ESCUELA SUPERIOR POLITECNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

“Investigación Forense: Estudio para determinar los métodos usados en la Intrusión del caso del Banco JBR”

INFORME DE MATERIA DE GRADUACIÓN

Previa a la obtención del Título de:
“LICENCIATURA EN REDES Y SISTEMAS OPERATIVOS”

Presentada por:
DIEGO JAIME DONOSO PAYNE
DANIEL ADRIÁN QUIÑÓNEZ APRAEZ

GUAYAQUIL - ECUADOR

AÑO
2013

AGRADECIMIENTO

Aprovecho tener esta oportunidad para demostrar los mucho que estoy agradecido con mi madre, aunque no se pueda describir a través de palabras todo el amor que me brinda, ella siempre es y será la luz en mi camino. A mi padre y mis hermanas por estar siempre a mi lado confiando y brindándome su apoyo incondicional. A todas las personas que me han ayudado en este difícil y largo viaje.

Gracias

Diego Donoso Payne

A Dios por bendecirme y brindarme entendimiento y paciencia. A mis padres que me han apoyado siempre en este largo camino. A mi abuela que es mi sustento y estado de ánimo en este gran objetivo. A mi novia por ser una persona que me demuestra apoyo incondicional. A todos mis familiares y amigos preocupados de mi bienestar.

Daniel Quiñónez Apraez

DEDICATORIA

Este trabajo está dedicado a la persona que es mi fuente de admiración, orgullo y respeto, que me dio la vida y ha seguido formándome con valores y principios “Mi mamá”.

Diego Donoso Payne

Este gran trabajo está dedicado a mis padres y especialmente a mi abuela que es una persona sincera, honesta y con sus oraciones he alcanzado este gran objetivo.

Daniel Quiñónez Apraez

TRIBUNAL DE SUSTENTACIÓN

_____

Ing. Karina Astudillo

PROFESOR DE MATERIA DE GRADUACIÓN

_____

Ing. Miguel Molina

PROFESOR DELEGADO POR EL DECANO DE LA FACULTAD

DECLARACION EXPRESA

"La responsabilidad del contenido de este Informe de Materia de Graduación nos corresponden exclusivamente; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITECNICA DEL LITORAL".

(Reglamento de Graduación de la ESPOL)

Daniel Quiñonez Apraez

Diego Donoso Payne

RESUMEN

El trabajo investigativo en esta tesis, nos demuestra como un ataque informático puede llegar a dañar tantas áreas de una organización y así mismo también pretende ser una guía de aprendizaje para poder implantar las costumbres de realizar auditorías cada cierto tiempo, de esta manera se podrán evitar este tipo de ataques.

En la primera sección nos introducimos profundamente en el caso a investigar, dándoles a conocer los protagonistas y antagonistas de esta intrusión.

En la segunda sección daremos a conocer todos los métodos y fundamentos en los que se basa nuestra investigación. Explicando cada concepto y las normas que se deben seguir para realizar una investigación forense de confianza.

En el tercer capítulo se dan a conocer cuáles son las herramientas más apropiadas para realizar una investigación de este tipo, estas herramientas nos ayudarán a efectuar desde un simple proceso de recolección de evidencias hasta la presentación de un análisis.

El cuarto capítulo es la parte más importante de nuestra investigación, aquí aplicaremos los procedimientos de análisis previamente demostrados, para determinar con razonabilidad como se efectúo el ataque informático.

En nuestra última sección se explicará de manera resumida los daños que el atacante logró por medio de su intrusión, también daremos a conocer que métodos se deberían aplicar para evitar este tipo de ataques a futuro.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

INDICE DE TABLAS

INTRODUCCIÓN

Cada día automatizamos más los procesos cotidianos gracias al avance tecnológico y así mismo los procesos ilegales también son facilitados. Este hecho ha creado la necesidad de que las autoridades deban especializarse y capacitarse en nuevas áreas en donde las tecnologías de Información y Comunicación se conviertan en herramientas necesarias en auxilio de la Justicia y la persecución de delito y delincuente.

La informática forense está adquiriendo una gran importancia dentro del área de la información electrónica, esto debido al aumento del valor de la información y/o al uso que se le da a ésta, al desarrollo de nuevos espacios donde es usada (por Ej. El Internet), y al extenso uso de computadores por parte de las compañías de negocios tradicionales (por Ej. bancos). Es por esto que cuando se realiza un crimen, muchas veces la información queda almacenada en forma digital. Sin embargo, existe un gran problema, debido a que los computadores guardan la información de información forma tal que no puede ser recolectada o usada como prueba utilizando medios comunes, se deben utilizar mecanismos

diferentes a los tradicionales. Es de aquí que surge el estudio de la computación forense como una ciencia relativamente nueva.

Resaltando su carácter científico, tiene sus fundamentos en las leyes de la física, de la electricidad y el magnetismo. Es gracias a fenómenos electromagnéticos que la información se puede almacenar, leer e incluso recuperar cuando se creía eliminada.

La informática forense, aplicando procedimientos estrictos y rigurosos puede ayudar a resolver grandes crímenes apoyándose en el método científico, aplicado a la recolección, análisis y validación de todo tipo de pruebas digitales.

CAPÍTULO I

ANTECEDENTES Y JUSTIFICACIÓN

En este capítulo, daremos a conocer el estado del problema presentado en la institución bancaria JBR, para esto procederemos a explicar cuáles fueron las primeras sospechas de que el Banco JBR haya sufrido un ataque

1.1. Antecedentes

Banco JBR es una grande, muy respetada institución financiera y es usada por muchos usuarios debido a su confianza en sus servicios. JBR tiene un sitio Web para que los clientes puedan comprobar la actividad de su cuenta, pagar las facturas por vía electrónica, y ejecutar otras tareas financieras. Para ayudar a solucionar adecuadamente las quejas del cliente, JBR brinda un servicio Help Desk que es un conjunto de máquinas que utiliza a la hora

de investigar los errores en su software en línea. Después de hacer algunas preguntas clave, se entera de que estas máquinas no están protegidas por un firewall. El personal de IT mantiene a estos sistemas de escritorio de los clientes de la simulación en un "ambiente abierto" para reflejar la configuración que el cliente puede operar en su dial-up o conexión de banda ancha. El conjunto de las máquinas contiene de todo, desde Linux a FreeBSD, Apple OS X, Windows 2000, Windows XP y mucho más. Cada máquina cuenta con varios programas instalados para emular de todas las maneras diferentes lo que un cliente puede tener configurado en sus computadoras cuando experimenta un error y solicita el servicio Help Desk del Banco.

1.2. Justificativos del Trabajo

El director de IT de JBR indica que en octubre, 1, 2003, uno de los empleados de Help Desk encontró un archivo extraño en uno de los sistemas de simulación del escritorio de los clientes. El accedió a la estación de trabajo Windows 2000 (en la dirección IP 103.98.91.41) y notó el archivo update.exe localizado en C: \ que fue de cero bytes de longitud. Este archivo no fue colocado en la máquina durante la práctica comercial normal,

por lo que el empleado del Help Desk llama a Seguridad Corporativa. La política del Banco de respuesta a incidentes indica que la máquina debe ser investigada utilizando un proceso de respuesta en vivo, que recoge los datos volátiles que pueden ser perdidos si el ordenador está apagado. La dirección IP de la estación de trabajo forense era 103.98.91.200. Después de que la respuesta en vivo había sido completada, el equipo del Banco JBR de respuesta a incidentes adquirió una duplicación forense con la utilidad dd. El personal del Help Desk estaba llevando a cabo una resolución de problemas de red durante el momento que se sospecha que la intrusión sucedió y puede haber recogido tráfico de red de interés.

1.3. Objetivos

Los objetivos del presente trabajo de Investigación Forense son:

- Investigar la información juntada por el equipo de Respuesta a Incidentes.
- Aplicar metodologías, herramientas y técnicas que permitan identificar las causas del ataque informático de la empresa.

- Conocer los mecanismos que utilizó el atacante para realizar la intrusión.
- Identificar al causante del ataque informático.
- Dar a conocer a la institución bancaria si el ataque fue lo suficientemente grave para tener que presentar un reporte a sus clientes en caso de que su información haya logrado ser comprometida.

1.4. Alcance

El presente trabajo delimita su alcance exclusivamente al caso analizado en la Institución bancaria JBR, por la magnitud del problema informático suscitado. Pretendemos presentar un reporte de los daños causados por esta intrusión, así como también presentar un reporte donde se especifique si la información confidencial de los clientes de esta institución bancaria fue manipulada de alguna manera.

Es importante señalar que la investigación no pretende abarcar aspectos legales o temas que involucren la forma en que los sistemas judiciales deben manejar la evidencia que se pudiera aportar para resolver delitos que

involucren el uso de esta tecnología de comunicación personal. Esto queda como responsabilidad de los profesionales del derecho judicial. Sin embargo, sí considera los aspectos técnicos relacionados con la obtención y análisis de la evidencia, así como los procedimientos para el manejo responsable y con garantías de custodia de evidencia digital.

1.5. Equipo de Consultoría

Esta investigación forense será llevada a cabo de manera profesional y confidencial por los consultores el Lcdo. Diego Donoso Payne y el Lcdo. Daniel Quiñonez.

CAPÍTULO II

MARCO TEÓRICO

En el presente capítulo se darán a conocer los diferentes temas relacionados con la computación forense, para que el lector comprenda la evolución de los diferentes ataques informáticos que se pueden suscitar en empresas de diferentes sectores comerciales y más en las que manejan información confidencial como lo son las instituciones bancarias, y así conseguir el entendimiento del porque las organizaciones deben revisar sus controles periódicamente con la finalidad de determinar sus seguridades tanto lógicas como físicas y además mejorar sus procesos.

2.1. Informática Forense

Según el FBI[1], la informática forense es la ciencia de adquirir, preservar, obtener y presentar datos que han sido procesados electrónicamente y guardados en un medio computacional.

La informática forense hace entonces su aparición como una disciplina auxiliar de la justicia moderna, para enfrentar los desafíos y técnicas de los intrusos informáticos, así como garante de la verdad alrededor de la evidencia digital que se pudiese aportar en un proceso[2].

Desde 1984, el Laboratorio del FBI y otras agencias que persiguen el cumplimiento de la ley empezaron a desarrollar programas para examinar evidencia computacional.

Dentro de lo forense encontramos varias definiciones (Martines Jeimy, 2006):

Computación forense (computer forensics)

Entendemos por disciplina de las ciencias forenses, que considerando las tareas propias asociadas con la evidencia, procura descubrir e interpretar la información en los medios

[1] (Noblett, 2000)
[2] (Martines Jeimy, 2006)

informáticos para establecer los hechos y formular las hipótesis relacionadas con el caso; o como la disciplina científica y especializada que entendiendo los elementos propios de las tecnologías de los equipos de computación ofrece un análisis de la información residente en dichos equipos.

Forensia en redes (network forensics)

Es un escenario aún más complejo, pues es necesario comprender la manera como los protocolos, configuraciones e infraestructuras de comunicaciones se conjugan para dar como resultado un momento específico en el tiempo y un comportamiento particular.

Esta conjunción de palabras establece un profesional que entendiendo las operaciones de las redes de computadores, es capaz, siguiendo los protocolos y formación criminalística, de establecer los rastros, los movimientos y acciones que un intruso ha desarrollado para concluir su acción. A diferencia de la definición de computación forense, este contexto exige capacidad de correlación de evento, muchas veces disyuntos y aleatorios, que en equipos particulares, es poco frecuente.

Forensia digital (digital forensics)

Forma de aplicar los conceptos, estrategias y procedimientos de la criminalística tradicional a los medios informáticos especializados, con el fin de apoyar a la administración de justicia en su lucha contra los posibles delincuentes o como una disciplina especializada que procura el esclarecimiento de los hechos (¿quién?, ¿cómo?, ¿dónde?, ¿cuándo?, ¿por qué?) de eventos que podrían catalogarse como incidentes, fraudes o usos indebidos bien sea en el contexto de la justicia especializada o como apoyo a las acciones internas de las organizaciones en el contexto de la administración de la inseguridad informática.

2.1.2 Importancia de la Informática Forense

"High-tech crime is one of the most important priorities of the Department of Justice"[3] (“El crimen de Alta Tecnología es una las prioridades más importantes del Departamento de Justicia”). Con esta frase podemos ver cómo poco a poco los crímenes informáticos, su prevención, y procesamiento se vuelven cada vez más importantes.

[3] (Reno, 1996)

Esto es respaldado por estudios sobre el número de incidentes reportados por las empresas debido a crímenes relacionados con la informática[4].

Sin embargo, la importancia real de la informática forense proviene de sus objetivos.

2.1.3 Objetivos de la Informática Forense

La informática forense tiene 3 objetivos, a saber:

- La compensación de los daños causados por los criminales o intrusos.
- La persecución y procesamiento judicial de los criminales.
- La creación y aplicación de medidas para prevenir casos similares.

Estos objetivos son logrados de varias formas, entre ellas, la principal es la recolección de evidencia.

[4] (CERT/CC, 1988-2006)

2.1.4 Usos de la Informática Forense

Existen varios usos de la informática forense, muchos de estos usos provienen de la vida diaria, y no tienen que estar directamente relacionados con la informática forense[5]:

Prosecución Criminal: Evidencia incriminatoria puede ser usada para procesar una variedad de crímenes, incluyendo homicidios, fraude financiero, tráfico y venta de drogas, evasión de impuestos o pornografía infantil.

Litigación Civil: Casos que tratan con fraude, discriminación, acoso, divorcio, pueden ser ayudados por la informática forense.

Investigación de Seguros: La evidencia encontrada en computadores, puede ayudar a las compañías de seguros a disminuir los costos de los reclamos por accidentes y compensaciones.

Temas corporativos: Puede ser recolectada información en casos que tratan sobre acoso sexual, robo, mal uso o apropiación de información confidencial o propietaria, o aún de espionaje industrial.

Mantenimiento de la ley: La informática forense puede ser usada en la búsqueda inicial de órdenes judiciales, así como en la búsqueda de

[5] (López, Amaya, & León, 2001)

información una vez se tiene la orden judicial para hacer la búsqueda exhaustiva.

2.1.5 Fases de la Informática Forense

La Policía Nacional, ante el manejo de evidencias sobre un crimen o delito informático cometido, deberá actuar como cualquier proceso criminal, el primer paso es asegurara la escena del delito restringiendo el acceso a la misma para no modificar la evidencia. Los peritos que manejen el caso deberán poseer conocimientos sobre las metodologías del análisis forense informático que se deben aplicar según el caso. Las fases para mantener la cadena de custodia intacta se explicarán en el siguiente diagrama[6]:

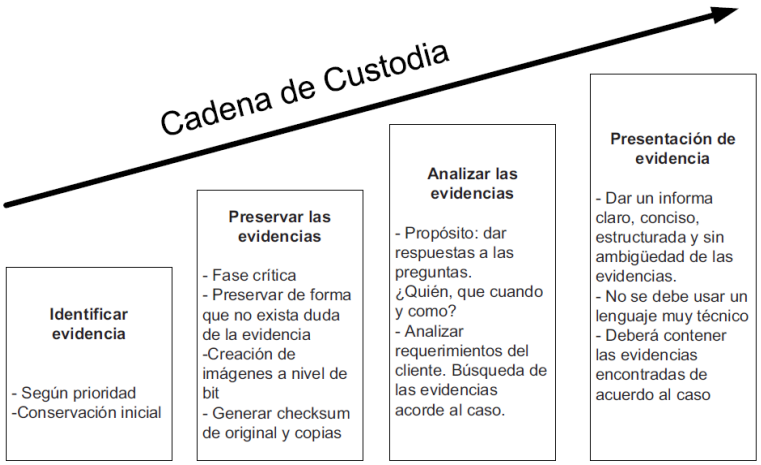


Figura 2-1: Cadena de Custodia

[6] (Galarza, 2010)

Existen modos de Análisis para la Informática Forense, estos son:

- **Análisis post-mortem**: se realiza con un equipo dedicado específicamente para fines forenses para examinar discos duros, datos o cualquier tipo de información recabada de un sistema que ha sufrido un incidente. En este caso, las herramientas de las que se puede disponer son aquellas que existan en el laboratorio destinado al análisis de discos duros, archivos de logs de firewalls, etc.
- **Análisis en caliente**: se lleva a cabo cuando un sistema presume que ha sufrido un incidente o está sufriendo un incidente de seguridad. En este caso, se debe emplear un CD con las herramientas de Respuesta ante Incidentes y Análisis Forense compiladas de forma que no realicen modificaciones en el sistema. Una vez hecho este análisis en caliente, y confirmado el incidente, se realiza el análisispost-mortem.
- **Cadena de custodia**: el es conjunto de pasos o procedimientos seguido para preservar la prueba digital que permita convertirla y usarla como evidencia digital en un proceso judicial. No existe un estándar reconocido públicamente.

La cadena de custodia debe:

- Reducir al máximo la cantidad de agentes implicados en el manejo o tratamiento de evidencias.
- Mantener la identidad de las personas implicadas desde la obtención hasta la presentación de las evidencias.
- Asegurar la firmeza de las evidencias.
- Registros de tiempos, firmados por los agentes, en los intercambios entre estos de las evidencias. Cada uno de ellos se hará responsable de las evidencias en cada momento.

La secuencia de la cadena de la evidencia debe seguir el siguiente orden:

- Recolección e identificación de evidencia.
- Análisis.
- Almacenamiento.
- Preservación.
- Transporte.
- Presentación en el juzgado.
- Retorno a su dueño.

La cadena de la evidencia muestra:

- Quién obtuvo la evidencia.
- Dónde y cuándo la evidencia fue obtenida.
- Quién protegió la evidencia.
- Quién ha tenido acceso a la evidencia.

2.1.6 Evidencia Digital

Casey define a la evidencia de digital como “cualquier dato que puede establecer que un crimen se ha ejecutado puede proporcionar un vínculo entre un crimen y su víctima o un crimen y su autor”[7].

La Evidencia Digital Es el conjunto de datos en formato binario, comprende los ficheros, su contenido o referencias a éstos (meta-datos= datos acerca de datos) que se encuentren en los soportes físicos o lógicos del sistema atacado, los mismos pueden ser recolectados y analizados con herramientas y técnicas especiales.

Se debe localizar los dispositivos donde podemos encontrar evidencias, ya que muchas veces la información que directa o indirectamente se relaciona con esta conducta criminal queda almacenada de forma digital dentro de estos Sistemas Informáticos.

[7] (CASEY, 2004)

En el mundo digital dado la facilidad de realizar copias exactas de documentos, generalmente se considera la evidencia digital cómo evidencia demostrativa.

Existe un RFC 3227 - “Guidelines for Evidence Collection and Archiving” que establece normas para la recopilación y almacenamiento de evidencias[8].

2.1.7 Clasificación de la Evidencia Digital

Cano clasifica la evidencia digital que contiene texto en 3 categorías[9]:

Registros generados por computador: Estos registros son aquellos, que como dice su nombre, son generados como efecto de la programación de un computador. Los registros generados por computador son inalterables por una persona. Estos registros son llamados registros de eventos de seguridad (logs) y sirven como prueba tras demostrar el correcto y adecuado funcionamiento del sistema o computador que generó el registro.

- **Registros no generados** sino simplemente almacenados por o en computadores: Estos registros son aquellos

[8] (Brezinski & Killalea, 2002)
[9] (Cano Martines, Mosquera González, & Certain Jaramillo, 2005)

generados por una persona, y que son almacenados en el computador, por ejemplo, un documento realizado con un procesador de palabras. En estos registros es importante lograr demostrar la identidad del generador, y probar hechos o afirmaciones contenidas en la evidencia misma. Para lo anterior se debe demostrar sucesos que muestren que las afirmaciones humanas contenidas en la evidencia son reales.

- **Registros híbridos** que incluyen tanto registros generados por computador como almacenados en los mismos: Los registros híbridos son aquellos que combinan afirmaciones humanas y logs. Para que estos registros sirvan como prueba deben cumplir los dos requisitos anteriores.

2.1.8 Recolección de Evidencias

La recopilación de evidencias permite determinar el método de entrada al sistema, la actividad de los intrusos, su identidad y origen, para todo ello se debe poseer mucha precaución para evitar alterar las evidencias durante el proceso de recolección.

La recolección de evidencia, varía de país en país, y por lo tanto, un análisis exacto y completo está fuera de los límites. Sin embargo, se presentan guías básicas que pueden ayudar a cualquier investigador forense:

La IOCE[10] (Organización Internacional de Evidencias en Computadora) define cinco puntos principales para el manejo y recolección de evidencia digital[11]:

1. Al recolectar evidencia digital, las acciones tomadas no deben cambiar por ningún motivo esta evidencia.
2. La persona que tenga acceso a evidencia digital original, deberá ser un profesional forense.
3. Toda la actividad referente a la recolección, el acceso, almacenamiento o a la transferencia de la evidencia digital, debe ser documentada completamente, preservada y disponible para la revisión.
4. Un individuo es responsable de todas las acciones tomadas con respecto a la evidencia digital mientras que ésta esté en su posesión.

[10] (IOCE, International Organization of Computer Evidence)
[11] (IOCE, Guidelines for the best practices in the forensic examination of digital technology, 2002)

5. Cualquier agencia que sea responsable de recolectar, tener acceso, almacenar o transferir evidencia digital es responsable de cumplir con estos principios.

2.1.9 Preservación de la Evidencia

La preservación se enfoca en resguardar los objetos que tengan valor como evidencia, de manera que estos permanezcan de forma completa, clara y verificable, es importante que cualquier examen que se lleve a cabo no genere cambios, en caso de suscitarse un cambio de manera inevitable, es esencial que se presente la razón por la que se dio tal acontecimiento, explicando el suceso detalladamente, posterior a ello debe ser registrado y justificado.

En esta fase se utiliza técnicas criptográficas como códigos de seguridad (función hash, checksums).

La fase de preservación interviene a lo largo de todo el proceso de investigación forense, la misma interactúa con las demás fases.

Las tareas que se deben seguir para preservar la evidencia digital son:

- Realizar dos copias de las evidencias obtenidas.

- Generar una suma de comprobación de la integridad de cada copia empleando función hash (MD5 o SHA1).
- Incluir las firmas obtenidas en la etiqueta de cada copia de la evidencia en el CD o DVD, incluir fecha, hora de la creación de la copia y el nombre de la misma.
- Proteger los dispositivos de factores externos como: cambios bruscos, temperatura o campos electromagnéticos, ya que pueden alterar la evidencia.

2.2 Análisis de la Evidencia Digital

AFD (Análisis Forense Digital)[12], es un conjunto de principios y técnicas que comprende el proceso de adquisición, conservación, documentación, análisis y presentación de evidencias digitales y que llegado el caso puedan ser aceptadas legalmente en un proceso judicial.

Este análisis se dará por concluido cuando se conozca cómo se produjo el ataque, quién o quienes lo llevaron a cabo, bajo qué circunstancias se produjo, cuál era el objetivo del ataque, qué daños causaron, etc.

[12] (Galarza, 2010)

Antes de realizar un análisis se debe tener en cuenta la siguiente información:

- Sistema operativo afectado.
- Inventario de software instalado en el equipo
- Tipo de hardware del equipo
- Accesorios y/o periféricos conectados al equipo
- Si posee firewall
- Si esta en el ámbito del DMZ (Zona desmilitarizada)
- Conexión a Internet.
- Configuración.
- Parches y/o actualizaciones de software
- Políticas de seguridad implementadas
- Forma de almacenamiento de la información (cifrada o no)
- Personas con permisos de acceso al equipo
- El computador esta dentro del DMZ
- Existe IDS
- Cuantos equipos en red se encuentran conectados.
- Listar usuarios conectados local y remotamente al sistema.

2.2.2 Preparación para el análisis

Es importante establecer estaciones de trabajo para realizar las distintas pruebas y estudios al surgir un caso, dependiendo del ataque o crimen cometido. Para ello se deberá:

- Clasificar el tipo de incidente
- Seguir el proceso inter-departamental para el manejo de las evidencias
- Identificar el tipo de dispositivo (computador, celular, memorias, PDA's, etc.) y las herramientas necesarias para su análisis.

En las estaciones se deberá operar de la siguiente manera:

- Montar imágenes de discos duros.
- Instalar Sistemas Operativos para realizar el estudio de evidencias.
- Realizar copias exactas del disco duro con la finalidad de realizar pruebas y verificaciones conforme surjan las hipótesis del ataque.

- En caso de no disponer de recursos se puede usar de software para crear una plataforma de trabajo con varias máquinas virtuales.17
- Se puede crear un entorno de trabajo hipotético con las copias obtenidas para realizar la emulación de los ataques.

2.2.3 Reconstrucción de la secuencia del ataque

Una vez establecida la estación de trabajo, el primer paso es crear una línea temporal de sucesos o timeline, para ello se deberá recopilar la siguiente información sobre los ficheros:

- Marcas de tiempo MACD (fecha y hora de modificación, acceso, creación y borrado).
- Ruta completa del fichero.
- Tamaño en bytes y tipo de fichero.
- Usuarios y grupos a quien pertenece el fichero.
- Permisos de acceso.
- Identificar si fue borrado o no.

Esta información es la que más tiempo lleva recopilar pero es el punto de partida para el análisis. Es importante preparar un script con la finalidad de automatizar el proceso de creación del timeline.

Luego de realizar lo antes mencionado se deberá:

- Ordenar los archivos por sus fechas MAC, esto se debe realizar debido a que los archivos tendrán la fecha de instalación del sistema operativo, por lo que un sistema que se instaló hace meses y que fue comprometido recientemente presentará en los ficheros nuevas fechas MAC muy distintas a las de los ficheros más antiguos.
- Buscar ficheros y directorios que han sido creados, modificados o borrados recientemente.
- Buscar instalaciones de programas posteriores a la del sistema operativo y que además se encuentren en rutas poco comunes.
- Buscar en lugares donde no se suele mirar, por ejemplo en los directorios temporales.
- Buscar los archivos de sistema modificados tras la instalación del sistema operativo, averiguar archivos ocultos donde se encuentran y que tipo son.

- Buscar archivos borrados, ya que pueden ser restos de logs y registros borrados por sus atacantes.
- En las imágenes realizadas a los discos duros se puede acceder al espacio residual que hay detrás de cada archivo ya que los mismos suelen almacenarse por bloques, de tal manera que se pueda leer zonas que el sistema operativo no ve.
- Recuperar archivos borrados, al momento de hacerlo, se deberá intentar recuperar su contenido y fecha de borrado.
- Examinar y las horas de manera más detallada de los ficheros logs y registros que ya se revisaron con la finalidad de encontrar una correlación entre eventos.
- Revisar el archivo de contraseñas, buscar la creación de usuarios y cuentas extrañas relacionar la hora de la creación de estas cuentas en caso de que existan con la hora en la que se inició el ataque al sistema.

2.2.4 Determinación de cómo se realizó el ataque

Una vez que se disponga de la cadena de acontecimientos que se han producido, se deberá determinar cuál fue la vía de entrada al sistema, averiguando qué vulnerabilidad o fallo de administración causó el agujero de seguridad y que herramientas utilizó el atacante para aprovecharse de tal brecha. Para ello se deberá:

- Combinar consultas a archivos logs, registro, claves cuentas de usuarios etc.
- Prestar atención a los servicios y procesos abiertos, puertos abiertos TCP/UDP y conexiones que ya se tomaron como evidencia volátil cuando el sistema estaba aún vivo.
- Examinar las circunstancias sospechosas encontradas al indicio del ataque, y buscar con ellas si son o no vulnerabilidades a través de Internet, ejemplo: (http://www.google.com), (http://www.cert.com), (MELBOURNE IT, 1999) (MELBOURNE IT, 1999).
- Si ya esta claro cual fue la vulnerabilidad del sistema, se deberá buscar en Internet algún exploit anterior a la fecha del ataque, que utilice esa vulnerabilidad.

- Reforzar cada una de las hipótesis mediante la fórmula causa-efecto.
- Utilizar la máquina “conejillo de Indias” con la finalidad de realizar las pruebas y exploits encontrados.
- Comprobar si la ejecución del exploit sobre una máquina igual a la atacada, genera los mismos eventos que se han encontrado entre las evidencias.

2.2.5 Identificación del autor o autores del incidente

Una vez que se determinó como se infiltraron al sistema, ahora se tiene que saber quién o quienes lo hicieron, para ello se deberá consultar nuevamente algunas evidencias volátiles que fueron recopiladas en la primera fase:

- Revisar las conexiones que se encontraban abiertas, que puertos y que direcciones IP las solicitaron, a más de ello se deberá buscar entre las entradas a los logs de conexiones.
- Indagar entre los archivos borrados que se han recuperado.

Para Identificar a los atacantes se debe realizar algunas averiguaciones como parte del proceso de identificación:

- Averiguar la dirección IP del atacante, para ello se deberá revisar detenidamente los registros de conexiones de red, los procesos y servicios que se encontraban a la escucha. Esta información se podría encontrar en fragmentos de las evidencias volátiles, la memoria virtual o archivos temporales y borrados, como restos de e-mail, conexiones fallidas, etc.
- Al adquirir la dirección IP sospechosa, se deberá comprobar en el registro (Nigel Titley) a quien pertenece, es importante considerar que no se puede sacar conclusiones prematuras, debido a que muchos atacantes falsifican la dirección IP con técnicas de spoofing . Los atacantes también pueden utilizar ordenadores zombis, éstos son comprometidos en primera instancia por el atacante y posteriormente son utilizados como instrumentos del ataque final sin que sus propietarios sepan que están siendo cómplices de tal hecho. Por ello, para identificar a su atacante tendrá que verificar y validar la dirección IP obtenida.
- Se puede emplear técnicas hacker para identificar al atacante ya que el equipo del mismo debe tener

inevitablemente un puerto que se encuentre esperando noticias o buscando víctimas. Averiguar el perfil del atacante, se puede encontrar con los siguientes tipos:

Hackers: personas con conocimientos en técnicas de programación, redes, Internet y sistemas operativos, sus ataques son en sentido ideológico y pacifista.

ScriptKiddies: son personas nuevas que han saltado a la escena de la delincuencia informática recientemente. Se trata de jóvenes que con unos conocimientos aceptables en Internet y programación emplean herramientas ya fabricadas por otros para realizar ataques y "ver qué pasa".

Profesionales: son personas con muchísimos conocimientos en lenguajes de programación, redes y su equipamiento (routers, firewall, etc.), Internet y sistemas operativos tipo UNIX.

2.2.6 Evaluación del impacto causado al sistema

El análisis forense ofrece la posibilidad de investigar qué es lo que han hecho los atacantes una vez que accedieron al sistema. Esto permitirá evaluar el ataque cometido a los equipos y realizar una

estimación del impacto causado. Generalmente se pueden dar dos tipos de ataques:

- **Ataques pasivos:** en los que no se altera la información ni la operación normal de los sistemas, limitándose el atacante solo a fisgonear.
- **Ataques activos:** en los que se altera la información, y en ocasiones seriamente, la capacidad de operación del sistema.

Se deberá tener en cuenta los efectos y el impacto que cause el ataque a sistemas, servidores de Bases de Datos, servidores WEB, cortafuegos, router con la finalidad de ser un aporte, presentando los daños encontrados, al personal de seguridad informática de la institución atacada o en último de los casos a la compañía de seguros de la misma.

2.2.7 Presentación de Evidencia Digital

Esta es la fase final de la investigación forense informática ya que se presentan los resultados y hallazgos del investigador. Tan pronto como el incidente haya sido detectado es importante tomar nota

sobre las actividades que se llevan a cabo, cada paso dado debe ser documentado y fechado desde que se descubre el incidente hasta finalizar la presentación, la misma debe ser entendible, creíble, confiable y convincente, es decir se deberá especificar claramente los procedimientos y las técnicas utilizadas para recolectar, preservar y filtrar la evidencia de tal manera que sea legalmente aceptable para ser presentadas a las entidades investigadoras y judiciales[13][14].

2.2.8 Utilización de formularios de registro del incidente

La aplicación de formularios ayudará a presentar una resolución del incidente mediante la presentación de informes uno Técnico y otro Ejecutivo. Estos formularios deben ser llenados por departamentos o entidades afectadas o por el equipo que gestiona el incidente, los formularios que se deben preparar son:

- Documento de custodia de la evidencia.
- Formulario de identificación de equipos y componentes.
- Formulario de incidencias.
- Formulario de publicación del incidente.

[13] (López, Amaya, & León, 2001)
[14] (López Delgado)

- Formulario de recogida de evidencias.
- Formulario de discos duros.

2.2.9 Informe Técnico

Este informe consiste en una exposición detallada del análisis efectuado. Deberá describir en profundidad la metodología, técnicas y hallazgos del equipo forense.

Deberá contener, al menos, los siguientes puntos:

- Antecedentes del incidente.
- Recolección de los datos.
- Descripción de la evidencia.
- Entorno del análisis.
    - Descripción de las herramientas.
- Análisis de la evidencia.
    - Información del sistema analizado.
        - Características del SO.
        - Aplicaciones.
        - Servicios.
        - Vulnerabilidades.
        - Metodología.

- Descripción de los hallazgos.
    - Huellas de la intrusión.
    - Herramientas usadas por el atacante.
    - Alcance que ha tenido el delito.
    - El origen del ataque
- Cronología del delito.
- Conclusiones.
- Recomendaciones específicas.
- Referencias.

2.2.10 Informe Ejecutivo

Este informe es un resumen del análisis efectuado a las evidencias digitales, el mismo deberá:

- Ser redactado en un lenguaje común que sea legible para cualquier persona.
- No ser escrito de manera técnica.
- Exponer los hechos más destacables de lo ocurrido en el sistema analizado.
- Constará de pocas páginas, entre tres y cinco,

- Deberá ser de interés para exponer lo sucedido a personal no especializado en sistemas informáticos, como el departamento de
- Recursos Humanos, Administración, e incluso algunos directivos.

En el mismo se debe describir:

- Motivos de la intrusión.
- Desarrollo de la intrusión.
- Resultados del análisis.
- Recomendaciones.

2.3 Definiciones Conceptuales

1.- Auditoría Forense: es una alternativa para combatir la corrupción, porque permite que un experto emita ante los jueces conceptos y opiniones de valor técnico, que le permiten a la justicia actuar con mayor certeza, especialmente en lo relativo a la vigilancia de la gestión fiscal.

