ESCUELA SUPERIOR POLITECNICA DEL LITORAL

"ANALIZAR, IDENTIFICAR Y DAR SOLUCIÓN A LAS VULNERABILIDADES DE BRJ SOFTWARE"

Facultad de Ingeniería en Electricidad y Computación

INFORME DE MATERIA DE GRADUACIÓN

Previa a la obtención del TÍTULO de:

LICENCIADO EN REDES Y SISTEMAS OPERATIVOS

Presentada por:

ANDREA MARGARITA CAYAMBE GUAMÁN
MARÍA FERNANDA NARANJO FRANCO

GUAYAQUIL – ECUADOR

AÑO 2013

AGRADECIMIENTO

A DIOS, quien nos ha acompañado día a día guiando nuestros pasos. Permitiéndonos culminar una de nuestras metas.

A nuestros padres, quienes nos inculcaron valores morales y éticos, brindándonos su apoyo incondicional, confiando plenamente en nuestras capacidades.

Al Ing. Albert Espinal Santana, quien mediante sus consejos, apoyo y experiencia nos guio para alcanzar esta meta. Gracias Albert por la confianza brindada.

A nuestros maestros, que con sus sabias enseñanzas y consejos nos han dado bases sólidas para desenvolvernos en el mundo laboral que nos espera.

DEDICATORIA

Agradezco a Dios, por llenarme de inspiración como de energía para seguir adelante, por bridarme la oportunidad de culminar este sueño.

Sin duda alguna, quiero agradecer a mis padres y hermana por siempre estar presentes a mi lado, ayudándome en todos los aspectos para que pueda hacer realidad cada uno de los sueños que me he propuesto en mi vida.

Andrea Margarita Cayambe Guamán

DEDICATORIA

Agradezco a Dios, por permitirme cumplir una meta más en mi vida y haberme guiado en todos estos años de estudios, a mis padres Javier e Ivonne que siempre han estado ahí para guiarme y apoyarme en todo momento y ser mis mejores ejemplos. A mi hermana la cual ha sido mi amiga incondicional.

A mis profesores y amigos que con sus consejos y sus palabras de aliento han estado siempre a mi lado y nunca permitieron que me rinda para así cumplir una de mis metas.

María Fernanda Naranjo Franco

TRIBUNAL DE SUSTENTACIÓN

_____

Ing. Karina Astudillo

PROFESOR DE LA MATERIA DE GRADUACIÓN

_____

Ing. Albert Espinal

PROFESOR DELEGADO POR LA UNIDAD ACADEMICA

DECLARACIÓN EXPRESA

"La responsabilidad del contenido de este Informe, nos corresponde exclusivamente; y el patrimonio intelectual de la misma a la Escuela Superior Politécnica del Litoral".

(Reglamento de Graduación de la ESPOL)

Andrea Margarita Cayambe Guamán

María Fernanda Naranjo Franco

RESUMEN

En los últimos años hablar sobre crímenes informáticos causa un gran impacto entre los habitantes del mundo, ya que desconocen el alcance que tiene la internet en manos de una persona con conocimientos sólidos o de una persona que sepa utilizar las vulnerabilidades de una organización a su conveniencia.

El presente proyecto consistió en el análisis de datos de una organización para determinar como el atacante pudo tener acceso a los recursos de dicha organización.

Para el desarrollo de este proyecto utilizaremos herramientas open source, que nos permita aplicar una metodología adecuada dependiendo del caso.
También usamos otras herramientas que poseen varias funciones muy útiles a la hora de analizar la evidencia, que incluyan un malware en su código.

Con este proyecto conseguimos que la organización realice cambios para corregir las vulnerabilidades existentes y tenga una seguridad informática robusta.

ÍNDICE GENERAL

HERRAMIENTAS PARA LA SOLUCIÓN E IMPLEMENTACIÓN

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

ABREVIATURAS Y SIMBOLOGÍA

ATA	Interfaz de transferencia de datos entre la placa madre y un dispositivo de almacenamiento.
DLL	Biblioteca de vínculos dinámicos
FAT	Sistema de archivos
FTP	Protocolo de Transferencia de Archivos
GNU/GPL	Proteger la libre distribución del software
MBR	Primer sector de disco
NTFS	Sistema de archivos de Windows
SCSI	Interfaz estándar de transferencia
SSH	Intérprete de órdenes segura
UID	Identificador de usuario

INTRODUCCIÓN

Este proyecto responde al propósito de ofrecer, un resumen actualizado sobre este tema. Mucho se ha escrito sobre esto; por su índole y finalidad se ha desarrollado a base de investigación; dando así un aporte positivo a este hecho de estudio.

Desde sus inicios la humanidad ha presenciado crímenes, pero crímenes informáticos no existían anteriormente. En la actualidad existe un gran desarrollo de herramientas y tecnologías las mismas que ayudan a cometer este tipo de delitos.

Es así como el objetivo de este proyecto es resolver un caso basándonos en las evidencias obtenidas y los hallazgos realizados haciendo uso de la metodología de la computación forense, aplicando todos los conocimientos adquiridos durante el seminario de graduación.

Como primer punto, realizaremos una clara descripción de los conceptos básicos de la computación forense para así comprender sus técnicas.

Segundo, mencionaremos cada uno de los aspectos legales, que debemos tener en cuenta antes de realizar la recolección de la evidencia para que en lo posterior evitemos dificultades al momento de presentar el informe final, más aún si el informe va ser utilizado como parte de la evidencia en un proceso legal.

Tercero, detallaremos las herramientas que utilizaremos para proceder con el análisis de la información.

Cuarto, procederemos a realizar un reconocimiento del problema y ejecutaremos los pasos obligatorios; para localizar cual fue la vía que tomó el atacante para ingresar y cometer el delito.

Quinto, presentaremos los resultados y las conclusiones a las cuales hemos llegado, después de realizar un correcto análisis.

Y como último paso procederemos a dar las recomendaciones que creemos que deberían ejecutarse en la organización; para que no vuelvan a ser víctimas de estos tipos de ataques.

Adicionalmente proveeremos recomendaciones para que si ocurriera esto nuevamente, el personal de la empresa sepa los pasos que debe seguir para que la información pueda ser utilizada como evidencia en un juicio. Si el caso amerita ser llevado a los tribunales.

CAPÍTULO 1

ANTECEDENTES Y JUSTIFICACIÓN

1.1 Antecedentes

En la actualidad el avance tecnológico ha provocado el surgimiento de nuevas modalidades que inducen a cometer infracciones, así como también eludir a las autoridades; con lo cual han transformado al Internet, y las Tics, en herramientas hostiles.

El constante reporte de vulnerabilidades en los sistemas informáticos, el aprovechamiento de fallas bien sea humano, procedimental o tecnológico sobre infraestructuras de computación en el mundo, ofrecen un escenario perfecto para que se cultiven tendencias relacionadas con intrusos informáticos.

El auge de la informática forense hace su aparición como una disciplina auxiliar de la justicia moderna, ya que con ello se puede enfrentar desafíos y técnicas de intrusos informáticos, la evidencia digital puede aportar de la misma manera que la evidencia física en un proceso legal. Utilizando varias herramientas, las cuales nos permiten detectar y analizar evidencias, para así esclarecer un caso.

1.2 Justificación

Cada día se descubren nuevos puntos débiles y por lo general, son pocos los responsables de IT que comprenden en su justa medida, la importancia que tiene la seguridad y cómo pueden abordar el grave problema que existe detrás de vulnerabilidades, que permiten a un atacante violar la seguridad de un entorno y cometer delitos en función de los datos robados.

La finalidad de este informe es analizar la evidencia por medio de metodologías, herramientas y técnicas con el fin de obtener pruebas, y descubrir a los autores de dichas infracciones en donde se vean involucrados sistemas de información o redes para así resolver dicho caso.

1.3 Descripción del Proyecto

Para el desarrollo y recopilación de evidencias de nuestro proyecto nos hemos propuesto alcanzar los siguientes objetivos:

1.3.1 Objetivo General

- Recopilar evidencias digitales sobre el ataque ocurrido a la empresa BRJ Software, y así identificar las debilidades y riesgos para luego desplegar de manera inteligente estrategias de seguridad efectivas.
- Determinar el comportamiento de un malware dentro sistema operativo, basándonos en la recolección de la información, de forma estática, dinámica y online.

1.3.2 Objetivos Específicos

- Definir las herramientas que existen y que pueden ser utilizadas para una investigación forense.
- Analizar la evidencia digital, y poder detectar alguna incidencia o rastro de lo acontecido.

- Elaborar un informe claro, conciso, para así poder justificar el análisis ejecutado.
- Proporcionar sugerencias a la empresa para mitigar las vulnerabilidades encontradas de sus equipos.

1.4 Metodología

En el análisis de nuestro proyecto, los elementos que vamos a utilizar es una laptop en la cual está instalada una distribución de Linux basada en Ubuntu 10.04 que es CAINE 2.0 para poder efectuar las diferentes tareas como:

- Analizar Logs para evaluar y determinar qué fue lo que ocurrió.
- Documentar toda información válida para demostrar todo lo encontrado durante el Análisis Forense, y poder dar las conclusiones y recomendaciones del caso.

Crear un ambiente virtualizado para recluir al malware, en un ambiente seguro inmunizando al sistema anfitrión. Esto lo realizaremos:

- Determinando cuales son las herramientas que serán requeridas para el posterior análisis.

CAPÍTULO 2

MARCO TEÓRICO

2.1 Computación Forense

La computación forense es una ciencia que nos permite obtener, mantener y mostrar los datos que hayan sido procesados electrónicamente y posteriormente almacenados de manera digital.

Cabe recalcar que no es utilizada solo para investigar crímenes tecnológicos, como lo son el ingreso no autorizado a una red o la distribución de material ilegal, sino que también es utilizada para investigar cualquier crimen donde una computadora puede tener alguna evidencia almacenada (por ejemplo, correo electrónico).

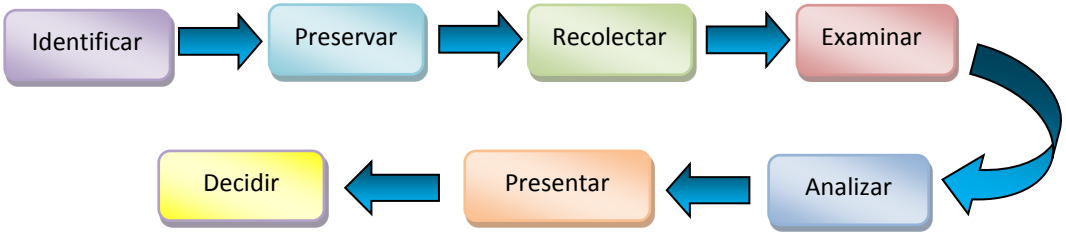


Figura 2.1: Proceso de Investigación

Fuente: Patrón Lineal de la Investigación Cuantitativa, Elaborado: Las Autoras

2.2 Propósitos de la Computación Forense

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

Estos propósitos se obtienen de diferentes métodos, para así poder realizar un verdadero análisis de lo ocurrido.

2.3 Uso de la Computación Forense

El uso de la computación forense en primera instancia se emplea como medida preventiva; para que la organización o empresa pueda auditar y saber cuáles son sus vulnerabilidades de seguridad con el fin de corregirlas, evitando

futuros ataques que ocasionarían grandes pérdidas. Muchos de estos usos provienen de la vida diaria, y no tienen que estar directamente relacionados con la computación forense, como lo son:

- Persecución Criminal.
- Litigación Civil.
- Investigación de Seguros.
- Temas corporativos.
- Mantenimiento de la ley.

2.4 Funcionalidad

La computación forense es una herramienta indispensable que toda empresa debe contemplar dentro de su política de seguridad y enmarcar dentro del proceso de respuesta a incidentes en los sistemas informáticos.

La protección de las evidencias, firmas digitales, controles de acceso a los usuarios, vigilancia y las implicaciones legales asociadas, conllevan a plantear nuevos modelos de gestión de seguridad.

Por ende la computación forense, permite realizar la búsqueda de manera minuciosa de lo que produjo el incidente o hurto de información.

Por otro lado, cuenta con procedimientos que permiten recuperar información que haya sido eliminada del medio físico o lógico.

2.5 Importancia de la Computación Forense

Hablar de computación forense hace algún tiempo atrás era casi inadmisible e impensable, pero con el crecimiento desaforado de la tecnología esto ya es un hecho palpable.

Existen individuos con altos conocimientos técnicos, los cuales los emplean para ocasionar perjuicio monetario o solo para satisfacción personal.

Para determinar qué fue lo que ocurrió se necesita personal entrenado para llevar a cabo una investigación de manera exhaustiva.

El uso de la computación forense no está orientado solo a delitos o fraudes informáticos, sino también es un excelente instrumento para poder realizar un autoanálisis, del uso inapropiado de los recursos, de la organización y de las vulnerabilidades de seguridad; tales como el mal uso del correo electrónico corporativo, el uso del internet para fines personales, y la violación de las políticas de seguridad de la empresa.

2.5.1 Forense en Redes

Para realizar un análisis de manera exhaustiva en la red de una organización, es necesario que el auditor tenga sólidos conocimientos, no solo de técnicas de computación forense, sino como actúan los protocolos de enrutamiento, la configuración de los equipos de red, etc.

2.5.2 Forense Digital

Las computadoras como los medios de comunicación son utilizadas en los fraudes informáticos, por eso existen procedimientos específicos para la criminalística informática.

Para determinar cuáles son las falencias existentes en la seguridad de la organización que se haya visto afectada por estos delitos informáticos. Y así poder establecer acciones legales pertinentes.

Esto nos permite esclarecer ciertas interrogantes, que son muy comunes, como lo son: ¿Quién?, ¿Cómo?, ¿Dónde?, ¿Cuándo?, ¿Por qué?

2.6 Ventajas de Computación Forense

Las ventajas que obtenemos con la computación forense son muchas indiscutiblemente. Algunas de ellas son las siguientes:

- Rapidez en la recolección de la información y análisis de la misma.
- Garantizar la efectividad de las políticas de seguridad y la protección no solo de la información sino de las tecnologías aplicadas.
- Optimizar las posibles vulnerabilidades existentes en el sistema.
- Recolectar evidencia digital presente en toda clase de infracciones.
- Asegurar el menor costo en la gestión de procesos, sin la presión de una situación de emergencia.

2.7 Dificultades de la Computación Forense

Pueden relacionarse con el personal, los procesos, la documentación de los sistemas, etc. Aquí enumeramos algunas:

- No contar con los registros de auditoría. Esto puede suceder, porque el aplicativo no los tiene implementados o si los tiene, están desactivados (la entidad podría justificar que los Logs están degradando la máquina).
- Registros incompletos o no claros de las pistas de auditoría. Esto ocurre porque solo se graban algunos campos, para no cargar el sistema o no existen descripciones detalladas de los Logs.
- No se realiza un buen levantamiento de información de la arquitectura del sistema. Esto se dificulta determinar la forma y quién realizó la transacción fraudulenta.
- Poca habilidad en el manejo de las herramientas.
- Resistencia por parte de los funcionarios para suministrar información porque no les agrada ser investigados o porque podrían estar relacionados con el ilícito.
- Restricción de acceso a la información de la entidad. Si se cuenta con el conocimiento y las herramientas necesarias, los funcionarios de seguridad informática y/o auditoría de sistemas de la entidad podrían adelantar la investigación forense y no habría mayor dificultad en el acceso a la información; pero si se requiere que por la especialización del tema lo realice un tercero, éste investigador deberá trabajar de

manera estrecha con las áreas de seguridad bancaria, jurídica y la auditoría de sistema.

2.8 Políticas de Seguridad

Las políticas de seguridad son una forma de establecer reglas o normas en una organización, para que ésta no esté expuesta a ningún tipo de riesgo; y así también establecer responsabilidades a los usuarios del uso correcto de los recursos y servicios.

La RFC 1244 define a las políticas de seguridad como: “Una declaración de intenciones de alto nivel; que cubre la seguridad de los sistemas informáticos y que proporciona las bases para definir y delimitar responsabilidades; para las diversas actuaciones técnicas y organizaciones que se requerirán”.

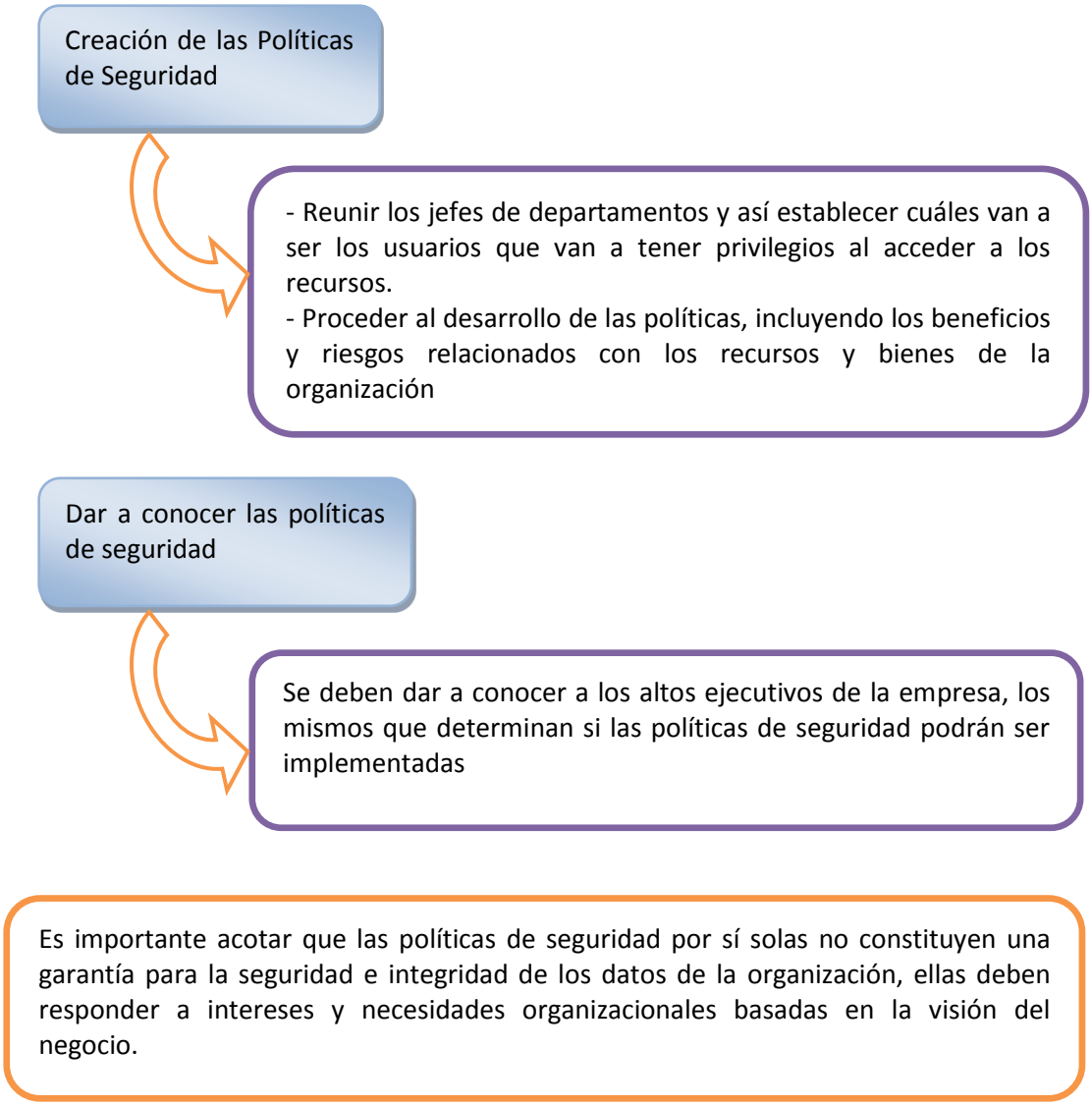


Figura 2.2: Procedimiento de las Políticas de Seguridad

Elaborado: Las Autoras

A continuación mencionaremos algunos parámetros, que se deben tener presentes al momento de definir las políticas de seguridad:

- Adecuarse a las necesidades y recursos de la organización.
- Las políticas deben proveer integridad, disponibilidad autenticidad y utilidad.
- Minimizar los puntos vulnerables existentes dentro de la infraestructura de la organización.
- Desarrollar planes de contingencia para solucionar futuros desastres.
- Auditar y ejercer un control a dichas políticas de seguridad.
- Tener un respaldo de la información relevante de la organización, la misma que debe ser actualizada de forma periódica y debe reposar en una ubicación diferente a la original.

2.9 Aspectos Legales en el Ecuador

El tema de delitos informáticos no es algo que ocurre solo en países de primer mundo. Sino que puede suceder en cualquier parte del mundo en la que exista acceso al Internet, ya que es una puerta abierta a este tipo de delitos.

Por eso Ecuador no es la excepción, las leyes se han ajustado a estos cambios, en la nueva constitución que toma en cuenta estos delitos para que sean juzgados y reciban la sentencia correspondiente.

Las Reformas al Código Penal de las Infracciones Informáticas incluyen los ataques que se producen contra el derecho a la intimidad, a la obtención y utilización no autorizada de información, sabotajes informáticos, falsificación electrónica, daños informáticos, apropiación ilícita. La seguridad jurídica y la seguridad informática, se deben complementar entre sí.

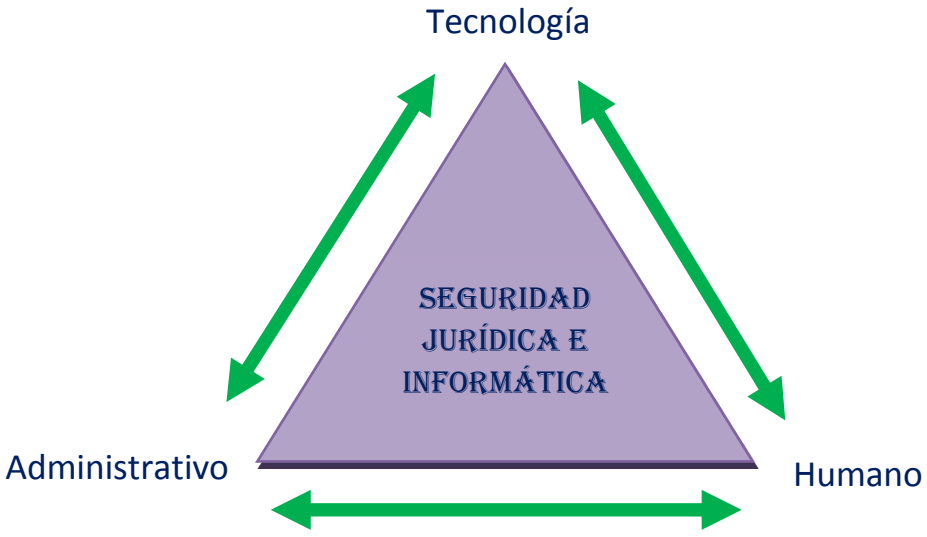


Figura 2.3: Seguridad Jurídica e informática

Fuente: Evidencia digital reflexiones técnicas, administrativas y legales, Elaborado: Las autoras

Las leyes deben fortalecer la evidencia digital, tomando en cuenta que es válida dentro de cualquier procedimiento legal.

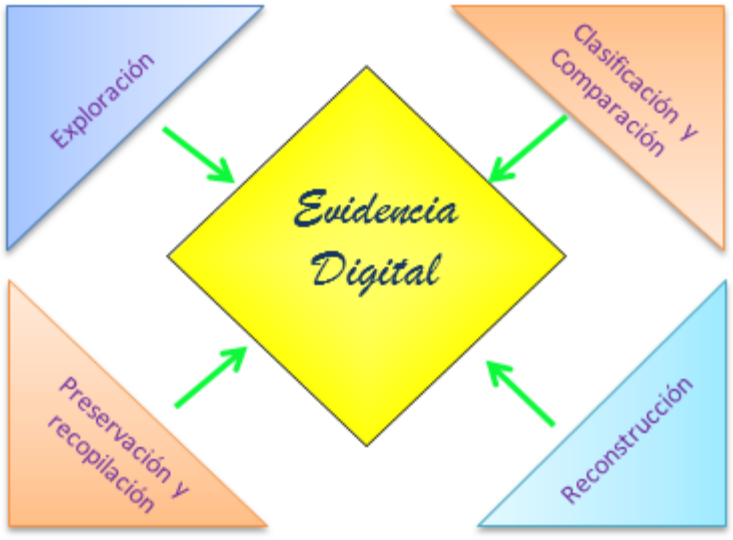


Figura 2.4: Evidencia Digital

Fuente: Evidencia digital reflexiones técnicas, administrativas y legales, Elaborado: Las autoras

A continuación podremos ver los artículos con sus penalidades correspondientes al delito cometido, estas leyes se ejercen en el Ecuador.

Tabla 1.- Art 58 delitos contra la información protegida
Fuente: Perfil sobre los delitos informáticos en el Ecuador, Elaborado: Las autoras

ART 58 DELITOS CONTRA LA INFORMACION PROTEGIDA	PENALIDAD CARCELARIA	PENA PECUNIARIA
Violentando claves o sistemas	6 meses a un año	US$500.00 a US$ 1.000.00
Información obtenida sobre la seguridad nacional, secretos comerciales o industriales	3 años	US$1000.00 a US$ 1.500.00
Divulgación o utilización fraudulenta de los rubros anteriores	3 a 6 años	US$2000.00 a US$ 10.000.00
Divulgación o utilización por funcionarios a cargo de dicha información	3 a 9 años	US$2000.00 a US$ 10.000.00
Obtención y uso no autorizados de datos personales para cederla o utilizarla	2 meses a 2 años	US$1.000.00 a US$ 2.000.00

Tabla 2.- Art 59 destrucción maliciosa de documentos por funcionarios de servicio público.
Fuente: Perfil sobre los delitos informáticos en el Ecuador, Elaborado: Las autoras

ART 59 DESTRUCCION MALICIOSA DE DOCUMENTOS POR FUNCIONARIOS DE SERVICIO PUBLICO	PENALIDAD CARCEARIA	PENA PECUNIARIA
	3 a 6 años	

Tabla 3.-Art 60 falsificación electrónica según el siguiente detalle y con ánimo de lucro con perjuicio a terceros.
Fuente: Perfil sobre los delitos informáticos en el Ecuador, Elaborado: Las autoras

ART 60 FALSIFICACIÓN ELECTRÓNICA SEGÚN EL SIGUIENTE DETALLE Y CON ÁNIMO DE LUCRO CON PERJUICIO A TERCEROS	PENALIDAD CARCEARIA	PENA PECUNIARIA
- **Alterar un mensaje de datos** - **Simulación de mensaje** - **Suposición de intervención en actos, declaración.**	Serán juzgados de acuerdo a lo que se dispone en este capítulo del artículo. Eso quiere decir 6 años	

Tabla 4.- Art 61 Daños informáticos.
Fuente: Perfil sobre los delitos informáticos en el Ecuador, Elaborado: Las autoras

ART 61 DAÑOS INFORMÁTICOS	PENALIDAD CARCEARIA	PENA PECUNIARIA
Daño fraudulento de información contenida en un sistema	6 meses a 3 años	US $ 60,00 a US$150,00
Cometido por funcionario público o vinculado a la defensa nacional	3 a 5 años	US$200,00 a US$600,00
Si no se tratare de un delito mayor, la destrucción, alteración o inutilización de infraestructura para la trasmisión	8 meses a 4 años	US$200,00 a US$600,00

Tabla 5.- Art 62 Apropiación ilícita
Fuente: Perfil sobre los delitos informáticos en el Ecuador, Elaborado: Las autoras

ART 62 APROPIACION ILICITA	PENALIDAD CARCEARIA	PENA PECUNIARIA
Uso fraudulento o ilícito para apropiación de un bien ajeno.	6 meses a 5 años	US $ 500,00 a US$1.000,00
Uso fraudulento mediante la utilización de los siguientes medios: - **Inutilización de sistemas de alarma o guarda.** - **Descubrimiento descifrado de claves secretas o encriptados.** - **De tarjetas magnéticas, carding o perforadas.** - **De controles o instrumentos de apertura a distancia.** - **Violación de seguridades electrónicas u otras semejantes.**	1 año a 5 años	US$1.000,00 a US$2.000,00

Tabla 6.- Art 63 Estafa a través de medios electrónicos
Fuente: Perfil sobre los delitos informáticos en el Ecuador, Elaborado: Las autoras

ART 63 ESTAFA A TRAVÉS DE MEDIOS ELECTRÓNICOS	PENALIDAD CARCEARIA	PENA PECUNIARIA
Daño fraudulento de información contenida en un sistema	1 año a 5 años	US $ 500,00 a US$1.000,00

Tabla 7.- Art 64 Derecho de la intimidad
Fuente: Perfil sobre los delitos informáticos en el Ecuador, Elaborado: Las autoras

ART 64 DERECHO A LA INTIMIDAD	PENALIDAD CARCEARIA	PENA PECUNIARIA
Si no fuera delito	La sanción aún está en sucres equivalente a casi centavos	De dos a cuatro días

2.10 Motivos de los ataques

La naturaleza de los motivos están identificados y se pueden incluir en:

- Venganza (es cometer algo mal intencionado para perjudicar).
- Codicia (Beneficio personal ya sea monetario o notoriedad).
- Rivalidad (Necesidad de poder monopolizar el mercado).
- Conducta compulsiva o adictiva para utilizar las vulnerabilidades existentes.
- Accidente o ignorancia (realizar un mal uso de los recursos con los que cuenta).

2.11 Tipos de Ataques

2.11.1 Ataques Externos

Éstos implican a los llamados piratas informáticos contratados por una entidad externa, cuyo objetivo es destruir la reputación del competidor; o de una persona inescrupulosa, que ataca las vulnerabilidades de la infraestructura de la organización, sin recibir a cambio una remuneración, solo la simple satisfacción de mandar abajo un sistema.

2.11.2 Ataques Internos

Éstos implican un abuso de confianza de los empleados, dentro de una organización. También son ocasionados por empleados insatisfechos en el ámbito laboral, o por ser separados de la empresa.

2.11.3 Ataques pasivos

Esto implica que el atacante no altera la comunicación, sino que únicamente la escucha o monitoriza, para obtener información que está siendo transmitida.

Sus objetivos son: la intercepción de datos y el análisis de tráfico, que puede consistir en:

Tabla 8.- Ataques Pasivos
Elaborado: Las autoras

ATAQUE	LO QUE HACE…
Adquirir el origen y destinatario	Leyendo Cabeceras de los paquetes transmitidos
Registrar el volumen de tráfico	Monitoreo de paquetes transmitidos sobre la actividad o inactividad de las transmisiones de datos
Registrar de las horas habituales	Monitoreo de paquetes transmitidos, para extraer información acerca de la actividad
Estos tipos de ataques son difíciles de detectar, porque no provocan modificación en los datos. Pero, es posible evitar estos ataques mediante el cifrado de información u otros mecanismos.	

2.11.4.- Ataques activos

Estos ataques implican algún tipo de modificación en los paquetes transmitidos o la creación falsos paquetes de datos, pudiendo subdividirse en cuatro categorías:

Tabla 9.- Ataques Activos
Elaborado: Las autoras

ATAQUES	LO QUE HACEN…
Suplantación de identidad	El intruso se hace pasar por una entidad o usuario diferente, accediendo a ciertos recursos suplantando una identidad que posee privilegios como contraseñas o accesos a cuentas.
Re-actuación	Es cuando uno o varios mensajes son capturados y repetidos para producir algo no deseado, como por ejemplo ingresar dinero repetidas veces en una cuenta dada.
Modificación de mensajes	Es cuando un mensajes puede ser alterado, reordenados para causar un efecto no autorizado Este tipo de ataque se lo considera como fraudes informáticos.
Degradación fraudulenta del servicio	Impide el uso normal o la administración de recursos informáticos y de comunicaciones. En este tipo de ataques esta la denegación de servicios, ya que puede paralizar ciertos servicios.

2.12 Evidencia

La evidencia física se la debe manejar con sumo cuidado más aún si la evidencia es computacional, por lo que se debe realizar una copia por seguridad, dado que si la evidencia es alterada se pierde la garantía del caso.

La evidencia computacional nos permite analizar hallazgos concernientes a robo de secretos comerciales, fórmulas y software propietarios entre otros.

Los investigadores deben revisar con frecuencia que sus copias sean exactas a la original. Para esto utilizan varias tecnologías como checksum o hash MD5.

2.12.1 Pasos Para la Recolección de Evidencia

El procedimiento para la recolección de evidencia varía dependiendo del caso. Sin embargo, existen procedimientos básicos que pueden ayudar al investigador forense:

- **Lineamientos Generales para la Recolección de Evidencia**

  Realizar una copia bit a bit de la evidencia, información valiosa se encuentra “escondida” dentro del disco.

  Los investigadores deberán ser capaces de demostrar que la evidencia que se encuentra en la red interna, externa de una empresa o en el internet, es auténtica y no ha sido modificada mientras estaba siendo recolectada.

  La IOCE define los siguientes cinco puntos como los principios para el manejo y recolección de evidencia computacional:

  1. Al recolectar la evidencia digital, las acciones tomadas no deben cambiar por ningún motivo esta evidencia original.

2. Cuando es necesario que una persona tenga acceso a evidencia digital original, esa persona debe 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 esta esté en su posesión.

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

- **Identificación de la Evidencia Digital**

Determinar qué elementos pueden ser de interés por el intruso y cuáles no. Tomar fotografías del entorno en el que ha sido el crimen informático, sin olvidar de documentar todos los procesos que se están llevando a cabo.

2.12.2 Cuidados en la Recolección de la Evidencia

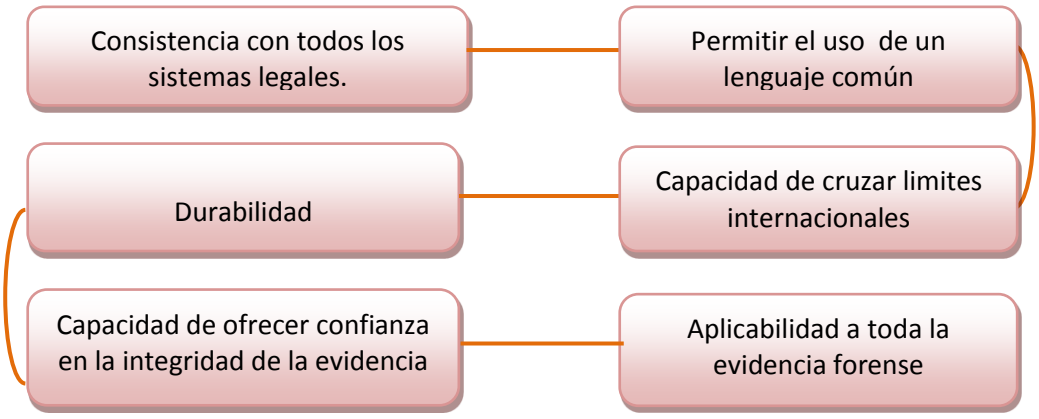


Figura 2.5: Cuidados en la Recolección de la Evidencia

Elaborado: Las autoras

- **Preservación de la Evidencia Digital**
  - La evidencia para un caso puede incluir un equipo completo y los medios asociados a dicho equipo.
  - Recoger evidencia informática, utilizando bolsas anti estáticas.
  - Guarde los datos en un entorno con temperatura y humedad pre especificado.

Figura 2.6: Preservación de la Evidencia Digital

Fuente: http://hacking.mx/computo-forense/el-maletin-del-investigador-forense/

2.12.3 Análisis de la Evidencia

Involucra aquellas tareas orientadas a localizar y extraer evidencia digital relevante para la investigación.

Mediante la aplicación de diversas técnicas y herramientas forenses se intenta dar respuesta a los puntos de pericia solicitados como son:

- El análisis de datos requiere un trabajo interdisciplinario entre el perito y el operador judicial –juez, fiscal- que lleve la causa.
- Tareas que se llevan a cabo dependiendo del tipo de investigación.
- Búsqueda de palabras claves o documentos, en todo el espacio de almacenamiento del dispositivo investigado.

- Determinar si ciertas aplicaciones fueron utilizadas por un determinado usuario.
- Determinar qué tipo de actividad tenía el usuario en la Web, análisis del historial de navegación, análisis de correo electrónico, etc.

Figura 2.7: Análisis de la Evidencia

Fuente: http://www.bsecure.com.mx/opinion/la-importancia-de-la-evidencia-y-el-analisis-forense-digital/

2.12.4 La Cadena de Custodia

La Cadena de custodia, es más que llevar un registro de todos los acontecimientos que se dan, ya que éste es uno de los pilares fundamentales para garantizar la responsabilidad de los involucrados.

