Hacker Highschool

SECURITY AWARENESS FOR TEENS



LECCIÓN 5
IDENTIFICACIÓN DE SISTEMAS

Hacker Highschool
SECURITY AWARENESS FOR TEENS

ISECOM

WWW.ISECOM.ORG - WWW.OSSTMM.ORG - WWW.HACKERHIGHSCHOOL.ORG - WWW.BADPEOPLEPROJECT.ORG - WWW.OSSTMMTRAINING.ORG

WARNING

The Hacker Highschool Project is a learning tool and as with any learning tool there are dangers. Some lessons if abused may result in physical injury. Some additional dangers may also exist where there is not enough research on possible effects of emanations from particular technologies. Students using these lessons should be supervised yet encouraged to learn, try, and do. However ISECOM cannot accept responsibility for how any information herein is abused.

The following lessons and workbooks are open and publicly available under the following terms and conditions of ISECOM:

All works in the Hacker Highschool Project are provided for non-commercial use with elementary school students, junior high school students, and high school students whether in a public institution, private institution, or a part of home-schooling. These materials may not be reproduced for sale in any form. The provision of any class, course, training, or camp with these materials for which a fee is charged is expressly forbidden without a license including college classes, university classes, trade-school classes, summer or computer camps, and similar. To purchase a license, visit the LICENSE section of the HHS web page at http://www.hackerhighschool.org/licensing.html.

The HHS Project is an open community effort and if you find value in this project we ask that you support us through the purchase of a license, a donation, or sponsorship.

AVISO

El proyecto Hacker Highschool es una herramienta de aprendizaje, y como tal existen riesgos. El mal uso de algunas lecciones puede terminar en daño físico. Existen riesgos adicionales ya que no existen estudios suficientes sobre los posibles efectos de las emisiones en algunas tecnologías. Los estudiantes que sigan estas lecciones deberían ser supervisados y motivados a aprenderlas, probarlas y utilizarlas. No obstante, ISECOM no acepta responsabilidad alguna por el mal uso de la información presentada.

Las siguientes lecciones y cuadernos de trabajo son abiertos y accesibles al público bajo los siguientes términos y condiciones de ISECOM:

Todas las obras del proyecto Hacker Highschool se proporcionan para su uso no comercial con estudiantes de escuelas primarias, secundaria y bachillerato ya sea en centros públicos, instituciones privada, o educación en casa. Este material no puede ser reproducido para su venta bajo ningún concepto. Impartir cualquier clase, formación o actividad con estos materiales cobrando por ello está expresamente prohibido sin la adquisición de una licencia, incluyendo cursos en escuelas, clases universitarias, cursos comerciales, cursos de verano, campamentos de informática, y similares. Para adquirir una licencia, visite la sección LICENCIA en la página web de Hacker Highschool en www.hackerhighschool.org/licensing.html.

El proyecto HHS es resultado del esfuerzo de una comunidad abierta. Si encuentra útil este proyecto, le pedimos que nos apoye mediante la compra de una licencia, una donación o patrocinio.

Índice de contenidos

Colaboradores

Pete Herzog, ISECOM
Glenn Norman, ISECOM
Marta Barceló, ISECOM
Chuck Truett, ISECOM
Kim Truett, ISECOM
Marco Ivaldi, ISECOM
Greg Playle, ISECOM
Bob Monroe, ISECOM
Simone Onofri, ISECOM
Ryan Oberto, Johannesburg South Africa
Dennis King
Mario Platt
Grigoris Chrysanthou
Elisa Vivancos
Alfonso Arjona, @alfonsoarjona.net

ISECOM

Introducción

—Creo que mi ordenador tiene un virus, —dijo uno de mis alumnos— ¿Puedes echarle un vistazo?

Tomé el ordenador portátil de sus manos, no lo abrí sino que lo gire en todas las direcciones, observándolo de cerca.

—A mí me parece un ordenador. —le dije devolviéndoselo.

—Pero, algo le pasa. —insistió Victor— Fui a casa de mi amigo, me metí en Internet y no sé qué entró en mi correo electrónico que envió mensajes a todos mis contactos.

—De acuerdo, ¿cómo miras tu correo? ¿tienes instalada una aplicación? —pregunté.

—No, lo veo en la web, es decir, en Internet.

—¿O sea, en un navegador web?

Asintió.

—Entonces, quiere decir que consultas tu correo electrónico en línea, no en tu ordenador. En ese caso, deberíamos empezar por tu cuenta de correo. ¿Has cambiado la contraseña?

—Sí. Me bloquearon la cuenta hasta que la cambié.

Bajó la cabeza, como si tuviera algo más que contar, pero no le presioné. Me apostaba que ya le habían regañado. Mucho.

—¿Han recibido tus amigos más mensajes? —le pregunte.

—No —dijo, sin dejar de mirar a sus zapatos.

—Y, ¿has escogido una buena contraseña?, ¿no 12345?

Ahora sonrió. —Es una realmente fuerte. Nadie la va a conseguir jamás.

Tenía mis dudas al respecto, pero asentí. —Bien, entonces, parece que ya lo tienes todo resuelto.

—No, —insistió —. ¿Por qué querría alguien hacer eso?

Ya había mordido el anzuelo. —¿Por qué no lo investigas? ¿Tienes alguno de esos correos que recibieron tus amigos?

—Si. Un montón. Me los han reenviado.

Y bien, ¡ahí estaba! Habría apostado que su lista de contactos superaba varias docenas. O cientos. Esto tomaba un cariz muy divertido.

—O me equivoco, o parece que quieres saber exactamente dónde lleva ese enlace en el email.

Sus ojos se abrieron como platos. —¿Quieres decir que podemos hacerlo?

—¡Ja! —sonreí—. Quiere decir que TÚ puedes hacerlo. Yo sólo voy a mostrarte cómo.

Victor se detuvo. —¿Esto es lo que quieres decir con eso que siempre nos estás contando sobre la oveja y el lobo?

—Sí, exactamente. Puedes ser oveja o lobo. ¡Tienes que elegir! —le dije.

De repente, ya no me parecía un niño. —¡Lobo! —me dijo.

* * *

En informática podemos decir que la identificación de sistemas es el paso más importante en cualquier ataque o defensa. Todo lo que se hace después depende de los datos que hayas obtenido en esta fase. ¿Qué sistema operativo tiene el *host* que estas atacando o defendiendo? ¿Puedes ver —u otros pueden— qué aplicaciones o servicios se están ejecutando? Y, qué encuentras sobre los datos personales del administrador? ¿están a la vista en texto plano en algún lugar? Estas son las preguntas que debemos plantearnos en esta fase. Según de que lado estés, te sentirás horrorizado o encantado con lo que se puede obtener fácilmente, si sabes dónde mirar.

Es genial conocer cómo funciona un ataque. Más aún saber como protegerse contra él o rechazarlo. Aquí es dónde empezamos a profundizar y aprender cómo identificar un sistema y encontrar sus debilidades, tanto si se trata de nuestro sistema, como si es otro cualquiera.

Usaremos herramientas que están disponibles públicamente, e incluso te enseñaremos cómo utilizarlas. No tendría sentido indicarte qué software utilizar y no enseñarte cómo usarlo. Como pasa con todos los programas de seguridad, puede ser usado con buenos o malos propósitos. Nuestra misión es mostrarte ambos usos, de manera que puedas fijar tus propios retos de seguridad, a la vez que te proteges de ataques similares.

En esta lección, seguirás a dos individuos: uno de ellos enseña y el otro aprende. El profesor no siempre conoce cuál será la respuesta, de manera que tampoco tú, como lector, tendrás toda la información disponible. Aprende a romper cosas, y aprende cómo arreglar lo que has estropeado. Repite cuando sea necesario.

Fíjate en los parámetros utilizados en los programas. Un pequeño cambio de sintaxis, como pasar una letra de mayúscula a minúscula, puede proporcionarte unos datos completamente diferentes, más aún en distintos sistemas operativos. Estas primeras lecciones son la base de las redes de ordenadores y de cómo funciona Internet. Cada lección se apoya en el conocimiento previo, de manera que no tengas prisa, no obstante saltar entre párrafos y páginas es una buena forma de familiarizarte con los materiales antes de volver sobre ellos para leerlos en profundidad.

Obviamente no querrás pasar por alto ningún conocimiento esencial.