2.- Delito Informático: Según María de la Luz Lima[15], dice que el "delito informático en un sentido amplio es cualquier conducta criminal que en

[15] (Lima, 2007)

su realización hace uso de la tecnología electrónica ya sea como método, medio o fin y que, en sentido estricto, el delito informático, es cualquier acto ilícito penal en el que las computadoras, sus técnicas y funciones desempeñan un papel ya sea con método, medio o fin".

3.-Fraude: Engaño, inexactitud, consistente, abuso de confianza, que produce o prepara un daño, generalmente material.

4.-Hallazgo: Es la recopilación de información especifica sobre una operación, actividad, organización, condición u otro asunto que se haya analizado y evaluado y que se considera de interés o utilidad para los funcionarios del organismo.

5.- Herramientas de la Auditoria Forense: Son artículos u objetos que ayuda a resolver un problema que puede ser de cualquier clase, técnico, labora, penal, etc.”

6.-Informe: Comunica a las autoridades pertinentes los resultados de la Auditoría. Los requisitos para la preparación del informe son claridad y simplicidad, importancia del contenido, respaldo adecuado, razonabilidad, objetividad entre otros.

7.- Metodología: Según el Diccionario, Método es el “modo de decir o hacer con orden una cosa”. Asimismo define el diccionario la palabra Metodología como “conjunto de métodos que se siguen en una investigación científica”. Esto significa que cualquier proceso cinético debe estar sujeto a una disciplina de proceso defina con anterioridad que llamaremos Metodología.

9.-Intrusión Informática: La Intrusión informática implica actividades criminales que no encuadran en las figuras tradicionales como robos, hurtos, falsificaciones, estafa, sabotaje, etc. Sin embargo, debe destacarse que el uso de las técnicas informáticas ha creado nuevas posibilidades del uso indebido de computadoras lo que ha propiciado a su vez la necesidad de regulación por parte del derecho.

CAPÍTULO III

HERRAMIENTAS

En este capítulo procederemos a explicar cada una de las herramientas que utilizaremos a lo largo de la investigación, esto quiere decir desde la adquisición de la evidencia hasta el momento q hallamos al causante de la intrusión informática

La presentación de cada una de las herramientas que se presentaran está basada en la siguiente clasificación orientada a la informática forense[16]:

- Herramientas para recolección de evidencias.
- Herramientas para el Monitoreo y/o Control de Computadores

[16] (López, Amaya, & León, 2001)

- Herramientas de Marcado de documentos.
- Herramientas de Hardware.

3.1 Herramientas para recolección de evidencias

Existen una gran cantidad de herramientas para recuperar evidencia. El uso de herramientas sofisticadas se hace necesario debido a:

- La gran cantidad de datos que pueden estar almacenados en un computador.
- La variedad de formatos de archivos, los cuales pueden variar enormemente, aún dentro del contexto de un mismo sistema operativo.
- La necesidad de recopilar la información de una manera exacta, y que permita verificar que la copia es exacta.
- Limitaciones de tiempo para analizar toda la información.
- Facilidad para borrar archivos de computadores.
- Mecanismos de encripción, o de contraseñas.

3.2 Herramientas para el Monitoreo de Computadores

Algunas veces se necesita información sobre el uso de los computadores, por lo tanto existen herramientas que monitorean el uso de las mismas, para poder recolectar información. Existen algunos programas simples como key loggers o recolectores de pulsaciones del teclado, que guardan información sobre las teclas que son presionadas, hasta otros que guardan imágenes de la pantalla que ve el usuario del computador, o hasta casos donde la máquina es controlada remotamente.

3.3 Herramientas de Marcado de documentos

Un aspecto interesante es el de marcado de documentos; en los casos de robo de información, es posible, mediante el uso de herramientas, marcar software para poder detectarlo fácilmente. La seguridad está centrada en la prevención de ataques. Algunos sitios que manejan información confidencial o sensitiva, tienen mecanismos para validar el ingreso, pero, debido a que no existe nada como un sitio 100% seguro, se debe estar preparado para cualquier tipo de incidentes.

3.4 Herramientas de Hardware

Debido a que el proceso de recolección de evidencia debe ser preciso y no debe modificar la información se han diseñado varias herramientas para ello.

3.5 Costos de herramientas forenses

Las herramientas que se presentan serán categorizadas según los siguientes factores:

- Software libre.
- Rendimiento y desempeño. Por medio de una evaluación = alta, media, baja.
- Costos.
- Confiabilidad ante la recuperación de evidencias.

Para realizar una investigación forense de manera eficiente el investigador deberás hacer uso de ciertas herramientas, porque además de automatizar tareas, también le ayudaran a secuenciar sus pasos y a documentar cada uno de ellos.

Tabla 1: Listado de Herramientas Forenses

Herramienta	Costo	Características	Plataformas	Evaluación
The Coroner's Toolkit	Libre	▯ Esta colección de programas sirve para realizar una 'autopsia' sobre sistemas UNIX después de que han 'muerto' completamente. ▯ El funcionamiento de este software se basa principalmente en la recogida de grandes cantidades de datos para proceder a su análisis posterior. Algunos de sus componentes son la herramienta 'ladrón de tumbas' (que captura información), los programas para detectar archivos 'muertos' o 'vivos', así como 'lázaro', que restaura archivos borrados, y otra herramienta que restaura claves criptográficas desde un proceso activo o desde algún archivo. ▯ Contiene varias herramientas importantes para el análisis forense. ▯ Aplicaciones importantes: ▯ **Grave-robber:** Una utilidad para capturar información sobre i-nodes, para ser procesada por el programa mactime del mismo toolkit. ▯ **Unrm y lazarus:** Herramientas para la recuperación de archivos borrados (logs, RAM, swap, etc.). Estas aplicaciones identifican y recuperan la información oculta en los sectores del disco duro. ▯ **Mactime:** El programa para visualizar los ficheros/directorios su timestamp MAC (Modification, Access, y Change). ▯ Se ejecutada cuando la evidencia encontrada, posee un Sistema Operativo (Linux)	FreeBSD 2-4.*, OpenBSD 2.*, BSD/OS 2-3.*, SunOS 4-5.* y Linux 2.*. SUN Solaris RedHat Linux	Media

Herramienta	Costo	Características	Plataformas	Evaluación
the Sleuth Kit y Autopsy	Libre	▯ Es una interfaz gráfica que trabaja en conjunto con la herramienta the sleuth kit utilizada para el análisis forense. ▯ Analiza discos y sistema de archivos (NTFS, FAT, UFS1/2, Ext2/3). ▯ Muestra el detalle de información sobre datos eliminados y estructuras del sistema de ficheros. ▯ Permite el acceso a estructuras de archivos y directorios de bajo nivel y eliminados ▯ Genera la línea temporal de actividad de los archivos (timestamp). ▯ Permite buscar datos dentro de las imágenes por palabras clave, ▯ Permite crear notas del investigador e incluso genera informes y muchas tareas. ▯ Es una colección de herramientas. ▯ Utiliza interfaz gráfica que facilita notablemente el trabajo. ▯ Se ejecutada cuando la evidencia encontrada, posee un Sistema Operativo (Windows/Linux) Descargas: (http://www.sleuthkit.org/autopsy/download.php)	Unix/Linux, Windows	Alta
Herramienta	**Costo**	**Características**	**Plataformas**	**Evaluación**
Mac-Robber	Libre	▯ Recopila información de ficheros localizados en un sistema de ficheros montado (mounted). ▯ Los datos obtenidos pueden ser utilizados por la herramienta mactime, contenida en el Sleuth Kit, para elaborar una línea temporal de actividad de los ficheros. ▯ Está basado en grave-robber (contenido en TCT) explicado anteriormente.	Linux	Bajo

		- Requiere que el sistema de ficheros esté montado por el sistema operativo, a diferencia de otras herramientas como el Sleuth Kit que procesan el sistema de ficheros ellos mismos. Por lo tanto, mac-robber no recopila datos de ficheros eliminados o ficheros que están ocultos. - Modificará los tiempos de acceso a directorios que están montados con permisos de escritura.		
Herramienta	**Costo**	**Características**	**Plataformas**	**Evaluación**
FoundStone Forensic ToolKit	Libre	Herramientas que forman parte: - Pasco: Herramienta para analizar la actividad realizada con el navegador web Internet - Explorer de MS. - Galleta: Examina el contenido del fichero de cookies de IE. - Rifiuti: Examina el contenido del fichero INFO2 de la papelera de reciclaje de Windows. - Vision: Lista todas los puertos TCP y UDP en escucha (abiertos) y los mapea a las aplicaciones o procesos que se encuentran detrás. - Forensic Toolkit: Es una suite de herramientas para el análisis de las propiedades de ficheros Examina los ficheros de un disco en busca de actividad no autorizada y los lista por su última fecha de acceso, permitiendo realizar búsquedas en franjas horarias, búsqueda de archivos eliminados, etc. (open source) Descarga: (http://www.foundstone.com)	Windows	Alto
Herramienta	**Costo**	**Características**	**Plataformas**	**Evaluación**
Foremost	GPL	- Recupera ficheros basándose en sus cabeceras.	Linux	Bajo

	(proyecto abierto al público)	Puede trabajar sobre archivos de imágenes, como los generados con dd, Safeback, Encase, etc. o directamente sobre un disco o partición. Las cabeceras pueden especificarse a través de su archivo de configuración, por lo que se puede especificar búsquedas para formatos específicos.		
Herramienta	**Costo**	**Características**	**Plataformas**	**Evaluación**
Helix/FIRE	Libre	HELIX está basada en KNOPPIX. Live CD. Posee una variedad de herramientas para realizar un análisis forense tanto a equipos como imágenes de discos. Para MS Windows posee un conjunto de herramientas de 90 Mb, permitiendo trabajar con sistemas vivos, y recuperar información volátil. En el entorno Linux, dispone de un Sistema Operativo completo, con un núcleo modificado para conseguir una excelente detección de hardware. No realiza el montaje de particiones swap, ninguna otra operación sobre el disco duro del equipo sobre el que se arranque.	Windows, Solaris, Linux	Alto
		Es muy bueno para el análisis de equipos muertos, sin que se modifiquen las evidencias pues montará los discos que encuentre en el sistema en modo sólo lectura. Contiene más y nuevas versiones de SleuthKit y Autopsy. Su documentación no es amplia. Permite elegir entre usar los kernels (2.4.26 o 2.6.5). Tiene una excelente detección de hardware. HELIX está pensado específicamente para no realizar ningún tipo de alteración sobre los sistemas en los que se usa. Tiene una configuración autorun para Windows con herramientas para este SO.		

		Descarga: (http://www.e-fense.com/helix/)		
Herramienta	**Costo**	**Características**	**Plataformas**	**Evaluación**
F.I.R.E (Forensic and Incident Response Evironment)	Libre	□ Es una distribución de un único cdrom, portable y bootable Live CD. Provee herramientas adecuadas para una actuación rápida en casos de análisis forense. □ Respuesta ante incidentes, recuperación de datos, ataque de virus. □ Contiene gran cantidad de herramientas de análisis forense. □ Es usable para análisis en caliente de sistemas, con lo que únicamente montando el CD se puede usar. □ Herramientas compiladas estáticamente sin necesidad de realizar un reboot de la máquina. □ Posee una interfaz gráfica que hace fácil su uso. □ No realiza ninguna modificación sobre los equipos en los que se ejecute, por lo que puede ser utilizado con seguridad □ Recupera datos de particiones dañadas. □ F.I.R.E posee las siguientes herramientas: □ Nessus, nmap, whisker, hping2, hunt, fragrouter. □ Ethereal, Snort, tcpdump, ettercap, dsniff, airsnort. □ Chkrootkit, F-Port □ TCT, Autopsy.	Windows, Solaris y FreeWare	Alto
		□ Testdisk, fdisk, gpart. □ SSH (cliente y servidor), VNC (cliente y servidor) □ Mozilla, ircII, mc, Perl, biew, fenris, pgp. Descarga: (http://fire.dmzs.com/?section=main)o (http://biatchux.dmzs.com) .		
Herramienta	**Costo**	**Características**	**Plataformas**	**Evaluación**
BackTrack	Libre	□ Es una de las más conocidas y apreciadas distribuciones GNU/Linux	GNU/Linux	Alto

		• Ocupa el puesto 32 en el famoso ránking de Insecure.org. • Se presenta como un LiveCD (no requiere de instalación) • Posee 300 herramientas de todo tipo (sniffers, exploits, auditoría wireless, análisis forense, etc) perfectamente organizadas. • La versión 2 (recién publicada) utiliza un kernel 2.6.20 con varios parches e incluye soporte para tarjetas inalámbricas. • Los programas que trae este software ya vienen todos configurados y listos para ser usados, por lo que no se debe emplear tiempo en buscarlos e instalarlos.		
Herramienta	**Costo**	**Características**	**Plataformas**	**Evaluación**
FLAG (Forensic and Log Analysis GUI)	Poyecto Abierto al Público	• Simplificar el proceso de análisis de ficheros de log en investigaciones forenses. • Está basado en web, por lo que puede instalarse en un servidor donde se centralice toda la información de los análisis, de forma que puede ser consultada por todo el equipo forense. • pyFlag es la implementación (empleada actualmente) en Python. Es una revisión/reescritura completa de FLAG, más potente, versátil y robusta.	Linux	Bajo
Herramienta	**Costo**	**Características**	**Plataformas**	**Evaluación**
E-ROL	Libre	• Es una aplicación on-line segura y de fácil manejo. • Permite a los usuarios recuperar los archivos que hayan sido borrados de unidades de disco duro, unidades ZIP y disquetes, en todos los sistemas operativos de la familia Microsoft Windows. • Registra una media de más de 350 entradas diarias a su página web.	Windows	Bajo
Herramienta	**Costo**	**Características**	**Plataformas**	**Evaluación**
Ilook	Libre	• Recupera datos incluso los borrados. • Identifica y soporta varios sistemas archivos.	Linux, Windows	Media

		- Analiza funciones has. - Basado en Linux. - Recupera todos los datos incluido los de archivos borrados.		
Herramienta	**Costo**	**Características**	**Plataformas**	**Evaluación**
Bad Copy	Comercial $50	- Recupera datos corruptos en disquetes, CD, dispositivos usb o el propio disco local. - Recuperar todo tipo de archivos, como por ejemplo documentos, imágenes, aplicaciones, etc. - Utiliza un sistema inteligente de recuperación de datos y disco, para el contenido de ficheros originales; puede leer el contenido de archivos corruptos y en la mayoría de los casos, recuperarlos en todo o en parte, en el directorio que se especifique.	Windows	Media
Herramienta	**Costo**	**Características**	**Plataformas**	**Evaluación**
Encase	Comercial Sector Privado $290 - $6750	- Software líder en el mercado y de mayor uso en el campo de análisis forense. - Los formatos de archivos con extensión .cda contenidos en CD para equipos de música no son bien reconocidos por EnCase. - Muy usada en EEUU por el FBI. - No es multiplataforma. - Encase posee una variedad de funciones que se requiere de otro documento para explicar cada una de ellas.	Windows	Alta

3.6 Toolkit Básico de Computación Forense

Dejando aparte el software comercial, en el que se puede encontrar herramientas específicas como EnCase de la empresa Guidance Software, considerado un estándar en el análisis forense de sistemas pero no indispensable, nos centramos en herramientas de código abierto (Open Source) muchos de estos poseen una colección de herramientas en un solo software (toolkit) que pueden ser descargadas libremente desde las páginas de sus correspondientes autores o miembros del proyecto y que cumplen similar funcionalidad a las herramientas que son comerciales. A más de ello Linux es un entorno ideal en el cual realizar tareas de análisis forense permite proveer de una gran variedad de herramientas que facilitan todas las etapas que se deben llevar a cabo en la realización de un análisis exhaustivo de un sistema comprometido.

Mediante el cuadro anterior las herramientas que son claramente más recomendables y utilizadas; en el toolkit básico de computación forense son:

- **TCT** es una herramienta de código abierto, es decir no requiere la obtención de licencia tiene una evaluación media.

- **Forensic Toolkit** forma parte de la organizacion Foundstone, la misma está orientada a plataformas Windows.
- **The Sleuth Kit y Autopsy** es una herramienta altamente recomendable debido las funciones que posee, es gratuita y multiplataforma[17].

Otra de las herramientas recomendables pero posee una limitante es **BackTrack,** ya que solo funciona en Sistema Operativo Linux, pero tiene un alto rendimiento.

La siguiente herramienta es **E-ROL** se le considera con una evaluación baja debido a que carece de las funciones que poseen las herramientas antes mencionadas, pero puede ser de utilidad, al enfrentarse con entornos Windows para tomar indicios del ataque.

Uno de los elementos más importantes que un informático forense debe emplear al momento de recaudar evidencia digital, son los conocidos Live CD's o DVD's, que son una colección de herramientas, que permiten realizar un examen forense de imágenes sin tener que dedicar un equipo específico para ello y sin necesidad de cargar otro sistema operativo. Entre los más recomendables ellos tenemos:

[17]ANEXO I - Sleuth Kit & Autopsy

- **Helix** posee una evaluación alta, contiene un sin número de herramientas, entre una de ellas el Autopsy, es multiplataforma, y open source.
- **F.I.R.E** (Forensic and Incident Response Environment), este Live CD, es altamente recomendable. Si Helix contiene ha Autopsy, F.I.R.E. contiene a las herramientas TCT y Autopsy a más de un sin número de herramientas, útiles para la recolección de evidencias, es multiplataforma y gratuito.

3.7 Herramientas para recolección de evidencias

- TCT (Linux).
- Forensic Toolkit (Windows).
- The Sleuth Kit y Autopsy (Unix/Linux, Windows).
- BackTrack (Linux).
- E-ROL (Windows).
- Helix (Windows, Solaris, Linux).
- F.I.R.E (Windows, Solaris y Freeware).

3.8 Herramientas para el Monitoreo de Computadores

Honeypot

Es un software o conjunto de computadores cuya intención es atraer a crackers o spammers, simulando ser sistemas vulnerables o débiles a los ataques, permite recoger información sobre los atacantes y sus técnicas, los mismos pueden distraer a los atacantes de las máquinas más importantes del sistema, y advertir rápidamente al administrador del sistema de un ataque, además de permitir un examen en profundidad del atacante, durante y después del ataque al honeypot.

Algunos honeypots son programas que se limitan a simular sistemas operativos no existentes en la realidad y se les conoce como honeypots de baja interacción y son usados fundamentalmente como medida de seguridad. Otros sin embargo trabajan sobre sistemas operativos reales y son capaces de reunir mucha más información; sus fines suelen ser de investigación y se los conoce como honeypots de alta interacción.

También se llama honeypot a un website o sala de chat, que se ha creado para descubrir a otro tipo de usuarios con intenciones criminales.

KeyLogger

Es una herramienta que puede ser útil cuando se quiere comprobar actividad sospechosa; guarda los eventos generados por el teclado, es decir, cuando el usuario teclea la tecla de 'enter', esto es guardado en un archivo o es enviado por e-mail.

Existen dos versiones: la registrada y la de demostración. La principal diferencia es que en la versión registrada se permite correr el programa en modo escondido. Esto significa que el usuario de la máquina no notará que sus acciones están siendo registradas.

3.9 Entornos de Trabajo Forense sobre computación virtual

Una de las maneras más factibles de trabajar en una investigación forense es la implementación de una arquitectura de servidores virtuales.

La labor pericial demanda el uso de distintos Sistemas Operativos (Windows, Linux, Solaris, etc.) como plataforma de ejecución de aplicaciones forenses. Para ello se puede implementar puestos de trabajo con máquinas virtuales, facilitando a los investigadores la utilización de un mayor número de recursos de software en forma simultánea.

La **virtualización** se refiere a una capa de abstracción que separa el hardware del sistema operativo, optimizando y flexibilizando de esta manera la utilización de los recursos computacionales. Permite que múltiples máquinas virtuales con sistemas operativos heterogéneos puedan funcionar simultáneamente en la misma computadora. Cada máquina virtual tiene asignado un conjunto propio de recursos de hardware sobre el que pueden funcionar diferentes aplicaciones. La virtualización brinda la posibilidad de mejorar la infraestructura informática en cuanto a escalabilidad, seguridad y una variedad de modalidades en la administración de servidores.

Los beneficios de la virtualización pueden ser apreciados sobre tres aspectos de alto impacto para la administración de sistemas:

- Particionamiento (permite que múltiples aplicaciones y sistemas operativos puedan compartir el mismo hardware).
- Aislamiento de componentes (si una máquina virtual falla esta situación no afecta al funcionamiento de las restantes).
- Encapsulación (permite que una máquina virtual pueda almacenarse en un simple archivo facilitando el backup de la misma, la copia, o el traslado).

CAPÍTULO IV

ANÁLISIS

En este capítulo se lleva a cabo el análisis de la Intrusión Informática realizada el día 1 de Octubre del 2003 en las instalaciones del Help Desk de la institución bancaria JBR.

Se explicará como el atacante logró penetrar las seguridades del Banco JBR, que daños logró causar y así mismo crear la línea de tiempo de los ataques realizados por este individuo.

4.1 Identificación y Planteamiento del Problema

La Institución bancaria JBR se comunicó con nuestros consultores en Investigación Forense el día 1 de Octubre del 2003 solicitando nuestra

ayuda, ellos habían sufrido un ataque el cual gracias al Equipo de Respuesta a Incidentes del Banco ya había sido detectado. Ellos logran capturar toda la información volátil al momento del ataque mediante el uso de distintos métodos y herramientas forenses que serán explicados más adelante en este capítulo.

Se cree que el atacante logró instalar herramientas para su beneficio por medio de una puerta trasera anteriormente abierta por el mismo sujeto. Nuestro equipo de consultores analizará toda la evidencia que nos facilitó el banco para comprobar si esta suposición es cierta.

Aunque el Equipo de Respuesta a Incidentes del Banco hizo un excelente trabajo capturando toda la información volátil posible, solo eso no bastaría para identificar de forma eficaz al atacante por esto también se realizó una duplicación forense del equipo afectado utilizando las herramientas respectivas para mantener la cadena de custodia intacta y de esta manera realizar una investigación transparente y confiable.

4.2 Adquisición de Imágen

El equipo de Respuesta Incidentes del Banco JBR realizó una duplicación forense utilizando una herramienta no comercial llamada

dd[18]. Con esta herramienta realizaron una duplicación de todo el sistema de archivos del equipo afectado, de esta manera lograremos identificar cualquier tipo de evidencia que el atacante haya dejado involuntariamente, ya sean archivos ocultos o borrados.

4.3 Análisis de la Información Volátil

En nuestro escenario actual, la mayoría de los datos más importantes que adquirimos estaba en los datos volátiles. Los datos volátiles de un ordenador victima suele contener información importante que nos ayuda a determinar el "quién", "cómo", y, posiblemente, "por qué" del incidente.

Para responder estas preguntas, entre los datos volátiles recopilados podemos adquirir esta información:

- La Fecha y Hora del sistema
- Las conexiones actuales de red
- Puertos TCP/UDP abiertos
- Cuales aplicaciones ejecutables están abriendo puertos TCP/UDP
- Tabla de nombres de NetBIOS en el caché
- Sesiones de Usuarios abiertas

[18] ANEXO II – Herramienta dd

- La tabla de enrutamiento interna
- Procesos que se están ejecutando
- Servicios en ejecución
- Tareas programadas
- Archivos abiertos
- Proceso de volcados de memoria

Vamos a abordar cada una de estas áreas vitales en sus respectivas secciones y analizar los datos que adquirimos de JBRWWW.

4.3.1 Hora y Fecha del Sistema

Esta es probablemente la información más fácil de recopilar y entender, sin embargo, es una de las piezas más importantes de información para el investigador.

Aunque en nuestro escenario tenemos un sistema único, su intrusión puede implicar decenas o cientos de sistemas. Mantener la hora del sistema a tiempo y observando el desplazamiento de una fuente de confianza (tales como un servidor NTP) es primordial a la hora de examinar los archivos de registro de tiempo o de otro tipo basadas en pruebas a partir de múltiples servidores.

La hora y la fecha son simplemente recogidas de los archivos facilitados por el equipo de Respuesta a Incidentes. La hora y la fecha de JBRWWW se consideraron como sigue:

- La fecha actual es: Miércoles 10/01/2003
- La hora actual es: 21:58:19.29

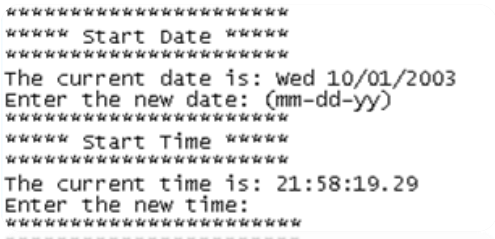


Figura 4-1: Hora y Fecha de la captura de datos

4.3.2 Conexiones Actuales de Red

Es posible que mientras realizáramos nuestro proceso de captura de datos volátiles, el atacante esté conectado al servidor. También podría ser posible que el atacante esté ejecutando un mecanismo de fuerza bruta contra otras máquinas.

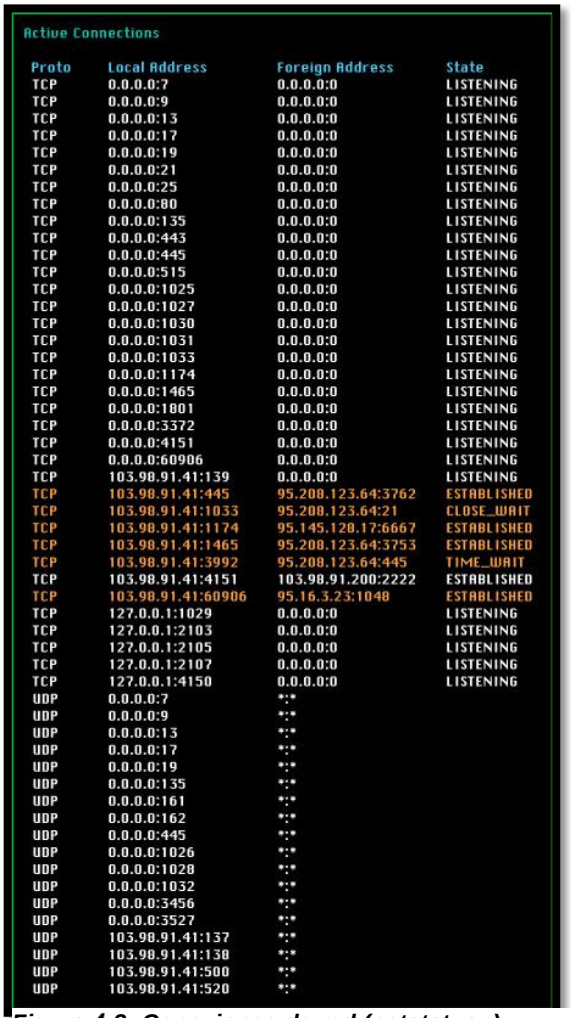
En escenarios similares a los mencionados anteriormente el atacante sería detectado si examinamos las conexiones actuales de red.

Podemos visualizar las conexiones de red actuales de una máquina mediante el uso del comando netstat[19]. Además utilizando la opción –an al final del comando logrará capturar todas las conexiones de red y

[19] ANEXO III – Comando Netstat

mostrar la dirección IP cruda en vez del Fully Qualified Domain Names (FQDN)

Al ejecutar el comando netstat –an recibimos esta repuesta, la cual hemos ilustrado para mejor comprensión en la siguiente imagen:



Active Connections

Proto	Local Address	Foreign Address	State
TCP	0.0.0.0:7	0.0.0.0:0	LISTENING
TCP	0.0.0.0:9	0.0.0.0:0	LISTENING
TCP	0.0.0.0:13	0.0.0.0:0	LISTENING
TCP	0.0.0.0:17	0.0.0.0:0	LISTENING
TCP	0.0.0.0:19	0.0.0.0:0	LISTENING
TCP	0.0.0.0:21	0.0.0.0:0	LISTENING
TCP	0.0.0.0:25	0.0.0.0:0	LISTENING
TCP	0.0.0.0:80	0.0.0.0:0	LISTENING
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING
TCP	0.0.0.0:443	0.0.0.0:0	LISTENING
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING
TCP	0.0.0.0:515	0.0.0.0:0	LISTENING
TCP	0.0.0.0:1025	0.0.0.0:0	LISTENING
TCP	0.0.0.0:1027	0.0.0.0:0	LISTENING
TCP	0.0.0.0:1030	0.0.0.0:0	LISTENING
TCP	0.0.0.0:1031	0.0.0.0:0	LISTENING
TCP	0.0.0.0:1033	0.0.0.0:0	LISTENING
TCP	0.0.0.0:1174	0.0.0.0:0	LISTENING
TCP	0.0.0.0:1465	0.0.0.0:0	LISTENING
TCP	0.0.0.0:1801	0.0.0.0:0	LISTENING
TCP	0.0.0.0:3372	0.0.0.0:0	LISTENING
TCP	0.0.0.0:4151	0.0.0.0:0	LISTENING
TCP	0.0.0.0:60906	0.0.0.0:0	LISTENING
TCP	103.98.91.41:139	0.0.0.0:0	LISTENING
TCP	103.98.91.41:445	95.208.123.64:3762	ESTABLISHED
TCP	103.98.91.41:1033	95.208.123.64:21	CLOSE_WAIT
TCP	103.98.91.41:1174	95.145.120.17:6667	ESTABLISHED
TCP	103.98.91.41:1465	95.208.123.64:3753	ESTABLISHED
TCP	103.98.91.41:3992	95.208.123.64:445	TIME_WAIT
TCP	103.98.91.41:4151	103.98.91.200:2222	ESTABLISHED
TCP	103.98.91.41:60906	95.16.3.23:1048	ESTABLISHED
TCP	127.0.0.1:1029	0.0.0.0:0	LISTENING
TCP	127.0.0.1:2103	0.0.0.0:0	LISTENING
TCP	127.0.0.1:2105	0.0.0.0:0	LISTENING
TCP	127.0.0.1:2107	0.0.0.0:0	LISTENING
TCP	127.0.0.1:4150	0.0.0.0:0	LISTENING
UDP	0.0.0.0:7	*:*	
UDP	0.0.0.0:9	*:*	
UDP	0.0.0.0:13	*:*	
UDP	0.0.0.0:17	*:*	
UDP	0.0.0.0:19	*:*	
UDP	0.0.0.0:135	*:*	
UDP	0.0.0.0:161	*:*	
UDP	0.0.0.0:162	*:*	
UDP	0.0.0.0:445	*:*	
UDP	0.0.0.0:1026	*:*	
UDP	0.0.0.0:1028	*:*	
UDP	0.0.0.0:1032	*:*	
UDP	0.0.0.0:3456	*:*	
UDP	0.0.0.0:3527	*:*	
UDP	103.98.91.41:137	*:*	
UDP	103.98.91.41:138	*:*	
UDP	103.98.91.41:500	*:*	
UDP	103.98.91.41:520	*:*	

Figura 4-2: Conexiones de red (netstat -an)

Estas líneas resaltadas representan conexiones de red activas, entre estas podemos obviar la línea de la dirección 103.98.91.200, ya que esta pertenece a la estación de trabajo forense que realizo al captura de información volátil, el puerto 2222 es una conexión esperada porque es el puerto que abre la estación de trabajo forense para capturar los datos mediante el comando netcat. Después de separar la información extraña, nos quedamos con las 6 líneas interesantes.

95.208.123.64:3762

La primera línea nos indica una conexión al puerto NetBIOS de Windows 2000. Por lo tanto, la dirección IP 95.208.123.64 podría estar emitiendo comandos con una herramienta como psexec[20], compartiendo un archivo mediante el uso del comando net, o la explotación de algunas otras funcionalidades de Microsoft Windows.

95.208.123.64:21

La segunda línea es muy interesante. JBRWWW está conectándose al puerto 21, el puerto FTP del sistema de 95.208.123.64. Asumiendo que el administrador no ha abierto esta conexión de ninguna manera, esta línea la colocamos como actividad sospechosa.

95.145.128.17:6667

[20] ANEXO IV – PsTools Suite

La tercera línea es una conexión a un servidor IRC (el puerto TCP 6667) en 95.145.128.17. Otra vez asumimos que el administrador no participó en esta conexión.

95.208.123.64:3753

La cuarta no es muy familiar, por lo que haciendo una búsqueda en internet nos demuestra que puede ser el servicio "nattyserver" o "ChilliASP". Esta información no nos dice mucho por lo que marcamos esta línea como posiblemente sospechosa.

95.208.123.64:445

La quinta línea nos dice que tenemos una conexión NetBIOS desde nuestra máquina víctima a la IP 95.208.123.64. Esto podría indicar que el atacante ha usado el comando net para compartir algo desde máquina atacante a su máquina víctima.

95.16.3.23:1048

La última línea muestra una conexión con JBRWWW desde el puerto TCP 60906. Los puertos por encima de 1024 suelen ser temporales. Observe que también se conecta a un puerto temporal en una dirección IP diferente de destino en 95.16.3.23. A simple vista parecería una conexión normal, pero el uso de puertos temporales nos hace creer que es una conexión sospechosa.

4.3.3 Puertos TCP/UDP abiertos

Revisando la respuesta del comando netcat demostrado anteriormente, todas las líneas que no están resaltadas son los puertos abiertos. Lo que nos llama la atención de estas líneas es un puerto abierto usualmente es una puerta trasera en la maquina víctima. Notamos que Windows abre un montón de puertos legítimo para poder ejecutar sus funciones normalmente, pero podemos eliminar muchos de ellos rápidamente.

Desde la primera línea hasta la del puerto TCP 515 son puertos normales de Windows, por lo general son iniciados cuando IIS y simples servicios de TCP / IP son instalados en la máquina. Los próximos puertos hasta los de conexiones establecidas son puertos por encima de 1024, quiere decir que son temporales.

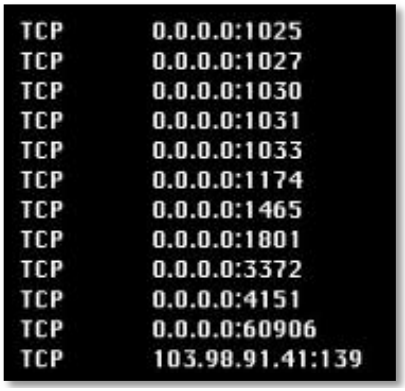


Figura 4-3: Conexiones con puertos por encima de 1024

Esta información no es suficiente para sacar una conclusión si el atacante está usando estos puertos como una puerta trasera, por lo que debemos ver cuales ejecutables abrieron estos puertos para poder tener una mejor idea de cuál es su propósito.