- **Los pasos para la Cadena de Custodia**

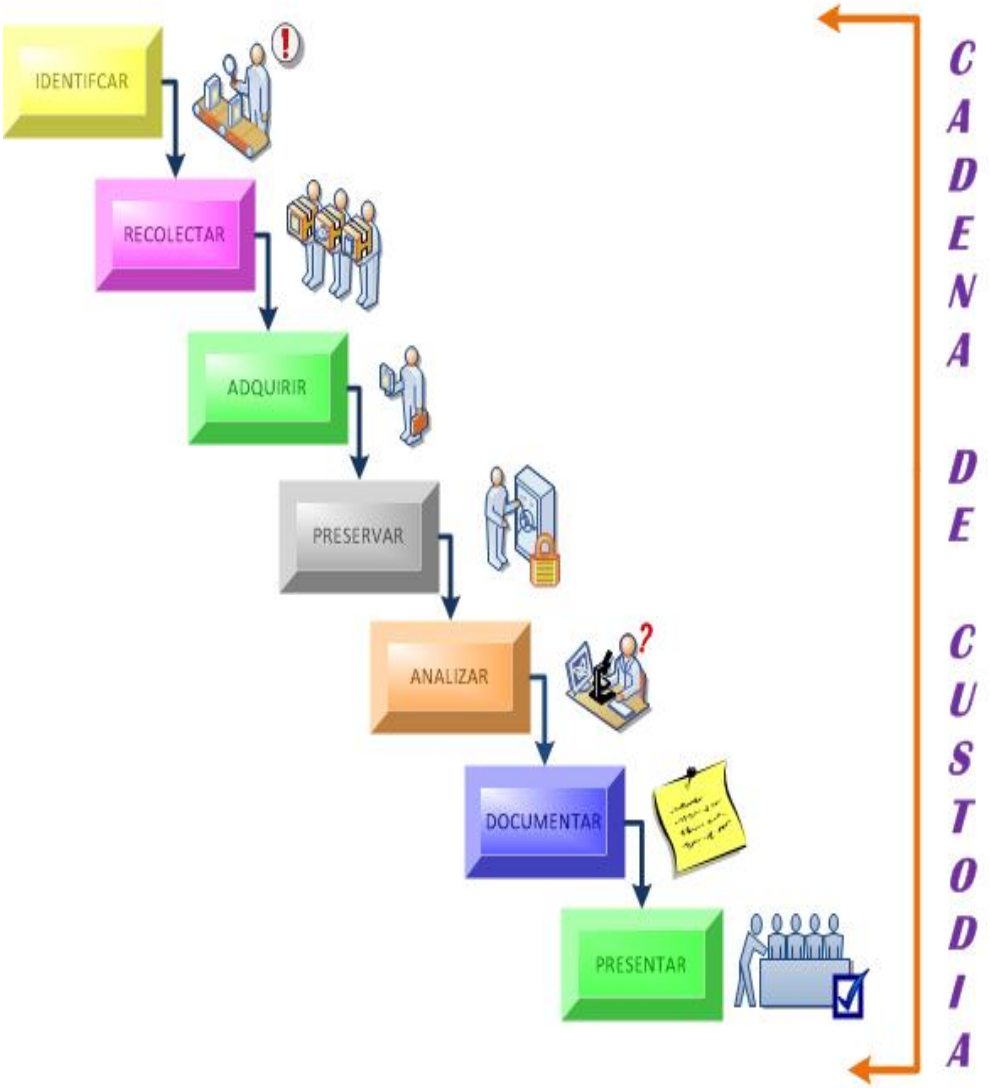


Figura 2.8: Proceso de la cadena de custodia

Fuente: Informática forense: Oscar López, Elaborado: Las autoras

- **Mantenimiento de la cadena de custodia**

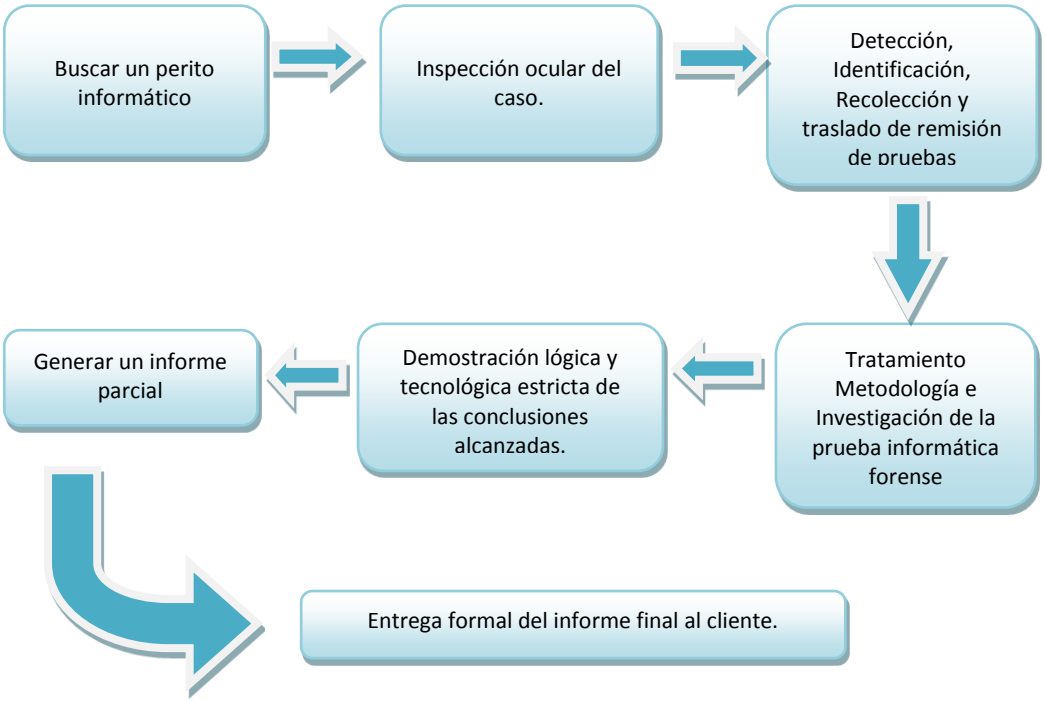


Figura 2.9: Mantenimiento de la Cadena de Custodia

Fuente: Informática forense Oscar López, Elaborado: Las autoras

2.13 Herramientas para el análisis forense

2.13.1 Herramientas Comerciales

Las herramientas comerciales cuentan con un software robusto, el mismo que tiene un precio bastante caro, por eso su implementación la realizan empresas

grandes las cuales requieren que su información se mantenga salvaguardada de cualquier tipo de ataque, estas herramientas cuentan con licencia.

Tabla 10.- Herramientas comerciales para análisis forense
Fuente: 101 utilidades forense, Elaborado: Las autoras

NOMBRE	PLATAFORMA	VERSION
Spector CNE investigador	Windows	7.0
Encase	Windows	7.03
FTK	Windows	4.0.1
SafeBack		3.0
Nuix	Windows	4.0.1

2.13.1.1 Encase

Encase proporciona herramientas que permiten que el análisis se realice de forma profunda tomando en cuenta toda la información, sin desestimar la más mínima evidencia, por lo que el investigador puede manejar grandes volúmenes de evidencia sin ningún problema, de esta manera logra un informe completo y conciso del siniestro suscitado.

De esta manera optimizan el tiempo y aumentan el rendimiento. Entre sus características tenemos:

- Incluyen la programación En scripts, permitiendo crear scripts para automatizar tareas de análisis prolongados.
- Compatibilidad con diferentes sistemas de archivos.
- Compatibilidad con RAID avanzada.
- Vista de línea de tiempo expandida.

2.13.1.2 Forensic toolkit

Forensic toolkit es una herramienta concisa, calcula valores hash MD5 y confirma la integridad de los datos, realiza cracking de contraseñas todo dentro de una interfaz intuitiva. FTK es conocida por el análisis de correos electrónicos.

- Procesamiento más veloz gracias al procesamiento distribuido.
- Categorización automática de archivos.
- Análisis de RAM (32 bits y 64 bits).
- Adquisiciones remotas.
- Reporte más completos y flexibles.
- Soporta archivos de evidencia encriptados.

- Interfaz en español.

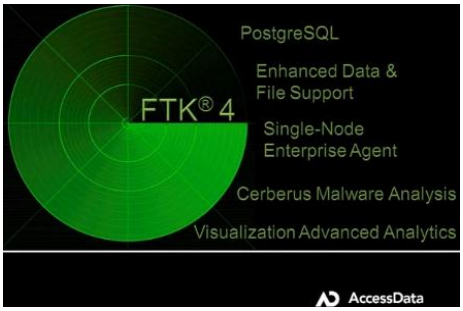


Figura 2.10: Forensic toolkit

Fuente: http://www.cimasix.com/herramientas-forense Forensic Toolkit (FTK)

2.13.2 Herramientas Open Source

2.13.2.1 Caine

Caine es una distribución GNU/LINUX, construida a base de la estructura de Ubuntu. Es una herramienta totalmente open source ya que podemos utilizarla sin necesidad, de pagar por algún tipo de licencia de registro.

Ofrece una interfaz gráfica amable, ofrece un proceso semiautomático durante la documentación.

Caine nos brinda los siguientes programas para realizar un análisis:

- Mount manager
- Air

- Grissom Analyzer
- Autopsy 2.20
- Guymaker
- Storage device manager

2.13.2.2 The Sleuth kit

The Sleuth kit es una colección de herramientas de análisis forense de volumen de sistemas y archivos, permitiendo examinar los sistemas de archivos de la computadora víctima.

Soporta particiones DOS, particiones BSD, particiones Mac, particiones Sun y disco GTP.

Entre sus características podemos destacar las siguientes:

- Es una herramienta que tiene una interfaz gráfica amigables.
- Permite el acceso a estructuras de archivos y directorios de bajo nivel, genera una línea temporal de actividad de los archivos.
- Muestra el detalle de información sobre datos eliminados y estructuras del sistema de ficheros.
- Permite crear notas al investigador para generar un reporte.

- Puede identificar donde se ubican las particiones y extraerlas de manera que pueda ser analizadas de forma segura.

2.14 ¿Qué son los Malware?

Los malware también llamados software malicioso o código malicioso, pertenecen a una familia numerosa como lo son: virus, troyano, gusano, joke, keyloggers, etc... Que tienen como función principal causar un incorrecto funcionamiento de su sistema tales como modificar, dañar, eliminar archivos del sistema, o propagarse a otros equipos.

Los malware pueden replicarse y enviarse por sí mismo de modo automático a otros equipos cuando consiguen controlar determinados programas de software.

De manera silenciosa, dañan el sistema del equipo y de otros equipos si se encuentran en red.

Figura 2.11: Malware

Fuente: http://commons.wikimedia.org/wiki/File:Malware_logo.svg

2.14.1 Cómo se propagan los Malware

En la actualidad permanecemos gran parte de nuestro tiempo conectados al internet, no sería errado decir entonces que el software malicioso está ahí intangible esperando atraparnos con algunos de su señuelos, para infectar nuestras máquinas o sustraer datos confidenciales.

Existen un sin número de medios como:

- A través de la red local de la empresa o de la casa.
- En las páginas webs que con solo entrar a ellas nos pueden infectar.

- A través del e-mail al ejecutar algún archivo que nos envían o pulsando en algún enlace.
- Wireless
- A través de pendrive, CD o DVD, que hayan sido grabados en un equipo infectado.

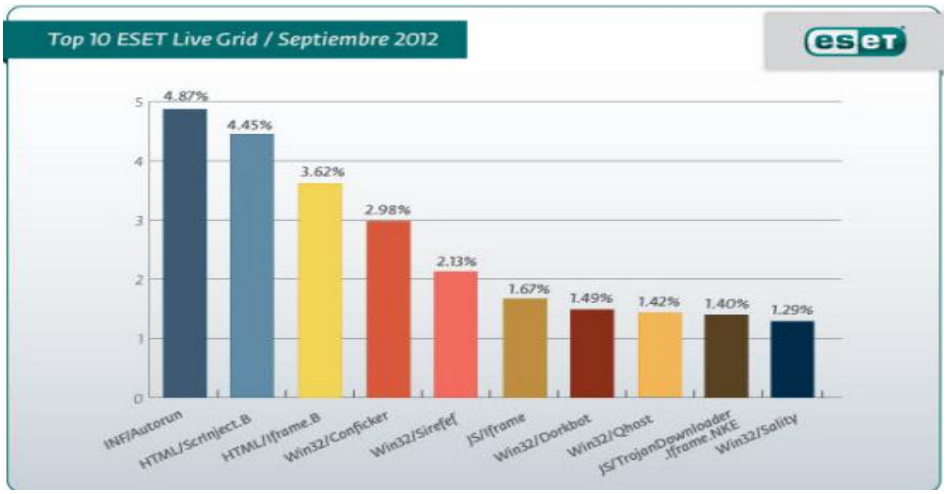


Figura 2.12: estadística de cuales son archivos infectados por malware
Fuente: Top 10 ESET live Grid/Septiembre 2012.

2.14.2 Clasificaciones de los Malware

Tabla 11: Clasificación de los malware
Fuente: Definición de tipos de virus, Elaborado: Las autoras

CLASIFICACIÓN DE LOS MALWARE	
VIRUS	Son programas creados para infectar el sistema
SPYWARES	Son programas creados para recopilar información sobre las actividades realizadas por un usuario y distribuirlas a agencias de publicidad u otras organizaciones interesadas.
PHISHING	Practica para obtener información confidencial como (número de cuentas, tarjetas de crédito, etc). Estos ataques son llevados a cabo a través de un scan (email falso).
PHARMING	Explotación de vulnerabilidades en un servidor DNS, permitiendo que el atacante modifique los registros de dominio de una página; para re direccionarla hacia otra.
ROOTKITS	Son técnicas que modifican el sistema operativo de un ordenador; para permitir que el malware permanezca oculto al usuario.
ADWARE	Son programas que muestran publicidad al usuario de manera intrusiva; en forma de ventanas pop-up de cualquier forma, esta publicidad aparece inesperadamente en el equipo y resulta muy molesta para el usuario.
BACKDOORS	Son métodos para eludir los procedimientos habituales de autenticación al conectarse a un ordenador, siendo mas fácil para el malware permanecer oculto ante una posible inspección.
HIJACKERS	Software encargado de cambiar la página de inicio de cualquier navegador, imposibilitando al usuario de cambiarla.

KEYLOGGER	Son programas encargadas de almacenar en un archivo todo lo que el usuario a ingresado por el teclado, para posterior envió al creador quien puede obtener beneficios económicos.
STEALERS	Son programas que roban la información privada que se encuentra guardado en el equipo

2.14.3 Cómo prevenir los ataques de Malware

La prevención contra los malware o códigos maliciosos es el punto clave para garantizar la completa protección de los equipos informáticos existentes en la organización o empresa.

A continuación se detalla alguno de estos puntos:

- Tener instalados en los equipos antivirus y antispyware o aplicaciones con la combinación de ambas.
- Utilizar las herramientas de desinfección; para escanear dispositivos de almacenamientos externos; llámense estos pendrive, disco duros.
- Evite utilizar software sin licencia.
- Analice todo archivo que descargue en especial las carpetas comprimidas ya que pueden contener códigos maliciosos.
- Si cuenta con un correo electrónico de la organización o empresa en la que labora utilícela solo para fines laborales.

- No abra ningún archivo adjunto proveniente de un desconocido, en especial si son .exe o zip/.rar
- No abra enlaces si no conoce al remitente de dicho correo.
- Tome las precauciones del caso al chatear con usuarios desconocidos.

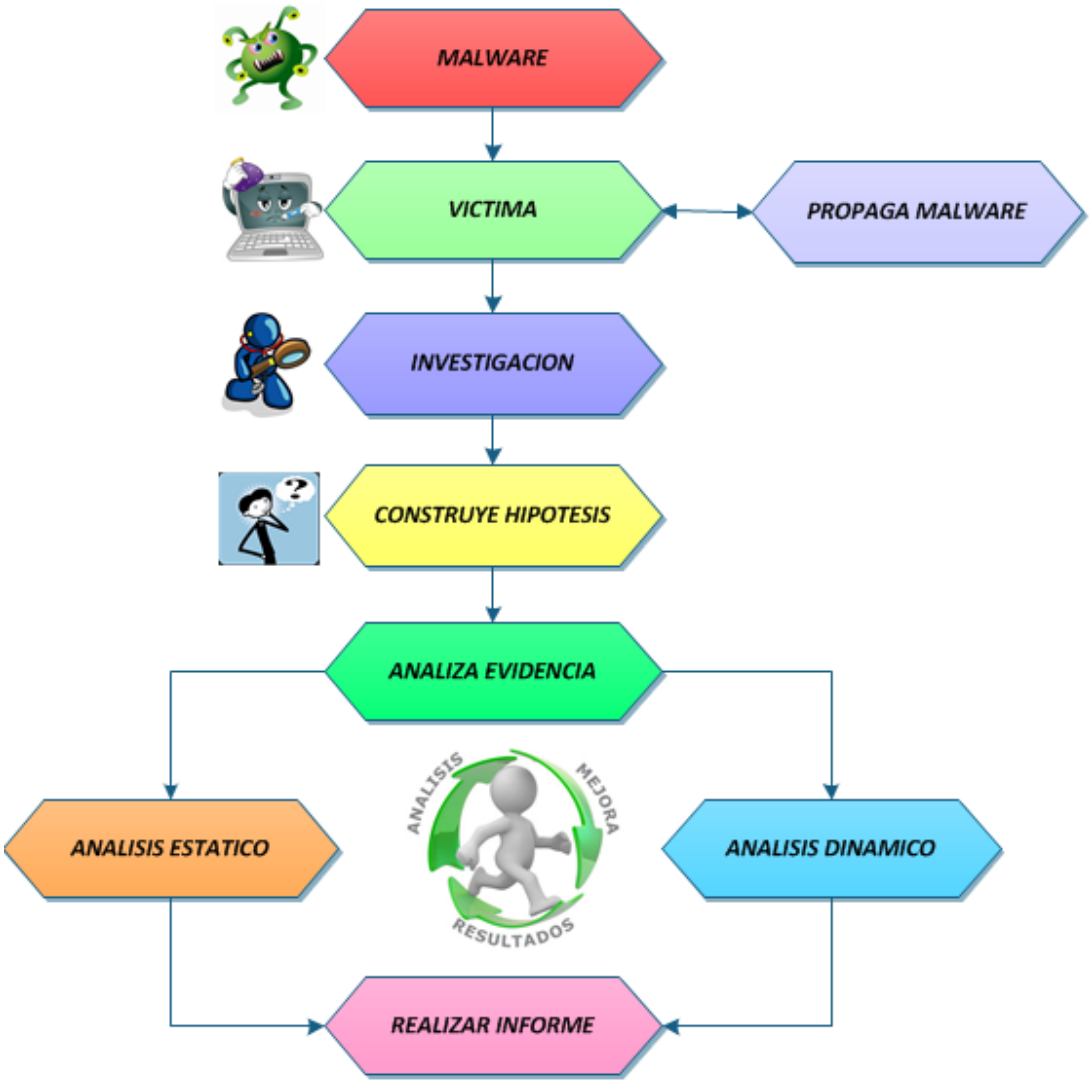


Figura 2.13: Proceso de los malware

Fuente: software dañino, Elaborado: Las autoras

2.14.4 Crear un entorno de trabajo

Para realizar cualquier tipo de análisis se debe manejar un ambiente, adecuado ya que este es un punto transcendental, que se debe tomar en cuenta antes de realizar el análisis propiamente dicho. La opción que más se utiliza es realizar los análisis a base de un entorno virtual.

Los sistemas más conocidos de virtualización son:

- VMWare
- VirtualBox
- Virtual PC

La máquina que se utilice para la investigación, se debe encontrar aislada del segmento de red. La línea base debe ser tomada en dos tiempos, la primera antes del que el malware sea ejecutado y una después de que el malware sea inyectado al sistema.

Esto permitirá realizar un balance para determinar cuáles han sido las partes afectadas por el código.

Estos cambios se pueden producir en los registros, sistemas de archivos, puertos, etc.

2.14.5 Recolección de la información

La recolección de la información debe realizarse utilizando un proceso planeado paso a paso, que de forma coherente se puedan obtener resultados que contribuyan favorablemente al logro de los objetivos propuestos.

La búsqueda de la información se realiza con base en los elementos del problema, una vez identificadas las necesidades de información se elabora los métodos de recolección de información. Se recomienda realizar una evaluación, por medio de una prueba.

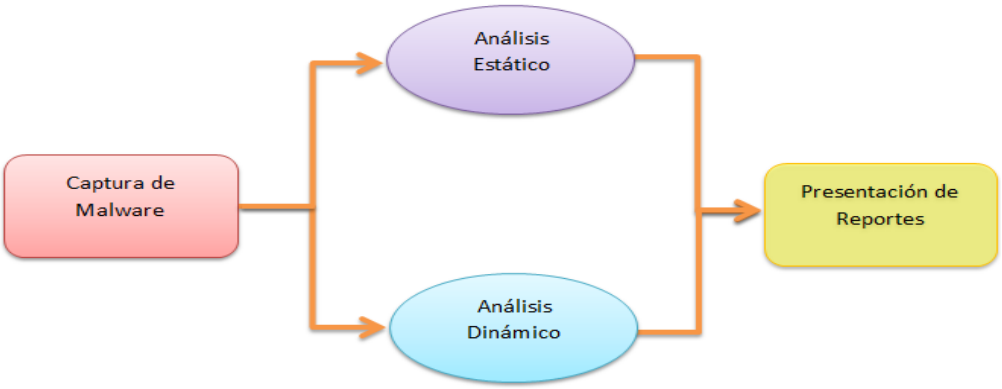


Figura 2.14: Recolección de la información

Fuente: Análisis forense de sistemas informáticos, Elaborado: Las autoras

2.14.5.1 Estático

En la recolección de la información de manera estática, se basa en obtener información del malware sin ejecutarlo, esta información puede ser el nombre del archivo, la versión, tamaño, etc. Esto tipo de análisis se realiza para tener un mayor entendimiento de lo que realiza el malware dentro del sistema.

Entre sus características tenemos:

- Correr varios antivirus
- Abrir un editor hexadecimal
- Búsqueda de strings

2.14.5.2 Dinámico

Se basa en el comportamiento que se presenta cuando se ejecuta en un sistema. Se debe tomar una línea base para identificar las similitudes que se presenten entre las diferentes herramientas que se utilicen.

Entre los archivos que se ven afectados por la inyección del malware al sistema están los archivos de sistema, proceso, cambios en la actividad de la red, etc.

Toda la información recolectada se la usará para el posterior análisis, la recolección de información de manera dinámica se la puede realizar cuantas veces crea necesario el investigador.

2.15 Herramientas para el análisis de malware

2.15.1 Análisis de comportamiento

Una vez ya creado un entorno seguro para el análisis del malware tenemos que identificar los puntos en el que el sistema se ha visto afectado por el malware.

Para ello existen herramientas que puntualizan el sitio que se desea revisar entre ellas tenemos:

- Process monitor que nos muestra registro de ficheros.
- Process explorer que nos muestra el proceso en memoria, los procesos, procesos que se crean, etc.
- Wireshark es un analizador de protocolos que captura datos que circulan en la red, sirva para determinar cómo actúa la red en un determinado intervalo de tiempo.
- Regshot sirve para controlar los cambios que se hacen en el registro de Windows, ya sean estos de por instalación, desinstalación o por el uso del equipo.

2.15.2 Análisis de código

El análisis de código es la parte que requiere determinación y paciencia, para poder saber qué hace el software.

¿Cómo funciona? ¿Cómo opera? ¿Qué decisiones toma en base a qué factores? Debemos saber todo el desenvolvimiento del malware, para esto existen herramientas que nos ayudarán a que esta tarea no sea engorrosa.

Entre las cuales tenemos a las siguientes:

- OllyDbg es un depurador de código ensamblador de 32 bits para sistema operativos Windows, analiza el código binario, de gran utilidad si se posee el código fuente.
- IDAPro es un desensamblador, que soporta diferentes sistemas operativos. Es un software comercial por lo robusta que es pero cuenta con una versión gratuita que es la versión 5.0
- LordPE nos va a ser de mucha utilidad ya que se utiliza para realizar el volcado de memoria .Lo que generalmente se lo llama como "DUMP".

2.15.3 Análisis online

El análisis online nos permite hacer uso de las herramientas disponibles en la web. En donde se realiza un análisis rápido de varios antivirus a la vez el cual, arroja datos considerables dentro del respectivo análisis.

Entre las herramientas disponibles en Internet tenemos las siguientes:

- Virscan proporciona un servicio gratuito para analizar ficheros con o sin compresión, en buscada de virus o infecciones que estén ocasionando problemas en el sistema donde está hospedado el archivo.

- Virustotal es un servicio que ofrece escanear virus, permitiendo al usuario subir el archivo o ingresando la url del mismo. Permite analizar ficheros de hasta 32 MB.
- Metascan es una herramienta para análisis de virus de forma online, la misma cuenta con 31 motores, lo que lo hace más preciso a la hora de detectar la amenaza, puede escanear archivos de hasta 50 MB como tamaño máximo al momento de cargar el archivo.
- Anubi analiza malware de forma gratuita, generando un informa que contiene la información necesaria para comprender cuales son los cambios que se han realizado.

2.15.4 Ingeniería Inversa

La ingeniería inversa es una metodología, que nos permite analizar cualquier tipo de sistema, ya sea con la finalidad de duplicar o mejorar dicho sistema.

La ingeniería inversa puede ser utilizada en cualquiera de las áreas que involucren, es aplicable a sistemas con las siguientes características:

- Documentación inexistente o totalmente obsoleta.
- Programación en bloque de códigos muy grandes o sin estructura.
- La aplicación está sujeta a cambios frecuentes, que pueden afectar a pare del diseño original.

La redocumentación también forma parte de la ingeniería inversa, es el proceso mediante el que se produce una documentación retroactiva desde un sistema existente, es usada para ayudar al conocimiento del programa.

2.15.4.1 Metodología

La metodología para realizar el proceso de ingeniería inversa incluye los siguientes pasos:

- Definición y delimitación del componente de software
- Recolección de funcionalidades existentes
- Diagrama automático mediante una herramienta semi-automatizada
- Análisis de código y relación entre paquetes

2.16 Estándares

Los estándares son normas, acuerdos documentados que contienen especificaciones técnicas u otros criterios precisos para ser usados consistentemente como reglas, guías o definiciones de características.

Para asegurar que los materiales productos, procesos y servicios se ajusten a su propósito. En la seguridad informática existen estándares que nos ayudan a sobre guardar la credibilidad del análisis del caso.

2.16.1 Encase

Encase es una herramienta creada para el análisis forense, pero debido a que ha sido perfeccionada, para manipular la complejidad cada vez mayor de las configuraciones y contenidos informáticas. Se la puede definir como un estándar a seguir para los investigadores forenses.

Las NIST definen que Encase Imaging Enginer, opera con mínimos defectos de credibilidad, otorgada por profesionales del campo informático, corte de justicia y agentes independientes.

Trabaja de la mano con la norma internacional ISO 17799:2000, el mismo es un estándar dentro de la seguridad informática. Fijando su estructura de la siguiente manera:

- Política de seguridad de la información.
- Organización de la seguridad.
- Clasificación y control de activos.
- Seguridad del personal.
- Seguridad física y ambiental.
- Gestión de comunicaciones y operaciones.
- Control de accesos.
- Desarrollo y mantenimiento de sistemas.
- Administración de la continuidad de los negocios.
- Cumplimiento.

2.17 Certificados Internacionales

Las certificaciones profesionales son un proceso por medio del cual una persona prueba que cuenta con sólidos conocimientos, experiencia y las cualidades necesarias para realizar un trabajo específico como:

- Ser parte de un selecto grupo de profesionales altamente solicitado internacionalmente.
- Adherir a buenas prácticas reconocidas internacionales.
- Contar con acceso y comunicación con todos los profesionales certificados.

Algunas de las certificaciones mejores remuneradas son las relacionan con la seguridad informática, las principales aéreas de interés son:

- Informática Forense
- Gestión y análisis de incidentes
- Análisis de intrusiones
- Auditoría
- Hacking Ético

CHFI (Computer Hacking Forensic Investigator) es la certificación oficial de investigador forense acreditada por el EC-Council. El objetivo de esta certificación es adquirir conocimientos prácticos sobre el proceso de detección de ataques de hackers y extraer, de la forma apropiada, evidencias digitales que sirvan para informar del crimen y dirigir auditorías para prevenir futuros ataques.

Las certificaciones del EC-Council obtuvieron la acreditación ANSI 17024, garantizando los procesos de certificación, exámenes, apelaciones, proctoring, etc. del más alto nivel.

En nuestro país contamos con empresas las cuales nos capacitan sobre la informática forense nos brindan lineamientos para poder acceder a certificaciones internacionales. Como Elixicorp nos brinda el curso de computación forense cuyo objetivo es, que el profesional cuente con conocimientos generales necesarios para poder llevar a cabo una investigación de computación forense siguiendo los pasos apropiados.

CAPÍTULO 3

HERRAMIENTAS PARA LA SOLUCIÓN E IMPLEMENTACIÓN

3.1 Organización interna de laboratorio forense

Para empezar la revisión de un caso de fraude o crimen computacional, primero debemos establecer un espacio para realizar este trabajo. Puesto que necesitamos una estación de trabajo para poder realizar las distintas pruebas y estudios, lo más recomendado seria tener mínimo 2 estaciones de trabajo una con acceso a internet, o tres estaciones dependiendo de qué tan complicado sea el ataque recordando siempre que en la estación que se realiza la manipulación de la evidencia no debe tener acceso a Internet ni a una red interna.

Tareas que se realizan en una estación de trabajo forense:

- 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.
- Se puede usar de software para crear una plataforma de trabajo con máquinas virtuales.
- Se puede crear un entorno de trabajo hipotético con las copias obtenidas para realizar la emulación de los ataques.

3.2 Herramientas para el análisis forense

3.2.1 Caine

CAINE es una distribución GNU/Linux en formato live CD su origino como un proyecto universitario sobre la informática forense elaborado por Giancarlo Guistini.

CAINE está basado en la infraestructura en Ubuntu Linux, el mismo que proporciona un entorno de análisis con las herramientas forenses más destacadas del mundo Open Source.

De la misma manera nos brinda un perfecto entorno forense que está organizado para integrar herramientas de software existente y proporcionar una interfaz gráfica amigable para el auditor forense o investigador, permitiendo al auditor obtener replicas idénticas de las unidades físicas en una estación forense, de igual manera es posible realizar todo el proceso de análisis forense, para luego finalizar con la creación de un reporte el mismo que ayudara a encontrar las posibles falencias en seguridad dentro de la organización tanto a nivel de infraestructura como de colaboradores.

3.2.2 Objetivos de Caine

El objetivo principal de Caine es permitir al auditor forense o investigador realizar varios análisis utilizando diferentes herramientas, optimizando el tiempo de respuesta del mismo, con un margen de error mínimo o casi imperceptible, claro está esto dependerá la vasta experiencia del auditor.

Figura 3.15: interfaz gráfica de caine

Fuente: caine-live.net

3.2.3 Características de Caine

Las principales características de caine apuntan a garantizar lo siguiente:

- CAINE ha sido creado desde Ubuntu 8.04 utilizando Remastersys.
- Proporciona una interfaz gráfica homogénea que guía a los investigadores digitales durante la adquisición y el análisis de las pruebas electrónicas.
- Fácil interoperabilidad durante todo el análisis (Preservación, Recolección, Análisis, Reportes).

- Ofrece un proceso semi-automático durante la documentación y generación de informes y resultados.
- CAINE soporta los siguientes lenguajes: Inglés, Italiano, Francés, Alemán y Portugués.
- Tiene herramientas de apoyo, como editores y visores gráficos de documentos. Utilidades del sistema, de redes, de dispositivos, de administración, etc.

3.2.4 Cómo funciona caine

Caine contiene herramientas que son capaces de realizar una réplica exacta a la original física, tiene dos modos de funcionamiento un para Windows y el otro para Linux.

Windows

- Para Windows se creó una nueva interfaz denominada WinTaylor incluida en el live CD de Caine.
- Para maximizar la compatibilidad con sistemas Windows antiguos está escrito en VB6.
- Facilita un conjunto interno de varios programas forense conocidos.
- Windows: FTK. Nigilant. RAM: MDD, Win32dd, WinEN

GNU/LINUX

- Basado en Ubuntu.
- Contiene herramientas forenses para realizar la captura de evidencia y de un análisis forense.
- Todas las herramientas forenses incluidas en el sistema operático son Open Source.
- GNU/Linux: AIR (Automated Image & Restore), GuyMager, DC3DD.
- Modo Consola: dcfldd, dd, etc.

3.2.5 Como está organizado Caine

Caine está organizado en cuatro grupos los cuales son:

Tabla 12: Organización de las herramientas de Caine
Fuente: tutorial-de-Linux-caine, Elaborado: Las autoras

GRISSOM ANALYZER	Herramientas que permiten el análisis de imágenes o copias bit a bit.
COLLECTION	Herramientas que permiten realizar la captura y creación de las réplicas bit a bit de los dispositivos de almacenamiento del sistema
ANALYSIS	Herramientas que permiten realizar el análisis forense, recuperar archivos eliminados, esteganografia etc.
REPORT	Permite generar un reporte en diversos formatos

3.2.6 Montado de dispositivos

La política de montado de un dispositivo interno o externo adoptado por caine jamás se montara de manera automática.

Cuando el auditor hace clic al dispositivo el sistema lo montará de modo solo lectura.

Si el auditor decide montar utilizando un terminal, puede utilizar el comando “mount” pero deben especificarse todas las opciones de montaje.

Si el usuario desea montar y escribir en un medio NTFS debe de utilizar el comando “ntfs-3g" (Ejemplo: sudo ntfs-3g /dev/sda1 /media/sda1).

Figura 3.16: Montado de dispositivos
Fuente: tutorial-de-Linux-caine

3.2.7 Herramientas de Caine

Captura con CAINE

- **AIR:** (Automated Image and Restore). es un GUI para dd/dcfldd diseñada para crear de manera fácil imágenes forenses bit a bit.
- **Guymager:** Es una herramienta de réplica para adquisición de medios. Es rápido debido a que es multitarea y puede hacer uso completo de máquinas multiprocesador. Y puede generar réplicas en dd, EWF (EO1)y AFF.
- **Autopsy 2.21:** El navegador forense Autopsy es una interfaz gráfica para las herramientas de investigación digital en línea de comando The Sleutht Kit. Juntas, permiten investigar el sistema de archivos y volúmenes de computadoras.
- **Bash Script Tools:** Es un conjunto de herramientas y scripts para realizar análisis forense digital. Estas herramientas y scripts se desarrollaron principalmente por la gente de la comunidad CFI-Computer Forensics Italy (Cómputo Forense de Italia).
- **PhotoRec:** Es un software de recuperación de datos diseñado para recuperar archivos perdidos incluyendo vídeo, documentos y archivos de discos duros y CD, además de fotos perdidas de memorias de cámaras. digitales. PhotoRec ignora el sistema de archivos y va directo a los datos subyacentes.

- **TestDisk:** Es un software libre muy poderoso para realizar la recuperación de datos. Principalmente desarrollado para ayudar a recuperar particiones perdidas y/o hacer que un disco, el cual no se inicia o “arranca”, en un disco que pueda iniciarse nuevamente cuando estos síntomas son causados por fallas de software, ciertos tipos de virus o un error humano.

3.3 Herramientas de análisis de malware

3.3.1 Análisis online

3.3.1.1 Virscan

Virscan es una herramienta que proporciona su servicio de analizar ficheros de manera gratuita a los usuarios del internet. El análisis se lo realiza con diferentes antivirus, para garantizar la efectividad del informe que facilitará al finalizar el análisis.

El utilizar herramientas de análisis online no reemplaza el hecho de tener instalado un programa de antivirus para proteger la pc.

3.3.1.2 Anubi

Anubi es una herramienta para analizar el comportamiento ficheros, en los que se aduce que contiene malware. La ejecución de análisis genera un reporte (pdf, txt, html, xml) indicando de manera detallada acerca de las modificaciones realizadas en el registro del sistema.

El análisis se basa en correr el binario en un entorno simulado.

Entre las tareas que desarrolla Anubi se encuentran las siguientes:

- Clave de registro
- Análisis de la red
- Ficheros modificados
- Actividad de disco

3.4 Virtualización

Cuando se realiza con análisis de malware o código malicioso, no se debe realizar el análisis en un sistema en producción para probar el malware. La forma más segura es la utilización de las máquinas virtuales, aislando totalmente el comportamiento del malware del equipo real.

Para tener un ambiente seguro se procederá a instalar VirtualBox de Oracle, el sistema operativo que se elija instalar para realizar el análisis dependerá de las plataformas que soporte el malware.

Una vez ya instalado el sistema operativo, se debe instalar las herramientas que se utilizara para monitorizar la actividad del código malicioso.

