para supervisar encaminadores en una red. Igualmente, esta herramienta compara cadenas que habitualmente son usadas como cadenas de comunidad SNMP, para comprobar la fortaleza de éstas.
URL:	http://www.secure-bytes.com/SNMP+Scanner.php

LDAP Admin	
Subcategoría:	Enumeración.
Descripción:	Esta herramienta gratuita para entornos Microsoft Windows nos permite visualizar, buscar, modificar, crear y borrar objetos en un servidor LDAP.
URL:	http://www.ldapadmin.org/

LDAP Explorer	
Subcategoría:	Enumeración.
Descripción:	Se trata de un navegador y cliente LDAP para la administración de directorios como OpenLDAP, Active Directory, eDirectory, iPlanet, DirX, etc.
URL:	http://www.ldapexplorer.com/

Ldp.exe	
Subcategoría:	Enumeración.
Descripción:	Este herramienta gráfica es un cliente LDAP que permite a los usuarios realizar operaciones como: conectar, enlazar, buscar, modificar, añadir y eliminar; sobre cualquier directorio compatible con LDAP. Como por ejemplo, el Directorio Activo de entornos Microsoft Windows.
URL:	http://technet.microsoft.com/en-us/library/cc772839(v=ws.10).aspx

5.3 Demostración

A continuación se incluyen enlaces a vídeos donde se demuestra como se realiza la técnica de escáner de puertos y escáner de vulnerabilidades a partir de las herramientas *nmap* y *nessus*:

Escáner de puertos (Internet)

Escáner de vulnerabilidades (Internet)

6 Obtención de acceso

6.1 Ataques a nivel de red: Sniffers y evasión

6.1.1 Descripción general

Los pasos básicos para la comunicación entre dos sistemas están definidos por el modelo de referencia OSI y la pila de protocolos TCP/IP. En cada capa se encuentran protocolos que ofrecen una función única dentro del proceso de comunicación. Los protocolos de aplicación suelen enviar la información en claro (i.e. no cifrada). Los protocolos de la capa de transporte gestionan la secuencia en la transmisión de los datos y el establecimiento de sesión de extremo a extremo.

La direcciones físicas (i.e. direcciones de capa 2 o direcciones MAC), se encuentran grabadas de forma física en la propia tarjeta de interfaz de red (*Network Interface Card* - NIC). Cuando se crea una trama en la máquina que va a enviar información, ésta utiliza un protocolo llamado ARP (*Address Resolution Protocol*) para obtener la dirección MAC de la máquina en la subred actual a la que debe ir dirigida la trama. Se mantiene en cada máquina una memoria cache con las resoluciones ARP previamente ejecutadas. El proceso de cambiar la memoria cache ARP de una máquina para inyectar entradas falsas se conoce con el nombre de envenenamiento ARP (*ARP Poisoning*). Dos herramientas que permiten llevar a cabo un envenenamiento ARP son *Cain and Abel* y *dsniff*.

Sniffing es el acto de capturar paquetes conforme se transmiten por cable o radio-ondas para la revisión de los mismos. La información capturada pueden ser direcciones, información sobre otros objetivos o, incluso, contraseñas enviadas sin cifrar. Un *sniffer* sitúa a una tarjeta de interfaz de red en modo promiscuo, lo que quiere decir que independientemente de la dirección de la trama, ésta es recogida por la NIC y procesada por el sistema. *Wireshark* es probablemente el *sniffer* (también conocido como analizador de protocolos) disponible más popular, ya que puede capturar paquetes transmitidos en medios guiados y no guiados con una interfaz de usuario fácil de usar.

Además del envenenamiento ARP (*ARP Poisoning*), existe otra técnica para explotar las limitaciones de algunos conmutadores (*switch*) conocida como inundación MAC (*MAC flooding*). Esta técnica consiste en inundar, de manera continuada en el tiempo, la tabla de direccionamiento del conmutador con más direcciones MAC de las que puede almacenar. De esta forma, un conmutador termina comportándose como un concentrador (*hub*), ya que al no saber qué dirección asignar a qué puerto, envía cada trama por todos los puertos. Sin embargo, la mayoría de los conmutadores actuales cuentan con mecanismos de protección contra esta técnica.

Los sistemas de detección de intrusos (*Intrusion Detection Systems*) son una combinación de hardware y software que examinan flujos de paquetes con la intención de encontrar comportamientos maliciosos o anormales. En función del método utilizado, tenemos dos tipos de IDS: los basados en firmas y los basados en anomalía. Los basados en firmas comparan los paquetes con una lista de patrones de tráfico conocidos por indicar la presencia de un ataque. Los basados en anomalía registran el comportamiento habitual del tráfico de la subred e inician una alerta cuando hay tráfico que se sale del comportamiento habitual (i.e. es anómalo). Un IDS basado en firmas es tan bueno como lo es su base de datos de firmas, si ésta no está actualizada puede que haya ataques que pasen inadvertidos. Un IDS basado en comportamiento puede detectar los últimos ataques (aquellos para los que incluso aún no hay firmas). Sin embargo, son más susceptibles a falsos positivos (i.e. indican que se está llevando a cabo un ataque cuando en realidad no es así). Por otro lado, un falso negativo se produce cuando no se ha alertado de un ataque cuando en realidad éste sí se ha producido.

Los IDS pueden ser basados en máquina (*Host* IDS – HIDS) o basados en red (*Network* IDS – NIDS). Los HIDS se instalan en un sistema específico y proveen de una protección individual. Los NIDS, por su parte, proveen de control sobre toda una red en función del "grifo de red" (*network tap*) donde se instalen. Algunos ejemplos de HIDS son *Cybersafe*, *Tripwire* o *Norton Internet Security*.

Snort es un NIDS de código abierto que combina los beneficios de la detección basada en firmas y la basada en anomalía. Se ha convertido en un estándar de facto compuesto por un poderoso *sniffer*, un registrador de tráfico y un analizador de protocolos que puede detectar ataques de desbordamientos de *buffer*, escaneo de puertos y prácticamente cualquier tipo de ataque o prueba que se pueda concebir.

Un cortafuegos (*firewall*) es un dispositivo ubicado en una red que está diseñado para proteger los recursos internos de ésta del acceso externo no autorizado. Los cortafuegos funcionan con un conjunto de reglas que indican expresamente qué puede pasar de un lado del cortafuegos al otro. Normalmente, la mayoría de los cortafuegos funcionan con un principio de denegación implícito (i.e. todo lo que no está aceptado explícitamente se deniega). Los cortafuegos pueden ser descubiertos a través de la utilidad *traceroute*, herramientas de escaneo de puertos y por otros medios.

Los cortafuegos, entre otras clasificaciones, se pueden clasificar en cortafuegos de filtrado de paquetes (*packet filter firewall*) o cortafuegos con estado (*stateful firewall*). Los cortafuegos de filtrado de paquetes inspeccionan las cabeceras de los paquetes que llegan a través de un puerto y toman la decisión de permitir su paso o no en función de unas reglas preestablecidas (también conocidas como *Access Control List* – ACL). Los cortafuegos con estado, por su parte, tienes los medios para rastrear el estado completo de una conexión.

Como técnicas de evasión de cortafuegos, tenemos los túneles ACK y los túneles HTTP, aunque los cortafuegos con estado pueden prevenir los túneles ACK. *Firewalking* es el proceso por el que se examina cada puerto de un cortafuegos para poder determinar vectores de ataque.

Finalmente, haremos mención los tarros de miel (*honeypots*). Los *honeypots* son sistemas configurados como señuelos para atraer a los atacantes. La localización de los *honeypots* es muy importante, dado que la máquina está pensada para ser atacada debe ser ubicada de forma aislada al resto de la red de producción.

6.1.2 Herramientas

ArpSpoof	
Subcategoría:	Envenenamiento ARP.
Descripción:	Este herramienta de tipo Unix / Linux, perteneciente al paquete Dsniff, permite realizar el envenenamiento de una cache ARP.
URL:	http://monkey.org/~dugsong/dsniff/

Cain & Abel	
Subcategoría:	Envenenamiento ARP.
Descripción:	Este herramienta de entornos Microsoft Windows, posee muchas capacidades y características. Una de las fundamentales, ya que se basan sobre ella otras capacidades, es la de realizar envenenamiento ARP.
URL:	http://www.oxid.it/cain.html

Wireshark	
Subcategoría:	Captura y análisis de paquetes.
Descripción:	Se trata de un analizador de protocolos que nos permite capturar y de forma interactiva ver el tráfico que pasa a través de una red de computadores.
URL:	http://www.wireshark.org/

Tripwire	
Subcategoría:	Sistemas de Detección de Intrusos basados en host.
Descripción:	La versión de código abierto de esta herramienta de seguridad e integridad de datos, permite supervisar y alertar sobre los cambios que se producen en objetos del sistema de ficheros de un conjunto de sistemas de tipo Unix / Linux.
URL:	http://sourceforge.net/projects/tripwire/

Snort	
Subcategoría:	Sistemas de Detección de Intrusos basados en red.
Descripción:	Se trata de un sistema de detección de intrusos basado en red de código abierto. Combina los beneficios de la inspección basado en anomalía, protocolo y firma.
URL:	http://www.snort.org/

6.1.3 Demostración

A continuación se incluyen enlaces a vídeos donde se demuestra como se realiza la técnica de *ARP Spoofing* o *ARP Poisoning* a partir de la herramienta *Arpspoof*:

ARP Spoofing / Poisoning (Internet)

6.2 Ataques a nivel de sistema

6.2.1 Descripción general

La autenticación de cualquier sistema se basa sobre uno o varios de los siguientes factores: algo que eres (i.e. biometría, como las huellas digitales), algo que tienes (como tarjetas) o algo que sabes (como las contraseñas). La fortaleza de una contraseña se determina por dos factores principales: longitud y complejidad. Los tipos de contraseñas se definen por lo que hay en ellas, pueden estar compuestas por letras (mayúsculas y minúsculas), números, caracteres especiales o alguna combinación de entre los cuatro grupos. De forma genérica, las contraseñas no deben contener ninguna parte del nombre del usuario, deben tener un mínimo de ocho caracteres y deben contener elementos de al menos tres de los cuatro componentes principales de complejidad: caracteres especiales, letras mayúsculas, letras minúsculas y números.

Existen cuatro tipos de ataques a contraseñas. Un ataque pasivo en línea conlleva la captura de paquetes en medios guiados o no guiados con la esperanza de capturar una contraseña en claro (i.e. sin cifrar). Un ataque activo en línea se da cuando el atacante simplemente trata de adivinar contraseñas. Este tipo de ataques lleva mucho más tiempo que los ataques pasivos y son mucho más fáciles de detectar. Estos ataques tratan de sacar partido de contraseñas débiles y malas prácticas de seguridad. Los ataques fuera de línea se dan cuando el atacante obtiene una copia del fichero de contraseñas y trata de “romperlo” (*password cracking*). Este tipo de ataques requieren habitualmente de acceso físico a la máquina, donde el atacante puede copiar el fichero de contraseñas a un medio móvil. El *craking* del fichero de contraseñas se puede realizar de tres formas: ataques de diccionario, ataques de fuerza bruta y ataques híbridos.

El ataque de diccionario es el más fácil y usa una lista de palabras en un fichero de texto, las cuales son cifradas siguiendo el mismo algoritmo / proceso al que fueron sometidas las contraseñas originales. Este proceso puede ser acelerado con el uso de las llamadas Tablas Arcoíris (*Rainbow Tables)*. Estas tablas contienen palabras de diccionario ya procesadas por un algoritmo de cifrado o *hash* concreto y que, por tanto, se pueden comparar directamente con las contraseñas cifradas originales ubicadas en el fichero de contraseñas. Un ataque híbrido se ubica un paso por encima del ataque de diccionario. En este tipo de ataque, la herramienta de *cracking* se alimenta de un fichero de diccionario pero además puede sustituir caracteres finales por números o símbolos. Este tipo de ataque también puede añadir números o símbolos, en vez de sustituir, al final de las palabras procedentes del diccionario. Los ataques de fuerza bruta prueban con todas las combinaciones de letras, números y caracteres especiales para intentar determinar las contraseñas. Este tipo de ataque conlleva mucho tiempo y tiene un consumo de proceso muy alto.