4.3.4 Ejecutables abriendo puertos TCP/UDP

Para examinar los puertos extraños que están abiertos, tenemos que vincular los puertos abiertos a los ejecutables que les abrió. Para conseguir esta información podemos usar la herramienta libre Fport que la podemos encontrar en la siguiente dirección (http://www.mcafee.com/us/downloads/free-tools/fport.aspx) . FPort no necesita líneas adicionales de comando para ejecutarlo durante nuestra captura de datos. Después de haber ejecutado FPort, hemos recibido el siguiente resultado:

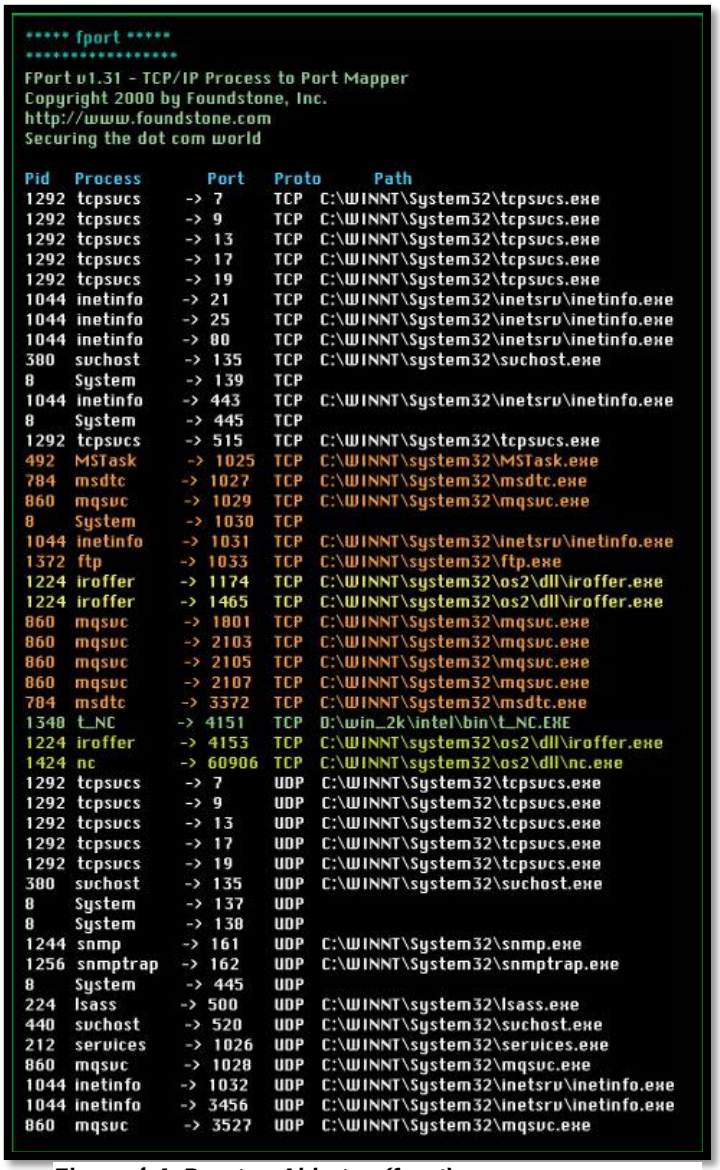


Figura 4-4: Puertos Abiertos (fport)

Hemos resaltado los puertos no identificados. Las primeros 5 líneas resaltadas podemos atribuirlas a sistemas binarios abriendo puertos TCP 1025, 1027, 1029, 1030 y 1031. La siguiente línea nos muestra que alguien estaba corriendo el servicio FTP cliente en JBRWWW, asumimos

que el administrador no inició este servicio, por lo cual marcamos esta acción como sospechosa.

Las siguientes líneas nos muestran un ejecutable corriendo en C:\winnt\system32\OS2\dll que se denomina iroffer.exe:

Figura 4-5: Puertos sospechosos (fport) 1

Sin lugar a duda esta línea nos parece sospechosa porque no estamos conscientes de los DLLs del sistema que abran puertos de red relacionados. Realizando una rápida búsqueda en (http://www.iroffer.org) Se trata de un sitio Web real, y la herramienta tiene fines legítimos. Aparentemente, esta herramienta es un robot que se conecta a canales de IRC y ofrece control remoto de JBRWWW. Por lo tanto, estas dos líneas nos confirman que hubo un incidente.

Las próximas 5 líneas nos muestran puertos abiertos por mqsvc.exe afiliados a Microsoft Message Queue Server. La siguiente línea nos muestra nuestra sesión de captura de datos con netcat.

Figura 4-6: Puertos sospechosos (fport) 2

Esta sesión de netcat fue renombrada siguiendo las guías de mejores práctica de investigación forense, de esta manera evitamos copiarla encima de otra sesión de netcat con el nombre por default, el cual es nc.

Las siguientes líneas nos muestran información muy importante acerca de las puertas traseras del atacante.

Figura 4-7: Puertos sospechosos (fport) 3

Al parecer el atacante no solo tenía una sesión de iroffer iniciada sino que también una sesión de netcat. No podemos saber con certeza lo que el atacante planeaba hacer con la sesión de netcat. Podría ser una conexión de salida, o puede ser en modo de escucha, permitiendo las conexiones de entrada libre acceso a un comando shell. Al reexaminar la salida de netstat demostrado anteriormente, vemos que el puerto 60906 está escuchando activamente. Por lo tanto, podemos concluir gracias a netcat y FPort que la puerta trasera en 60906 está escuchando conexiones y activamente conectado a una dirección IP extraña.

4.3.5 Tablas de Nombres de NetBIOS en el cache

Cuando examinamos logs en un sistema operativo Windows debemos tener en cuenta que hasta la versión Windows 2003, esta almacenaba las conexiones específicas por NetBIOS en vez de por direcciones IP. Esto no nos favorece, ya que un atacante fácilmente podría cambiarse su NetBIOS para realizar un atacante y al terminar dicho ataque volver a cambiarlo a su valor original, dejándonos sin rastro alguno.

Porque queremos asignar un nombre de NetBIOS a una dirección vamos a ejecutar el comando nbtstat, el cual nos mostrará una tabla de nombres de NetBIOS almacenada en el cache, por lo que cabe recalcar que no será un historial completo de conexiones. Las conexiones que nos mostrará habrán sido hace un corto periodo de tiempo.

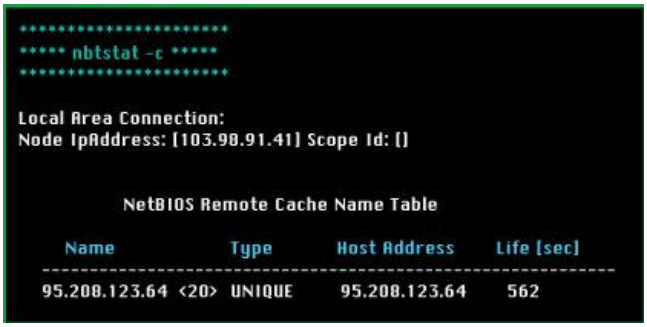


Figura 4-8: NetBIOS almacenado en el cache

Esta es una buena respuesta, ya que el nombre de este server es el mismo que la dirección IP localizado en 95.208.123.64. Por lo general, el nombre NetBIOS aparece en la columna “nombre”.

4.3.6 Sesiones de Usuarios Abiertas

Para obtener esta información una de las herramientas más efectivas sería PsLoggedOn[21], esta herramienta la podemos encontrar gratuitamente en el vínculo (http://technet.microsoft.com/en-us/sysinternals/bb897545.aspx). Esta herramienta nos muestra a los usuarios que actualmente están registrados o accediendo a los recursos compartidos. Al ejecutar esta herramienta en JBRWWW, recibimos la siguiente información:

Figura 4-9: Sesiones Abiertas (psloggedon)

Tenemos un usuario registrado localmente, este es el usuario Administrador desde el cual se están llevando a cabo la ejecución de las

[21] ANEXO IV – PsTools Suite

diferentes herramientas forenses. El segundo usuario que tenemos registrado también posee permisos de Administrador, pero este se encuentra registrado remotamente. Por lo tanto sabemos que alguien más se encuentra registrado mientras se realiza la investigación en el sistema JBRWWW.

Regresando a nuestras conexiones de red anteriormente mostradas notamos lo siguiente:

Figura 4-10: Conexión sospechosa confirmada por psloggedon

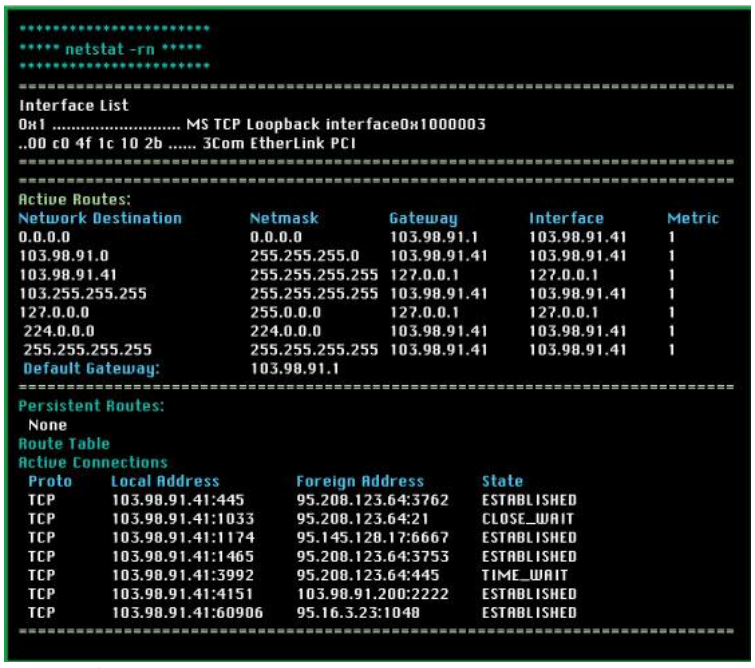
Para que un usuario esté conectado remotamente debería hacerlo mediante un puerto de NetBIOS. En Windows 2000, el puerto es el 445 ó 139, en versiones anteriores solo era el puerto 139. Por lo tanto ahora sabemos con certeza que la dirección IP del atacante es 95.208.123.64

4.3.7 Tablas de Enrutamiento Interna

Uno de los más terribles usos de un servidor comprometido es alterar las tablas de enrutamiento para redirigir tráfico de alguna manera. El atacante puede beneficiarse de esto para evadir dispositivos de seguridad, tales como un firewall. En caso de haber un firewall en el

camino hacia la siguiente víctima, el atacante podría llegar a través de otro router que posea una Lista de Control de Acceso más permisiva. Otra de las razones para que el atacante quiera alterar las tablas de enrutamiento es redirigir el tráfico de tal manera que pueda capturar la información que se está transmitiendo entre los usuarios sin ser detectado.

Podemos examinar la tabla de enrutamiento ejecutando el comando netstat con el parámetro –rn:



```
***********************
***** netstat -rn *****
***********************
===========================================================================
Interface List
0x1 ........................... MS TCP Loopback interface0x1000003
..00 c0 4f 1c 10 2b ...... 3Com EtherLink PCI
===========================================================================
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0      103.98.91.1    103.98.91.41       1
       103.98.91.0    255.255.255.0    103.98.91.41    103.98.91.41       1
      103.98.91.41  255.255.255.255        127.0.0.1       127.0.0.1       1
 103.255.255.255  255.255.255.255    103.98.91.41    103.98.91.41       1
        127.0.0.0        255.0.0.0        127.0.0.1       127.0.0.1       1
        224.0.0.0        224.0.0.0    103.98.91.41    103.98.91.41       1
  255.255.255.255  255.255.255.255    103.98.91.41    103.98.91.41       1
  Default Gateway:      103.98.91.1
===========================================================================
Persistent Routes:
  None
Route Table
Active Connections
  Proto  Local Address          Foreign Address        State
  TCP    103.98.91.41:445       95.208.123.64:3762     ESTABLISHED
  TCP    103.98.91.41:1033      95.208.123.64:21       CLOSE_WAIT
  TCP    103.98.91.41:1174      95.145.128.17:6667     ESTABLISHED
  TCP    103.98.91.41:1465      95.208.123.64:3753     ESTABLISHED
  TCP    103.98.91.41:3992      95.208.123.64:445      TIME_WAIT
  TCP    103.98.91.41:4151      103.98.91.200:2222     ESTABLISHED
  TCP    103.98.91.41:60906     95.16.3.23:1040        ESTABLISHED
===========================================================================
```

Figura 4-11: Tabla de Enrutamiento Interna

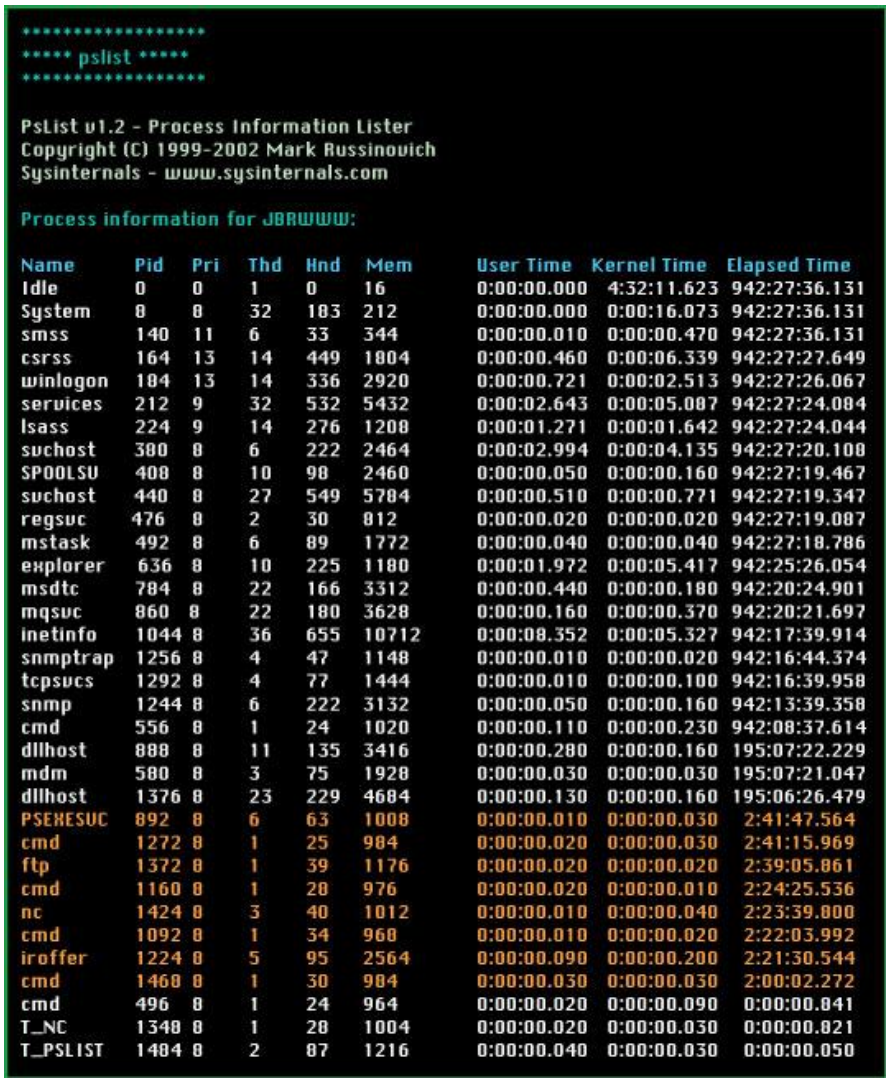
La tabla de enrutamiento parece ser normal en este servidor. Este comando además nos muestra las conexiones de red abiertas. Esta lista de conexiones de red concuerda con la mostrada anteriormente mediante el comando netstat –an[22].

4.3.8 Procesos Ejecutándose

Necesitamos averiguar que procesos el atacante ejecutó en JBRWWW, ya que así podremos saber si tiene puertas traseras. Podemos mostrar una lista de procesos mediante el uso de la herramienta Pslist[23], que puede ser encontrado gratuitamente en la siguiente dirección (http://technet.microsoft.com/en-us/sysinternals/bb896682.aspx).

Tras examinar está información notamos que las primeras líneas son procesos del sistemas ejecutados a lo largo del tiempo de funcionamiento. Esto no indica que son procesos que están ejecutándose desde el arranque del sistema, los cuales son procesos típicos de sistema. El atacante podría haber ejecutado algo desde el arranque por lo que podríamos estar perdiendo esta información, así que deberíamos comparar esta lista de procesos con la de otro servidor no comprometido.

[22] ANEXO III – Comando Netstat
[23] ANEXO IV – PsTools Suite



```
******************
***** pslist *****
******************

PsList v1.2 - Process Information Lister
Copyright (C) 1999-2002 Mark Russinovich
Sysinternals - www.sysinternals.com

Process information for JBRWWW:

Name        Pid   Pri   Thd   Hnd   Mem      User Time   Kernel Time   Elapsed Time
Idle        0     0     1     0     16       0:00:00.000  4:32:11.623  942:27:36.131
System      8     8     32    183   212      0:00:00.000  0:00:16.073  942:27:36.131
smss        140   11    6     33    344      0:00:00.010  0:00:00.470  942:27:36.131
csrss       164   13    14    449   1804     0:00:00.460  0:00:06.339  942:27:27.649
winlogon    184   13    14    336   2920     0:00:00.721  0:00:02.513  942:27:26.067
services    212   9     32    532   5432     0:00:02.643  0:00:05.087  942:27:24.084
lsass       224   9     14    276   1208     0:00:01.271  0:00:01.642  942:27:24.044
svchost     380   8     6     222   2464     0:00:02.994  0:00:04.135  942:27:20.108
SPOOLSV     408   8     10    98    2460     0:00:00.050  0:00:00.160  942:27:19.467
svchost     440   8     27    549   5784     0:00:00.510  0:00:00.771  942:27:19.347
regsvc      476   8     2     30    812      0:00:00.020  0:00:00.020  942:27:19.087
mstask      492   8     6     89    1772     0:00:00.040  0:00:00.040  942:27:18.786
explorer    636   8     10    225   1180     0:00:01.972  0:00:05.417  942:25:26.054
msdtc       784   8     22    166   3312     0:00:00.440  0:00:00.180  942:20:24.901
mqsvc       860   8     22    180   3628     0:00:00.160  0:00:00.370  942:20:21.697
inetinfo    1044  8     36    655   10712    0:00:08.352  0:00:05.327  942:17:39.914
snmptrap    1256  8     4     47    1148     0:00:00.010  0:00:00.020  942:16:44.374
tcpsvcs     1292  8     4     77    1444     0:00:00.010  0:00:00.100  942:16:39.958
snmp        1244  8     6     222   3132     0:00:00.050  0:00:00.160  942:13:39.358
cmd         556   8     1     24    1020     0:00:00.110  0:00:00.230  942:08:37.614
dllhost     888   8     11    135   3416     0:00:00.280  0:00:00.160  195:07:22.229
mdm         580   8     3     75    1928     0:00:00.030  0:00:00.030  195:07:21.047
dllhost     1376  8     23    229   4684     0:00:00.130  0:00:00.160  195:06:26.479
PSEXESVC    892   8     6     63    1008     0:00:00.010  0:00:00.030  2:41:47.564
cmd         1272  8     1     25    984      0:00:00.020  0:00:00.030  2:41:15.969
ftp         1372  8     1     39    1176     0:00:00.020  0:00:00.020  2:39:05.861
cmd         1160  8     1     28    976      0:00:00.020  0:00:00.010  2:24:25.536
nc          1424  8     3     40    1012     0:00:00.010  0:00:00.040  2:23:39.800
cmd         1092  8     1     34    968      0:00:00.010  0:00:00.020  2:22:03.992
iroffer     1224  8     5     95    2564     0:00:00.090  0:00:00.200  2:21:30.544
cmd         1468  8     1     30    984      0:00:00.030  0:00:00.030  2:00:02.272
cmd         496   8     1     24    964      0:00:00.020  0:00:00.090  0:00:00.841
T_NC        1348  8     1     28    1004     0:00:00.020  0:00:00.030  0:00:00.821
T_PSLIST    1484  8     2     87    1216     0:00:00.040  0:00:00.030  0:00:00.050
```

Figura 4-12: Lista de procesos en ejecución

La parte que hemos resaltado nos muestra los procesos ejecutados por el atacante. Los procesos fueron ejecutados aproximadamente 2 horas y 40 minutos antes de ejecutar nuestra captura de datos en vivo. Esta evidencia nos da un rastro de tiempo de cuando el atacante estuvo conectado a JBRWWW. La máquina fue iniciada hace un largo tiempo, por lo que su ataque inicial puede haber sido cercanamente 3 horas antes de nuestra captura de datos en vivo. Si calculamos la hora en que

se inició la captura de datos en vivo fue 21:58 y le restamos las 2 horas y 40 minutos, el ataque pudo haber empezado a las 19:18 del 1 de Octubre del 2003.

Al parecer el atacante ejecutó PSEXECSVC, el cual es el resultado de un comando ejecutado por PsExec, esta herramienta habilita a un usuario para conectarse de una máquina con Microsoft Windows a otra y ejecutar comandos sobre una conexión NetBIOS, puede ser encontrada en el siguiente vínculo (http://technet.microsoft.com/es-es/sysinternals/bb897553.aspx). Esto podría explicar la conexión mediante el puerto 445 que descubrimos anteriormente. Po lo general los atacantes usan esta herramienta para ejecutar cmd.exe. Sabiendo que el atacante está ejecutando PsExec nos dice un montón acerca de esta intrusión. Primero PsExec abrirá un canal si le proporcionas los permisos de nivel administrador, por ende sabemos que el atacante tiene en su posesión una contraseña de un usuario de nivel Administrador. Segundo el atacante posee una contraseña que funciona a través de toda la empresa JBR. Tercero, el atacante debe estar ejecutando su ataque a través de una máquina con un sistema de Microsoft Windows para poder ejecutar PsExec.

Además vemos que el atacante está ejecutando el comando ftp. Una de las primeras cosas que realiza un atacante cuando ganan acceso es transferir sus herramientas a la máquina víctima.

Las ultimas 3 líneas pertenecen a la captura de datos en vivo que realizamos, las cuales esperábamos encontrar.

4.3.9 Servicios Ejecutándose

En la lista de procesos que mostramos anteriormente tuvimos una salida de PsExecSvc, donde SVC por lo general significa servicio. Podemos obtener una lista de servicios utilizando la herramienta PsService[24]. Esta herramienta muestra detalladamente las propiedades de cada servicio ejecutándose al momento de su uso, por lo que la respuesta obtenida es demasiado extensa. La podemos encontrar de forma gratuita en el siguiente vínculo (http://technet.microsoft.com/en-us/sysinternals/bb897542.aspx). El único servicio que parecía ser sospechoso es el siguiente:

[24] ANEXO IV – PsTools Suite

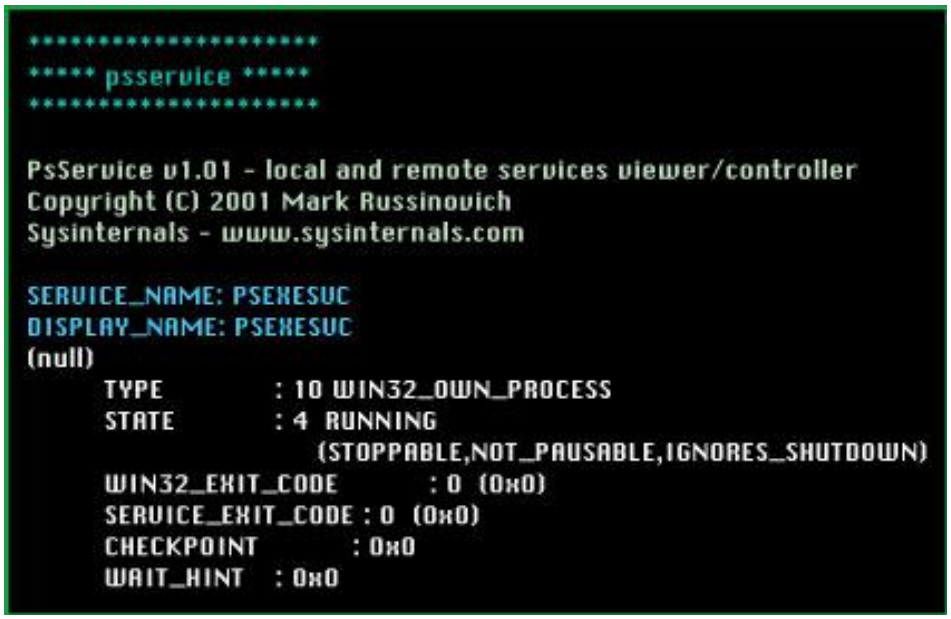


Figura 4-13: Servicios en ejecución (solo se muestra el más relevante para la investigación)

Los demás servicios mostrados con esta herramienta eran principalmente de Microsoft Windows, estos contenían descripciones válidas acerca de sus propósitos. Este servicio no posee una descripción, solo nos muestra el valor “null” donde la descripción debería encontrarse. Un atacante puede fácilmente esconder programas dentro de los servicio. La salida que nos muestra psservice es muy extensa, por lo que es normal para un investigador pasar por alto algún servicio extraño. De acuerdo a las tendencias de los atacantes, estos siempre prefieren usar servicios que obligan a los programas a ejecutarse durante el arranque del sistema, logrando así instalar su puerta trasera; utilizan herramientas como Firedaemon, esta herramienta convierte cualquier proceso en un servicio y le obliga su ejecución durante el arranque del sistema.

4.3.10 Tareas Programadas

Los atacantes con acceso administrativo pueden programar tareas. Esto les permite ejecutar comandos cuando ni siquiera están registrados en el sistema. Por ejemplo un atacante podría programar una tarea para que cada noche a las 2:00am ejecute un programa que abra una puerta trasera, de esta manera los métodos de seguridad preventiva que trabajan durante horas normales no servirían de nada.

Usamos el comando At y no nos muestra ningún tipo de información por lo que no tenemos preocupación de que el ataque sea dado de esta manera.

4.3.11 Archivos Abiertos

Examinando la lista de archivos abiertos obtendremos más información útil para nuestra investigación. Para esto haremos uso de la herramienta Psfile[25], la cual podemos encontrar gratuitamente en el siguiente vínculo (http://technet.microsoft.com/en-us/sysinternals/bb897552.aspx). Al ejecutarlo nos mostrará los siguientes resultados:

[25] ANEXO IV – PsTools Suite

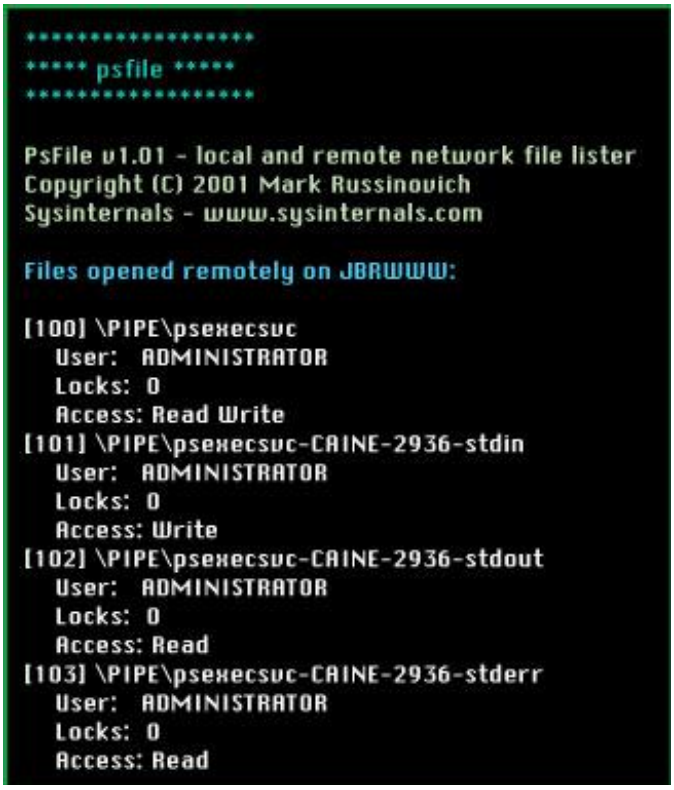


Figura 4-14: Lista de Archivos Abiertos (Psfile)

Podemos observar que un sistema PIPE abrió PSEXECSVC. También nos muestra el nombre Caine, este debería ser el nombre de NetBIOS del ordenador que se conectó a JBRWWW a través de PsExec.

4.3.12 Procesos de Volcado de Memoria

Hemos visto que el atacante inició su proceso de intrusión a través de JBRWW, pero aún no sabemos con certeza que procesos ejecutó. Siguiendo las guías de las mejores prácticas de Investigaciones

Forenses, hemos decidido capturar el espacio de memoria de los procesos sospechosos.

Este proceso de captura de volcado de memoria raramente es realizado debido a la falta de métodos documentados, técnicas y herramientas para llevarlo a cabo. La naturaleza de los sistemas operativos, combinado con las restricciones impuestas en áreas protegidas, hace que la adquisición de la memoria sea compleja y problemática. Debido a la constante mejoría y sofisticación en las herramientas y técnicas de intrusión, estamos obligados a realizar este proceso de importancia primordial.

Microsoft provee una utilidad llamada userdump.exe para la familia de sistemas operativos de Windows NT que no permite capturar el espacio de memoria utilizado por cualquier proceso en ejecución. Esta herramienta es un componente de las herramientas de soporte de Microsoft OEM y está disponible en el siguiente vínculo (http://www.microsoft.com/en-us/download/details.aspx?id=4060).

Porque userdump escribe la memoria extraída en el disco, no podemos usar la sesión de netcat para transferir la información directamente. Como queremos tener el mínimo impacto posible dentro del sistema

víctima, debemos tomar ciertas medidas antes de ejecutar el comando userdump, ya que este escribe grandes archivo pudiendo borrar evidencia valiosa en el proceso. Para esto crearemos una unidad en red compartida de nuestra estación forense utilizando los siguientes comandos:

```
C:\> net use Z: \\103.98.91.200\data
```

Si ejecutamos userdump.exe sin ningún parámetro este nos mostrará la ayuda. Debemos proveerle un Process ID (PID), el cual obtuvimos de los resultados del comando pslist anteriormente. Para guardar la sesión de netcat del atacante a nuestra unidad en red, ejecutamos el siguiente comando:

```
userdump 1424 Z:\nc_1424.dmp
```

Realizamos el proceso de volcado de memoria para los procesos 1092, 1160, 1272, 1468, 1372 1224, 1424 y 892. Una vez obtenida esta evidencia procedemos a utilizar la herramienta dumpchk.exe, el cual es un componente del Windows Debugging Tool Kit, el cual podemos encontrar en el siguiente vínculo (http://msdn.microsoft.com/en-US/windows/hardware/hh852362)

La herramienta dumpchk está diseñada para validar los volcados de memoria; por lo que nos provee información valiosa. Hemos ejecutado esta herramienta para examinar los archivos de volcado de memoria que creamos anteriormente. Podemos examinar estos archivos más a fondo mediante el uso del comando strings el cual lo podemos encontrar en el siguiente vinculo de manera gratuita (http://technet.microsoft.com/en-us/sysinternals/bb897439.aspx). Al ejecutar este comando rápidamente podremos hacernos una idea del ambiente en que se ejecutó la aplicación, ya que nos brinda información como el nombre del computador, la dirección del sistema, la dirección de la aplicación ejecutada y la línea de comando usada.

El resultado nos confirma el nombre de archivo y la ubicación y nos provee una lista de vínculos dinámicos de archivos de biblioteca junto a la línea de tiempo y los comandos utilizados para iniciar un proceso netcat.

Podemos observar que la línea de comando de netcat indica que fue configurado para separarse de la consola, escuchar en el puerto 60906 y ejecutar un comando Shell cuando tenga una conexión. Esta información

volátil hubiera sido perdida de no haber realizado el proceso de volcado de memoria.

Continuando con la examinación de los archivos de volcado de memoria encontrados que el PID 1224 fue iniciado por una línea de comandos de iroffer, myconfig, y que el PID 1372 mediante FTP 95.20.123.64

Ahora podemos examinar los volcados de memoria para información adicional mediante la búsqueda a través de las cadenas de caracteres ASCII que están incrustados dentro. Debido a que los datos almacenados por una aplicación o proceso en la memoria pueden estar en formato Unicode, es necesario utilizar una versión Unicode de Windows capaz de leer las cadenas de comando.

4.3.13 Volcado de memoria del sistema completo

Ahora poseemos la memoria de las aplicaciones sospechosas, pero también necesitamos capturar toda la memoria completa del sistema, la cual puede tener rastros de otros procesos intrusos o sesiones anteriores. Utilizando la herramienta dd podemos realizar una captura completa. Esta herramienta la podemos adquirir gratuitamente en el siguiente vínculo (http://www.chrysocome.net/dd). Con esta utilidad

podemos crear un objeto de mapeado a un archivo de página u otro archivo en el disco

4.4 Analizando Información No Volátil

La Información no volátil la podríamos conseguir analizando mediante el uso de una duplicación de imagen forense, pero este procedimiento puede ser complicado ciertas veces. La información no volátil que se puede conseguir antes de la duplicación forense es la siguiente:

- Versión del Sistema y el Nivel del Parche
- Sistema de archivos de hora y marca de fecha
- Registro de datos
- La política de auditoría
- Un historial de los inicios de sesión
- Registro de eventos del sistema
- Cuentas de usuario
- Registros de IIS
- Archivos sospechosos

4.4.1 Versión del Sistema y el Nivel del Parche

Es importante saber cuál es el sistema operativo y que parches de seguridad se han instalado en la máquina víctima, sabiendo esto podremos hacernos una idea de que vulnerabilidades tenía el sistema al momento de ser atacado.