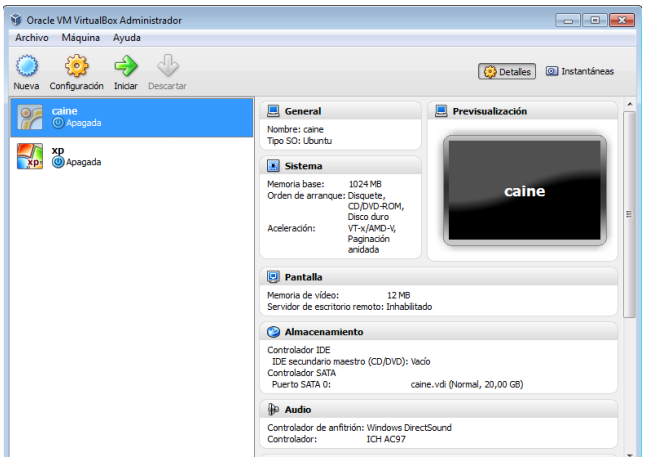


Figura 3.17: interfaz de VirtualBox

Fuente: Entrono virtual para el análisis, Elaborado: Las autoras

3.5 Análisis de código

3.5.1 Depuradores

- **OllyDbg** es un depurador para sistemas Windows de 32 bits, su análisis se enfatiza en el código binario del fichero, la función de este programa es editar programas exe y dll. Traduciendo el código del programa para facilitar su interpretación.

  Este depurador gratuito no cuenta con una versión para Windows de 64 bits así que para ciertos análisis este serían un problema.

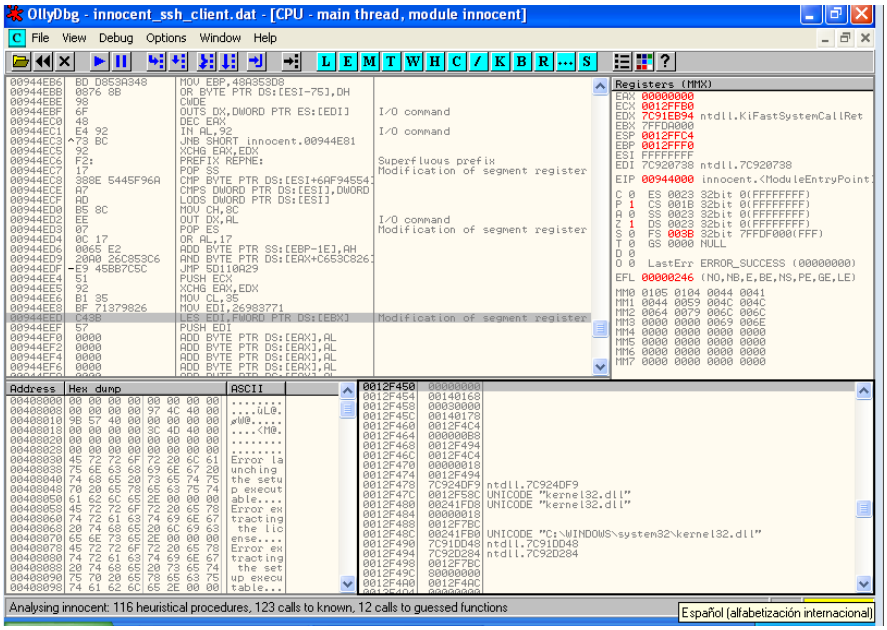


Figura 3.18: Ventana principal de OllyDbg
Fuente: Entorno virtual Windows xp, Elaborado: Las autoras

- **IDA PRO** es un desensamblador y depurador al mismo tiempo resultando esto muy versátil, es compatible con diferentes plataformas. Es un sistema interactivo, programable, prorrogable que soporta una diversidad de formatos ejecutables. Posee una utilidad muy robusta que se considera un estándar al momento de analizar códigos maliciosos. Realizar análisis automático del código, usando referencias cruzadas entre las secciones del código

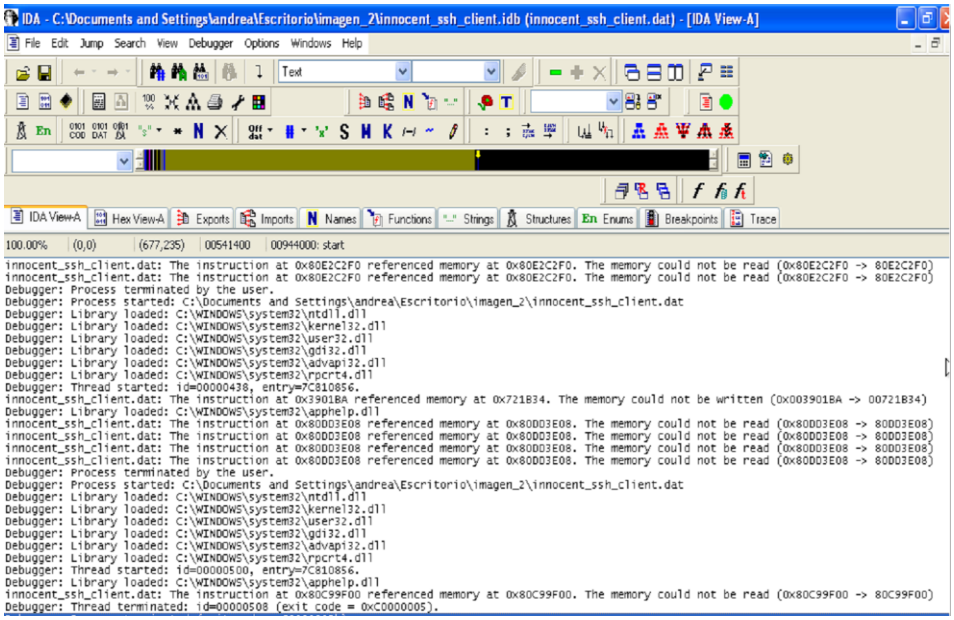


Figura 3.19: Ventana principal de IDA
Fuente: Entorno virtual Windows xp, Elaborado: Las autoras

3.6 Análisis de comportamiento

3.6.1 Process monitor

Es una aplicación que monitoriza el funcionamiento del equipo, explorando todo lo que ocurre a nivel de archivos, registros y actividades de procesos. Permite que el análisis se direccione hacia algo específico, toda actividad capturada por el Process monitor puede ser revisada cuando se guarda en un fichero. La información obtenida del Process monitor es completa pues incluye todos los detalles del proceso, una imagen de la ruta de la línea de comando, el usuario y el ID de sesión.

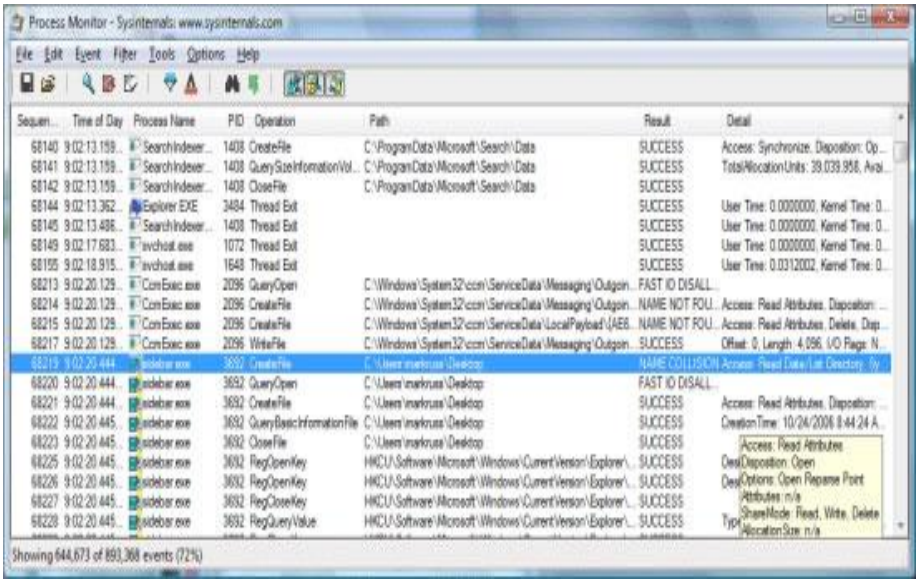


Figura 3.20: Process Monitor
Fuente: Entorno virtual Windows xp, Elaborado: Las autoras

3.6.2 Process explorer

El Process explorer es una herramienta que posee más funcionalidades que el administrador de tareas, permite no solo ver la información detallada de cada proceso sino sus íconos, línea de comandos, la ruta completa del ejecutable, estadísticas de uso de memoria, la cuenta del usuario al que le pertenece, etc.

El Process explorer permite visualizar los procesos en una forma jerárquica, permitiendo ver con claridad que proceso depende de otro, cuenta con un código de colores el mismo que permite identificar cada uno de los procesos que se están efectuando.

- Verde proceso iniciados.
- Rojo proceso se ha terminado o está por finalizar.
- Rosa procesos del sistema operativo.
- Lila indica que son procesos corrientes, que se utilizan con frecuencia para tareas cotidianas.
- Gris oscuro indica que son procesos que se encuentran pausados o suspendido.
- Azul procesos ejecutados por el usuario.
- Café tareas programadas.
- Morado procesos comprimidos.

CAPÍTULO 4

Desarrollo del Proyecto

4.1 Información preliminar

En el presente capítulo se procederá a desarrollar la parte del análisis de la evidencia que fue proporcionada, por el instructor la misma que se divide en dos partes:

- Brj_Software
- Innocent_ssh_client

Utilizando las metodologías, herramientas y técnicas adquiridas en el seminario.

4.2 Introducción

4.2.1 Brj_Software

BRJ Software es una empresa de ingeniería de software pequeña que realiza aplicaciones de soporte para las entidades financieras. En el departamento de BRJ software se trabaja y se administra varios servidores.

Los desarrolladores de software de la empresa crea el código fuente de los productos de sus cliente en los servidores que ellos controlan, también son los responsables de mantener el nombre de dominio de su empresa. Por lo que su dirección de correo electrónico aparece en la información de contacto de acceso público WHOIS.

4.2.2 Innocent_ssh_client

Innocent_ssh_client es un archivo aparentemente malicioso que se encuentra comprimido y se ejecuta en una plataforma de sistemas Windows.

Además es un archivo ejecutable, que ha sido introducido en una máquina que de aquí en adelante llamaremos “anfitrión”.

4.3 Justificación

4.3.1 Justificación del trabajo en Brj_Software

Brj_Software, se ha visto afectada por un siniestro informático originado en el departamento de desarrolladores de software. El personal de IT alarmados por la situación, capturan el tráfico de red de la máquina involucrada en el siniestro y la graban en un CD-R, que es entregado a su superior inmediato. El mismo que solicita se efectúe un estudio e investigación exhaustiva hasta lograr identificar a él o los responsables.

4.3.2 Justificación del trabajo en Innocent_ssh_client

Innocnt_ssh_client es un archivo ejecutable que puede ocasionar que se borre o dupliquen archivos del sistema operativo, lo que impediría el funcionamiento normal del mismo.

4.4 Objetivos

4.4.1 Objetivos de Brj_Software

Los objetivos del presente caso de computación forense son:

- Aplicar metodologías, herramientas y técnicas que permitan identificar las causas del siniestró informático.
- Establecer controles informáticos que eviten siniestros y aseguren una mayor confidencialidad de la información.
- Conocer los mecanismos que se usó para la manipulación de los datos.
- Identificar al o los responsables del siniestro.

4.4.2 Objetivos de Innocent_ssh_client

Los objetivos del presente trabajo de auditoría forense son:

- Aplicar herramientas que permitan conocer al malware.
- Establecer cuál es el funcionamiento del malware dentro del sistema operativo.
- Establecer controles que eviten que el equipo sea infectado posteriormente por código malicioso.

4.5 Alcance

4.5.1 Alcance del Análisis de Brj_Software

El presente trabajo delimita su alcance exclusivamente al análisis de la información contenida dentro del CD-R entregado a los investigadores por parte de la empresa.

Por la magnitud del problema suscitado, la solución a presentar no será aplicable a todos los casos. Así mismo la metodología, herramientas y técnicas se ajustan dependiendo de la evidencia.

Como el siniestro se ocasionó, desde el departamento de desarrolladores este será el entorno en el cual se trabajará el análisis.

4.5.2 Alcance del Análisis de Innocent_ssh_client

El alcance de nuestro análisis dependerá, del comportamiento del malware dentro de una máquina anfitrión, en la que el código será ingresado para el análisis. De esta manera poder registrar cuales son los archivos de programa que se ven afectados por este código malicioso.

Para determinar cuál es el procedimiento que se debe realizar, para proteger nuestro equipo de este tipo de código malicioso.

4.6 Descripción del entorno informático

El ambiente informático que se manejó para realizar el análisis de la evidencia fue en una máquina virtual que cuenta con la distribución de Caine basada en Ubuntu, en la que se analizará la evidencia obtenida del caso Brj_Software; y una máquina Windows XP donde realizaremos pruebas del comportamiento del malware, instalando ciertas herramientas útiles para el caso Innocent_ssh_client.

4.7 Definición y reconocimiento del problema

4.7.1 Brj_Software

El 8 de septiembre del 2003, el usuario conocido como Richard informa al encargado de IT, que descubrió que otra persona acceda a su cuenta. La persona de IT, se cerciora de lo informado por Richard y efectivamente se da cuenta que alguien ha accedido a la cuenta.

Richard, descubre que esa otra persona accede a su cuenta al emitir el comando “w”. La persona de IT, manifiesta que la dirección IP que Richard informó es, 102.60.21.3 la misma que pertenece a una máquina Linux que los desarrolladores de la empresa la utilizan ocasionalmente para probar su software.

El encargado de realizar las actualizaciones de los parches de seguridad, se da cuenta de que no ha realizado las actualizaciones a la máquina involucrada.
Inmediatamente decide escribir todo el tráfico de red que se ha capturado desde hace un par de meses en un CD-R.

Después de eso, se ejecuta un proceso de respuesta en directo en la máquina de la víctima para recoger los datos volátiles. La dirección IP del respondedor es 102.60.21.149. Debido a que la máquina no se utiliza mucho, usted es capaz de poner fuera de línea y realizar una duplicación forense para preservar las pruebas que pueden contener archivos borrados.

4.7.2 Innocent_ssh_client

A los investigadores se les entrega un CD-R con un código malicioso, el mismo que esta comprimido. Dada las circunstancias, los investigadores establecen pasos para definir bien el problema en el cual identificaremos, el procedimiento y herramientas a aplicar:

- **Procedimiento:** Identificar como se introdujo el malware, cuáles fueron las áreas afectados por la intrusión del malware.
- **Herramienta:** Anubi, Virustotal, Virscan y Metascan (Ver anexo 2),

4.8 Preservando Evidencia

4.8.1 Preservar la evidencia de Brj_Software

La evidencia debe siempre estar salvaguardada en un lugar seguro para mantener la integridad de la misma, y así poder formar parte en una auditoría forense.

Siendo este su principio fundamental el investigador debe realizar su análisis y pruebas a base de una copia exacta de la evidencia original. Con esto se está legalizando la integridad de la cadena de custodia.

Para obtener una copia exacta de la evidencia, realizaremos los siguientes pasos:

1. Tener una computadora para el análisis forense, la misma que por prudencia no tendrá conexión a internet.
2. Tener instalado un sistema operativo para el análisis, en nuestro caso la máquina cuenta, como único sistema operativo la distribución de Ubuntu caine versión 2.0

3. Colocaremos el CD-R que contiene la evidencia original.
4. Abrimos una terminal en la cual ingresaremos como root.
5. Por línea de comando, crearemos el directorio donde guardaremos la copia de evidencia que realizaremos, para posteriormente analizarla.
6. Montaremos el CD-R que contiene la evidencia, ya que la distribución no reconoce de manera automática a los dispositivos externos, sean estos CD-R, pendrive, disco duros.
7. Para realizar una copia idéntica a la original, usaremos una herramienta de caine llamada Air, la que nos permite realizar este proceso.
8. Verificamos que la copia sea igual a la original por medio de los hash.
9. Desmontaremos el CD-R con la evidencia original, ya que no la requerimos, y la guardemos en un lugar seguro y apropiado para salvaguardar su integridad.
10. Colocamos la copia de la evidencia, en el directorio creado para el caso.

4.8.2 Preservar la evidencia de Innocent_ssh_client

Este es un punto crucial para que el resultado del análisis no pueda ser cuestionado, ya que se tomó en cuenta todos los parámetros para realizar una auditoría forense, manteniendo la integridad de la evidencia.

Por dicha razón las pruebas se las realiza las veces que el investigador las considere necesarias, pero basándose en un copia de la evidencia sin manipular la evidencia original.

Para lograr este objetivo procederemos a realizar los siguientes pasos:

1. Tener una máquina virtual con el sistema operativo que nos permita manipular el malware, en nuestro caso será una Windows XP.
2. Utilizaremos otra máquina virtual que tenga como distribución Caine, para realizar la copia de la evidencia.
3. Colocaremos el CD-R que contiene la evidencia original.
4. Verificamos que la copia sea igual a la original por medio de los hash.

5. Una vez ya lista la copia de la evidencia, se procederá a ubicar la evidencia original en un lugar seguro.
6. En la maquina Windows XP, instalaremos las herramientas para el análisis del malware.
7. Se realizará una comparación entre los reportes obtenidos de los análisis realizados al malware.
8. Identificaremos las áreas afectadas por el siniestro.
9. Realizaremos un informe indicando, el procedimiento para evitar ser víctima de ese tipo de amenaza.

4.9 Extracción de la información

4.9.1 Extracción de la información de Brj_Software

Puesto que ahora contamos con una copia de la evidencia, procederemos a extraer la información, contenida en la imagen forense.

- Utilizando la terminal ingresamos al directorio que contiene la evidencia, con el comando ls listamos el contenido.
- Observamos que la información de la evidencia, esta comprimida así que procederemos a descomprimirlo utilizando el comando gzip –d.

- Nuestra evidencia se divide en tres carpetas:
    - Forensic_duplication, que contiene la imagen que realizo en personal de IT.
    - Network_activity.
    - Live_response.

4.9.1.1 Forensic_duplication

Forensic Duplication contiene la imagen del equipo (es una copia exacta sin manipulaciones del equipo involucrado), la cual fue realizada por los de IT de la empresa para preservar las pruebas que pueden contener archivos valiosos para averiguar e indagar lo ocurrido y que sirva como evidencia.

Infortunadamente los de IT no realizaron la copia de la imagen correctamente, ya que la misma está corrupta, por ello no se pudo analizar la imagen e investigar lo ocurrido por medio de la imagen. Así que nuestro análisis se basará en la evidencia que se encuentra en Network_activity y Live_response.

4.9.1.2 Network_activity

4.9.1.2.1 Portscan.log

Portscan es una herramienta muy útil y básica para el escaneo de puertos, observando el resultado nos damos cuenta que hay una dirección IP la cual está fuera de la red de la empresa; la dirección IP 94.90.84.93 realiza un escaneo de puertos a un grupo de puertos de un servidor de la empresa (102.60.21.3)

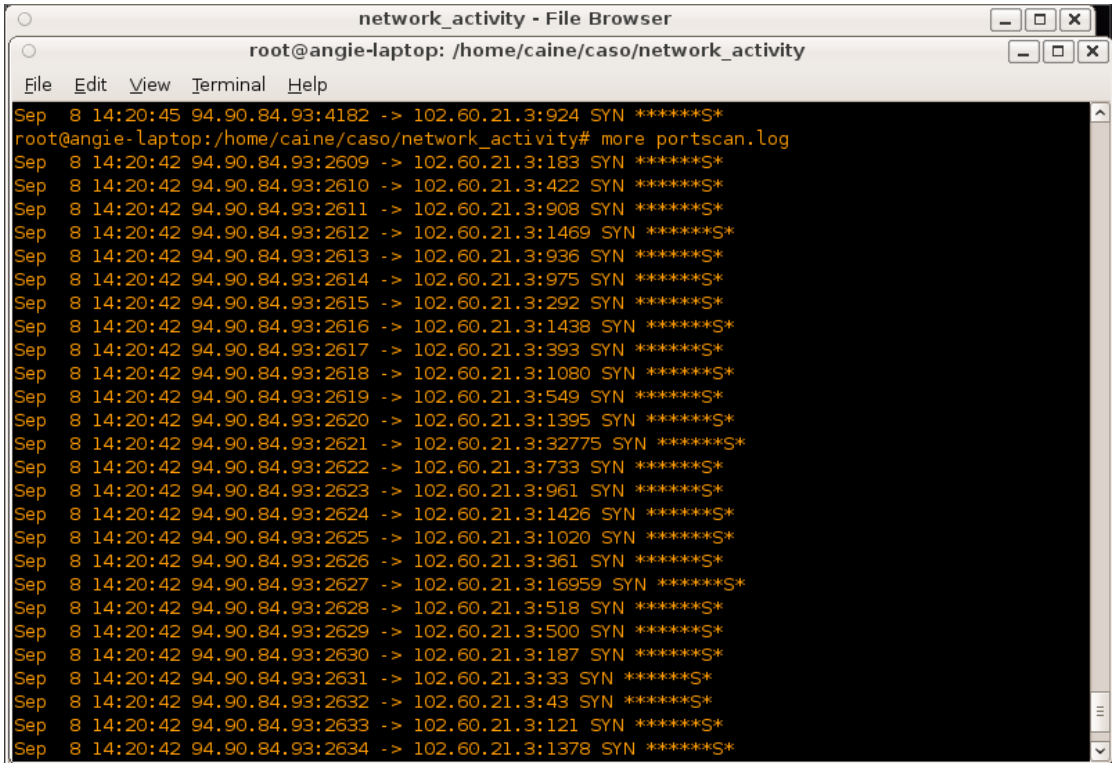


Figura 4.21: Portscan.log
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

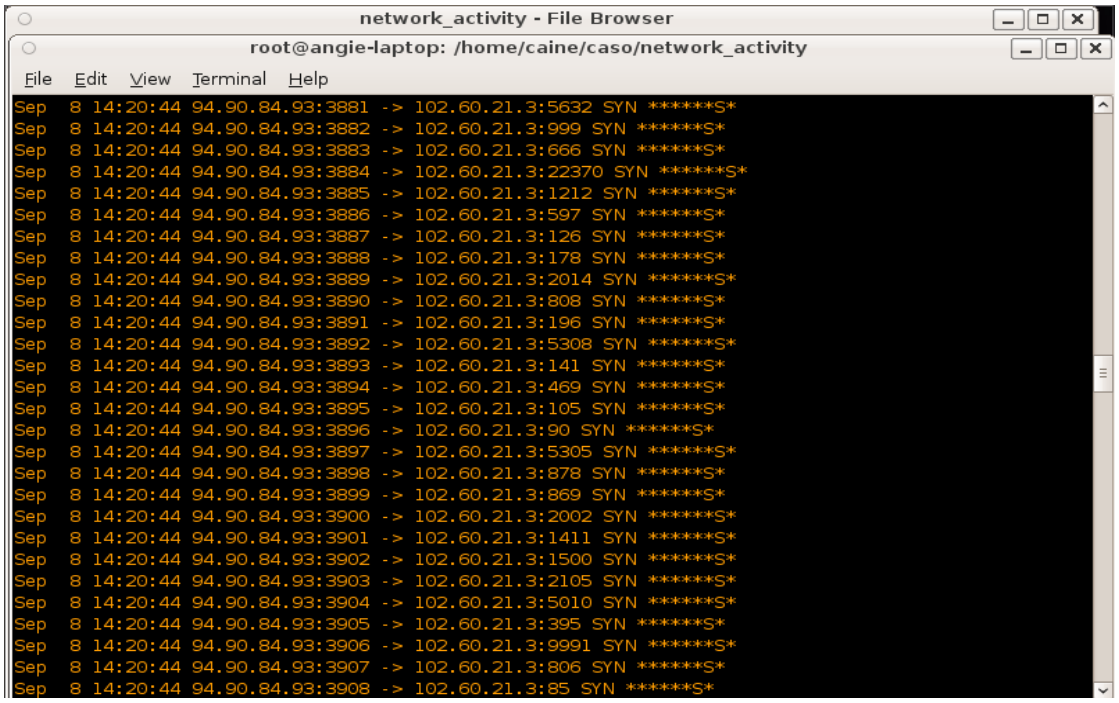


Figura 4.22: Portscan.log_1
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

4.9.1.2.2 s3.tcpdstat.txt

Tcpdstat esta herramienta realiza una recopilación de información que la cual nos mostrará aún más el alcance de los investigadores forenses digitales.

Tabla 13: comando Tcpdstat
Fuente: Datos obtenidos del análisis del s3.tcptat.txt, Elaborado: Las autoras

En este archivo podemos visualizar datos concretos como lo son:
Hora de inicio y hora de finalización de captura
Tiempo en segundos
Tamaño de la captura total (14.73 Mb)
Protocolos

Cabe recalcar que la mayoría de los datos han viajado a través de TCP/IP. Los protocolos más utilizados fueron FTP, Telnet y SSH. Estos protocolos pueden ser usados por algún atacante ya que por ellos se puede transferir información y en algunos casos no son cifradas como lo son por FTP y Telnet.

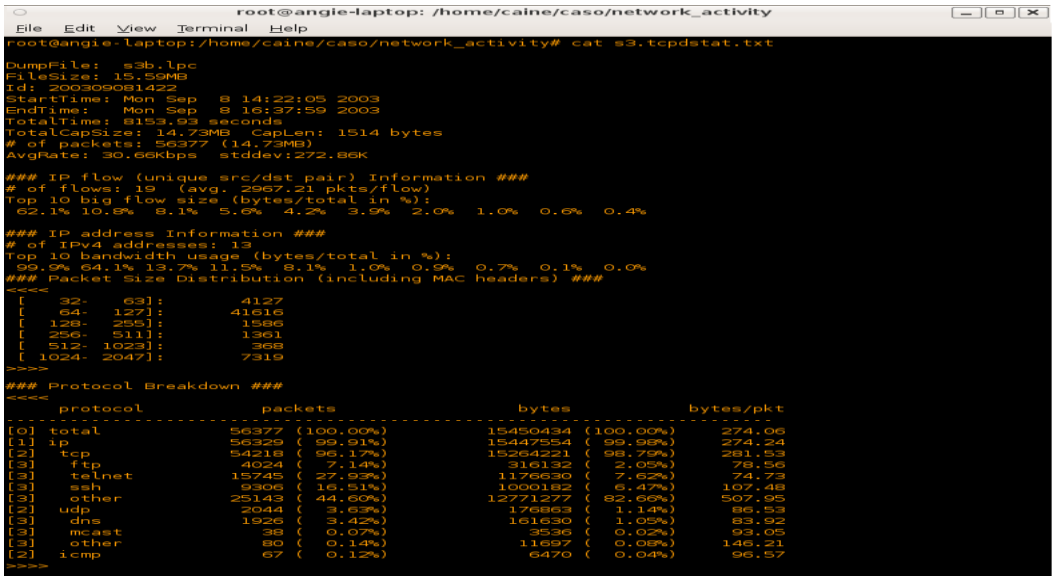


Figura 4.23: s3.tcpstat.txt
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

4.9.1.2.3 alert.s3

Este archivo nos damos cuenta que el intruso intenta obtener acceso al EXPLOIT de un demonio de impresión (LPR). El Exploit se lo utiliza para hallar una vulnerabilidad de seguridad en este caso para conseguir acceso de forma no autorizada a una cuenta con privilegios.

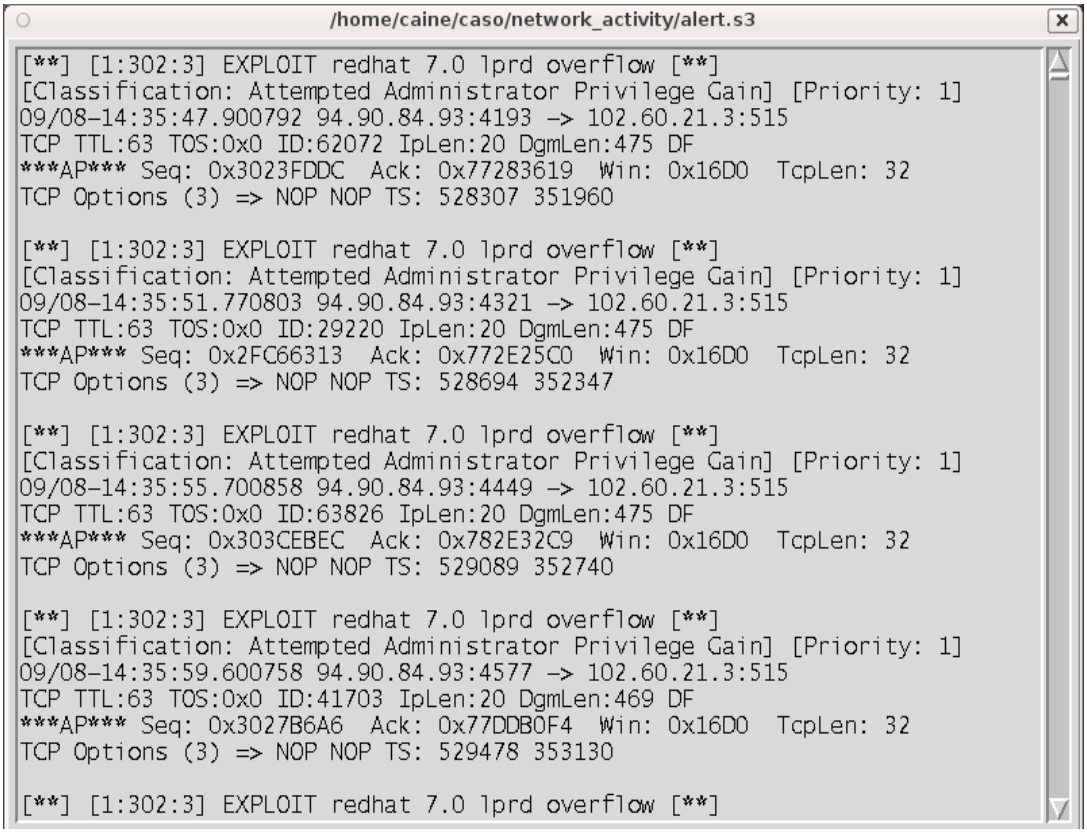


Figura 4.24: Exploit
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

4.9.1.2.4 s3.tcptrace.txt

Tcptrace esta herramienta nos ayudará a examinar los datos registrados a partir de la sesión de telnet.

Este archivo nos ayudará a ver las sesiones que han ocurrido y así poder sacar algunas conclusiones.

Nos podemos dar cuenta que desde la sesión 3 se estable conexiones a la dirección 94.90.84.93 (no corresponde a la red de BRJ Software) hacia la dirección 102.60.21.3 (corresponde a la red de BRJ Software).

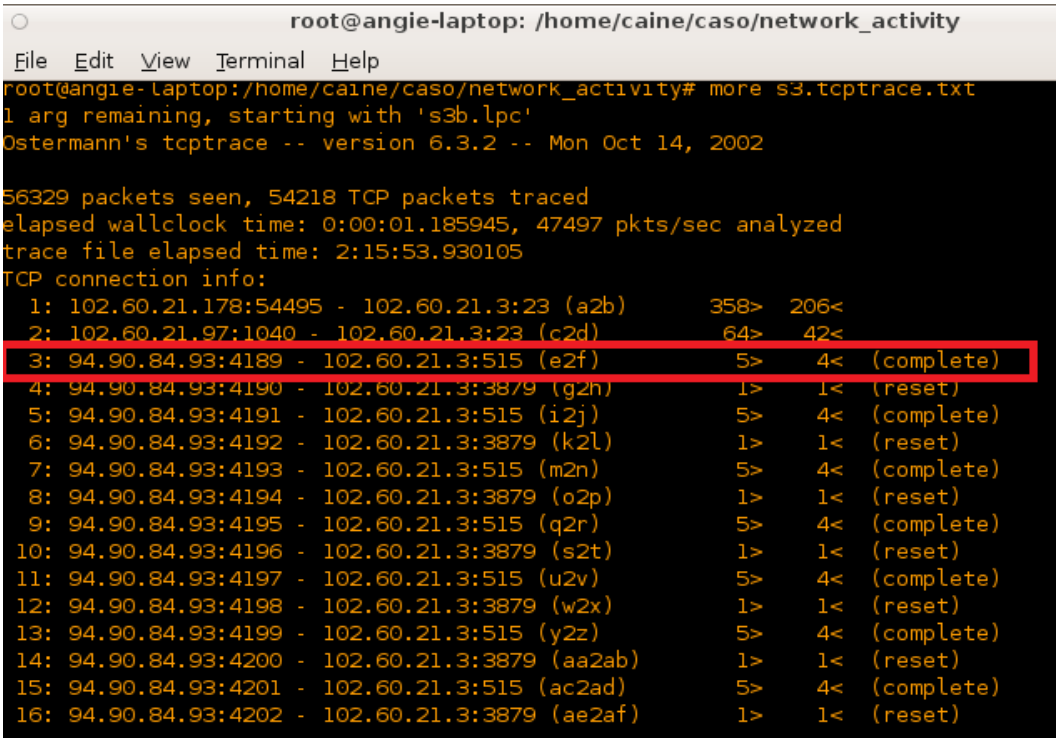


Figura 4.25: s3.tcptrace.txt
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

En la sesión 1880 el intruso accede al sistema, una vez que ingresa el intruso comienza a establecer conexiones a un sistema remoto con una nueva dirección IP 94.178.4.82 con el puerto 21 (servidor FTP, la cual sirve para la transferencia de archivos entre sistemas conectados a una red).

Esto no debería de ocurrir porque nos indica que puede existir un ataque de fuerza bruta.

Ahora nos damos cuenta que el intruso es capaz de ingresar por telnet, acceder al root y poner al FTP fuera del sistema. Además accede al servidor a través del puerto 514 (este puerto accede al Shell), inmediatamente después de este suceso notamos que existen otras conexiones. Una de estas conexiones nos demuestra que el intruso accedió por SSH al servidor de BRJ, ocasionando un gran problema porque el servicio de SSH envía la información cifrada por lo tanto no podemos saber que exactamente está haciendo en la red.

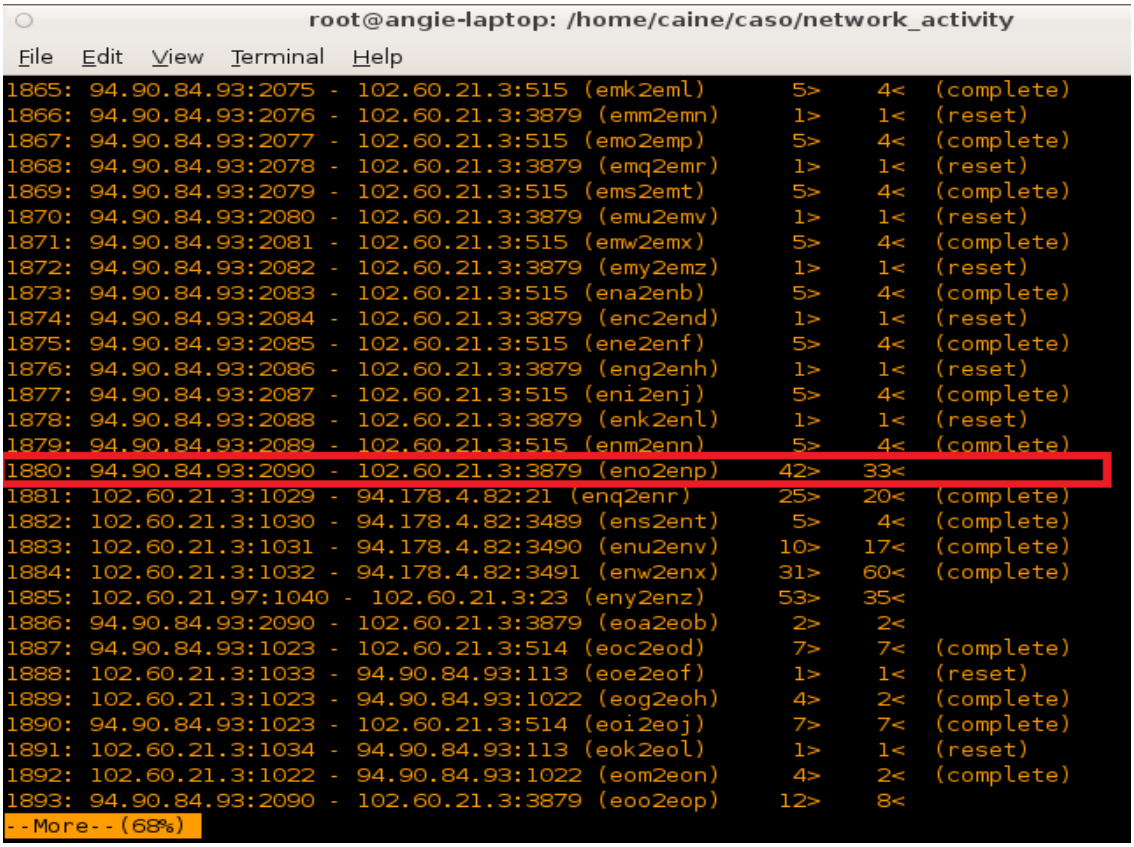


Figura 4.26: s3.tcptrace.txt_1
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

Sin embargo, lo curioso es que el intruso, comienza a utilizar el servidor BRJ como una manera de atacar a otro servidor como está ocurriendo desde la sesión 1916 con dirección IP 94.200.10.71.