Keylogging es el acto de usar un dispositivo hardware o una aplicación software para capturar las pulsaciones que un usuario teclea. Con este método, las pulsaciones son capturadas conforme son tecleadas, independientemente del objeto para el cuál están siendo tecleadas. Los *keyloggers* pueden ser dispositivos hardware—pequeños dispositivos conectados entre el cable del teclado y el ordenador—o aplicaciones software instaladas y que se ejecutan en segundo plano, normalmente de forma oculta.

En los sistemas con sistema operativo Microsoft Windows, las contraseñas se almacenan después de haber sido procesadas por una función *hash*. En las primeras versiones de este sistema operativo, se utilizaba un esquema de codificación conocido por el nombre de *Lan Manager hashing* (LM). Este esquema se caracteriza porque convierte todos los caracteres de las contraseñas en mayúsculas y posteriormente añade espacios en blanco hasta completar una longitud de 14. Finalmente, separa la contraseña en dos grupos de 7 caracteres y aplica la función *hash* de forma independientemente a cada grupo. El valor del LM *hash* de 7 caracteres blancos es siempre el mismo (0xAAD3B435B51404EE). La autenticación LM fue usada en los sistemas operativos Windows 95/98. *New Technology Lan Manager* (NTLM) se usó en los sistemas con sistema operativo Windows NT SP3. NTLM v2 fue el siguiente esquema de autenticación. Finalmente, la autenticación con Kerberos fue incorporada con el sistema operativo Windows 2000.

La escalda de privilegios es el puente entre haber ganado acceso a un sistema y ser capaz de mantenerlo. La escalada de privilegios consiste en obtener los privilegios de administrador una vez que se ha obtenido acceso al sistema. Existen cuatro opciones principales para obtener los privilegios de administrador en un sistema. La primera opción consiste en “romper” (cracking) la contraseña de la cuenta de administrador. La segunda opción consiste en sacar partido de una vulnerabilidad presente en el sistema, en el sistema operativo o en una aplicación, que te permita obtener los permisos de un usuario privilegiado. La tercera opción consiste en utilizar alguna herramienta que pueda facilitar dichos privilegios. Por último, la cuarta opción es utilizar la Ingeniería Social (enviando, por ejemplo, un correo electrónico a un usuario con un ejecutable adjunto y pidiendo que lo ejecute).

Un *rootkit* es una colección de software instalada por el atacante que está diseñada para ocultar el compromiso del sistema. Los *rootkits* están diseñados para ofrecer puertas traseras a los atacantes para que éste las pueda utilizar en el futuro. Existen varios tipos de *rootkits* según al nivel donde se instalan: a nivel de aplicación (*user rootkit*), a nivel de kernel (*kernel rootkit*) y a nivel de librería de software (*library rootkit*).

Linux es un sistema operativo poderoso. El sistema de ficheros de Linux comienza por un directorio raíz como lo hace Windows. El directorio raíz de Windows suele ser *C:*, y el directorio raíz de Linux es "/". Al igual que Windows, Linux dispone de directorios con propósitos específicos. La seguridad de los ficheros y directorios es gestionada a través de las cuentas de usuario, la pertenencia de dichas cuentas a un grupo o varios grupos de usuarios y tres opciones de seguridad que se pueden asignar por usuario, por grupo de usuario y al resto, por cada recurso: leer, escribir y ejecutar. Estos permisos se suelen asignar a través del mandato *chmod* y el equivalente binario para cada grupo *rwx*: lectura es equivalente a 4, escritura es equivalente a 2 y ejecución es equivalente a 1. Por ejemplo, el siguiente mandato otorga todos los permisos (rwx) a todo el mundo (usuario, grupo y resto) para el fichero *file1*:

```
chmod 777 file1
```

Todos los usuarios y grupos están organizados a través de los identificadores únicos UID (*user ID*) y GID (*Group ID*). La información de ambos atributos puede ser localizada dentro del fichero */etc/paswword*. Las contraseñas en Linux se almacenan o en el fichero *passwd*, o en el fichero *shadow*, siendo éste último donde se encuentran las contraseñas cifradas.

Linux permite la adición de características o módulos al kernel (i.e. núcleo del sistema operativo) a través de Módulos Kernel Linux (LKM). LKM permite añadir funcionalidad al sistema operativo sin necesidad de que el kernel sea modificado y nuevamente compilado. El mandato para cargar un LKM es:

```
modprobe nombre_LKM
```

6.2.2 Herramientas

Cain & Abel	
Subcategoría:	Cracking de contraseñas.
Descripción:	Este herramienta de entornos Microsoft Windows posee muchas capacidades y características. Una de ellas es la posibilidad de romper contraseñas usando ataques de diccionario, ataques de fuerza bruta y ataques de criptoanálisis.
URL:	http://www.oxid.it/cain.html

John The Ripper	
Subcategoría:	Cracking de contraseñas.
Descripción:	Esta herramienta multiplataforma permite romper contraseñas de sistemas operativos Unix y contraseñas de esquemas *Lan Manager* de sistemas operativos Microsoft Windows.
URL:	http://www.openwall.com/john/

THC-Hydra	
Subcategoría:	Cracking de contraseñas.
Descripción:	Esta herramienta soportada en todas las plataformas basadas en UNIX permite romper contraseñas en aquellas pantallas donde se solicitan credenciales (*logon screens*). Soporta una gran variedad de servicios (FTP, Telnet, VNC, etc.)
URL:	http://www.thc.org/thc-hydra/

Ophcrack	
Subcategoría:	Cracking de contraseñas.
Descripción:	Esta herramienta multiplataforma permite romper contraseñas de sistemas operativos Windows a través del uso de tablas arcoíris.
URL:	http://ophcrack.sourceforge.net/

Brutus	
Subcategoría:	Cracking de contraseñas.
Descripción:	Esta herramienta soportada en sistemas operativos Microsoft Windows permite romper contraseñas en aquellas pantallas donde se solicitan credenciales (*logon screens*). Soporta una gran variedad de servicios (FTP, Telnet, SMB, etc.)
URL:	http://www.hoobie.net/brutus/

Actual Keylogger	
Subcategoría:	Keyloggers y captura de pantallas.
Descripción:	Esta herramienta gratuita permite capturar las pulsaciones de tecleado. Se ejecuta en segundo plano de forma oculta y dispone de muchas opciones de personalización.
URL:	http://www.actualkeylogger.com/

Actual Spy	
Subcategoría:	Keyloggers y captura de pantallas.
Descripción:	Esta herramienta, versión de pago de la anterior, permite capturar las pulsaciones de tecleado. Se ejecuta en segundo plano de forma oculta y dispone de muchas opciones de personalización. Además, incluye mucho más funciones que su versión gratuita, como por ejemplo la toma periódica de capturas de pantalla y el envío de éstas a través de correo electrónico.
URL:	http://www.actualspy.com/

USB Grabber	
Subcategoría:	Keyloggers y captura de pantallas.
Descripción:	Esta herramienta permite configurar una memoria de tipo USB para que, una vez conectada a un sistema arbitrario, copie un conjunto de ficheros predefinidos a la memoria sin notificación e indicio alguno.
URL:	http://digitaldream.persiangig.com/

6.2.3 Demostración

A continuación se incluyen enlaces a vídeos donde se demuestra como se realiza la técnica de *Password Craking* a partir de la herramienta *John the Ripper* junto con las herramientas *Metasploit* y *Fgdump*:

Password Cracking (Internet)

6.3 Ataques de bajo perfil técnico: Ingeniería Social

6.3.1 Descripción general

La Ingeniería Social es el arte de manipular a una persona, o a un grupo de personas, para que den información o realicen un servicio que en condiciones normales nunca hubiesen realizado. Los ingenieros sociales se aprovechan del deseo natural de las personas de ayudar a otros, a obedecer a una autoridad o a confiar en determinadas entidades. La Ingeniería Social es un método no técnico de atacar sistemas. Todos los ataques de Ingeniería Social se pueden clasificar en: basados en humanos o basados en ordenadores. La Ingeniería Social basada en humanos usa la interacción entre personas para obtener información útil.

Dumpster Diving (“*buceo en la basura*”) es un ataque donde el atacante busca entre la basura información útil. Buscando entre los contenedores de basura, las papeleras de reciclaje o las papeleras de oficina se puede encontrar una gran cantidad de información, como contraseñas (que han sido escritas para ser fáciles de recordar), diagramas de red, directorio de teléfonos de empleados, etc.

La suplantación es un ataque donde el ingeniero social pretende ser un empleado, un usuario válido o, incluso, un directivo. Tanto si se falsifica una tarjeta de identificación, como si se trata simplemente de convencer a los demás de que se ostenta una determinada posición en la compañía, un atacante podría conseguir acceso a zonas restringidas, obteniendo de esta forma más oportunidades de realizar ataques. Fingiendo ser una persona de mayor autoridad, el atacante podría incluso utilizar la intimidación sobre empleados de menor nivel para conseguir que le ayuden a obtener acceso a un sistema.

Un ataque de soporte técnico es una forma de suplantación orientada al personal de dicho soporte. Un atacante podría realizar una llamada haciéndose pasar por un usuario y solicitando el reinicio de una contraseña. La persona del servicio de atención, creyendo que está ayudando a un cliente o usuario final, involuntariamente reinicia la contraseña y la da a conocer al atacante, otorgándole un acceso fácil al sistema.

Shoulder Surfing (“*navegar por encima del hombro*”) es un ataque básico donde el atacante simplemente mira por encima del hombro de un usuario autorizado. Suponiendo presencia física, se puede observar como los usuarios inician sesión, acceden a información sensible o muestran pasos críticos en el proceso de autenticación.

Tailgating (seguir de muy cerca) y *Piggybacking* (ir colgado de alguien, colarse detrás de) son dos tipos de ataques muy relacionados. El *Tailgating* ocurre cuando un atacante tiene un identificador falso y simplemente sigue a un usuario autorizado a través de una puerta de seguridad. *Piggybacking* es un poco diferente en el sentido que el atacante no tiene identificador alguno y simplemente solicita a un usuario autorizado que le deje pasar. El atacante puede decir que se ha dejado el identificador en la mesa o que se lo olvidó en casa. En cualquier caso, un usuario autorizado mantiene la puerta abierta a pesar del hecho de que el atacante no tienen un identificador visible.

En la Ingeniería Social Inversa, el atacante se postula como alguien con autoridad o con capacidad técnica y creará un escenario donde la victima crea que debe llamarle para recibir ayuda. Hay una secuencia específica de pasos en este tipo de ataques: anuncio, sabotaje y soporte. En primer lugar, el atacante se anuncia o publicita su posición como soporte técnico en algún área concreta. En segundo lugar, el atacante lleva a cabo algún tipo de sabotaje. Por ejemplo, un ataque de denegación de servicio. Entonces, la víctima acude al atacante en busca de ayuda. Finalmente, el atacante, en su ánimo de “ayudar” a la víctima, solicita credenciales o cualquier otra información sensible para ganar acceso al sistema.

La Ingeniería Social basada en ordenadores o sistemas ocurre cuando el ataque es llevado a cabo con la participación de un ordenador o algún dispositivo de proceso de datos. El método más común de Ingeniería Social basada en ordenadores es el *phishing* (“*ir de pesca*”). Un ataque de *phishing* conlleva manipular un correo electrónico para que parezca legítimo, pero que de hecho contiene enlaces a sitios web falsos o es capaz de descargar código malicioso. Los enlaces contenidos dentro del correo electrónico podrían conducir al usuario a un formulario web falso, en el cuál la información introducida es obtenida por el atacante para su uso posterior.