Para averiguar esta información utilizamos la herramienta PsInfo distribuido gratuitamente con Pstools que puede ser encontrado en el siguiente vínculo (http://technet.microsoft.com/es-es/sysinternals/bb897550.aspx)

Al ejecutar PsInfo[26] nos mostrará los parches que han sido instalados, si utilizamos el parámetro –h nos mostrará os hotfixes instalados también, si usamos el parámetro –s muestra los software instalados y con el parámetro –d nos muestra el volumen del disco.

```
psinfo –h –s –d
```

[26] ANEXO IV – PsTools Suite

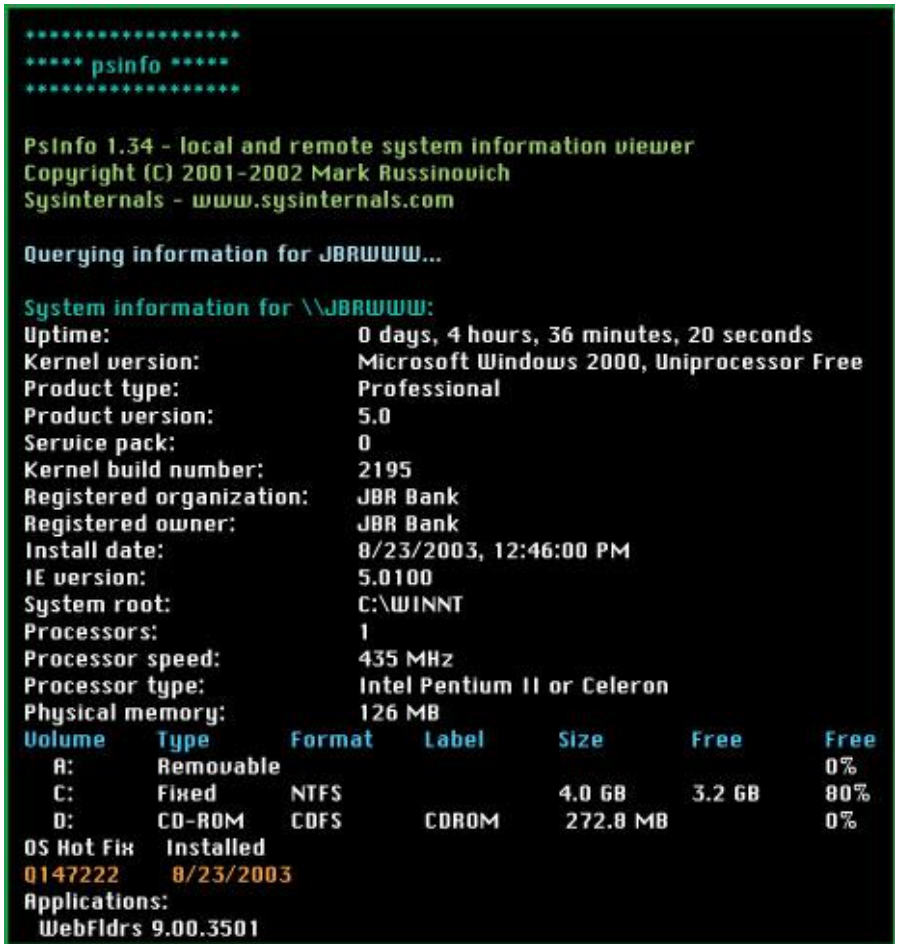


Figura 4-15: Versión del Sistema y Parches Instalados

Podemos observar que solo un hotfix ha sido instalado Q147222. Realizando una rápida búsqueda sabemos que JBRWWW es vulnerable

a una multitud de ataques como el “Unicode”[27] (Bugtraq ID#1806) y el “Double Decode”[28] (Bugtraq ID #2708). Porque estos son ataques dirigidos a los Servidores Web y JBRWWW está ejecutando un Servidor Web (lo sabemos gracias a las salidas de los comandos netstat y fport anteriormente mostrados).

4.4.2 Sistema de Archivos de Tiempo y Marca de Fecha

Para capturar esta información haremos uso de la herramienta llamada UnxUtils package, disponible gratuitamente en el siguiente vínculo (http://unxutils.sourceforge.net/), usaremos el comando find de esa herramienta, este nos mostrará una línea por cada archivo y sus atributos. Por lo que podemos saber información importante como permisos, última fecha y hora de acceso, fecha de modificación, fecha de creación y la ruta completa de cada archivo en la unidad C.

```
find c:\ -printf
"%m;%Ax;%AT;%Tx;%TT;%Cx;%CT;%U;%G;%s;%p\n"
```

Con este comando se está delimitando cada atributo con un punto y coma. Mostraremos la información más relevante encontrada con el uso de este comando en la siguiente tabla:

[27] ANEXO V - Unicode Attack
[28] Anexo VI - Double Decode Attack

Tabla 2: Lista de Archivos mostrados por el comando find

Last Accessed	File Created	Logical Size	Full Path
10/01/03 10:26:23PM	08/23/03 08:14:18AM	8,192	C\WINNT\system32\os2\dll
10/01/03 07:25:07PM	10/01/03 07:25:07PM	13,929	C\WINNT\system32\os2\dll\Configure
10/01/03 07:25:07PM	10/01/03 07:25:07PM	15,427	C\WINNT\system32\os2\dll\COPYING
10/01/03 07:25:07PM	10/01/03 07:25:07PM	68,016	C\WINNT\system32\os2\dll\cygregex.dll
10/01/03 07:25:08PM	10/01/03 07:25:07PM	971,08	C\WINNT\system32\os2\dll\cygwin1.dll
08/23/03 12:43:04PM	12/07/99 07:00:00AM	12,646	C\WINNT\system32\os2\dll\doscalls.dll
10/01/03 07:25:08PM	10/01/03 07:25:08PM	902	C\WINNT\system32\os2\dll\iroffer.cron
10/01/03 07:25:09PM	10/01/03 07:25:08PM	213,3	C\WINNT\system32\os2\dll\iroffer.exe
10/01/03 07:25:09PM	10/01/03 07:25:09PM	2,924	C\WINNT\system32\os2\dll\Makefile.config
10/01/03 07:25:09PM	10/01/03 07:25:09PM		C\WINNT\system32\os2\dll\mybot.ignl
10/01/03 10:26:23PM	10/01/03 07:25:09PM	4	C\WINNT\system32\os2\dll\mybot.ignl.tmp
10/01/03 10:46:22PM	10/01/03 07:25:09PM	25,774	C\WINNT\system32\os2\dll\mybot.log
10/01/03 07:25:09PM	10/01/03 07:25:09PM	168	C\WINNT\system32\os2\dll\mybot.msg
10/01/03 07:36:49PM	10/01/03 07:25:09PM	5	C\WINNT\system32\os2\dll\mybot.pid
10/01/03 10:26:23PM	10/01/03 10:26:23PM	49	C\WINNT\system32\os2\dll\mybot.xdcc
10/01/03 09:56:22PM	10/01/03 09:56:22PM	49	C\WINNT\system32\os2\dll\mybot.xdcc.bkup
10/01/03 10:26:23PM	10/01/03 10:26:23PM	233	C\WINNT\system32\os2\dll\mybot.xdcc.txt
10/01/03 07:25:09PM	10/01/03 07:25:09PM	19,792	C\WINNT\system32\os2\dll\myconfig
10/01/03 07:24:37PM	10/01/03 07:24:37PM	120,32	C\WINNT\system32\os2\dll\nc.exe
08/23/03 12:44:34PM	12/07/99 07:00:00AM	247,86	C\WINNT\system32\os2\dll\netapi.dll
10/01/03 07:25:09PM	10/01/03 07:25:09PM	5,08	C\WINNT\system32\os2\dll\README
10/01/03 07:55:51PM	10/01/03 07:55:51PM	36,864	C\WINNT\system32\os2\dll\samdump.dll

10/01/03 07:25:09PM	10/01/03 07:25:09PM	19,767	C\WINNT\system32\os2\dll\sample.config
10/01/03 07:55:42PM	10/01/03 07:55:42PM	32,768	C\WINNT\system32\os2\dll\setup.exe
10/01/03 07:58:39PM	10/01/03 07:58:38PM	342	C\WINNT\system32\os2\dll\temp.txt
10/01/03 07:52:44PM	10/01/03 07:52:44PM	122,88	C\WINNT\system32\os2\dll\update.exe
10/01/03 07:25:10PM	10/01/03 07:25:10PM	16,735	C\WINNT\system32\os2\dll\WHATSNEW
11/08/03 09:23:54PM	12/07/99 07:00:00AM	108,095	C\WINNT\system32\os2\oso001.009

Podemos observar que la mayoría de los archivos fueron creados la noche del 10/01/2003. Presentaremos otra tabla ordenando por hora de creación donde veremos todos los archivos que fueron creados casi al mismo tiempo.

Tabla 3: Lista de archivos sospechosos creados a la hora de la intrusión

Created Date	Created Time	Logical Size	Full Path
10\01\2003	19:16:30	61440	c:\WINNT\system32\PSEXESVC.EXE
10\01\2003	19:24:37	120320	c:\WINNT\system32\os2\dll\nc.exe
10\01\2003	19:25:07	13929	c:\WINNT\system32\os2\dll\Configure
10\01\2003	19:25:07	15427	c:\WINNT\system32\os2\dll\COPYING
10\01\2003	19:25:07	68016	c:\WINNT\system32\os2\dll\cygregex.dll
10\01\2003	19:25:07	971080	c:\WINNT\system32\os2\dll\cygwin1.dll
10\01\2003	19:25:08	902	c:\WINNT\system32\os2\dll\iroffer.cron
10\01\2003	19:25:08	213300	c:\WINNT\system32\os2\dll\iroffer.exe
10\01\2003	19:25:09	2924	c:\WINNT\system32\os2\dll\Makefile.config
10\01\2003	19:25:09	0	c:\WINNT\system32\os2\dll\mybot.ignl
10\01\2003	19:25:09	0	c:\WINNT\system32\os2\dll\mybot.ignl.bkup
10\01\2003	19:25:09	4	c:\WINNT\system32\os2\dll\mybot.ignl.tmp
10\01\2003	19:25:09	25774	c:\WINNT\system32\os2\dll\mybot.log
10\01\2003	19:25:09	168	c:\WINNT\system32\os2\dll\mybot.msg
10\01\2003	19:25:09	5	c:\WINNT\system32\os2\dll\mybot.pid

10\01\2003	19:25:09	19792	c:\WINNT\system32\os2\dll\myconfig
10\01\2003	19:25:09	5080	c:\WINNT\system32\os2\dll\README
10\01\2003	19:25:09	19767	c:\WINNT\system32\os2\dll\sample.config
10\01\2003	19:25:10	16735	c:\WINNT\system32\os2\dll\WHATSNEW
10\01\2003	19:48:44	0	c:\update.exe
10\01\2003	19:52:44	122880	c:\WINNT\system32\os2\dll\update.exe
10\01\2003	19:55:42	32768	c:\WINNT\system32\os2\dll\setup.exe
10\01\2003	19:55:51	36864	c:\WINNT\system32\os2\dll\samdump.dll
10\01\2003	19:58:38	342	c:\WINNT\system32\os2\dll\temp.txt
10\01\2003	21:56:22	49	c:\WINNT\system32\os2\dll\mybot.xdcc.bkup
10\01\2003	22:22:59	16384	c:\Documents and Settings\Administrator\Application Data\Microsoft\Internet Explorer\MSIMGSIZ.DAT
10\01\2003	22:26:23	49	c:\WINNT\system32\os2\dll\mybot.xdcc
10\01\2003	22:26:23	233	c:\WINNT\system32\os2\dll\mybot.xdcc.txt

Ya sabíamos que el atacante había instalado iroffer y que había establecido una puerta trasera gracias a PsExec, por eso hemos resaltado la información nueva para realizar análisis más adelante, ya que por ahora no sabemos con certeza su propósito.

4.4.3 Datos de los Registros

Tenemos dos líneas de investigación que podemos seguir mediante los rastros de registros, aunque estos registros son demasiado extensos (en el caso de JBRWWW fue más de 7mb), podemos ir rápidamente por estás búsquedas:

- Programas ejecutados en el arranque

- Entradas creadas por las herramientas del intruso

Esta información fue capturada gracias al uso del comando RegDmp sin ningún parámetro. La salida está en formato ASCII por lo que otras herramientas del registro de Windows podrían alterar el contenido.

Después de revisar la salida del RegDmp podemos observar que la sección \HKEY_LOCAL_MACHINE\Software\Microsoft\ Windows\CurrentVersion contiene tres sub-secciones que son de interés para nosotros: Run, RunOnce y RunOnceEx. Cualquier valor con las palabras Run significa programas que serán ejecutados cuando el sistema arranque.

Figura 4-16: Registro de Windows – Programas que se ejecutan en el arranque

Mobsync.exe es un sistema binario, asi que no vemos herramientas sospechosas que se ejecuten al momento del arranque del sistema. Si el atacante fuera más profesional habría localizado el siguiente comando en el registro para automáticamente abrir una puerta trasera.

```
nc –d –L –p 10000 –e C:\winnt\system32\cmd.exe
```

Otra cosa que podemos buscar mediante los registros es cualquier rastro sospechoso de las herramientas del intruso. Este paso no es muy útil en el caso de JBRWWW, ya que solo tenemos u servidor comprometido y realizando una búsqueda en el sistema de archivos con palabras como PsExec o iroffer obtendríamos la misma respuesta, pero en caso de tener un rack amplio de servers nos vendría bien una búsqueda por medio de los registros.

4.4.4 Políticas de Auditoria

La siguiente serie de herramientas que serán ejecutadas dependerán de las políticas de Auditoria de JBRWWW. Sin un apropiado sistema de auditoria no tendremos registros de seguridad relacionados. El comando para determinar las políticas de auditoria es auditpol, el cual es distribuido en los kits de recursos de Windows. La siguiente información es mostrada cuando ejecutamos el comando auditpol.

Figura 4-17: Configuración de las Políticas de Auditoria

No existe ningún evento generado por inicios de sesión u otros eventos relacionados. Es muy común encontrar este tipo de respuesta, ya que la mayoría de empresas no configura las políticas de auditoria por lo que no realizan ningún tipo de registro, esta es una mala señal para nuestra investigación porque perdemos información valiosa aquí.

4.4.5 Historial de Inicios de Sesión

El historial de inicios de sesión puede ser obtenido mediante el uso del comando NTlast, el cual puede ser encontrado en el siguiente vínculo (http://www.mcafee.com/mx/downloads/free-tools/ntlast.aspx)

Figura 4-18: Historial de inicios de sesión

Al ejecutar el comando obtenemos una respuesta desfavorable para nosotros y otra vez es por culpa de no haber configurado las políticas de auditoria.

4.4.6 Registro de Eventos del Sistema

Normalmente tenemos 3 tipos de registros de eventos en una máquina Windows:

- Seguridad
- Aplicación
- Sistema

El comando PsLogList[29] que se encuentra dentro de Pstools nos extraerá los registros en un formato fácil de leer. El siguiente comando nos dará la respuesta que queremos.

```
psloglist –s –x security
```

El parámetro –s nos mostrará la respuesta en una línea por evento haciéndolo más fácil de leer. El parámetro –x le dice que nos muestre

Figura 4-19: Registros de Tipo Seguridad (PsLogList)

[29]ANEXO IV – PsTools Suite

una información detallada de cada evento.

Los registros de tipo seguridad serían una información muy útil en caso de haber configurado las políticas de auditoria.

Los registros de tipo aplicación contienen información generada por las aplicaciones instaladas, algunos eventos son informacionales mientras que otros solo muestran fallos de aplicación. Todo lo que nos mostró fue la instalación de programas estándares en el sistema empezando en Agosto 23, 2003.

Figura 4-20: Registros de Tipo Aplicación (PsLogList)

Los Registro de tipo sistema, contienen mensajes de los servicios del sistema. Donde podremos observar fallos de drivers de dispositivos, conflictos con direcciones IP, entre otras. La salida solo nos muestra mensajes creados por el sistema de uso estándar. Al parecer no hemos obtenido ninguna información útil mediante los registros.

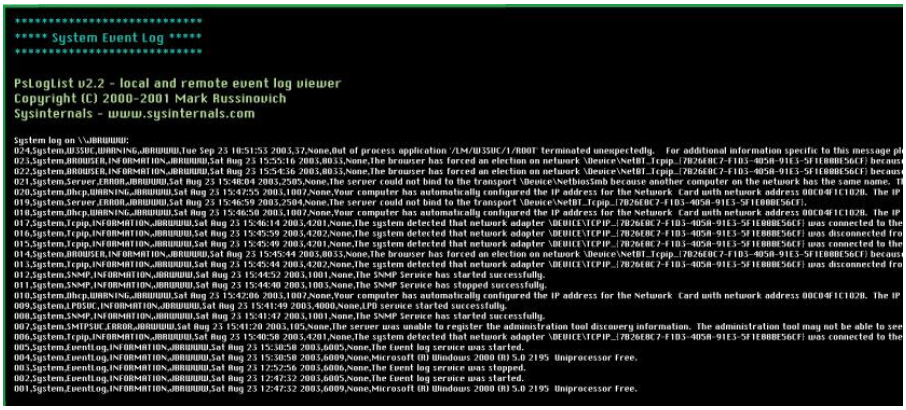


Figura 4-21: Registros de Tipo Sistema (PsLogList)

4.4.7 Cuentas de Usuarios

La puerta trasera de un intruso más fácil de usar es aquella que parece tráfico normal en la máquina víctima. Por lo tanto, sería normal que el atacante traté de crear un nuevo usuario con permisos válidos para ejecutar sus herramientas. Para extraer la información de los usuarios podemos utilizar el comando pwdump, al ejecutarlo obtenemos la siguiente respuesta:

Figura 4-22: Cuentas de Usuario de JBRWWW (pwdump)

Tenemos 4 usuarios donde Administrator es una cuenta de súper usuario (RID 500) que cada sistema debe tener. La cuenta Guest esta deshabilitado, pero existe en todos los sistemas de Windows. Los otros dos usuarios IUSER_JBRWWW e IWAM_JBRWWW son cuentas de usuarios normales que procesan el Servidor Web IIS. Estas cuentas están para limitar el daño que un atacante podría causar a un sistema a través de un ataque basado en Web. Vemos que no hay otras cuentas de interés en JBRWWW.

4.4.8 Registros IIS

La mayoría de ataques actualmente se realizan a través del puerto TCP 80 (HTTP), esto es debido a que hay millones de servidores que necesitan pasar información a través de este puerto, ya que no podemos bloquear lo que debemos permitir, convierte a este puerto en una posible entrada para nuestro atacante.

El servidor Web IIS escribe cualquier actividad a los registros en C:\winnt\system32\logfiles que es el directorio por defecto. Dentro de este directorio encontramos otro nombrado W3SVCn, donde n es el número único del Servidor Web y es numerado dependiendo de los dominios que tenga. JBRWWW solo posee un dominio así que encontramos el directorio W3SVC1. Dentro de este directorio encontramos dos archivos ex030923.log y ex031001.log. Cada archivo contiene actividad del servidor Web de todo el día, estos archivos son nombrados mediante el formato ffyymmdd.log donde ff es el formato, yy es el año, mm es el mes, y dd es el día. Por lo tanto sabemos que tenemos información de los día Septiembre 9, 2003 y Octubre 1, 2003.

Al abrir el archivo ex030923.log encontramos el siguiente encabezamiento:

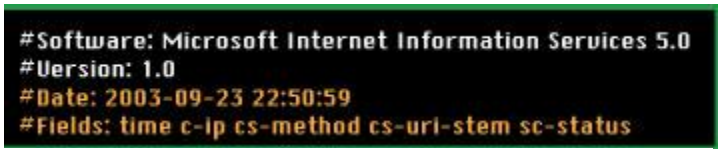


Figura 4-23: Registros generados por el servidor Web (IIS) 1

La hora y fecha resaltada, nos indica que este informe fue reportado en GMT, no EDT (huso horario local de JBRWWW). Esta información es útil

cuando manejas muchos informes de auditoría. La segunda línea nos muestra los campos que están siendo registrados.

Mientras empezamos a analizar estos registros notamos varios accesos muy interesantes de parte de la IP 95.16.3.79, la rapidez con que se realizan estos accesos son demasiado rápidos como para ser realizados por un ser humano. La cuarta línea nos muestra unas palabras interesantes embebidas.

Figura 4-24: Registros generados por el servidor Web (IIS) 2

Nikto es un servidor Web q brinda una herramienta de escaneo de vulnerabilidades, lo podemos encontrar en el siguiente vínculo (http://cirt.net/). Asumimos que fue esta herramienta la que accedió varias veces en un corto periodo de tiempo al servidor JBRWWW, otro dato interesante es el último número 404, siempre que posean un número por encima de los 400s quiere decir que el intento no tuvo éxito, si el numero esta entre los 200s quiere decir que si tuvo éxito. Tras revisar el log de Septiembre 9, vemos que toda la actividad vino de una

sola dirección IP en menos de un minuto. JBRWWW fue víctima de un escaneo de vulnerabilidades ese día.

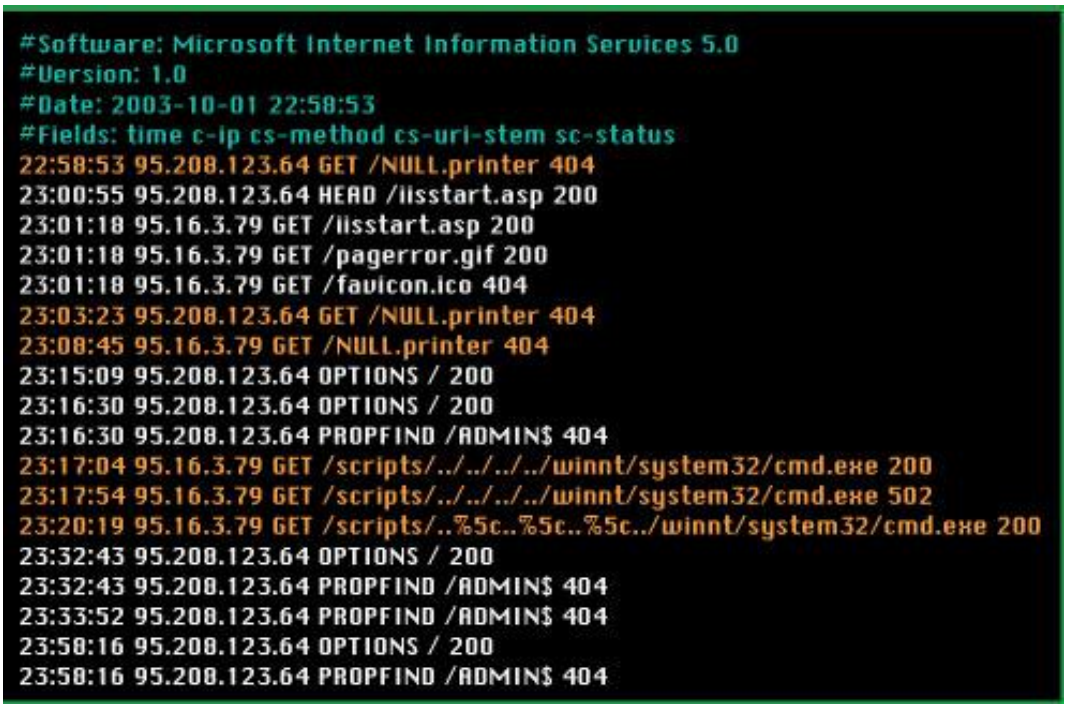


Figura 4-25: Registros generados por el servidor Web (IIS) 3

La primera línea resaltada es un signo revelador “.printer” un desbordamiento de búfer de Microsoft Windows 2000 de la dirección IP 95.208.123.64. Pero al parecer no tuvo éxito, por lo general si este ataque tiene éxito causa un crash en el sistema, así la actividad no es registrada en los Informes de IIS. Las siguientes 4 líneas son accesos de

usuarios normales quizás para saber si el sistema está en línea o no. Luego tenemos otro grupo de líneas resaltadas que muestran intentos fallidos de ejecutar ".printer" desde las IPs 95.208.123.64 y 95.16.3.79. Al ver estas mismas IPs creemos que pertenecen a la misma persona o a varias personas trabajando juntos.

El último grupo de líneas resaltadas nos muestra ataques con éxito porque tienen los números 200 y 502 al final. Podemos ver que alguien accedió a C:\winnt\system32\cmd.exe. Un servidor Web nunca debería acceder al comando Shell cmd.exe, ahora sabemos que la dirección IP 95.208.123.64 logró ejecutar comandos en JBRWWW bajo el contexto de un usuario no administrador IUSR_JBRWWW. Las primeras dos líneas resaltadas muestran que es un ataque Unicode. La última muestra un ataque Doble Decode. Ambos ataques son ataques de directorio transversales para lograr que el Servidor Web ejecute programas arbitrarios en la máquina víctima. Porque JBRWWW no habilitó más campos en los registros de IIS no podemos saber que comandos se ejecutaron en el comando Shell.

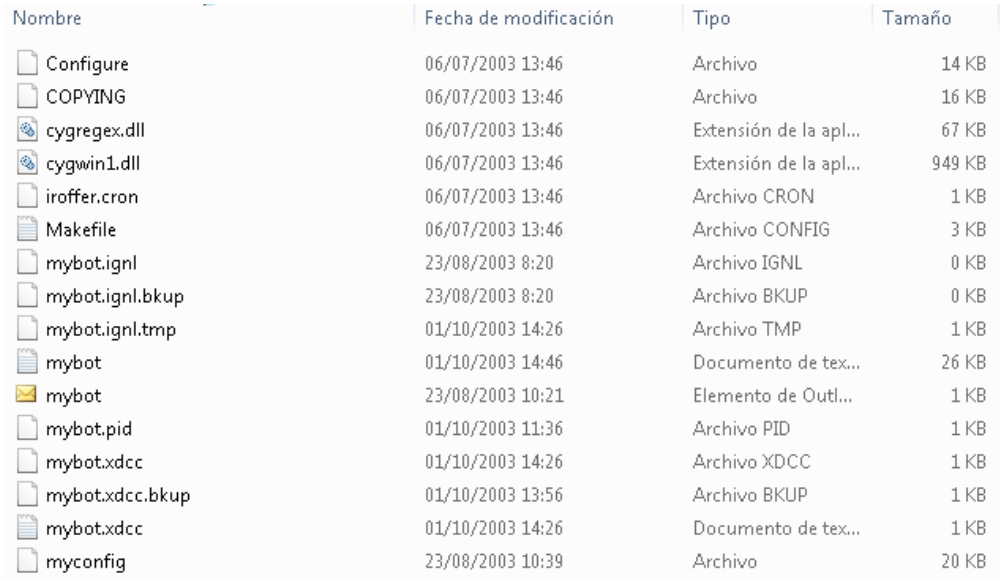
4.4.9 Archivos Sospechosos

En caso de que nos vayamos a adquirir un duplicación forense de JBRWWW, podríamos transferir todos los archivos sospechosos aprovechado la sesión netcat que se había abierto para actividades anteriores.

Para transferir un archivo llamado datos a nuestra sesión de netcat debemos utilizar un comando parecido a este

```
type datos I nc ip_address 2222
```

Los archivos que creemos sospechosos porque fueron credos durante la intrusión son los siguientes:



Nombre	Fecha de modificación	Tipo	Tamaño
Configure	06/07/2003 13:46	Archivo	14 KB
COPYING	06/07/2003 13:46	Archivo	16 KB
cygregex.dll	06/07/2003 13:46	Extensión de la apl...	67 KB
cygwin1.dll	06/07/2003 13:46	Extensión de la apl...	949 KB
iroffer.cron	06/07/2003 13:46	Archivo CRON	1 KB
Makefile	06/07/2003 13:46	Archivo CONFIG	3 KB
mybot.ignl	23/08/2003 8:20	Archivo IGNL	0 KB
mybot.ignl.bkup	23/08/2003 8:20	Archivo BKUP	0 KB
mybot.ignl.tmp	01/10/2003 14:26	Archivo TMP	1 KB
mybot	01/10/2003 14:46	Documento de tex...	26 KB
mybot	23/08/2003 10:21	Elemento de Outl...	1 KB
mybot.pid	01/10/2003 11:36	Archivo PID	1 KB
mybot.xdcc	01/10/2003 14:26	Archivo XDCC	1 KB
mybot.xdcc.bkup	01/10/2003 13:56	Archivo BKUP	1 KB
mybot.xdcc	01/10/2003 14:26	Documento de tex...	1 KB
myconfig	23/08/2003 10:39	Archivo	20 KB

Figura 4-26: Lista de Archivos Sospechosos

4.4.10 Recuento de la evidencia capturada en vivo

El objetivo principal de este análisis era identificar si hubo o no hubo un incidente. Con la información volátil y no volátil durante la captura de datos en vivo nos indica que una intrusión no autorizada si ocurrió.

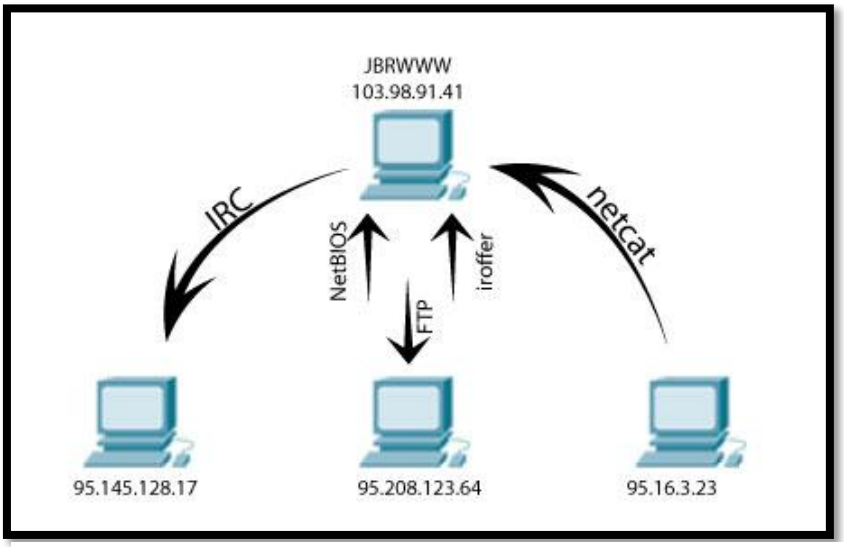


Figura 4-27: Conexiones de Red durante la Intrusión a las 9:08 de 01/Oct/03

A pesar de que no teníamos los registros de los eventos de seguridad de Windows, los registros IIS indicaron que JBRWWW fue escaneado con una herramienta conocida como Nikto a las 6:51:17PM el 23 de Septiembre del 2003, desde la dirección IP 95.16.3.79.

Aproximadamente unos 18 segundos antes del escaneo, una página Web por defecto de IIS fue accedida desde la dirección IP 95.16.3.23. Es común antes y después de un ataque revisar el estado del sitio web para acceder a tal página. Esto puede indicar que el atacante tenía acceso o control del sistema en 95.16.3.73 o quizás estaba trabajando con alguien más quien lo hizo.

Luego el 1 de Octubre del 2003 un atacante desde la dirección IP 95.208.123.64, posiblemente trabajando en conjunto con 95.16.3.79, inició un ataque exitoso Unicode después de haber fallado con el desbordamiento de búfer del ".printer".

Aunque los detalles no han sido determinados, parece que los atacantes fueron capaces de ejecutar comandos en JBRWWW a través del ataque Unicode IIS y estableció una sesión FTP de regreso en nuestros sistemas. Además fueron capaces de instalar netcat e iroffer en el directorio C:\WINNT\system32\os2\dll. Esta figura nos mostrará una secuencia general de las actividades basadas en la información anteriormente analizada.

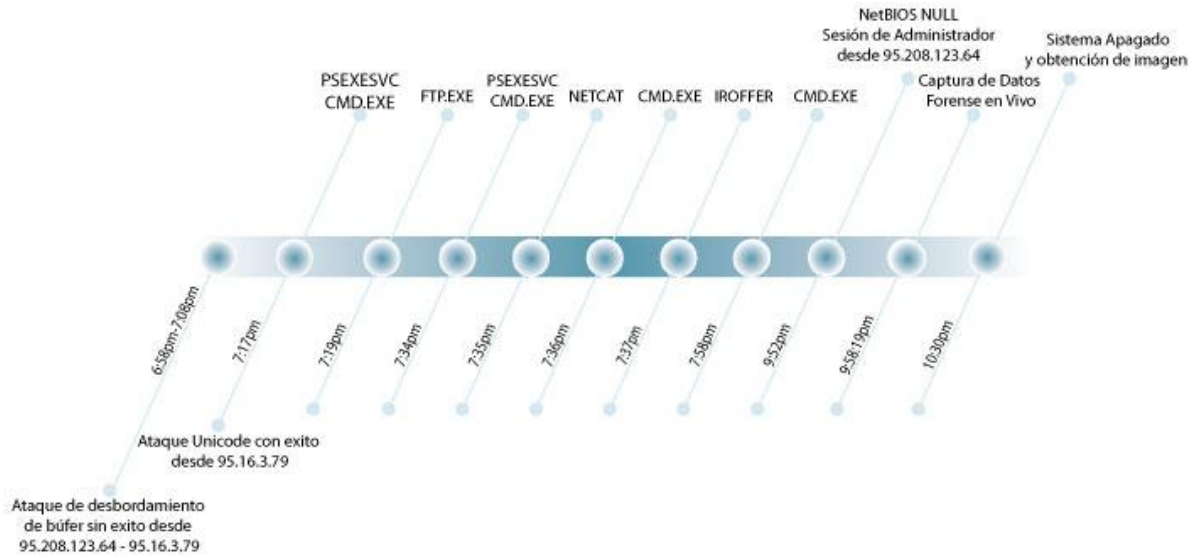


Figura 4-28: Línea de tiempo del ataque el 01/Oct/03

4.5 Análisis de la Evidencia basada en red

La información que se analizará a continuación fue capturada por una plataforma Linux que estaba monitoreando el sistema víctima 103.98.91.41. Se utilizó el siguiente comando tcpdump para capturar la información necesaria

```
tcpump –n –i etho –s 1515 –w capture_file.lpc
```

El equipo de respuesta a incidentes del Banco JBR separó este archivo en 2 pequeños archivos utilizando TCPslice dejándonos dos archivos llamados s2a.lpc y s2b.lpc a los que llamaremos primer y segundo rastro respectivamente.