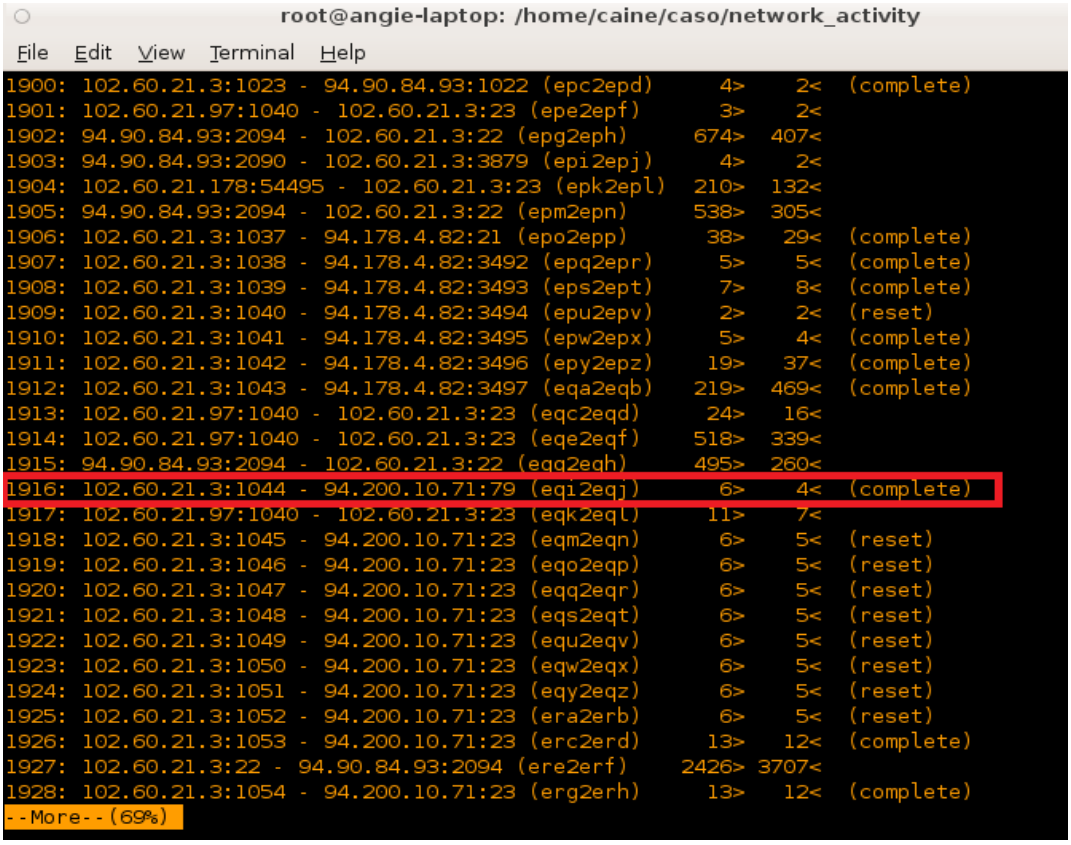


Figura 4.27: s3.tcptrace.txt_2
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

4.9.1.2.5 diff

Este log muestra las diferencias de lo que hizo que el ataque de 2089 en la sesión demonio LPR tuviera éxito y qué no tuviera éxito en la sesión de 2087.

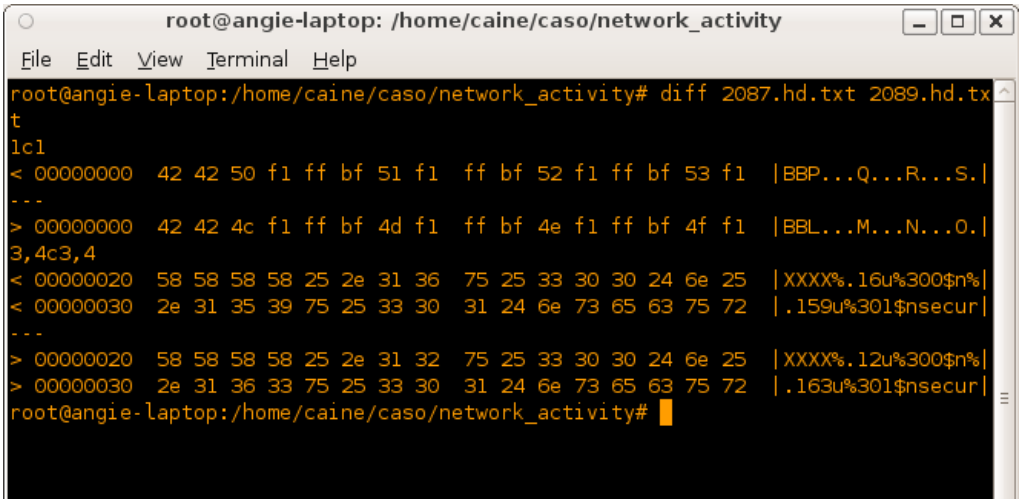


Figura 4.28: diff 2087.hd.txt 2089.hd.txt
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

4.9.1.2.6 094.090.084.093.02090-102.060.021.003.03879

En este archivo observamos que hay comandos que se ingresan en el sistema para averiguar información del mismo, también se crea un directorio oculto / tmp / .kde FTP a 94.178.4.82 y descarga los archivos binarios que comienzan con "knark".

Al mismo tiempo crea un usuario llamado "LPD" dentro del sistema con la misma UID que permite tener acceso como Root y RSH, rápidamente realizó un ataque de fuerza bruta al FTP para ocultar.

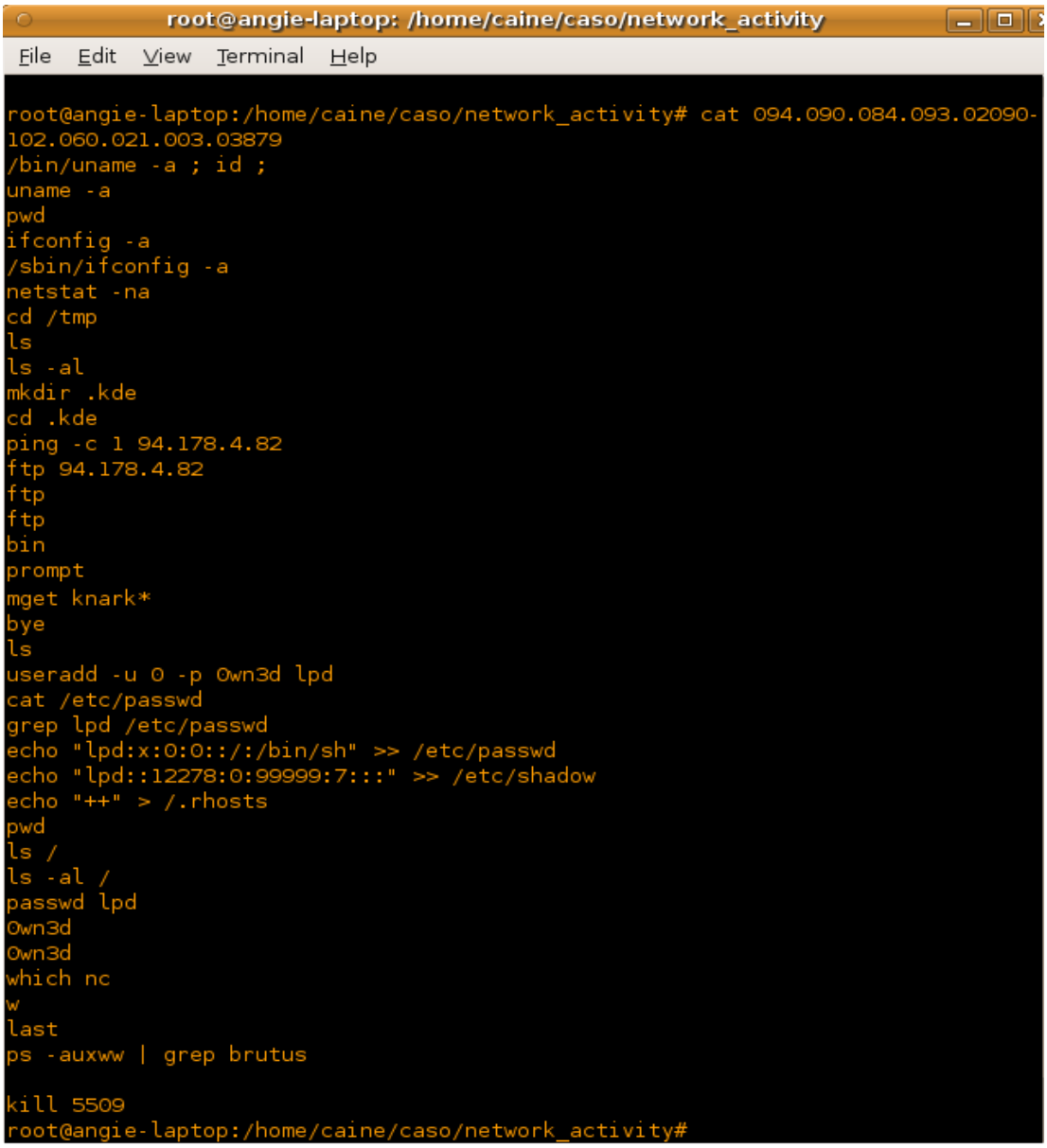


Figura 4.29: cat 094.090.084.093.020990-102.060.021.003.03879
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

4.9.1.2.7 102.060.021.003.03879-094.090.084.093.02090

En este archivo podemos observar lo que el servidor de BRJ SOFTWARE respondió sobre todas las acciones que fueron realizadas por comandos y las conexiones activas en los equipos a través de telnet.

Estas acciones son importantes, porque nos permite encontrar varios indicios sobre el siniestro ocurrido en BRJ SOFTWARE.

Se puede observar que por varios puertos muestra conexión las cuales algunas están muy sospechosas. Cabe recalcar que existen un sin número de eventos en los puertos los cuales se procederá a explicar de forma breve en la siguiente tabla.

Tabla 14: Conexiones activas de TCP
Fuente: Datos obtenidos del análisis, Elaborado: Las autoras

PROTOCOLO	DIRECCIÓN LOCAL	DIRECCIÓN EXTRANJERA	ESTADOS	DESCRIPCIÓN
Tcp	102.30.21.3:1820	127.0.0.1:23	ESTABLISHED	
Tcp	102.60.21.3:2323	94.178.4.82:3502	ESTABLISHED	Este puerto se encuentra en estado listen y esperando conexiones
Tcp	102.60.21.3:2323	0.0.0.0:*	LISTEN	
Tcp	102.60.21.3:22	94.90.84.93:2094	ESTABLISHED	Muestra conexión desde esta IP sospechosa hacia el Puerto ssh
Tcp	102.6021.3:3879	94.90.84.93:2090	ESTABLISHED	Establece conexión al puerto 3879 y este si se le da un mal uso sirve para el ataque de negación de servicio
Tcp	0.0.0.0:3879	0.0.0.0:*	LISTEN	
Tcp	102.60.21.3:515	94.90.84.93:1761	CLOSE_WAIT	El puerto 515 de tcp es un servicio de impresión, el cual envía datos en crudo a un puerto tcp remoto

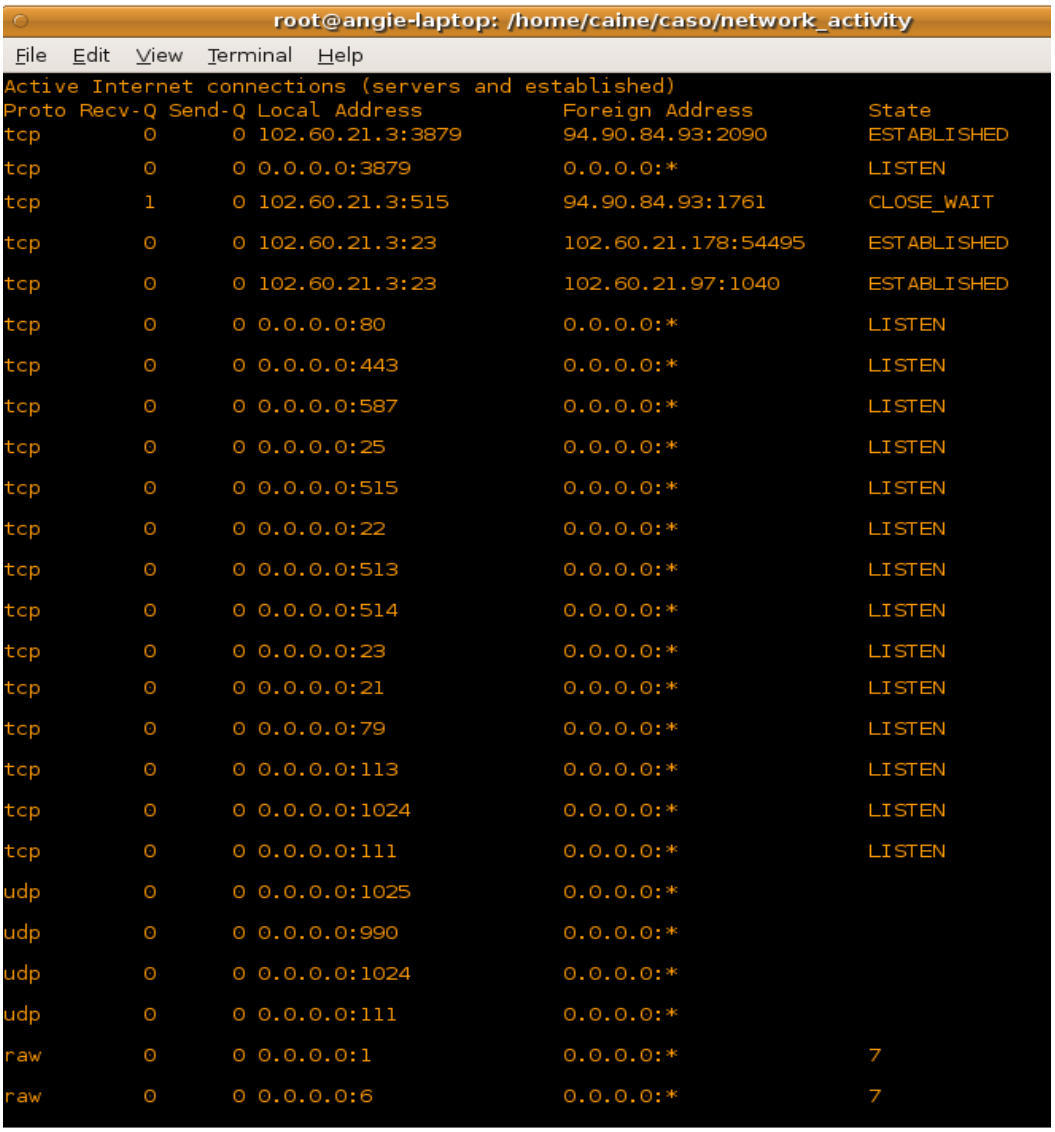


Figura 4.30: Conexiones activas en 102.060.021.003.03879-094.090.084.093.02090
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

4.9.1.2.8 102.060.021.003.02323-094.178.004.082.03515

En este archivo se puede observar que se establece una sesión en este caso con el usuario Richard y ejecuta / usr / sbin / rootlpd / bin / bash, la cual envía un mensaje diciendo “¿Te sientes afortunado hoy, hax0r?”. Una vez dentro envía información a un directorio oculto, anteriormente llamado .kde, intenta realizar una conexión ftp a la dirección ip 94.20.1.9, luego de varios intentos fallidos el intruso logra conectarse a ese servidor a través de ftp.

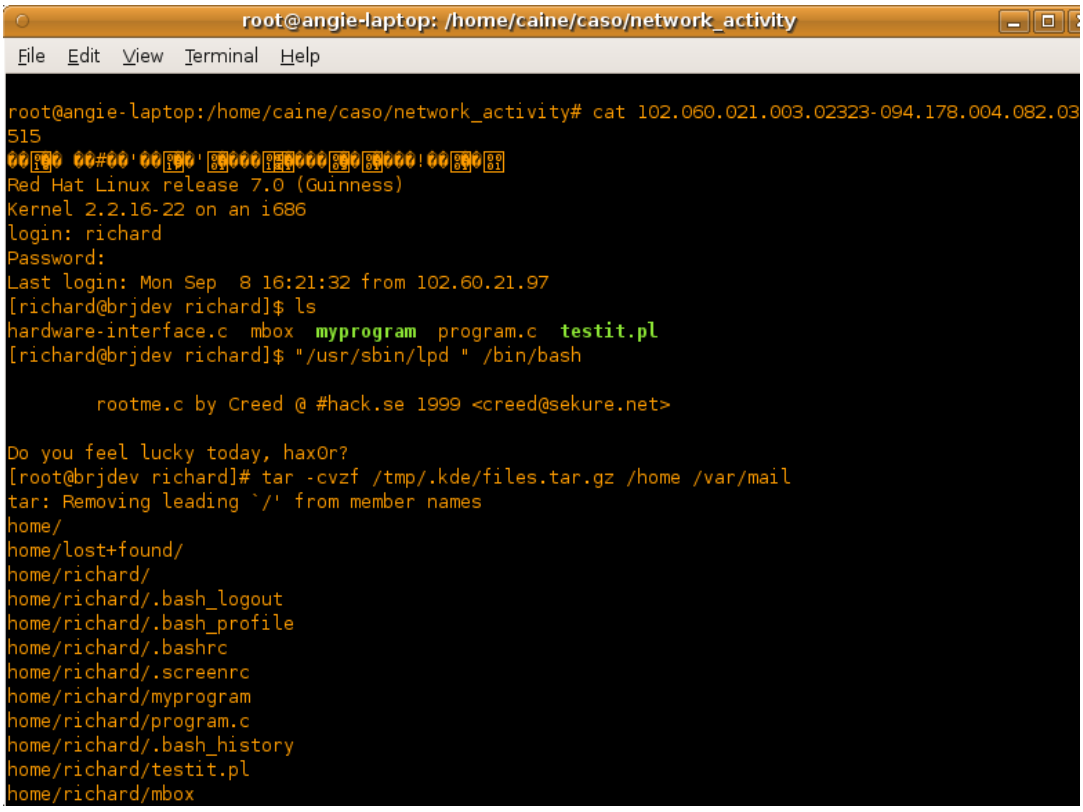


Figura 4.31: cat 102.060.021.003.02323-094.178.004.082.03515
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

4.9.1.2.9 094.178.004.082.03502 – 102.060.021.003.01819

En este archivo encontramos el Datapipe que es una aplicación que realiza acciones de búsqueda y reemplazo dentro de bases de datos específicas. En este archivo se visualiza información y la programación de dicha aplicación.

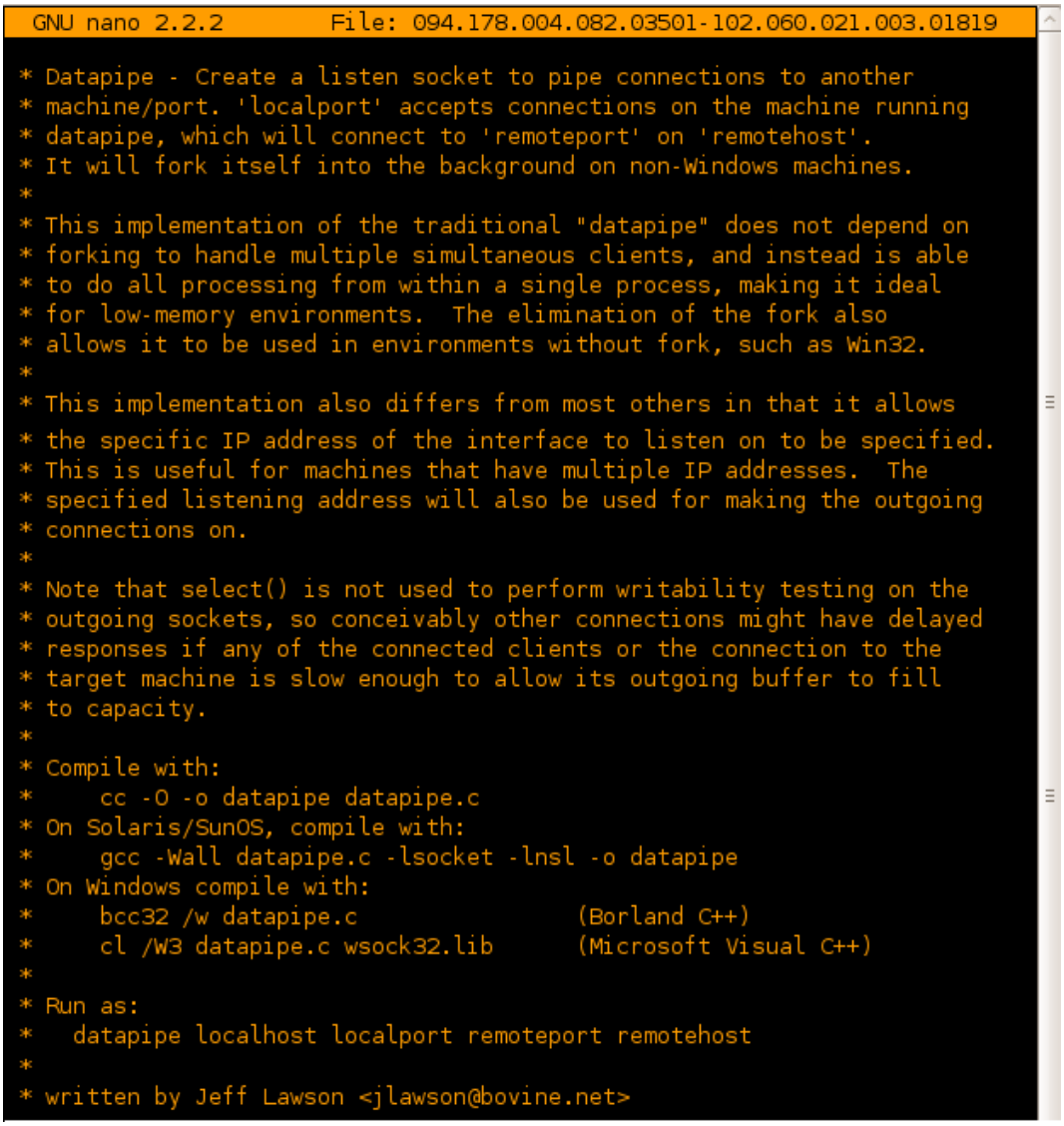
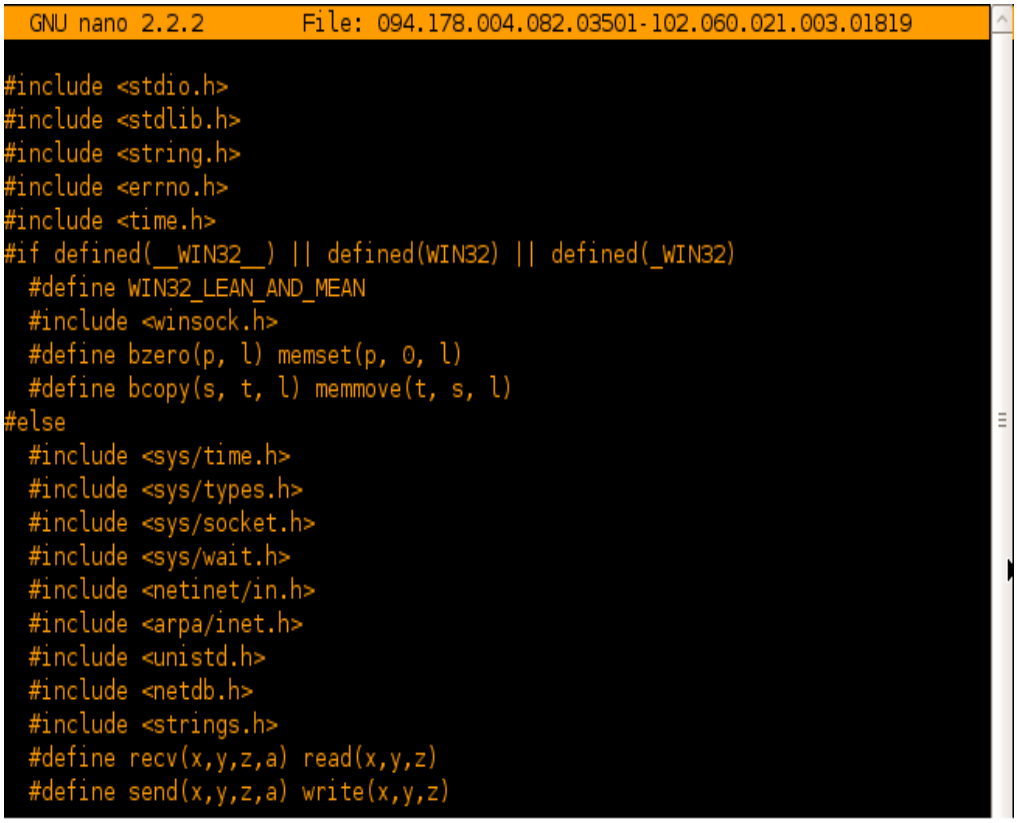


Figura 4.32: 094.178.004.082.03502 – 102.060.021.003.01819
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras



```
 GNU nano 2.2.2        File: 094.178.004.082.03501-102.060.021.003.01819

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <time.h>
#if defined(__WIN32__) || defined(WIN32) || defined(_WIN32)
  #define WIN32_LEAN_AND_MEAN
  #include <winsock.h>
  #define bzero(p, l) memset(p, 0, l)
  #define bcopy(s, t, l) memmove(t, s, l)
#else
  #include <sys/time.h>
  #include <sys/types.h>
  #include <sys/socket.h>
  #include <sys/wait.h>
  #include <netinet/in.h>
  #include <arpa/inet.h>
  #include <unistd.h>
  #include <netdb.h>
  #include <strings.h>
  #define recv(x,y,z,a) read(x,y,z)
  #define send(x,y,z,a) write(x,y,z)
```

Figura 4.33: 094.178.004.082.03501 – 102.060.021.003.01819
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

Tabla 15: Tabla de comandos y procesos
Fuente: Datos obtenidos del análisis del archivo citado, Elaborado: Las autoras

COMANDOS Y PROCESOS EJECUTADOS
w
Id
whoami
Netstat –na | less
Ps –auxww |grep datapipe
Ls –al
Kill -31 5688
Ls –al
Ps –auxww | grep datapipe 5683

4.9.1.2.10 102.060.021.003.02323 – 094.178.082.03502

En este archivo encontramos varios procesos que se ejecutaron la cual nos muestra información valiosa para la resolución de este caso.

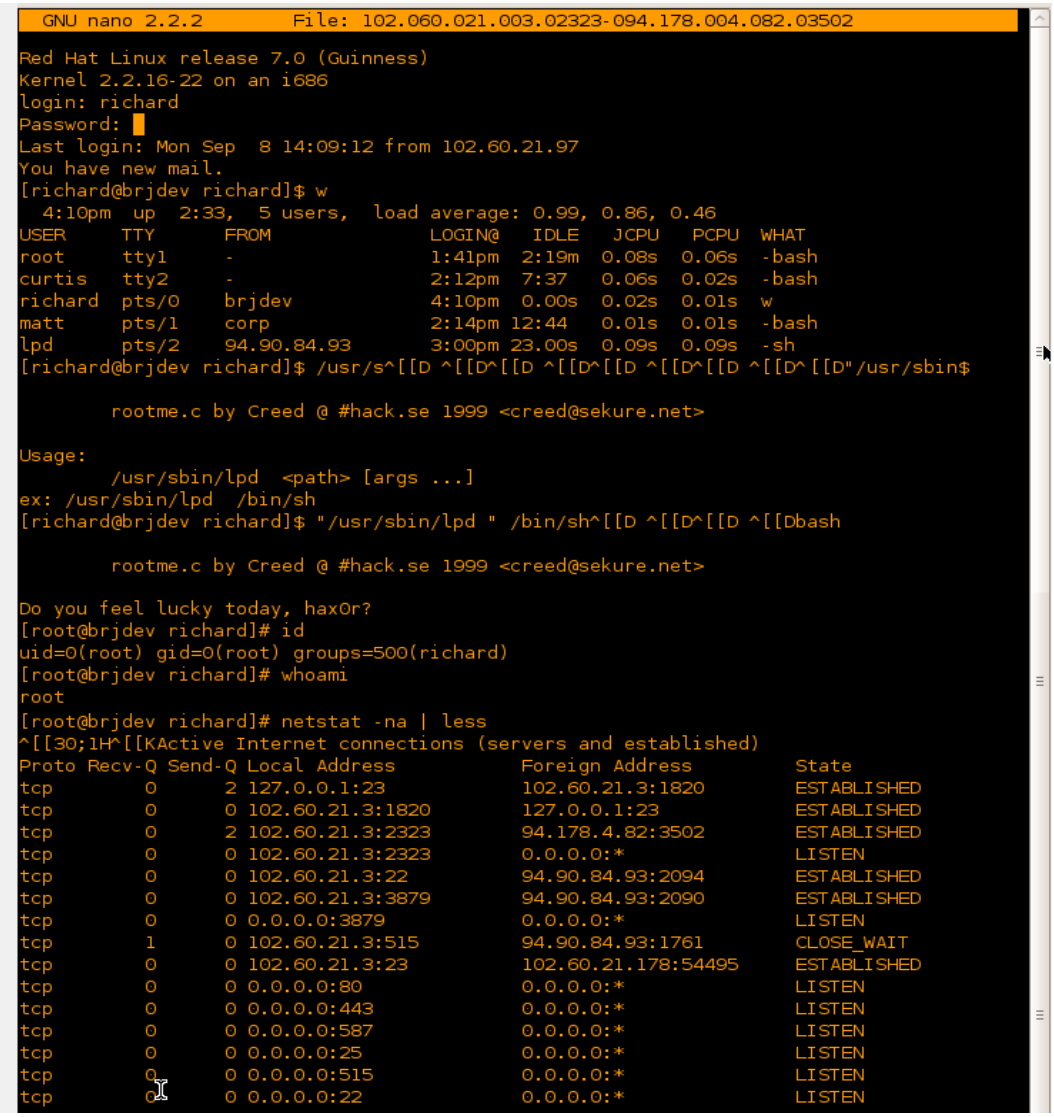


Figura 4.34: 102.060.021.003.02323 – 094.178.082.03502
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

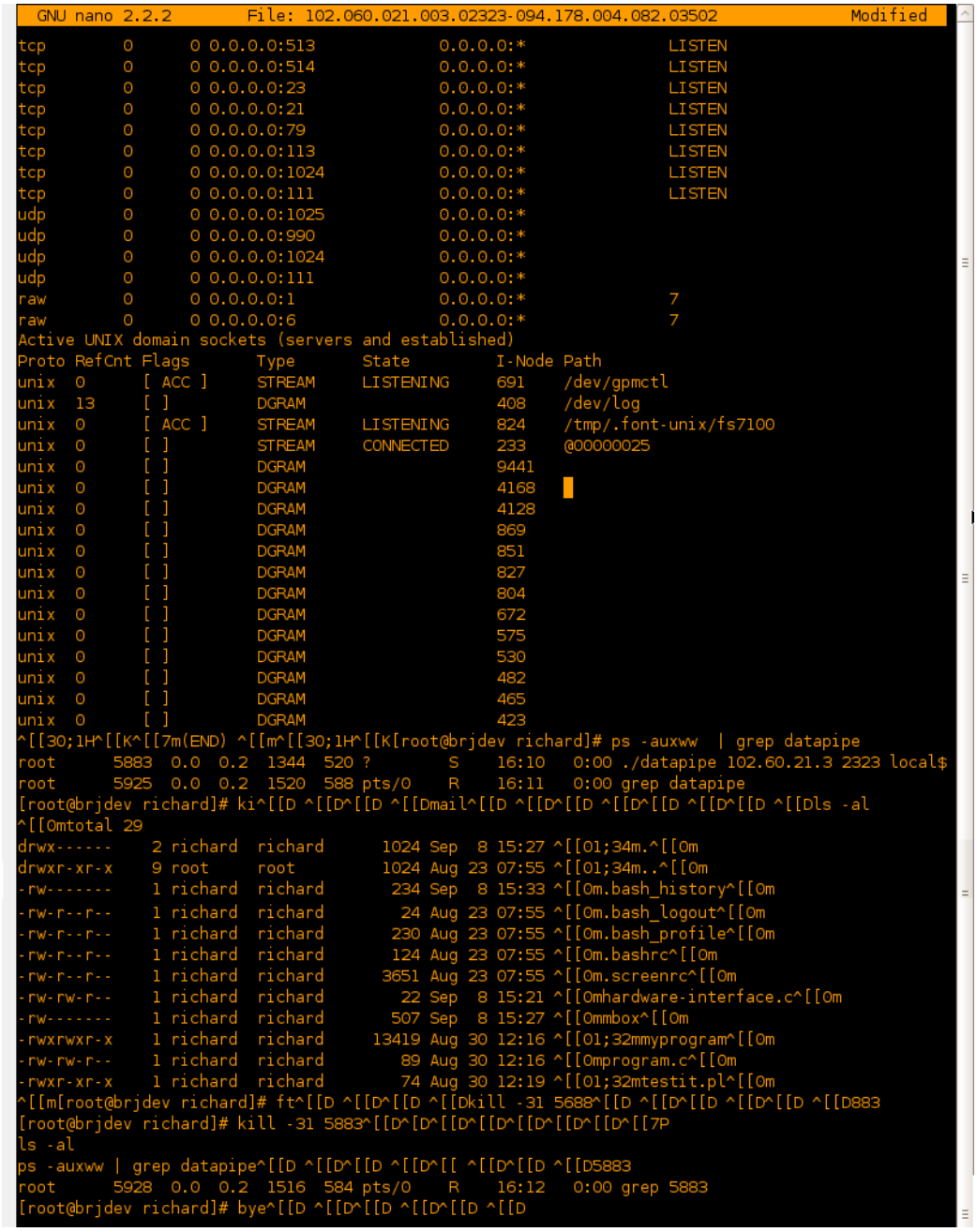


Figura 4.35: 102.060.021.003.02323 – 094.178.082.03502_1
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

4.9.1.2.11 Interpretación de los archivos de FTP

En los logs de FTP nos encontramos con dos casos con el servidor FTP a continuación explicaremos ciertos hallazgos:

- **CASO 1**

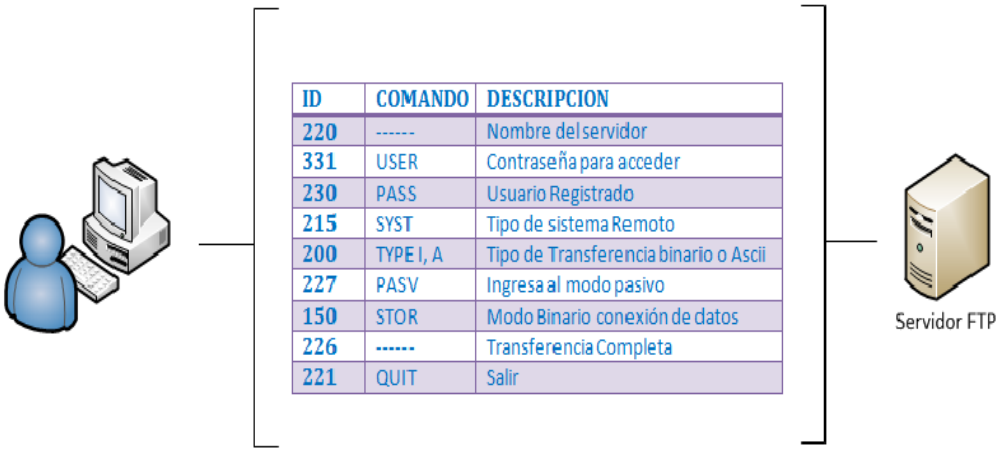
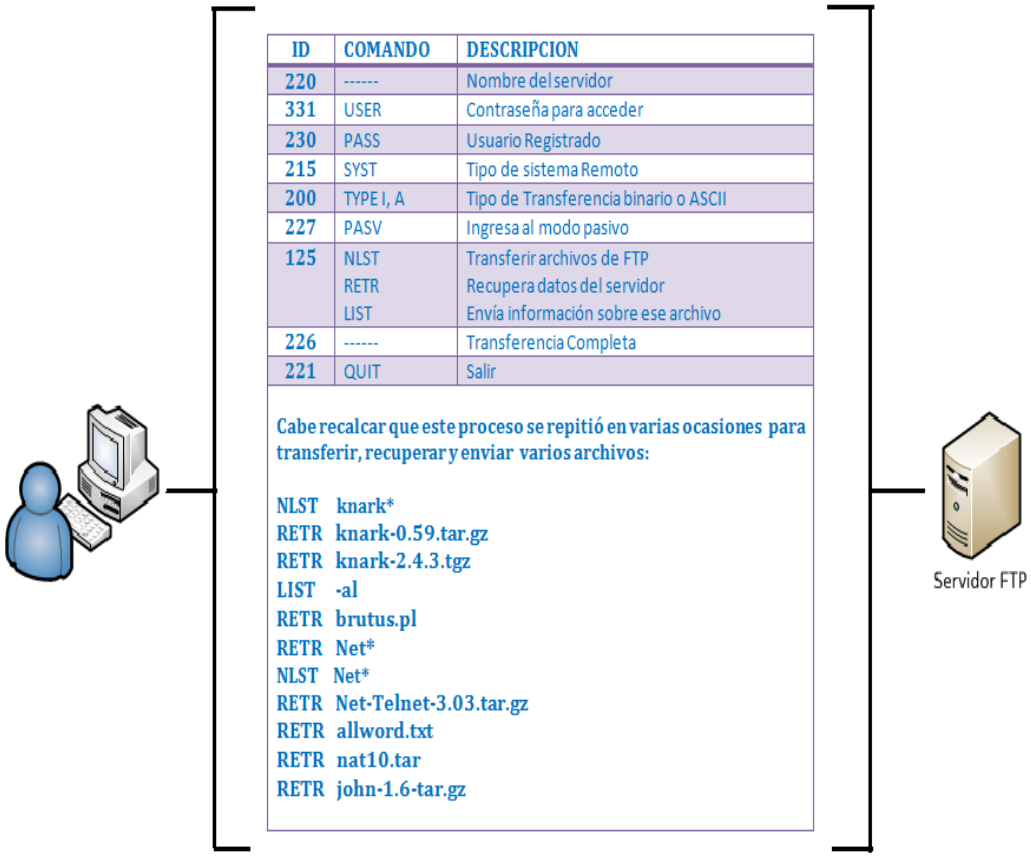


Figura 4.36: Análisis de logs de FTP
Fuente: Datos obtenidos del análisis de los logs FTP, Elaborado: Las autoras

Lo que podemos indicar es que una vez analizado el log podemos decir que se establece una conexión con el servidor FTP de modo Pasivo a un sistema Unix para poder transferir datos en forma binaria.