Otra forma de Ingeniería Social basada en ordenadores es el uso de canales de *chat* o mensajería instantánea. Los atacantes no sólo usan los canales de *chat* para encontrar información personal a usar en futuros ataques, sino que también hacen uso de los mismos para dispersar *malware* e instalar programas.

De hecho, IRC (*Internet Relay Chat*) es el protocolo usado principalmente para el control de los *zombies* (ordenadores infectados por código malicioso y que forman parte de una red) por parte de los atacantes y, normalmente, creadores de este tipo de redes.

Creando varias capas de defensa—incluyendo procedimientos en la gestión del cambio y medios de autenticación fuerte—es un buen comienzo en la mitigación de ataques de Ingeniaría Social. En cualquier caso, la única defensa efectiva contra la Ingeniería Social es la educación continua del usuario. Formar a los usuarios—especialmente a aquellos en posiciones relacionadas con el soporte técnico—en como reconocer y prevenir ataques de Ingeniería Social es la mejor contramedida disponible.

6.4 Ataques a servidores y aplicaciones web

6.4.1 Descripción general

Un servidor web se comporta como cualquier otro servidor. En este caso, responde a peticiones TCP en el puerto 80 (HTTP) o en el puerto 443 (HTTPS). El servidor espera una petición HTTP GET desde el cliente y responde con un código HTML específico que representa una página del sitio web. El servidor busca en un área de almacenamiento, encuentra el código que corresponde a la petición y se la entrega al cliente.

Los servidores web Apache conforman la mayoría de los servidores web desplegados en Internet. Apache es un servidor web de código abierto, muy poderoso y rápido que habitualmente se ejecuta sobre plataformas Unix o Linux. Por otro lado, tenemos el servidor web Microsoft IIS (*Internet Information Services*). Este servidor es una opción basada en sistemas operativos Microsoft y es fácil de gestionar.

La interfaz CGI (*Common Gateway Interface*) ofrece un método estándar a los servidores web para pasar una petición web de un usuario a un programa de aplicación, y recibir de éste datos dinámicos de vuelta para enviárselos al cliente (i.e. navegador). Dado que los programas CGI pueden ejecutar mandatos arbitrarios en el sistema con los permisos del usuario bajo el cuál se ejecutan, pueden ser muy peligrosos si no están correctamente controlados.

Las SSI (*Server Side Includes*) son directivas incluidas en páginas HTML que son evaluadas en el servidor mientras las páginas están siendo servidas por éste. Las SSI permiten añadir contenido dinámico a una página HTML existente, sin necesidad de servir la página entera desde un programa CGI. Las SSI son útiles para añadir fragmentos de código común a un sitio web determinado, como pueden ser las cabeceras de las páginas, los píes o un menú de navegación.

El ataque de recorrido de directorio (*Directory Traversal Attack*) es un tipo de ataque web mediante el cuál, un atacante manipula una URL (*Uniform Resource Locator*) para navegar la estructura de directorios del propio servidor web. Esto puede derivar en el acceso a ficheros o, incluso, a la obtención de una terminal de mandatos. También es conocido como el ataque *punto-punto-slash* (../)., escalada de directorio (*Directory Climbing*) o marcha atrás (*Backtracking*).

Básicamente, consiste en que los atacantes envían peticiones HTTP solicitando al servidor volver al directorio raíz y, a partir de ahí, obtener acceso a otros directorios. Un ejemplo de la mencionada petición sería:

http://servidor/../../../../directorio_elegido/comand.exe

La intención de este ataque es alcanzar el directorio raíz y ejecutar una terminal de mandatos desde un directorio arbitrario. La versión Unicode de este ataque incluiría los siguientes caracteres: *%2e%2e%2f*.

La manipulación de parámetros de una URL (*Parameter Tampering* o *URL Tampering*), se da cuando el atacante manipula los parámetros ubicados en la cadena URL con la intención de modificar datos como: permisos, precios, cantidades de productos o credenciales.

Una aplicación web ocupa el “espacio” que hay entre el frontal web y la componente que almacena los resultados de procesar las peticiones de los usuarios (normalmente, un sistema de gestión de base de datos relacional –SGBDR). Las aplicaciones web suelen ser atacadas aprovechando vulnerabilidades existentes desde el comienzo de su concepción. Los desarrolladores podrían pasar por alto vulnerabilidades conocidas, olvidar parchear importantes fallos de seguridad, dejar contraseñas por defecto, etc.

Un ataque que suele tener mucho éxito consiste en introducir mandatos maliciosos en los campos de entrada de una aplicación web. El objetivo es pasar código de explotación al servidor aprovechando la pobre verificación de la entrada por parte de la aplicación web. Esto se puede llevar a cabo a través de diferentes métodos, como por ejemplo: la inyección de fichero (donde el atacante inyecta un puntero en un campo de entrada de un formulario web a un código de explotación almacenado en un fichero en un sitio remoto); la inyección de mandatos (cuando el atacante inyecta mandatos en los campos de entrada de un formulario web en vez de la entrada esperada), etc.

La inyección SQL es, de lejos, el ataque de inyección más común y exitoso. SQL (*Structured Query Language*) es un lenguaje diseñado para manejar datos almacenados en un SGBDR. La base de datos relacional es, en sí misma, una colección de relaciones (que almacenan tuplas compuestas a su vez de columnas) relacionadas a partir de columnas clave y que pueden ser consultadas y actualizadas. Cada relación tiene un nombre que se referencia cada vez que se quiere consultar o actualizar la relación. Se utiliza SQL cuando se quiere añadir, suprimir, mover, actualizar o ver los datos almacenados en las columnas de las relaciones.

Las consultas SQL empiezan generalmente con el mandato SELECT. SELECT permite seleccionar los datos sobre los que se quiere realizar alguna acción. Además de SELECT, hay otras opciones y mandatos de interés para un atacante. Como por ejemplo: *DROP TABLE relation_name* (suprimir una relación del SGBDR), así como los mandatos INSERT y UPDATE que permiten insertar y actualizar datos respectivamente en una relación.

La inyección SQL se da cuando el atacante inyecta consultas SQL directamente en el campo de entrada de un formulario web o en un parámetro de una URL de petición. Este ataque permite que el mandato SQL circunvale el cometido del frontal web y sea ejecutado directamente por parte del SGBDR. Para averiguar si una aplicación web es vulnerable a este ataque, bastaría con encontrar un formulario web de un sitio e, en vez de introducir lo que se pide o espera, introducir una comilla simple (') y ver qué clase de error se obtiene. Si no funciona, se puede probar introduciendo la cadena:

cualquier_texto' or 1=1--

Nombres habituales de ataques de inyección SQL son: *SQL UNION query attack*, tautología, *blind SQL injection* y *error-based SQL injection*.

Los ataques de *Cross Site Scripting* (XSS) se aprovechan del contenido dinámico de algunos sitios. XSS se da cuando los atacantes, utilizando algún lenguaje de programación interpretado (como Javascript), inyectan código malicioso para que sea ejecutado en el cliente (i.e. en el navegador). Con este ataque se pueden realizar ataques de denegación de servicio, robar las *coockies* del navegador, mostrar mensajes en ventanas emergentes al usuario, etc.

Un ataque de desbordamiento de buffer (*buffer overflow*) intenta escribir más datos de los esperados por el área de almacenamiento en memoria asignado. El objetivo puede ser sobrescribir áreas de almacenamiento adyacentes para ejecutar código arbitrario, inhabilitar una aplicación o todo un sistema. El resultado final puede ser la inhabilitación de un sistema o la alteración de los punteros de control de la aplicación permitiendo la ejecución de un código ejecutable arbitrario. Dentro de este tipo de ataques tenemos las variedades: desbordamiento basado en pila (*stack-based overflow*), desbordamiento basado en montículo (*heap-based overflow*), *tobogán de NOPs* (*NOP slide*).

Para protegerse ante este tipo de ataques, además de utilizar buenas prácticas en la codificación, los desarrolladores hacen uso de *palabras canario*. Las palabras canario son valores conocidos que se sitúan entre el área de memoria que va a recibir los datos (i.e. el buffer) y los datos de control. Si se produce un desbordamiento de buffer, la palabra canario se verá afectada y se disparará una señal para parar el sistema se pare. Herramientas como *StackGuard* hacen uso de este sistema de protección.

Una *coockie* es un fichero de texto muy pequeño, que es almacenado en el sistema cliente para ser usado por parte del servidor web la próxima vez que éste sea visitado. Las *coockies* pueden contener cualquier tipo de información, incluyendo detalles de autenticación, preferencias, contenido de carros de compra virtuales, detalles de sesión, etc. Las *coockies* pueden ser muy interesantes para un atacante, ya que le podrían permitir, entre otras cosas, manipular parámetros, cambiar precios e, incluso, autenticarse ante un servidor web suplantando la identidad de un usuario legítimo.

Para proteger un servidor web valen las mismas buenas prácticas empleadas para fortificar cualquier servidor: obtener las últimas versiones, parchear los fallos de seguridad reportados, asegurar que los permisos establecidos son correctos, establecer configuraciones correctas, eliminar o cambiar valores por defecto con implicaciones de seguridad, vigilar y supervisar los ficheros de registro, etc. Un escáner de seguridad proveerá la información necesaria para fortificar un servidor web. En este sentido, existe el escáner de vulnerabilidades específico para servidores web *Nikto*.

6.4.2 Herramientas

Httprecon	
Subcategoría:	Ataques Web.
Descripción:	Esta herramienta permite identificar (fingerprint) servidores web. Su objetivo es la identificación exacta de una implementación httpd. Para ello, se basa en la utilización de banner-grabbing, enumeración de códigos de estado, análisis del orden de las cabeceras http, etc.
URL:	http://www.computec.ch/projekte/httprecon/

cURL	
Subcategoría:	Ataques Web.
Descripción:	Esta herramienta de línea de mandatos permite transferir datos a partir de una sintaxis URL que soporta una gran variedad de protocolos (FTP, FTPS, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, etc.)
URL:	http://curl.haxx.se/

CookieDigger	
Subcategoría:	Ataques Web.
Descripción:	Esta herramienta permite identificar la generación débil de cookies y la implementación insegura de gestión de sesiones por parte de aplicaciones web. La herramienta informa de cómo de predecible y cuánta entropía tiene una cookie. Así mismo, informa si hay información crítica contenida como valor en la cookie.
URL:	http://www.mcafee.com/es/downloads/free-tools/cookiedigger.aspx

WebScarab	
Subcategoría:	Ataques Web.
Descripción:	Esta herramienta multiplataforma analiza aplicaciones que se comunican a través de los protocolos HTTP y HTTPS. En su modo de operación habitual, se utiliza como proxy interceptador entre un navegador y un servidor para revisar y modificar peticiones y respuestas.
URL:	https://www.owasp.org/index.php/Category:OWASP_WebScarab_Project

WebBrute	
Subcategoría:	Ataques Web.
Descripción:	Esta herramienta permite escanear los directorios web que están protegidos con autenticación HTTP, probando la fortaleza de las contraseñas de los usuarios.
URL:	http://www.rawlogic.com/products.html

BSQL Hacker	
Subcategoría:	Ataques Web.
Descripción:	Esta herramienta es un sistema diseñado para explotar vulnerabilidades de inyección SQL sobre prácticamente cualquier base de datos.
URL:	http://labs.portcullis.co.uk/application/bsql-hacker/

Havij	
Subcategoría:	Ataques Web.
Descripción:	Esta herramienta es un sistema automatizado de inyección SQL que ayuda a los hackers éticos a encontrar y explotar vulnerabilidades de inyección SQL en una página web.
URL:	http://www.itsecteam.com/products/havij-v116-advanced-sql-injection/index.html

6.4.3 Demostración

A continuación se incluyen enlaces a un vídeo donde se demuestra como se realizan las técnicas de *Cross Site Scripting* (XSS), *Cross Site Request Forgery* (CSRF), *SQL Injection y Parameter Tampering* a partir de la herramienta *WebGoat* junto con las herramienta *Paros Proxy*:

Ataques a aplicaciones web (Internet)

6.5 Ataques a redes inalámbricas

6.5.1 Descripción general

Existe un conjunto de estándares recogidos bajo la denominación 802.11 que determinan las características y bases que rigen las redes inalámbricas. 802.11a puede obtener velocidades de hasta 54 Mbps y usa la banda frecuencial de los 5 GHz. 802.11b es capaz de operar a velocidades de 11 Mbps y opera en la banda frecuencial de los 2,4 GHz. 802.11g alcanza velocidades de hasta 54 Mbps y opera en la banda frecuencial de los 2,4 GHz. Por último, incluimos el estándar 802.11n, que puede alcanzar velocidades de hasta 100 Mbps y opera tanto en la banda frecuencial de los 2,4 GHz como en la banda frecuencia de los 5 GHz.

