Penetration Testing

En este capítulo, comenzaremos definiendo algunos conceptos clave de la seguridad informática y analizaremos, brevemente, distintos tipos de análisis de seguridad. Luego nos centraremos en el Penetration Testing y veremos sus distintas fases: reconocimiento, escaneo, enumeración, acceso y, finalmente, mantenimiento del acceso.

SERVICIO DE ATENCIÓN AL LECTOR: usershop@redusers.com

INTRODUCCIÓN

En esta primera sección repasaremos algunos conceptos para ponernos de acuerdo con la terminología. Algunos de ellos son los de la tríada **CIA** (**Confidencialidad**, **Integridad**, **Disponibilidad**), que tiene que ver con la **identificación**, **autenticación** y **autorización**, entre otros aspectos. Luego haremos una breve recorrida por los distintos tipos de controles que pueden ser implementados y, finalmente, veremos algunos de los tipos de análisis que se pueden realizar.

Definiciones y conceptos generales

Mucho se ha escrito ya sobre conceptos de seguridad informática, sobre la **tríada CIA** y otros conceptos asociados, por lo que no profundizaremos demasiado en ellos, pero sí los refrescaremos brevemente.

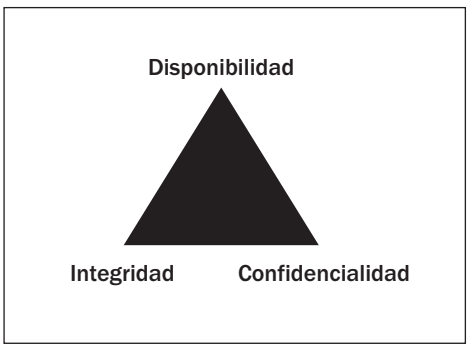

***Figura 1.** Tríada CIA (Confidencialidad, Integridad y Disponibilidad).*

En primer lugar, definiremos esa frase tan conocida que solemos repetir continuamente y que tanto misterio despierta: **seguridad informática**. Con más o menos palabras, se la define como el conjunto de medidas preventivas, de detección y de corrección, destinadas a proteger la integridad, confidencialidad y disponibilidad de los recursos informáticos. En términos generales, todos coincidiremos con ello y si partimos de la segunda parte de esta definición, nos encontramos con los tres

III CONCEPTOS ASOCIADOS A LA TRÍADA

Otros conceptos que se desprenden de la tríada son: **identificación:** mecanismo por el cual los usuarios comunican su identidad a un sistema. **Autenticación:** proceso que comprueba que la información de identificación corresponda al sujeto que la presenta. **Autorización:** corresponde a los derechos y permisos otorgados a un usuario que le permiten acceder a un recurso.

pilares de la seguridad informática: **integridad**, **confidencialidad** y **disponibilidad**, también conocidos por sus siglas en inglés como la tríada **CIA** (Confidentiality, Integrity, Availability, en español Confidencialidad, Integridad y Disponibilidad).
Para desempolvar más conceptos, definámoslos brevemente antes de continuar con nuestro aprendizaje. Hablamos de **confidencialidad** cuando nos referimos a la característica que asegura que los usuarios (sean personas, procesos, etcétera) no tengan acceso a los datos a menos que estén autorizados para ello. Por otro lado, la **integridad** nos indica que toda modificación de la información sólo es realizada por usuarios autorizados, por medio de procesos autorizados. Finalmente, la **disponibilidad** garantiza que los recursos del sistema y la información estén disponibles sólo para usuarios autorizados en el momento que los necesiten.
Retomando la definición de seguridad informática, si nos centramos en la primera parte de la definición, debemos analizar las medidas o controles que se implementan para preservar la tríada, ya que cualquier medida de seguridad que se tome, siempre tiende a preservar uno o más de sus componentes. En la siguiente sección las veremos en detalle para comprender de qué se tratan.

CONTROLES EN SEGURIDAD INFORMÁTICA

Como ya mencionamos, el objetivo de la seguridad informática es **fortalecer** una o varias de las características de seguridad mencionadas, mitigando de esta forma los efectos producidos por las **amenazas** y **vulnerabilidades**. El riesgo de sufrir un incidente de seguridad nunca lo vamos a poder eliminar por completo, pero sí vamos a reducirlo a un nivel tolerable por nuestra organización.
Estos controles pueden clasificarse según dos criterios. Por un lado, dependiendo del **momento** en el que se actúa, tendremos **controles** preventivos, disuasivos, detectivos, correctivos y recuperativos. Los **preventivos** y **disuasivos** toman acción en momentos **anteriores** al incidente, con el objetivo de evitarlo. Los **detectivos** buscan detectar el incidente en el momento en que éste está ocurriendo. Finalmente, los **correctivos** y **recuperativos** tienen lugar una vez que el incidente ocurrió.

MÁS SOBRE LA TRÍADA

Otros conceptos que se desprenden de la tríada son **trazabilidad** (**accountability**), la habilidad para determinar las acciones individuales de un usuario dentro de un sistema, **privacidad**, que determina el nivel de confidencialidad que se brinda a un usuario dentro de un sistema y **no repudio**, la utilización de elementos de información única para validar la acción de un usuario.