- **CASO 2**



ID	COMANDO	DESCRIPCION
220	------	Nombre del servidor
331	USER	Contraseña para acceder
230	PASS	Usuario Registrado
215	SYST	Tipo de sistema Remoto
200	TYPE I, A	Tipo de Transferencia binario o ASCII
227	PASV	Ingresa al modo pasivo
125	NLST RETR LIST	Transferir archivos de FTP Recupera datos del servidor Envía información sobre ese archivo
226	------	Transferencia Completa
221	QUIT	Salir

Cabe recalcar que este proceso se repitió en varias ocasiones para transferir, recuperar y enviar varios archivos:

NLST knark*
RETR knark-0.59.tar.gz
RETR knark-2.4.3.tgz
LIST -al
RETR brutus.pl
RETR Net*
NLST Net*
RETR Net-Telnet-3.03.tar.gz
RETR allword.txt
RETR nat10.tar
RETR john-1.6-tar.gz

Figura 4.37: Análisis de logs de FTP_1
Fuente: Datos obtenidos del análisis de los logs FTP, Elaborado: Las autoras

Una vez analizados los logs podemos decir que se establecen conexiones con el servidor FTP de modo Pasivo a un sistema Windows NT para poder transferir, recuperar y enviar datos en forma binaria y ASCII. (Ver anexo 1)

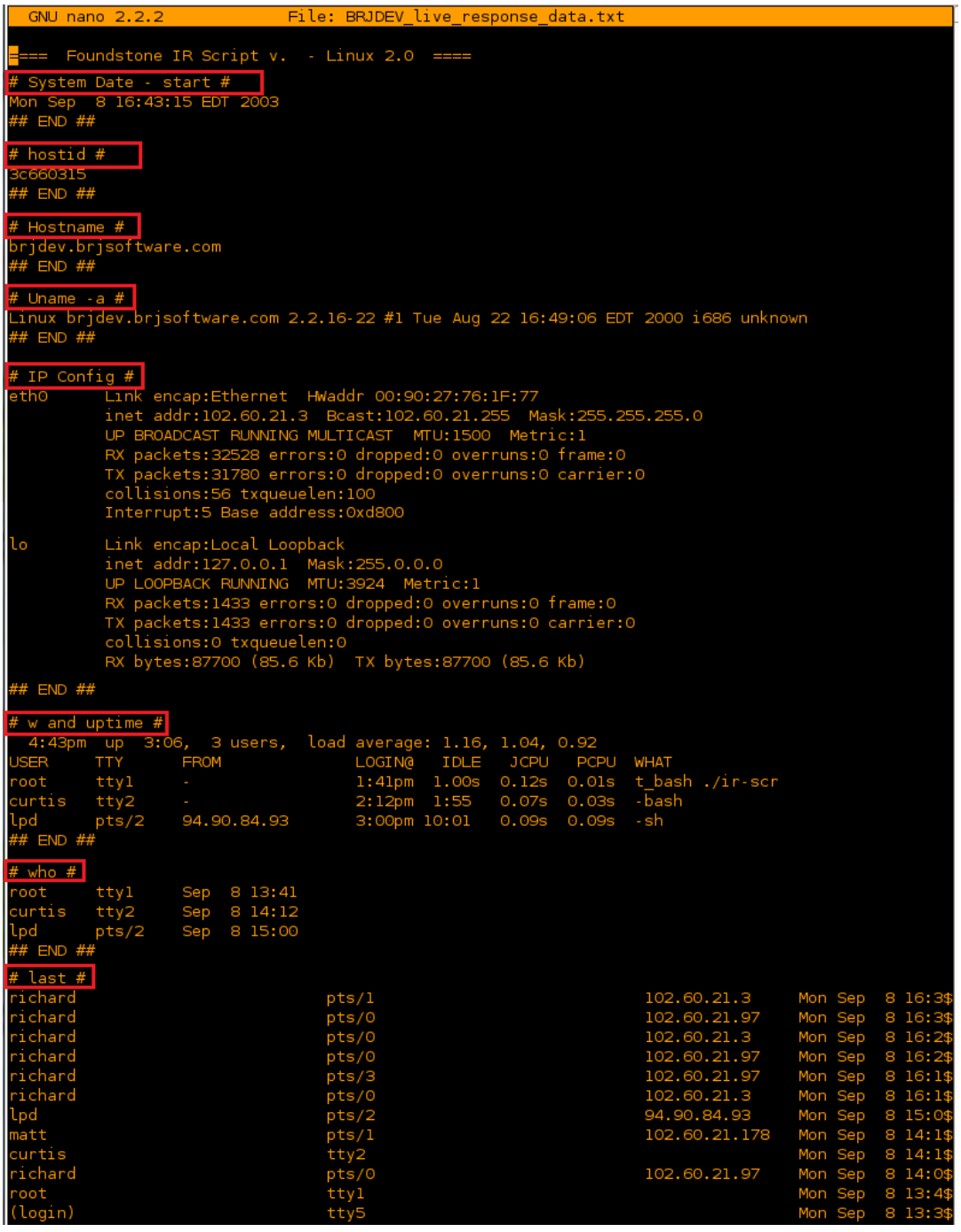
4.9.1.3 Live_response

4.9.1.3.1 BRJDEV_live_response_data.txt

En este archivo encontramos varios procesos que se ejecutaron la cual nos especifica considerable información de utilidad para la resolución de este caso.

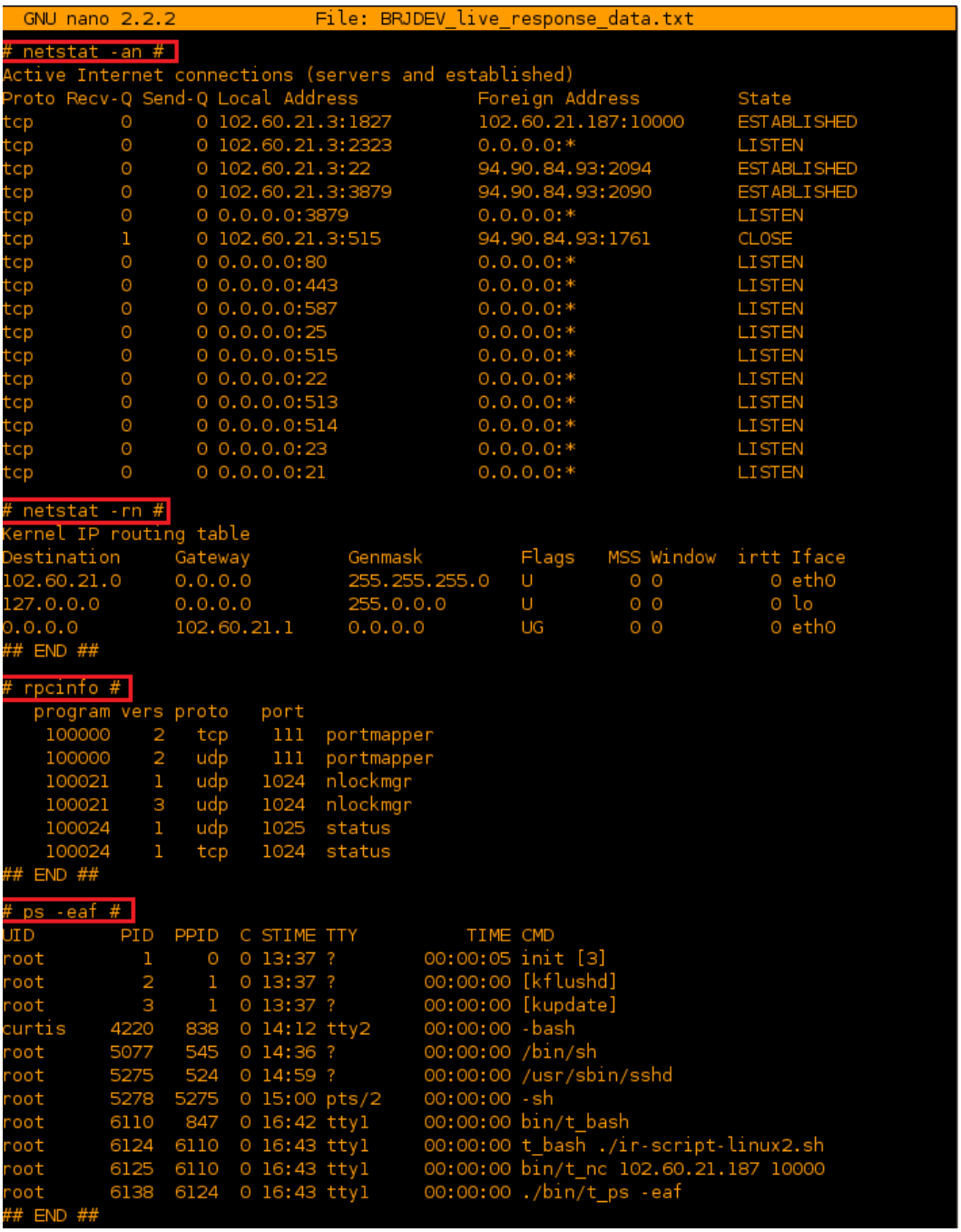
Tabla 16: Comandos ejecutados en BRJDEV_live _response_data
Fuente: Datos obtenidos del análisis de BRJDEV_live_response_data.txt, Elaborado: Las autoras

COMANDOS Y PROCESOS EJECUTADOS	
System Date – start	File Listings
Hostid	MD5 Sums
Hostname	Lsof_4.63
Uname –a	/etc/passwd
IP config	/etc/group
W and uptime	/etc/inetd.conf
Who	RPM
Last	Lsmod
Netstat –an	Mount
Netstat –rn	df –k
Rpcinfo	System Date – end
Ps –eaf	



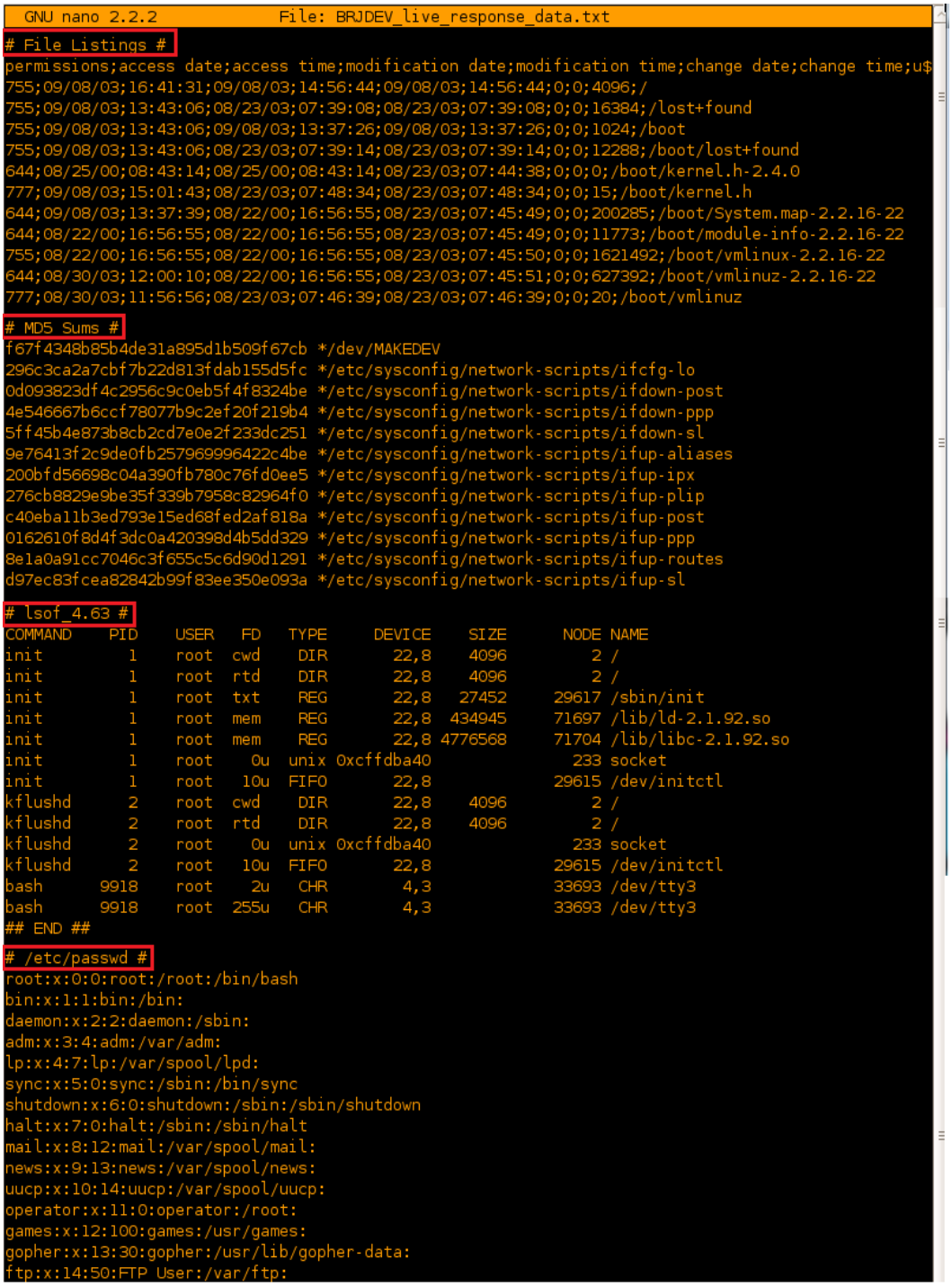
```
  GNU nano 2.2.2              File: BRJDEV_live_response_data.txt

====  Foundstone IR Script v.  - Linux 2.0  ====
# System Date - start #
Mon Sep  8 16:43:15 EDT 2003
## END ##

# hostid #
3c660315
## END ##

# Hostname #
brjdev.brjsoftware.com
## END ##

# Uname -a #
Linux brjdev.brjsoftware.com 2.2.16-22 #1 Tue Aug 22 16:49:06 EDT 2000 i686 unknown
## END ##

# IP Config #
eth0      Link encap:Ethernet  HWaddr 00:90:27:76:1F:77
          inet addr:102.60.21.3  Bcast:102.60.21.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:32528 errors:0 dropped:0 overruns:0 frame:0
          TX packets:31780 errors:0 dropped:0 overruns:0 carrier:0
          collisions:56 txqueuelen:100
          Interrupt:5 Base address:0xd800

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:3924  Metric:1
          RX packets:1433 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1433 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:87700 (85.6 Kb)  TX bytes:87700 (85.6 Kb)

## END ##

# w and uptime #
  4:43pm  up  3:06,  3 users,  load average: 1.16, 1.04, 0.92
USER     TTY      FROM              LOGIN@   IDLE   JCPU   PCPU  WHAT
root     tty1     -                 1:41pm  1.00s  0.12s  0.01s  t_bash ./ir-scr
curtis   tty2     -                 2:12pm  1:55   0.07s  0.03s  -bash
lpd      pts/2    94.90.84.93       3:00pm 10:01   0.09s  0.09s  -sh
## END ##

# who #
root     tty1     Sep  8 13:41
curtis   tty2     Sep  8 14:12
lpd      pts/2    Sep  8 15:00
## END ##

# last #
richard                  pts/1                                  102.60.21.3     Mon Sep  8 16:3$
richard                  pts/0                                  102.60.21.97    Mon Sep  8 16:3$
richard                  pts/0                                  102.60.21.3     Mon Sep  8 16:2$
richard                  pts/0                                  102.60.21.97    Mon Sep  8 16:2$
richard                  pts/3                                  102.60.21.97    Mon Sep  8 16:1$
richard                  pts/0                                  102.60.21.3     Mon Sep  8 16:1$
lpd                      pts/2                                  94.90.84.93     Mon Sep  8 15:0$
matt                     pts/1                                  102.60.21.178   Mon Sep  8 14:1$
curtis                   tty2                                                   Mon Sep  8 14:1$
richard                  pts/0                                  102.60.21.97    Mon Sep  8 14:0$
root                     tty1                                                   Mon Sep  8 13:4$
(login)                  tty5                                                   Mon Sep  8 13:3$
```

Figura 4.38: Comandos ejecutados en BRJDEV_live _response_data.txt
Fuente: Entorno virtual Caine 2.0, Elaboración: Las autoras



```
  GNU nano 2.2.2            File: BRJDEV_live_response_data.txt

# netstat -an #
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 102.60.21.3:1827        102.60.21.187:10000     ESTABLISHED
tcp        0      0 102.60.21.3:2323        0.0.0.0:*               LISTEN
tcp        0      0 102.60.21.3:22          94.90.84.93:2094        ESTABLISHED
tcp        0      0 102.60.21.3:3879        94.90.84.93:2090        ESTABLISHED
tcp        0      0 0.0.0.0:3879            0.0.0.0:*               LISTEN
tcp        1      0 102.60.21.3:515         94.90.84.93:1761        CLOSE
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:443             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:587             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:25              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:515             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:513             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:514             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:21              0.0.0.0:*               LISTEN

# netstat -rn #
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
102.60.21.0     0.0.0.0         255.255.255.0   U         0 0          0 eth0
127.0.0.0       0.0.0.0         255.0.0.0       U         0 0          0 lo
0.0.0.0         102.60.21.1     0.0.0.0         UG        0 0          0 eth0
## END ##

# rpcinfo #
   program vers proto   port
    100000    2   tcp    111  portmapper
    100000    2   udp    111  portmapper
    100021    1   udp   1024  nlockmgr
    100021    3   udp   1024  nlockmgr
    100024    1   udp   1025  status
    100024    1   tcp   1024  status
## END ##

# ps -eaf #
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 13:37 ?        00:00:05 init [3]
root         2     1  0 13:37 ?        00:00:00 [kflushd]
root         3     1  0 13:37 ?        00:00:00 [kupdate]
curtis    4220   838  0 14:12 tty2     00:00:00 -bash
root      5077   545  0 14:36 ?        00:00:00 /bin/sh
root      5275   524  0 14:59 ?        00:00:00 /usr/sbin/sshd
root      5278  5275  0 15:00 pts/2    00:00:00 -sh
root      6110   847  0 16:42 ttyl     00:00:00 bin/t_bash
root      6124  6110  0 16:43 ttyl     00:00:00 t_bash ./ir-script-linux2.sh
root      6125  6110  0 16:43 ttyl     00:00:00 bin/t_nc 102.60.21.187 10000
root      6138  6124  0 16:43 ttyl     00:00:00 ./bin/t_ps -eaf
## END ##
```

Figura 4.39: Comandos ejecutados en BRJDEV_live _response_data.txt_1
Fuente: Entorno virtual Caine 2.0, Elaboración: Las autoras



```
  GNU nano 2.2.2              File: BRJDEV_live_response_data.txt

# File Listings #
permissions;access date;access time;modification date;modification time;change date;change time;u$
755;09/08/03;16:41:31;09/08/03;14:56:44;09/08/03;14:56:44;0;0;4096;/
755;09/08/03;13:43:06;08/23/03;07:39:08;08/23/03;07:39:08;0;0;16384;/lost+found
755;09/08/03;13:43:06;09/08/03;13:37:26;09/08/03;13:37:26;0;0;1024;/boot
755;09/08/03;13:43:06;08/23/03;07:39:14;08/23/03;07:39:14;0;0;12288;/boot/lost+found
644;08/25/00;08:43:14;08/25/00;08:43:14;08/23/03;07:44:38;0;0;0;/boot/kernel.h-2.4.0
777;09/08/03;15:01:43;08/23/03;07:48:34;08/23/03;07:48:34;0;0;15;/boot/kernel.h
644;09/08/03;13:37:39;08/22/00;16:56:55;08/23/03;07:45:49;0;0;200285;/boot/System.map-2.2.16-22
644;08/22/00;16:56:55;08/22/00;16:56:55;08/23/03;07:45:49;0;0;11773;/boot/module-info-2.2.16-22
755;08/22/00;16:56:55;08/22/00;16:56:55;08/23/03;07:45:50;0;0;1621492;/boot/vmlinux-2.2.16-22
644;08/30/03;12:00:10;08/22/00;16:56:55;08/23/03;07:45:51;0;0;627392;/boot/vmlinuz-2.2.16-22
777;08/30/03;11:56:56;08/23/03;07:46:39;08/23/03;07:46:39;0;0;20;/boot/vmlinuz

# MD5 Sums #
f67f4348b85b4de31a895d1b509f67cb */dev/MAKEDEV
296c3ca2a7cbf7b22d813fdab155d5fc */etc/sysconfig/network-scripts/ifcfg-lo
0d093823df4c2956c9c0eb5f4f8324be */etc/sysconfig/network-scripts/ifdown-post
4e546667b6ccf78077b9c2ef20f219b4 */etc/sysconfig/network-scripts/ifdown-ppp
5ff45b4e873b8cb2cd7e0e2f233dc251 */etc/sysconfig/network-scripts/ifdown-sl
9e76413f2c9de0fb257969996422c4be */etc/sysconfig/network-scripts/ifup-aliases
200bfd56698c04a390fb780c76fd0ee5 */etc/sysconfig/network-scripts/ifup-ipx
276cb8829e9be35f339b7958c82964f0 */etc/sysconfig/network-scripts/ifup-plip
c40eba11b3ed793e15ed68fed2af818a */etc/sysconfig/network-scripts/ifup-post
0162610f8d4f3dc0a420398d4b5dd329 */etc/sysconfig/network-scripts/ifup-ppp
8e1a0a91cc7046c3f655c5c6d90d1291 */etc/sysconfig/network-scripts/ifup-routes
d97ec83fcea82842b99f83ee350e093a */etc/sysconfig/network-scripts/ifup-sl

# lsof_4.63 #
COMMAND    PID    USER   FD   TYPE     DEVICE    SIZE      NODE NAME
init         1    root  cwd    DIR       22,8    4096         2 /
init         1    root  rtd    DIR       22,8    4096         2 /
init         1    root  txt    REG       22,8   27452     29617 /sbin/init
init         1    root  mem    REG       22,8  434945     71697 /lib/ld-2.1.92.so
init         1    root  mem    REG       22,8 4776568     71704 /lib/libc-2.1.92.so
init         1    root    0u  unix 0xcffdba40               233 socket
init         1    root   10u  FIFO      22,8             29615 /dev/initctl
kflushd      2    root  cwd    DIR       22,8    4096         2 /
kflushd      2    root  rtd    DIR       22,8    4096         2 /
kflushd      2    root    0u  unix 0xcffdba40               233 socket
kflushd      2    root   10u  FIFO      22,8             29615 /dev/initctl
bash      9918    root    2u   CHR       4,3             33693 /dev/tty3
bash      9918    root  255u   CHR       4,3             33693 /dev/tty3
## END ##

# /etc/passwd #
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:
daemon:x:2:2:daemon:/sbin:
adm:x:3:4:adm:/var/adm:
lp:x:4:7:lp:/var/spool/lpd:
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:
news:x:9:13:news:/var/spool/news:
uucp:x:10:14:uucp:/var/spool/uucp:
operator:x:11:0:operator:/root:
games:x:12:100:games:/usr/games:
gopher:x:13:30:gopher:/usr/lib/gopher-data:
ftp:x:14:50:FTP User:/var/ftp:
```

Figura 4.40: Comandos ejecutados en BRJDEV_live _response_data.txt_2
Fuente: Entorno virtual Caine 2.0, Elaboración: Las autoras

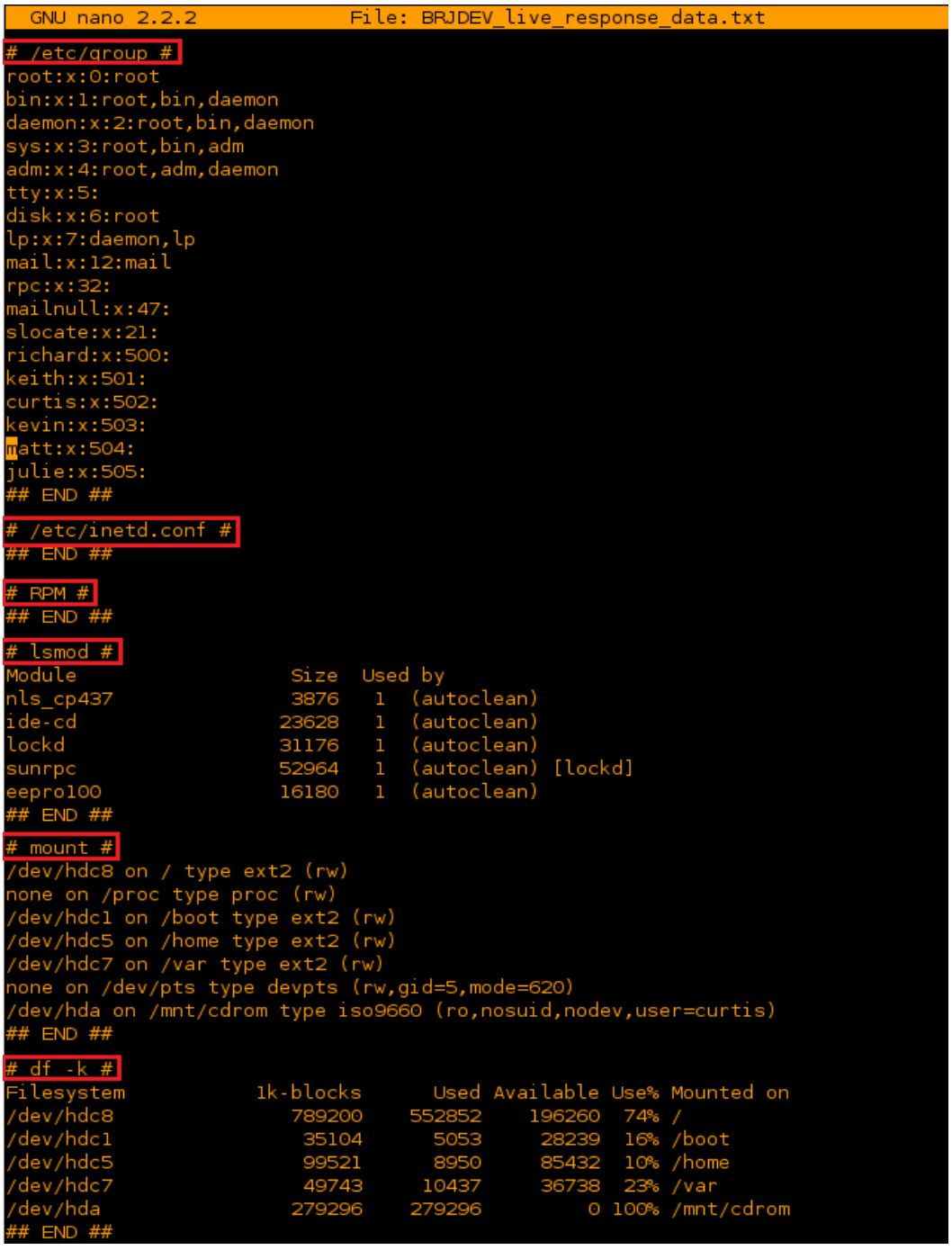


Figura 4.41: Comandos ejecutados en BRJDEV_live _response_data.txt_3
Fuente: Entorno virtual Caine 2.0, Elaboración: Las autoras

4.9.1.3.2 PASSWD

En este log podemos observar los usuarios propios de la empresa, pero existe el usuario “lpd:x:0::/:/bin/sh”el que cuenta con privilegios de root y no fue creado por el personal de IT.

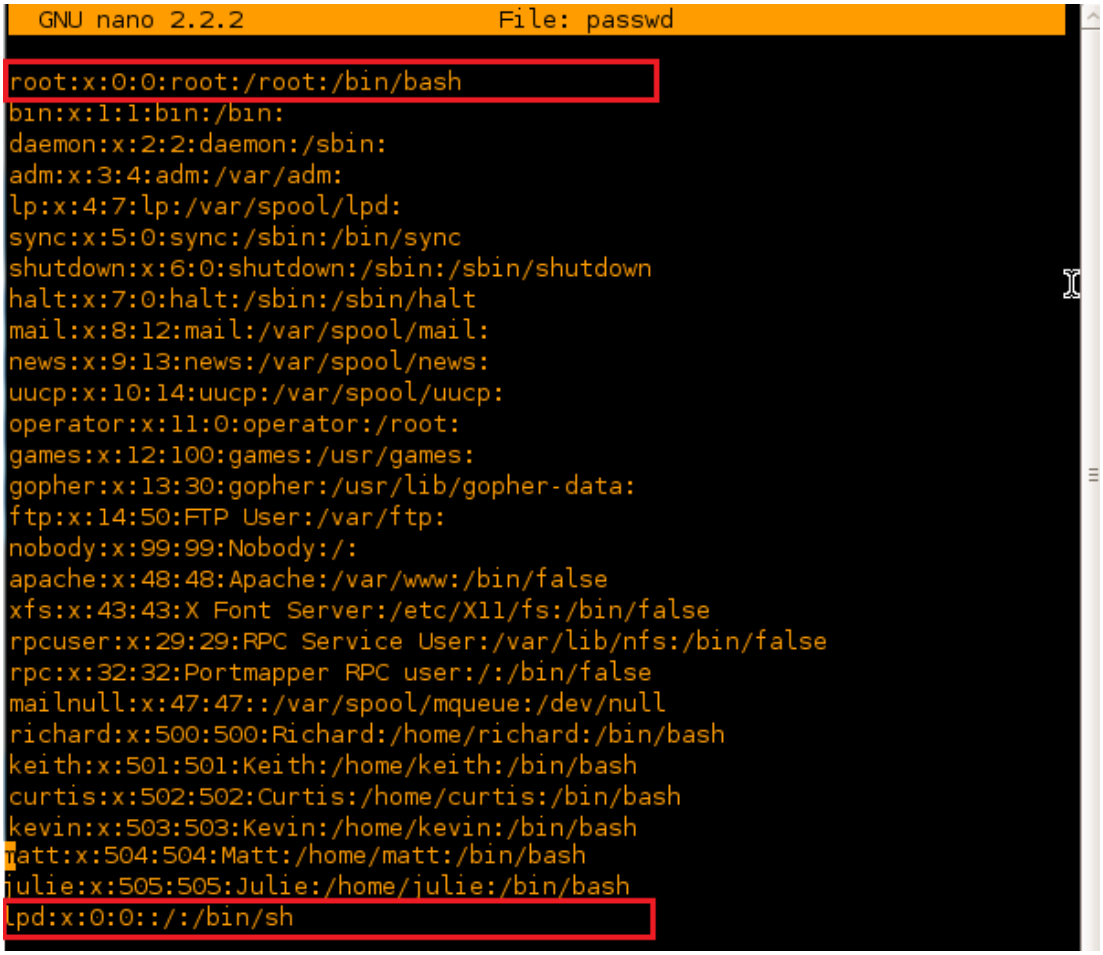


Figura 4.42: Passwd
Fuente: Entorno virtual Caine 2.0, Elaboración: Las autoras

4.9.1.3.3 Richard.bash_history

En este log podemos observar todos los registros de comandos que realizo RICHARD (sin embargo en este caso se supone que el atacante, utilizo la cuenta de Richard para acceder).

Existen comandos que nos ayudan a ver varios eventos que realiza este usuario, uno de ellos es que muestra en lista los procesos y busca el proceso llamado Datapipe (Datapipe es un programa utilizado para re-direccionar puertos de un equipo a otro, para evitar ciertos filtros de seguridad como por ejemplo un firewall).

Luego envía una señal a un proceso de un usuario con información adicional, luego empaqueta archivos en uno solo archivo, además imprime quién está autenticado en el sistema.

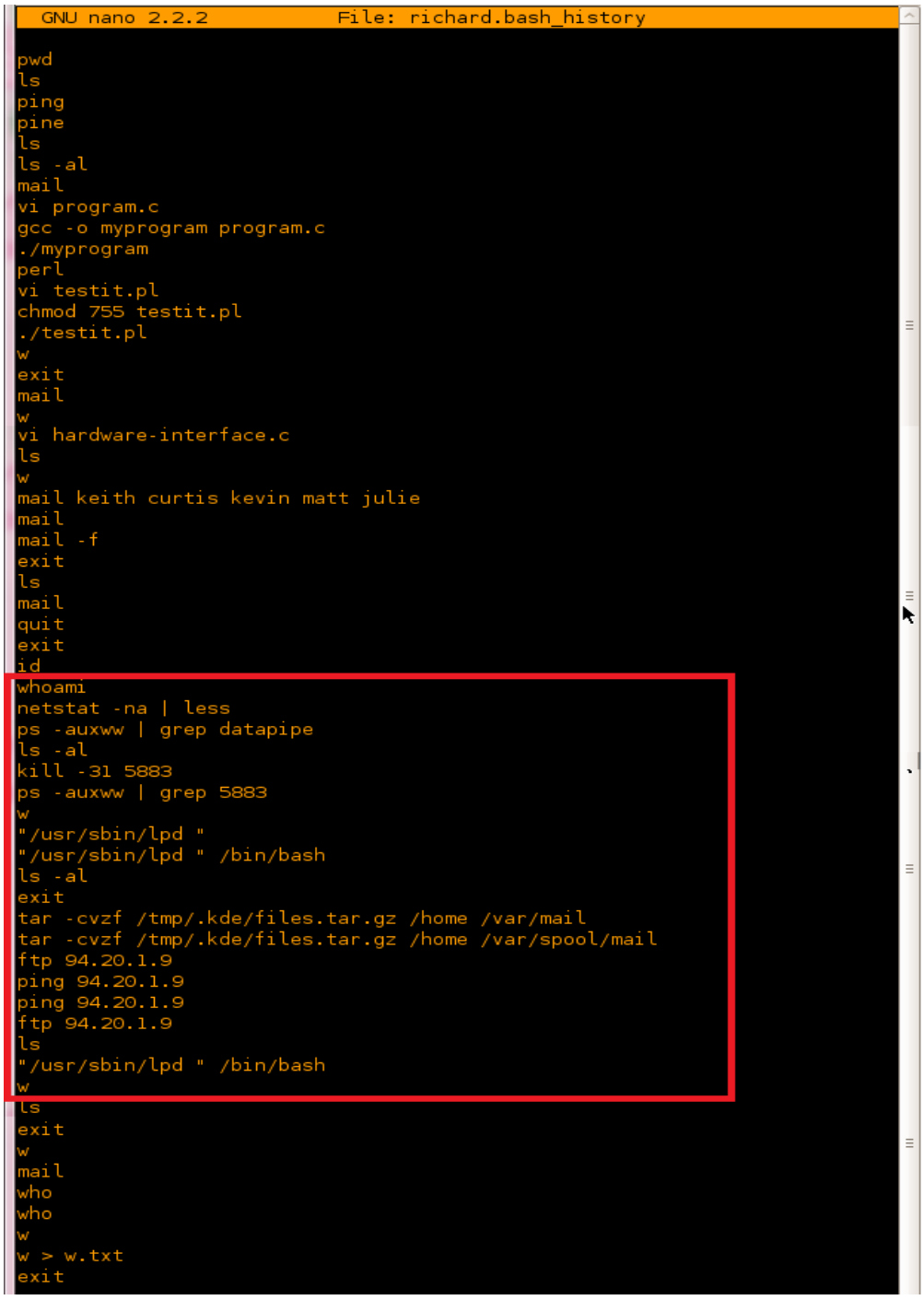


Figura 4.43: *Richard.bash_history*
Fuente: Entorno virtual Caine 2.0, Elaboración: Las autoras

4.9.1.3.4 Secure

En este archivo podemos observar que la dirección 94.90.84.93 (dirección del intruso) realiza login, finger y shell en varias ocasiones en la fecha y hora en la que ocurren varios eventos.