4.5.1 Primer Rastro (s2a.lpc)

4.5.1.1 Primer Rastro: Datos Estadísticos

Empezaremos analizando el archivo s2a.lpc, para poder visualizar el contenido utilizamos la herramienta tcpdstat con el siguiente comando:

Tcpdstat s2a.lpc > s2a.tcpdstat.txt

```
DumpFile:  s2a.lpc
FileSize: 8.21MB
Id: 200309231852
StartTime: Tue Sep 23 18:52:29 2003
EndTime:   Tue Sep 23 18:55:26 2003
TotalTime: 177.22 seconds
TotalCapSize: 7.81MB  CapLen: 1514 bytes
# of packets: 26084 (7.81MB)
AvgRate: 1.10Mbps  stddev:0.56M

### IP flow (unique src/dst pair) Information ###
# of flows: 5  (avg. 5216.80 pkts/flow)
Top 10 big flow size (bytes/total in %):
 85.6% 14.3%  0.1%  0.0%  0.0%

### IP address Information ###
# of IPv4 addresses: 4
Top 10 bandwidth usage (bytes/total in %):
 100.0% 99.9%  0.1%  0.0%
### Packet Size Distribution (including MAC headers) ###
<<<<
 [   32-    63]:        3238
 [   64-   127]:        15281
 [  128-   255]:        1836
 [  256-   511]:        368
 [  512-  1023]:        1779
 [ 1024-  2047]:        3582
>>>>


### Protocol Breakdown ###
<<<<
     protocol      packets            bytes                bytes/pkt
------------------------------------------------------------------------
[0]  total         26084 (100.00%)    8187014 (100.00%)    313.87
[1]  ip            26084 (100.00%)    8187014 (100.00%)    313.87
[2]  tcp           26077 ( 99.97%)    8186206 ( 99.99%)    313.92
[3]  http(s)       11344 ( 43.49%)    6914617 ( 84.46%)    609.54
[3]  http(c)       11491 ( 44.05%)    1076775 ( 13.15%)    93.71
[3]  squid         4 (  0.02%)        240 (  0.00%)        60.00
[3]  smtp          3 (  0.01%)        180 (  0.00%)        60.00
[3]  nntp          2 (  0.01%)        120 (  0.00%)        60.00
[3]  ftp           2 (  0.01%)        120 (  0.00%)        60.00
[3]  pop3          2 (  0.01%)        120 (  0.00%)        60.00
[3]  imap          2 (  0.01%)        120 (  0.00%)        60.00
[3]  telnet        2 (  0.01%)        120 (  0.00%)        60.00
[3]  ssh           2 (  0.01%)        120 (  0.00%)        60.00
[3]  dns           2 (  0.01%)        120 (  0.00%)        60.00
[3]  bgp           2 (  0.01%)        120 (  0.00%)        60.00
[3]  napster       2 (  0.01%)        120 (  0.00%)        60.00
[3]  realaud       2 (  0.01%)        120 (  0.00%)        60.00
[3]  rtsp          2 (  0.01%)        120 (  0.00%)        60.00
[3]  other         3213 ( 12.32%)     193074 (  2.36%)     60.09
[2]  udp           4 (  0.02%)        618 (  0.01%)        154.50
[3]  other         4 (  0.02%)        618 (  0.01%)        154.50
[2]  icmp          3 (  0.01%)        190 (  0.00%)        63.33
>>>>
```

Figura 4-29: Datos estadísticos del primer rastro

Este archivo capturado por tcpdump pesa alrededor de los 8mb, tiene una hora de inicio de martes 23 de septiembre del 2003 a las 18:52:29. Este archivo de captura es de solo 3 minutos de longitud, podemos observar mucha actividad Web (Puerto TCP 80), con uno o más servidores web enviando 43.49% de todos los paquetes y os clientes enviando un 44.05% de los paquetes y en la

categoría “otro” que es el tercer contribuyente más alto con 12.32%. Esta última categoría podría contener dos escenarios: podría ser un protocolo no reconocido por tcpdstat, o varios protocolos no reconocidos. Los demás protocolos que nos muestra es normal encontrarlos tan bajos con la ejecución de un escaneo de puertos.

4.5.1.2 Primer Rastro: Datos de Alerta

Para el siguiente análisis estaremos usando la herramienta Snort, esta herramienta es capaz de encontrar patrones de actividad maliciosa. Para ejecutar Snort en modo batch para analizar el archivo s2a.lpc usamos el siguiente comando:

```
Snort –c /usr/local/etc/snort.conf –r s2a.lpc –b –l /var/rdf/s2a
```

El modo batch se lo usa cuando la información no es en vivo, sino que es un archivo anteriormente capturado. Este comando lo que hace es decirle que ejecute un IDS, cargando su configuración de snort.conf, le pide que registre el archivo s2a.lpc en modo binario –b y lo guarde en el directorio /var/rdf/s2. Esto nos dejará 2 archivos llamados alert y snort.log.TIMESTAMP.

Analizando el archivo alert.s2a encontramos varias alertas y las primeras 7 son las siguientes:

```
[**] [1:1668:5] WEB-CGI /cgi-bin/ access [**]
[Classification: Web Application Attack] [Priority: 1]
09/23-18:52:49.322441 95.16.3.79:51767 -> 103.98.91.41:80
TCP TTL:63 TOS:0x0 ID:13538 IpLen:20 DgmLen:181 DF
***AP*** Seq: 0xA50D689C  Ack: 0x1EDB04F1  Win: 0x8218  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1462497415 0

[**] [1:1201:6] ATTACK RESPONSES 403 Forbidden [**]
[Classification: Attempted Information Leak] [Priority: 2]
09/23-18:52:49.421027 103.98.91.41:80 -> 95.16.3.79:51773
TCP TTL:128 TOS:0x0 ID:516 IpLen:20 DgmLen:386 DF
***AP*** Seq: 0x1EDFB033  Ack: 0x7E945F39  Win: 0x43EF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 120360 1462497415

[**] [1:1852:3] WEB-MISC robots.txt access [**]
[Classification: access to a potentially vulnerable web application] [Priority: 2]
09/23-18:52:49.949036 95.16.3.79:51779 -> 103.98.91.41:80
TCP TTL:63 TOS:0x0 ID:13608 IpLen:20 DgmLen:183 DF
***AP*** Seq: 0x5170F14  Ack: 0x1EE5B1E6  Win: 0x8218  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1462497416 0
[Xref => http://cgi.nessus.org/plugins/dump.php3?id=10302]

[**] [1:1145:6] WEB-MISC /~root access [**]
[Classification: Attempted Information Leak] [Priority: 2]
09/23-18:52:50.384827 95.16.3.79:51781 -> 103.98.91.41:80
TCP TTL:63 TOS:0x0 ID:13619 IpLen:20 DgmLen:178 DF
***AP*** Seq: 0xCF382597  Ack: 0x1EE90111  Win: 0x8218  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1462497417 0

[**] [1:1497:6] WEB-MISC cross site scripting attempt [**]
[Classification: Web Application Attack] [Priority: 1]
09/23-18:52:50.837038 95.16.3.79:51786 -> 103.98.91.41:80
TCP TTL:63 TOS:0x0 ID:13647 IpLen:20 DgmLen:221 DF
***AP*** Seq: 0x5BDDD224  Ack: 0x1EEEE5E5  Win: 0x8218  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1462497418 0

[**] [1:1122:4] WEB-MISC /etc/passwd [**]
[Classification: Attempted Information Leak] [Priority: 2]
09/23-18:52:50.889165 95.16.3.79:51788 -> 103.98.91.41:80
TCP TTL:63 TOS:0x0 ID:13659 IpLen:20 DgmLen:232 DF
***AP*** Seq: 0x8C660F99  Ack: 0x1EF09726  Win: 0x8218  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1462497418 0

[**] [1:1113:4] WEB-MISC http directory traversal [**]
[Classification: Attempted Information Leak] [Priority: 2]
09/23-18:52:51.035194 95.16.3.79:51797 -> 103.98.91.41:80
TCP TTL:63 TOS:0x0 ID:13713 IpLen:20 DgmLen:229 DF
***AP*** Seq: 0xA78D0CAA  Ack: 0x1EF88EA8  Win: 0x8218  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1462497418 0
[Xref => http://www.whitehats.com/info/IDS297]
```

Figura 4-30 Datos de Alerta del primer rastro (Snort)

Podemos notar que esas alertas tienen la misma clasificación “Attempted Information Leak” o “Web Application Attack”. La dirección IP de origen es 95.16.3.79 y la de destino es 103.98.41, esta última pertenece al Web Server. Estos eventos están pasando rápidamente, con varios ataques lanzados cada centésima de segundo, esto es claramente un ataque de reconocimiento contra el Web Server. Dado que todo está siendo enfocado contra el puerto TCP 80, podemos concluir que el intruso está buscando vulnerabilidades en el Web Server, él podría estar usando herramientas como Nikto http://www.cirt.net/nikto2.

Al final de este archivo encontramos un patrón diferente de alertas, al parecer el ataque de reconocimiento terminó con la alerta *WEB-CGI tst.bat alert at 09/23-18:53:26.166980*. Empezando con la alerta etiquetada como *ICMP PING NMAP at 09/23-18:55:18.604340*, una nueva clase de ataque comienza.

Al parecer el intruso lanzó algún tipo de reconocimiento general en contra del Web Server, demostrado por todas las alertas anteriores que estaban enfocadas al puerto TCP 80, pero en las siguientes alertas veremos actividad usando ICP(Internet Control Message Protocol),

SNMP (Simple Network Management Protocol), SOCKS (un servidor Proxy), entre otros protocolos y servicios. Tras haber hecho el reconocimiento general, el atacante ahora está buscando por algún servicio para aprovechar alguna debilidad de este.

Figura 4-31: Datos de Alerta del primer rastro (Snort)2

Esta segunda ronda esta simplemente chequeando por otros servicios activos. Por lo tanto es razonable asumir que el atacante está usando la herramienta de escaneo Nmap (http://nmap.org/) para lograr su reconocimiento.

Snort ha reconocido ciertos aspectos de paquetes ofensivos comparando la forma que usa Nmap, también reporta archivos con combinaciones extrañas de flags como URG, PSH, SYN, y FIN en el paquete con el timestamp *09/23-18:55:24.394939*. Este y otros paquetes nos hacen creer que el intruso está usando Nmap para perfilar el Web Server.

4.5.1.3 Primer Rastro: Datos de Sesión

Una vez habiendo comprobado la actividad consistida en reconocimiento, vale la pena dar una rápida revisión a la información de sesión en busca de anomalías.

La información mostrada a continuación son extractos del archivo s2a.argus.all, con el cual hemos ilustrado sus partes más relevantes en las siguientes imágenes:

Date	Time	Proto	Source IP. Port		Dest IP. Port	SrcPkts	DstPkts	SrcBytes	DstBytes	SessionClose
23 Sep 03	18:52:29	tcp	95.16.3.23.1044	->	103.98.91.41.80	6	7	906	4909	EST
23 Sep 03	18:53:15	tcp	95.16.3.79.53236	->	103.98.91.41.80	6	6	545	3791	FIN
23 Sep 03	18:53:15	tcp	95.16.3.79.53237	->	103.98.91.41.80	6	6	544	3965	FIN
23 Sep 03	18:53:15	tcp	95.16.3.79.53238	->	103.98.91.41.80	6	6	591	3965	FIN
23 Sep 03	18:53:15	tcp	95.16.3.79.53239	->	103.98.91.41.80	6	6	551	3965	FIN
23 Sep 03	18:53:15	tcp	95.16.3.79.53240	->	103.98.91.41.80	6	6	593	3965	FIN

Figura 4-32: Datos de Sesión del primer rastro (Argus) 1

Todo este tráfico es dirigido al puerto TCP 80, principalmente con un patrón de seis o siete paquetes enviado de la fuente y destino en cada caso. Esto es consistente para el escaneo de vulnerabilidades en la Web, Argus nos muestra la mayoría de conexiones cerradas con el paquete FIN. Más adelante encontramos un patrón sospechoso a las 18:55:18.

Date	Time	Proto	Source IP. Port		Dest IP. Port	SrcPkts	DstPkts	SrcBytes	DstBytes	SessionClose
23 Sep 03	18:55:18	tcp	95.16.3.79.47990	->	103.98.91.41.1359	1	1	54	54	RST
23 Sep 03	18:55:18	tcp	95.16.3.79.47990	->	103.98.91.41.305	1	1	54	54	RST
23 Sep 03	18:55:18	tcp	95.16.3.79.47990	->	103.98.91.41.698	1	1	54	54	RST
23 Sep 03	18:55:18	tcp	95.16.3.79.47990	->	103.98.91.41.155	1	1	54	54	RST
23 Sep 03	18:55:18	tcp	95.16.3.79.47990	->	103.98.91.41.937	1	1	54	54	RST
23 Sep 03	18:55:18	tcp	95.16.3.79.47990	->	103.98.91.41.764	1	1	54	54	RST
23 Sep 03	18:55:18	tcp	95.16.3.79.47990	->	103.98.91.41.1669	1	1	54	54	RST

Figura 4-33: Datos de Sesión del primer rastro (Argus) 2

Esas líneas nos demuestran claramente un escaneo de puertos, ya que los puertos destino van cambiando aleatoriamente. El origen envía un paquete y el destino responde con un paquete, a diferencia de los paquetes anteriormente mostrados estas sesiones son cerradas con RST (estas ni siquiera son sesiones, ya que el método de three-ways handshake nunca fue completado). Podemos asumir que todos los

puertos escaneados aquí fueron cerrados porque el origen envió u solo paquete SYN y el destino respondió con un solo paquete ACK RST. Cuando el puerto escaneado está abierto vemos un patrón como el siguiente:

Date	Time	Proto	Source IP. Port		Dest IP. Port	SrcPkts	DstPkts	SrcBytes	DstBytes	SessionClose
23 Sep 03	18:55:20	tcp	95.16.3.79.47990	->	103.98.91.41.80	2	1	108	58	RST
23 Sep 03	18:55:20	tcp	95.16.3.79.47990	->	103.98.91.41.1027	2	1	108	58	RST
23 Sep 03	18:55:19	tcp	95.16.3.79.47990	->	103.98.91.41.135	2	1	108	58	RST
23 Sep 03	18:55:19	tcp	95.16.3.79.47990	->	103.98.91.41.139	2	1	108	58	RST

Figura 4-34: Datos de Sesión del primer rastro (Argus)3

Aquí el origen envía dos paquetes y el destino envía uno. El origen probablemente envió un paquete SYN, al cual el destino respondió con un paquete ACK SYN. Parra terminar el intento de conexión, el origen envía un RST.

Observando la información de Argus notamos que las horas no van de la más vieja a la más nueva. Esto es porque Argus mantiene sus tablas en memoria y las escribe al disco cuando se les acabe el tiempo o cuando Argus decide que la sesión está terminada. Esto no cambia nuestras conclusiones, pero explica porque algunas actividades parecen estar mal generadas y repetidas.

Hemos averiguado que un reconocimiento ocurrió y no tuvimos que observar los paquetes crudos en el archivo Libpcap.

Observemos estos paquetes para determinar qué tipo de reconocimiento fue ejecutado. Observemos primero el archivo snort.log. s2a, donde las copias de los paquetes que generaron alertas en Snort están almacenadas.

Los primeros dos paquetes están asociados con WEB-CGI /cgi-bin/ acceso y alertas de respuestas de ataque mostradas anteriormente.

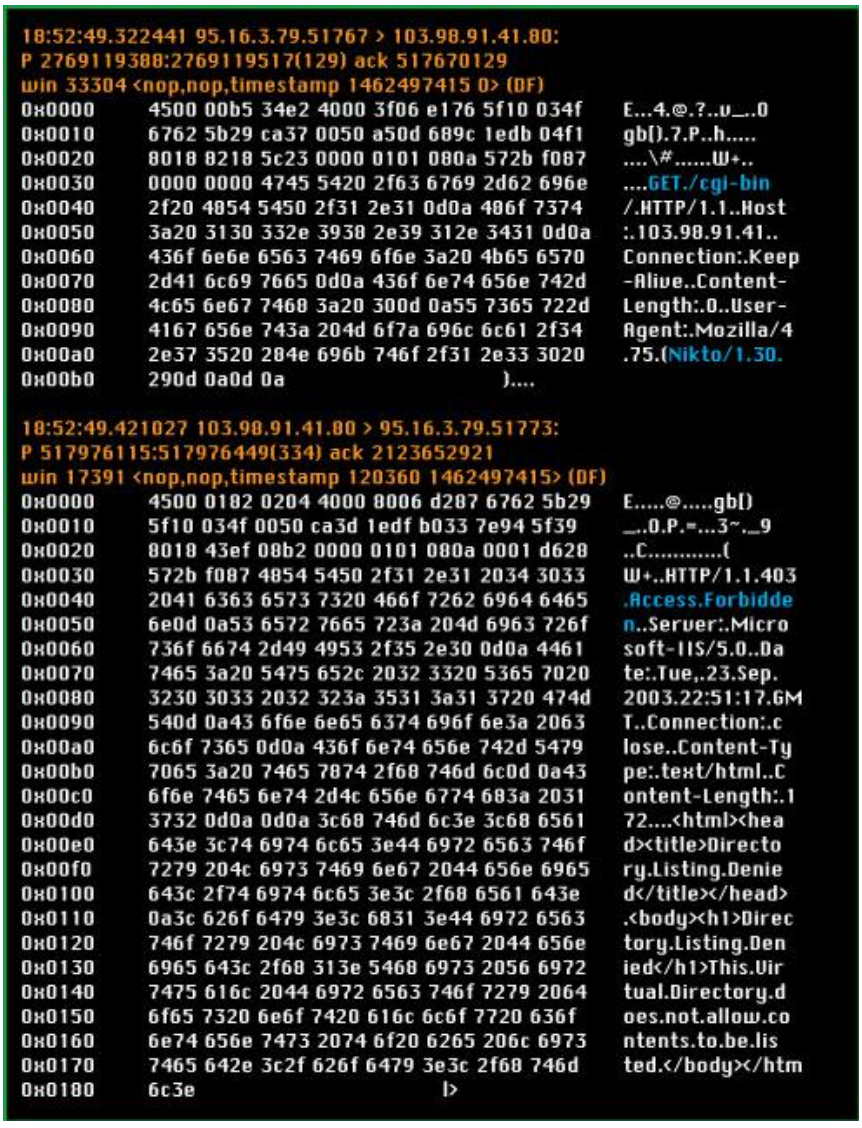


Figura 4-35: Datos de sesión del primer rastro en formato crudo

Hemos resaltado algunos campos para esta revisión. El primer campo resaltado, GET/cgi-bin/, nos muestra que un intruso estuvo intentando determinar si el directorio estaba presente en el Web Server. Esta es una vulnerabilidad común porque algunos CGI (Common Gateway Interface) son raramente escritos, ofreciendo oportunidades a los atacantes. El segundo campo resaltado, Nikto /1.30, confirma nuestras sospechas del uso de la herramienta Nikto en contra del Web Server. El último campo resaltado, 403 Access Forbidden, es la respuesta del Web Server a sus esfuerzos de reconocimiento. Esta alerta nos muestra que Snort puede ser programado no solo para observar actividad de intrusos sino que también monitorear respuestas de víctimas.

4.5.1.4 Primer Rastro: Datos Íntegros

Para confirmar nuestra interpretación del reconocimiento observado a través de las alertas de Snort y los datos de sesión de Argus, regresamos al archivo original Libpcap s2a.lpc. Los siguientes paquetes demuestran como el atacante envía un solo paquete SYN, al que el puerto responde con un solo RST ACK.

Encontramos un posible rastro del uso de la herramienta en hexadecimal 5555 5555 5555 o en ASCII UUUUUU. Una Búsqueda en google nos demuestra que se han observado los mismos caracteres asociados con la herramienta de reconocimiento Nmap.

Hemos resaltado estos elementos en la siguiente imagen:

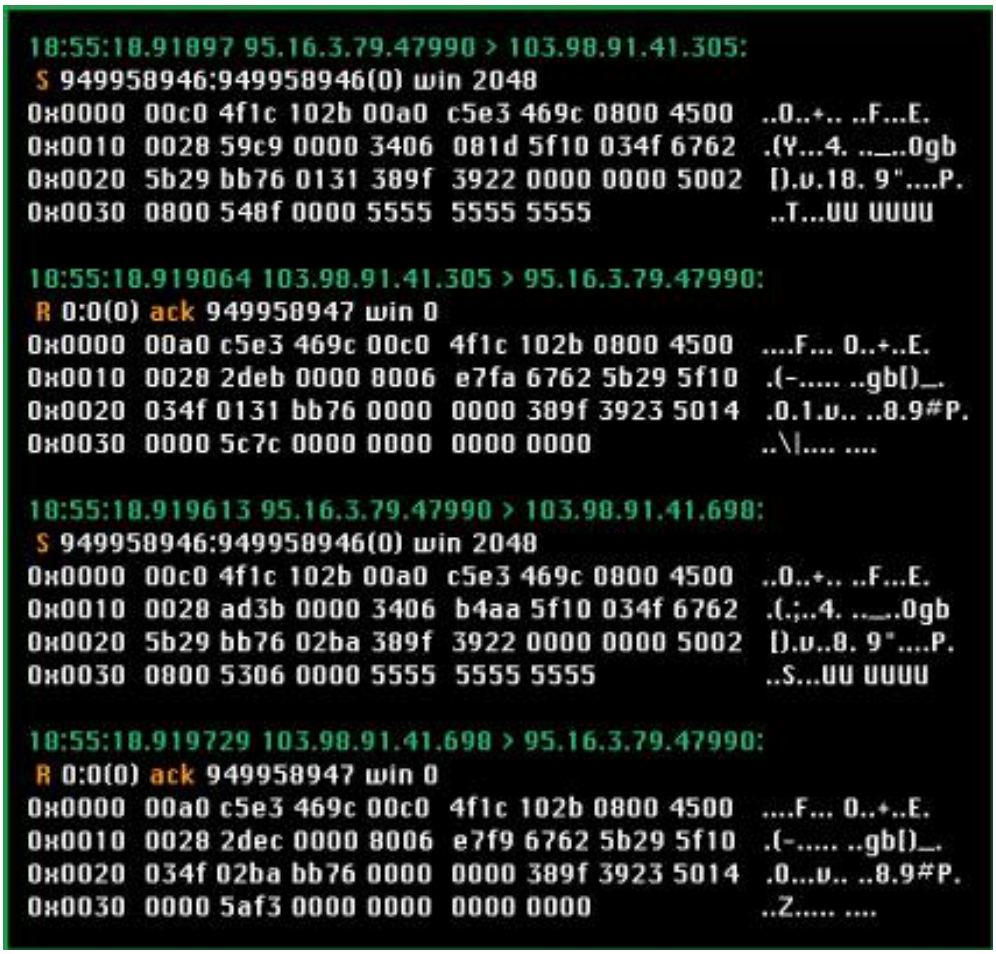


Figura 4-36: Datos íntegros del primer rastro

Comparamos el modelo de solicitud-respuesta con los siguientes paquetes y pudimos concluir que en puertos cerrados, el origen envía un paquete y recibe un paquete de respuesta del destino; y en puertos abiertos el origen envía dos paquetes y el destino responde con un paquete. Luego de realizar un análisis al archivo s2a.lpc hemos logrado agrupar los paquetes asociados a este ataque y los presentaremos agrupados según sus puertos.

Figura 4-37: Paquetes durante el ataque de reconocimiento

Tras haber analizado nuestro primer rastro sabemos que un atacante usando la dirección IP 95.16.379 ejecutó un ataque de reconocimiento basado en puertos y servicios en contra del Web Server 103.98.91.41

4.5.2 Segundo Rastro (s2b.lpc)

4.5.2.1 Segundo Rastro: Datos Estadísticos

Mostraremos el resultado de la salida de Tcpdstat:

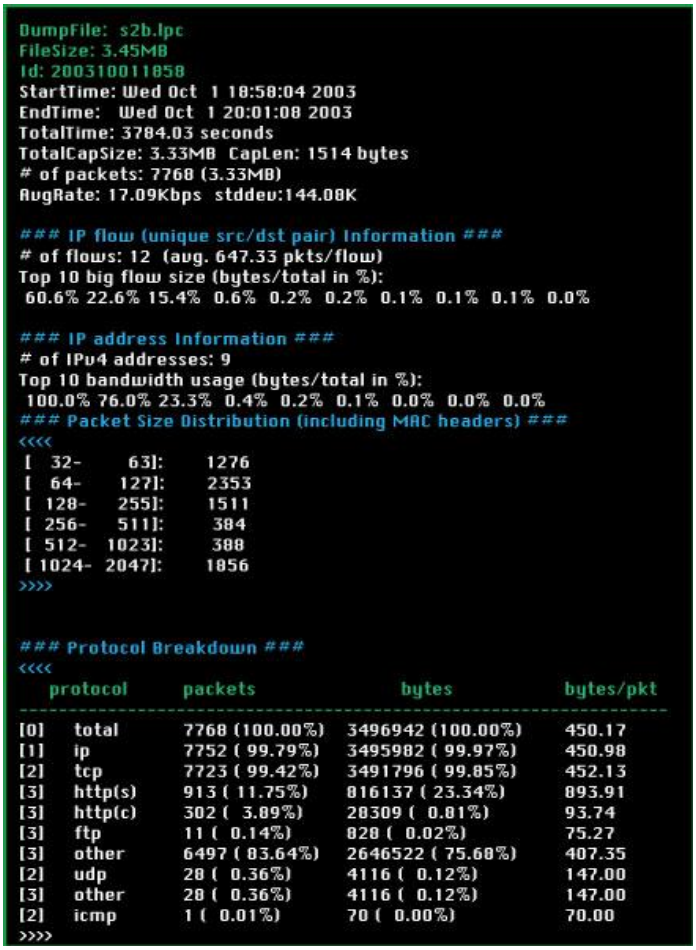


Figura 4-38: Datos estadísticos del segundo rastro (tcpdstat)

Estas estadísticas del archivo s2b.lpc son mucho más pequeñas que las del archivo s2a.lpc, pero son más oscuros. A pesar de que pocos protocolos están listados, la mayoría de la acción aparece en la categoría “otros”, contando con el 83.64% de todo el tráfico. Esto demuestra que es importante saber cómo las herramientas funcionan cuando interpretas su salida. ¿Qué protocolos puede reconocer Tcpdstat exactamente? La respuesta la tenemos dentro del archivo stat.c que se encuentra junto las demás evidencias que tenemos y los mostraremos a continuación:

```
{" tcp    ",    2},
{"  http(s)",   3},
{"  http(c)",   3},
{"  squid  ",   3},
{"  smtp   ",   3},
{"  nntp   ",   3},
{"  ftp    ",   3},
{"  pop3   ",   3},
{"  imap   ",   3},
{"  telnet ",   3},
{"  ssh    ",   3},
{"  dns    ",   3},
{"  bgp    ",   3},
{"  napster",3},
{"  realaud",3},
{"  rtsp   ",   3},
{"  icecast",   3},
{"  hotline",3},
{"  other  ",   3},
{" udp    ",    2},
{"  dns    ",   3},
{"  rip    ",   3},
{"  mcast  ",3},
{"  realaud",3},
{"  halflif",    3},
{"  starcra",   3},
{"  everque",3},
{"  unreal ",3},
{"  quake  ",3},
{"  cuseeme",3},
{"  other  ",   3},
{" icmp   ",    2},
{" igmp   ",    2},
{" ospf   ",    2},
{" ipip   ",    2},
{" ipsec  ",    2},
{" ip6    ",    2},
{" pim    ",    2},
{" sctp   ",    2},
{" other  ",    2},
{" frag   ",    2},
{"ip6     ",    1},
{" tcp6   ",    2},
{"  http(s)",   3},
{"  http(c)",   3},
{"  squid  ",   3},
{"  smtp   ",   3},
{"  nntp   ",   3},
{"  ftp    ",   3},
{"  pop3   ",   3},
{"  imap   ",   3},
{"  telnet ",   3},
{"  ssh    ",   3},
{"  dns    ",   3},
{"  bgp    ",   3},
{"  napster",3},
{"  realaud",3},
{"  rtsp   ",   3},
{"  icecast",   3},
{"  hotline",3},
{"  other  ",   3},
{" udp6   ",    2},
{"  dns    ",   3},
{"  rip    ",   3},
{"  mcast  ",3},
{"  realaud",3},
{"  halflif",    3},
{"  starcra",   3},
{"  everque",3},
{"  unreal ",3},
{"  quake  ",3},
{"  cuseeme",3},
{"  other  ",   3},
{" icmp6  ",2},
{" ospf6  ",    2},
{" ip4    ",    2},
{" ip6    ",    2},
{" ipsec6 ",    2},
{" hbhopt6 ",2},
{" rtopt6 ",    2},
{" dstopt6 ",2},
{" pim6   ",    2},
{" sctp6  ",    2},
{" other6 ",2},
{" frag6  ",    2},
{"other   ",    1}
```

Estos son todos los protocolos TCP y UDP que TcpdStat reconoce. Notamos que Tcpdstat no sabe cómo interpretar protocolo de Windows NetBios/Server Message Block (SMB), el cual se ejecuta en los puertos TCP 139 y 445 o en los puertos UDP 137 y 138. A pesar de estas limitaciones, TcpdStat provee un útil resumen del tráfico.

Aunque este archivo solo pesa 4MB, la duración de la trama es mucho más larga. La trama empieza el miércoles 1 de octubre a las 18: 58:04 del 2003 y termina el miércoles 1 de octubre a las 20:01:08 del 2003. Otra diferencia del primer rastro es la presencia de 16 paquetes no IP. Podemos ver esto comprando el número total de paquetes (7768) al número de paquetes IP (7752). Aunque esto podría ser un protocolo como NetBEUI para viejas redes de Windows o IPX para clientes Novell, es probablemente tráfico ARP (Address Resolution Protocol).

4.5.2.2 Segundo Rastro: Datos de Alerta

Esta vez el archivo de alerta fue generado con un archivo de estadísticas extra al final, esto fue logrado gracias al siguiente comando:

```
snort – c usr/local/etc/snort.conf –r s2b.lpc –b –l /var/rdf/s2b 2>&1 >
snort.stats
```

Observamos la sección de paquetes procesados de snort y podemos ver lo siguiente:

Figura 4-39: Paquetes procesados por Snort (s2b.lpc)

Podemos confirmar que los 16 paquetes no contados por TcpdStat son ARP, paquetes usados para resolver direcciones IP a MAC (Media Access Control) o direcciones de hardware. Los otros números concuerdan con el reporte de TcpdStat. Este es un buen ejemplo de que es una buena práctica usar más de una herramienta para validar resultados.

Procederemos a revisar las alertas generadas en el archivo alert.s2b

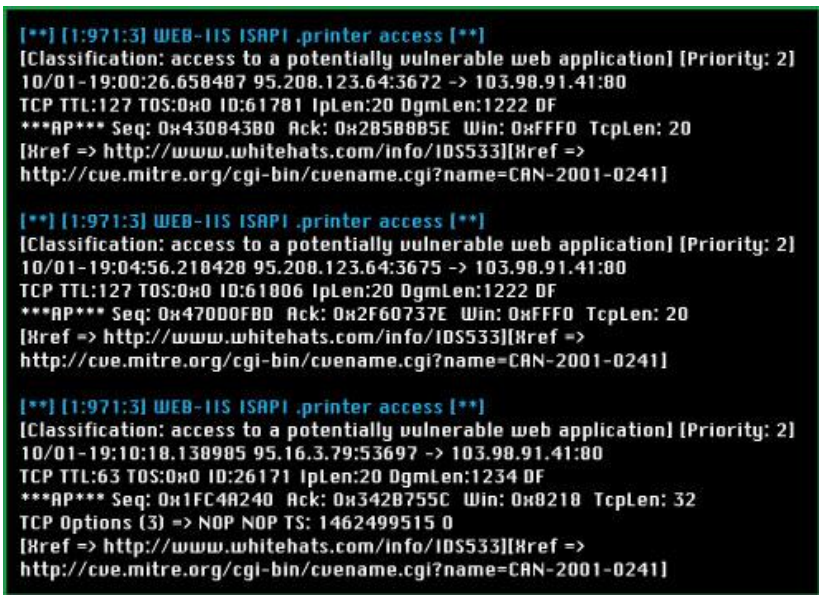


Figura 4-40: Datos de Alerta del segundo rastro (s2b.lpc) 1

Chequeando en la base de datos de la CVE (Common Vulnerabilities and Exposure), encontramos que CAN-2001-0241 es descrito como sigue[30]:

El desbordamiento de búfer de impresiones en internet con la extensión ISAPI en Windows 2000 permite al atacante ganar privilegios de root a través de una larga solicitud de impresión la cual es pasada a la extensión a través de IIS5.0.

[30] (CVE)

Esto no parece nada bueno, notamos gracias al timestamp que el segundo evento sucedió 4 minutos después y el tercer evento 6 minutos después del segundo, por lo cual indica que el intruso intentó el ataque una vez y falló, frustrado intentó el ataque dos veces más.

Además notamos que estos eventos no fueron realizados desde la misma dirección IP que usó para realizar el reconocimiento. La dirección IP usada para el reconocimiento fue 95.16.3.79, mientras que la dirección IP usada ahora es 95.208.123.64. Tenemos dos posibilidades que explican esta situación, podemos tener un atacante con varias máquinas o tenemos un grupo de atacantes.

Tras las 3 primeras alertas de WEB-IIS ISAPI.printer, observamos una docena de alertas como la siguiente:

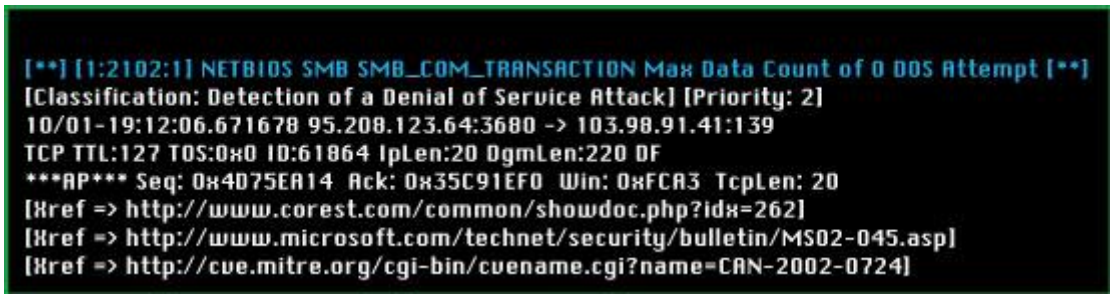


Figura 4-41: Datos de Alerta del segundo rastro (s2b.lpc) 2

ealizando una rápida búsqueda en internet sobre esta alerta encontramos lo siguiente[31]:

SMB (Server Message Block) es el protocolo que Microsoft usa para compartir archivos, impresoras, puertos seriales y también comunicación entre computadoras usando tubos nombrados y ranuras de correo. En un ambiente de red, los servidores hacen disponibles los sistemas de archivos y recursos para los clientes. Los clientes hacen solicitudes SMB para los recursos y los servidores hacen las respuestas SMB.