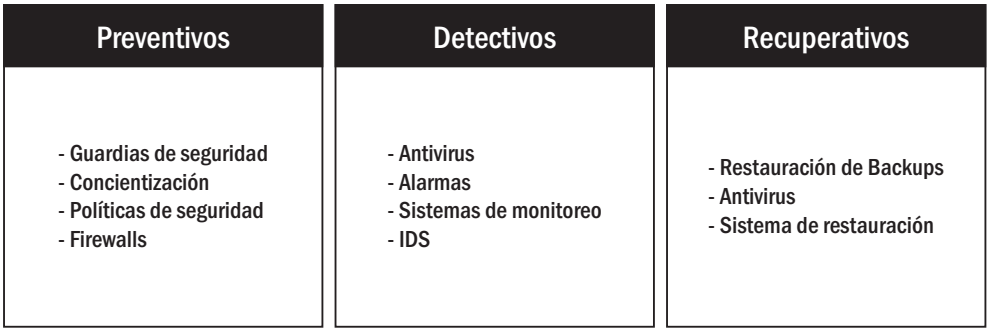

***Figura 2.** Controles divididos en función del momento del incidente.*

Por otro lado, según el tipo de **recursos utilizados**, vamos a clasificarlos en controles físicos, técnicos o lógicos y administrativos. Los **controles físicos** serán aquellos que implementen medidas de seguridad física, como por ejemplo, cerraduras electrónicas, sistemas de acceso biométrico, etcétera. Los **controles técnicos o lógicos** implementan, usualmente, medidas de carácter tecnológico, como sistemas de detección de intrusos, seguridad de las aplicaciones y sistema operativo, etcétera. Finalmente, son muy importantes, aunque muchas veces desvalorizados, los **controles administrativos**. La importancia de estas medidas radica en que son las que suelen determinar, en función de la política de seguridad, las configuraciones que deben cumplir el resto de los controles, por ejemplo, las configuraciones de los controles de acceso y las reglas (desde el punto de vista de las políticas de acceso) que deben implementarse en un firewall.

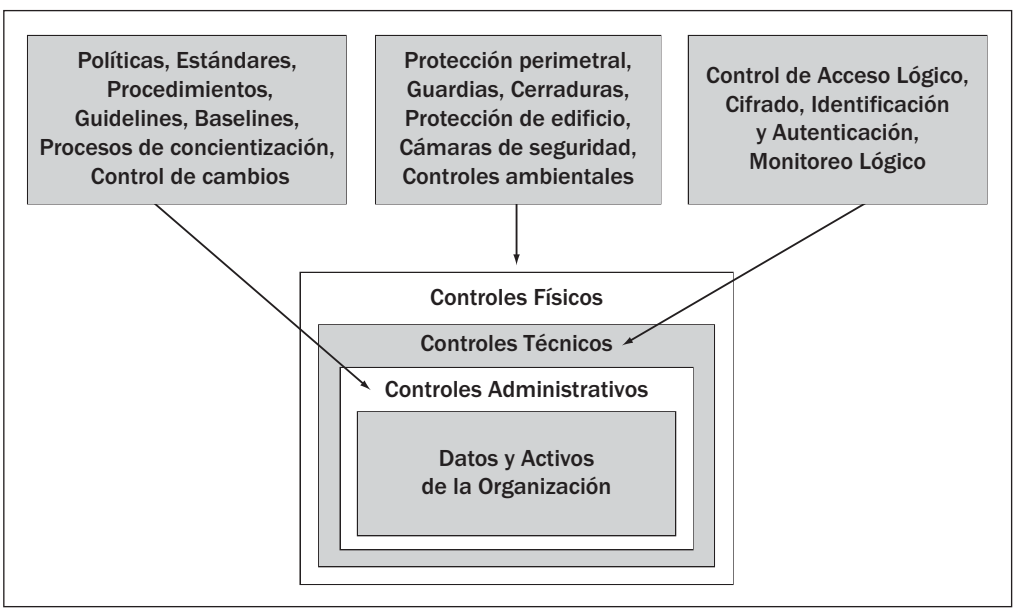

***Figura 3.** Controles organizados en función de los recursos y ejemplos de cada uno.*

Como podemos observar, muchas veces estos controles pertenecen a más de una categoría a la vez, según el punto de vista que tengamos en cuenta. Para analizar la efectividad de esos controles se realizan distintos **análisis de seguridad**. A continuación, veremos dos de ellos: **vulnerability assessment** y **ethical hacking**.

Vulnerability Assessment

Un **VA** (Vulnerability Assessment) es un análisis de **puntos débiles** o **vulnerabilidades** de carácter técnico realizado a un sistema, el cual no necesariamente tiene que estar relacionado con los sistemas informáticos o de telecomunicaciones. Este tipo de análisis también se aplica a diversos campos, como plantas de energía nuclear, procesos de biotecnología, sistemas de distribución de agua, sistemas de distribución de energía y un sinfín de otros ejemplos. En términos generales, estas evaluaciones buscan determinar las amenazas, agentes de amenaza y vulnerabilidades a las que está expuesto el sistema. Esas debilidades están relacionadas con aspectos técnicos que dependen de las características y del contexto en que está implementado el sistema que es evaluado.
En nuestro caso, vamos a referirnos a un VA cuando realicemos un análisis técnico de las vulnerabilidades de una infraestructura de informática y de telecomunicaciones. Puntualmente, se analizarán vulnerabilidades asociadas a distintos servidores, redes, sistemas operativos, aplicaciones, etcétera, todas ellas relacionadas a aspectos técnicos.

Ethical Hacking

A simple vista y considerando la mala fama que tiene la palabra **hacker** en la sociedad, el término **ethical hacking** podría parecer contradictorio. Para echar un poco de luz, veamos cuál es el sentido de este término. En el ámbito de la seguridad informática, el término hacker es utilizado a modo de título por la comunidad, y es otorgado por sus miembros a aquellos que hicieron notables aportes a su desarrollo. En términos generales, vamos a hablar de un hacker como aquel **experto** de una o varias áreas de dominio específicas. Extendiendo el concepto, incluso fuera del ámbito de la informática y la tecnología, podemos decir que define a aquella persona que posee una

{} MÁS INFORMACIÓN SOBRE CONTROLES

Para mayor información sobre los tipos de controles, es recomendable consultar bibliografía específica. En particular, conviene aquélla relacionada con la certificación **CISSP**, por ejemplo, **CISSP All-in-One Exam Guide** (3rd Edition, Shon Harris), **Official (ISC)2 Guide to the CISSP Exam** (Susan Hansche) y **The CISSP Prep Guide: Gold Edition** (Ronald L. Krutz y Russell Dean Vines).

mente curiosa, que le apasiona el conocimiento, el proceso de aprendizaje, el descubrimiento y el deseo de comprender el funcionamiento de las cosas en general. Hasta el momento, sólo hemos mencionado los aspectos técnicos del perfil de un hacker. Si nos enfocamos en el aspecto ético, podremos llegar a clasificarlos en **white hat hackers, black hat hackers** y **grey hat hackers**. Como podemos imaginarnos, los white hat hackers son aquellos profesionales que poseen un **código de ética**, usualmente alineado con el de alguna organización relacionada, y están encargados de **proteger** los sistemas informáticos de los ataques que puedan sufrir. También utilizan su conocimiento en beneficio de la sociedad, por ejemplo, brindando charlas de concientización o participando de entidades sin fines de lucro relacionadas con la seguridad. En la vereda de enfrente tenemos a los hackers black hat, quienes no respetan ningún código de ética y cuyos valores están más relacionados con el dinero o con la falsa sensación de rebeldía frente a la sociedad. Finalmente, los grey hat hackers suelen estar en el borde de la legalidad y pueden cambiar su senda en función de cual sea el bando del **mejor postor**.