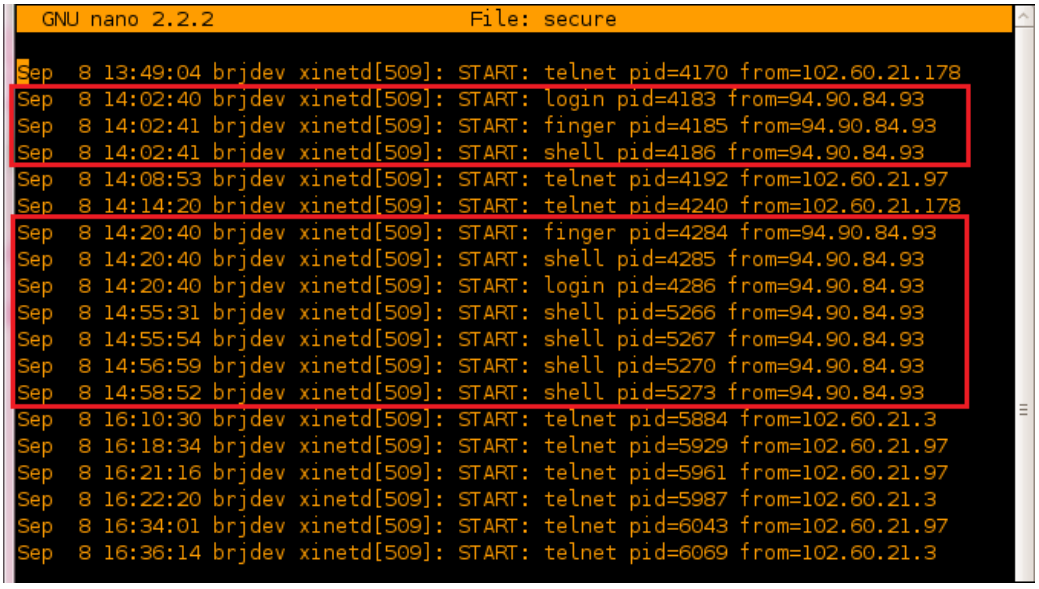


Figura 4.44: File_Secure
Fuente: Entorno virtual Caine 2.0, Elaboración: Las autoras

4.9.1.3.5 Secure.1

Podemos observar que por medio de telnet y ftp acceden varias ocasiones pero cabe recalcar que todas las direcciones IP son de la red de la empresa.

Figura 4.45: Secure.1
Fuente: Entorno virtual Caine 2.0, Elaboración: Las autoras

4.9.1.3.6 Syslog.conf

Nos muestra varias líneas de comandos que han sido usados en el equipo.

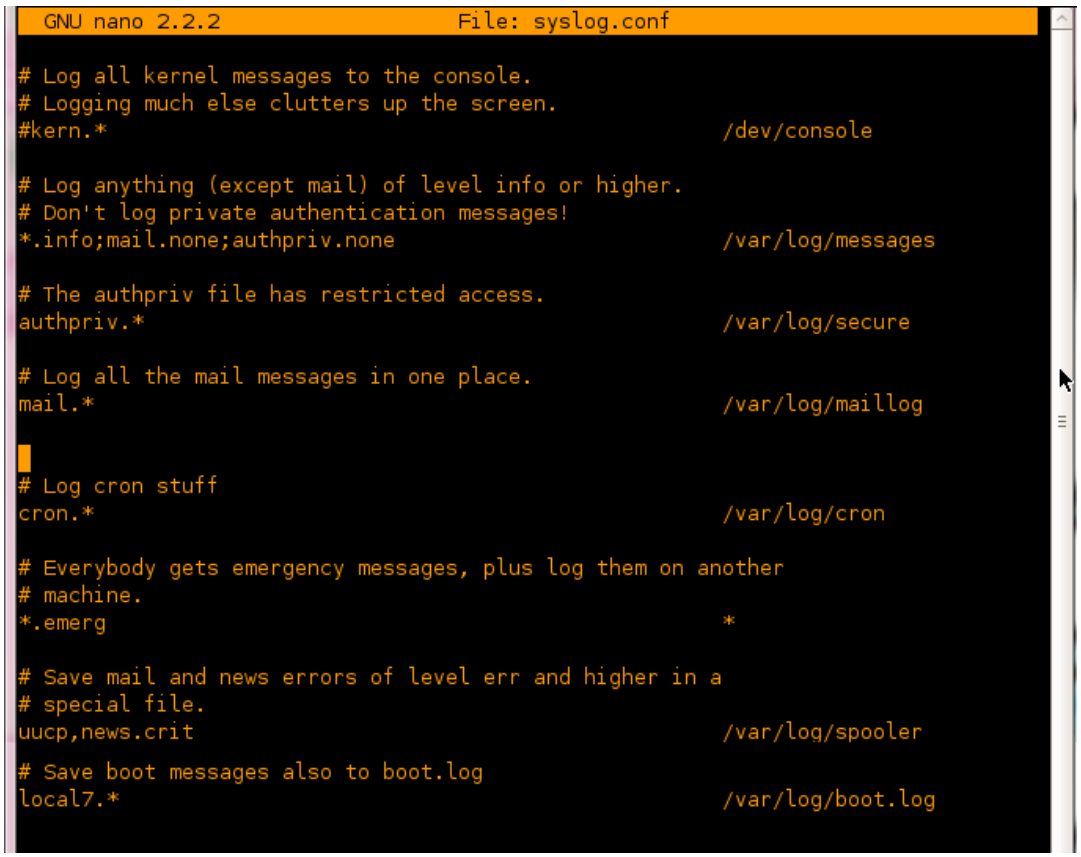


Figura 4.46: Syslog.conf
Fuente: Entorno virtual Caine 2.0, Elaboración: Las autoras

4.9.2 Innocent_ssh_client.dat

Una vez ya obtenida la copia de la evidencia, exploraremos la información de la misma.

- Inicializamos el sistema caine, abrimos una terminal dentro del directorio creado para el caso.
- La información se encuentra comprimida, por lo que utilizamos el comando gzip –d.

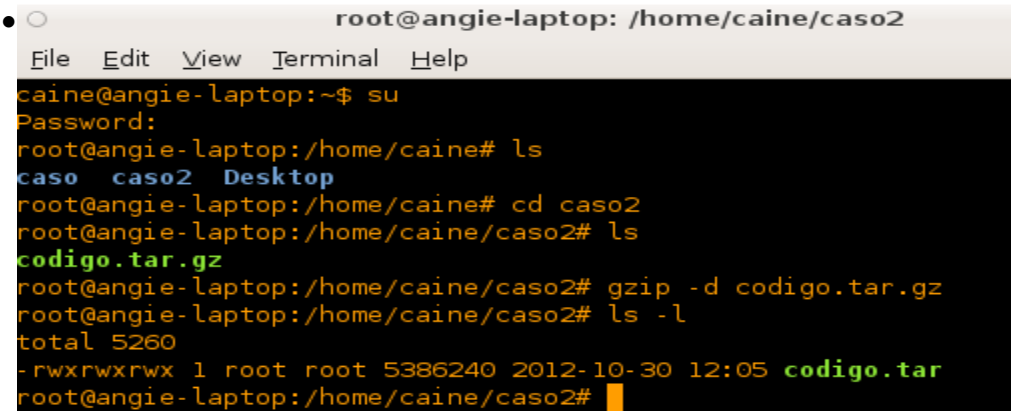


Figura 4.47: Evidencia digital
Fuente: Entorno virtual Caine 2.0, Elaboración: Las autoras

- Como la evidencia aún se encuentra comprimida utilizaremos el comando tar –xvf, para extraer los archivos. Como resultado obtenemos un archivo.gz que con el comando gzip –d procederemos a descomprimir.

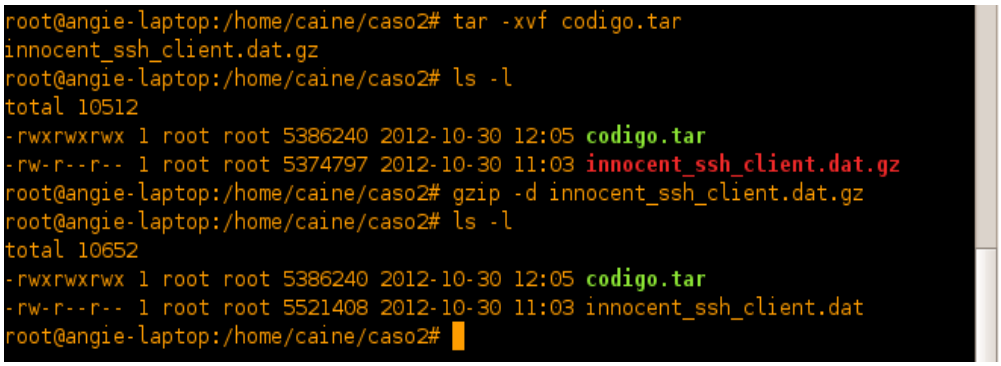


Figura 4.48: Evidencia Digital_1
Fuente: Entorno virtual Caine 2.0, Elaboración: Las autoras

4.9.2.1 Análisis del código utilizando herramientas online

En la web encontramos una variedad de herramientas que nos permiten realizar, análisis de archivos que contengan código malicioso de manera gratuita.

Entre ellos están los que hemos manipulado para obtener un reporte del comportamiento del código. El principio de esta herramienta es el mismo, utilizan un número determinado de antivirus para realizar un escaneo de forma

rápido del archivo, para saber dónde se ha producido el error o la falla ocasionada, por la inyección del código al sistema.

4.9.2.1.1 virscan

Cargamos el archivo Innocent_ssh_client en Virscan para que realice un escaneo del archivo.

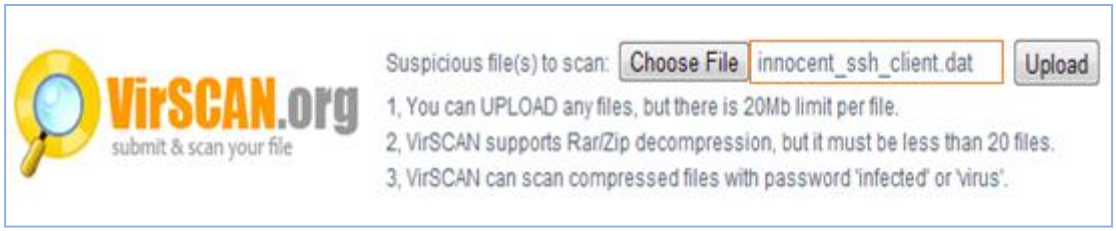


Figura 4.49: Cargado de la imagen en la Url de Virscan.org
Fuente: virscan.org

Una vez realizado el escaneo nos presenta una tabla, donde define el tipo de archivo, genera un MD5 y SHA1, establece cual o cuales han sido los archivos que se encuentran comprometidos por la entrada del código malicioso.

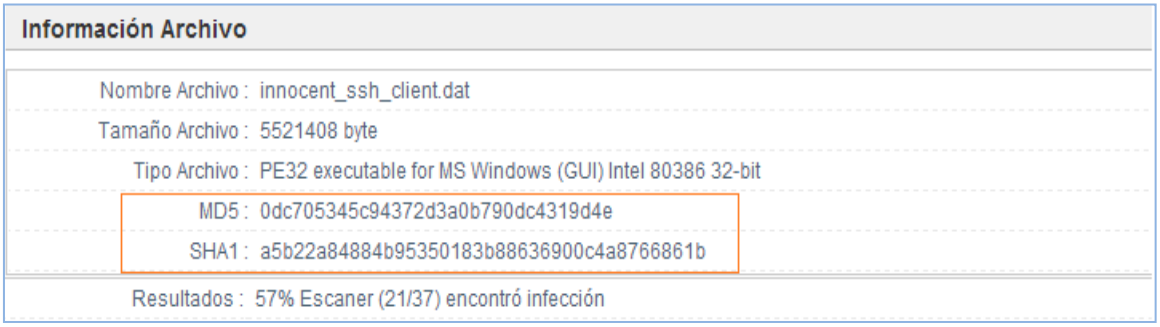


Figura 4.50: información de virscan
Fuente: virscan.org

4.9.2.1.2 Anubi

Los servidores de Anubi, permite que se suban archivos o direcciones URL, para realizar el análisis con sus motores de forma interna y transparente para el usuario.

Una vez ya finalizado el análisis presenta una tabla con el MD5 del archivo, nos permite obtener 4 tipos de formatos del reporte final estos son: HTML, XML,

PDF y test. Para la interpretación del usuario para determinar cuáles van a ser las directrices a ejecutarse.

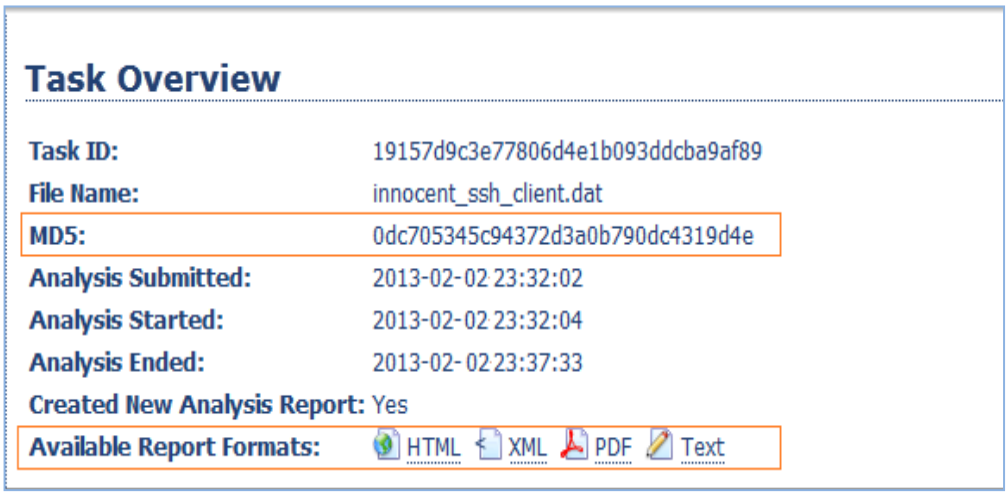


Figura 4.51: Task Overiew de Anubi
Fuente: http://anubis.iseclab.org

4.9.2.2 Análisis Estático

4.9.2.2.1 OllyDBG

Procederemos a utilizar este depurador que frecuente es usado por crackers software hecho por otros desarrolladores.

Al inicializar el programa, procederemos a cargar el archivo dando click en file open. Se abrirá la ventana para que busquemos el archivo a debuggear en este caso: innocent_ssh_client.dat. Allí vemos las cuatro partes de la ventana principal del programa: desensamblado, registros, stack o pila, dump.

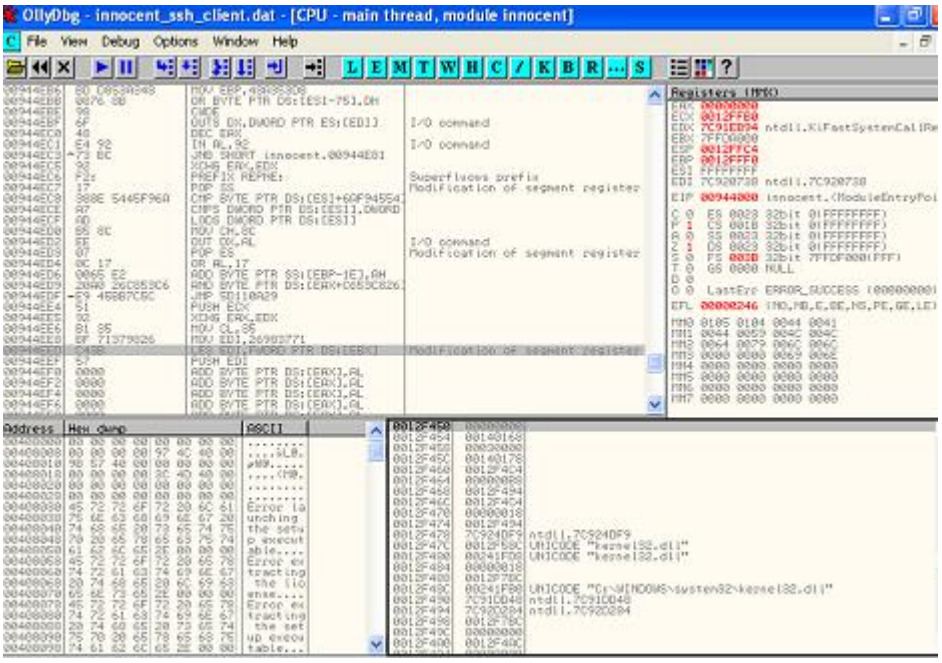


Figura 4.52: Ventana principal de OllyDBG
Fuente: Programa OllyDBG, Elaboración: Las autoras

- Desensamblado, ahí presenta el listado de innocent_ssh_client.dat analizado.

- Registros, muestra información de manera más detallada, permitiendo cambiar el modo de visualización en tres formas (View FPU registers, View 3D now registers y view debug registers).
- Stack o pila, muestra la información relativa de ESP o al REGISTRO EBP.
- Dump, visualiza los hexadecimales.

OllyDBG cuenta con una barra de herramientas, en la parte superior la misma de gran utilidad para el investigador.

- El botón L o View-log, nos muestra los log lo cual puede ser configurado para mostrar diferentes tipos de información del código. Es la información sobre el arranque y de la información escrita del mismo por diferentes tipos de breakpoints condicional logs.

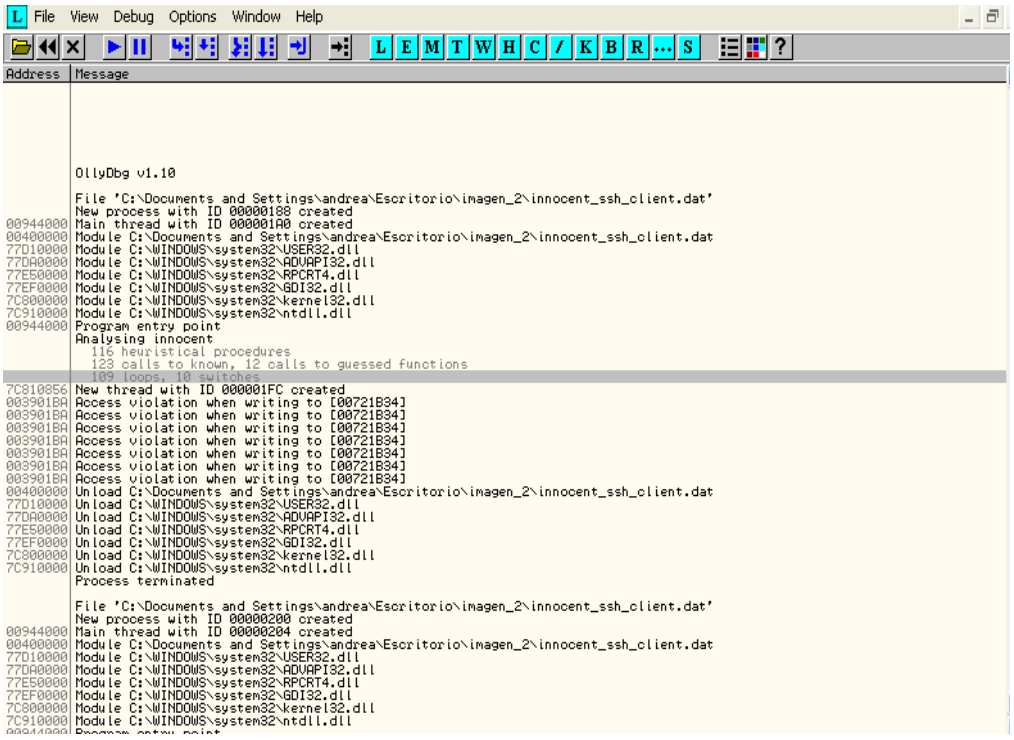


Figura 4.53: Ventana de View-log
Fuente: Programa Olly DBG, Elaboración: Las autoras

- El botón E o View-executables, nos muestra la lista de los ejecutables que utiliza el programa, exe, dlls, ocxs, etc.

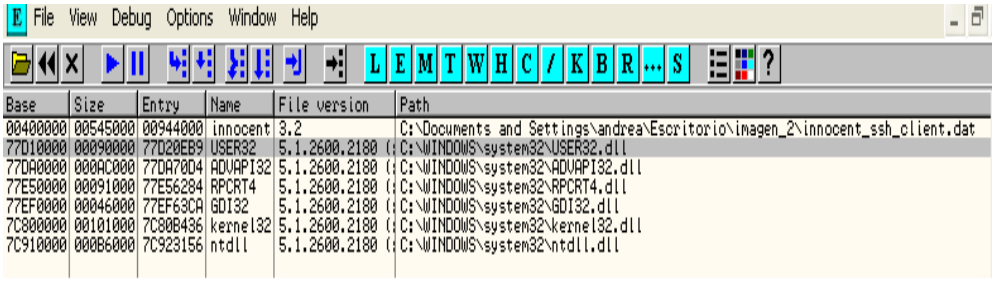


Figura 4.54: ventana de View-executables
Fuente: Programa Olly DBG, Elaboración: Las autoras

- El botón M o View-memory, nos enseña la memoria ocupada por nuestro programa, ahí se ven los dlls que utiliza los procesos, así como el stack y diversas secciones del sistema. Podemos buscar los strings, cadena hexadecimal, Unicode.

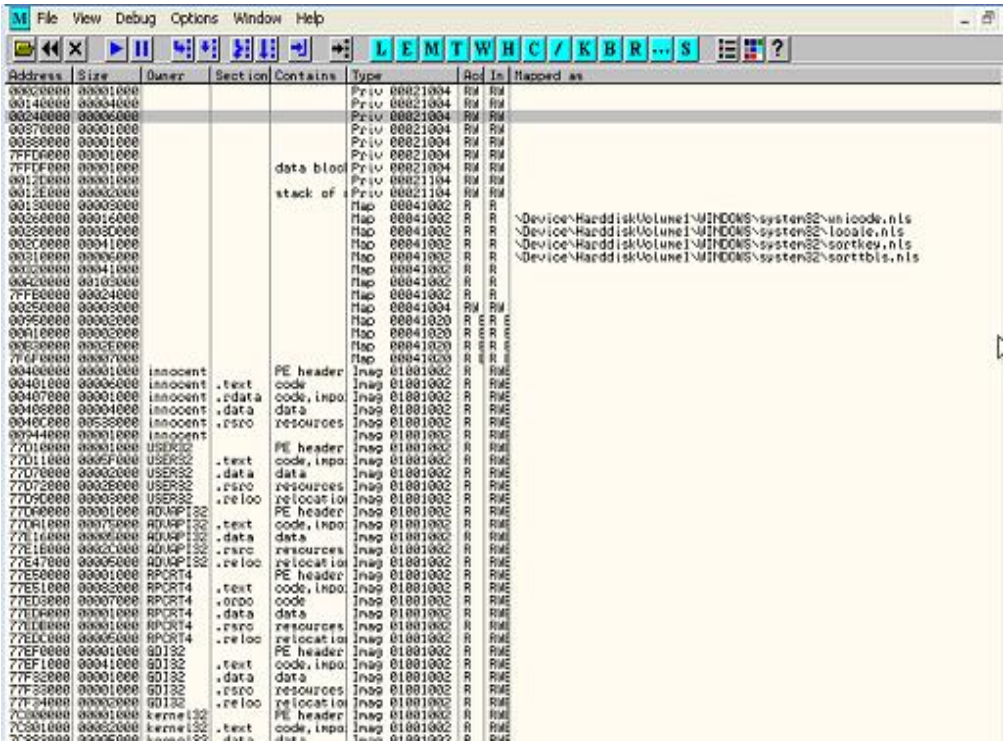


Figura 4.55: Ventana de View-memory
Fuente: Programa Olly DBG, Elaboración: Las autoras

- El botón R o View-References, nos permite ver los resultados cuando realizamos una búsqueda de referencia en el Olly.

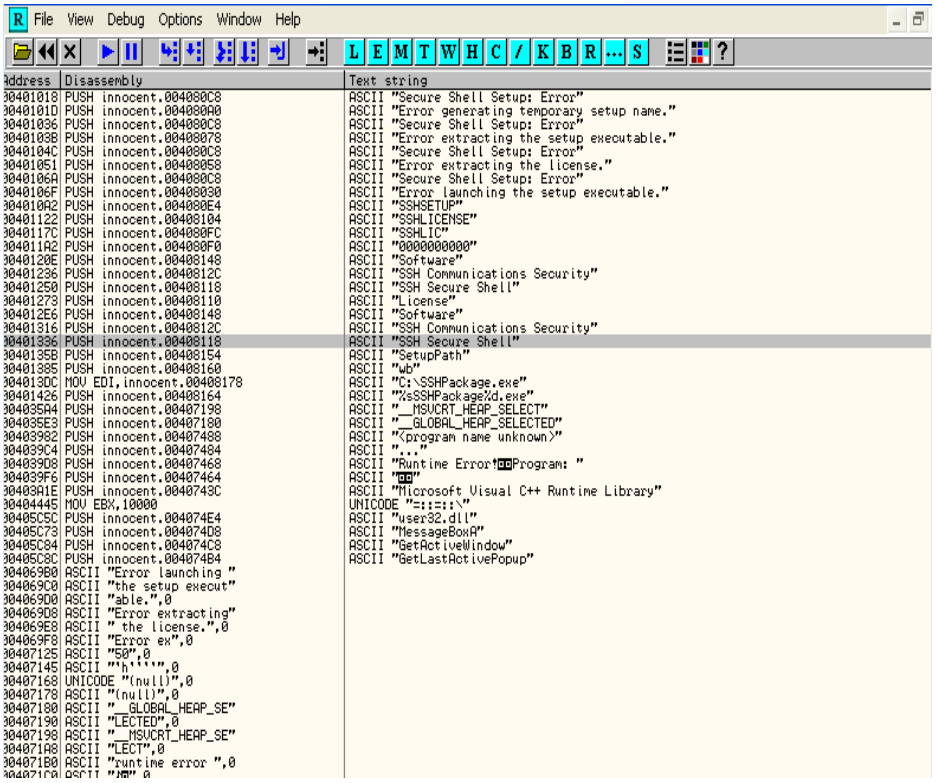


Figura 4.56: Ventana de View-references
Fuente: Programa Olly DBG, Elaboración: Las autoras

- El botón T o View-treads nos da el listado de los threads del programa
- El botón W o View Windows nos presenta las ventanas del programa.
- El botón H o View-Handles, nos muestra los handles.

4.9.2.2.2 IDA

La herramienta Interactive Disassembler, más conocida por su acrónimo IDA, es un desensamblador, el mismo que soporta una variedad de formatos ejecutables para diferentes plataformas de sistemas operativos.

Al inicializar el programa nos presenta una ventana en la cual nos aparece tres opciones de empezar a realizar el análisis: new, go, previous.

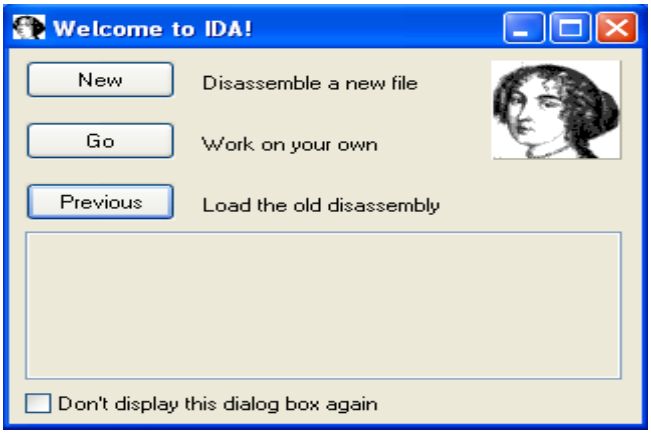


Figura 4.57: Inicio de IDA
Fuente: Programa IDA, Elaboración: Las autoras

En nuestro caso seleccionaremos la opción new para proceder a cargar al archivo: innocent_ssh_client.dat para el análisis correspondiente, especificamos el tipo de archivo en la ventana de load a new file. Si ya se ha analizado en archivo se escoge la opción de previous.

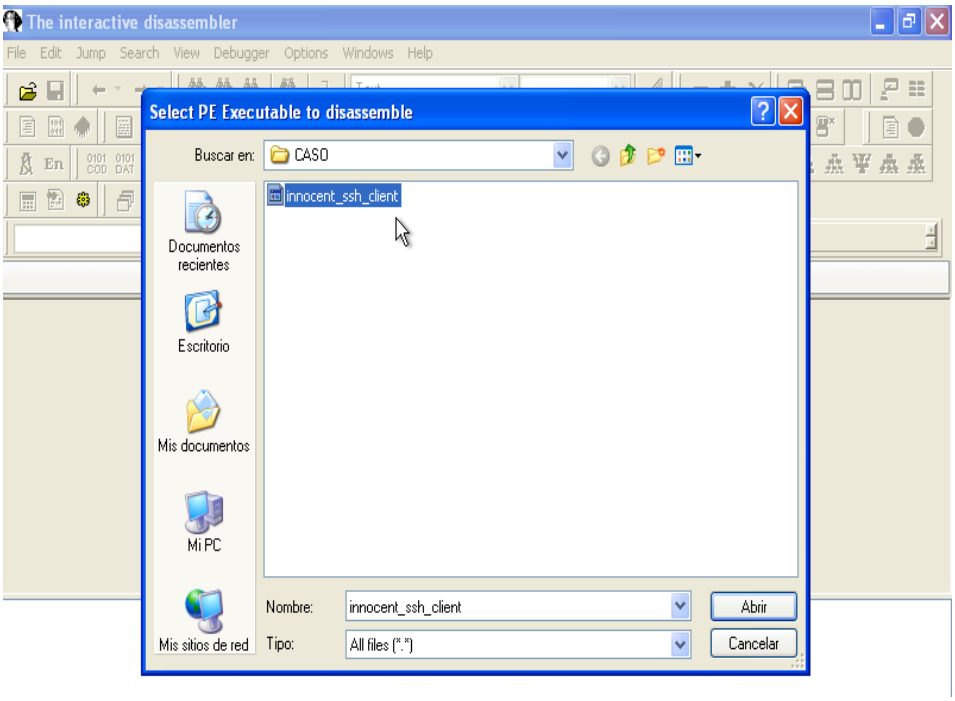


Figura 4.58: Ingreso del archivo a IDA
Fuente: Programa IDA, Elaboración: Las autoras.

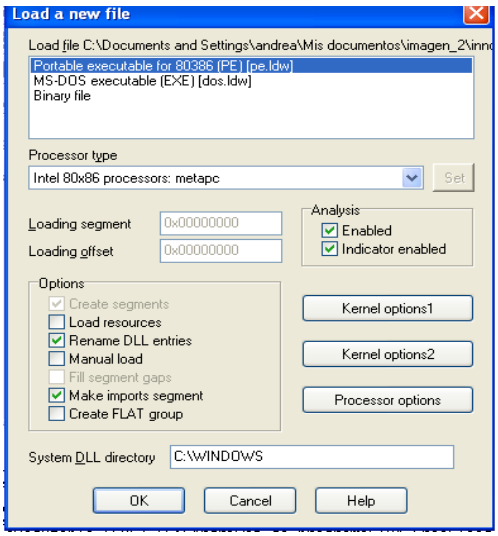


Figura 4.59: Ventana de load a new file
Fuente: Programa IDA, Elaboración: Las autoras

Inmediatamente cargado el archivo para el análisis nos presentará una ventana con el análisis completo del archivo en el cual encontraremos, texto hexadecimal, archivos que importa, archivos que exporta, strings, funciones y estructura.

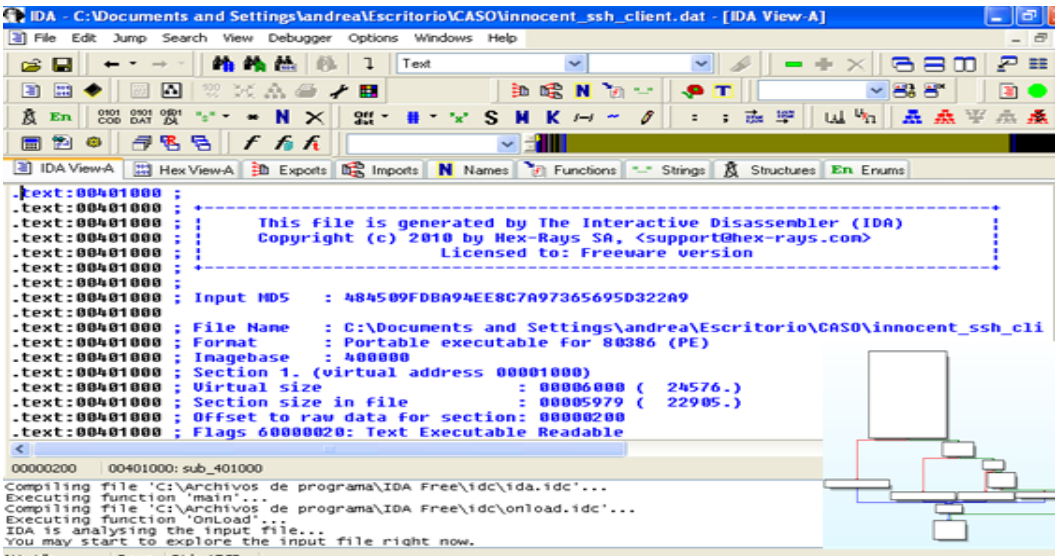


Figura 4.60: Ventana principal de IDA
Fuente: Programa IDA, Elaboración: Las autoras

Aquí ya observamos el código del archivo, desensamblado para poder analizar, cada una de funciones de cómo se comporta el código malicioso al momento de ser ejecutado por usuario. De esta manera podremos determinar cuáles son los archivos que se van a ver afectados al momento de su ejecución.

Una de las herramientas de IDA es esquema gráfico, el cual nos enseña cómo se encuentran definidos los procesos y el orden en el cual se ejecutan para lograr su objetivo, en nuestro caso es inyectar un error dentro de nuestro sistema operativo. Lo que determinaremos es como logra perjudicar al sistema anfitrión en este caso nuestra máquina de prueba.