Identificando servidores

—Ok, Victor, ¿qué has encontrado? —Intentaba no apretar los dientes por el miedo que me daba haber hecho clic en aquel estúpido enlace del email que su cuenta hackeada había enviado.

—No hice clic en él con el botón izquierdo del ratón—, dijo Victor, sonriéndome cómo si hubiera leído mis pensamientos. —Lo copié y pegué en un fichero de texto plano.

—¿El texto que puedes ver?, o ¿el enlace en sí?

Frunció el ceño. —¡No soy estúpido! Hice clic con el botón derecho y escogí "Copiar el enlace". Después lo pegué aquí. Mira link.txt.

—Perdona. Tenía que asegurarme. Así que, ¿a dónde va?

—Este dominio raro, Chewmoogoo.com o algo así. También hay una serie cosas detrás —dijo, abriendo su portátil y mostrándome el enlace.

—¡Vaya! —le dije. —Ahora les hemos pillado. Veamos que información podemos recoger y las herramientas que pueden ayudarnos a captarla. Hablemos primero sobre los nombres de dominio y las direcciones IP.

Identificando el propietario de un dominio

El primer paso para identificar un sistema remoto es mirar su nombre de *host*, el nombre de dominio o la dirección IP. Una búsqueda en **whois** sobre un nombre de dominio nos devuelve un montón de información:

- la identidad del propietario del dominio, generalmente nombre y apellidos
- información de contacto, incluyendo dirección postal, números de teléfono y direcciones de correo electrónico
- los servidores DNS donde está registrado el dominio, que pueden darte también información sobre el ISP que le da servicio
- la dirección IP del servidor, otra posible pista del ISP
- información del nombre de dominio, como la fecha que fue creado, cuando ha sido actualizado o cuando expira

Recuerda que hay muchos y distintos registradores de nombres de dominio, y que no todas las bases de datos whois contienen información sobre todos los dominios. Puede que tengas que mirar en más de una base de datos whois para encontrar información sobre el dominio que estas investigando.

Victor lo cazó al vuelo. —Lo tengo, ¿ahora qué hago?

—Aquí tienes tus tareas, —le dije.

Ejercicio

5.1 Consigue el nombre del dominio que estás investigando. (Si no eres Victor, utiliza isecom.org.) Prueba con el siguiente comandos de Linux, Windows y OSX.

```
whois isecom.org
```

¿De quién es el dominio?
¿Cuándo se creó?, ¿cuándo expira? (¿Supone esta fecha de expiración una oportunidad?)
¿Cuándo fue la última vez que se actualizó?
¿Quienes son los distintos contactos listados?
¿Cuales son sus servidores de nombres primario y secundario?

5.2 Ahora haz la misma búsqueda en un navegador (por ejemplo, http://www.whois.net -> "sample.com"). Aquí está la pregunta crítica: ¿Concuerda con lo que obtuviste del comando whois?
Prueba al menos con dos sitios web whois. Intenta con: http://whois.domaintools.com; ¿Puedes encontrar más?

Identificando la dirección IP de un dominio

—Entonces, ¿qué has conseguido? —le pregunté

—Todo este "churro" que he copiado aquí. —Me mostró el fichero de texto.

—¡Está bien! Conserva toda la información. ¿Cuál es la IP del dominio?

—Esta ¿no? —Victor señalaba un número largo.

—Sí. Puedes obtener la dirección IP del dominio con un comando **whois**, o puedes hacer una búsqueda de DNS con un comando **ping**:

```
ping isecom.org
```

—Lo primero que verás es la dirección IP del dominio.

> Si puedes capturar un email del objetivo, examina sus **cabeceras** (mira la Lección 9, Seguridad en correo electrónico); estas te darán la dirección IP del host de origen del correo. También puedes utilizar otros recursos, como por ejemplo buscadores (Lección 20, Ingeniería Social) o herramientas como **Maltego** o **FOCA**. Busca términos como el nombre de la organización que estás analizando, los datos de contacto del registro del dominio, números telefónicos y direcciones. Cada uno de estos puede llevarte a obtener más información.

—Una vez que tienes una IP, o más de una, tienes que averiguar dónde está. Los números IP se asignan a proveedores de servicio por todo el mundo en grandes grupos. Averigua a qué grupo pertenece una IP (y, si puedes, quién tiene los derechos sobre ese grupo). Esto puede ayudarte a encontrar qué servidor o proveedor de servicios utiliza el sitio web y lo mejor, qué país alberga este servidor —le dije a Victor—. Apuesto a que no es éste. Así que esto es lo que harás a continuación.

Ejercicios

Ahora vamos a ver los registros DNS directamente. Otra forma de encontrar información a cerca de un dominio y de servidor(es) es usar la información en el DNS. Para empezar hay tres comandos.

5.3 Abre una ventana de terminal. Prueba este comando:

```
dig isecom.org
```

¿Funciona en tu sistema operativo? Pruébalo en Windows, Linux y OSX.

5.4 Ahora prueba este comando:

```
host isecom.org
```

¿Funciona en tu sistema operativo? Pruébalo de nuevo en Windows, Linux y OSX.

5.5 Por último, prueba este comando:

```
nslookup isecom.org
```

¿Funciona en tu sistema operativo? Una vez más, pruébalo en Windows, Linux y OSX.

¿Cuál es el servidor DNS para tu objetivo?, ¿tiene la organización un servidor de correo electrónico?, ¿tiene el servidor de correo electrónico la misma dirección IP que el servidor web?, ¿qué te sugiere esto?, ¿qué más puedes aprender?

5.6 Una vez tienes la dirección IP, puedes acceder a los registros de los distintos miembros de la **Organización de Recursos Numéricos**, (en inglés, **Number Resource Organization** http://www.arin.net/, http://www.ripe.net/, o http://www.apnic.net/), para comprender cómo se distribuyen las direcciones IP.

Comienza el juego: siega y quema de rastrojos

Era la gran revancha, por lo que respectaba a Jace. La batalla del siglo, así pensaba en ella. No importaba cuanto sudor, sangre, dolor, fuerza física o intelectual fuera necesaria; la ambiciosa adolescente estaba preparada para ganar esta pelea. Tenía que salir victoriosa por que no había un plan B. Su pelo de color chocolate se mecía sobre sus ojos como un torero templando con la muleta. Una última inhalación profunda y relajante, y la asesina de redes estaba preparada para ello.

Con sus ágiles dedos saltando sobre el teclado, valoró la situación e hizo acopio de todos los recursos disponibles. Jace tenía una copia de *Nmap* preparada en su ordenador. Ya había ejecutado *Ping* y *Traceroute* así que, la experta hacker ya estaba preparada para dar una estocada.

Lanzó la primera de una serie de ráfagas de pulsaciones en el teclado. Una ametralladora no podría disparar tan rápido como tecleaba Jace trabajando con comandos de ordenador. *Ping*, ¡lanzado! *Traceroute*, ¡lanzado! Los comandos IP no podían ni respirar frente a su descarga masiva de repiques de teclado. *Time to live*,

¡lanzado! El baño de sangre fue tremendo, a medida que los bits y bytes caían a uno y otro lado del monitor en ráfagas. El interfaz de comandos de línea, (en inglés CLI o *command line interface*) parecía dirigir el bombardeo entrante con potentes switches, con parámetros de ataque flanqueando la red principal.

Jace lanzó su ataque principal para ganar un punto de apoyo dentro de la red. Sus exploradores realizaron un reconocimiento intensivo de los cortafuegos, servidores y routers desplegados delante suyo. Comparó estos datos en el diccionario de riesgos y vulnerabilidades comunes o CVE (*Common Vulnerabilities and Exposures*) y comprobó las referencias con la información obtenida por el escaneo de red de Nmap. Analizó cada debilidad y todas las vulnerabilidades y *exploits* para obtener ventajas tácticas y hacer una valoración de daños. Para Jace, la tregua no era una opción. Estaba ganando.

Aunque aún no había terminado, se dijo. De hecho, todo lo que había hecho era capturar una pequeña parte de los recursos enemigos, sin embargo la inteligencia no tenía precio. Jace sufrió algunas bajas en sus filas. Sus dedos y nudillos estaban un poco doloridos. Tenía una pequeña magulladura cerca de la frente dónde se golpeó con la pantalla de pura frustración. Los tiempos de vida (*Time to Live*, TTL) estaban matándola.