Referiremos igualmente a los estándares 802.11i y 802.16. 802.11i es una corrección a la serie 802.11 original y especifica mecanismos de seguridad en el uso de redes inalámbricas. 802.16 se creó para el desarrollo global de redes inalámbricas de ámbito metropolitano. Conocida como *Wimax*, es capaz de ofrecer velocidades de hasta 40 Mbps aunque se pretende alcanzar velocidades de Gigabit.

Existen dos modos principales en los que una red inalámbrica puede operar. El primero se conoce como modo *ad hoc*, donde un sistema se conecta a otro sistema de forma directa, como si hubiera un cable directo entre ellos. En el modo infraestructura se hace uso de un punto de acceso (*Access Point—AP*) para canalizar todas las conexiones inalámbricas a través de él. El punto de acceso se configura para conectarse mediante un enlace al mundo exterior, normalmente a través de un encaminador. Al AP los equipos clientes se asocian y se autentican. Los clientes se conectan al AP a través del uso de NICs inalámbricas, siempre y cuando los clientes estén dentro del alcance del AP y están configurados con la información necesaria para poder conectarse. Las redes inalámbricas pueden estar formadas por un único AP o por varios, creando en este último caso celdas que se solapan y que permiten a los clientes vagar (*roam*) libremente sin perder conectividad. El cliente necesita asociarse con un AP en primer lugar y, posteriormente, desvincularse cuando entra dentro del alcance de otro AP.

Cuando hablamos de un único AP, su área de cobertura se llama BSA (*Basic Service Area*). La comunicación entre este AP y sus clientes se conoce con las siglas BSS (*Basic Set Service*). Si se amplía el área de cobertura de una red inalámbrica con múltiples APs, esta configuración se conoce con las siglas ESS (*Extended Service Set*). Conforme un cliente se mueve de un AP a otro, y mientras todo esté configurado correctamente, se desvinculará de uno y se asociará a otro de manera transparente.

El movimiento entre múltiples APs dentro de un ESS determinado es lo que se conoce como *Roaming*.

Dentro del diseño de la seguridad de las redes inalámbricas es necesario no sólo determinar qué tipo de antena se selecciona, sino que también la ubicación de la misma. La instalación física de las antenas es un asunto importante, porque se querrá evitar la dispersión de la señal y la pérdida de potencia. La mayoría de los APs estándares usan antenas omnidireccionales, lo que significa que la señal que emana desde la antena es de igual intensidad alrededor de los 360 grados de la fuente. Las antenas direccionales, por su parte, permiten dirigir la señal en una dirección determinada, lo que incrementa significativamente la fuerza de la señal y la distancia de cobertura. Otras antenas que pueden ser usadas son antenas dipolo y antenas parabólicas de rejilla. Las antenas dipolo tienen dos torres de señal que funcionan en modo omnidireccional. Las antenas parabólicas de rejilla funcionan de forma similar a las antenas de plato de satélite y tienen un alcance muy significativo (hasta 16 Km).

Para identificar una red inlámbrica a sus potenciales clientes, se debe asignar un SSID (*Service Set ID*). El SSID no es una contraseña y no provee de ninguna seguridad a la red inalámbrica. Se trata de una palabra de texto cuyo único objetivo es distinguir unas redes de otras. Los SSIDs son emitidos por defecto y son fácilmente obtenibles incluso cuando la emisión por defecto es deshabilitada. El SSID es parte de la cabecera de cada paquete, por lo que averiguarlo es relativamente fácil para casi cualquier atacante y, por tanto, su ocultación no tiene mucho sentido.

La autenticación en redes inalámbricas puede darse de diferentes formas, desde muy simples hasta muy complejas. Un cliente puede enviar una trama de autenticación 802.11 a un AP con el SSID apropiado y obtener una respuesta en forma de trama de verificación. Por otro lado, podría tratarse de un esquema de autenticación basado en un escenario reto/respuesta. O, incluso, podría habilitarse el concurso de un servidor de autenticación; por ejemplo, un servidor *Radius*. Es importante resaltar la diferencia entre asociación y autenticación. Asociación es el acto donde un cliente se conecta a un AP, mientras que autenticación implica la verificación de la autenticidad de la identidad del cliente por parte de la AP antes que el cliente puede acceder a la red.

El protocolo WEP (*Wired Equivalent Privacy*) es un protocolo de seguridad muy débil utilizado en redes inalámbricas. Usando claves desde 40 bits hasta 232 bits con el algoritmo de cifrado en flujo RC4. La debilidad de WEP se origina en el hecho de la reutilización de Vectores de Inicialización (*Initialition Vectors*—IV). Un atacante podría obtener suficientes paquetes como para llegar a obtener la clave WEP compartida entre un AP y sus clientes.

WEP no fue diseñado correctamente para proveer de seguridad ya que puede dar el mismo nivel de seguridad que se puede esperar en una red cableada con un único dominio de colisión (i.e. una red establecida con el uso de un concentrador). Los Vectores de Inicialización WEP son relativamente pequeños y, consiguientemente, suelen ser reutilizados con frecuencia. Además, los IVs se envían en texto claro como parte de la cabecera. Un atacante simplemente necesita generar un número suficiente de paquetes como para poder capturar IVs y poder derivar la clave. Esto permite a un atacante obtener la clave utilizada sobre la marcha, en tiempo real y convierte el cifrado en inútil. Los atacantes pueden generar tráfico en una red inalámbrica enviando mensajes de desvinculación. Este tipo de mensajes no necesitan estar autenticados, por tanto el bombardeo de este tipo de mensajes es suficiente para llevar a cabo este tipo de ataques.

Una opción más recomendable como tecnología de cifrado en redes inalámbricas es WPA (*Wi-Fi Protected Access*) o WPA-2. WPA utiliza el protocolo TKIP (*Temporary Key Integrity Protocol*), una clave de 128 bits y la dirección MAC del cliente para conseguir un cifrado mucho más fuerte. En resumen, WPA cambia la clave cada 10.000 paquetes, en vez de mantener la misma clave y reutilizarla. Además, las claves son transferida de una lado a otro durante una sesión EAP (*Extensible Authentication Protocol*). Dicha sesión EAP se establece mediante un "*apretón de manos*" (*handshake*) de cuatro pasos en el que se produce una autenticación mutua entre el cliente y el AP.

WPA-2 es muy parecido a WPA con la salvedad que se concibió para las organizaciones empresariales y gubernamentales en mente. Se puede incluir EAP o *Radius* en lo que la autenticación WPA-2 se refiere, lo que permite la utilización de tickets *Kerberos*. Además, WPA-2 utiliza el algoritmo AES para el cifrado, asegurando el cumplimiento con el estándar FIPS 140-2.

El sitio web http://wigle.net ayuda a la hora de localizar las ubicaciones geográficas de redes inalámbricas. Equipos de personas han relacionado las localizaciones de redes inalámbricas usando GPS y una herramienta llamada *NetStumbler*. *NetStumbler* puede ser usado para identificar áreas con cobertura pobre dentro de un ESS, detectar posibles causas de interferencia y encontrar puntos de acceso falsos (*Rogue Access Points*). Esta herramienta está basada en el sistema operativo Microsoft Windows, es fácil de usar y es compatible con 802.11a, 802.11b y 802.11g.

Kismet es otra herramienta para el descubrimiento de redes inalámbricas. Funciona sobre sistemas operativos Linux y, a diferencia de *NetStumbler*, funciona de forma pasiva. Es decir, es capaz de detectar APs y clientes sin necesidad de enviar paquete alguno. Puede detectar APs con configuraciones por defecto, así como determinar qué sistema de cifrado utilizan. Utiliza la técnica de "saltado" de canales para descubrir el mayor número posible de redes inalámbricas, tiene la capacidad de capturar paquetes y de almacenarlos en ficheros de registro cuyo formato es legible por *Wireshark* y *TcpDump*.

Otra herramienta para el descubrimiento de redes inalámbricas es *NetSurveyor*. Esta herramienta gratuita y basada en sistemas operativos Windows, tiene capacidades similares a las aportadas por *NetStumbler* y *Kismet*. Además, tiene soporte para la gran mayoría de adaptadores de red inalámbricos. Es una gran herramienta para la resolución de problemas y la verificación de redes inalámbricas.

El punto de acceso falso (*Rogue Access Point*) es un ataque relativamente fácil de llevar a cabo sobre redes inalámbricas. En este ataque, un atacante configura un AP cerca de otros APs legítimos y trata de engañar a usuarios para que éstos se asocien con el AP falso. A veces referido como el ataque del "*gemelo malo*", es un tipo de ataque relativamente fácil de acometer. La única desventaja es que es un ataque fácil de detectar y se corre un riego alto de ser localizado.

Un ataque de conexión *ad hoc* se aprovecha del modo de conexión del mismo nombre de las redes inalámbricas. Este ataque se da cuando un atacante simplemente activa un ordenador portátil, anuncia una red inalámbrica en modo *ad hoc* desde dicho portátil y espera a que potenciales víctimas se conecten a él.

Ataques de denegación de servicio (DOS) son también ataques relativamente fáciles de llevar a cabo. Pueden hacerse de dos formas, no siendo ninguna de ellas especialmente difícil. En primer lugar, se puede usar un buen número de herramientas para manipular y enviar paquetes de desvinculación a los clientes de un AP, lo que forzará a que éstos finalicen la conexión. Por supuesto, los clientes pueden tratar de volver a establecer la conexión, pero es simplemente cuestión de seguir enviando paquetes de desvinculación por parte del atacante. La otra forma de llevar ataques tipo DOS en redes inalámbricas es la utilización de algún tipo de dispositivo que realice la inhibición de frecuencias junto con una antena / amplificador de alta ganancia. Todos los dispositivos inalámbricos son susceptibles a algún tipo de interferencia.

Una defensa que algunos administradores de redes inalámbricas usan es el establecimiento de filtros de direcciones MAC. Básicamente, se trata de una lista de direcciones MAC que están permitidas a asociarse a un AP; si la dirección MAC de tu NIC no se encuentra en la lista no puede asociarse al AP. La forma fácil de circunvalar este filtro es supervisar el tráfico de la red para averiguar que direcciones MAC están en uso y simplemente suplantar una de ellas.