Como conclusión, podemos decir que aquellos que cometen delitos utilizando Internet como medio no dejan de ser delincuentes, sólo que cometen sus actividades empleando los sistemas informáticos y tecnológicos como medio. Independientemente del nivel de conocimiento que posean, los fines de sus acciones son **ilícitos**. A partir de ahora, nos referiremos a estos individuos como **atacantes maliciosos** o, simplemente, **delincuentes**.

Habiendo demostrado que el término **ethical hacking** no tiene por qué ser contradictorio, pasemos a analizar en qué consiste. En la sección anterior mencionamos el vulnerability assessment haciendo foco en el contexto de la informática y las telecomunicaciones. Es un análisis puramente técnico, que suele realizarse en forma remota: el **tester** prueba la seguridad de los sistemas a través de Internet. Si extendemos el concepto de VA para que quien realiza el análisis pueda tener acceso físico a las instalaciones e interactuar con el personal de la organización, nos encontramos frente a un **penetration test** o **pentest**. Un ethical hacker tendrá en cuenta lo mencionado anteriormente y usualmente se pondrá en la piel de un atacante, simulando su comportamiento a fin de evaluar cuán efectivas son las medidas tomadas frente a un ataque.

III TEXTOS SAGRADOS

De la misma manera que varias disciplinas tienen sus textos de cabecera, toda biblioteca digital hacker debería contar con los siguientes recursos, que podemos encontrar en Internet: **Hacker Crackdown** (Bruce Sterling, 1992), **Hackers, Heroes of The Computer Revolution** (Steven Levy, 1996), **¿Cómo llegar a ser hacker?** (Eric S. Raymond) y **La catedral y el bazar** (Eric S. Raymond).

FASES DE UN PENETRATION TEST

En esta sección haremos una breve descripción del concepto de Penetration Test y luego veremos sus distintas fases. Vale la pena aclarar que la clasificación de las fases que presentaremos no es única, sino que está hecha sobre la base de criterios y experiencia de los autores y otros colegas. En primera instancia, veremos la fase de **reconocimiento**, donde analizaremos distintas técnicas y métodos. Luego, la fase de **escaneo**, en la cual relevaremos información relativa a la infraestructura, y algo análogo haremos en la fase de **enumeración**. En la fase de **acceso** utilizaremos los medios necesarios para ingresar al sistema objetivo y, finalmente, en la etapa de **mantenimiento**, tomaremos las medidas necesarias para poder acceder al sistema cada vez que lo necesitemos.

Fase de reconocimiento

Antes de comenzar con el análisis de esta etapa, repasemos brevemente algunas características de un **pentest**. En primera instancia, podremos categorizarlo en función de los datos disponibles y los alcances de la evaluación. Así, tendremos los análisis tipo **White box** y **Black box**. En el primero de los casos, el tester tiene a su disposición información sobre la infraestructura de la empresa y la profundidad del análisis está pactada de antemano. En el segundo, no se dispone casi de información del objetivo, con lo cual en este caso la fase de reconocimiento es fundamental. El analista llegará hasta donde sus habilidades y las medidas de seguridad implementadas se lo permitan.

En la práctica, la mayoría de estos tests suelen ser **híbridos**, por lo que encararemos el análisis de estas fases teniendo este punto en mente. Ahora sí, sin más preámbulos, comencemos a ver las características de la fase de reconocimiento. Esta fase es la que más tiempo insume dentro de la planificación. Lo que se busca en primera instancia es **definir** al objetivo y, a partir de ello, obtener la mayor cantidad de información sobre él. Para el caso de **personas físicas**, ejemplos de recopilación de información serían direcciones de e-mail, direcciones físicas, información personal, etcétera. En el ámbito corporativo, además se buscarán direcciones IP, resolución de nombres DNS, etcétera. En esta parte, denominada

{} EL INGENIERO SOCIAL

Este título de honor corresponde a Kevin David Mitnick, el mítico hacker sobre quien se han escrito varias novelas e incluso una película (**Takedown**). Dos libros de su autoría muy interesantes y de fácil lectura son **The Art of Deception** y **The Art of Intrusion**, ambos de la editorial Wiley & Sons.

gathering information, el atacante utiliza varias técnicas o metodologías, por ejemplo, el **footprinting**, **ingeniería social** y **dumpster diving** (trashing).
La importancia de esta fase radica en la necesidad de determinar el objetivo y obtener toda la información posible (dependiendo del alcance que se haya pactado con la organización), que permita realizar un ataque exitoso. En este sentido, la preparación es crítica ya que, al momento del ataque, no hay tiempo para detenerse y volver a empezar. Asociado a esto, dependiendo de cómo se realice la búsqueda de información, tenemos dos métodos distintos. El primero de ellos es la **búsqueda online**, donde vamos a buscar información utilizando Internet. En cambio, la **búsqueda offline** abarca técnicas como las mencionadas dumpster diving e ingeniería social (debido a su extensión e importancia, estas técnicas tienen un capítulo completo dedicado a ellas).