Finalmente, los *banners* de la batalla desvelaron detalles sin necesidad de un interrogatorio o de someterlos a un tortura repetitiva utilizando la técnica de prototipado o "*bread-boarding*". Se reservaba la Raspberry-pi. Jace tenía suficiente información sobre el enemigo para pasar a la fase dos de su ataque a la red. La siguiente etapa requería de correos electrónicos cargados maliciosamente y de la ayuda no intencionada de un usuario del sistema.

Esta era siempre la parte más siniestra de todas las batallas: conseguir traidores, Jace necesitaba usuarios internos que simpatizaran con su causa. Era el momento de romper las buenas prácticas de seguridad. La ingeniería social era la herramienta de extorsión de masas de su arsenal. Tendría que trabajar con correos electrónicos legítimos cargados con soldados Troyanos para traspasar los muros de la red.

A medida que Jace comenzaba a construir cada correo malicioso, sabía que estaba en el bando correcto en este enfrentamiento. No importaba de lo que se apropiara, o cuanto tiempo le llevara, Jace estaba dispuesta a averiguar en que nuevo y secreto sabor de helado estaba trabajando la pastelería del barrio.

El juego continua...

Identificando servicios

—Así que has guardado todo eso ¿correcto? —Sonreí irónicamente intentando que no se notara mucho, pues ya conocía la respuesta aunque pregunté sin poder evitar mi naturaleza de profesor.

Victor me miró de reojo: *calvorota* —estaba pensando, pero dijo: —échale un vistazo. —Y me pasó su portátil.

—Un montón de información ¿no? —Me desplacé hacia abajo por las páginas.

—Sí, necesito otra forma mejor de seguir la pista —dijo Victor, tomando de nuevo el ordenador.

—Seguro que lo consigues. ¿Cuál es la IP del objetivo? —Esta vez sonreí descaradamente.

—Bueno... hay unas cinco. Puede que alguna más. Estoy tratando de imaginarme de dónde salen, porque no a todas se les puede hacer *ping*.

—¡Buen chico!, —pensé. —Una vez que tienes las IP de un dominio puedes empezar a profundizar en los servicios, y eso quiere decir manejar el host. !Qué divertido!

Ping y Traceroute

—Estás empezando por el sitio correcto. Tienes que asegurarte de que las máquinas están activas. Y estás en lo cierto: *ping* es tu aliado. No te olvidarías de hacer *ping* al nombre de dominio, las direcciones IP y los nombres de host, ¿no?

—¿Cuáles de estos son los nombres de *host*? —preguntó Victor.

—Son los que tienen letras y un punto antes del nombre de dominio, como www.isecom.org, —le dije.

—No he visto ninguno.

—Revisa los resultados de tus indagaciones. ¿No probaste con otros nombres?, ¿con www.isecom.org, ftp.isecom.org y mail.isecom.org?

—No...

—Bueno, si obtienes una respuesta, hay vida en esa dirección. Y estás atravesando el cortafuegos. Te dejan colarles comandos ICMP. —Abrí un interfaz CLI y lancé un comando.

```
C:\>ping isecom.org

Pinging isecom.org [216.92.116.13] with 32 bytes of data:
Reply from 216.92.116.13: bytes=32 time=186ms TTL=56
Reply from 216.92.116.13: bytes=32 time=186ms TTL=56
Reply from 216.92.116.13: bytes=32 time=186ms TTL=56
Reply from 216.92.116.13: bytes=32 time=186ms TTL=56

Ping statistics for 216.92.116.13:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 186ms, Maximum = 186ms, Average = 186ms
```

—Puedes intuir a qué distancia está el servidor, tanto en la red como físicamente, por los tiempos de ida y vuelta (*round trip times*). Divídelos por la mitad, y podrás hacerte una idea de la distancia al servidor. Quiero que pruebes otra herramienta, *traceroute*. Se teclea **tracert** en Windows y **traceroute** en Linux. Te mostrará los pasos que dan los paquetes desde tu ordenador hasta el objetivo. Así, —dije y tecleé otra vez:

```
C:\>tracert isecom.org
```

—Ahora, esto es lo que quiero que hagas.

Ejercicios

5.7 Utiliza *traceroute/tracert* para recoger toda la información que puedas encontrar sobre los ordenadores y routers que hay entre tu ordenador y tu objetivo.

5.8 Los ordenadores con direcciones IP similares forman generalmente parte de la misma red. Haz *ping* a la dirección de un sitio web válido (por ejemplo: *ping www.isecom.org* o *ping 216.92.116.13*). Si consigues una respuesta buena, haz *ping* a la siguiente dirección IP. ¿Obtienes respuesta? Prueba con más direcciones adyacentes.

5.9 Utiliza un buscador para encontrar cómo estimar la distancia al servidor.

5.10 Busca una herramienta que te ayude a asociar el servidor con una ubicación física.

5.11 Busca una herramienta en línea de rastreo de rutas visual (*Visual Trace Route*). Existen pocos sitios que ofrezcan herramientas de este tipo. Obtendrás una visualización gráfica de por dónde va tu tráfico.

Nmap

—¿Lo tienes todo? Ahora déjame que te presente a un amiguito —dije, poniendo la voz del mafioso Scarface. Victor me miró como si yo tuviera dos cabezas, así que aclaré mi garganta —¡ejem!—, y terminé: —*Nmap*.

—Puede ser sencillo, muy elaborado. Ejecuta el comando *nmap* con un nombre de host o una dirección IP y escaneará ese *host*. O utilízalo con un conjunto de parámetros para hacer cosas realmente interesantes. Si preguntas correctamente, intentará darte el sistema operativo de tu objetivo. Vamos a usar la opción '*scan TCP*', es decir -sT.

```
nmap -sT 216.92.116.13

Starting Nmap 5.51 ( http://nmap.org ) at 2012-05-28 10:58 GTB Daylight
Time
Nmap scan report for 216.92.116.13
Host is up (1.1s latency).
Not shown: 969 closed ports
PORT   STATE SERVICE
25/tcp  open smtp
80/tcp  open http
110/tcp  open pop3
119/tcp  open nntp
```

```
135/tcp  open msrpc
139/tcp  open netbios-ssn
143/tcp  open imap
445/tcp  open microsoft-ds
465/tcp  open smtps
554/tcp  open rtsp

Nmap done: 1 IP address (1 host up) scanned in 215.42 seconds
```

Es importante recordar que *Nmap* no es la única herramienta para hacer estas exploraciones, lo cual es bueno. Otras herramientas devolverán resultados distintos, aunque de hecho, alguno podría ser intencionalmente engañoso.

Puedes pedirle a *Nmap*, por ejemplo, que adivine el sistema operativo, aunque ¡no deberías fiarte de su predicción! Verifica esta teoría utilizando otras herramientas.

Banner Grabbing (o capturando cabeceras)

Victor estaba eufórico. —¿Ahora, mira lo que he conseguido! —Tenía documentos de texto y una hoja de cálculo en su portátil, dibujos en un block de notas e impresos a color que seguro le habían costado una fortuna en cartuchos de tinta.

—Bien, ahora sabes que tienes algunas máquinas activas, quién las maneja y aproximadamente dónde están. Lo siguiente que tienes que averiguar es el tipo de máquina: ¿cuál es el sistema operativo que se está ejecutando?, ¿qué servicios están en ejecución? —Le pregunté.

Esto le dejó un poco menos eufórico. —¡Uf! ¿Cómo se lo pregunto?

—No tienes que preguntárselo. Deja que la máquina te lo muestre: sistema operativo, servicios y niveles de actualización. Cuando eres el atacante esto facilita tu trabajo; todo lo que tienes que hacer es consultar los *exploits* para ese servicio, software y versión. Si eres el defensor, debes suprimir esta información. O mejor incluso, mentir. —Esto último le dejó pensativo.