La "rotura" del protocolo WEP es muy fácil de llevar a cabo y existen varias herramientas que permiten su ejecución. La idea consiste básicamente en generar suficientes paquetes como para averiguar la clave de cifrado. La manera de atacar al protocolo WEP suele ser acometida con los siguientes pasos:

- ✔ Arrancar un adaptador inalámbrico compatible y asegurar que puede capturar e inyectar paquetes.
- ✔ Empezar a capturar paquetes.
- ✔ Usar algún método para generar miles y miles de paquetes (habitualmente a través del envío de mensajes de desvinculación al AP)
- ✔ Analizar los paquetes capturados (tanto en tiempo real o fuera de línea) con una herramienta de tipo *cracking*.

Dos de las herramientas que permiten llevar a cabo la obtención de la clave en WEP son *Cain and Able* y *Aircrack.*

Bluetooth es una tecnología inalámbrica para el intercambio de datos en un rango de distancia relativamente corto (10 metros o menos). Fue diseñado originalmente como un medio de reducir el cableado, pero se ha convertido en omnipresente en un amplio espectro de dispositivos móviles.

Los dispositivos *Bluetooth* tiene dos modos: un modo de descubrimiento y un modo de emparejamiento. El modo de descubrimiento determina cómo reacciona el dispositivo a consultas de otros dispositivos que quieren conectarse; tiene tres posibles opciones. La opción *detectable* permite que el dispositivo responda a todas las consultas; *detectable limitado* restringe la opción de ser detectado y la opción *no detectable* indica que se ignoren todas las consultas de descubrimiento.

El modo de emparejamiento detalla cómo el dispositivo reaccionará cuando otro dispositivo *Bluetooth* solicite conectarse con él. Existen básicamente dos opciones: sí, se establecerá la conexión o no, no se establecerá. La opción de no emparejamiento rechazará todas las solicitudes de conexión y la de emparejamiento justo lo contrario.

Los ataques *Bluetooth*, que sacan partida de su facilidad de uso, se clasifican en cuatro catagorías. *Bluesmacking* es simplemente la ejecución de ataques de denegación de servicio contra un dispositivo a través de Bluetooth. *Bluejacking* consiste en el envío de mensajes no solicitados desde y hacia dispositivos móviles. *Bluesniffing* que consiste en la captura de datos transmitidos por *Bluetooth*. Finalmente, tenemos los ataques de tipo *Bluesnarfing*, que consiste en el robo de datos de un dispositivo móvil a través de *Bluetooth*. Algunas de las herramientas que permiten llevar este tipo de ataques son: *BlueScanner*, *BT Browser*, *Bluesniff* y *btCrawler*.

6.5.2 Herramientas

NetStumbler	
Subcategoría:	Redes Inalámbricas.
Descripción:	Esta herramienta para sistemas operativos Microsoft Windows permite, entre otras, las siguientes funciones: verificar que una WLAN está bien configurada, estudiar la cobertura de la señal, detectar otras redes que pueden causar interferencia, detectar AP no autorizados (*Rogue AP*), etc.
URL:	http://www.netstumbler.com/downloads/

Kismet	
Subcategoría:	Redes Inalámbricas.
Descripción:	Esta herramienta es un detector de redes inalámbricas, un sniffer y un sistema de detección de intrusos. Puede identificar redes inalámbricas de forma pasiva, incluso redes ocultas que no envían paquetes de tipo baliza.
URL:	http://www.kismetwireless.net/

Aircrack	
Subcategoría:	Redes Inalámbricas.
Descripción:	Esta herramienta permite romper contraseñas de redes 802.11 WEP y WPA-PSK una vez que se han capturado suficientes paquetes.
URL:	http://www.aircrack-ng.org/

BlueScanner	
Subcategoría:	Redes Inalámbricas.
Descripción:	Esta herramienta es un script en bash que implementa un escáner para detectar los dispositivos Bluetooth que se encuentran dentro del radio de alcance de nuestro sistema. Trabaja de forma no intrusiva, es decir, sin llegar a establecer una conexión con los dispositivos encontrados y sin ser detectado.
URL:	http://bluescanner.sourceforge.net/

BlueDiving	
Subcategoría:	Redes Inalámbricas.
Descripción:	Esta herramienta es una suite de pruebas de penetración a través del uso del protocolo Bluetooth. Implementa ataques como Bluebug, Bluesnarf, etc.
URL:	http://bluediving.sourceforge.net/

Super Bluetooth Hack	
Subcategoría:	Redes Inalámbricas.
Descripción:	Esta herramienta permite controlar y leer información desde un teléfono remoto a través de Bluetooth o Infrarojos. La lista de contactos y los SMS pueden ser almacenados en formato HTML.
URL:	http://mobile.brothersoft.com/super-bluetooth-hack-135.html

6.5.3 Demostración

A continuación se incluyen enlaces a un vídeo donde se demuestra cómo se realiza la técnica de ruptura de clave en una red inalámbrica con esquema de seguridad WPA2 Preshared key, con un ataque de diccionario y el concurso de las herramientas *Airmon-ng, Airodump-ng y Aircrack-ng*:

Ataques a redes inalámbricas (Internet)

6.6 Ataques a través de Malware

6.6.1 Descripción general

El malware es generalmente definido como el software diseñado para dañar o secretamente acceder a un ordenador sin el consentimiento del propietario de éste. Se considera un software como malware basándonos en la intención percibida del creador, en vez de por las características que el software pueda presentar.

Un Troyano es un software que aparentemente realiza una función deseada por parte del usuario pero que además realiza otras funciones, sin el conocimiento del usuario, como el robo de información, el daño del sistema o datos, etc. Los troyanos se entregan al objetivo con una apariencia inocente, invitando a que sean abiertos. Una vez abiertos, el troyano se instala para robar información específica, para actuar como un *keylogger*, para aportar acceso circunvalando los métodos de acceso normales (i.e. *Backdoor access*), etc. Los canales abiertos (*Overt Channles*) son canales de comunicación legítimos usados por programas a través de un sistema o red, mientras que los canales ocultos (*Covert Channels*) son canales para transmitir información de formas no previstas. La mayoría del malware es descargado de Internet, obtenido a través de canales IRC (*Internet Relay Chat*) o a través de adjuntos de correo electrónico.

Para hacer que un troyano parezca inocente o legítimo, se hacen uso de los encapsuladores (*wrappers*). Los encapsuladores son programas que permiten enlazar un programa ejecutable a elegir (el troyano), con otro fichero de aspecto inocente que una víctima no tendrá sospecha en abrir. Por ejemplo, la aplicación *EliteWrap* nos permitirá enlazar un ejecutable de tipo *backdoor* a un ejecutable de un juego.

Existen diferentes categorías de troyanos. Una de ellas son los troyanos de línea de mandatos. El objetivo de este tipo de troyanos es el de proveer de una puerta trasera (*backdoor*) al sistema a través de una línea de mandatos remota. Un ejemplo de troyano de esta categoría es *Netcat*. Netcat ofrece todo tipo de control a través de una consola remota en el objetivo. Cuando se instala y se ejecuta en un sistema remoto, abre un puerto a elegir en modo de escucha. Ejecutando el mandato:

```
nc -l -p 5555
```

Se abre el puerto 5555 en modo de escucha. A partir de ahí, podemos conectarnos al sistema con una conexión parecida a Telnet con el siguiente mandato:

nc ip_address -p 5555

Netcat puede ser utilizado para conexiones de salida y de entrada, sobre TCP o UDP, desde o hacia cualquier puerto del sistema.

Algunos de los puertos comunes utilizados por troyanos significativos son:

Nombre de Troyano	Puerto Utilizado
TCPWrappers	421
Doom	666
Snipernet	667
Tini	7777
WinHole	1080-1081
RAT	1095-1097-1098
SpySender	1807
Deep Throat	2140-3150
NetBus	12345-12346
Whack a Mole	12362-12363
Back Orifice	31337-31338

Existen varios programas que permiten ver qué puertos están abiertos en un sistema. Una opción es el mandato existente en la mayoría de sistemas operativos *Netstat*. Introduciendo el mandato:

netstat -an

Nos mostrará todas las conexiones hechas y puertos en escucha de nuestro sistema en formato numérico.

Existen también herramientas de escaneo de puertos que facilitan esta labor, algunos ejemplos de las mismas son: *Fport*, *TCPView*, *IceSword*.

La prevención ante troyanos requiere mantener la vigilancia sobre el registro (en sistemas operativos Microsoft Windows), los controladores de dispositivos (i.e. drivers), los servicios activos así como los programas configurados para arrancar con el sistema. En el último caso, Microsoft Windows ejecutará en el arranque del sistema todo aquello que se encuentre especificado en las ramas: *Run*, *RunServices*, *RunOnce* y *RunServicesOnce* del registro.

Un virus, por su parte, se define como un programa con capacidad de reproducirse haciendo copias de sí mismo anexándose dentro de otros ejecutables o ficheros. Los virus se clasifican en los siguientes tipos:

- ✔ Virus de sector de arranque: este tipo de virus mueve el sector de arranque a otra ubicación del disco duro, forzando a que el código del virus se ejecute antes que el propio sistema operativo. Son muy difíciles de eliminar, ya que requieren la regeneración del sector de arranque con herramientas como *fdisk* o *mbr*.
- ✔ Virus de infección de ficheros: es el tipo más común de virus. Infectan un fichero huésped y a partir de ahí realizan las acciones programadas cuando son ejecutados. Estos virus pueden reemplazar completamente al huésped, sólo partes de él, o no reemplazar nada del huésped pero recrearlo de nuevo para que, cuando se ejecute, sea el virus el que tome el control.
- ✔ Virus multiparte: intenta tanto infectar otros archivos como el sector de arranque. Son habitualmente referidos como virus con múltiples vectores de infección.
- ✔ Virus macro: este tipo de virus infecta ficheros de tipo plantilla de productos ofimáticos. El virus *Melissa*, por ejemplo, pertenece a esta categoría.
- ✔ Virus de código polimórfico: estos virus cambian sus código usando un motor polimórfico incorporado. Son virus difíciles de detectar ya que cambian su firma frecuentemente.
- ✔ Virus metamórficos: estos virus se reescriben completamente cada vez que infectan un nuevo archivo.

Un gusano (*worm*), por su parte, es un programa que hace copias de sí mismo a través de una red sin la intervención humana. Habitualmente, no altera o infecta archivos, sino que reside en memoria y se duplica a sí mismo consumiendo recursos y causando estragos. El uso más común de los gusanos hoy en día es la creación de *bot-nets*.

Para protegerse contra el malware en general, es necesario disponer de un buen programa antivirus o antimalware. La clave para el buen funcionamiento de este tipo de programas es mantener las base de datos de firmas actualizada. Una aplicación antimalware es tan buena como actualizada esté su base de datos de firmas. Otra opción es la de utilizar un sistema "*desinfectante*". Este tipo de sistemas están configurados para probar en ellos controladores de dispositivos, medios físicos y cualquier tipo de fichero antes de ser introducidos en la red real de producción. Consecuentemente, este tipo de sistemas sólo se usan para este menester y están aislados del resto de sistemas. Este tipo de sistemas están configurados con un par de programas de detección de malware, monitores de puertos, monitores del registro (en MS Windows), verificadores de integridad de archivos, etc.

Un ataque de denegación de servicio (DOS) es visto, en muchos casos, como un ataque de último recurso: si no se ha podido obtener acceso, se va a inhabilitar el sistema o red. El ataque estándar DOS busca inhabilitar un sistema o red de sistemas, o inhabilitar el acceso por parte de los usuarios legítimos. Un ataque de denegación de servicio distribuido (DDOS) no proviene de un único equipo, sino que de varios.

Una *bot-net* es una red de ordenadores *zombies* a partir de la cuál un atacante podría lanzar ataques distribuidos. En este tipo de redes, los sistemas permanecen latentes a la espera de ser llamados para la acción. Dicha acción puede conllevar el envío de pings o cualquier otra acción conveniente para el ataque en curso. Normalmente, para el canal de control de los sistemas que forman una *bot-net* se utiliza el protocolo IRC (*Internet Relay Chat*) o el protocolo ICQ (i.e. mensajería instantánea).