Enviando un paquete de solicitud especial, un atacante puede montar un ataque de denegación de servicio en el servidor y detener el sistema. El atacante podría usar una cuenta de usuario o una cuenta anónima para acceder y cumplir esto. Aunque no está confirmado aún, el atacante podría ejecutar comandos arbitrarios.

Esto significa que el frustrado intruso empezó un intento de denegación de servicio en contra del Web Server a través de SMB, no podemos estar seguros en este momento. Más adelante podremos afirmar o desacreditar esta teoría. Por ahora, examinemos el último set

[31] (Core Security Technologies, 2011)

de alertas del archivo s2b.lpc que le siguen al último evento de alertas SMB.

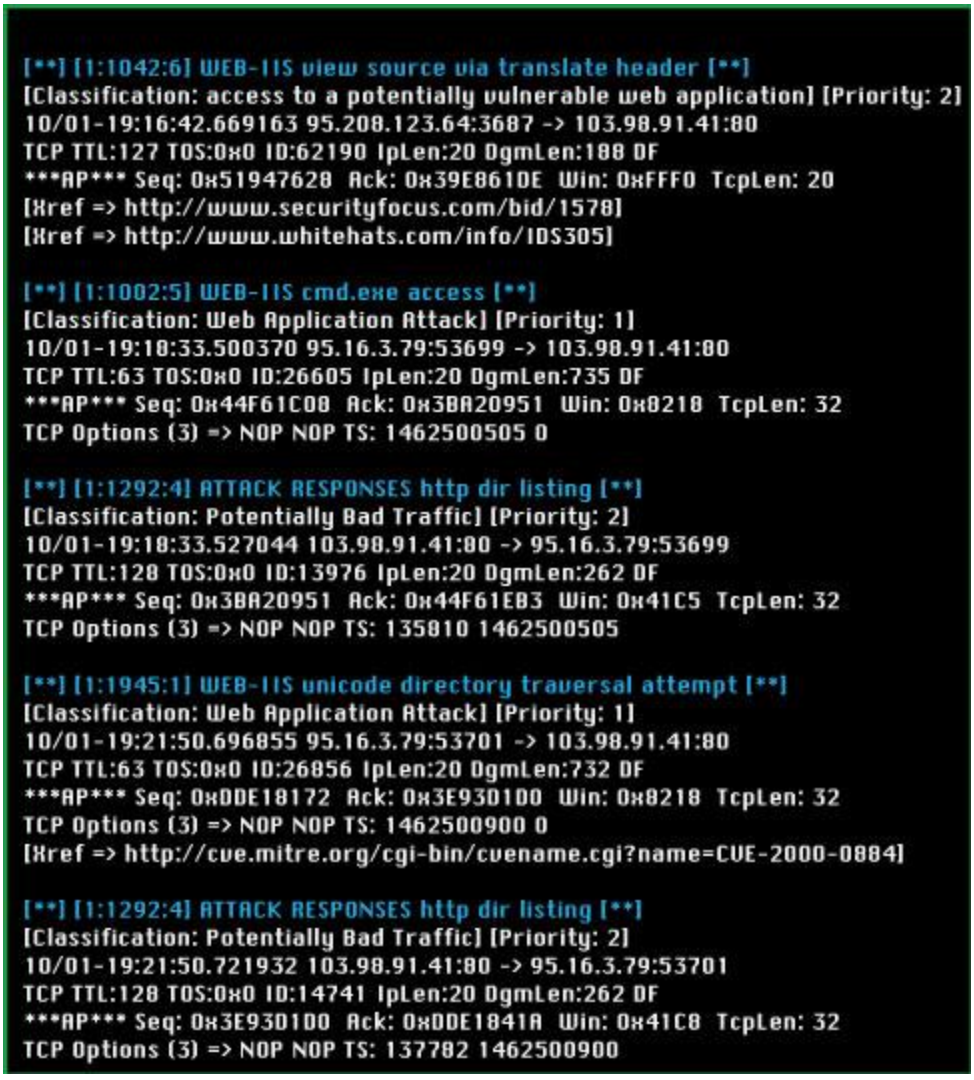


Figura 4-42: Datos de Alerta del segundo rastro (s2b.lpc) 3

Para poder comprender esta alerta, citamos a las vulnerabilidades almacenadas en la base de datos en http://www.securityfocus.com/bid/1578/discuss [32]

Microsoft IIS 5.0 tiene un motor de encriptamiento dedicado para tipos de archivos avanzados como archivos ASP, ASA, HTR, etc. Este motor maneja varias solicitudes para estos tipos de archivos, los procesa y luego los ejecuta en el server.

Es posible forzar al server para enviar de regreso la fuente conocida de encriptamiento de estos archivos al cliente si la solicitud HHTP GET contiene una cabecera especializada con "Translate:f" al final de esta, y si un "/" es utilizado al final de la URL. El motor de encriptamiento será capaz de localizar el archivo solicitado, de alguna manera lo reconocerá como si fuera un archivo que necesita ser procesado y procederá a enviar archivos fuente al cliente.

Esto parece ser un tipo de intento de reconocimiento, donde el cliente puede forzar al server a revelar los contenidos de archivos en el Web Server. Estas alertas WEB-IIS con traducción en la cabecera son enviadas desde la dirección IP 95.208.123.64. Las siguientes alertas

[32] (Symantec Corporation, 1999)

son más problemáticas que esta, empezando con que se originan de una tercera dirección IP 95.16.3.79, también podemos ver que tienen un mensaje más serio. Estas alertas que contienen el Web-IIS cmd.exe sabemos que es un exploit que ha causado muchos daños a algunos sistemas de administrador. El intruso ha logrado de alguna manera forzar el comando Shell cmd.exe de Windows a mostrar los contenidos de un directorio a través del servicio Web.

Las últimas dos alertas proveen detalles adicionales de cómo el intruso obtuvo este acceso. La cuarta alerta titulada "WEB-IIS Unicode directory traversal attempt" nos lleva a investigar y encontramos lo siguiente:

IIS 4.0 y 5.0 permiten a atacantes leer documentos afuera de la raíz de la web, y posiblemente ejecutar comandos arbitrarios, a través de las URLs que contienen caracteres UNICODE.[33]

Esta vulnerabilidad arrasó los Web Servers con Microsoft Internet Information Server (IIS) durante el año 2000. El hecho de que el intruso forzó al Web Server a mostrar un listado de directorios significa que el sistema es vulnerable a muchos más problemas.

[33] (Paller, 2012)

4.5.2.3 Segundo Rastro: Datos de Sesión

Este archivo fue generado de la misma manera que el anterior, solo que esta vez se generan dos archivos para poder filtrar el tráfico de Web y servicios SMB, por ende tenemos un archivo llamado s2b.argus.all, el cual procederemos a analizar y otro archivo llamado s2b.argus.no_80_137_138_139_445_arp.argus, el cual posee el tráfico filtrado.

Hemos resumido el amplio archivo de registros a estos 10 registros que procederemos a analizar:

Date	Time	Proto	Souce Ip Port		Dest. Ip Port	SrcPckt	DstPackt	SrcByte	DstByte	Session Close
01 Oct 03	19:20:46	tcp	103.98.91.41.1033	->	95.208.123.64.21	5	3	288	269	EST
01 Oct 03	19:36:49	tcp	95.208.123.64.21	?>	103.98.91.41.1033	2	1	193	54	FIN
01 Oct 03	19:38:25	icmp	103.98.91.1	->	103.98.91.41	1	0	70	0	SRC
01 Oct 03	19:37:48	tcp	95.16.3.23.1048	->	103.98.91.41.60906	14	10	878	3195	EST
01 Oct 03	19:38:57	tcp	95.145.128.17.32830	->	103.98.91.41.113	1	1	74	54	RST
01 Oct 03	19:38:22	tcp	103.98.91.41.1034	->	63.98.19.242.6667	3	0	186	0	TIM
01 Oct 03	19:38:36	tcp	103.98.91.41.1089	->	63.98.19.242.6667	3	0	186	0	TIM
01 Oct 03	19:38:57	tcp	95.16.3.23.1048	->	103.98.91.41.60906	4	4	216	508	EST
01 Oct 03	19:38:57	tcp	103.98.91.41.1174	->	95.145.128.17.6667	31	34	2259	3564	EST
01 Oct 03	19:40:09	tcp	95.208.123.64.3753	->	103.98.91.41.1465	22	19	1232	5319	EST

Figura 4-43: Datos de sesión del segundo rastro (argus)

Podemos observar que el Web Server victima 103.98.91.41 creó una conexión de salida FTP a la dirección 95.208.123.64, la cual fue el origen de las alertas de acceso WEB-IIS ISAPI.printer. La segunda sesión es la respuesta del servidor FTP, y la tercera sesión es solo un paquete ICMP echo. El cuarto registro de sesión, con la hora y fecha de 01 Oct 03 19:37:48, nos muestra a la dirección 95.16.3.23

conectándose al puerto 60906 en el Web Server. Más adelante averiguaremos que se ejecuta en este puerto.

Lo siguiente que vemos es una nueva dirección IP 95.145.128.17, envió un solo paquete al puerto 113 TCP del Web Server. Este es probablemente un paquete de solicitud de identificación (IDENT), comúnmente usado con e-mail o servidores de Internet Relay Chat (IRC). Puede ser la respuesta a una conexión solicitada por el Web Server a la dirección IP 95.145.128.17. Siguiendo a este evento, vemos al Web Server intentar establecer sesiones en el puerto TCP 6667 con la dirección IP 63.98.19.242 Esta sesión tiene el flag TIM al final que nos indica que el tiempo de espera terminó, esto significa que 63.98.19.242 nunca respondió. Al parecer la victima quiere hablar a un servidor IRC.

La penúltima entrada es extremadamente importante. Observamos a la víctima conectarse con éxito a 95.145.128.17 en el puerto 6667. Esto indica que el intruso forzó al Web Server a hablar con IRC con esperanzas de controlarlo a través de este medio de comunicación.

La última entrada nos muestra a la dirección IP 95.208.123.64 conectarse al puerto TCP 1465 en el servidor víctima. Podría ser algún tipo de puerta trasera, aún no sabemos con certeza su objetivo.

4.5.2.4 Segundo Rastro: Datos Íntegros

Esta vez se utilizará la herramienta Tcpflow para reconstruir sesiones de interés en los puertos TCP 21, 60906, 1465 y 6667.

```
tcpflow –r s2b.lpc port 21 or port 60906 or port 1465 or port 6667
```

Tcpflow nos produce 8 archivos para nuestra investigación:

- 095.016.003.023.01048-103.098.091.041.60906
- 095.145.128.017.06667-103.098.091.041.01174
- 095.208.123.064.00021-103.098.091.041.01033
- 095.208.123.064.03753-103.098.091.041.01465
- 103.098.091.041.01033-095.208.123.064.00021
- 103.098.091.041.01174-095.145.128.017.06667
- 103.098.091.041.01465-095.208.123.064.03753
- 103.098.091.041.60906-095.016.003.023.01048

La sintaxis muestra la dirección IP origen y el puerto seguidos por la dirección IP destino y el puerto. Estos ocho archivos son los datos

enviados por el origen y el destino para 4 sesiones por separado. Analizaremos cada una más adelante.

Primero examinaremos el contenido del archivo 103.098.091.041.01033-095.208.123.064.00021, el cual es la conexión de salida a un servidor FTP desde el Web Server víctima.

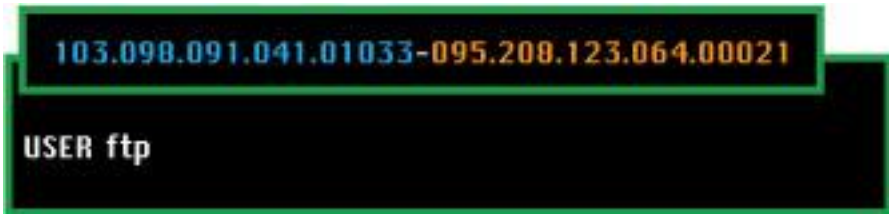


Figura 4-44: Contenido de la sesión en el puerto 21(solicitando)

Al parecer el intruso no logró mucho con esta sesión. Veamos que respondió el server

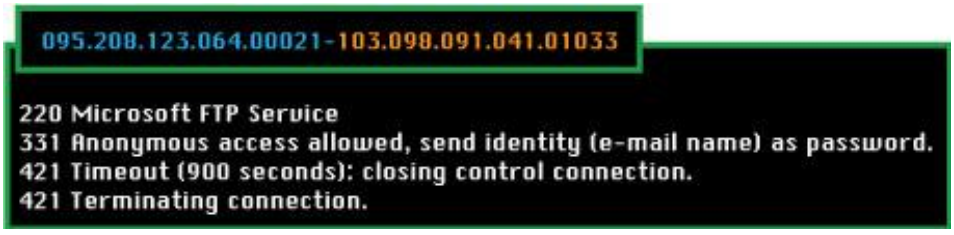


Figura 4-45: Contenido de la sesión en el puerto 21 (respondiendo)

El servidor FTP tampoco tenía mucho que decir. Al parecer el intruso no tuvo mucha suerte intentando colocar archivos en este servidor FTP.

Ahora veremos el tráfico causado por el intruso usando el puerto 60906, en el archivo 095.016.003.023.01048-103.098.091.041.60906

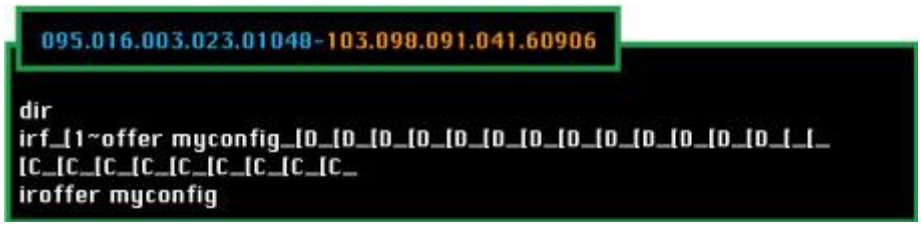


Figura 4-46: Contenido de la sesión en el puerto 60906(solicitando)

Estos parecen comandos, veamos que encontramos en la respuesta.

Figura 4-47: contenido de la sesión en el puerto 6906 (respondiendo)

En este momento podemos confirmar que el atacante ha estado por encima de la víctima. Él ya ha cargado varios archivos dentro del directorio C:\WINNT\system32\0s2\dll. Observamos nc.exe o netcat resaltado en el listado de archivos, el cual puede ser usado para crear puertas trasera basadas en TCP. De hecho este tráfico puede ser resultado de una sesión de netcat. Esta data también explica los intentos de conexión IRC porque observamos el uso del programa iroffer.exe

Ahora observaremos el tráfico del archivo 103.098.091.041.01174-095.145.128.017.06667 para ver el tráfico del cliente al servidor IRC.

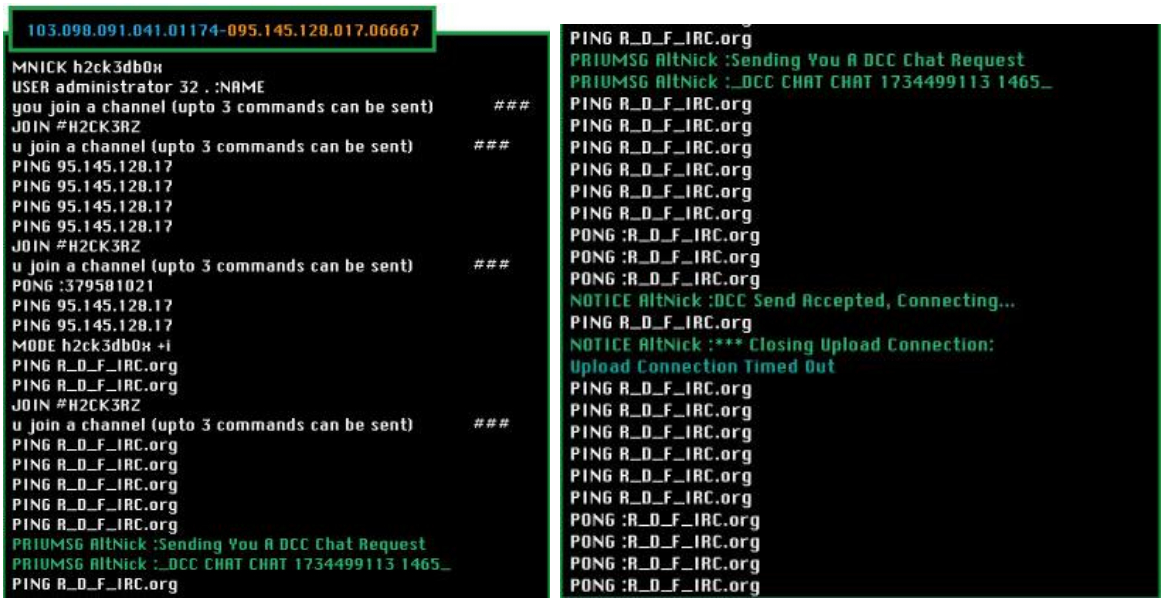


Figura 4-48: Contenido de la sesión en el puerto 6667 (solicitando)

Estos parecen ser comandos estándar IRC. A través del Web Server, el intruso usa h2ck3db0x como su Nick en IRC. Él se une al canal #H2CK3RZ. Las primeras dos líneas resaltadas AltNick quieren hablar directamente al cliente IRC en el Web Server. Esto indica que el real intruso está intentando comunicarse con su nueva víctima a través de IRC. Las dos últimas entradas resaltadas muestran un intento fallido para enviar data directamente a través de DCC, pero el tiempo de subida se agota. Esto es debido a una configuración erróneo ya sea del cliente o del server, ya que el canal no está bloqueado ni por un firewall ni por un router de filtrado.

Lo siguiente que haremos será leer la respuesta del servidor IRC, listado en el archivo 095.145.128.017.06667-103.098.091.041.1174.

Hemos resaltado las partes más relevantes en la imagen, la primera línea resaltada indica que AltNick, también conocido como Helevius, envió un mensaje privado al cliente IRC desde el Web Server víctima.

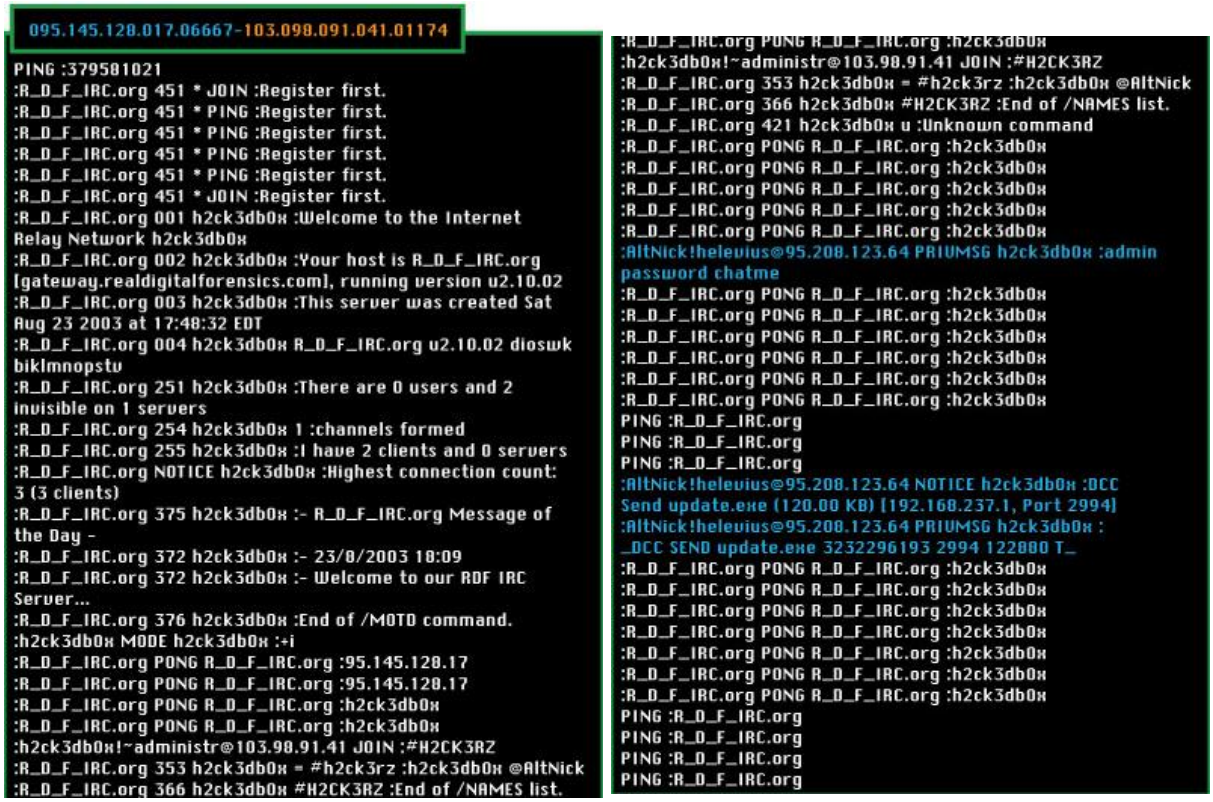


Figura 4-49: Contenido de la sesión en el puerto 6667 (respondiendo)

Las últimas dos entradas resaltadas muestran un intento de transferir el archivo update.exe desde el cliente IRC usado por AltNick hacia la víctima. Parece ser que el intruso intentó localizar el archivo update.exe en el Web Server, pero falló.

La última sesión de interés que nos queda por examinar es la realizada desde 95.208.123.64 con el puerto TCP 1465 hacia 103.98.91.41. Examinemos la información enviada desde

95.208.123.64 al sistema víctima, listado en el archivo 095.208.123.064.03753-103.098.091.041.01465.

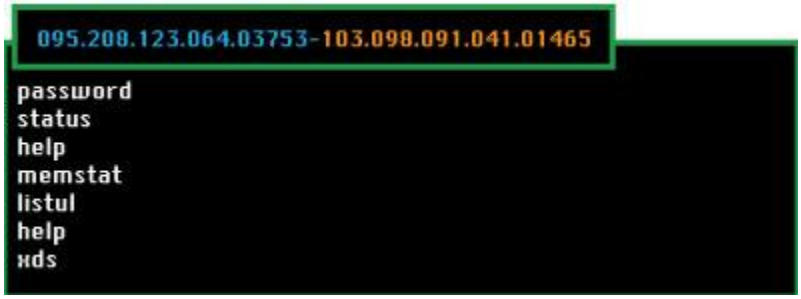


Figura 4-50: Contenido de la sesión en el puerto 1465 (solicitando)

Esto no es muy informativo, pero parecen ser comandos. Ahora observemos la respuesta del cliente mostrada en el archivo 103.098.091.041.01465-095.208.123.064.03753.

Por la extensión del archivo hemos omitido ciertas secciones poco relevantes, dejando solo las partes importantes de esta sesión.

En este archivo podremos observar respuestas enviadas desde el programa iroffer.exe ejecutándose en 103.98.91.41. Tras la sintaxis de ayuda (la cual hemos omitido por su extensión), veremos varias líneas de interés. Al inicio de la salida veremos que fue recién iniciada. También veremos un listado de directorios, mostrando que

el intruso tiene visibilidad del Web Server usando iroffer. También podemos observar entradas relacionadas a los intentos de transferencia del archivo update.exe, el cual falló por el agotamiento de espera de conexión.

Hemos avanzado mucho en la investigación, pero aún no sabemos

Figura 4-51: Contenido de la sesión en el puerto 1465(respondiendo)

cómo el intruso logró instalar iroffer.exe en la víctima o cómo él lo inició. Analicemos el archivo s2b.ngrep.psexec generado por ngrep en búsqueda de alguna línea de caracteres sospechosa.

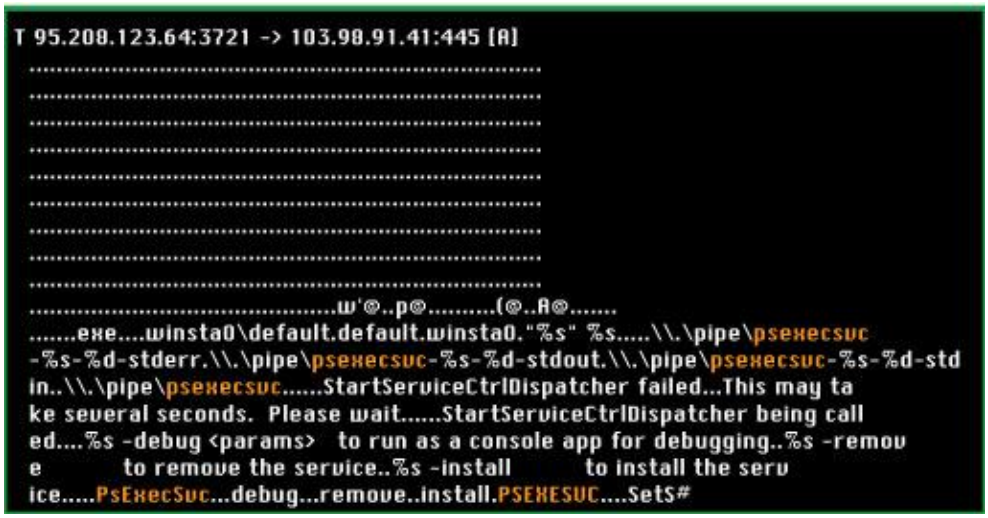


Figura 4-52: Fragmento del archivo s2b.ngrep.pseexec

PsExec fue claramente involucrado en esta intrusión. Desafortunadamente, ninguna herramienta gratis o pagada, hasta donde sabemos, tiene la capacidad de leer y reconstruir sesiones SMB. Podríamos utilizar tcpflow para reconstruir conversaciones SMB, pero no podríamos tener una copia clara de cualquier archivo transferido a través del compartimiento de SMB.

PsExec requiere un nivel administrativo para establecer una conexión con el sistema víctima. ¿Cómo el intruso ganó este nivel

de acceso? Debido a la complejidad involucrada para interpretar el tráfico SMB, no podemos estar seguros. Mirando a la información cruda, hay indicaciones de que el intruso pudo haber enumerado información en el servidor víctima, pero no podemos estar seguros. La evidencia basada en host podría ser más conclusiva. En algunos casos, un solo registro de eventos basados en host puede proveer una mejor vista que cientos de paquetes.

4.5.3 Recuento de la evidencia basada en red

Pudimos observar como el atacante concentró su actividad en contra del Web Server 103.98.91.41 y no atacó ninguna otra máquina del Banco JBR. El ejecutó su reconocimiento desde la dirección IP 95.16.3.23 y ataques desde las direcciones IP 95.208.123.64 y 95.16.3.79. Aseguramos que el atacante usó PsExec para ganar acceso interactivo al sistema y lo usó para transferir un programa llamado iroffer.exe. Vemos que iroffer.exe fue usado para acceder a un servidor IRC en la dirección IP 95.145.128.17.

Toda la evidencia basada en red fue derivada de archivos Libpcap creados con Tcpdump. A pesar de que algunas organizaciones coleccionan información de esta manera algunas otras no lo hacen, aun

así se pudo recrear la información estadística, de alerta y de sesión de los dos archivos Libpcap iniciales; existen otros métodos para capturar esta información sin necesidad de coleccionar todo el tráfico en crudo.

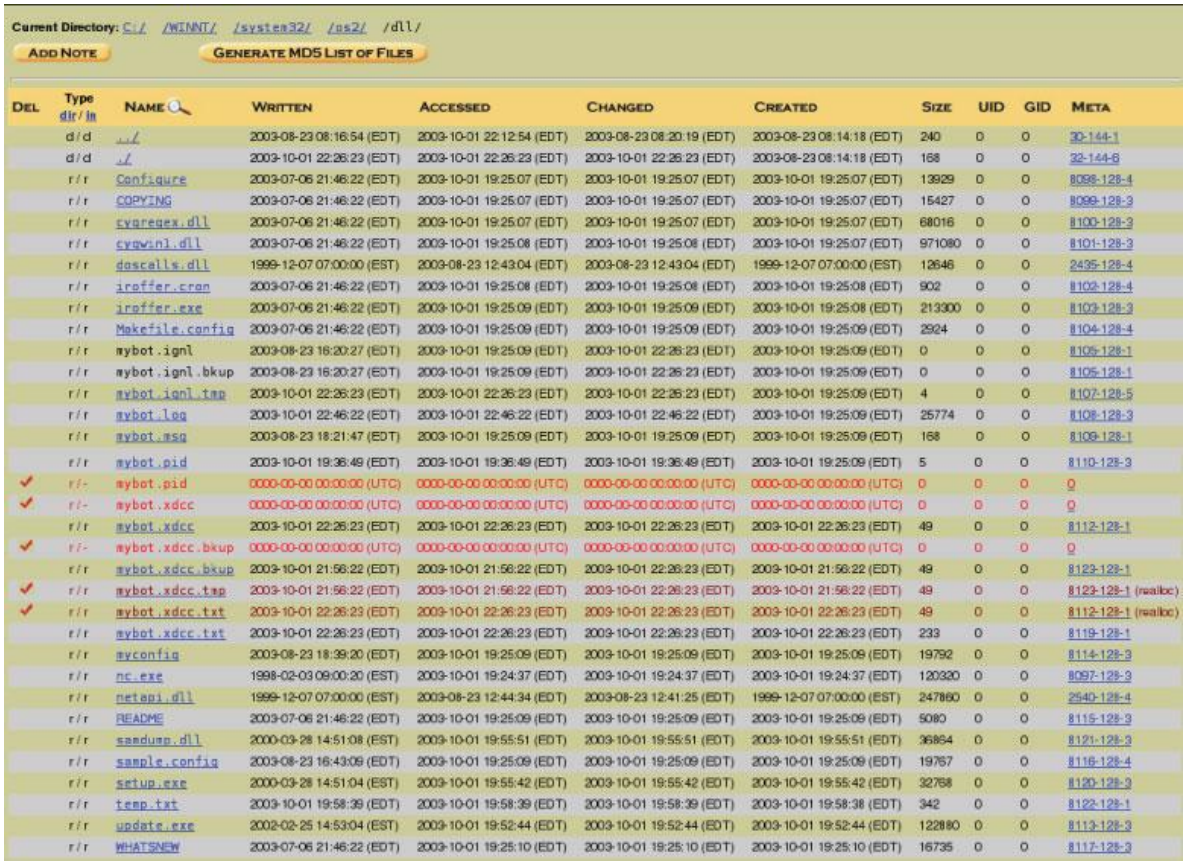
4.6 Análisis de archivos sospechosos

Para completar nuestra investigación debemos indagar en el sistema de archivos de la víctima y tratar de recuperar cualquier archivo sospechoso, pero nuestra experiencia nos ha enseñado que el atacante siempre borra los archivos más relevantes a la investigación.

Para llevar a cabo este análisis primero debemos reducir nuestro conjunto de datos, así podremos analizar los datos eficientemente. Además realizaremos una búsqueda de caracteres para poder identificar archivos relevantes o fragmentos de estos.

Anteriormente habíamos determinado que el directorio C:\winnt\system32\os2\dll contenía herramientas que el atacante había depositado, tales como iroffer.exe. Empezaremos analizando este directorio con la herramienta Authopsy del Sleuth Kit y ver si encontramos algún archivo de interés.

En esta figura encontramos varias rarezas que han sido marcadas de color rojo por el programa Authopsy. Primero tenemos unos archivos con el inodo de número cero, los cuales somos incapaces de poder recuperar. Los otros archivos tienen bloques de datos que han sido reasignados, el archivo mybot.xdcc.tmp y el archivo mybot.xdcc.txt poseen el mismo inodo que el archivo mybot.xdcc.bkup, el cual es 8123-128-1.