—Así que lo que vas a hacer a continuación se llama ***banner grabbing***. Qué nombre más sofisticado: es una **técnica enumeración** para obtener todo tipo de información sobre los servicios y puertos activos en tu objetivo. Te voy a enseñar algunos comandos más. Puedes utilizar *telnet*, *ftp* o *netcat* para capturar el *banner*. El *banner* es ese texto que obtienes en la línea de comandos (al estilo de la vieja escuela) cuando te conectabas y que indicaba qué programa se estaba ejecutándo en el servidor. Así que pruébalo: cuando me conecto a un servidor *FTP* anónimo, me devuelve un *banner*. —Tecleé en mi ventana de terminal:

```
ftp isecom.org

Connected to anon.server.
```

```
220 ProFTPD Server (Welcome . . . )
User (anon.server:(none)):
```

—Ese número 220 es un código que dice que el0 servidor está esperando a un nuevo usuario. ¿No es curioso? El Servidor *ProFTPD* es el programa *FTP* que se está ejecutando en ese host. Ahora estrujaremos la web para encontrar en qué sistemas operativos se puede ejecutar *ProFTPD*, qué puede hacer,... qué se puede estropear, si es que hay algo. —Aporreé el teclado—. Aquí tienes, tu próxima tarea es usar el comando *ftp*.

Ejercicio

5.12 Puedes utilizar FTP tanto con el nombre de host como con la dirección IP, así:

```
ftp isecom.org
```

o

```
ftp 216.92.116.13
```

Prueba con ambos para ver qué *banner* te devuelve el servidor FTP. Tus resultados se parecerán a estos:

```
Connected to isecom.org.
220 ftp316.pair.com NcFTPd Server (licensed copy) ready.
User (isecom.org:(none)):
```

5.13 También puedes utilizar *Telnet* tanto con el nombre de host como con la dirección IP. En cualquiera de los dos casos puedes especificar el puerto, que es el 21 cuando te conectas a FTP:

```
telnet isecom.org 21
```

o

```
telnet 216.92.116.13 21
```

De nuevo, observa qué *banner* te devuelve el servidor —si es que devuelve alguno. Obtendrás algo parecido a esto:

```
220 ftp316.pair.com NcFTPd Server (licensed copy) ready.
```

5.14 Usa *netcat* tanto con el nombre de host como con la dirección IP. De la misma forma que *Telnet*, puedes especificar el puerto, que es el 21 para FTP:

```
nc isecom.org 21
```

o

```
nc 216.92.116.13 21
```

De nuevo, observa qué *banner* te devuelve el servidor, —si es que devuelve alguno.

Banners engañosos

—Aquí está el truco, —le dije a Victor. —Puedes cambiar el *banner*. Esto es un tipo de suplantación o **spoofing,** mentir sobre tu identidad. Así que puedo cambiar mi *banner* para que ponga Servidor "AtiQueTeImporta", que queda simpático, aunque es mejor un sistema Unix con un *banner* que diga: "*WS_FTP Server*", pues ahuyenta a cualquiera, porque es un servidor FTP Windows.

—Espera un minuto, ¿cómo cambias el *banner*? —preguntó.

—Me alegra que lo preguntes —respondí.

Ejercicio

5.15 Entra en Internet y localiza cómo cambiar los *banners* para SMTP, FTP, SSH, HTTP y HTTPS. ¿Es difícil? En otras palabras, ¿deberías confiar sin más en lo que dicen los *banners*?

Banner Grabbing automatizado

—Ahora prueba con esto. Podemos volver a *Nmap* para automatizarlo; tenemos que usar el parámetro -sTV para obtener los *banners* —escribí la primera línea y me devolvió el siguiente informe:

```
nmap -sTV -Pn -n --top-ports 10 --reason -oA hhs_5_06 hackerhighschool.org

Starting Nmap 6.00 ( http://nmap.org ) at 2012-06-23 05:10 CEST
Nmap scan report for hackerhighschool.org (216.92.116.13)
Host is up, received user-set (0.30s latency).
PORT    STATE  SERVICE     REASON      VERSION
21/tcp  open   ftp         syn-ack     NcFTPd
22/tcp  open   ssh         syn-ack     OpenSSH 5.9 (protocol 2.0)
23/tcp  closed telnet      conn-refused
25/tcp  filtered smtp      no-response
80/tcp  open   http        syn-ack     Apache httpd 2.2.22
110/tcp open   pop3        syn-ack     Dovecot pop3d
139/tcp closed  netbios-ssn  conn-refused
443/tcp open   ssl/http    syn-ack     Apache httpd 2.2.22
445/tcp closed  microsoft-ds conn-refused
```

```
3389/tcp closed  ms-wbt-server conn-refused
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at
http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 17.32 seconds
```

—*Nmap* encontró *NcFTPd*, *OpenSSH 5.9 (protocol 2.0)* y *Apache httpd 2.2.22*. ¡Bingo!: el sistema operativo es Unix. A veces los *banners* te indican la versión del sistema operativo, pero vamos a necesitar alguna información más específica —continué—. Esto es lo que quiero que hagas.

Ejercicios

5.16 Utiliza *Nmap* sobre tu objetivo (hackerhighschool.org, si no eres Victor).

5.17 Prueba de nuevo con la opción **–version-intensity number** utilizando números de 0 a 9 para obtener resultados más precisos. ¿Qué diferencias observas en estos informes?

Identificando servicios desde puertos y protocolos

—*Nmap* realizó este último escaneo buscando servicios por defecto. Pero puedes hacerlo también al revés: busca primero puertos abiertos y después mira qué servicio está en realidad tras ellos—, le dije.

—Espera un minuto —protestó Victor—. Los puertos, ¿no son siempre los mismos?

—Sí, en teoría sí. Pero en realidad, los números de puertos son una especie de pacto entre caballeros. Puedo poner mis servicios en puertos distintos, si quiero.

—¡Vale!, ¿cómo lo hago?

—Empieza por mirar en tu ordenador. Ve a la línea de comandos y ejecuta el comando **netstat** con el parámetro **-a** para escanear todos los puertos. Así —, le demostré:

```
netstat -a
```

El joven hacker siguió mi ejemplo, después soltó: —¡eh!, ¿todos estos están abiertos?

Miré a su pantalla —¿tu ordenador, se llama Quasimodo?

```
Active Connections
Proto  Local Address                  Foreign Address        State
TCP    Quasimodo:microsoft-ds         Quasimodo:0            LISTENING
TCP    Quasimodo:1025                 Quasimodo:0            LISTENING
TCP    Quasimodo:1030                 Quasimodo:0            LISTENING
TCP    Quasimodo:5000                 Quasimodo:0            LISTENING
```

```
TCP    Quasimodo:netbios-ssn  Quasimodo:0           LISTENING
TCP    Quasimodo:1110         216.239.57.147:http   TIME_WAIT
UDP    Quasimodo:microsoft-ds *:*
UDP    Quasimodo:isakmp       *:*
UDP    Quasimodo:1027         *:*
UDP    Quasimodo:1034         *:*
UDP    Quasimodo:1036         *:*
UDP    Quasimodo:ntp          *:*
UDP    Quasimodo:netbios-ns   *:*
UDP    Quasimodo:netbios-dgm  *:*
```

—Sí, Quasimodo —sonrió Victor—, el jorobado.

—Pues bien, Victor Hugo. Esto es lo que quiero que hagas.

Ejercicios

5.18 Ejecuta *netstat* en tu ordenador, utilizando el parámetro -a.

```
netstat -a
```

¿Qué puertos están abiertos?

5.19 Ejecuta *netstat* en tu ordenador, utilizando el parámetro -o.

```
netstat -o
```

¿Qué servicios están escuchando bajo los puertos abiertos?

5.20 Ejecuta *netstat* en tu ordenador, utilizando la combinación de parámetros -aon.

```
netstat -aon
```

¿Qué resultado te da esta combinación de parámetros?

5.21 Utilizando un buscador web, busca la correspondencia de estos puertos con los servicios que se ejecutan en ellos. Algunos los necesitarás para las conexiones de red. Pero, ¿de verdad se necesitan todos los servicios que ves en ejecución?

5.22 Ejecuta *nmap*, utilizando el parámetro -sS (para realizar un SYN, también llamado escaneo silencioso) y el parámetro -O (para averiguar el sistema operativo) sobre la

dirección IP 127.0.0.1 como objetivo. La IP 127.0.0.1 se conoce como dirección **loopback** o de bucle local. Siempre se refiere al localhost, es decir, a tu ordenador.

```
nmap -sS -O 127.0.0.1
```

¿Qué puertos abiertos encuentra *nmap*? ¿Qué servicios y programas están usando esos puertos?
Ahora intenta ejecutar *nmap* mientras tienes un navegador o un cliente *telnet* abierto. ¿Cómo cambia esto los resultados?