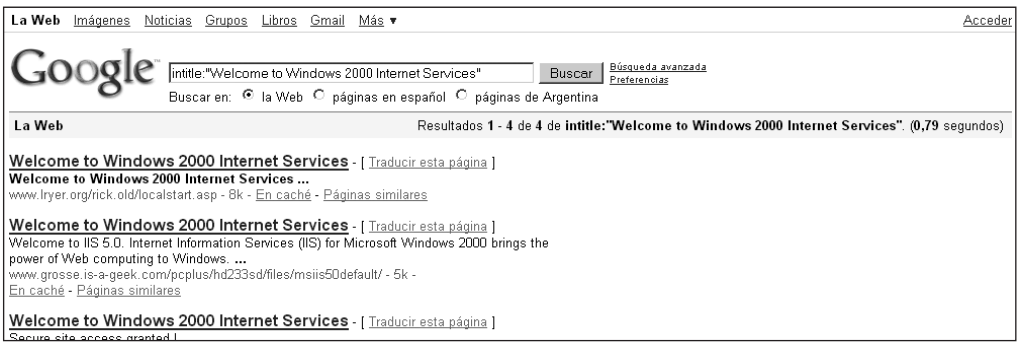

***Figura 4.** Búsqueda de servidores web IIS corriendo sobre Windows 2000 (potencialmente vulnerables).*

Una de las técnicas más utilizadas para realizar búsquedas online es la de Google Hacking. **Google Hacking** consiste en utilizar las funciones de búsquedas avanzadas del conocido buscador, combinadas de forma tal que permitan obtener información muy precisa, como por ejemplo, equipos conectados a Internet que utilicen un sistema operativo en particular que tiene ciertas vulnerabilidades conocidas. Otro ejemplo sería, mediante ciertas cadenas de búsqueda, encontrar dispositivos específicos conectados a Internet, etcétera.

III GOOGLE HACKING

Google Hacking es un término propuesto por Johnny Long que hace referencia al uso de los parámetros de búsqueda avanzada de Google para obtener información en la fase de reconocimiento. Por otro lado, también desarrolló el concepto de **GHDB** (Google Hacking Data Base), que se encarga de almacenar y analizar la información relacionada con estas técnicas.

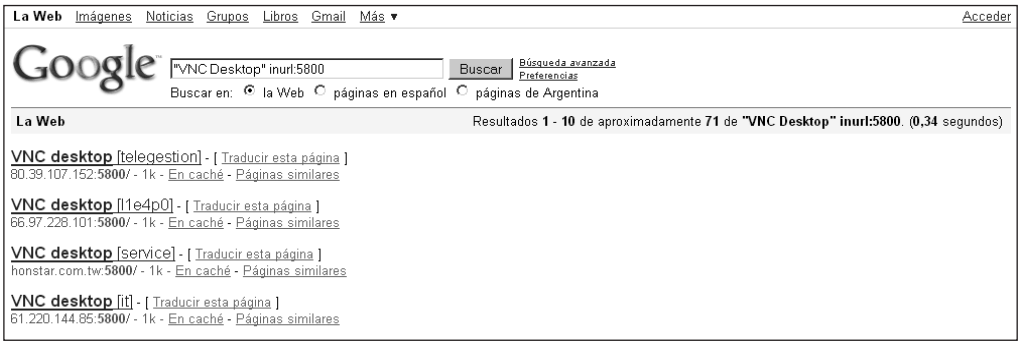

***Figura 5.** Búsqueda de equipos que habilitan la conexión por VNC a través de HTTP.*

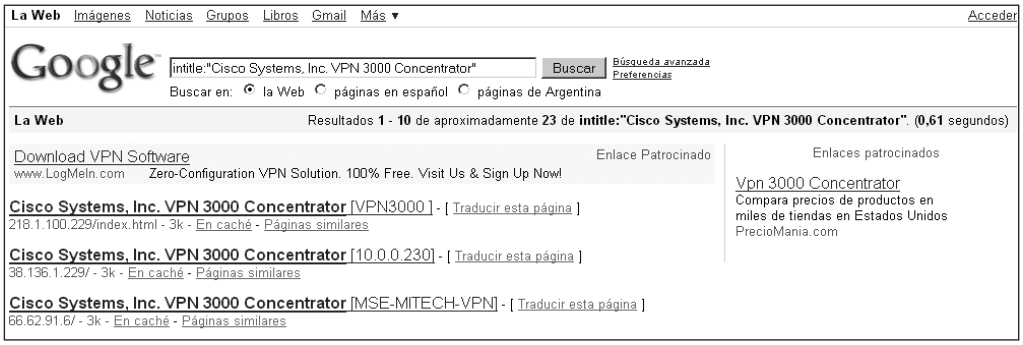

***Figura 6.** Búsqueda de dispositivos Cisco VPN 3000 Concentrators.*

En esta etapa casi no se utilizan herramientas de software, ya que en la mayoría de los casos, con una alta dosis de paciencia y bastante pericia en el uso de los parámetros avanzados de búsqueda de los navegadores, es posible encontrar una gran cantidad de información. Por otro lado, para complementar esa información, existen varios sitios web con recursos online que ofrecen mucha información referente a dominios, servidores DNS y demás. Por ejemplo, **Goolag** es un recurso online (**www.goolag.org**) que podemos utilizar para buscar vulnerabilidades en dominios o sitios de Internet, utilizando técnicas de Google Hacking. Otro sitio de utilidad es **KartOO** (**www.kartoo.org**), que nos permite ver en forma gráfica cómo se relacionan los enlaces que posee un sitio.

RECURSOS ONLINE

A continuación, encontramos algunos recursos online complementarios a técnicas como la de Google Hacking y al uso de herramientas del sistema: Traceroute.org (**www.traceroute.org**), Whois.Net (**www.whois.net**), Maltego (**www.paterva.com/maltego**), FixedOrbit (**www.fixedorbit.com**), Robtex (**www.robtex.com**), Sam Spade (**www.samspade.com**).

Figura 7. Goolag es un buscador optimizado para buscar sitios vulnerables.

Figura 8. KartOO permite relacionar en forma intuitiva los enlaces que referencia un sitio.

Además de Google Hacking y los sitios que vimos hasta aquí, otra alternativa interesante para buscar información online es el uso de ciertas extensiones para el navegador **Mozilla Firefox**. Actualmente, existe una gran cantidad de plugins que agregan funcionalidades desde la óptica del tester de seguridad informática. Debido a esto, es recomendable tomarse un tiempo para recorrer el sitio de extensiones de este popular navegador.