Current Directory: C:/ /WINNT/ /system32/ /os2/ /dll/

ADD NOTE | GENERATE MD5 LIST OF FILES

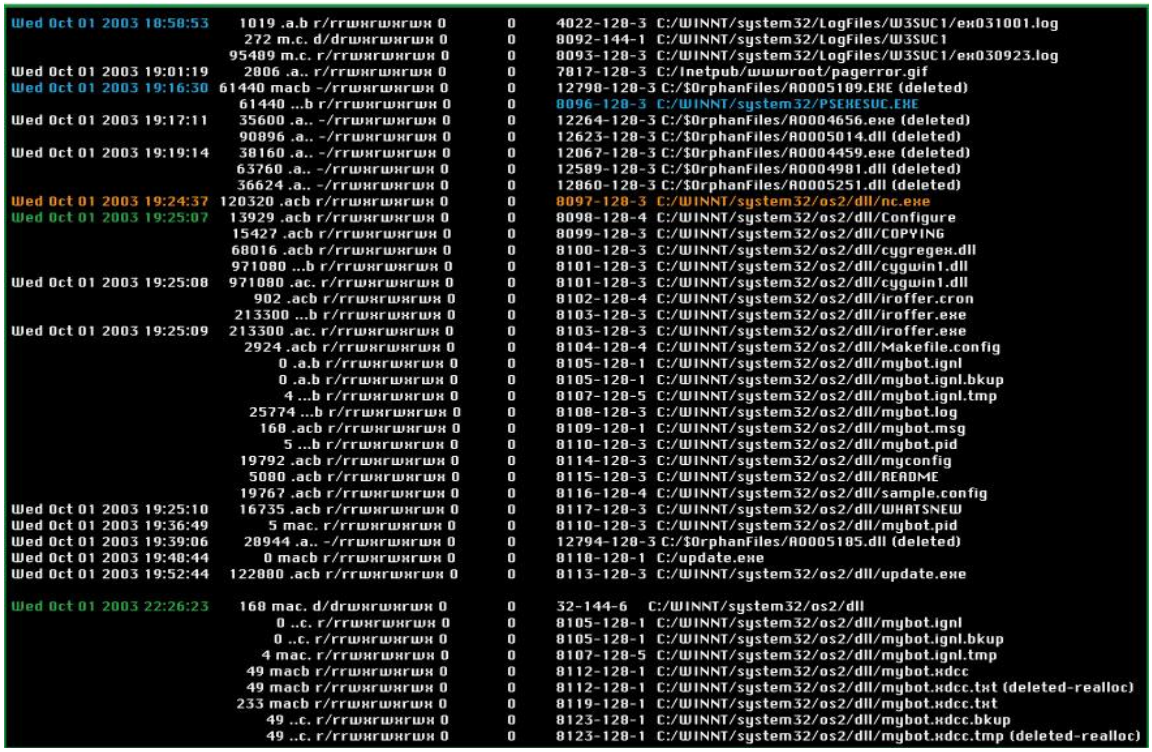
DEL	Type dir / in	NAME	WRITTEN	ACCESSED	CHANGED	CREATED	SIZE	UID	GID	META
	d / d	../	2003-08-23 08:16:54 (EDT)	2003-10-01 22:12:54 (EDT)	2003-08-23 08:20:19 (EDT)	2003-08-23 08:14:18 (EDT)	240	0	0	30-144-1
	d / d	./	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	2003-08-23 08:14:18 (EDT)	168	0	0	32-144-6
	r / r	Configure	2003-07-06 21:46:22 (EDT)	2003-10-01 19:25:07 (EDT)	2003-10-01 19:25:07 (EDT)	2003-10-01 19:25:07 (EDT)	13929	0	0	8098-128-4
	r / r	COPYING	2003-07-06 21:46:22 (EDT)	2003-10-01 19:25:07 (EDT)	2003-10-01 19:25:07 (EDT)	2003-10-01 19:25:07 (EDT)	15427	0	0	8099-128-3
	r / r	cygregex.dll	2003-07-06 21:46:22 (EDT)	2003-10-01 19:25:07 (EDT)	2003-10-01 19:25:07 (EDT)	2003-10-01 19:25:07 (EDT)	68016	0	0	8100-128-3
	r / r	cygwin1.dll	2003-07-06 21:46:22 (EDT)	2003-10-01 19:25:08 (EDT)	2003-10-01 19:25:08 (EDT)	2003-10-01 19:25:07 (EDT)	971080	0	0	8101-128-3
	r / r	doscalls.dll	1999-12-07 07:00:00 (EST)	2003-08-23 12:43:04 (EDT)	2003-08-23 12:43:04 (EDT)	1999-12-07 07:00:00 (EST)	12646	0	0	2435-128-4
	r / r	iroffer.cron	2003-07-06 21:46:22 (EDT)	2003-10-01 19:25:08 (EDT)	2003-10-01 19:25:08 (EDT)	2003-10-01 19:25:08 (EDT)	902	0	0	8102-128-4
	r / r	iroffer.exe	2003-07-06 21:46:22 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:08 (EDT)	213300	0	0	8103-128-3
	r / r	Makefile.config	2003-07-06 21:46:22 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:09 (EDT)	2924	0	0	8104-128-4
	r / r	mybot.ignl	2003-08-23 16:20:27 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 19:25:09 (EDT)	0	0	0	8105-128-1
	r / r	mybot.ignl.bkup	2003-08-23 16:20:27 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 19:25:09 (EDT)	0	0	0	8105-128-1
	r / r	mybot.ignl.tmp	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 19:25:09 (EDT)	4	0	0	8107-128-5
	r / r	mybot.log	2003-10-01 22:46:22 (EDT)	2003-10-01 22:46:22 (EDT)	2003-10-01 22:46:22 (EDT)	2003-10-01 19:25:09 (EDT)	25774	0	0	8108-128-3
	r / r	mybot.msg	2003-08-23 18:21:47 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:09 (EDT)	168	0	0	8109-128-1
	r / r	mybot.pid	2003-10-01 19:36:49 (EDT)	2003-10-01 19:36:49 (EDT)	2003-10-01 19:36:49 (EDT)	2003-10-01 19:25:09 (EDT)	5	0	0	8110-128-3
✔	r / -	mybot.pid	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0	0	0	0
✔	r / -	mybot.xdcc	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0	0	0	0
	r / r	mybot.xdcc	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	49	0	0	8112-128-1
✔	r / -	mybot.xdcc.bkup	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0	0	0	0
	r / r	mybot.xdcc.bkup	2003-10-01 21:56:22 (EDT)	2003-10-01 21:56:22 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 21:56:22 (EDT)	49	0	0	8123-128-1
✔	r / r	mybot.xdcc.tmp	2003-10-01 21:56:22 (EDT)	2003-10-01 21:56:22 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 21:56:22 (EDT)	49	0	0	8123-128-1 (realloc)
✔	r / r	mybot.xdcc.txt	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	49	0	0	8112-128-1 (realloc)
	r / r	mybot.xdcc.txt	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	2003-10-01 22:26:23 (EDT)	233	0	0	8119-128-1
	r / r	myconfig	2003-08-23 18:39:20 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:09 (EDT)	19792	0	0	8114-128-3
	r / r	nc.exe	1998-02-03 09:00:20 (EST)	2003-10-01 19:24:37 (EDT)	2003-10-01 19:24:37 (EDT)	2003-10-01 19:24:37 (EDT)	120320	0	0	8097-128-3
	r / r	netapi.dll	1999-12-07 07:00:00 (EST)	2003-08-23 12:44:34 (EDT)	2003-08-23 12:41:25 (EDT)	1999-12-07 07:00:00 (EST)	247860	0	0	2540-128-4
	r / r	README	2003-07-06 21:46:22 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:09 (EDT)	5080	0	0	8115-128-3
	r / r	samdump.dll	2000-03-28 14:51:08 (EST)	2003-10-01 19:55:51 (EDT)	2003-10-01 19:55:51 (EDT)	2003-10-01 19:55:51 (EDT)	36864	0	0	8121-128-3
	r / r	sample.config	2003-08-23 16:43:09 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:09 (EDT)	2003-10-01 19:25:09 (EDT)	19757	0	0	8116-128-4
	r / r	setup.exe	2000-03-28 14:51:04 (EST)	2003-10-01 19:55:42 (EDT)	2003-10-01 19:55:42 (EDT)	2003-10-01 19:55:42 (EDT)	32768	0	0	8120-128-3
	r / r	temp.txt	2003-10-01 19:58:39 (EDT)	2003-10-01 19:58:39 (EDT)	2003-10-01 19:58:39 (EDT)	2003-10-01 19:58:38 (EDT)	342	0	0	8122-128-1
	r / r	update.exe	2002-02-25 14:53:04 (EST)	2003-10-01 19:52:44 (EDT)	2003-10-01 19:52:44 (EDT)	2003-10-01 19:52:44 (EDT)	122880	0	0	8113-128-3
	r / r	WHATSNEW	2003-07-06 21:46:22 (EDT)	2003-10-01 19:25:10 (EDT)	2003-10-01 19:25:10 (EDT)	2003-10-01 19:25:10 (EDT)	16735	0	0	8117-128-3

Figura 4-53: Archivos borrados en el directorio C:\winnt\system32\os2\dll

Solo con esto no podemos obtener mucha información relevante ya que la mayoría de atacantes sobrescribe sobre sus huellas, por ende procederemos a adquirir los metadatos de todos los archivos que existen en la evidencia ya sean borrados o lógicos; con los metadatos podremos realizar una línea de tiempo de la actividad en los archivos y así podremos reducir nuestro rango a investigar. Para esto utilizaremos las facilidades que nos brinda Authopsy y procederemos a examinar los archivos que fueron creados aproximadamente a las 7:25 PM del 01/10/2003, que fue el momento en que se creó la mayoría de archivos del directorio C:\winnt\system32\os2\dll.

En esta imagen observaremos que primero fue creado psexecsvc.exe en el sistema aproximadamente a las 6:58PM. Este es el servicio que es creado cuando un usuario remoto ejecuta PsExec, una herramienta a control remoto. Asumimos que alguien usó PsExec, el cual requiere ser validado como Administrador en la máquina víctima. Luego, vemos que nc.exe fue creado en la máquina aproximadamente a las7:24PM, este archivo pertenece a la herramienta netcat. A las 7:25PM Iroffer fue transferido a la

máquina. A las 7:48 y 7:52, update.exe fue transferido a JBRWWW. Luego después de las 10:00PM varios componentes de Iroffer fueron creados.



```
Wed Oct 01 2003 18:58:53    1019 .a.b r/rrwxrwxrwx 0       0    4022-128-3 C:/WINNT/system32/LogFiles/W3SVC1/ex031001.log
                             272 m.c. d/drwxrwxrwx 0       0    8092-144-1 C:/WINNT/system32/LogFiles/W3SVC1
                           95489 m.c. r/rrwxrwxrwx 0       0    8093-128-3 C:/WINNT/system32/LogFiles/W3SVC1/ex030923.log
Wed Oct 01 2003 19:01:19    2006 .a.. r/rrwxrwxrwx 0       0    7817-128-3 C:/Inetpub/wwwroot/pagerror.gif
Wed Oct 01 2003 19:16:30   61440 macb -/rrwxrwxrwx 0       0    12790-128-3 C:/$OrphanFiles/A0005189.EXE (deleted)
                           61440 ...b r/rrwxrwxrwx 0       0    8096-128-3 C:/WINNT/system32/PSEXESVC.EXE
Wed Oct 01 2003 19:17:11   35600 .a.. -/rrwxrwxrwx 0       0    12264-128-3 C:/$OrphanFiles/A0004656.exe (deleted)
                           90896 .a.. -/rrwxrwxrwx 0       0    12623-128-3 C:/$OrphanFiles/A0005014.dll (deleted)
Wed Oct 01 2003 19:19:14   38160 .a.. -/rrwxrwxrwx 0       0    12067-128-3 C:/$OrphanFiles/A0004459.exe (deleted)
                           63760 .a.. -/rrwxrwxrwx 0       0    12589-128-3 C:/$OrphanFiles/A0004981.dll (deleted)
                           36624 .a.. -/rrwxrwxrwx 0       0    12860-128-3 C:/$OrphanFiles/A0005251.dll (deleted)
Wed Oct 01 2003 19:24:37  120320 .acb r/rrwxrwxrwx 0       0    8097-128-3 C:/WINNT/system32/os2/dll/nc.exe
Wed Oct 01 2003 19:25:07   13929 .acb r/rrwxrwxrwx 0       0    8098-128-4 C:/WINNT/system32/os2/dll/Configure
                           15427 .acb r/rrwxrwxrwx 0       0    8099-128-3 C:/WINNT/system32/os2/dll/COPYING
                           68016 .acb r/rrwxrwxrwx 0       0    8100-128-3 C:/WINNT/system32/os2/dll/cygregex.dll
                          971080 ...b r/rrwxrwxrwx 0       0    8101-128-3 C:/WINNT/system32/os2/dll/cygwin1.dll
Wed Oct 01 2003 19:25:08  971080 .ac. r/rrwxrwxrwx 0       0    8101-128-3 C:/WINNT/system32/os2/dll/cygwin1.dll
                             902 .acb r/rrwxrwxrwx 0       0    8102-128-4 C:/WINNT/system32/os2/dll/iroffer.cron
                          213300 ...b r/rrwxrwxrwx 0       0    8103-128-3 C:/WINNT/system32/os2/dll/iroffer.exe
Wed Oct 01 2003 19:25:09  213300 .ac. r/rrwxrwxrwx 0       0    8103-128-3 C:/WINNT/system32/os2/dll/iroffer.exe
                            2924 .acb r/rrwxrwxrwx 0       0    8104-128-4 C:/WINNT/system32/os2/dll/Makefile.config
                               0 .a.b r/rrwxrwxrwx 0       0    8105-128-1 C:/WINNT/system32/os2/dll/mybot.ignl
                               0 .a.b r/rrwxrwxrwx 0       0    8105-128-1 C:/WINNT/system32/os2/dll/mybot.ignl.bkup
                               4 ...b r/rrwxrwxrwx 0       0    8107-128-5 C:/WINNT/system32/os2/dll/mybot.ignl.tmp
                           25774 ...b r/rrwxrwxrwx 0       0    8108-128-3 C:/WINNT/system32/os2/dll/mybot.log
                             168 .acb r/rrwxrwxrwx 0       0    8109-128-1 C:/WINNT/system32/os2/dll/mybot.msg
                               5 ...b r/rrwxrwxrwx 0       0    8110-128-3 C:/WINNT/system32/os2/dll/mybot.pid
                           19792 .acb r/rrwxrwxrwx 0       0    8114-128-3 C:/WINNT/system32/os2/dll/myconfig
                            5080 .acb r/rrwxrwxrwx 0       0    8115-128-3 C:/WINNT/system32/os2/dll/README
                           19767 .acb r/rrwxrwxrwx 0       0    8116-128-4 C:/WINNT/system32/os2/dll/sample.config
Wed Oct 01 2003 19:25:10   16735 .acb r/rrwxrwxrwx 0       0    8117-128-3 C:/WINNT/system32/os2/dll/WHATSNEW
Wed Oct 01 2003 19:36:49       5 mac. r/rrwxrwxrwx 0       0    8110-128-3 C:/WINNT/system32/os2/dll/mybot.pid
Wed Oct 01 2003 19:39:06   28944 .a.. -/rrwxrwxrwx 0       0    12794-128-3 C:/$OrphanFiles/A0005185.dll (deleted)
Wed Oct 01 2003 19:48:44       0 macb r/rrwxrwxrwx 0       0    8118-128-1 C:/update.exe
Wed Oct 01 2003 19:52:44  122880 .acb r/rrwxrwxrwx 0       0    8113-128-3 C:/WINNT/system32/os2/dll/update.exe

Wed Oct 01 2003 22:26:23     168 mac. d/drwxrwxrwx 0       0    32-144-6   C:/WINNT/system32/os2/dll
                               0 ..c. r/rrwxrwxrwx 0       0    8105-128-1 C:/WINNT/system32/os2/dll/mybot.ignl
                               0 ..c. r/rrwxrwxrwx 0       0    8105-128-1 C:/WINNT/system32/os2/dll/mybot.ignl.bkup
                               4 mac. r/rrwxrwxrwx 0       0    8107-128-5 C:/WINNT/system32/os2/dll/mybot.ignl.tmp
                              49 macb r/rrwxrwxrwx 0       0    8112-128-1 C:/WINNT/system32/os2/dll/mybot.xdcc
                              49 macb r/rrwxrwxrwx 0       0    8112-128-1 C:/WINNT/system32/os2/dll/mybot.xdcc.txt (deleted-realloc)
                             233 macb r/rrwxrwxrwx 0       0    8119-128-1 C:/WINNT/system32/os2/dll/mybot.xdcc.txt
                              49 ..c. r/rrwxrwxrwx 0       0    8123-128-1 C:/WINNT/system32/os2/dll/mybot.xdcc.bkup
                              49 ..c. r/rrwxrwxrwx 0       0    8123-128-1 C:/WINNT/system32/os2/dll/mybot.xdcc.tmp (deleted-realloc)
```

Figura 4-54: Archivos creados durante el 1 de Octubre del 2003

Como podemos ver alguien usó PsExec para ganar acceso a JBRWWW, seguramente adivinando las contraseñas, después varias herramientas fueron transferidas a la máquina por el atacante y por último Iroffer fue ejecutado tras una hora de su transferencia.

Por último procederemos a realizar una búsqueda de caracteres con la opción que nos brinda Autopsy; primero usamos la opción extraer caracteres y luego podremos realizar la búsqueda de palabras claves, como sabemos nuestro atacante dejó una herramienta llamada IROffer en el servidor, veremos si podemos averiguar algo con esto:

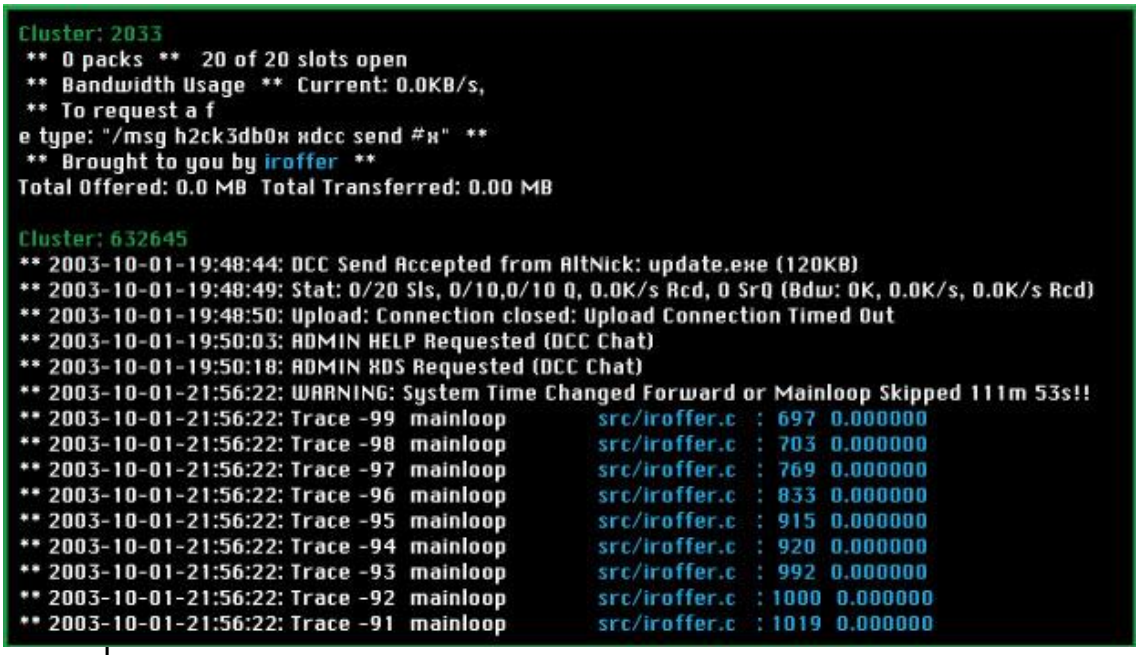


Figura 4-55: Búsqueda de caracteres (iroffer)

a

siguiente búsqueda relevante que haremos es la del archivo

encontrado en el directorio C:\update.exe veremos si podemos encontrar alguna relación con otros archivos sospechosos:

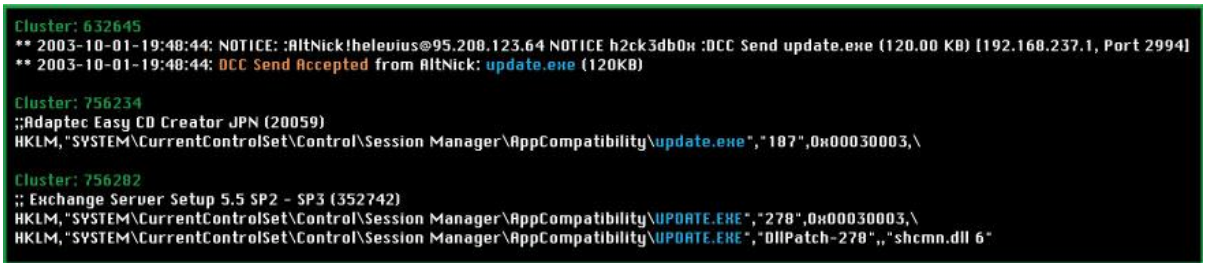


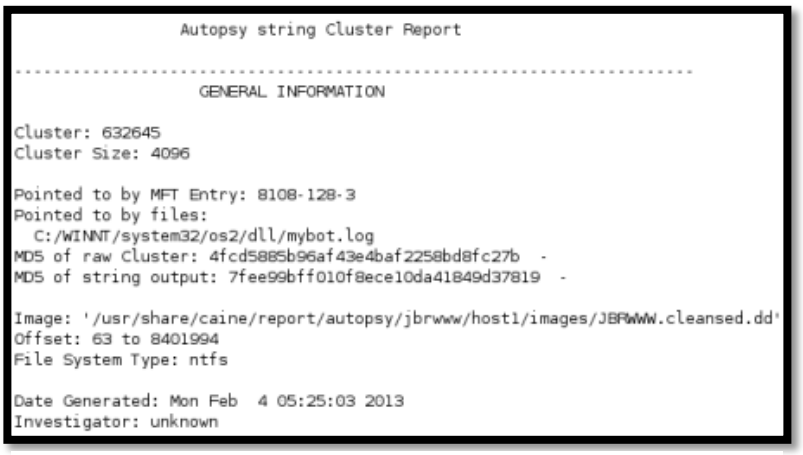
Figura 4-56: Búsqueda de caracteres (update.exe)

Hemos presentado las salidas más relevantes en las figuras 59-60 y en esta última podemos apreciar que update.exe pudo haber sido localizado en nuestro sistema por el protocolo [34]DCC. El protocolo DCC por lo general es asociado con la herramienta IRC, el cual es una herramienta muy común que los atacantes usan para controlar computadoras remotamente.

Además podemos continuar este rastro generando el informe del clúster 632645 y nos mostrará una relación con un archivo lógico llamado c:/winnt/system32/os2/dll/mybot.log, el cual parece ser un

[34] Anexo VII – Protocolo DCC

archivo de registros para las herramientas del atacante. Sin más información podemos suponer que este registro pertenece al robot IRC llamado IROffer.



```
                Autopsy string Cluster Report

----------------------------------------------------------------------
                   GENERAL INFORMATION

Cluster: 632645
Cluster Size: 4096

Pointed to by MFT Entry: 8108-128-3
Pointed to by files:
  C:/WINNT/system32/os2/dll/mybot.log
MD5 of raw Cluster: 4fcd5885b96af43e4baf2258bd8fc27b  -
MD5 of string output: 7fee99bff010f8ece10da41849d37819  -

Image: '/usr/share/caine/report/autopsy/jbrwww/host1/images/JBRWWW.cleansed.dd'
Offset: 63 to 8401994
File System Type: ntfs

Date Generated: Mon Feb  4 05:25:03 2013
Investigator: unknown
```

Figura 4-57: Informe generado del clúster 632645

Tras examinar el archivo mybot.log encontramos una cabecera el nombre de iroffer y más adelante en este mismo archivo veremos el intento de la transmisión del archivo update.exe.

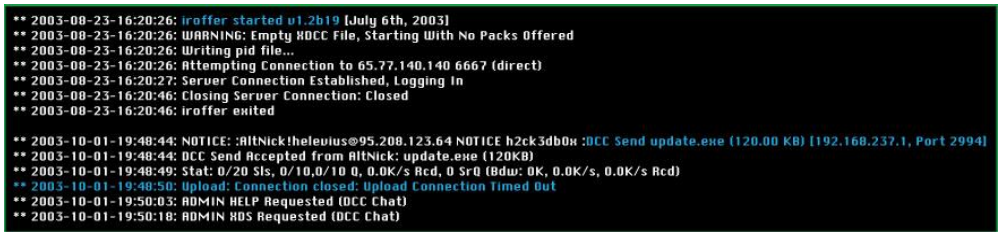


Figura 4-58: Partes relevantes del archivo mybot.log

Este archivo nos muestra que el archivo update.exe no pudo ser transmitido con éxito hacia la máquina víctima. El atacante intentó subir el archivo, pero falló dejándonos un archivo con longitud cero, como vemos en el directorio raíz de la máquina victima JBRWWW.

CONCLUSIONES

1. Tras haber analizado varias evidencias hemos confirmado la presencia de un atacante gracias al registro de las conexiones que nos brinda la herramienta netstat, pudimos localizar conexiones desde la IP 95.28.123.64 el cual tenía varios puertos abiertos para herramientas como Psexec, IROffer, IRC
2. La revisión de los registros IIS nos demuestran que hubo un ataque anteriormente, el día 9 de Septiembre del 2003 hubo un ataque de reconocimiento desde la dirección 95.16.3.79 con la herramienta online Nikto.
3. El día 1 de Octubre del 2003, un ataque Unicode y Double Decode fueron ejecutados a nuestra victima desde las direcciones IP 95.208.123.64 y 95.16.3.79 logrando darle privilegios al atacante de ejecutar la línea de comandos cmd.exe bajo el usuario IUSR_JBRWWW.

Esto le permitió establecer una sesión FTP, además pudo instalar netcat e iroffer en el directorio C:\WINNT\system32\os2\dll.

4. Tras analizar los archivos de la actividad de red capturados con snort pudimos confirmar que hubo un escaneo de vulnerabilidades con la herramienta Nmap. La revisión de los registros de alertas de Argus nos muestra varios ataques de reconocimiento al server desde la IP 95.16.3.79, pero además una conexión de nuestro server a la IP 95.145.128.17, el cual asumimos que pertenece a un servidor IRC.
5. En las sesiones capturadas por tcpflow encontramos un sesión IRC con el Nick h2ck3db0x donde al atacante intenta transferir archivos a través del protocolo DCC, pero sin éxito por algún error de configuración entre el cliente y servidor.
6. Al analizar la línea de tiempo de actividad en los archivos creada por Autopsy pudimos asegurar que luego del ataque de desbordamiento de búfer el atacante logró ejecutar psexec con un usuario Administrador, el cual pudo ser obtenido por ataques de fuerza bruta. Esto le permitió instalar netcat e iroffer y con este pudo conectarse al servidor IRC y tener visión de los archivos en JBRWWW.
7. Gracias a la búsqueda de caracteres con palabras claves sospechosas pudimos confirmar el intento de transmisión de update.exe a través del

protocolo DCC, el cual no tuvo éxito dejándonos un archivo de longitud cero.

8. Por último podemos concluir que un ataque bien planeado tuvo éxito, éste logró tener acceso solo al Web Server en la dirección IP 103.98.91.41, pudo haber tenido más control en el sistema, pero fue detectado a tiempo sin causar grandes pérdidas.
9. Al ser una institución bancaria se pudo haber comprometido mucha información valiosa, pero gracias a que el ataque fue detenido a tiempo el atacante no pudo conseguir nada, por lo que la institución bancaria no debería presentar un informe a sus clientes finales.

RECOMENDACIONES

1. Una de las primeras cosas que pudimos notar en nuestra investigación fue la falta de politicias de auditoria, esto nos hizo perder mucha información valiosa para nuestra investigación que nos pudo haber facilitado el trabajo, por lo que se recomienda configurar estas políticas para poder mantener un registro de inicios de sesión.
2. Se recomienda también mantener actualizado todo el sistema para evitar que los intrusos aprovechen exploits conocidos y tomar control del sistema como lo hizo en este caso.
3. Aunque en este caso el ataque pudo ser neutralizado por un usuario no está de más decir que mantener informados a los usuarios sobre cómo aplicar las políticas de seguridad y mantener sus contraseñas privadas como es debido evitará que ocurran incidentes parecidos en el futuro.
4. Los administradores del sistema deben mantenerse siempre informados sobre las nuevas amenazas que aparecen día a día para poder defenderse de la mejor manera posible contra estos atacantes.

5. Al instalar actualizaciones o programas de cualquier tipo asegurarse que solo sean de una fuente confiable, así el sistema se mantendrá limpio y en un funcionamiento óptimo. Asegurarse de descargar y ejecutar archivos de los cuales sabemos su procedencia.
6. Al momento de instalar scripts en el sitio Web asegurarse de que no contengan errores, esto lo logramos relizando un lookfor de los scripts en sitios como milw0rm.com para ver estos no tienen errores.
7. Hacer una copia de seguridad del sitio. Mantener los archivos en otro PC, USB o Disco Duro Externo.

ANEXOS

ANEXO I - Sleuth Kit & Autopsy

sleuthkit.org es la web oficial para The Sleuth Kit (TSK) y Autopsy Browser, ambas herramientas open source para la investigación forense y ejecutables en entornos Windows y Unix (tales como Linux, OS X, Cygwin, FreeBSD, OpenBSD, y Solaris).

Está basado en The Coroner's Toolkit y en los añadidos a éste que hacia tct-utils, ampliando las funcionalidades de ambos[35].

The Sleuth Kit (TSK)

Es una librería y colección de herramientas de línea de comando que nos permite investigar imagines de disco. La funcionalidad principal del TSK nos permite analizar volumenes y sistemas de archivos.

[35] (Carrier)

The Sleuth Kit Hadoop Framework

The Sleuth Kit Hadoop Framework es un framework que incorpora TSK en cloud computing para el análisis de grandes volúmenes de datos.

Autopsy

El navegador forense Autopsy es una interfaz gráfica para las herramientas de línea de comandos digitales de investigación de análisis en Sleuth.Kit. Juntos, pueden analizar los discos de Windows y UNIX y sistemas de archivos (NTFS, FAT, UFS1 / 2, Ext2 / 3).

El Sleuth Kit y Autopsy son de código abierto y se ejecutan en plataformas UNIX (se puede usar Cygwin para ejecutar también en Windows). Autopsy se basa en HTML, por lo que se puede conectar con el servidor de Autopsy desde cualquier plataforma con un navegador HTML. Autopsy proporciona un "Administrador de archivos"-a manera de interfaz y muestra detalles acerca de los datos eliminados y estructuras del sistema de archivos.

Modos de Análisis

Un **análisis muerto** se produce cuando un sistema de análisis dedicado se utiliza para examinar los datos de un sistema

sospechoso. En este caso, Autopsy y Sleuth Kit se ejecutan en un entorno de confianza, por lo general en un laboratorio. Autopsy y TSK soportan imágenes crudas, y los formatos de archivos de AFF.

Un **análisis en vivo** se produce cuando el sistema sospechoso está siendo analizado, mientras que se está ejecutando. En este caso, autopsy y TSK se ejecutan desde un CD en un ambiente de confianza. Esto se utiliza con frecuencia durante la respuesta a incidentes, mientras que el incidente está siendo confirmado. Después de que se confirma, el sistema puede ser adquirido y se realizó un análisis muerto.

Técnicas de búsqueda de evidencia

Listado de archivos: Analiza los archivos y directorios, incluyendo los nombres de los archivos eliminados y archivos con nombres basados en Unicode.

El contenido del archivo: El contenido de los archivos se pueden ver en hexadecimal prima, o las cadenas de caracteres ASCII puede ser extraído. Cuando la información es interpretada, autopsy se desinfecta para evitar daños en el sistema de análisis

local. Autopsy no utiliza ningún tipo de lenguajes de script del lado del cliente.

Bases de datos hash: Búsqueda archivos desconocidos en una base de datos de hash para identificar rápidamente como bueno o malo. Autopsy utiliza el NIST National Software Reference Library (NSRL) y el usuario crea bases de datos de archivos buenos y malos.

Clasificación Tipo de Archivo: Ordena los archivos basados en sus firmas internas para identificar los archivos de un tipo conocido. Autopsy también se puede extraer sólo las imágenes gráficas (incluidas las miniaturas). La extensión del archivo también se puede comparar con el tipo de archivo para identificar los archivos que pueden haber tenido su extensión cambiada para ocultarlos.

Línea de tiempo de actividad de los archivos: En algunos casos, tener una línea de tiempo de actividad de los archivos puede ayudar a identificar las áreas de un sistema de archivos que pueden contener pruebas. Autopsy puede crear líneas de tiempo

que contienen entradas para la modificación, según las veces de Acceso y modificación de archivos asignados y sin asignar.

Buscar palabra clave: las búsquedas de palabras clave de la imagen del sistema de archivos puede realizarse utilizando cadenas de caracteres ASCII y expresiones regulares grep. Las búsquedas se pueden realizar ya sea en el sistema de archivos de imagen completa o sólo el espacio no asignado. Un archivo de índice puede ser creado para búsquedas más rápidas. Las cadenas que con frecuencia buscado se puede configurar fácilmente en autopsy para una automatizada consulta.

Análisis de Meta-Data: Las estructuras Meta-Data contienen los detalles sobre los archivos y directorios. Autopsy le permite ver los detalles de cualquier estructura de metadatos del sistema de archivos. Esto es útil para recuperar el contenido eliminado. Autopsy buscará en los directorios para identificar la ruta completa del archivo que se ha asignado la estructura.

Análisis de unidad de datos: Unidad de datos es donde el contenido del archivo se almacena. Autopsy le permite ver el contenido de cualquier unidad de datos en una variedad de

formatos, incluyendo ASCII, hexdump y cuerdas. El tipo de archivo también se da y autopsia buscará las estructuras de metadatos para identificar qué ha asignado la unidad de datos.

Datos de la imagen: Datos del fichero del sistema pueden ser vistos, incluyendo la disposición en disco y tiempo de actividad. Este modo proporciona información que es útil durante la recuperación de datos.

Manejo de un Caso

Manejo de Casos: Las investigaciones están organizados por casos, que puede contener uno o más huéspedes. Cada host está configurado para tener su propia configuración de zona de tiempo y desplazamiento de reloj de modo que los tiempos mostrados son los mismos que el usuario original habría visto. Cada host puede contener una o varias imágenes del sistema de archivos a analizar.

Secuenciador de evento: Basado en el tiempo se pueden añadir eventos de actividad de los archivos y registros de IDS o de firewall. Autopsy ordena los eventos de modo que la secuencia de eventos de incidentes se pueda determinar más fácilmente.

Notas: Las notas se pueden guardar en una base por huésped y por el investigador. Estas te permiten hacer notas rápidas sobre archivos y estructuras. La ubicación original se puede recuperar fácilmente con sólo pulsar un botón cuando las notas son posteriormente revisadas. Todas las notas se almacenan en un archivo ASCII.

Integridad de la imagen: es crucial para asegurar que los archivos no se modifican durante el análisis. Autopsy, por defecto, generará un valor MD5 para todos los archivos que se importan o creado. La integridad de cualquier archivo que utiliza autopsy se puede validar en cualquier momento.

Informes: Autopsy puede crear informes ASCII para los archivos y otras estructuras del sistema de archivos. Esto le permite realizar rápidamente las hojas de datos consistentes durante la investigación.

Registro: Los registros de auditoría se crean en un caso, el huésped y el nivel investigador de modo que las acciones puedan ser fácilmente registradas. Los comandos TSK que se ejecutan a menudo también son registrados.