Dentro de los ataques DOS y DDOS tenemos los siguientes ejemplos:

- ✔ Ataque SYN: el atacante enviará miles y miles de segmentos TCP con el indicador lógico SYN activado y, probablemente, con una dirección IP origen falsa. La máquina víctima responderá con segmentos SYN / ACK. En un momento dado, se pueden llegar a comprometer todos los recursos de la máquina víctima ocasionando el ataque DOS.
- ✔ Inundación SYN: en este ataque, el atacante envía miles de segmentos TCP con el indicador lógico SYN activado, pero nunca responde a ninguno de los segmentos TCP SYN / ACK recibidos de vuelta. Dado que el sistema objetivo debe esperar un cierto tiempo a que llegue una respuesta, puede darse el caso de que, eventualmente, el sistema objetivo se quede sin la capacidad de establecer nuevas ocasiones lo que deriva en un ataque DOS.
- ✔ Inundación ICMP: el atacante envía paquetes ICMP de tipo *Echo Request* a la víctima con una dirección IP origen falsa. La víctima responde a dichos paquetes y puede alcanzar el límite de paquetes enviados por segundo.

- ✔ A nivel de aplicación: el atacante envía más tráfico legítimo de lo que la aplicación tiene capacidad para procesar.
- ✔ *Smurf*: el atacante envía un alto número de pings a la dirección de difusión de la subred, con la dirección IP origen suplantada con la dirección IP de la victima. De esta forma, todas las respuestas a los pings se encaminarán hacia la víctima, agotando los recursos y ocasionando la consecución del ataque DOS. Existe un ataque similar, pero utilizando el protocolo de transporte UDP, conocido con el nombre *Fraggle*.
- ✔ Ping de la muerte: en este ataque un atacante envía mensajes ICMP fragmentados a la víctima. Cuando los fragmentos son ensamblados, el paquete ICMP resultante es de mayor tamaño que el máximo permitido. Esto ocasiona la inhabilitación del sistema y la consecución del ataque DOS.

Las contramedidas que su pueden implementar contra los ataques DOS incluyen las inhabilitación de los servicios innecesarios, el uso de una una buena política de cortafuegos y la actualización constante de parches y actualizaciones de software. Adicionalmente, el uso de un buen NIDS (*Network Intrussion Detection System*) puede ayudar contra ataques DOS a través de la red.

A diferencia de los ataques DOS, los intentos de secuestros de sesión (*session hijacking*) no tratan de “romper” nada sino que tratan de sacar partido de una conexión ya establecida y autenticada. *Session Hijacking* se refiere al intento activo de robar una sesión entera a un cliente: el servidor ni siquiera se entera de qué ha pasado y el cliente simplemente se conecta de nuevo e inicia una nueva sesión. Para llevar a cabo un secuestro de sesión, se siguen habitualmente los siguientes pasos:

- ✔ Se captura el tráfico entre el cliente y el servidor.
- ✔ Se controla el tráfico y se predice el siguiente número de secuencia de los segmentos TCP.
- ✔ Se cancela la sesión del cliente, con un ataque DOS por ejemplo.
- ✔ Se toma control de la sesión por parte del atacante utilizando el número de secuencia obtenido en el paso 2.
- ✔ Se inyectan paquetes al servidor objetivo por parte del atacante.

Las contramedidas contra el secuestro de sesión son el uso de números de secuencia impredecibles, el uso de cifrado para la protección del canal, limitar el número de conexiones entrantes y minimizar el acceso remoto. Igualmente, la formación del usuario final puede ser de ayuda. Sesiones que desaparecen de forma repentina pueden indicar un problema de red, pero también el secuestro de una sesión.

6.6.2 Herramientas

EliteWrap	
Subcategoría:	Encapsuladores.
Descripción:	Esta herramienta permite encapsular varios ejecutables en uno para las plataformas Microsoft Windows 95/98/2000/NT. Su uso habitual es para secretamente instalar y ejecutar programas.
URL:	http://homepage.ntlworld.com/chawmp/elitewrap/

Netcat	
Subcategoría:	Consolas línea de mandatos.
Descripción:	Se trata de una utilidad de red que lee y escribe datos a través de conexiones de red usando el protocolo TCP/IP. Entre otros usos, se puede utilizar para obtener consolas de líneas de mandatos directas o inversas.
URL:	http://netcat.sourceforge.net/

Hunt	
Subcategoría:	Session hijacking.
Descripción:	Se trata de un programa que permite realizar secuestro de sesiones a nivel de red.
URL:	http://packetstormsecurity.org/sniffers/hunt/

6.6.3 Demostración

A continuación se incluyen enlaces a un vídeo donde se demuestra cómo se realiza la técnica de encapsulado de un troyano a partir de la herramienta *Iexpress* (en sustitución de la herramienta *EliteWrap*) junto con las herramienta *Netcat*:

Encapsulado de troyano (Internet)

7 Conclusiones

7.1 Líneas de ampliación

En el presente trabajo se ha mostrado una introducción a las Evaluaciones de Seguridad. Como se ha visto, existen dos tipos de Evaluaciones de Seguridad: Auditoría de Seguridad (también conocidas como Análisis de Vulnerabilidades) y Test o Prueba de Penetración / Intrusión. La diferencia que marca un tipo u otro es si se intentan o no explotar las vulnerabilidades encontradas.

Se han presentado algunas de las técnicas y herramientas utilizadas en un Test de Penetración siguiendo una de las metodologías de facto más ampliamente difundidas. Como resultado de este trabajo, podemos concluir que existen un amplio espectro de técnicas y herramientas para realizar pruebas / ataques sobre sistemas y redes. Dichas pruebas / ataques apuntan a los diferentes niveles existentes en un sistema de información: nivel de red, nivel de sistemas operativos, nivel de aplicaciones, nivel humano, etc. Igualmente, podemos concluir que cualquier organización con conciencia de seguridad, deberá llevar a cabo Evaluaciones de Seguridad de forma periódica para conocer su postura global de seguridad, así como la evolución de ésta en el tiempo. En caso negativo, podría tener que afrontar las consecuencias (sabotajes, robos de información confidencial, pérdida de prestigio, responsabilidades penales, etc.) que se puedan derivar de una mala o nula implementación del cuidado debido (D*ue care*).

La Seguridad de la Información se compone de múltiples dominios de conocimiento. En este trabajo se han presentado las bases de muchos de ellos pero sin llegar a un alto nivel de profundización. Por tanto, existen múltiples posibles líneas de ampliación de este trabajo. A día de la elaboración de este trabajo, parece que hay dos tendencias claramente en alza en el contexto de la Seguridad de la Información. La primera sería la Seguridad de Aplicaciones web y, por tanto, las pruebas encaminadas a detectar vulnerabilidades en ellas. Aparece de hecho, un término específico para designar al proceso que lleva a cabo un hacker ético en la evaluación de la seguridad de las aplicaciones web: *web application penetration test*. En segundo lugar, con la exponencial proliferación de dispositivos móviles inteligentes (*smart phones*, *tablets*, etc.), así como la frecuente difusión en los medios de comunicación de robos de información confidencial almacenada en dichos dispositivos, parece más que previsible que todo lo relacionado con la seguridad de estos dispositivos sea un tema en auge. Abarcando una gran cuota de atención por parte de la comunidad profesional de la Seguridad de la Información en los años venideros.

8 Glosario

- **802.11** – Estándares de redes inalámbricas creados por el IEEE. 802.11a tiene una velocidad de 54Mbps y opera en la banda de los 5 GHz. 802.11b tiene una velocidad de 11 Mbps y opera en la banda de los 2,4 GHz. 802.11g tiene una velocidad de 54 Mbps y opera en la banda de los 2,4 GHz y 802.11n puede alcanzar hasta los 150 Mbps.
- **802.11i** – Un estándar de red inalámbrica desarrollado por el IEEE. Requiere el Protocolo de Integridad de Clave Temporal (TKIP) y del algoritmo de cifrado AES (*Advanced Encryption Standard*)
- **Access Control List (ACL)** – Un método por el cuál se definen qué derechos y permisos tiene una entidad sobre un recurso. En redes, las ACL suelen estar asociadas con cortafuegos y encaminadores en la creación de reglas de filtrado de tráfico.
- **Access Point (AP) -** Un dispositivo LAN inalámbrico que actúa como punto central para todo el tráfico no guiado. El AP es conectado tanto a la red cableada como a la red inalámbrica, proveyendo de acceso a los clientes inalámbricos a los recursos de la red.
- **Acknowledgment (ACK)** – Un indiciador lógico TCP que notifica a una estación origen que el paquete o paquetes anteriores han sido recibidos.
- **Active attack** – Un tipo de ataque que conlleva la interacción con la víctima. Habitualmente debido a que el atacante inyecta algo o altera la red o sistema objetivo.
- **Active Directory (AD)** – El servicio de directorio creado por Microsoft para su uso en redes de este fabricante. Ofrece un conjunto de servicios de red usando el protocolo LDAP (*Lightweight Directory Access Protocol*), la autenticación basada en Kerberos y el inicio de sesión único a los recursos basados en red.
- **Active Fingerprinting** – El inyectado de tráfico en la red para averiguar el sistema operativo de un dispositivo.
- **Ad hoc mode** – Un modo de operación en una red LAN inalámbrica en el cuál los clientes se envían los datos de forma directa sin utilizar un AP. Similar a una conexión por cable punto a punto.
- **Address Resolution Protocol (ARP)** – Definido en la RFC 826, ARP es un protocolo usado para enlazar una dirección IP conocida con una dirección física MAC.
- **Address Resolution Protocol (ARP) Table** – Una tabla de direcciones IP y su correspondientes direcciones físicas MAC almacenada de forma local en un equipo.

- ✔ **Algorithm** – Un método de resolver un problema paso a paso. En Seguridad de la Información, un algoritmo suele ser referido cuando se trata de reglas matemáticas aplicadas en el proceso de cifrado o descifrado.
- ✔ **Anonymizer** – Un dispositivo o servicio diseñado para ofuscar el tráfico entre un cliente e Internet. Generalmente utilizado con la intención de hacer que alguna actividad llevada a cabo en Internet sea lo menos rastreable posible.
- ✔ **Antivirus Software** – Una aplicación que supervisa un sistema o red para identificar y prevenir malware. Este tipo de software se suelen basar en firmas y pueden realizar diferentes acciones sobre el malware identificado.
- ✔ **Authentication** - El proceso por el cuál se determina si una entidad de red (usuario o servicio) es legítimo. Los métodos de llevar a cabo la autenticación se clasifican en: algo que sabes (p.e. contraseñas), algo que tienes (p.e. tarjetas inteligentes) o algo que eres (p.e. Biometría).
- ✔ **Backdoor** – Tanto si es a propósito como si es como resultado de un malware o de otro tipo de ataque, una puerta trasera es la capacidad oculta en un sistema o programa de circunvalar los sistemas de autenticación establecidos.
- ✔ **Banner Grabbing** – Una técnica de enumeración usada para obtener información sobre un sistema.
- ✔ **Bastion Host** – Un sistema ubicado fuera de un cortafuegos para proveer servicios públicos. Suelen estar reforzados para resistir ataques externos.
- ✔ **Biometrics** – El uso de alguna característica física de un demandante para verificar su identidad. Imágenes faciales, huellas dactilares, manuscritos, etc., son ejemplos de características usadas en Biometría.
- ✔ **Black Box Testing** – En pruebas de penetración, es el método de prueba de la seguridad de un sistema sin conocimiento previo alguno sobre el mismo. Diseñado para simular el ataque de un atacante externo.
- ✔ **Black Hat** – Un atacante que irrumpe en sistemas de información con intenciones malintencionadas, sin el conocimiento y el permiso del propietario.
- ✔ **Bluejacking** – Es el envío de mensajes no solicitados a través de Bluetooth a dispositivos con este protocolo habilitado.
- ✔ **Bluesnarfing** – Acceso no autorizado a información como calendarios, lista de contactos, correo electrónico, mensajes de texto, etc., en dispositivos móviles a través de Bluetooth.
- ✔ **Bluetooth** – Una tecnología inalámbrica propietaria y abierta para transmitir información entre dispositivos fijos y móviles en distancias cortas.