Algunas de estas extensiones son **AS Number**, que nos brinda información sobre los sistemas autónomos (si queremos ampliar nuestros conocimientos, podemos encontrar información sobre lo que son estos sistemas en **http://es.wikipedia.org/wiki/Sistema_autónomo**), **PassiveRecon**, que centraliza varios de los recursos online vistos para darnos información sobre un determinado sitio y **HackBar**, que nos permite auditar la seguridad de distintos sitios web.

***Figura 9.** El complemento **AS Number** es muy utilizado para recopilar información sobres sistemas autónomos.*

***Figura 10. PassiveRecon** es un complemento para que Firefox pueda obtener información útil para el reconocimiento de un sitio web particular.*

FIREFOX Y LAS EVALUACIONES DE SEGURIDAD

Desde la aparición de Firefox, el mundo de los navegadores ya no es el mismo. Continuamente están apareciendo extensiones que agregan funcionalidades que ningún otro navegador posee. En el siguiente enlace podremos ver una recopilación de **extensiones** para Firefox, que se utilizan en la fase de reconocimiento: **www.security-database.com/toolswatch/turning-firefox-to-an-ethical**.

Fase de escaneo

En esta fase utilizaremos la información previa con el objetivo de **detectar vectores de ataque** en la infraestructura de la organización. En primer lugar, comenzaremos con el **escaneo de puertos** y **servicios del objetivo**. Determinamos qué puertos se encuentran abiertos y luego, en reglas generales, asociamos el puerto a un servicio dado. Una vez que hemos finalizado con esto, llega el turno del **escaneo de vulnerabilidades**. Éste nos permitirá encontrar vulnerabilidades en él o los equipos objetivo, tanto del sistema operativo como de las aplicaciones.

***Figura 11. HackBar** es un complemento muy completo que se utiliza para realizar auditorías de sitios y aplicaciones web.*

Conceptualmente, a todo este proceso lo podremos dividir en seis etapas. En cada una de ellas buscaremos distintos tipos de información, desde los equipos online en una red o segmento hasta la planificación del ataque en sí mismo. Vale la pena aclarar que esta división es conceptual, ya que las herramientas suelen cubrir varias etapas juntas en un mismo análisis. Estas etapas son: detección de sistemas vivos o activos, escaneo de puertos, detección del sistema operativo, identificación de servicios, escaneo de vulnerabilidades y planificación del ataque. Para empezar, la forma más simple de ver si un **host** está activo es a partir de la técnica de

LOS FLAGS TCP EN EL ESCANEO DE PUERTOS

Los seis flags de TCP relacionados con los escaneos son: **SYN**, **ACK**, **PSH**, **URG**, **FIN** y **RST**. Si queremos obtener más información sobre los flags TCP y su uso en las técnicas de escaneo, podemos visitar el siguiente enlace: **http://sun-microsystems.org/Tecnicas_de_Deteccion/x215.html**, donde encontraremos información en español.

ping sweep, que consiste en enviar paquetes ping por **broadcast** a los hosts de una red. Si responde, implica que está online y que es un objetivo potencial de ataque. Pero si un escaneo realizado con ping sweep no detecta hosts vivos, no significa que éstos no existan. Suele utilizarse como complemento de otras técnicas, ya que por sí sola no es muy precisa.

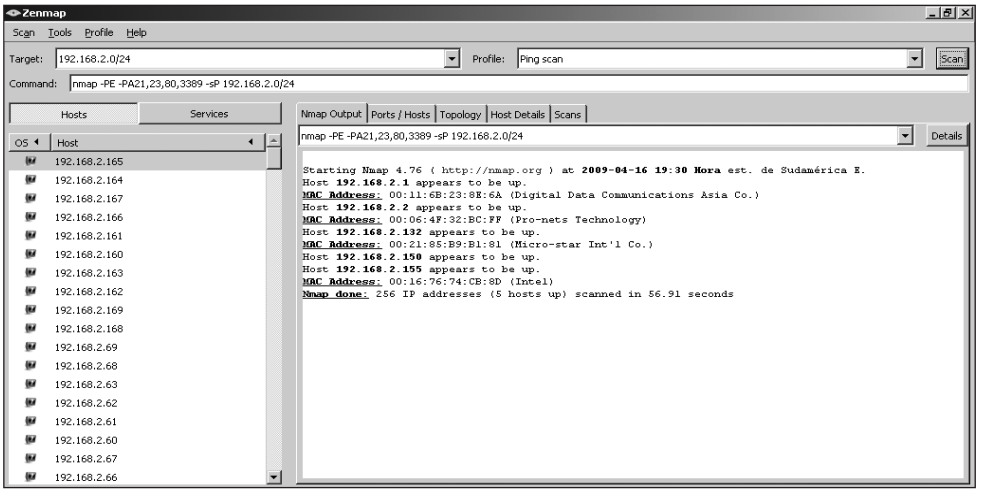

***Figura 12.** El escáner de puertos **Zenmap**, versión gráfica del clásico **Nmap**, realizando una detección de sistemas vivos mediante el **ping scanner**.*

Como segunda etapa, el análisis a partir de los puertos abiertos es el complemento ideal para el ping sweep: si a un equipo se le pueden analizar los puertos, implica que está activo. Sin entrar en detalles, para este análisis se pueden utilizar varios tipos de escaneos que aprovechan distintas características del protocolo TCP (particularmente, la combinación de sus flags y la implementación del protocolo para distintos sistemas operativos). Algunos de ellos son **SYN stealth can**, **FIN scan**, **XMAS tree scan**, **NULL scan**, **FIN scan**, etcétera.

La tercera fase, la de detección del sistema operativo, se realiza a partir de las respuestas que el host brinda frente a determinados paquetes. Como mencionamos anteriormente, cada sistema operativo tiene su implementación del protocolo TCP, por lo cual responde de manera diferente a ciertos paquetes que son interpretados por la aplicación una vez recibidos.

Como cuarta etapa, tenemos la **identificación de servicios**. A grandes rasgos, esto podemos hacerlo a partir del **banner grabbing**, que implica obtener información de la aplicación leyendo los banners predeterminados. Recordemos que los banners son leyendas que traen las aplicaciones donde se brinda información sobre ellas, como la versión, la arquitectura, etcétera. De forma más sencilla, esto también podemos hacerlo asociando los puertos abiertos, hallados en la etapa de escaneo, con el servicio brindado en ese puerto.