El escaneo "*stealth*" o silencioso utiliza sólo la primera parte del protocolo de acuerdo de tres pasos (*three way handshake*) de TCP, el paquete SYN, para sondear un puerto sin establecer una conexión completa. Si bien esto te permite soslayar los logs del sistema (que no registrarán tu sondeo mientras no hagas realmente una conexión), NO es indetectable. Cualquier sistema de detección de intrusiones verá tus enormes y pringosas huellas digitales por toda la red, así que no te creas que estás siendo totalmente silencioso.

5.23 *Nmap* tiene otros parámetros de comando en linea. ¿Qué hacen -sV, -sU, -sP, -A, --top-ports y --reason? ¿Qué otras posibilidades existen? Si fueras un atacante y quisieras permanecer oculto, en lugar de ir aporreando el servidor ¿qué parámetros no deberías usar y cuáles sí?

5.24 Vete a www.foundstone.com, localiza, descarga e instala **fport** en tu Windows. Es parecida a *netstat*, pero también da detalles sobre los programas que están usando los puertos y protocolos abiertos. Ejecútalo. ¿Cómo lo hace en comparación con *netstat*?

System Fingerprinting, las huellas dactilares digitales del sistema

—No irás tropezando y haciendo ruido, ¿no? —le pregunté.

Victor respondió lentamente, pensando en serio lo que le había dicho. —No, creo que no. Pero, ¿es tan importante?, es decir, sus servidores están de alguna forma...

Le interrumpí. —No sé dónde están, ni me importa, vas a trabajar con ética, y con cuidado, al menos mientras trabajes conmigo.

—Ok, —dijo, avergonzado.

—Es una buena práctica no dejar pistas. Lo cual es casi imposible. Pero, deberías procurarlo siempre. Precisamente en las pistas es en lo que vas a trabajar a continuación. En realidad sobre las huellas digitales...

—¡Eh! ¡No son lo mismo!

—¡Bien! Me has pillado. Pero dejando eso de lado, vamos a agrupar todo para tomar la huella digital de tu objetivo, encontrar el SO y todos sus servicios.

Escaneando ordenadores remotos

—Por fin ¿qué obtuviste de tus escaneos silenciosos? —le pregunté. Victor me mostró un informe que había copiado en un documento de texto.

```
nmap -sS -O 216.92.116.13


Starting Nmap 5.51 ( http://nmap.org ) at 2012-05-28 16:54 GTB Daylight
Time
Nmap scan report for isecom.org (216.92.116.13)
Host is up (0.19s latency).
Not shown: 965 closed ports

PORT   STATE  SERVICE
21/tcp  open   ftp
22/tcp  open   ssh
25/tcp  filtered smtp
26/tcp  open   rsftp
80/tcp  open   http
110/tcp  open   pop3
111/tcp  filtered rpcbind
113/tcp  filtered auth
135/tcp  filtered msrpc
139/tcp  filtered netbios-ssn
143/tcp  open   imap
161/tcp  filtered snmp
179/tcp  filtered bgp
306/tcp  open   unknown
443/tcp  open   https
445/tcp  filtered microsoft-ds
465/tcp  open   smtps
514/tcp  filtered shell
543/tcp  open   klogin
544/tcp  open   kshell
587/tcp  open   submission
646/tcp  filtered ldp
800/tcp  filtered mdbs_daemon
993/tcp  open   imaps
995/tcp  open   pop3s
```

```
1720/tcp filtered H.323/Q.931
2105/tcp open   eklogin
6667/tcp filtered irc
7000/tcp filtered afs3-fileserver
7001/tcp filtered afs3-callback
7007/tcp filtered afs3-bos
7777/tcp filtered cbt
9000/tcp filtered cslistener
12345/tcp filtered netbus
31337/tcp filtered Elite
Device type: general purpose|storage-misc
Running (JUST GUESSING): FreeBSD 7.X|6.X (88%)
Aggressive OS guesses: FreeBSD 7.0-BETA4 - 7.0 (88%), FreeBSD 7.0-RC1
(88%), FreeBSD 7.0-RELEASE - 8.0-STABLE (88%), FreeBSD 7.0-STABLE (88%),
FreeBSD
 7.1-RELEASE (88%), FreeBSD 6.3-RELEASE (86%), FreeNAS 0.7 (FreeBSD 7.2-
RELEASE) (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 8 hops
OS detection performed. Please report any incorrect results at
http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 24.09 seconds
```

—¿Ves todos esos puertos marcados como **filtered**? Significa que están protegidos por un cortafuegos. Son muy comunes y vulnerables, de manera que siempre deberían estar bloqueados. Pero mira: los puertos 21, 22 y 80, son FTP, Secure Shell y HTTP, todos están abiertos. — Le miré de reojo.

—¿Objetivos potenciales? —preguntó esperanzado.

—Bien, una presa legítima, por lo menos. ¡De acuerdo! Lo último que hace *Nmap* es intentar averiguar el sistema operativo de tu objetivo. La inmensa mayoría de las veces, como ahora, sólo realiza una suposición arriesgada (*aggressive OS guesses*), aunque esta suele ser bastante buena. Ya que la exploración muestra abiertos FTP y SSH, los *banners* que capturaste serán la siguiente evidencia.

—Busca en la web, nos dirá que *NcFTPd* es un programa Unix y que *FreeBSD* es un sistema operativo del tipo Unix. *SSH* por lo general lo encontrarás en sistemas operativos tipo Unix. Así que es muy probable que en el servidor se esté ejecutando alguna versión de *FreeBSD*. Sabes que esos *banners* pueden ser falsos, pero es una suposición razonable.

—Ahora dependiendo de dónde esté tu objetivo, tu próximo paso podría ser encontrar el ISP. Es posible que el ISP sea un ISP conocido por alojar *spammers* y sitios maliciosos —haz una búsqueda— pero deberías poder presentar una queja y conseguir que bloquearan a tu malvado atacante. En tu caso, creo que no va a ser un ISP con el que se pueda tratar...

—Porque está en…—exclamó Victor, pero levanté mi dedo.

—¡Alto! Tu información es tu información. No la necesito,y debe ser así mientras seas ético y seguro. Lo que eres.

Victor asintió.

—Y bien, ¿qué vamos a hacer? —pregunté.

—Bueno, tienen un servidor web ejecutándose, ¿cierto? —comenzó Victor, y no pude más que sonreír.

Alimenta tu mente: profundizando con *Nmap*

Digamos que has identificado el nombre de host, el propietario, la red y que has verificado que el host está activo. Ahora de cara a identificar un sistema necesitas encontrar algunos puertos abiertos. No te olvides que el host puede estar activo pero tener todos los puertos cerrados (o incluso filtrados).

Para ello puedes utilizar la famosa herramienta *Network Mapper* (alias **nmap**) de Fyodor. *Nmap* es un escáner de puertos capaz de sondear remotamente ordenadores en búsqueda de puertos abiertos y sus correspondientes servicios. Cuando ejecutas *nmap*, según los parámetros que utilices en el comando, obtendrás una lista de puertos abiertos y los servicio o protocolos que utilizan esos puertos. *Nmap* sería incluso capaz de averiguar qué sistema operativo está utilizando tu ordenador.

Nmap tiene muchas opciones y tipos de escaneo. Utilizaremos algunos parámetros de *nmap*, si bien siempre puedes utilizar:

```
nmap --help
```

o

```
man nmap
```

para ver los detalles.

Antes de que empecemos, ¿has leído la Lección 3? ?No? ¡Ahora es el momento! ¿Ya está? ¿No? ?¡Entonces hazlo ahora!

Ok, explica las diferencias entre TCP y UDP y describe la negociación en tres pasos (*three-way handshake*). Conocer su funcionamiento es importante para entender cómo trabaja *nmap*.