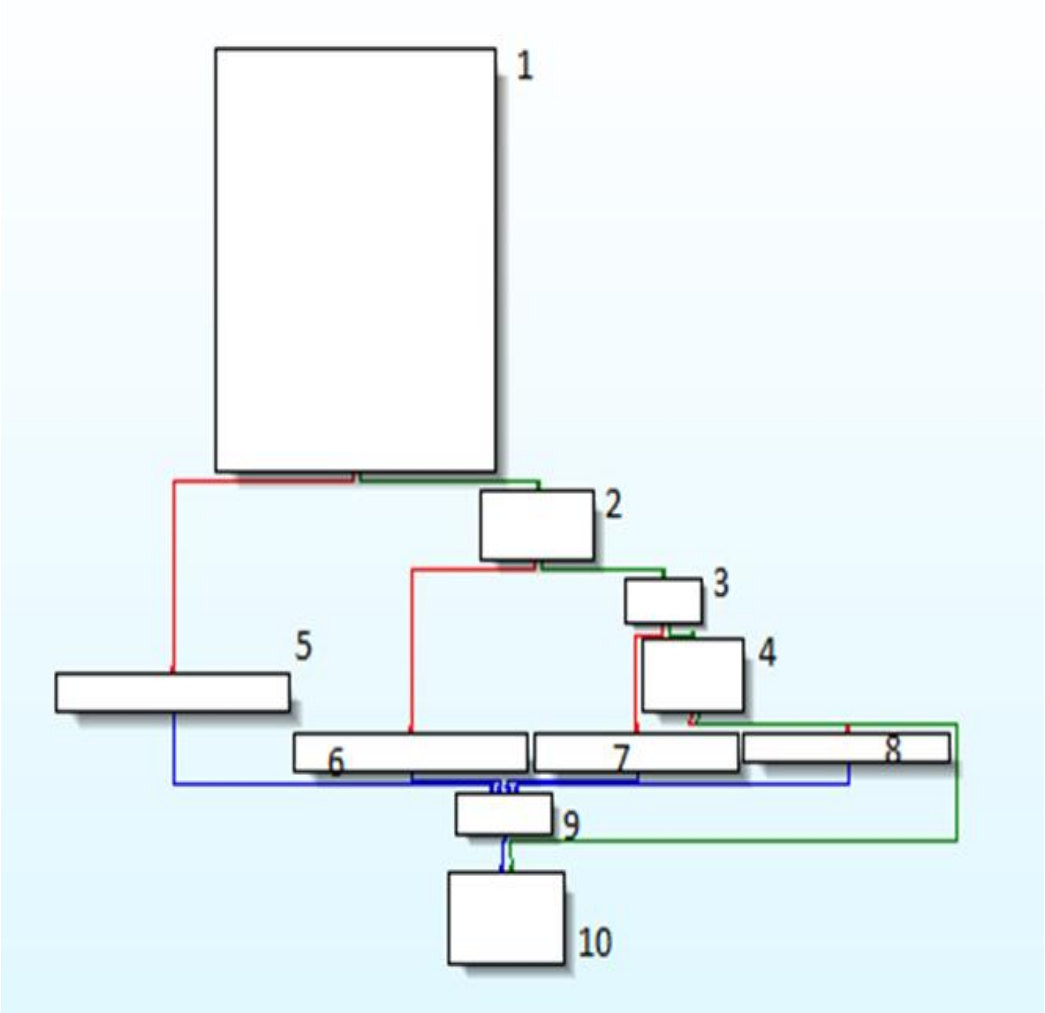


Figura 4.61: Diagrama de innocent_ssh_client.dat
Fuente: Programa IDA, Elaboración: Las autoras

En la parte superior IDA, cuenta con una barra de herramientas, claras y específica para descubrir los directrices que debe tomar cada proceso, para cuando el código sea ejecutado, responda sin problema.

Entre ellos tenemos: exports, imports, names, functions, strings, hex view-A, IDA View-A, segment registers.

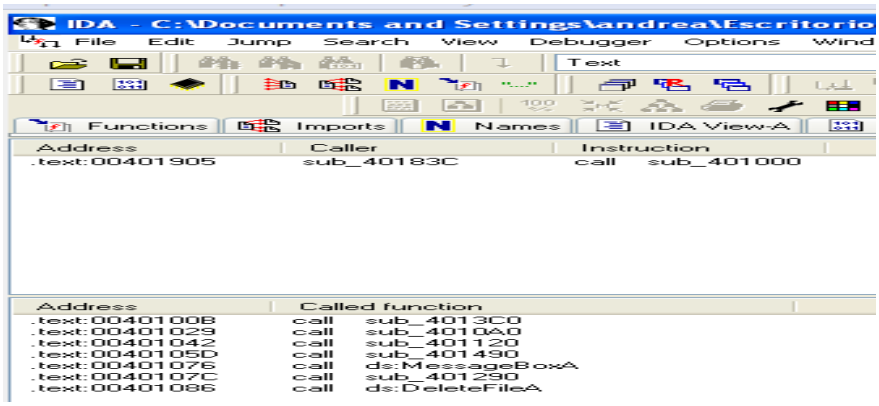


Figura 4.62: Ventana de functions de IDA
Fuente: Programa IDA, Elaboración: Las autoras

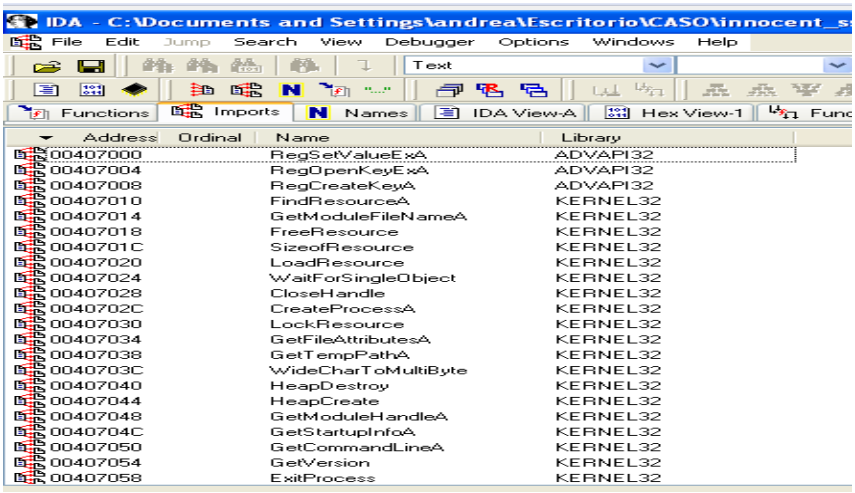


Figura 4.63: ventana de Imports de IDA
Fuente: Programa IDA, Elaboración: Las autoras

Procederemos a utilizar la herramienta debugger, que nos permite ejecutar el archivo y examinar de forma meticulosa cada proceso, damos click en start Process.

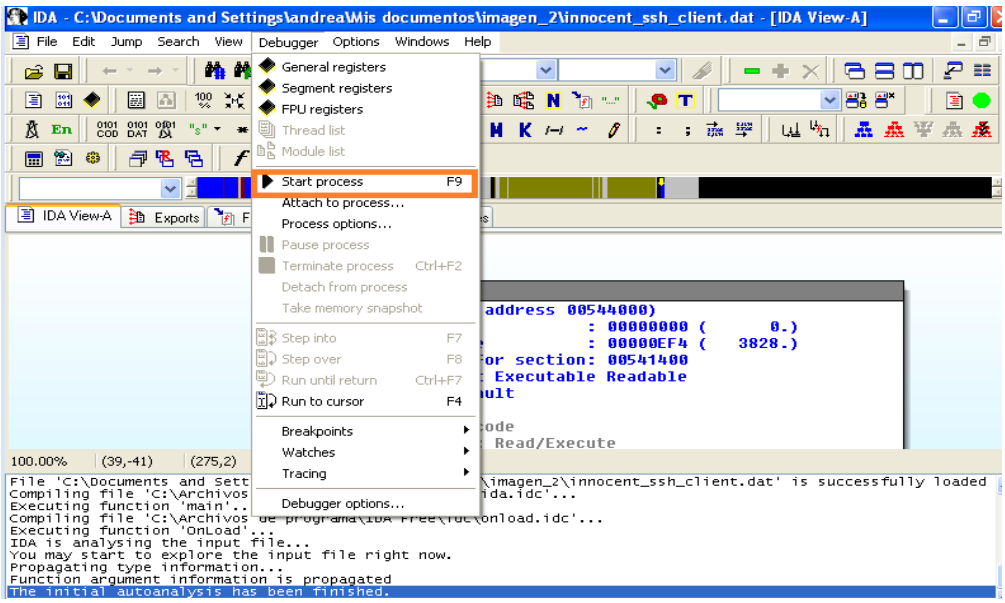


Figura 4.64: Inicio del debugger en IDA
Fuente: Programa IDA, Elaboración: Las autoras

Después de iniciar el proceso nos aparece un mensaje en el cual nos indica que se ejecutará en el sistema un programa malicioso de tipo virus troyanos. Damos click en yes.

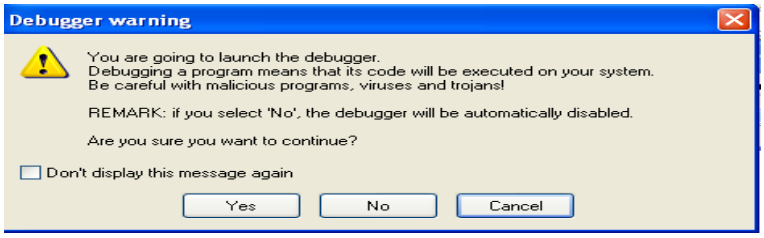


Figura 4.65: Ventana de debugger warning
Fuente: Programa IDA, Elaboración: Las autoras

Luego de inicializar el proceso nos presenta el siguiente mensaje: ***"innocent_ssh_client.dat: The instruction at 0x3901BA referenced memory at 0x721B34. The memory could not be written (0x003901BA -> 00721b34***)".

Podemos notar que ha ocurrido un movimiento sospechoso nos ubicamos en el proceso debug017, para interpretar su desenlace.

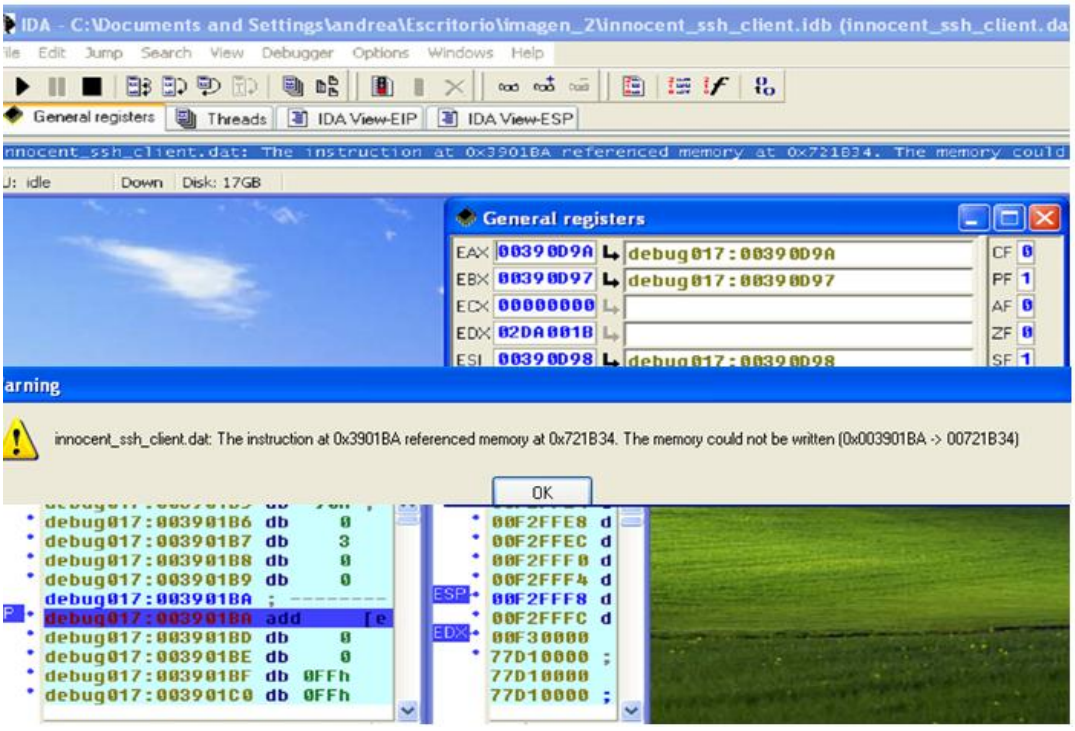


Figura 4.66: innocent_ssh_client.dat 0x3901BA 0X721B34
Fuente: Programa IDA, Elaboración: Las autoras

Analizando cada uno de los pasos, establecidos en el diagrama, notamos que su terminación, origina una llamada a Aplication Programming Interface “DeleteFileA”.

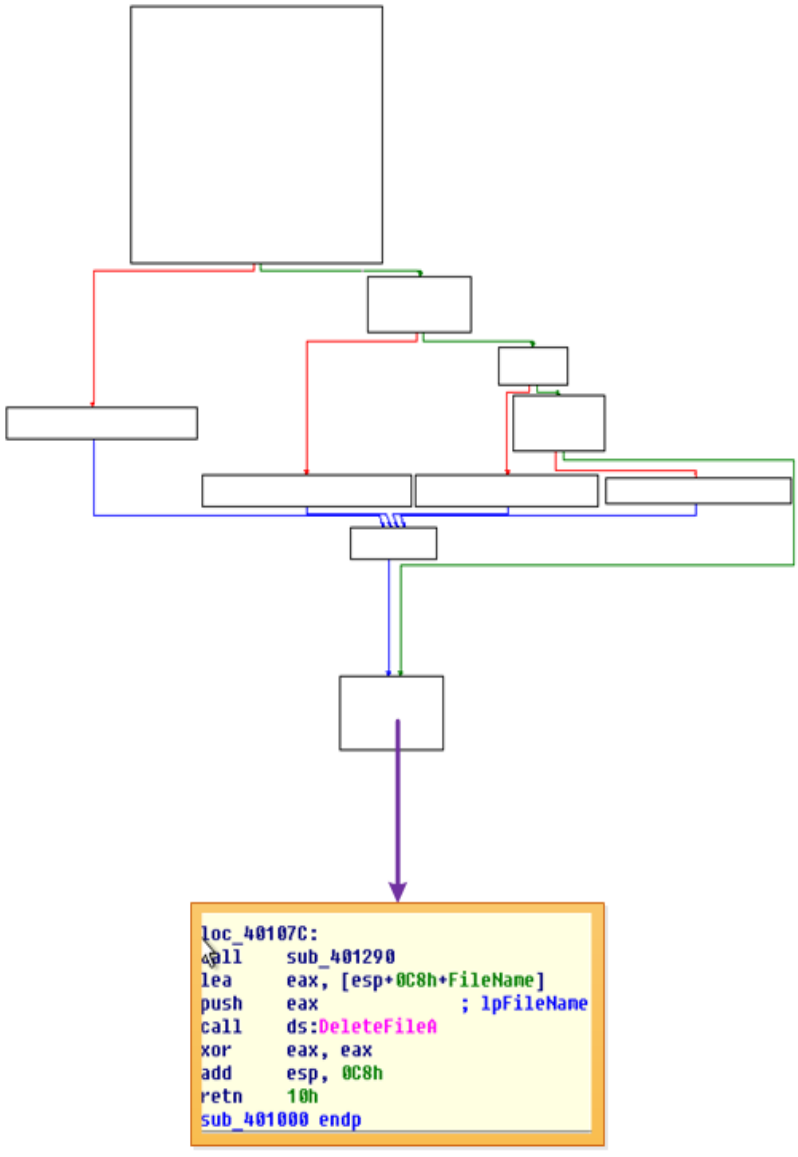


Figura 4.67: Mensaje de DeleteFileA
Fuente: Programa IDA, Elaboración: Las autoras

4.9.2.3 Análisis Dinámico

4.9.2.3.1 Process Explorer

El Process Explorer es de las herramientas que podemos utilizar para realizar, este tipo de análisis, ejecuta el código malicioso para observar lo que hace.

Antes de ejecutar el malware se debe tomar una captura de los procesos, luego se procederá a ejecutar el malware y se tomará otra captura de los procesos. Luego compararemos ambas capturas para saber cuáles han sido los procesos, que se han inicializado.

- Capturamos los procesos que se están ejecutando, en nuestra maquina víctima.

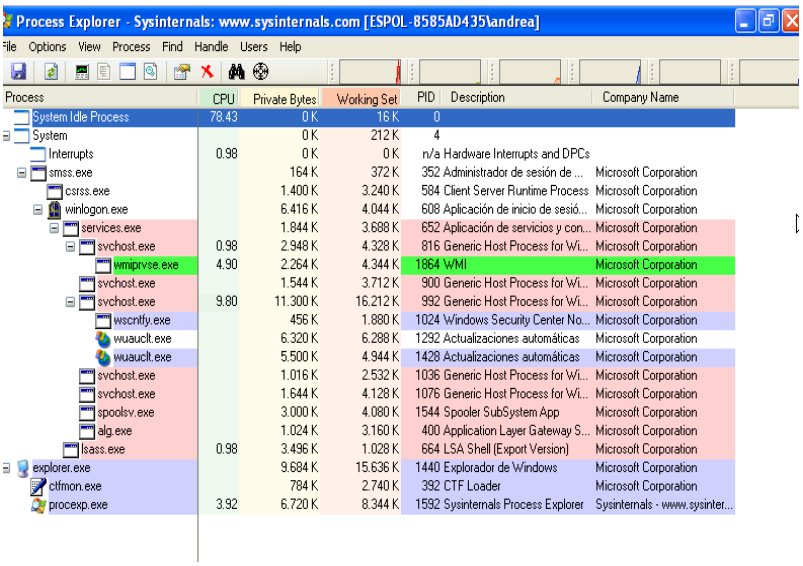


Figura 4.68: Process Explorer
Fuente: Programa Process Explorer, Elaboración: Las autoras

- Ejecutamos el malware dentro de nuestro sistema, como el código se encuentra comprimido, procederemos a descomprimirlo antes llevar al ejecutable.

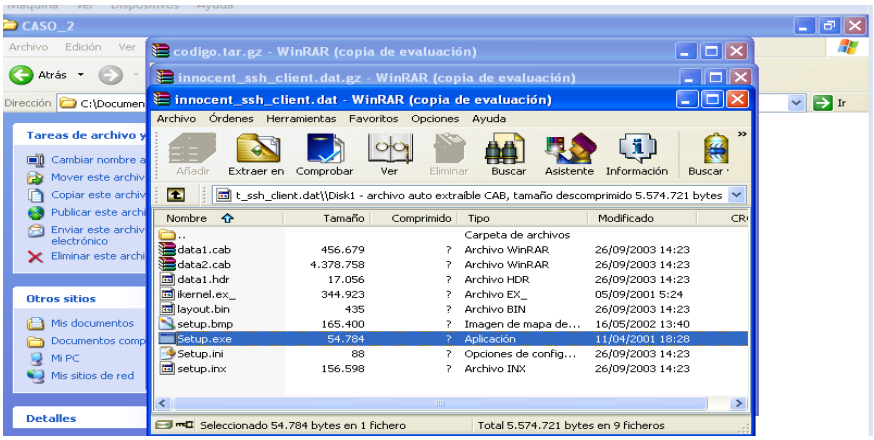


Figura 4.69: setup.exe
Fuente: Entorno virtual Windows xp , Elaboración: Las autoras

- Procederemos a ejecutar el código malicioso, y notamos que nos presenta un mensaje:

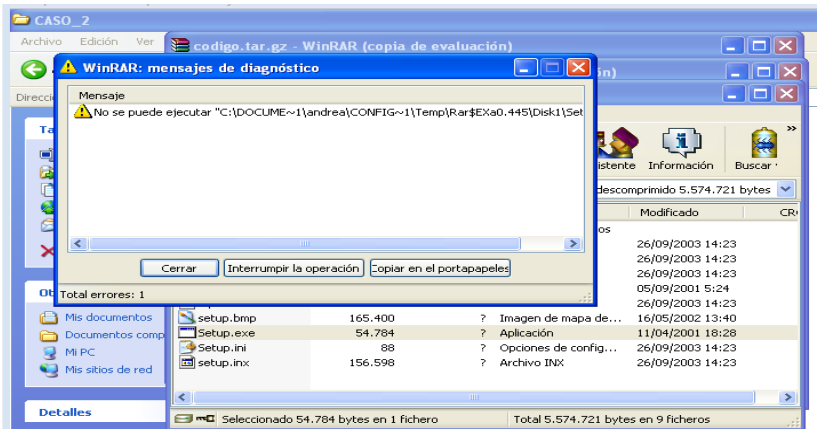


Figura 4.70: messages.box
Fuente: Entorno virtual Windows xp, Elaboración: Las autoras

- Realizamos la segunda captura de los procesos que está corriendo para saber si se ha inicializado un nuevo proceso, pero al ejecutar el Process Explorer el equipo se nos reinicia mostrándonos un pantallazo azul el cual indica un error.

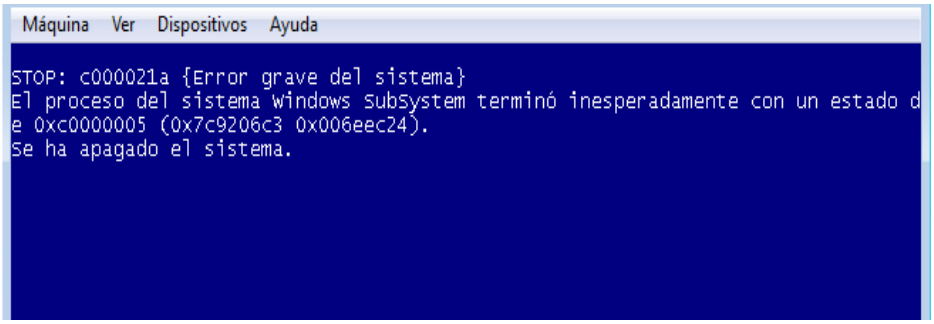


Figura 4.71: Error del Sistema
Fuente: Entorno virtual Windows xp, Elaboración: Las autoras

CONCLUSIONES Y RECOMENDACIONES

- **Conclusión del caso práctico Brj_Software**

El objetivo general era determinar si se produjo malversación de la información que contenía en el equipo involucrado en el siniestró informático.

La finalidad del análisis, era determinar por medio de un uso de una correcta metodología de procesos reconstruir lo ocurrido para así saber que era realmente lo que pretendía realizar el intruso.

Es viable determinar que el atacante accedió al sistema y se conectó al equipo desde la dirección IP 94.90.84.93, mientras que utilizó un segundo equipo con la dirección IP 94.20.1.9 como repositorio de archivos (ninguna de las dos direcciones pertenece a la red de la empresa).

Para acceder al equipo después de varios intentos, el atacante busca vulnerabilidades de seguridad por medio de varios medios como lo es Exploit (este medio busca las vulnerabilidades de seguridad), una de las vulnerabilidades que utilizó fue del servicio **lpd**(por el puerto 515).

El método utilizado fue el de desbordamiento de buffer. Una vez obtenido este acceso, el atacante creó una cuenta de usuario llamada **lpd**, con todos los privilegios de **root** para así poder obtener toda la información que necesitaba del equipo atacado.

Revisando la información con la que contamos, se pudo determinar que el atacante creo el directorio **/tmp/.kde** como repositorio, la cual almacena toda la información que desea. Dentro de las utilidades se encontró un popular *programa* de claves llamado **jhon the ripper versión 1.6**, utilizado para descifrar la clave del usuario **Richard.**

Para evitar que un firewall pudiera impedir la conexión del equipo atacado con el exterior, el atacante utilizó el programa **Datapipe** para re direccionar el puerto TCP 23 al puerto TCP 2323.

Al finalizar este análisis se logró determinar que Richard no es culpable y que existió un intruso que accedió a la red de la empresa sin problema, lo cual nos demuestra que la tecnología evoluciona cada día y que los tipos y métodos de ataques informáticos también evolucionan.

Al culminar nuestra investigación hemos adquirido conocimiento de herramientas que son altamente confiables y que muchos son de fácil uso. Pero no podemos olvidar que siempre se debe llevar un control de ciertos procedimientos.

- **Recomendaciones del caso práctico "BRJ SOFTWARE"**

Toda organización debe contar con herramientas que la protejan de los ataques malintencionados, tanto internos como externos. Somos conscientes de que en un mundo que avanza a pasos agigantados con la tecnología, debemos prevenir cualquier tipo de ataques y mantener nuestra infraestructura actualizada para poder mitigar estos o cualquier tipo de ataques.

La mejor forma de evitar situaciones engorrosas de fraudes, robo y siniestros informáticos es estableciendo controles, pero por sobre todo promover una cultura de seguridad en las organizaciones.

- Adecuar a la organización con seguridad física y lógicas básicas.
- Centralizarse en las deficiencias de seguridad más comunes las cuales suelen ser el camino para que el atacante tenga acceso a nuestros sistemas y por consiguiente a nuestros datos.
- Realizar revisiones periódicamente, en el entorno informático.
- Capacitar al personal para establecer objetivos institucionales.
- Ante los cambios tecnológicos se recomienda mantener la infraestructura tecnológica de la organización actualizada en el uso de herramientas y técnicas.
- Concientizar a los colaboradores sobre las responsabilidades legales que se tomarán en caso de fraudes, robos y siniestros informáticos.
- Implantar logs de auditoría.
- Establecer planes de contingencia, para poder resolver los problemas que se presentan de manera imprevista. Sin que ocasionen contrariedades de mayor índole.
- Implementar ACL, para evitar ataques y establecer un control de los privilegios con los que cuentan los usuarios, para proteger la institución.

- Capacitar al personal de IT y a los usuarios sobre ciertas medidas de seguridad o la información que ellos manejan, para establecer niveles de seguridad.
- Actualizar parches, hardware y software, ya que la mayor parte de los ataques se basan en explotar las vulnerabilidades como puertos, sistemas o aplicaciones.

- **Conclusión del caso práctico Innocent_ssh_client.dat**

El objetivo era determinar las instrucciones del programa, antes y después de inyectar el código malicioso a un máquina o sistema “víctima”.

Inmediatamente, terminado el análisis de innocent_ssh_client.dat, se puede determinar que el archivo cuenta con un ejecutable que produce un error en los archivos de Windows.

Realizando pruebas con Caine pudimos descubrir que el archivo al ser descomprimido, era un archivo .dat el mismo que pertenece a Windows. Por medio del análisis estático, se estableció que el archivo efectivamente era un malware que ocasionaba un error en el código en la línea 0x3901BA -> 00721B34.

El mismo que genera una serie de importaciones de archivos del win 32.dll y funciones específicas ya establecidas como la llamada al delete file A.

El análisis dinámico realizado confirma que al ser ejecutado el archivo, manipula los procesos de forma transparente para el usuario. Ya que emite un error al arrancar la máquina, presentando un pantallazo azul con un error grave del sistema, lo que si no es corregido provocara que la máquina sea inutilizada por completo.

Se ha hecho uso de las metodologías y herramientas con las que se cuenta para realizar un análisis de malware, enfatizando que no existen una variedad de herramientas que pueden ser utilizadas sin necesidad de pagar una licencia para su uso.

En definitiva, creemos que el trabajo realizado nos ha permitido adquirir nuevos conocimientos en casos de análisis de evidencia, tomando como directrices que la integridad de la evidencia es fundamental, para que el proceso de adquisición y análisis no sea cuestionado o puesto en tela de duda.

- **Recomendaciones del caso práctico innocent_ssh_client.dat**

Para obtener una administración adecuado de los recursos tecnológicos y de las herramientas, es preferible realizar los análisis dentro de un ambiente virtualizado sin acceso a internet, para salvaguardar los datos de in infortunio informático.

1. Implementacion de contrafuegos internos o un sistema de prevencion de intrusos, en la cual este crea una capa de seguridad a nivel interno.
2. Tener instalado en los equipos Antivirus y Antispyware o aplicaciones con la combinación de ambas.
3. Utilizar las Herramientas de Desinfección (Antivirus/Antispyware) para escanear dispositivos de almacenamiento externos.
4. Realizar de forma periódica copias de seguridad para evitar la pérdida de datos, en caso de que el malware dañes el sistema operativo.
5. Evite descargar archivos desde sitios no seguros tales como juegos, crackers o alguno que ofrezca algún tipo de servicio, cerciorémonos que sea una página segura.
6. Analice con su antivirus cada archivo que descargue (especialmente si son carpetas comprimidas) pues los códigos maliciosos pueden mezclar con los datos ocasionando errores en el sistema.
7. Descargue software desde sitios oficiales o confiables, para evitar la descarga de algún programa malicioso adjunto.

8. Desconfíe principalmente de archivos .exe y carpetas comprimidas .zip/.rar
9. No se recomienda que se mantenga conversaciones por chat con personas no conocidas, en caso de hacerlo tome todo tipo de precauciones para que no introduzca malware en su red sin tener conocimiento de lo que está ocurriendo.
10. No abra links si no conoce el remitente del correo, menos si el contenido trata acerca de una noticia increíble.
11. No acepte archivos si el usuario no le ha mencionado de que se trate o usted no lo haya solicitado.
12. No presione sobre links (enlaces) sin antes preguntar a la contraparte si él o ella ha enviado ese archivo.

ANEXOS

Anexo 1

Logs de Network Activity.

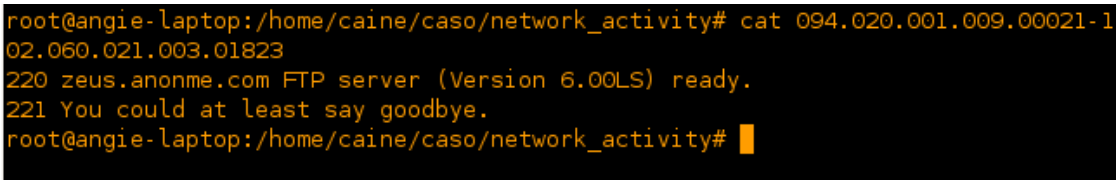


Figura 72: cat 094.020.001.009.00021-102.060.021.003.01823
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

Figura 73: 094.020.001.009.00021-102.060.021.003.01824
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

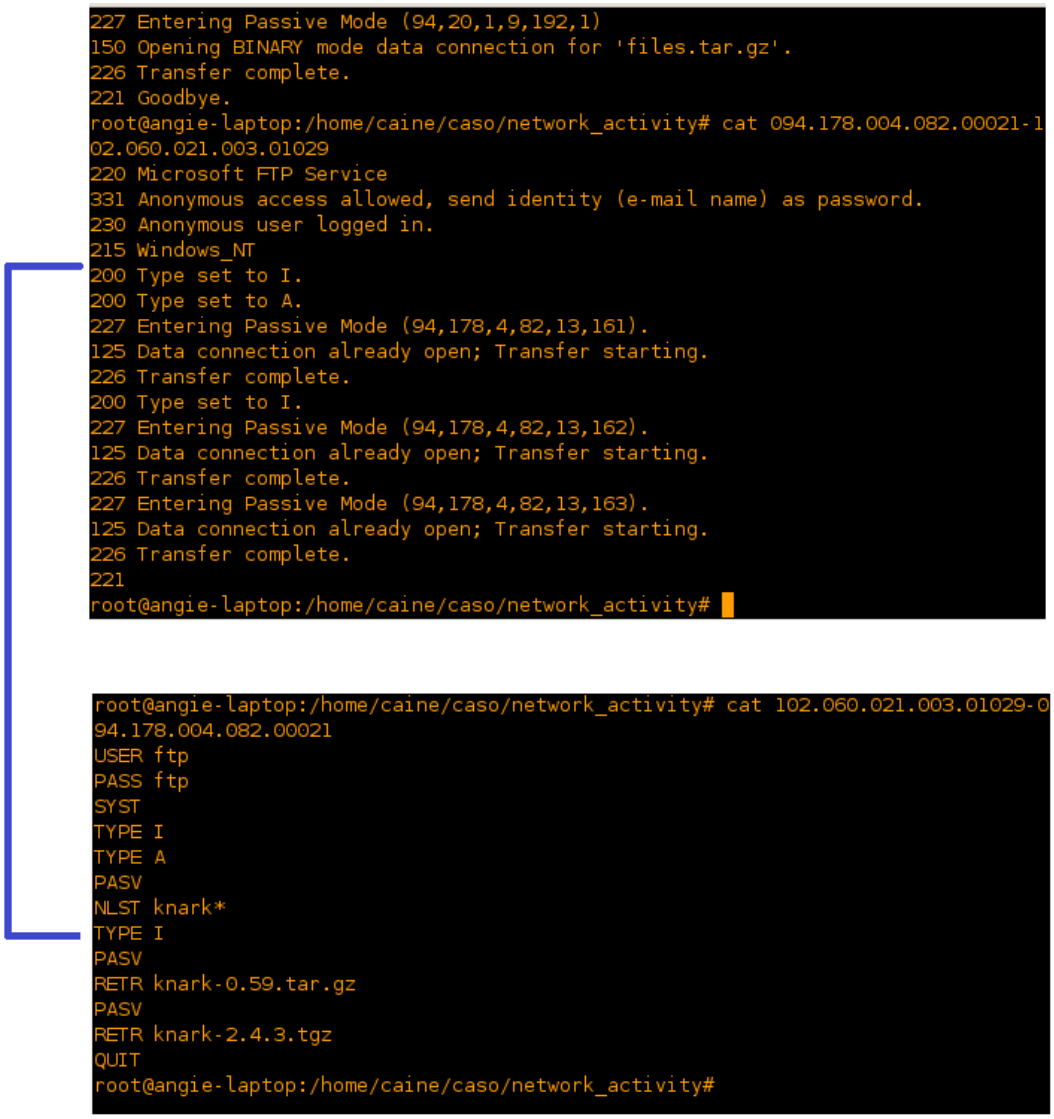


Figura 74: 102.060.021.003.01029-094.178.004.082.00021
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

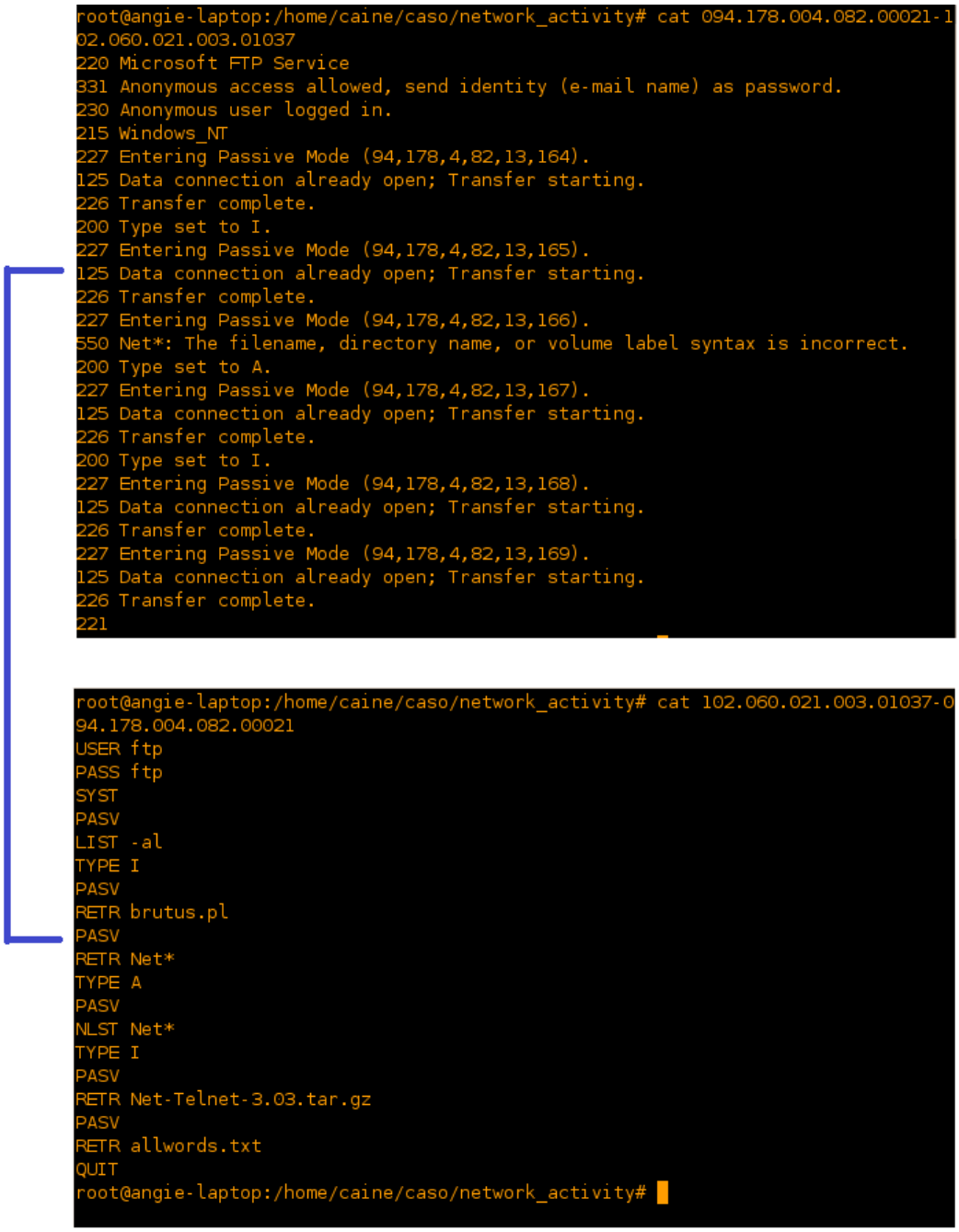


Figura 75: 102.060.021.003.01037-094.178.004.082.00021
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