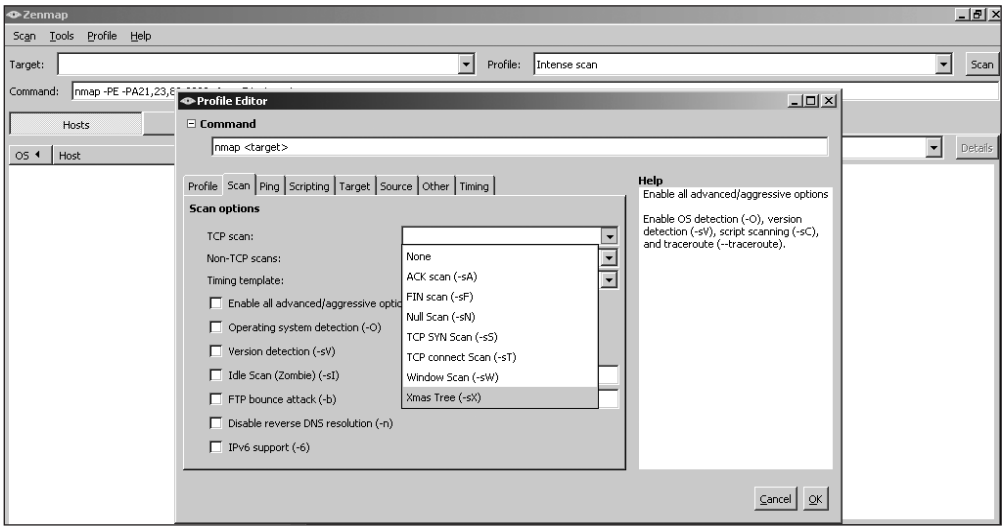

***Figura 13.** En Zenmap podemos generar y definir un perfil de escaneo en función de nuestras necesidades. Tenemos muchas opciones para determinar sus características.*

Con los datos que hemos recopilado en las etapas anteriores comenzaremos con el escaneo de vulnerabilidades. Esto es, dependiendo de los servicios que se estén brindando (como web, e-mail, FTP, etcétera), del sistema operativo base que se encuentre en el equipo (por ejemplo, Windows, Linux, Solaris, Mac OSX, etcétera) y de la aplicación involucrada (que podría ser IIS, Apache, etcétera), se podrá determinar la existencia de vulnerabilidades conocidas y así poder explotarlas posteriormente. Vale la pena aclarar que cuando se trata de vulnerabilidades desconocidas se utilizan otras técnicas.

Finalmente, la planificación del ataque tendrá como objetivo llevar a cabo el proceso de **anonimización** y **ocultación** de huellas del ataque. Esto se debe a que como estamos en la piel del atacante, es importante que, al momento de ingresar al sistema, no queden rastros de lo que se hizo ni cómo se hizo. Esta sexta etapa tiene en cuenta diversas técnicas para llevar esto a cabo, pero escapan al alcance de este libro y no las veremos en profundidad.

III LOS CAZADORES DE VULNERABILIDADES

Un halo de misterio cubre a quienes están en busca de nuevas vulnerabilidades en los sistemas. Cuenta la leyenda que son oscuros personajes con gran conocimiento técnico. Para llevar adelante sus investigaciones sobre nuevas vulnerabilidades, estos personajes utilizan una serie de técnicas entre las que se destacan la auditoría del código fuente, fuzzing e ingeniería inversa.

Fase de enumeración

El objetivo de esta fase es obtener información relativa a los usuarios, nombres de equipos, recursos y servicios de red. Para esto, se generan **conexiones activas** con los sistemas y se realizan **consultas directas** para obtener esa información. Es decir, a diferencia del caso anterior, las consultas siempre se hacen al equipo objetivo y en forma activa, lo que trae aparejado que las conexiones puedan ser detectadas y registradas. En las fases anteriores, un punto importante es que estas técnicas usualmente se realizan dentro de la red interna.

Con estas consideraciones, resulta evidente que la forma de encarar la enumeración de sistemas Windows y Unix/Linux es distinta. Deberemos utilizar técnicas y herramientas diferentes, dependiendo del tipo de sistema que analicemos. No será lo mismo obtener información de usuarios de un **Active Directory**, de un **OpenLDAP** o de un servidor **NIS**. Respecto de los recursos de red y compartidos, éstos podrían enumerarse a partir del mismo protocolo **NETBIOS** o a través de **SNMP** cuando fuese posible.

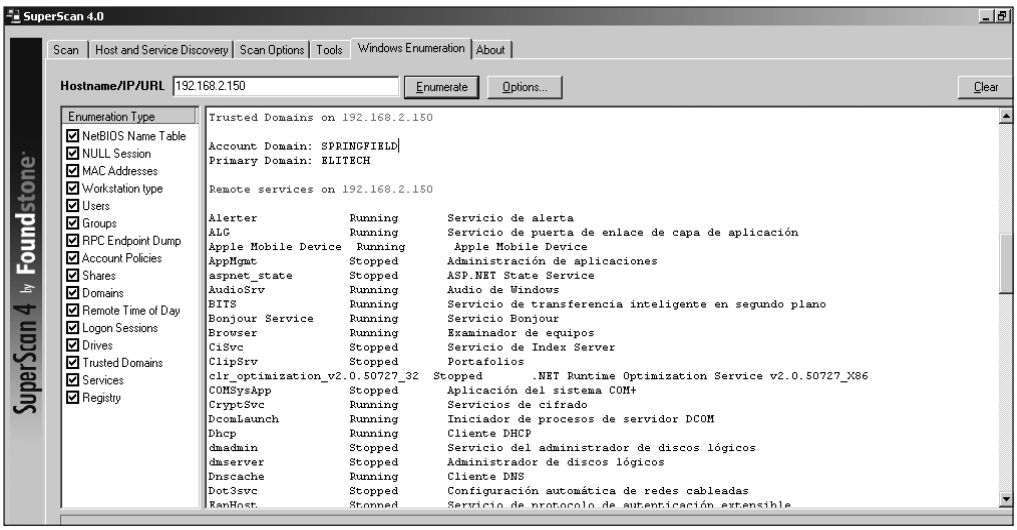

***Figura 14. SuperScan**, de la empresa **Foundstone**, es un escáner de puertos, que además incluye utilidades de enumeración.*

III LA PIEDRA FUNDAMENTAL

Foundstone Inc. es una empresa fundada por George Kurts en 1999. Quizá recordemos su nombre ya que fue autor de algunos libros de la serie **Hacking Exposed**. En sus inicios ofrecía software y servicios, hasta que en 2004 fue adquirida por **McAfee**. Muchas herramientas clásicas de seguridad fueron creadas por esta compañía y puestas a disposición de la comunidad.