La sintaxis de *nmap* es:

```
nmap scan-techniques host-discovery options target
```

- **scan-techniques** especifica qué tipo de paquetes se utilizarán y cómo se deberían interpretar las respuestas del objetivo. Las principales técnicas disponibles son:
  - **-sS** escaneo SYN (sí, solo la primera parte del three-way handshake)
  - **-sT** escaneo TCP Connect (el three-way handshake completo)
  - **-sA** escaneo ACK (envía sólo paquetes ACK)
  - **-sU** escaneo UDP
  - **-O** detección de SO
  - **-A** todas las funcionalidades como detección de SO, *plugins*, *traceroute*
- **host-discovery** especifica las técnicas utilizadas para averiguar si un host está activo o no; si está activo será escaneado, en caso contrario no.
  - **-PE** comprueba si el host responde a un *ping*
  - **-PS** comprueba si el host responde a un SYN
  - **-PA** comprueba si el host responde a un ACK
  - **-PU** comprueba si el host responde a un datagrama UDP

- **-PN** no comprueba si están activos, trata a todos los host como activos (utilizaremos esta porque sabemos que nuestro objetivo está activo, pues ya lo hemos comprobado previamente)

- **options** especifica más detalles para el tipo de escaneo seleccionado, como por ejemplo:
  - **-p1-65535** números de puerto a escanear (en este ejemplo de 1 a 65535).
  - **--top-ports <number>** *nmap* conoce cuales son los puertos que más se utilizan, y puede escanear sólo un número de ellos, especificado en <number>
  - **-T0, -T1, -T2, -T3, -T4** para indicar la velocidad de escaneo, siendo 0 lento y 4 rápido (más lento quiere decir también más silencioso y con menor congestión de red)
  - **-oA <filename>** para indicar el archivo donde guardaremos los resultados en cualquiera de los tres formatos principales soportados por *nmap* (siempre lo utilizaremos para llevar un control de nuestras actividades)
  - **--reason** *nmap* escribe cómo interpretar los resultados (recomendado)
  - **--packet-trace** parecido a –reason pero verás las trazas de tráfico (utilízalos para aprender sobre una técnica de exploración y para descubrir errores)
  - **-n** no resuelve DNS (no utilizaremos DNS porque lo hemos analizado ya manualmente)

Escaneo TCP

Nuestro primer escaneo comienza con le siguiente comando:

```
nmap -sT -Pn -n --top-ports 10 -oA hhs_5_tcp hackerhighschool.org
```

Lo que nos da el este resultado:

```
Starting Nmap 6.00 ( http://nmap.org ) at 2012-06-23 04:10 CEST
Nmap scan report for hackerhighschool.org (216.92.116.13)
Host is up (0.23s latency).
PORT    STATE  SERVICE
21/tcp  open   ftp
22/tcp  open   ssh
23/tcp  closed  telnet
25/tcp  filtered smtp
80/tcp  open   http
110/tcp open   pop3
139/tcp closed  netbios-ssn
443/tcp open   https
445/tcp closed  microsoft-ds
3389/tcp closed  ms-wbt-server
```

```
Nmap done: 1 IP address (1 host up) scanned in 2.04 seconds
```

Encontramos algunos puertos abiertos (*open*), otros cerrados (*closed*) y uno filtrado (*filtered*). ¿Qué significa esto? Depende del tipo de escaneo (en este caso -sT). Y podemos utilizar la opción --reason para ver por qué *nmap* ha inferido ese estado en particular.

```
nmap -sT -Pn -n --top-ports 10 --reason -oA hhs_5_tcp_02
hackerhighschool.org

Starting Nmap 6.00 ( http://nmap.org ) at 2012-06-23 04:17 CEST
Nmap scan report for hackerhighschool.org (216.92.116.13)
Host is up, received user-set (0.22s latency).
PORT    STATE  SERVICE    REASON
21/tcp  open   ftp        syn-ack
22/tcp  open   ssh        syn-ack
23/tcp  closed telnet     conn-refused
25/tcp  filtered smtp     no-response
80/tcp  open   http       syn-ack
110/tcp open   pop3       syn-ack
139/tcp closed netbios-ssn conn-refused
443/tcp open   https      syn-ack
445/tcp closed microsoft-ds conn-refused
3389/tcp closed ms-wbt-server conn-refused

Nmap done: 1 IP address (1 host up) scanned in 2.26 seconds
```

Ahora conocemos cómo *nmap* "mapea" las respuestas a los estados de un **escaneo TCP**:

- ***open***: el objetivo responde con un paquete SYN ACK
- ***closed***: conexión TCP rechazada
- ***filtered***: sin respuesta desde el objetivo

Cuando encuentras puertos abiertos y filtrados utiliza otras técnicas de escaneo para averiguar exactamente por qué.

Escaneo SYN

Otra conocida técnica de exploración es el escaneo SYN. Cuando *Nmap* está haciendo este tipo de escaneo, envía sólo un paquete SYN sin completar la negociación en tres pasos (*three-way handshake*). A este escaneo también se le conoce como medio-abierto (*half-open*) o silencioso (*stealth*) porque las conexiones TCP no se completan. (Es evidente que aunque tu objetivo puede no estar

registrando la conexión, aún estás haciendo "ruido" digital que puede ser detectado.) Utiliza el escaneo del tipo -sS de la siguiente forma:

```
nmap -sS -Pn -n --top-ports 10 --reason -oA hhs_5_syn
hackerhighschool.org


Starting Nmap 6.00 ( http://nmap.org ) at 2012-06-24 12:58 CEST
Nmap scan report for hackerhighschool.org (216.92.116.13)
Host is up, received user-set (0.15s latency).
PORT   STATE  SERVICE    REASON
21/tcp  open   ftp      syn-ack
22/tcp  open   ssh      syn-ack
23/tcp  closed  telnet    reset
25/tcp  filtered smtp     no-response
80/tcp  open   http     syn-ack
110/tcp open   pop3     syn-ack
139/tcp filtered netbios-ssn  no-response
443/tcp open   https     syn-ack
445/tcp filtered microsoft-ds no-response
3389/tcp closed  ms-wbt-server reset


Nmap done: 1 IP address (1 host up) scanned in 1.81 seconds
```

Los resultados son similares al escaneo TCP, aunque fíjate en las diferencias entre un escaneo TCP completo (*full*) y un escaneo SYN medio abierto (*half-open*), comparando los resultados (con –reason y –packet-trace) utilizando el mismo objetivo con -sT, -sS y -sA (escaneo ACK).

Escaneo UDP

Otra técnica de exploración es el escaneo UDP (-sU); es fundamental utilizar --reason para obtener buenos resultados.

```
nmap -sU -Pn -n --top-ports 10 --reason -oA hhs_5_udp
hackerhighschool.org


Starting Nmap 6.00 ( http://nmap.org ) at 2012-06-23 04:28 CEST
Nmap scan report for hackerhighschool.org (216.92.116.13)
Host is up, received user-set (0.23s latency).
PORT   STATE      SERVICE   REASON
53/udp  closed     domain    port-unreach
67/udp  open|filtered dhcps     no-response
123/udp closed     ntp     port-unreach
```

```
135/udp closed    msrpc    port-unreach
137/udp closed    netbios-ns  port-unreach
138/udp closed    netbios-dgm port-unreach
161/udp closed    snmp     port-unreach
445/udp closed    microsoft-ds port-unreach
631/udp closed    ipp      port-unreach
1434/udp closed    ms-sql-m   port-unreach

Nmap done: 1 IP address (1 host up) scanned in 2.05 seconds
```

Puede resultar un poco confuso. ¿Qué ha pasado? Vemos alguna de las razones: *port-unreach* (inalcanzable, i.e. cerrado) y *no-response* (abierto|filtrado). ¿Por qué? Necesitamos más detalles. Podemos utilizar la opción de seguimiento de paquetes (--packet--trace) y limitar le escaneo a dos puertos, por ejemplo los puertos UDP 53 y 67:

```
nmap -sU -Pn -n -p53,67 --reason --packet-trace -oA hhs_5_udp_02
hackerhighschool.org

Starting Nmap 6.00 ( http://nmap.org ) at 2012-06-23 04:32 CEST
SENT (0.0508s) UDP 192.168.100.53:54940 > 216.92.116.13:67 ttl=46
id=54177 iplen=28
SENT (0.0509s) UDP 192.168.100.53:54940 > 216.92.116.13:53 ttl=37
id=17751 iplen=40
RCVD (0.3583s) ICMP 216.92.116.13 > 192.168.100.53 Port unreachable
(type=3/code=3) ttl=54 id=1724 iplen=56
SENT (2.5989s) UDP 192.168.100.53:54941 > 216.92.116.13:67 ttl=49
id=33695 iplen=28
Nmap scan report for hackerhighschool.org (216.92.116.13)
Host is up, received user-set (0.31s latency).
PORT  STATE     SERVICE REASON
53/udp closed    domain port-unreach
67/udp open|filtered dhcps  no-response

Nmap done: 1 IP address (1 host up) scanned in 4.15 seconds
```

Descubrimos que 192.168.100.53 enviaba paquetes UDP a los puertos 53 y 67 de hackerhighschool.org. ¿Qué estaba pasando? El puerto 67 no da respuesta y del 53 recibimos *unreachable* (T03C03).