- ✔ **Boot Sector Virus** – Un virus que se sitúa a sí mismo en el sector de arranque de un sistema.
- ✔ **Brute-Force Password Attack** – Un método de averiguar contraseñas mediante el cuál todas las posibles combinaciones son enumeradas hasta que se encuentra una coincidencia.
- ✔ **Buffer** – Una porción de memoria usada para almacenar temporalmente datos de entrada o salida.
- ✔ **Buffer Overflow** – Una condición que se da cuando se escriben más datos en un *buffer* del que tiene espacio para recibir.
- ✔ **CAM Table** – Tabla que almacena las relaciones entre direcciones físicas MAC y puertos en un conmutador.
- ✔ **CNAME Record** – Tipo de registro usado en el sistema DNS para registrar un alias a un nombre de dominio.
- ✔ **Community String** – Una cadena usada para la autenticación en SNMP. La *community string* pública es usada para sólo lectura, mientras que la *community string* privada es usada para lectura y escritura. Las *community string* son transmitidas sin cifrar en SNMPv1.
- ✔ **Competitive Intelligence** – Información sobre una organización gratuita y de libre disposición que puede ser utilizada por un atacante para realizar ataques.
- ✔ **Coockie** – Un fichero de texto almacenado en un navegador por parte de un servidor web que mantiene información sobre la conexión. Las *Coockies* son usadas para almacenar información que permite una experiencia de navegación consistente, pero también pueden almacenar parámetros de autenticación. Las *Coockies* pueden estar cifradas y tienen fechas de caducidad definidas.
- ✔ **Covert Channel** – Un canal de comunicación que está siendo usado para un fin para el que no fue previsto. Habitualmente, para transmitir información secretamente.
- ✔ **Cross-Site Scripting (XSS)** – Un ataque mediante el cuál un atacante inyecta código dentro de un sitio web legítimo. Posteriormente, dicho código es entregado a otros usuarios que visitan dicha web, ejecutándose el código en el ordenador de dichos usuarios.
- ✔ **Demilitarized Zone (DMZ)** – Una porción de una red no totalmente expuesta a Internet pero no totalmente aislada de ella. Esta técnica es habitualmente usada en partes de la red que deben permanecer abiertas al público (como un servidor web) pero que también deben acceder a recursos protegidos (como servidores de bases de datos).
- ✔ **Denial of Service (DOS)** – Un ataque con el objetivo de evitar que usuarios legítimos puedan acceder a sus legítimos servicios evitando el normal funcionamiento de redes y sistemas.

- ✔ **Directory Traversal** – También conocido como el ataque *punto-punto-slash*. Usando este ataque, el atacante intenta acceder a directorios restringidos y ejecutar mandatos fuera de los directorios web configurados a tal efecto.
- ✔ **Distributed DOS (DDOS)** – Una técnica de ataque DOS que usa una gran cantidad de sistemas distribuidos para llevar a cabo el ataque.
- ✔ **DNS Enumeration** – El proceso de usar los registros DNS accesibles para enumerar los sistemas internos de una red objetivo.
- ✔ **Domain Name System (DNS)** – Una red de sistemas que traducen direcciones IP numéricas en nombres de dominio y viceversa.
- ✔ **Domain Name System (DNS) Cache Poisoning** – Una técnica de ataque que trata de engañar a un servidor DNS enviándole información fraudulenta. Esta técnica conduce el tráfico de un usuario a un sitio erróneo o malicioso en vez de al sitio originalmente solicitado por el usuario.
- ✔ **DNS Lookup** – El proceso por el cuál un sistema envía un nombre de dominio cualificado a un servidor de nombres para que éste, a su vez, le provea de la dirección IP correspondiente.
- ✔ **Due Care** – La responsabilidad que los directivos y las organizaciones que gestionan tienen a la hora de asegurar que el tipo, coste y despliegue de un control de seguridad es apropiado para la seguridad del activo que se debe proteger.
- ✔ **Dumpster Diving** – Un ataque de seguridad física por el cuál un atacante escudriña la basura y las papeleras en busca de información útil para llevar a cabo ataques actuales o futuros.
- ✔ **Eavesdropping** – El acto de secretamente escuchar conversaciones privadas de terceros sin consentimiento. Se puede llevar a cabo también a través de medios electrónicos.
- ✔ **Enumeration** – En pruebas de penetración, la enumeración es el acto de interrogar un dispositivo o un segmento de red de forma concienzuda y sistemática en busca de información.
- ✔ **Ethical Hacker** – Un experto en seguridad informática que lleva a cabo auditorías de seguridad y pruebas de penetración contra sistemas o segmentos de red, con total conocimiento y consentimiento por parte del propietario de los mismos, con el objetivo final de reforzar la seguridad.
- ✔ **Firewalking** – El proceso sistemático de probar cada puerto de un cortafuegos para determinar las reglas del mismo y los puertos accesibles.
- ✔ **Firewall** – Componente software o hardware que restringe el acceso entre una red protegida e Internet, o entre redes arbitrarias, para bloquear los usos o ataques no deseados.
- ✔ **Footprinting** – Todas las medidas y técnicas llevadas a cabo para obtener información sobre un objetivo. Puede ser de naturaleza pasiva o activa.

- ✔ **Google Hacking** – La manipulación de una cadena de búsqueda, con operadores específicos adicionales, para buscar vulnerabilidades o información muy concreta con el concurso del motor de búsqueda de Google.
- ✔ **Gray Box Testing** – Una prueba de penetración en la que el Hacker Ético tiene conocimientos limitados sobre el objetivo. Diseñado para simular personal interno de una organización que no tiene nivel de administrador en la misma.
- ✔ **Gray Hat** – Un hacker que se sitúa a ambos lados de la línea que separa a los White Hat y a los Black Hat.
- ✔ **Hardware Keystore Logger** – Un dispositivo hardware usado para registrar las pulsaciones de teclado de forma secreta.
- ✔ **Hash** – Una cadena numérica única de tamaño fijo, creada por un algoritmo que se aplica sobre un conjunto de datos de longitud arbitraria y que habitualmente se usa para verificar la integridad.
- ✔ **Hasing Algorithm** – Una función matemática de un solo sentido que genera cadenas numéricas de longitud fija a partir de datos de longitud arbitraria. MD5 y SHA-1 son ejemplos de algoritmos de Hashing.
- ✔ **HIDS** – IDS basado en host. Se trata de un IDS que reside en un sistema, protegiendo contra la manipulación de ficheros y carpetas y contra otro tipo de ataques basados en sistema.
- ✔ **Honeypot** – Un sistema diseñado para recoger información sobre actividad sospechosa.
- ✔ **Human-based Social Engineering** – El uso de la conversación o algún otro tipo de interacción entre personas para conseguir información útil.
- ✔ **Hybrid Attack** – Un ataque que combina un ataque de fuerza bruta con un ataque de diccionario.
- ✔ **Infrastructure Mode** – Un modo de red inalámbrica donde todos los clientes se conectan a la red inalámbrica a través de un punto de acceso central.
- ✔ **Initial Sequence Number** – Un número asignado durante el arranque de sesiones TCP que indica cuánta información ha sido transmitida. Este número es usado por los atacantes para realizar secuestro de sesiones.
- ✔ **Intrusion Detection System** – Una herramienta de seguridad diseñada para proteger un sistema o red de ataques comparando el tráfico contra una lista de firmas conocidas de ataques y/o a través de la detección de comportamientos anómalos.

- ✔ **Intrusion Prevention System** – Una herramienta de seguridad diseñada para proteger un sistema o red de ataques comparando el tráfico contra una lista de firmas conocidas de ataques y/o a través de la detección de comportamientos anómalos. Además, actúa de forma proactiva tomando medidas de protección para evitar las amenazas más significativas.
- ✔ **Kerberos** – Un protocolo de autenticación muy usado desarrollado por el *Massachusetts Institude of Technology* (MIT). La autenticación Kerberos usa tickets, el Servicio de Entrega de Tickets y el Centro de Distribución de Claves.
- ✔ **Keylogger** – Una aplicación software o hardware que captura la pulsaciones de teclado de un usuario.
- ✔ **Lightweight Directory Access Protocol (LDAP)** – Un protocolo estándar de la industria de nivel 7 de la capa OSI, usado para acceder y gestionar información dentro de un servicio de directorio a través de TCP/IP.
- ✔ **Macro Virus** – Un virus escrito en un lenguaje de tipo macro y habitualmente embebido en ficheros ofimáticos.
- ✔ **Malware** – Un programa o código insertado en un sistema, normalmente de forma oculta, con la intención de comprometer la confidencialidad, integridad o disponibilidad de los datos de la víctima, las aplicaciones o el propio sistema operativo. El malware se compone de virus, gusanos y otros códigos maliciosos.
- ✔ **Man in the Middle (MIM) Attack** – Un ataque donde el atacante se posiciona a sí mismo entre los dos extremos de una cominicación, para interceptar y/o alterar los datos que transitan entre ambos.
- ✔ **Master Boot Record Infector** – Un virus diseñado para infectar el sector de arranque.
- ✔ **MD5** – Un algoritmo de hash que produce una salida de 128 bits.
- ✔ **Multipartite Virus** – Un virus que tiene la capacidad de infectar y de transmitirse de varias formas.
- ✔ **Network Address Translation (NAT)** – Una tecnología donde se traducen, en tiempo real, las direcciones IP utilizadas en los paquetes transportados. Una de las ventajas que aporta es que las direcciones IP de las máquinas ubicadas en la red interior quedan ocultas a externos.
- ✔ **Network Interface Card (NIC)** – Un adaptador que provee la conexión física para enviar y recibir datos entre ordenadores y redes.
- ✔ **Network Tap** – Cualquier clase de conexión que permite ver todo el tráfico que pasa. Generalmente, usado en conjunción con un NIDS para supervisar todo el tráfico.

- ✔ **NOP** – Un mandato que indica al procesador del sistema a no hacer nada. Muchos de los ataques de desbordamiento de buffer utilizan varias operaciones NOP contiguas.
- ✔ **NT Lan Manager (NTLM)** – El conjunto de protocolos de autenticación del sistema operativo Microsoft Windows NT 4.0 y que se ha mantenido en versiones posteriores por compatibilidad hacia atrás. NTLM es considerado inseguro y fue reemplazado por NTLMv2.
- ✔ **Null Session** – Una conexión anónima a un compartimiento (*IPC$*) en un ordenador con sistema operativo MS Windows. Este tipo de sesiones permiten la enumeración en máquinas con sistema operativo MS Windows.
- ✔ **Open System Interconnection (OSI) Reference Model** – Un marco de arquitectura de red desarrollado por ISO que describe el proceso de comunicación entre dos sistemas a través de siete capas diferenciadas.
- ✔ **Operating System Attack** – Un ataque que explota el error común de dejar los valores por defecto cuando se instala un sistema operativo.
- ✔ **Overt Channel** – Un camino de comunicación autorizado para la transmisión de datos dentro de un ordenador o de una red de ordenadores.
- ✔ **Passive Attack** – Un ataque contra un protocolo de autenticación en el cuál el atacante intercepta datos en tránsito por la red pero no los altera.
- ✔ **Payload** – El contenido de un paquete. Un ataque requiere que el atacante entregue un contenido malicioso y que éste sea ejecutado por la víctima.
- ✔ **Penetration Testing** – Un método de evaluar la seguridad de un sistema o de una red simulando un ataque desde una fuente maliciosa.
- ✔ **Phishing** – La utilización de medios basados en el uso de ordenadores para engañar a personas para revelar información personal sensible. Usualmente, se lleva a cabo a través de correos electrónicos especialmente manipulados.
- ✔ **Piggybacking** – Ocurre cuando personal autorizado permite (con o sin conocimiento) a alguien pasar a través de una puerta segura, a pesar de que dicha persona no posea un identificador.
- ✔ **Polymorphic Virus** – Código malicioso que utiliza una máquina polimórfica para mutar pero manteniendo su funcionalidad intacta.
- ✔ **Port Address Translation (PAT)** – Un método NAT en el cuál múltiples sistemas internos de una red, usando un esquema de direccionamiento privado, pueden ser relacionados a una única dirección IP pública.
- ✔ **Promiscuous mode** – Una configuración de una tarjeta de red que hace que la tarjeta pase todo el tráfico que recibe a la CPU, en vez de sólo pasar las tramas que están dirigidas al sistema.