Para el caso de las aplicaciones, podemos tener una primera aproximación utilizando comandos simples como **telnet** y **netcat** (**nc**), los cuales establecen conexiones a distintos puertos y permiten obtener banners, dependiendo de la aplicación y su configuración.

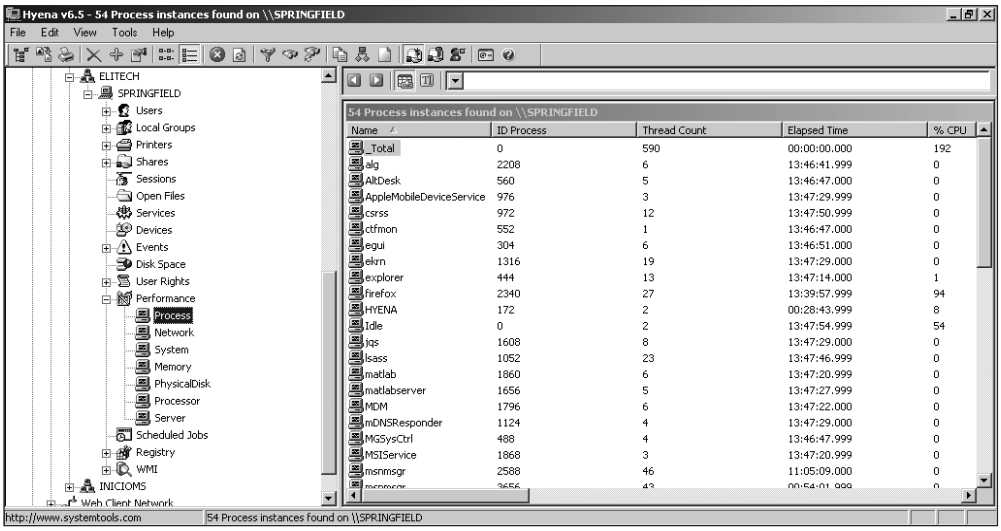

***Figura 15. Hyena** es una herramienta que permite realizar enumeración de distintos equipos dentro de una red.*

Fase de acceso

Una vez detectadas las vulnerabilidades, el gran paso es el **ingreso al sistema** definido como objetivo. Si esto se realiza en el marco de una **simulación** o de un penetration test realizado por profesionales, no se suele tomar control sobre el sistema sino, simplemente, detectar las vulnerabilidades y proponer soluciones para resolver los problemas. Para el caso de un ataque o simulación más realista, esta fase será quizá la que produzca la mayor descarga de adrenalina, ya que aquí se utilizan los recursos y conocimientos de manera condensada. Una vez encontrada una vulnerabilidad, el atacante buscará un **exploit** que le permita explotarla y

EL EFECTO FISIOLÓGICO

En el momento del ataque (aunque sea simulado), la sensación y adrenalina son tan altas que, en ocasiones, el atacante siente el sudor frío propio de los momentos de máximo estrés, previo a cumplir el objetivo que lo llenará de satisfacción. En la película **Swordfish** (2001) hay escenas donde la sensación de quien realiza el ataque (Hugh Jackman) se asemeja bastante a la realidad.

obtener el control, lo que en la jerga se conoce como **ownear el servidor**. Este proceso puede realizarse en forma manual o mediante el uso de algún sistema de **explotación**. Algunos de estos sistemas son **Metasploit Framework** (**www.metasploit.org**), **Core Impact** (**www.coresecurity.com**) o **Immunity Canvas** (**www.immunitysec.com**). En la actualidad, existen varios recursos online donde podemos conseguir exploits e información sobre vulnerabilidades, como por ejemplo, Milw0rm (**www.milw0rm.com**), Open Source Vulnerability Database (**http://osvdb.org**), Common Vulnerabilities and Exposures (**http://cve.mitre.org**), Bugtraq (**www.securityfocus.com/archive/1**), Common Vulnerability Scoring System (**www.first.org/cvss**), Packet storm (**www.packetstormsecurity.org**) y BugReport (**www.bugreport.ir**), entre otros.

***Figura 16. Milw0rm** es un sitio que brinda información de primera mano sobre las últimas vulnerabilidades.*

Dependiendo del tipo de exploit ejecutado, puede ser que el acceso conseguido no posea los privilegios elevados que el atacante desee, y será necesario emprender una

EXPLOIT

La palabra **exploit** proviene del inglés y en español significa **explotar** o **aprovechar**. En informática, es una porción de software, fragmento de datos o secuencia de comandos que aprovecha un error intencionalmente, a fin de causar un comportamiento no deseado en un sistema o aplicación, forzando cambios en su flujo de ejecución y permitiendo que sean controlados a voluntad.

escalada de privilegios con el objetivo de poseer control total del sistema atacado. Una de las formas más comunes de escalar privilegios es, a partir del ingreso al sistema, utilizar otro exploit (en este caso local) que otorgue privilegios de administrador (**root** para Unix/Linux, o **Administrador** o **System** para sistemas Windows).
Una vez que se obtuvo una cuenta con altos privilegios, el siguiente paso suele ser ejecutar comandos o aplicaciones en forma remota. Es decir, lanzar una aplicación desde la ubicación del atacante y que ésta se ejecute en el sistema comprometido. Para esto, es necesario haber establecido previamente un canal entre ambos equipos. Por ejemplo, una vez establecido el canal, podemos ejecutar aplicaciones en forma remota mediante la aplicación **PsExec** de **Sysinternals** (**http://technet.microsoft.com/en-us/sysinternals/bb897553.aspx**).
Una alternativa a la forma de acceso planteada es hacer que el **usuario** que está en el equipo objetivo **intervenga** de forma tal que facilite nuestro objetivo. Muchas veces, esto es necesario ya que se simplifica la explotación, o bien no es posible ejecutar remotamente el exploit. En estos casos, se suele engañar al usuario mediante técnicas de **ingeniería social**, solicitándole por algún medio (e-mail, mensajería instantánea, etcétera) que realice una determinada acción. Lo que el usuario no sabe es que esa acción explota una vulnerabilidad y brinda acceso remoto al atacante. Muchas de estas técnicas las veremos con mayor detenimiento en el próximo capítulo.