Que un puerto devuelva *Unreachable* (inalcanzable) quiere decir que está cerrado, y si no da respuesta, —incluso en el caso de que sea una repuesta normal para UDP—, no sabremos si el servicio está activo o no porque el protocolo UDP sólo puede responder si recibe los paquetes correctos. ¿Podemos investigarlo más? Sí, utilizando

el escaneo de servicios -sV con el cual *nmap* intenta enviar paquetes conocidos por los servicios UDP.

Escaneo de Servicios o Service Scan (UDP)

```
nmap -sUV -Pn -n -p53,67 --reason --packet-trace -oA hhs_5_udp_03
hackerhighschool.org


Starting Nmap 6.00 ( http://nmap.org ) at 2012-06-23 04:44 CEST
SENT (0.1730s) UDP 192.168.100.53:62664 > 216.92.116.13:53 ttl=48
id=23048 iplen=40
SENT (0.1731s) UDP 192.168.100.53:62664 > 216.92.116.13:67 ttl=48
id=53183 iplen=28
RCVD (0.4227s) ICMP 216.92.116.13 > 192.168.100.53 Port unreachable
(type=3/code=3) ttl=54 id=20172 iplen=56
SENT (2.4252s) UDP 192.168.100.53:62665 > 216.92.116.13:67 ttl=50
id=39909 iplen=28
NSOCK (3.8460s) UDP connection requested to 216.92.116.13:67 (IOD #1)
EID 8
NSOCK (3.8460s) Callback: CONNECT SUCCESS for EID 8 [216.92.116.13:67]
Service scan sending probe RPCCheck to 216.92.116.13:67 (udp)
...and 80 more packets...
Nmap scan report for hackerhighschool.org (216.92.116.13)
Host is up, received user-set (0.25s latency).
PORT  STATE      SERVICE REASON    VERSION
53/udp closed    domain port-unreach
67/udp open|filtered dhcps  no-response
```

Esta vez no hemos tenido suerte, pues hemos obtenido los mismos resultados. Un buen hacker podría intentarlo manualmente con paquetes UDP, o con el cliente adecuado para el puerto 67 estándar. Hasta aquí hemos utilizado el escaneo de servicios, un paso más en la identificación de servicios. Aprende a reconocer los servicios habituales en tu máquina, haz algunos ejercicios y después continua con la captura de cabeceras (0).

Ejercicios

5.25 Vete a la página http://nmap.org, descarga e instala la última versión de *nmap* para tu sistema operativo.

5.26 Repite todos los escaneos de esta sección utilizando más puertos. Recuerda que necesitas el comando "sudo" en los sistemas Linux o tener derechos de administrador en Windows.

5.27 Elabora una tabla de referencia para todas las técnicas de escaneo,

mapeando estado, razón y la respuesta real del objetivo (*packet-trace*).

Detección del Sistema Operativo

Conocer los servicios es importante para tomar la huella digital (*fingerprinting*) de una máquina. Además *nmap* puede ayudarnos utilizando otras opciones como -A para todos los escaneos y -O para detección de Sistema Operativo, utilizando los puertos por defecto:

```
sudo nmap -A -Pn -n --reason -oA hhs_5_all hackerhighschool.org


Starting Nmap 6.00 ( http://nmap.org ) at 2012-06-23 05:38 CEST
Nmap scan report for hackerhighschool.org (216.92.116.13)
Host is up, received user-set (0.21s latency).
Not shown: 971 closed ports
Reason: 971 resets
PORT    STATE  SERVICE     REASON   VERSION
21/tcp  open   ftp         syn-ack  NcFTPd
22/tcp  open   ssh         syn-ack  OpenSSH 5.9 (protocol 2.0)
| ssh-hostkey: 1024 cd:27:c2:bf:ad:35:e5:67:e0:1b:cf:ef:ac:2b:18:9a
(DSA)
|_1024 17:83:c5:8a:7a:ac:6c:90:48:04:0b:e5:9c:e5:4d:ab (RSA)
25/tcp  filtered smtp      no-response
26/tcp  open   tcpwrapped  syn-ack
80/tcp  open   http        syn-ack  Apache httpd 2.2.22
|_http-title: Hacker Highschool - Security Awareness for Teens
110/tcp  open   pop3       syn-ack  Dovecot pop3d
|_pop3-capabilities: USER CAPA UIDL TOP OK(K) RESP-CODES PIPELINING
STLS SASL(PLAIN LOGIN)
111/tcp  filtered rpcbind     no-response
113/tcp  open   tcpwrapped  syn-ack
143/tcp  open   imap        syn-ack  Dovecot imapd
|_imap-capabilities: LOGIN-REFERRALS QUOTA AUTH=PLAIN LIST-STATUS
CHILDREN CONTEXT=SEARCH THREAD=REFERENCES UIDPLUS SORT IDLE
MULTIAPPEND CONDSTORE ESEARCH Capability UNSELECT AUTH=LOGINA0001
IMAP4rev1 ID WITHIN QRESYNC LIST-EXTENDED SORT=DISPLAY THREAD=REFS
STARTTLS OK completed SEARCHRES ENABLE I18NLEVEL=1 LITERAL+ ESORT
SASL-IR NAMESPACE
161/tcp  filtered snmp      no-response
179/tcp  filtered bgp       no-response
306/tcp  open   tcpwrapped  syn-ack
443/tcp  open   ssl/http    syn-ack  Apache httpd 2.2.22
```

```
| ssl-cert: Subject: commonName=www.isecom.org/organizationName=ISECOM
-      The      Institute      for      Security      and      Open
Methodologies/stateOrProvinceName=New York/countryName=US

| Not valid before: 2010-12-11 00:00:00

|_Not valid after: 2013-12-10 23:59:59

|_http-title: Site doesn't have a title (text/html).

|_sslv2: server supports SSLv2 protocol, but no SSLv2 cyphers

465/tcp  open   ssl/smtp    syn-ack   Postfix smtpd

|_smtp-commands: kunatri.pair.com, PIPELINING, SIZE 41943040, ETRN,
AUTH PLAIN LOGIN, AUTH=PLAIN LOGIN, ENHANCEDSTATUSCODES, 8BITMIME,
DSN,

|   ssl-cert:   Subject:   commonName=*.pair.com/organizationName=pair
Networks, Inc./stateOrProvinceName=Pennsylvania/countryName=US

| Not valid before: 2012-01-10 00:00:00

|_Not valid after: 2015-01-09 23:59:59

543/tcp  open   tcpwrapped   syn-ack

544/tcp  open   tcpwrapped   syn-ack

587/tcp  open   smtp      syn-ack   Postfix smtpd

|_smtp-commands: kunatri.pair.com, PIPELINING, SIZE 41943040, ETRN,
STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN,

|   ssl-cert:   Subject:   commonName=*.pair.com/organizationName=pair
Networks, Inc./stateOrProvinceName=Pennsylvania/countryName=US

| Not valid before: 2012-01-10 00:00:00

|_Not valid after: 2015-01-09 23:59:59

646/tcp  filtered ldp       no-response

800/tcp  filtered mdbs_daemon   no-response

993/tcp  open   ssl/imap    syn-ack   Dovecot imapd

|   ssl-cert:   Subject:   commonName=*.pair.com/organizationName=pair
Networks, Inc./stateOrProvinceName=Pennsylvania/countryName=US

| Not valid before: 2012-01-10 00:00:00

|_Not valid after: 2015-01-09 23:59:59

|_sslv2: server supports SSLv2 protocol, but no SSLv2 cyphers

|_imap-capabilities:   LOGIN-REFERRALS   completed   OK   SORT=DISPLAY
Capability   UNSELECT   AUTH=PLAIN   AUTH=LOGINA0001   IMAP4rev1   QUOTA
CONDSTORE LIST-STATUS ID SEARCHRES WITHIN CHILDREN LIST-EXTENDED ESORT
ESEARCH    QRESYNC    CONTEXT=SEARCH    THREAD=REFS    THREAD=REFERENCES
I18NLEVEL=1  UIDPLUS  NAMESPACE  ENABLE  SORT  LITERAL+  IDLE  SASL-IR
MULTIAPPEND

995/tcp  open   ssl/pop3    syn-ack   Dovecot pop3d

|_sslv2: server supports SSLv2 protocol, but no SSLv2 cyphers

|_pop3-capabilities: OK(K) CAPA RESP-CODES UIDL PIPELINING USER TOP
SASL(PLAIN LOGIN)
```

```
|   ssl-cert:   Subject:   commonName=*.pair.com/organizationName=pair
Networks, Inc./stateOrProvinceName=Pennsylvania/countryName=US

| Not valid before: 2012-01-10 00:00:00

|_Not valid after: 2015-01-09 23:59:59

2105/tcp open   tcpwrapped   syn-ack

6667/tcp filtered irc       no-response

7000/tcp filtered afs3-fileserver no-response

7001/tcp filtered afs3-callback  no-response

7007/tcp filtered afs3-bos    no-response

7777/tcp filtered cbt       no-response

9000/tcp filtered cslistener   no-response

31337/tcp filtered Elite      no-response

Device type: general purpose|firewall|specialized|router

Running (JUST GUESSING): FreeBSD 6.X|7.X|8.X (98%), m0n0wall FreeBSD
6.X (91%), OpenBSD 4.X (91%), VMware ESX Server 4.X (90%), AVtech
embedded (89%), Juniper JUNOS 9.X (89%)

OS    CPE:    cpe:/o:freebsd:freebsd:6.3    cpe:/o:freebsd:freebsd:7.0
cpe:/o:freebsd:freebsd:8.1                    cpe:/o:m0n0wall:freebsd
cpe:/o:openbsd:openbsd:4.0                     cpe:/o:vmware:esxi:4.1
cpe:/o:m0n0wall:freebsd:6 cpe:/o:juniper:junos:9

Aggressive OS guesses: FreeBSD 6.3-RELEASE (98%), FreeBSD 7.0-RELEASE
(95%), FreeBSD 8.1-RELEASE (94%), FreeBSD 7.1-PRERELEASE 7.2-STABLE
(94%), FreeBSD 7.0-RELEASE - 8.0-STABLE (92%), FreeBSD 7.1-RELEASE
(92%), FreeBSD 7.2-RELEASE - 8.0-RELEASE (91%), FreeBSD 7.0-RC1 (91%),
FreeBSD 7.0-STABLE (91%), m0n0wall 1.3b11 - 1.3b15 FreeBSD-based
firewall (91%)

No exact OS matches for host (test conditions non-ideal).

Network Distance: 12 hops

Service Info: Host: kunatri.pair.com; OS: Unix


TRACEROUTE (using port 1723/tcp)

HOP RTT    ADDRESS

[...]

8  94.98 ms 89.221.34.153

9  93.70 ms 89.221.34.110

10 211.60 ms 64.210.21.150

11 ...

12 209.28 ms 216.92.116.13


OS and Service detection performed. Please report any incorrect
results at http://nmap.org/submit/ .
```

```
Nmap done: 1 IP address (1 host up) scanned in 57.94 seconds
```