- ✔ **Proxy Server** – Un dispositivo configurado para enviar peticiones y respuestas en nombre de otro nodo.
- ✔ **Reconnaissance** – Los pasos llevados a cabo para obtener pruebas e información sobre los objetivos que se desean atacar.
- ✔ **Replay Attack** – Un ataque donde el atacante repite una porción de un intercambio criptográfico con la esperanza de engañar a un servidor y establecer con él un canal de comunicación.
- ✔ **Reverse DNS Lookup** – Se usa para encontrar el nombre de dominio asociado con una dirección IP. Lo opuesto a una DNS lookup.
- ✔ **Reverse Social Engineering** – Un ataque de ingeniería social donde se manipula a la víctima para que ésta llame al atacante en busca de ayuda.
- ✔ **RID** – Identificador de Recurso. La última parte del SID que identifica el usuario al sistema en una máquina con sistema operativo MS Windows. Un RID de 500 identifica la cuenta de Administrador.
- ✔ **Rogue Access Point** – Un punto de acceso inalámbrico desplegado por un atacante para llevar a cabo un *ataque de hombre en medio*.
- ✔ **Rootkit** – Un conjunto de herramientas que permiten acceso a nivel de administrador a un sistema o a una red y que están diseñadas para ocultar el hecho de que el sistema ha sido comprometido.
- ✔ **SAM** – El fichero Gestor de Cuentas del Sistema en el sistema operativo MS Windows. Almacena los hashes de todas las contraseñas del sistema.
- ✔ **Service Set Identifier (SSID)** – Un valor asignado para identificar de forma única a una red inalámbrica. Los SSIDs son enviados por defecto y se ubican en la cabecera de cada paquete. Los SSIDs no proveen de ningún cifrado o seguridad.
- ✔ **Session Hijacking** – Un ataque en el cuál un atacante se sitúa entre dos extremos de una sesión ya establecida y autenticada, para posteriormente utilizar herramientas especializadas para tratar de averiguar los números de secuencia TCP para secuestrar la sesión.
- ✔ **Shoulder Surfing** – Mirar por encima de un usuario autorizado para robar información (como las credenciales de autenticación).
- ✔ **SID** – Identificador de Seguridad. Identificador único e inmutable de un usuario o de un grupo de usuarios dentro de un ordenador con sistema operativo MS Windows.
- ✔ **Smart Card** – Una tarjeta con microprocesasdor y memoria incorporados habitualmente usado con fines de identificación.

- ✔ **Smurf Attack** – Un ataque de denegación de servicio donde el atacante envía un ping a la dirección de difusión de la red, con la dirección IP origen del paquete suplantada con la dirección de la víctima.
- ✔ **Sniffer** – Hardware o software de ordenador que puede interceptar y registrar tráfico conforme éste pasa por una red digital.
- ✔ **SOA Record** – Registro de comienzo de aurotidad. Este registro identifica el servidor de nombres principal de una zona.
- ✔ **Social Engineering** – Un método de ataque no técnico. La Ingeniaría Social es el arte de manipular a las personas, tanto a través de interacción humana o a través de ordenadores, para conseguir información sensible.
- ✔ **Spoofing** – Un método de falsear el origen de los paquetes. Habitualmente, utilizado por los atacantes para hacer difícil el rastreo al origen real de un ataque.
- ✔ **Spyware -** Un tipo de malware que de forma secreta recoge información sobre un usuario.
- ✔ **Stateful Packet Filtering** – Un método de filtrado de tráfico de red que supervisa por completo el proceso de comunicación, incluyendo el origen de la sesión y desde dónde se inició.
- ✔ **Steganography** – El arte y ciencia de ocultar un mensaje o imagen dentro de otro mensaje, imagen, audio o vídeo.
- ✔ **SYN Attack** – Un tipo de ataque de denegación de servicio donde el atacante envía miles de paquetes SYN al objetivo con una dirección IP origen falsa.
- ✔ **SYN Flood Attack** – Un tipo de ataque usado para denegar el servicio a los usuarios legítimos de un recurso de red, sobrecargando la red con peticiones de conexión TCP ilegítimas. Se envían paquetes SYN de forma repetida a la víctima, pero las correspondientes respuestas SYN/ACK son ignoradas.
- ✔ **Telnet** – Un programa de control remoto en el cuál el cliente se ejecuta en un equipo local y se conecta a un servidor remoto. Los mandatos son introducidos localmente y ejecutados en el servidor remoto.
- ✔ **Temporal Key Integrity Protocol (TKIP)** – Un protocolo de seguridad usado en el estándar 802.11i para reemplazar al protocolo WEP sin tener que reemplazar hardware anterior.
- ✔ **Three-way (TCP) Handshake** – Un proceso de tres pasos que ejecutan los ordenadores para negociar una conexión TCP con otro ordenador en la red. Los tres pasos son: SYN, SYN/ACK, ACK.

- ✔ **Transmission Control Protocol (TCP)** – Un protocolo de nivel 4 orientado a conexión para el transporte de datos en segmentos. TCP es considerado fiable porque garantiza la entrega y la apropiada ordenación de los paquetes transmitidos.
- ✔ **Trojan Horse** – Un programa sin la capacidad de copiarse a sí mismo, que aparenta tener un propósito útil, pero que en realidad tiene un propósito malicioso.
- ✔ **Unicode** – Un estándar de codificación internacional, funciona con varios idiomas y que representa cada letra, número o símbolo con un único valor numérico que se aplica en diferentes plataformas.
- ✔ **Uniform Resource Locator (URL)** – Una cadena que representa la ubicación de un recurso web.
- ✔ **User Datagram Protocolo (UDP)** – Un protocolo de nivel 4 no orientado a conexión. UDP es más rápido que TCP, pero no ofrece fiabilidad. Se hace el mejor esfuerzo posible para entregar los datos, pero no se hacen comprobaciones ni verificaciones para garantizar la entrega. UDP es más simple de implementar y es usado cuando una pequeña cantidad de paquetes perdidos es aceptable.
- ✔ **Virus** – Un programa de ordenador malicioso con capacidad de copiarse a sí mismo, que se adjunta a un fichero huésped y que se desplaza con éste de un ordenador a otro.
- ✔ **Virus Hoax** – Un mensaje de correo electrónico que advierte a los usuarios de un virus que en realidad no existe, y que anima a dichos usuarios a pasar el correo a sus contactos.
- ✔ **Vulnerability** – Una debilidad en un sistema de información, en un procedimiento de seguridad, en controles internos o en una implementación, que podría ser aprovechada por una amenaza.
- ✔ **Vulnerability Assessment** – Descripción y evaluación formal de las vulnerabilidades de un sistema de información.
- ✔ **Vulnerability Scanning** – El envío de paquetes o peticiones a otro sistema para obtener información útil para detectar debilidades y poder proteger al sistema de ataques.
- ✔ **War Dialing** – El acto de llamar por teléfono a todos los teléfonos pertenecientes a una organización para encontrar modems abiertos.
- ✔ **White Box Testing** – Un método de prueba de penetración donde el atacante conoce toda la información sobre la red interna. Está diseñado para simular un ataque por parte de un administrador de sistema descontento o un nivel similar.

- ✔ **Wired Equivalent Privacy (WEP)** – Un protocolo de seguridad de redes inalámbricas definido en el estándar 802.11b, cuyo objetivo era el de proveer el mismo nivel de seguridad alcanzado en las redes por cable. Sin embargo, WEP no es considerado seguro a pesar de que autentica a los clientes inalámbricos, cifra las transmisiones y comprueba la integridad de cada paquete transmitido.
- ✔ **Wireless Local Area Network (WLAN)** – Una red de ordenadores confinada a un área relativamente pequeña en la cuál los dispositivos se conectan a través de ondas de radio de alta frecuencia usando el estándar 802.11.
- ✔ **Wi-Fi** – Un término registrado por la Alianza Wi-Fi para definir un estándar usado por dispositivos para conectarse a una red inalámbrica.
- ✔ **Wi-Fi Protected Access (WPA)** – Un protocolo que provee de cifrado de datos en redes inalámbricas que siguen el estándar 802.11.
- ✔ **Worm** – Un programa que se dispersa de forma automática, sin intervención humana a través de mecanismos de red.
- ✔ **Wrapper** – Software usado para enlazar un programa malicioso con un programa legítimo para que el primero se instale cuando se ejecute el segundo.
- ✔ **XOR Operation** – Una operación matemática que requiere dos entradas binarias: si las entradas coinciden, la salida es cero; si las entradas no coinciden, la salida es 1.
- ✔ **Zombie** – Un ordenador que realiza acciones dictadas por un atacante desde una ubicación remota. Los zombies pueden estar activos o ociosos y los propietarios de los sistemas generalmente desconocen que sus sistemas están comprometidos.
- ✔ **Zone Transfer** – Un tipo de transferencia DNS donde todos los registros de una zona son enviados al solicitante.

9 Bibliografía

- Harris Shon, 2007. Shon Harris (2007). *All in one CISSP*. McGraw-Hill Osborne Media; 4 edition (9 noviembre, 2007) 1145 p. ISBN: 978-0071497879

- Walker Matt, 2011. Matt Walker (2011). *All in one CEH*. McGraw-Hill Osborne Media; 1 edition (7 septiembre, 2011) 395 p. ISBN: 978-0071772297

- Skoudis Edward, 2001. Edward Skoudis (2001). *Counter Hack*. Prentice Hall PTR; 1 edition (23 julio, 2001) 592 p. ISBN: 978-0130332739

- Skoudis Edward, 2006. Edward Skoudis; Tom Liston (2006). *Counter Hack Reloaded*. Prentice Hall PTR; 2 edition (2 enero, 2006) 784 p. ISBN: 978-0131481046

- Skoudis Edward, 2003. Edward Skoudis; Lenny Zeltser (2003). *Malware – Fighting Malicious Code*. Prentice Hall PTR; 1 edition (17 noviembre, 2003) 672 p. ISBN: 978-0131014053

- Randall K. Nichols, 2001. Nichols K. Randall; Lekkas C. Panos (2001). *Wireless Security*. McGraw-Hill Telecom; 1 edition (13 diciembre, 2001) 657 p. ISBN: 978-0071380386

- McCarty Bill, 2003. Bill McCarty (2003). *Firewalls*. Anaya Multimedia; 1 edición (15 octubre, 2010) 560 p. ISBN: 978-8441515840

- Perez Agudín Justo, 2006. Justo Pérez Agudín; Carlos Miguel Pérez; Abel Mariano Matas García; Fernando Picouto Ramos; Antonio Ángel Ramos Varón (2006). *La biblia del Hacker*. Anaya Multimedia; edición 2006. 1132 p. ISBN: 978-8441519242