Open Design: El código de Autopsy es de código abierto y todos los archivos que utiliza están en un formato crudo. Todos los archivos de configuración se encuentran en texto ASCII y los casos son organizados por directorios. Esto hace que sea fácil de exportar los datos y archivarlo. Asimismo, no se restringe el uso de otras herramientas que pueden resolver el problema específico más apropiado.

Modelo cliente servidor: Autopsy se basa en HTML y por lo tanto usted no tiene que estar en el mismo sistema que las imágenes del sistema de archivos. Esto permite que múltiples investigadores utilicen el mismo servidor y se conecten desde sus sistemas personales.

ANEXO II – Herramienta dd

El comando dd (duplicate disk) es un comando bastante útil para transferir datos desde un dispositivo/archivo hacia un dispositivo/archivo/etc[36].

A continuación vamos a ver las diferentes utilidades que se le pueden dar al comando

[36] (El comando DD, 2005)

La sintaxis básica del comando es la siguiente:

```
dd if=origen of=destino
```

donde if significa "input file", es decir, lo que quieres copiar y of significa "output file", o sea, el archivo destino (donde se van a copiar los datos); origen y destino pueden ser dispositivos (lectora de CD, diskettera, etc.), archivos, etc.

Copiando Diskettes:

Primero insertamos el diskette origen y escribimos lo siguiente en una consola:

```
dd if=/dev/fd0 of=~/diskette.img
```

Después insertamos el diskette destino (en blanco) y escribimos lo siguiente:

```
dd if=~/diskette.img of=/dev/fd0
```

Nos queda eliminar la "imagen" que creamos y listo:

```
rm -f ~/diskette.img
```

El ~ significa "tu directorio home", es similar a escribir $HOME

Manejo de errores durante la copia:

Es posible que durante la copia o duplicación de un diskette se encuentren errores en la superficie del mismo. Para evitar que este error nos impida copiar los datos "buenos" del disco podemos hacer lo siguiente:

```
dd conv=noerror if=/dev/fd0
of=~/imagen_disco_con_errores.img
```

La opción noerror hace que se continúe con la copia aunque se produzcan errores de lectura.

Haciendo imágenes ISO de un CD:

La forma más fácil y efectiva de crear nuestras "imagenes" de CD es la siguiente:

```
dd if=/dev/cdrom of=micd.iso
```

El comando dd también sirve para copiar particiones o discos completos unos sobre otros. Básicamente podemos decir que mediante dd podemos "clonar" particiones o nuestro disco rígido completo. Para hacer esto hacemos lo siguiente:

dd if=/hdxa of=/hdyb (copia una partición en otra)

dd if=/hdx of=/hdy (copia de un disco duro en otro)

Donde x: disco rígido origen, y: disco rígido destino, a: partición origen, b: partición destino.

Es necesario que sepas como se definen los discos y particiones en Linux antes de mandarte con estos comandos

Debemos tener presente que el tamaño de la imagen resultante va a ser exactamente el mismo que el del dispositivo original. Es decir: dd te guarda también el espacio no utilizado.

Podemos redirigir la salida con una tubería (pipe) y comprimirlo con gzip, bzip o bzip2, pero aún así vamos a necesitar bastante espacio libre para poder guardar las imágenes que generes.

ANEXO III – Comando Netstat

netstat es una herramienta útil para comprobar la configuración y actividad de su red. Se llama netstat, aunque se trata en realidad de una colección de

herramientas combinadas. Describiremos cada una de las funciones en las secciones siguientes[37].

Consulta de la tabla de enrutamiento

Si ejecuta netstat usando el indicador –r, puede ver la información de la tabla de encaminamiento del núcleo igual que hemos venido haciendo hasta ahora con route. Para vstout, tendríamos:

Tabla 4: Tabla de enrutamiento netstat-nr

```
# netstat -nr
Kernel IP routing table
Destination   Gateway     Genmask         Flags  MSS Window  irtt Iface
127.0.0.1     *           255.255.255.255 UH       0 0          0 lo
172.16.1.0    *            255.255.255.0   U       0 0          0 eth0
172.16.2.0    172.16.1.1   255.255.255.0   UG      0 0          0 eth0
```

La opción -n hace que netstat imprima las direcciones IP en notación de cuaterna en vez de usar los nombres simbólicos de las máquinas o las redes. Esto es especialmente útil si pretende evitar consultas para esos nombres a través de la red (por ejemplo consultas a un servidor DNS o NIS).

La segunda columna de la salida producida por netstat informa sobre las pasarelas a las que apunta la información de encaminamiento. Si una ruta no usa pasarela, el programa imprime un asterisco. La tercera

[37] (Dawson & Kirch, 2002)

columna imprime el nivel de “generalización” de una ruta. Dada una dirección IP para la que encontrar una ruta apropiada, el núcleo recorre la tabla registro a registro haciendo un "Y" lógico de la dirección y la máscara de nivel de generalización antes de compararla con el destino que muestra dicho registro.

La cuarta columna muestra varios indicadores que describen la ruta:

G La ruta utiliza una pasarela.

U La interface esta activa.

H Esta interface permite el acceso a una sola máquina. Este es el caso de la interface de lazo 127.0.0.1.

D Esta ruta es creada dinámicamente. Aparece si la entrada de la tabla ha sido generada por un demonio de encaminamiento como gated o por un mensaje de redirección ICMP

M Presente cuando este registro ha sido modificado por un mensaje de redirección ICMP.

! La ruta es una ruta de rechazo, y los datagramas serán descartados.

Las siguientes tres columnas muestran el MSS, tamaño de ventana y irtt que serán aplicados a las conexiones TCP establecidas a través de esta

ruta. El MSS es el Tamaño Máximo de Segmento, y es el tamaño del datagrama más grande que construirá el núcleo para transmitir a través de esta ruta. La Ventana es la cantidad máxima de datos que el sistema aceptará de una sola vez desde una máquina remota. El acrónimo irtt significa “tiempo inicial de ida y vuelta”, por sus iniciales en inglés. El protocolo TCP se asegura de que los datos han sido transmitidos de forma fiable entre máquinas retransmitiendo un datagrama si éste ha sido perdido. El protocolo TCP mantiene un contador de cuánto tarda un datagrama en ser enviado a su destino, y el "recibo" que se recibe, de forma que sabe cuánto esperar antes de suponer que un datagrama necesita retrasmitirse. Este proceso se llama tiempo de ida y vuelta. El tiempo de ida y vuelta inicial es el valor que el protocolo TCP usará cuando se establezca una conexión por primera vez. Para la mayoría de los tipos de redes, el valor por defecto es válido, pero para algunas redes lentas, especialmente ciertos tipos de redes de radiopaquetes de aficionados, el tiempo es demasiado pequeño y causa retransmisiones innecesarias. El valor de irtt puede ajustarse usando el comando route. Los campos a 0 significan que se está usando el valor por defecto.

Consulta de las estadísticas de una interfaz

Cuando se invoca con el indicador –i netstat presenta las estadísticas para las interfaces de red configuradas en ese momento. Si también se pasa la opción –a, mostrará todas las interfaces presentes en el núcleo, y no sólo aquellas que hayan sido configuradas. En vstout, la salida para netstat sería algo así:

Tabla 5: Estadísticas de una Interfaz nestat -i

```
# netstat -i
Kernel Interface table
Iface MTU Met  RX-OK RX-ERR RX-DRP RX-OVR  TX-OK TX-ERR TX-DRP TX-OVR Flags
lo      0   0   3185      0      0      0   3185      0      0      0 BLRU
eth0 1500   0 972633     17     20    120 628711    217      0      0 BRU
```

Los campos MTU y Met muestran los valores actuales de MTU y de métrica para esa interfaz. Las columnas RX y TX muestran cuántos paquetes han sido recibidos o transmitidos sin errores (RX-OK/TX-OK) o dañados (RX-ERR/TX-ERR); cuántos fueron descartados (RX-DRP/TX-DRP); y cuántos se perdieron por un desbordamiento. (RX-OVR/TX-OVR).

La última columna muestra los indicadores activos para cada interface. Son abreviaturas del nombre completo del indicador, que se muestran con la configuración de la interfaz que ofrece ifconfig:

B Dirección de difusión activa.

L La interfaz es un dispositivo de lazo.

M Se reciben todos los paquetes (modo promiscuo).

O ARP no funciona para esta interfaz.

P Conexión punto a punto.

R La interfaz funciona.

U La interfaz está activa.

Mostrar conexiones

netstat ofrece una serie de opciones para mostrar los puertos activos o pasivos. Las opciones –t, –u, –w, y –x muestran conexiones activas a puertos TCP, UDP, RAW, o Unix. Si incluye además el indicador –a, se mostrarán también los puertos que estén esperando una conexión (es decir, que estén escuchando). Esto le dará una lista de todos los servidores que estén corriendo actualmente en su sistema.

Llamar a netstat -ta en vlager produce esta salida:

Tabla 6: Lista de Conexiones netstat -ta

$ netstat -ta **Active Internet Connections**

```
Proto Recv-Q Send-Q Local Address    Foreign Address    (State)
tcp        0      0 *:domain        *:*              LISTEN
tcp        0      0 *:time          *:*              LISTEN
tcp        0      0 *:smtp          *:*              LISTEN
tcp        0      0 vlager:smtp     vstout:1040      ESTABLISHED
tcp        0      0 *:telnet        *:*              LISTEN
tcp        0      0 localhost:1046  vbardolino:telnet  ESTABLISHED
tcp        0      0 *:chargen       *:*              LISTEN
tcp        0      0 *:daytime       *:*              LISTEN
tcp        0      0 *:discard       *:*              LISTEN
tcp        0      0 *:echo          *:*              LISTEN
tcp        0      0 *:shell         *:*              LISTEN
tcp        0      0 *:login         *:*              LISTEN
```

Esta salida muestra que la mayoría de los servidores están simplemente esperando una conexión externa. Sin embargo, la cuarta línea muestra una conexión SMTP desde vstout, y la sexta línea le indica que usted está haciendo una conexión telnet a vbardolino.

El indicador –a por sí sólo indicará todos los sockets de todo tipo.

ANEXO IV – PsTools Suite

A pesar que trabajemos en la línea de comandos podemos utilizar estas herramientas con nuestro VB6. Con un simple “Shell” es más que suficiente.

A modo de resumen:

Las herramientas incluidas en PsTools son[38]:

- PsExec – Ejecución remota de procesos.
- PsFile – Muestra remota de los ficheros abiertos.
- PsGetSid – Ver el SID del ordenador o de un usuario.
- PsInfo -Lista de información acerca del sistema remoto.
- PsKill – Matar procesos por nombre o ID del proceso.
- PsList – Lista información detallada acerca de los procesos.
- PsLoggedOn – Ver quien está conectado (logged) en local o a través de los recursos compartidos.
- PsLogList – Volcado de registros de eventos.
- PsPasswd – Cambio de contraseñas de cuentas.
- PsService – Vista y control de servicios.
- PsShutdown – Apagar o reiniciar una computadora.
- PsSuspend – Suspender o recuperar (resume) procesos.

PsExec

Utilidades como los programas de control de Telnet y remoto, como PC Anywhere de Symantec permiten ejecutar programas en sistemas remotos, pero puede ser un problema instalar y requieren la instalación

[38] (Russinovich, 2012)

de software de cliente en los sistemas remotos que desea acceder. PsExec es un reemplazo de Telnet ligero que le permite ejecutar procesos en otros sistemas, completo con interactividad completa para aplicaciones de consola, sin tener que instalar manualmente software de cliente. Los usos más potentes incluyen el lanzamiento de comando interactivo indicaciones que aparecen en sistemas remotos y herramientas que permitan a distancia como IpConfig que de otro modo no se podrían ejecutar sobre los sistemas remotos.

```
psexec [\\computer[,computer2[,...] | @file][-u user [-p psswd]][-n s][-l][-s|-e][-x][-i [session]][-c [-f|-v]][-w directory][-d][-<priority>][-a n,n,... ] cmd [arguments]
```

computer directo PsExec para ejecutar la aplicación en el equipo o equipos especificados. Si se omite el nombre del equipo PsExec ejecuta la aplicación en el sistema local y si escribe un nombre de equipo de "\ \ *" PsExec ejecuta las aplicaciones en todos los equipos del dominio actual. @ File Indica a PsExec para ejecutar el comando en cada equipo incluido en el archivo de texto especificado.

-a procesadores separados en los que la aplicación se pueden ejecutar con comas donde 1 es la CPU más bajo

numerado. Por ejemplo, para ejecutar la aplicación en la CPU 2 y la CPU 4, escriba: "-a 2,4"

-c Copia el programa especificado en el sistema remoto para su ejecución. Si se omite esta opción, la aplicación debe estar en el path del sistema en el sistema remoto.

-d No espere a que la aplicación termine. Utilice esta opción para aplicaciones no interactivas

-e No se carga el perfil de la cuenta especificada.

-f Copia el programa especificado en el sistema remoto, incluso si el archivo ya existe en el sistema remoto.

-i Ejecute el programa para que interactúe con el escritorio de la sesión especificada en el sistema remoto. Si no hay ninguna sesión se especifica el proceso se ejecuta en la sesión de consola.

-l proceso de ejecución como usuario limitado (las bandas del grupo Administradores y permite sólo privilegios asignados

al grupo Usuarios). En Windows Vista, el proceso se ejecuta con baja integridad.

-n Especifica tiempo de espera en segundos para conectarse a equipos remotos.

-p Especifica la contraseña opcional del nombre de usuario. Si se omite este se le pedirá que introduzca una contraseña oculta.

-s Ejecuta el proceso remoto en la cuenta del sistema.

-u Especifica el nombre de usuario opcional para iniciar sesión en el equipo remoto.

-v Copia el archivo especificado sólo si tiene un número de versión superior o es más reciente en que el uno en el sistema remoto.

-w Establece el directorio de trabajo del proceso (en relación con el equipo remoto).

-x Muestra la interfaz de usuario en el escritorio de Winlogon (sistema local solamente).

-Priority Especificar -low, -BelowNormal, -AboveNormal, -high o -realtime para ejecutar el proceso con una prioridad diferente. Uso de fondo para ejecutar en poca memoria y E / S de prioridad en Vista.

program Nombre del programa a ejecutar.

Arguments Los argumentos para aprobar (tenga en cuenta que las rutas de archivo deben ser rutas absolutas en el sistema de destino)

Usted puede encerrar las aplicaciones que tienen espacios en su nombre entre comillas, por ejemplo, "Psexec \ \ marklap" c: \. Nombre largo \ App.exe "Input sólo se pasa al sistema remoto cuando se presiona la tecla enter, y escribir Ctrl-C termina el proceso remoto.

Psfile

El comando “netfile” muestra una lista de los archivos que otros equipos han abierto en el sistema en que se ejecuta el comando, sin embargo, trunca los nombres de ruta largos y no le permite ver la información de los sistemas remotos. PsFile es una utilidad de línea de comandos que muestra una lista de archivos en un sistema que se abren a distancia, y

también le permite cerrar los archivos abiertos por nombre o por un identificador de archivo.

```
psfile [\\RemoteComputer [-u Username [-p Password]]] [[Id | path] [-c]]
```

-u Especifica el nombre de usuario opcional para iniciar sesión en el equipo remoto.

-p Especifica la contraseña para el nombre de usuario. Si se omite, se le pedirá que introduzca la contraseña sin que se reflejen en la pantalla.

Identificador Id (según lo asignado por PsFile) del archivo para el que desea mostrar información o cerca.

Ruta de acceso completa o parcial de los archivos de ruta para que coincida con la información para su visualización o cerrar.

-c Cierra los archivos identificados por ID o ruta

PsInfo

PsInfo es una herramienta de línea de comandos que obtiene información clave sobre el sistema local o remoto de Windows NT/2000, incluyendo el tipo de instalación, la construcción del kernel, la

organización social y el propietario, el número de procesadores y su tipo, la cantidad de memoria física, la instalación fecha del sistema, y si es una versión de prueba, la fecha de caducidad.

psinfo [[\\computer[,computer[,..] | @file [-u user[-p psswd]]] [-h] [-s] [-d] [-c [-t delimiter]] [filter]

\\computer	Ejecuta el comando en el equipo remoto o equipos especificados. Si se omite el nombre del equipo el comando se ejecuta en el sistema local, y si se especifica un comodín (\ \ *), el comando se ejecuta en todos los equipos del dominio actual.
@ file	Ejecuta el comando en cada equipo incluido en el archivo de texto especificado.
-u	Especifica el nombre de usuario opcional para iniciar sesión en el equipo remoto.
-p	Especifica la contraseña opcional del nombre de usuario. Si se omite este se le pedirá que introduzca una contraseña oculta.
-h	Muestra la lista de revisiones instaladas.

-s Muestra la lista de aplicaciones instaladas.

-d Muestra información del volumen del disco.

-c Imprime en formato CSV.

-t El delimitador predeterminado para la opción-c es una coma, pero puede ser anulado con el carácter especificado.

filter PsInfo sólo mostrará los datos del campo que coincida con el filtro. Por ejemplo "Servicio psinfo" enumera sólo el campo de Service Pack.

PsList

Al igual que la herramienta integrada en Windows NT/2K de supervisión PerfMon, PsList utiliza los contadores de rendimiento de Windows NT/2K para obtener la información que muestra. Puede encontrar documentación para los contadores de rendimiento de Windows NT/2K, incluyendo el código fuente de Windows NT incorporado en el Monitor de rendimiento, Monitor de rendimiento, en MSDN.

PsLogList

PsLogList es un clon de elogdump salvo que PsLogList le permite acceder a sistemas remotos en las situaciones de su actual conjunto de credenciales de seguridad no se permitirá el acceso al registro de sucesos, y recupera PsLogList cadenas de mensajes desde el ordenador en el que el registro de sucesos de ver reside.

PsloggedOn

PsLoggedOn determina quién está conectado mediante el escaneo de las claves en la clave HKEY_USERS. Para cada clave que tiene un nombre que es un usuario SID (identificador de seguridad), PsLoggedOn busca el nombre de usuario correspondiente y lo muestra. Para determinar quién ha iniciado sesión en un ordenador a través de los recursos compartidos, PsLoggedOn utiliza el NetSessionEnum API. Tenga en cuenta que PsLoggedOn le mostrará como conectado a través de compartir recursos a los equipos remotos que se consulta porque se requiere un inicio de sesión para PsLoggedOn acceder al Registro de un sistema remoto.

```
psloggedon [- ] [-l] [-x] [\\computername | username]
```

- Muestra las opciones admitidas y las unidades de medida utilizadas para los valores de salida.

-l Muestra sólo los inicios de sesión locales en lugar de los dos inicios de sesión de los recursos locales y de red.

-x No mostrar los tiempos de inicio de sesión.

\\computername Específica el nombre del equipo para el que desea mostrar información de inicio de sesión.

username Si se especifica un nombre de usuario PsLoggedOn busca en la red los equipos a los que se registra en ese usuario. Esto es útil si desea asegurarse de que un usuario particular no está conectado cuando está a punto de cambiar su configuración de perfil de usuario.

PsService

PsService es un visor de servicio y el controlador de Windows. Al igual que la utilidad de SC que está incluido en Windows NT y Windows 2000 kits de recursos, PsService muestra el estado, la configuración y las dependencias de un servicio, y le permite iniciar, detener, pausar, reanudar y reiniciar ellos. A diferencia de la utilidad de SC, PsService le

permite iniciar sesión en un sistema remoto utilizando una cuenta diferente, para los casos en que la cuenta desde la que se ejecuta no tiene los permisos necesarios en el sistema remoto. PsService incluye un exclusivo servicio de búsqueda de capacidad, que identifica las instancias activas de un servicio en su red. Usted puede utilizar la función de búsqueda si desea localizar los sistemas que ejecutan los servidores DHCP, por ejemplo.

```
psservice [\\computer [-u username] [-p password]] <command> <options>
```

query	Muestra el estado de un servicio.
config	Muestra la configuración de un servicio.
setconfig	Establece el tipo de inicio (desactivado, automático, demanda) de un servicio.
start	Inicia un servicio.
stop	Detiene un servicio.
restart	Detiene y reinicia un servicio.
pause	Detiene un servicio

cont	Reanuda un servicio pausado.
depend	Lista de los servicios dependienten de uno especificado.
security	Vuelca el servicio de seguridad descrito.
find	Busca en la red el servicio especificado.
\\computer	ataca el sistema NT/Win2K especificado. Incluya la opción-U con un nombre de usuario y contraseña para iniciar sesión en el sistema a distancia si sus credenciales de seguridad no le permiten obtener información de contadores de rendimiento del sistema remoto. Si se especifica la opción-u, pero no una contraseña con la opción-p, PsService le pedirá que introduzca la contraseña y no la echo a la pantalla.

ANEXO V - Unicode Attack

El infame exploit de Servidor Web IIS Unicode puede emplearse también como una herramienta de ataque de negación de servicio.

Gray hat (hacker Big Poop) ha publicado un sitio de internet donde hace referencia al bug Unicode, el cual permite la ejeción de comandos en un Servidor Web, y puede ser utilizado para restringir recursos de sistema para que usuarios legítimos no puedan accesar a un sitio web, una técnica clásica de ataque DoS.

Mark Read, un consejero de seguridad de MIS Corporate Defence Solutions, quien ha verificado el sitio de Big Poop, dijo que el exploit trabaja ejecutando un número de procesos en un servidor que no terminan, tales como un comp.exe (el cual espera indefinidamente hasta que los nombres de archivo a ser comparados sean ingresados)

"Debido a que el programa no termina, IIS mantiene la conexión abierta asumiendo que algo se está transfiriendo a través del navegador", dijo Reid. "Sin embargo, después de varias conexiones, IIS detendrá cualquier conexión en un intento por detener el servicio".

El exploit podría dar un intruso una buena oportunidad de dejar a un servidor web expuesto al bug Unicode utilizando un navegador de web y un simple procedimiento. Reiniciando el sistema, al menos, se interrumpirá la negación de servicio, pero este problema todavía causará seria preocupación, debido a la facilidad con la que se puede emplear este exploit.

Algunos expertos de seguridad dijeron que el ataque DoS Unicode, el cual puede ser automatizado a través del uso de scripts, es "más eficiente" que los más familiares ataques DDoS basados en red, los cuales comúnmente confían en programas Troyanos instalados en un rango de clientes "Zombies" comprometidos. Bajo el control del intruso, estos zombies envían una serie de comandos falsos contra los servidores víctimas con la intención de hacerlos sitios no disponibles.

A pesar de elevar la posibilidad de ataques de negación de servicio en los servidores vulnerables por medio del bug Unicode, Mark Read piensa que existe más probabilidad de que los intrusos alteren un sitio a que establezcan un ataque de negación de servicio contra ellos.

"Este es un ataque DoS, pero siendo honesto si el administrador del sistema no ha aplicado los parches correspondientes a la vulnerabilidad Unicode estonces es evidente que el servidor se encuentra totalmente vulnerable", es lo que dijo[39].

[39] (Leyden, 2001)

Anexo VI - Double Decode Attack

En mayo de 2001, los investigadores de NSFocus publicado un aviso sobre una vulnerabilidad de IIS que dio a luz una sorprendente similitud con el asunto del sistema de archivos transversal Unicode. En lugar de representaciones demasiado largas de barras Unicode (/ y \), NSFocus descubrió que el Unicode codificado doblemente con caracteres hexadecimales también permitió que las peticiones HTTP soliciten ser construidas para evadir los registros normales de seguridad de IIS y permitía accesos a recursos externos a la raíz de la Web. Por ejemplo, la barra invertida (\) puede representar a un servidor web por la notación hexadecimal % 5c. Del mismo modo, el carácter% está representado por 25%. Por ende, las cadenas 255c%, si fueran decodificados dos veces en secuencia, se traduce en una barra inversa.

La clave aquí es que dos decodificadores son necesarios, y esta es la naturaleza del problema con IIS: realiza dos decodificaciones de las peticiones HTTP que atraviesan los directorios ejecutables. Esta condición es explotable de la misma manera que el agujero Unicode[40].

Cabe destacar en este punto es que los ataques Unicode y Double Decode son tan similares, que la vullnerabilidad Unicode o Double Decode en hexadecimal

[40] (The McGraw-Hill Companies, 2005)

se pueden utilizar indistintamente en ataques si el servidor no ha sido parcheado para cualquiera de estas vulnerabilidades.

Anexo VII – Protocolo DCC

[41]Direct Client-to-Clientes un protocolo de IRC que permite interconectar dos peers (puntos) usando un servidor IRC como saludo (handshaking) para permitir intercambiar archivos o llevar a cabo tareas no relacionadas con el chat. Una vez conectados, una típica sesión de DCC corre independiente del servidor IRC. Originalmente diseñado para ser usado con IrcII es ahora soportado por varios clientes IRC.

La conexión DCC puede ser iniciada de dos formas diferentes:

- La forma más común es usando CTCP para iniciar una sesión DCC. El CTCP es enviado de un usuario sobre la red IRC hacia otro usuario.
- Otra forma es iniciar una sesión DCC en el cliente para conectarse directamente al servidor DCC. Usando este método, no necesita atravesar una red IRC (las partes involucradas no necesitan estar conectadas a una red IRC para usar DCC).

[41] (Rollo, 2004)

Iroffer es un servidor de archivos para IRC (comúnmente referido como un bot DCC. Este utiliza las características DCC de IRC para enviar archivos a otros usuarios. Iroffer puede conectarse a voluntad a un servidor IRC y dejar que la gente pida los archivos de la misma.

Glosario Técnico

Término Técnico	Descripción
Acceso Remoto	Acceso a redes informáticas desde una ubicación remota, en general localizada fuera de la red. Las redes VPN constituyen un ejemplo de tecnologías de acceso remoto.
Consola	Pantalla o teclado que permite obtener acceso al servidor, equipo mainframe u otro tipo de sistema y controlarlo dentro de un entorno de red.
Criptografía	Disciplina matemática e informática relacionada con la seguridad de la información, particularmente con el cifrado y la autenticación. En cuanto a la seguridad de aplicaciones y redes, es una herramienta para el control de acceso, la confidencialidad de la información y la integridad.
Dll	Dynamic-link library
DMZ	demilitarized zone o red perimetral es una red local que se ubica entre la red interna de una organización y una red externa, generalmente Internet
Exploit	Es una pieza de software, o una secuencia de comandos con el fin de causar un error o un fallo en alguna aplicación, a fin de causar un comportamiento no deseado o imprevisto en los programas informáticos, hardware, o componente electrónico
Firewall	Tecnología de hardware y/o software que protege los recursos de red contra el acceso no autorizado. Un firewall autoriza o bloquea el tráfico de computadoras entre redes con diferentes niveles de seguridad basándose en un conjunto de reglas y otros criterios.
FQDN	Fully Qualified Domain Name
FTP	File Transfer Protocol
Hashing	Proceso que vuelve ilegibles los datos de titulares de tarjetas convirtiendo los datos en un resumen de mensaje de longitud fija mediante la Criptografía sólida. El hashing es una función (matemática) en la cual un algoritmo conocido toma un mensaje de longitud arbitraria como entrada y produce un resultado de longitud fija (generalmente denominado “código hash” o “resumen de mensaje”).
Host	Computadora principal donde reside el software informático.
HTTP	Acrónimo de “hypertext transfer protocol” (protocolo de transferencia de hipertexto). Protocolo abierto de Internet que permite transferir o transmitir información en la World Wide Web

HTTPS	Acrónimo de “hypertext transfer protocol over secure socket layer” (protocolo transferencia de hipertexto a través de una capa de conexión segura). HTTP seguro que proporciona autenticación y comunicación cifrada en la World Wide Web diseñado para comunicaciones que dependen de la seguridad, tales como los inicios de sesión basados en la web.
IDS	Intrusion Detection System
IIS	Internet Information Services
IOCE	International Organisation for Cooperation in Evaluation
IP	Acrónimo de “internet protocol” (protocolo de Internet). Protocolo de capas de red que contiene información sobre direcciones y algunos datos de control, y permite el ruteo de paquetes. IP es el protocolo primario de capas de red en la suite de protocolos de Internet.
Logs	Un log es un registro oficial de eventos durante un rango de tiempo en particular.
MAC	Acrónimo de “message authentication code” (código de autenticación de mensajes). En criptografía, se refiere a la información breve que se utiliza para autenticar un mensaje.
MD5	Message-Digest Algorithm 5, (Algoritmo de Resumen del Mensaje 5) es un algoritmo de reducción criptográfico de 128 bits
MQSVC	Microsoft Message Queue Server
NTP	Network Time Protocol
PID	Process Identifier
Red	Dos o más computadoras interconectadas a través de un medio físico o inalámbrico.
Router	Hardware o software que conecta el tráfico entre dos o más redes Clasifica e interpreta la información mediante la comprobación de direcciones y transmisión de bits de datos a los destinos correctos. Algunas veces se denomina puerta de enlace al software de un router.
Script	Es un programa usualmente simple, que por lo regular se almacena en un archivo de texto plano. Los guiones son casi siempre interpretados, pero no todo programa interpretado es considerado un guion.
Servidor	Computadora que presta servicios a otras computadoras, como el procesamiento de comunicaciones, almacenamiento de archivos y acceso a impresoras. Los servidores incluyen entre otros: web, base de datos, aplicaciones, autenticación, DNS, correo, proxy y protocolos NTP.
SHA-1 y SHA-2.	Acrónimo de “Secure Hash Algorithm” (Algoritmo de hashing seguro). Una familia o conjunto de funciones criptográficas de ordenamiento relacionadas, que incluye SHA-1 y SHA-2.
Shell	El shell de comandos es un programa de software independiente que proporciona comunicación directa entre el usuario y el sistema operativo. La interfaz de usuario del shell de comandos no es gráfica y proporciona el entorno en que se ejecutan aplicaciones y utilidades

	basadas en caracteres.
TCP	Transmission Control Protocol
TELNET	Abreviatura de “telephone network protocol” (Protocolo de redes telefónicas). En general, se utiliza para proporcionar sesiones de inicio con líneas comandos orientadas al usuario para dispositivos de red. Las credenciales del usuario se transmiten en texto simple.
Timeline	En el área de computación forense se refiere a la cronología en que se realiza una instrusión informática.
UDP	User Datagram Protocol

Bibliografía

1. *El comando DD.* (28 de 05 de 2005). Obtenido de http://preguntaslinux.org/-guia-el-comando-dd-t-10.html
2. Brezinski, D., & Killalea, T. (2002). Obtenido de RFC 3227: Guidelines for Evidence Collection and Archiving. Network Working Group. February: http://www.rfceditor.org/rfc/rfc3227.txt
3. Cano Martines, J. J., Mosquera González, J. A., & Certain Jaramillo, A. F. (Abril de 2005). Obtenido de Evidencia Digital: contexto, situación e implicaciones nacionales: http://derecho.uniandes.edu.co/derecho1/export/derecho/descargas/texto/NasTecnologias6.pdf
4. Carrier, B. (s.f.). *sleuthkit.org.* Obtenido de http://www.sleuthkit.org/autopsy/docs.php
5. CASEY, E. (2004). Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet.
6. CERT/CC. (1988-2006). *Statistics.* Obtenido de http://www.cert.org/stats/cert_stats.html
7. Core Security Technologies. (17 de 09 de 2011). Obtenido de CoreSecuriy: http://www.coresecurity.com/content/vulnerability-report-for-windows-smb-dos
8. Database, N. V. (s.f.). Obtenido de CVE: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2001-0241
9. Dawson, K., & Kirch, O. (2002). *Guía de Administración de Redes con Linux.* Obtenido de http://linux.casa.cult.cu/docs/garl2/x-087-2-iface.netstat.html
10. Galarza, M. D. (2010). Obtenido de Informáticos, Equipo de Investigación de Incidentes y Delitos: WWW.EIIDI.COM
11. IOCE. (2002). *Guidelines for the best practices in the forensic examination of digital technology.* Obtenido de http://www.ioce.org/2002/ioce_bp_exam_digit_tech.html
12. IOCE. (s.f.). *International Organization of Computer Evidence*. Obtenido de http://www.ioce.org
13. Leyden, J. (3 de Julio de 2001). *The Register.* Obtenido de http://www.theregister.co.uk/2001/07/03/unicode_bug_restyled_as_dos/
14. Lima, M. d. (2007). Obtenido de http://www.monografias.com/trabajos17/delitos-informaticos/delitos-informaticos.shtml
15. López Delgado, M. (s.f.). *Análisis Forense Digital.* Obtenido de http://www.oas.org/juridico/spanish/cyb_analisis_foren.pdf

16. López, O., Amaya, H., & León, R. (2001). *Informática forense: generalidades, aspectos técnicos y herramientas.*
17. Martines Jeimy, J. C. (Junio de 2006). *Revista ACIS.* Obtenido de Introducción a la informática forense: http://www.acis.org.co/fileadmin/Revista_96/dos.pdf
18. MELBOURNE IT. (30 de 01 de 1999). Obtenido de Symantec Corporation: http://www.securityfocus.com
19. Nigel Titley. (s.f.). *Network Coordination Centre.* Obtenido de http://www.ripe.net
20. Noblett, M. G. (2000). *Recovering and Examining Computer Forensic Evidence.* Obtenido de FBI: http://www.fbi.gov/hq/lab/fsc/backissu/oct2000/computer.htm
21. Paller, A. (29 de 09 de 2012). Obtenido de The SANS Institute: http://www.sans.org/security-resources/malwarefaq/wnt-unicode.php
22. Reno, J. (28 de Octubre de 1996). U.S. Attorney General.
23. Rollo, T. (15 de Abril de 2004). *http://www.irchelp.org/irchelp/rfc/dccspec.html.* Obtenido de troy@plod.cbme.unsw.oz.au
24. Russinovich, M. (6 de Junio de 2012). *Windows Sysinternals.* Obtenido de http://technet.microsoft.com/en-us/sysinternals/bb896649.aspx
25. Symantec Corporation. (30 de 01 de 1999). Obtenido de SecurityFocus: http://www.securityfocus.com/bid/1578/discuss
26. The McGraw-Hill Companies. (2005). *Hacking Exposed Windows Server 2003.* New York: The McGraw-Hill Companies, Inc . Obtenido de http://www.techrepublic.com/i/tr/downloads/home/0072230614_chapter_10.pdf
27. Unblue. (s.f.). *ProcessLibrary.* Obtenido de http://www.processlibrary.com/es/directory/files/psexec/440243/#.UD_LXSKNFKZ