Al utilizar -A se pueden ver más datos. Hay *plugins* especializados que obtienen más información de un servidor, intentan averiguar el SO y emplean una variante *traceroute* que aplica métodos distintos de los habituales de *traceroute* o *tracert*. Para averiguar el SO, cuantos más puertos mejor.

Ejercicios

5.28 Escanea tu propia máquina con *nmap*. ¿Es válida la detección del SO?

5.29 Utiliza la opción *traceroute* de *nmap* utilizando distintos puertos:

```
nmap -n -Pn --traceroute --version-trace -p80 hackerhighschool.org
```

5.30 ¿Hay diferencias entre utilizar *nmap* con *traceroute* con distintos puertos y utilizar *tracert* o *traceroute* de tu SO?

5.31 Investiga la huella digital de la pila TCP/IP. ¿Cómo puedes hacerlo? ¿Es una investigación a prueba de suplantación (*spoof-proof*)?

Utilizando *scripts*

Nmap dispone, además, de muchos *scripts* muy prácticos para realizar escaneos. Puedes utilizar la opción `–script script-name` para cargarlos. Un *script* interesante es *ipidseq*, pues realiza un fingerprint de IP incremental. Este *script* se utiliza para localizar hosts para el escaneo *Idle (*-sI), utilizando una implementación de IP que busca host zombis para escanear otros objetivos.

```
nmap --script ipidseq -oA hhs_5_ipidseq hackerhighschool.org
```

```
Starting Nmap 6.00 ( http://nmap.org ) at 2012-06-23 05:47 CEST
Nmap scan report for hackerhighschool.org (216.92.116.13)
Host is up (0.23s latency).
rDNS record for 216.92.116.13: isecom.org
Not shown: 971 closed ports
```

Ejercicios

5.32 Investiga sobre las técnicas de *idle scan*. ¿Qué son y como se aplican?

Conclusión

Saber dónde buscar y qué buscar es solo una parte de la batalla por la seguridad. Las redes están siendo continuamente supervisadas, analizadas, observadas e instigadas. Si la red que estás protegiendo no está siendo espiada, entonces no estás utilizando las herramientas apropiadas para detectar ese comportamiento. Si la red en la que te estás introduciendo no está siendo vigilada, puede (sólo puede) que te salgas con la tuya en tu escaneo. Como experto en ciberseguridad, deberías conocer al dedillo los sistemas que estás protegiendo o analizando. Necesitas conocer dónde se encuentran las debilidades e igualmente los puntos fuertes, no importa de que lado estés.

Hoy en día, investigar datos sobre un servidor, es decir el sistema operativo y los puertos abiertos, no es suficiente. Una amenaza persistente avanzada o APT (*Advanced Persistent Threat*) tratará de aprender tanto como pueda sobre tu red. Esta información incluye:

- marca y modelo de cortafuegos, versión del *firmware* y si existen parches del software
- autenticación de conexiones remotas, privilegios de acceso y procesos
- otros servidores que se conectan a la red, incluidos Email, HTML, *back-up*, redundantes, *off-site,* servicios alquilados o externalizados e incluso empresas contratistas que pueden haber utilizado tu red o que la están usando en la actualidad
- impresoras, fax, fotocopiadoras, *routers* inalámbricos y conexiones de red en la sala de espera de tu empresa
- dispositivos portátiles como tabletas, teléfonos inteligentes (*smartphones*), marcos de fotos digitales y cualquier cosa que pueda conectarse a la red

A pesar de que en esta lección se han repasado muchos temas, la identificación de sistemas cubre muchísimo más. Por las redes fluye bastante información que identifica partes de cada dispositivo. Cada dispositivo en la red puede ser explotado y por tanto usado como punto de acceso por un atacante. Abordar este reto amenazante requiere algo más que sólo software. Investiga sobre tu propio equipo y aprende tanto como puedas. Amortizarás este conocimiento.

Today's teens are in a world with major communication and productivity channels open to them and they don't have the knowledge to defend themselves against the fraud, identity theft, privacy leaks and other attacks made against them just for using the Internet. This is the reason for Hacker Highschool.

The Hacker Highschool project is the development of security and privacy awareness learning materials for junior high and high school students.

Hacker Highschool is a set of lessons and a practical means of making hackers. Beyond just providing cybersecurity awareness and critical Internet skills, we need to teach the young people of today how to be resourceful, creative, and logical, traits synonymous with hackers. The program contains free security and privacy awareness teaching materials and back-end support for teachers of accredited junior high, high schools, and home schooling. There are multiple workbooks available in multiple languages. These are lessons that challenge teens to be as resourceful as hackers, including safe Internet use, web privacy, researching on the internet, avoiding viruses and Trojans, legalities and ethics, and more.

The HHS program is developed by ISECOM, a non-profit, open-source research group focused on security awareness and professional security development and accreditation.

Hacker Highschool
SECURITY AWARENESS FOR TEENS

ISECOM