Fase de mantenimiento del acceso

Una vez obtenido el acceso, lo que realmente se busca es mantener al equipo comprometido entre las filas del atacante. Para esto, hay que buscar la manera de que el acceso ganado sea perdurable en el tiempo. En la mayoría de los casos, esto se logra a partir de la instalación y ejecución de diversos tipos de **software malicioso**. Si bien el comportamiento va a cambiar dependiendo del tipo de software, el resultado siempre es el mismo: el atacante podrá retomar el acceso al equipo comprometido cada vez que lo desee. Algunos ejemplos de software que se utiliza en esta etapa son **troyanos** y **backdoors**, **keyloggers**, **spyware**, etcétera.
Retomando la planificación del ataque, ya mencionamos que siempre se busca mantener la **anonimidad** en el ataque y, por otro lado, ocultar huellas. En esta fase, el atacante buscará lo mismo. Intentará, con mayor o menor suerte, no dejar rastros de su paso y también esconder los medios por los cuales mantiene el acceso al equipo comprometido.
En Internet hay varios sitios donde podemos encontrar bastante información sobre Penetration Testing. Algunos de ellos son: **www.isecom.org/osstmm, http://csrc.nist.gov**, **www.vulnerabilityassessment.co.uk** y **www.oissg.org**. Una de las metodologías más reconocidas es la **OSSTMM** (Open Source Security Testing Methodology Manual), que especifica en forma muy clara y detallada los pasos necesarios para llevar adelante un Penetration Test.

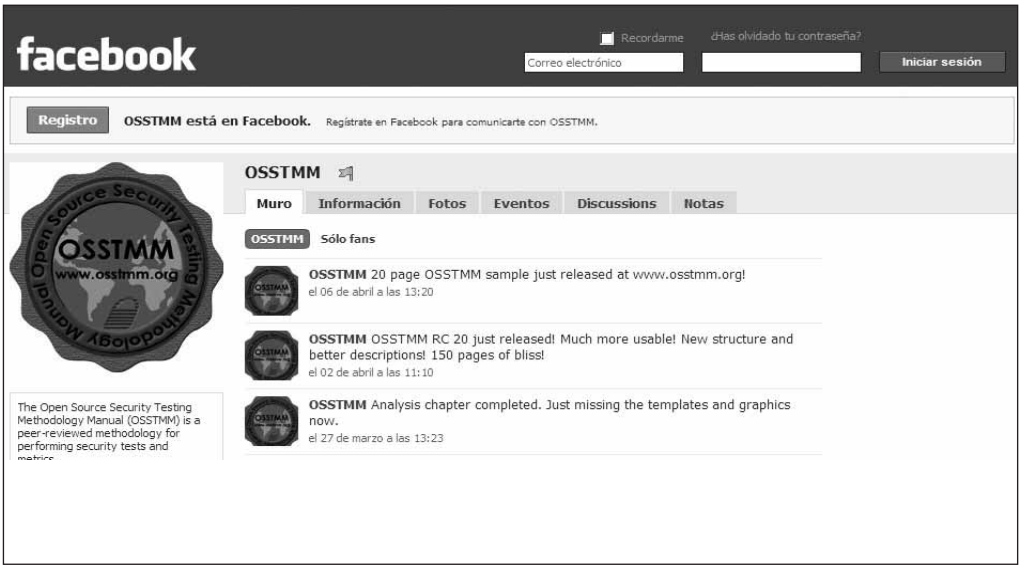

Figura 17. *En la imagen podemos ver el grupo de la* ***OSSTMM*** *en la popular red social* ***Facebook****.*

RESUMEN

En este capítulo hemos repasado conceptos relacionados con la seguridad informática y resumimos algunos tipos de evaluaciones de seguridad, como Vulnerability Assessment, Penetration Test y Ethical Hacking. Por otro lado, analizamos en detalle las fases de un Pentest: fase de reconocimiento, de escaneo, de enumeración, de acceso y de mantenimiento del acceso, haciendo foco en los puntos más importantes de cada una de ellas.

ACTIVIDADES

TEST DE AUTOEVALUACIÓN

1. ¿Qué es la tríada de la seguridad informática? Defina sus componentes.
2. ¿Con qué criterios se pueden clasificar los controles de seguridad informática? Enumere los distintos controles.
3. ¿Cuáles son las principales diferencias entre Vulnerability Assessment y ethical hacking?
4. ¿Cuáles son las diferencias entre white hat, grey hat y black hat hackers?
5. ¿En qué se distinguen los análisis de tipo White box y Black box? ¿Cuál es el más cercano a la realidad?
6. ¿Cuáles son las características principales de la fase de reconocimiento?
7. ¿Qué características tiene la fase de escaneo?
8. ¿En qué consiste la fase de enumeración?
9. Describa las características principales de la fase de acceso.
10. Describa las características principales de la fase de mantenimiento del acceso.

ACTIVIDADES PRÁCTICAS

1. Confeccione una tabla con ejemplos de controles en función del momento en el que ocurre el incidente (filas) y de los recursos utilizados (columnas).
2. En función de las bases expuestas en este capítulo, los enlaces recomendados e información extra disponible en Internet, realice una compilación de los distintos tipos de escaneo existentes y explique su funcionamiento.
3. Investigue acerca de la metodología utilizada para encontrar vulnerabilidades no conocidas (Bug Hunting).
4. Investigue distintas distribuciones o suites de herramientas disponibles orientadas a los análisis de seguridad vistos en el capítulo.
5. Pruebe las distribuciones o suites de herramientas de su preferencia y compárelas.