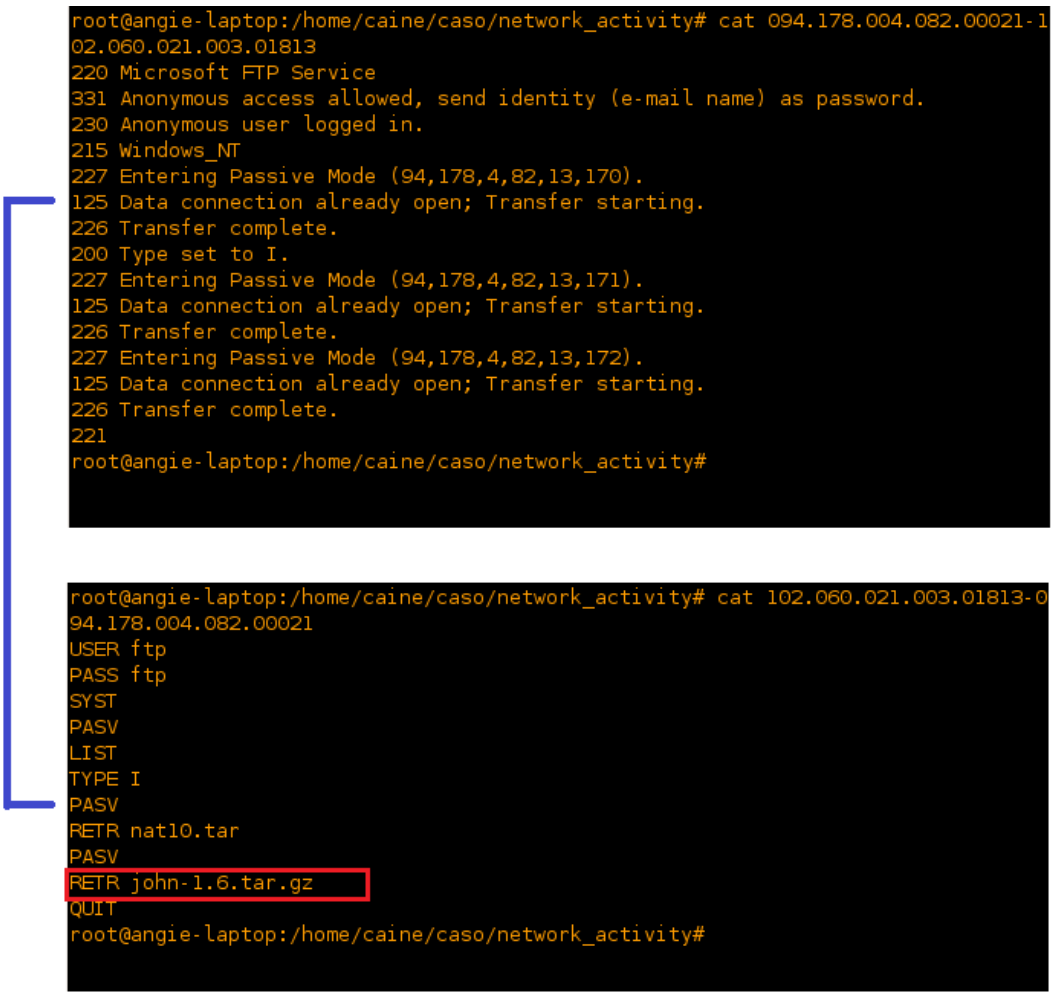


Figura 76: RETR John-1.6.tar.gz
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

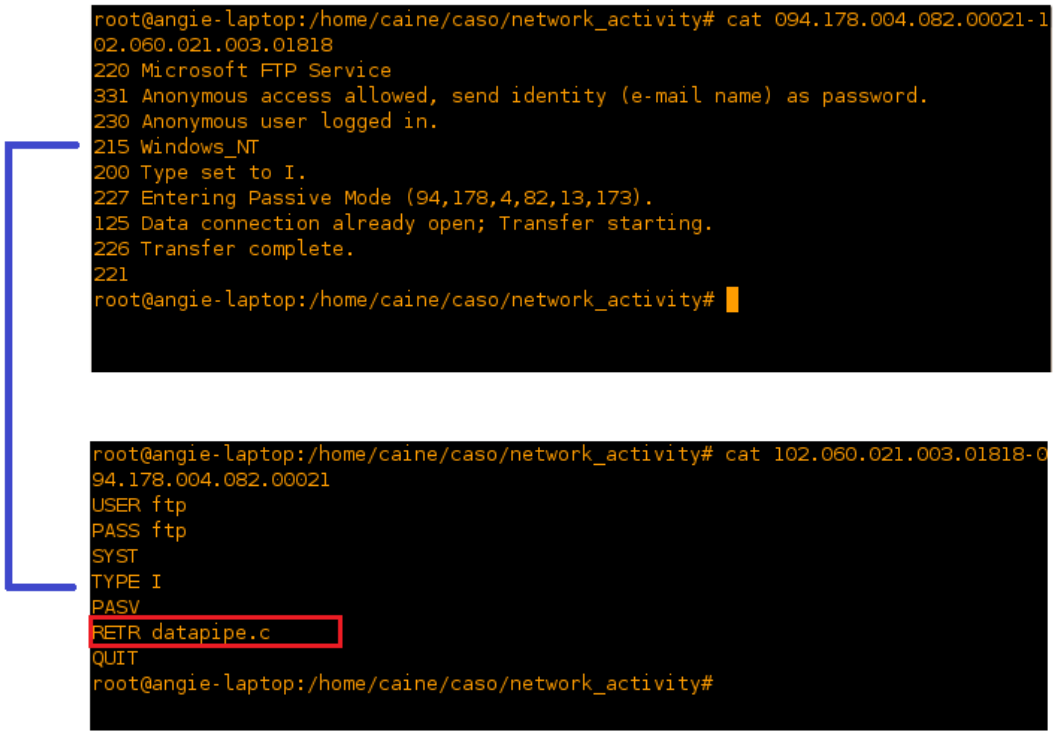


Figura 77: RETR Datapipe.c
Fuente: Entorno virtual Caine 2.0, Elaborado: Las autoras

Anexo 2

Reporte del análisis realizado por herramientas online

- **Virscan**

Tabla 17: Reporte del análisis de Virscan
Fuente: virscan.org.

Scanner	Engine Ver	Sig Ver	Sig Date	Scanresult	Time
a-squared	5.1.0.4	20130221080116	2013-02-21	Virus.Win32.Heur!IK	0.708
AhnLab V3	2013.02.19.02	2013.02.19	2013-02-19	-	2.707
AntiVir	8.2.10.202	7.11.50.58	2012-11-16	-	0.184
Antiy	2.0.18	2.0.18.	0002-18-00	-	0.174
Arcavir	2011	201302130333	2013-02-13	-	5.972
Authentium	5.1.1	201302201927	2013-02-20	W32/Swrort.D (Possible)	1.469
AVAST!	4.7.4	130220-1	2013-02-20	Win32:Patched-RH [Trj]	1.746
AVG	12.0.1794	2639/5620	2013-02-20	Win32/Heur	0.543
BitDefender	7.90123.8860038	7.45571	2013-02-20	Win32.Swrot.A	4.994
ClamAV	0.97.5	16707	2013-02-21	Trojan.MSF_Shellcode	1.555
Comodo	5.1	15321	2013-02-20	TrojWare.Win32.Rozena.A	2.436
CP Secure	1.3.0.5	2013.02.21	2013-02-21	-	2.019

Dr.Web	7.0.4.9250	2013.02.21	2013-02-21	Trojan.Swrort.1	15.463
F-Prot	4.6.2.117	20130201	2013-02-01	W32/Swrort.D	0.941
F-Secure	7.02.73807	2013.02.20.11	2013-02-20	Win32.Swrot.A [Aquarius]	2.509
Fortinet	4.3.392	16.549	2013-02-21	-	0.242
GData	22.8095	20130221	2013-02-21	Win32.Swrot.A [Engine:A]	6.950
Ikarus	T3.1.32.20.0	2013.02.20.83498	2013-02-20	Virus.Win32.Heur	7.604
JiangMin	16.0.100	2013.02.09	2013-02-09	Win32/PatchFile.jm	11.937
Kaspersky	5.5.10	2013.02.19	2013-02-19	-	0.312
KingSoft	2009.2.5.15	2013.2.21.9	2013-02-21	-	1.895
McAfee	5400.1158	6992	2013-02-20	Swrort.d	11.614
Microsoft	1.9203	2013.02.20	2013-02-20	Trojan:Win32/Swrort.A	4.747
NOD32	3.0.21	7951	2013-01-30	-	0.215
Norman	6.8.3	201208311030	2012-08-31	-	0.000
nProtect	20130220.01	13753685	2013-02-20	Win32.Swrot.A	5.721
Panda	9.05.01	2013.02.19	2013-02-19	-	0.591
Quick Heal	11.00	2013.02.20	2013-02-20	-	2.096
Rising	20.0	24.48.00.04	2013-02-04	Worm.Win32.Shekk.a	2.978

Sophos	3.39.0	4.85	2013-02-21	Mal/Swrort-C	6.115
Sunbelt	3.9.2558.2	15656	2013-02-20	Trojan.Win32.Swrort.B (v)	1.552
Symantec	1.3.0.24	20130219.003	2013-02-19	Trojan.Gen	1.326
The Hacker	6.8.0.0	v00195	2013-02-20	-	0.819
Trend Micro	9.500-1005	9.674.06	2013-01-22	-	0.261
VBA32	3.12.20.2	20130220.0754	2013-02-20	-	4.101
ViRobot	20130220	2013.02.20	2013-02-20	-	0.422
VirusBuster	5.5.2.13	15.0.354.0/10906293	2013-02-20	Win32.Swrort.Gen.2	0.418

- **Virustotal**

Tabla 18: Reporte del análisis de Virustotal
Fuente: www.virustotal.com

Antivirus	Result	Update
Agnitum	Win32.Swrort.Gen.2	20130220
AhnLab-V3	Trojan/Win32.Gen	20130220
AntiVir	TR/Crypt.XPACK.Gen	20130221
Antiy-AVL	-	20130220
Avast	-	20130221
AVG	Win32/Heur	20130221
BitDefender	Win32.Swrot.A	20130221
ByteHero	-	20130218
CAT-QuickHeal	-	20130220
ClamAV	Trojan.MSF_Shellcode	20130221
Commtouch	W32/Swrort.D	20130220
Comodo	TrojWare.Win32.Rozena.A	20130221
DrWeb	Trojan.Swrort.1	20130221
Emsisoft	Win32.Swrot.A (B)	20130221
eSafe	-	20130211
ESET-NOD32	a variant of Win32/Rozena.AG	20130221
F-Prot	W32/Swrort.D	20130220

Antivirus	Result	Update
F-Secure	Win32.Swrot.A	20130221
Fortinet	W32/Swrort.C!tr	20130221
GData	Win32.Swrot.A	20130221
Ikarus	Virus.Win32.Heur	20130221
Jiangmin	Win32/PatchFile.jm	20130220
K7AntiVirus	Virus	20130220
Kaspersky	HEUR:Trojan.Win32.Generic	20130221
Kingsoft	-	20130204
Malwarebytes	-	20130220
McAfee	Swrort.d	20130221
McAfee-GW-Edition	Swrort.d	20130220
Microsoft	Trojan:Win32/Swrort.A	20130221
MicroWorld-eScan	Win32.Swrot.A	20130221
NANO-Antivirus	Trojan.Win32.Swrort.bgcvrw	20130221
Norman	Swrort.S	20130220
nProtect	Win32.Swrot.A	20130220
Panda	Suspicious file	20130220
PCTools	Trojan.Gen	20130219
Rising	Worm.Win32.Shekk.a	20130205

Antivirus	Result	Update
Sophos	Mal/Swrort-C	20130221
SUPERAntiSpyware	-	20130221
Symantec	Trojan.Gen	20130221
TheHacker	-	20130221
TotalDefense	-	20130220
TrendMicro	TROJ_GEN.R47CDA9	20130221
TrendMicro-HouseCall	TROJ_GEN.R47CDA9	20130221
VBA32	-	20130220
VIPRE	Trojan.Win32.Swrort.B (v)	20130221
ViRobot	-	20130220

- **Anubi**

Tabla 19: Información General
Fuente:Anubi, Elaboración: Las autoras

Information about Anubis' invocation	
Time needed:	273 s
Report created:	02/02/13, 23:36:16 UTC
Termination reason:	All tracked processes have exited
Program version:	1.76.3886

Tabla 20: Innocent_a.exe
Fuente: Anubi, Elaboración: Las autoras

General information about this executable	
Analysis Reason:	Primary Analysis Subject
Filename:	innocent_s.exe
MD5:	0dc705345c94372d3a0b790dc4319d4e
SHA-1:	a5b22a84884b95350183b88636900c4a8766861b
File Size:	5521408
Command Line	"C:\innocent_s.exe"
Process-status at analysis end:	dead
Exit Code:	0

Load-time Dlls		
Module Name	**Base Address**	**Size**
C:\WINDOWS\system32\ntdll.dll	0x7C900000	0x000AF000
C:\WINDOWS\system32\kernel32.dll	0x7C800000	0x000F6000
C:\WINDOWS\system32\USER32.dll	0x7E410000	0x00091000
C:\WINDOWS\system32\GDI32.dll	0x77F10000	0x00049000
C:\WINDOWS\system32\ADVAPI32.dll	0x77DD0000	0x0009B000
C:\WINDOWS\system32\RPCRT4.dll	0x77E70000	0x00092000
C:\WINDOWS\system32\Secur32.dll	0x77FE0000	0x00011000

Run-time Dlls		
Module Name	**Base Address**	**Size**
C:\WINDOWS\system32\NETAPI32.dll	0x5B860000	0x00055000
C:\WINDOWS\system32\comctl32.dll	0x5D090000	0x0009A000
C:\WINDOWS\system32\faultrep.dll	0x69450000	0x00016000
C:\WINDOWS\system32\MSCTF.dll	0x74720000	0x0004C000
C:\WINDOWS\system32\WINSTA.dll	0x76360000	0x00010000
C:\WINDOWS\system32\USERENV.dll	0x769C0000	0x000B4000
C:\WINDOWS\system32\WTSAPI32.dll	0x76F50000	0x00008000
C:\WINDOWS\WinSxS\x86_Microsoft.Windows.Common-Controls_6595b64144ccf1df_6.0.2600.5512_x-	0x773D0000	0x00103000

ww_35d4ce83\comctl32.dll		
C:\WINDOWS\system32\SETUPAPI.dll	0x77920000	0x000F3000
C:\WINDOWS\system32\apphelp.dll	0x77B40000	0x00022000
C:\WINDOWS\system32\VERSION.dll	0x77C00000	0x00008000
C:\WINDOWS\system32\msvcrt.dll	0x77C10000	0x00058000
C:\WINDOWS\system32\SHLWAPI.dll	0x77F60000	0x00076000
C:\WINDOWS\system32\shell32.dll	0x7C9C0000	0x00817000

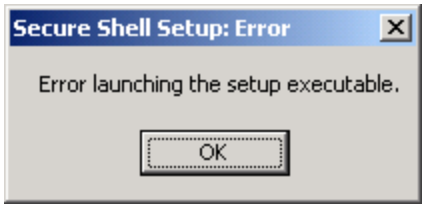
Popups			
Window Name	**Window Text**	**Screenshot**	**Number of Displayed Times**
Secure Shell Setup: Error	OK Error launching the setup executable.		1

Tabla21: Innocent_s.exe Registry Activities
Fuente: Anubi, Elaboración: Las autoras

Registry Keys Created:
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\SSH Communications Security
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\SSH Communications Security\SSH Secure Shell

Registry Values Modified:		
Key	**Name**	**New Value**
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\SSH Communications Security\SSH Secure Shell	SetupPath	C:

Registry Values Read:			
Key	**Name**	**Value**	**Times**
HKLM\SOFTWARE\Microsoft\CTF\SystemShared\	CUAS	0	1
HKLM\SYSTEM\Setup	OsLoaderPath	\	2
HKLM\SYSTEM\Setup	SystemPartition	\Device\HarddiskVolume1	2
HKLM\SYSTEM\Setup	SystemSetupInProgress	0	2
HKLM\SYSTEM\WPA\MediaCenter	Installed	0	1
HKLM\Software\Microsoft\PCHealth\ErrorReporting	AllOrNone	1	1
HKLM\Software\Microsoft\PCHealth\ErrorReporting	DoRepor	1	1

HKLM\Software\Microsoft\PCHealth\ErrorReporting	IncludeKernelFaults	1	1
HKLM\Software\Microsoft\PCHealth\ErrorReporting	IncludeMicrosoftApps	1	1
HKLM\Software\Microsoft\PCHealth\ErrorReporting	IncludeWindowsApps	1	1
HKLM\Software\Microsoft\PCHealth\ErrorReporting	ShowUI	1	1
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ AeDebug	Auto	1	1
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ AeDebug	Debugger	drwtsn32 -p %ld -e %ld -g	1
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ Windows	AppInit_DLLs		1
HKLM\Software\Microsoft\Windows\CurrentVersion	DevicePath	%SystemRoot%\inf	1
HKLM\Software\Microsoft\Windows\CurrentVersion	DriverCachePath	%SystemRoot%\Driver Cache	2
HKLM\Software\Microsoft\Windows\CurrentVersion	Setup LogLevel	0	2
HKLM\Software\Microsoft\Windows\CurrentVersion	Setup ServicePackCachePath	c:\windows\ServicePackFiles\ServicePackCache	2
HKLM\Software\Microsoft\Windows\CurrentVersion\ Setup	ServicePackSourcePath	D:\	2
HKLM\Software\Microsoft\Windows\CurrentVersion\ Setup	ServicePath	D:\	2
HKLM\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers	AuthenticodeEnabled	0	1
HKLM\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers	DefaultLevel	262144	1
HKLM\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers	PolicyScope	0	1
HKLM\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers	TransparentEnabled	1	2
HKLM\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers\0\Hashes\{349d35ab-37b5-462f-9b89-edd5fbde1328}	HashAlg	32771	1

Registry Values Read:			
Key	**Name**	**Value**	**Times**
HKLM\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers\0\Hashes\{349d35ab-37b5-462f-9b89-edd5fbde1328}	**ItemData**	**0x5eab304f957a49896a006c1c31154015**	**1**
HKLM\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers\0\Hashes\{349d35ab-37b5-462f-9b89-edd5fbde1328}	**ItemSize**	**779**	**1**
HKLM\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers\0\Hashes\{349d35ab-37b5-462f-9b89-edd5fbde1328}	**SaferFlags**	**0**	**1**
HKLM\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers\0\Hashes\{7fb9cd2e-3076-4df9-a57bb813f72dbb91}	**HashAlg**	**32771**	**1**

Registry Values Read:			
Key	**Name**	**Value**	**Times**
HKLM\System\CurrentControlSet\Control\Terminal Server	TSAppCompat	0	3
HKLM\System\CurrentControlSet\Control\Terminal Server	TSUserEnabled	0	1
HKLM\System\CurrentControlSet\Services\Tcpip\ Parameters	Domain		1
HKLM\System\CurrentControlSet\Services\Tcpip\ Parameters	Hostname	pc	1
HKLM\System\Setup	SystemSetupInProgress	0	2
HKLM\System\WPA\PnP	seed	1274198464	1
HKU\S-1-5-21-842925246-1425521274-308236825-500\Keyboard Layout\Toggle	Language Hotkey	1	4
HKU\S-1-5-21-842925246-1425521274-308236825-500\Keyboard Layout\Toggle	Layout Hotkey	2	4
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders	Cache	C:\Documents and Settings\Administrator\Local Settings\Temporary Internet Files	1
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders	Local Settings	%USERPROFILE%\Local Settings	1
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\Windows\Cur	Personal	%USERPROFILE%\My Documents	1

rentVersion\Explorer\User Shell Folders

Tabla22: Innocent_s.exe-File Activities
Fuente: Anubi, Elaboración: Las autoras

Files Created:
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\SSHPackage1.exe
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\b527_appcompat.txt

Files Read:
C:\WINDOWS\system32\winsock.dll
PIPE\lsarpc

Files Modified:
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\SSHPackage1.exe
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\b527_appcompat.txt
PIPE\lsarpc

File System Control Communication:		
File	Control Code	Times
C:\Program Files\Common Files\	0x00090028	1
PIPE\lsarpc	0x0011C017	6

Device Control Communication:		
File	Control Code	Times
\Device\KsecDD	0x00390008	1

Memory Mapped Files:
File Name
C:\WINDOWS\WinSxS\x86_Microsoft.Windows.Common-Controls_6595b64144ccf1df_6.0.2600.5512_x-ww_35d4ce83\comctl32.dll
C:\WINDOWS\WindowsShell.Manifest
C:\WINDOWS\system32\Apphelp.dll
C:\WINDOWS\system32\MSCTF.dll
C:\WINDOWS\system32\SETUPAPI.dll
C:\WINDOWS\system32\WINSTA.dll
C:\WINDOWS\system32\WTSAPI32.dll
C:\WINDOWS\system32\advapi32.dll

Memory Mapped Files:
File Name
C:\WINDOWS\system32\apphelp.dll

C:\WINDOWS\system32\comctl32.dll
C:\WINDOWS\system32\dwwin.exe
C:\WINDOWS\system32\faultrep.dll
C:\WINDOWS\system32\gdi32.dll
C:\WINDOWS\system32\imm32.dll
C:\WINDOWS\system32\kernel32.dll
C:\WINDOWS\system32\ntdll.dll
C:\WINDOWS\system32\ole32.dll
C:\WINDOWS\system32\oleaut32.dll
C:\WINDOWS\system32\shell32.dll
C:\WINDOWS\system32\user32.dll
C:\WINDOWS\system32\wininet.dll
C:\WINDOWS\system32\winsock.dll
C:\Windows\AppPatch\sysmain.sdb

Tabla23: Innocent_s.exe-Process Activities
Fuente: Anubi, Elaboración: Las autoras

Processes Created:	
Executable	**Command Line**
C:\WINDOWS\system32\dwwin.exe	
	C:\WINDOWS\system32\dwwin.exe -x -s 172

Remote Threads Created:
Affected Process
C:\WINDOWS\system32\dwwin.exe

Foreign Memory Regions Read:
Process: C:\WINDOWS\system32\dwwin.exe

Foreign Memory Regions Written:
Process: C:\WINDOWS\system32\dwwin.exe

Tabla24: Innocent_s.exe-Other Activities
Fuente: Anubi, Elaboración: Las autoras

Mutexes Created:
CTF.Asm.MutexDefaultS-1-5-21-842925246-1425521274-308236825-500
CTF.Compart.MutexDefaultS-1-5-21-842925246-1425521274-308236825-500
CTF.LBES.MutexDefaultS-1-5-21-842925246-1425521274-308236825-500
CTF.Layouts.MutexDefaultS-1-5-21-842925246-1425521274-308236825-500
CTF.TMD.MutexDefaultS-1-5-21-842925246-1425521274-308236825-500
CTF.TimListCache.FMPDefaultS-1-5-21-842925246-1425521274-308236825-500MUTEX.DefaultS-1-5-21-842925246-1425521274-308236825-500
MSCTF.Shared.MUTEX.IFG

Windows SEH exceptions:	
Description	**Times**
Exception 0xc0000005 (STATUS_ACCESS_VIOLATION) at 0xd80012	1
Exception 0xc0000005 (STATUS_ACCESS_VIOLATION) at 0xd80020	1

Tabla 25: dwwin.exe
Fuente: Anubi, Elaboración: Las autoras

General information about this executable	
Analysis Reason:	Started by innocent_s.exe
Filename:	dwwin.exe
MD5:	86042f6f6a5287eaf9379c91d0bf72b6
SHA-1:	532bf74e6aead7438aa7264d01759a065410ee68
File Size:	180224
Command Line:	C:\WINDOWS\system32\dwwin.exe -x -s 172
Process-status at analysis end:	dead
Exit Code:	0

Load-time Dlls Load-time Dlls		
Module Name	**Base Address**	**Size**
C:\WINDOWS\system32\ntdll.dll	0x7C900000	0x000AF000
C:\WINDOWS\system32\kernel32.dll	0x7C800000	0x000F6000
C:\WINDOWS\system32\ADVAPI32.DLL	0x77DD0000	0x0009B000
C:\WINDOWS\system32\RPCRT4.dll	0x77E70000	0x00092000
C:\WINDOWS\system32\Secur32.dll	0x77FE0000	0x00011000
C:\WINDOWS\system32\COMCTL32.DLL	0x5D090000	0x0009A000
C:\WINDOWS\system32\GDI32.dll	0x77F10000	0x00049000
C:\WINDOWS\system32\USER32.dll	0x7E410000	0x00091000
C:\WINDOWS\system32\OLEAUT32.DLL	0x77120000	0x0008B000
C:\WINDOWS\system32\msvcrt.dll	0x77C10000	0x00058000
C:\WINDOWS\system32\ole32.dll	0x774E0000	0x0013D000
C:\WINDOWS\system32\SHELL32.DLL	0x7C9C0000	0x00817000
C:\WINDOWS\system32\SHLWAPI.dll	0x77F60000	0x00076000

C:\WINDOWS\system32\URLMON.DLL	0x7E1E0000	0x000A2000
C:\WINDOWS\system32\VERSION.dll	0x77C00000	0x00008000
C:\WINDOWS\system32\WININET.DLL	0x771B0000	0x000AA000
C:\WINDOWS\system32\CRYPT32.dll	0x77A80000	0x00095000
C:\WINDOWS\system32\MSASN1.dll	0x77B20000	0x00012000
C:\WINDOWS\system32\ShimEng.dll	0x5CB70000	0x00026000
C:\WINDOWS\AppPatch\AcGenral.DLL	0x6F880000	0x001CA000
C:\WINDOWS\system32\WINMM.dll	0x76B40000	0x0002D000
C:\WINDOWS\system32\MSACM32.dll	0x77BE0000	0x00015000
C:\WINDOWS\system32\USERENV.dll	0x769C0000	0x000B4000
C:\WINDOWS\system32\UxTheme.dll	0x5AD70000	0x00038000
C:\WINDOWS\WinSxS\x86_Microsoft.Windows.Common-Controls_6595b64144ccf1df_6.0.2600.5512_x-ww_35d4ce83\comctl32.dll	0x773D0000	0x00103000

Run-time Dlls

Module Name	Base Address	Size
C:\WINDOWS\system32\1033\dwintl.dll	0x314C0000	0x0000C000
C:\WINDOWS\system32\NETAPI32.dll	0x5B860000	0x00055000
C:\WINDOWS\system32\WS2HELP.dll	0x71AA0000	0x00008000
C:\WINDOWS\system32\WS2_32.dll	0x71AB0000	0x00017000
C:\WINDOWS\system32\sensapi.dll	0x722B0000	0x00005000
C:\WINDOWS\system32\MSCTF.dll	0x74720000	0x0004C000
C:\WINDOWS\system32\riched20.dll	0x74E30000	0x0006D000
C:\WINDOWS\system32\imm32.dll	0x76390000	0x0001D000
C:\WINDOWS\system32\shfolder.dll	0x76780000	0x00009000
C:\WINDOWS\system32\PSAPI.DLL	0x76BF0000	0x0000B000
C:\WINDOWS\system32\rtutils.dll	0x76E80000	0x0000E000
C:\WINDOWS\system32\rasman.dll	C 0x76E90000	0x00012000
C:\WINDOWS\system32\TAPI32.dll	0x76EB0000	0x0002F000
C:\WINDOWS\system32\RASAPI32.DLL	0x76EE0000	0x0003C000

Popups

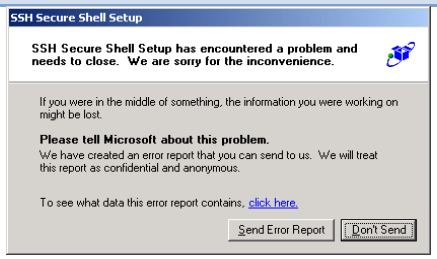
WindowName	Window Text	Screenshot	Number of Displayed Times
SSH Secure Shell Setup	&Don't Send SSH Secure Shell Setup has encountered a problem and needs to close. We are sorry for the inconvenience. SSH Secure Shell Setup has encountered a problem and needs to close. We are sorry for the inconvenience. If you were in the middle of something, the information you were working on might be lost. Please tell Microsoft about this problem. We have created an error report that you can send to us. We will treat this report as confidential and anonymous. To see what data this error report contains, Details &Send Error Report		1

Tabla 26: dwwin.exe-Registry Activies
Fuente: Anubi, Elaboración: Las autoras

RegistryValuesModified:		
Key	**Name**	**New Value**
HKLM\SYSTEM\CURRENTCONTROLSET\HARDWARE PROFILES\CURRENT\Software\Microsoft\windows\CurrentVersion\InternetSettings	ProxyEnable	0
HKLM\Software\Microsoft\Windows\CurrentVersion\Explorer\ShellFolders	CommonAppData	C:\Documents and Settings\All Users\ Application Data
HKLM\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Cache\Paths	Directory	C:\Documents and Settings\Administrator \Local Settings\Temporary Internet Files\Content.IE5
HKLM\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Cache\Paths	Paths	4
HKLM\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Cache\Paths\Path1	CacheLimit	40852
HKLM\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Cache\Paths\Path1	CachePath	C:\Documents and Settings\Administrator \Local Settings\Temporary Internet Files\Content.IE5\Cache1
HKLM\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Cache\Paths\Path2	CacheLimit	40852
HKLM\Software\Microsoft\Windows\CurrentVersion\Internet Settings\ Cache\Paths\Path2	CachePath	C:\Documents and Settings\Administrator \Local Settings\Temporary Internet Files\Content.IE5\Cache2
HKLM\Software\Microsoft\Windows\CurrentVersion\Internet Settings\ Cache\Paths\Path3	CacheLimit	40852
HKLM\Software\Microsoft\Windows\CurrentVersion\Internet Settings\ Cache\Paths\Path3	CachePath	C:\Documents and Settings\Administrator \Local Settings\Temporary Internet Files\ Content.IE5\Cache3
HKLM\Software\Microsoft\Windows\CurrentVersion\Internet Settings\ Cache\Paths\Path4	CacheLimit	40852
HKLM\Software\Microsoft\Windows\CurrentVersion\Internet Settings\ Cache\Paths\Path4	CachePath	C:\Documents and Settings\Administrator \Local Settings\Temporary Internet Files\Content.IE5\Cache4
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders	AppData	C:\Documents and Settings\Administrator\Application Data
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders	Cache	C:\Documents and Settings\Administrator\ Local Settings\Temporary InternetFiles
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders	Cookies	C:\Documents and Settings\Administrator\ Cookies
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders	History	C:\Documents and Settings\Administrator\Local Settings\History

HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders	Personal	C:\Documents and Settings\Administrator\My Documents
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\windows\CurrentVersion\Internet Settings	MigrateProxy	1
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\windows\CurrentVersion\Internet Settings	ProxyEnable	0
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\windows\CurrentVersion\Internet Settings\Connections	SavedLegacySettings	0x3c0000001600000001000000000000000 0000000000000000040000000000

Registry Values Read:			
Key	**Name**	**Value**	**Times**
HKLM\SOFTWARE\Microsoft\CTF\SystemShared	CUAS	0	1
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings	UrlEncoding	0x00000000	2
HKLM\SYSTEM\CurrentControlSet\Control\SessionManager	CriticalSectionTimeout	2592000	1
HKLM\SYSTEM\Setup	SystemSetupInProgress	0	1
HKLM\SYSTEM\WPA\MediaCenter	Installed	0	1
HKLM\Software\Microsoft\AudioCompressionManager\ DriverCache\msacm.iac2	aFormatTagCache	0x0100000010000000020400001400000	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.iac2	cFilterTags	0	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.iac2	cFormatTags	2	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.iac2	fdwSupport	1	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.imaadpcm	aFormatTagCache	0x0100000010000000110000001400000	1
HKLM\Software\Microsoft\AudioCompressionManager\ DriverCache\msacm.imaadpcm	cFilterTags	0	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.imaadpcm	cFormatTags	2	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.imaadpcm	fdwSupport	1	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.l3acm	aFormatTagCache	0x0100000010000000550000001e000000	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.l3acm	cFilterTags	0	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.l3acm	cFormatTags	2	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.l3acm	fdwSupport	1	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msadpcm	aFormatTagCache	0x0100000010000000020000003200000	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msadpcm	cFilterTags	0	1
HKLM\Software\Microsoft\AudioCompressionManager\ DriverCache\msacm.msadpcm	**cFormatTags**	**2**	**1**
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msadpcm	fdwSupport	1	1

HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msaudio1	aFormatTagCache	0x01000000120000006001000016000000661010000 1c000000	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msaudio1	cFilterTags	0	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msaudio1	cFormatTags	3	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msaudio1	fdwSupport	1	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msg711	aFormatTagCache	0x01000000100000000600000012000000007000000 12000000	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msg711	cFilterTags	0	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msg711	cFormatTags	3	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msg711	fdwSupport	1	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msg723	aFormatTagCache	0x01000000100000000420000001c000000	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msg723	cFilterTags	0	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msg723	cFormatTags	2	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msg723	fdwSupport	1	1

Registry Values Read:			
Key	**Name**	**Value**	**Times**
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msgsm610	aFormatTagCache	0x01000000100000003100000014000000	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msgsm610	cFilterTags	0	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msgsm610	cFormatTags	2	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.msgsm610	fdwSupport	1	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.sl_anet	aFormatTagCache	0x01000000100000003001000016000000	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.sl_anet	cFilterTags	0	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.sl_anet	cFormatTags	2	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.sl_anet	fdwSupport	1	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.trspch	aFormatTagCache	0x01000000100000002200000032000000	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.trspch	cFilterTags	0	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.trspch	cFormatTags	2	1
HKLM\Software\Microsoft\AudioCompressionManager\DriverCache\msacm.trspch	fdwSupport	1	1
HKLM\Software\Microsoft\Internet Explorer\Main\FeatureControl\FEATURE_BEHAVIORS	*	1	1
HKLM\Software\Microsoft\Internet Explorer\Main\FeatureControl\FEATURE_DISABLE_MK_PROTOCOL	*	1	1
HKLM\Software\Microsoft\Tracing	EnableConsoleTracing	0	1

HKLM\Software\Microsoft\Tracing\RASAPI32	ConsoleTracingMask	4294901760	2

HKLM\Software\Microsoft\Tracing\RASAPI32	**EnableConsoleTracing**	**0**	**2**
HKLM\Software\Microsoft\Tracing\RASAPI32	EnableFileTracing	0	2
HKLM\Software\Microsoft\Tracing\RASAPI32	FileDirectory	%windir%\tracing	4
HKLM\Software\Microsoft\Tracing\RASAPI32	FileTracingMask	4294901760	2
HKLM\Software\Microsoft\Tracing\RASAPI32	MaxFileSize	1048576	2
HKLM\Software\Microsoft\Windows NT\CurrentVersion	DigitalProductId	0xa4000000030000 0037363438372d36 343302d313435373 233362d32333833	1
HKLM\Software\Microsoft\Windows NT\CurrentVersion\AeDebug	Debugger	drwtsn32 -p %ld -e %ld -g	4
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	midimapper		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	msacm.iac2		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	msacm.imaadpcmi	maadp32.acm	3
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	msacm.l3acm		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	msacm.msadpcm	msadp32.acm	3
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	msacm.msaudio1		3
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	msacm.msg711	msg711.acm	3
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	msacm.msg723		3
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	msacm.msgsm610		3
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	msacm.sl_anet		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	msacm.trspch		3
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	vidc.I420		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	vidc.M261		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	vidc.M263		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	vidc.cvid		2

Registry Values Read:			
Key	**Name**	**Value**	**Times**
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	vidc.iv31		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	vidc.iv32		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	vidc.iv41		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Drivers32	vidc.iv50		1

HKLM\Software\Microsoft\Windows NT\CurrentVersion\ Drivers32	vidc.iyuv		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ Drivers32	vidc.mrle		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ Drivers32	vidc.msvc		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ Drivers32	vidc.uyvy		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ Drivers32	vidc.yuy2		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ Drivers32	vidc.yvu9		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ Drivers32	vidc.yvyu		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ Drivers32	wavemapper		2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ ProfileList	AllUsersProfile	All Users	2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ ProfileList	DefaultUserProfile	Default User	2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ ProfileList	ProfilesDirectory	%SystemDrive%\Documents and Settings	4
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ProfileList\ S-1-5-21-842925246-1425521274-308236825-500	ProfileImagePath	%SystemDrive%\Documents and Settings\ Administrator	2
HKLM\Software\Microsoft\Windows NT\CurrentVersion\ Windows	AppInit_DLLs		1
HKLM\Software\Microsoft\Windows\CurrentVersion	CommonFilesDir	C:\Program Files\Common Files	3
HKLM\Software\Microsoft\Windows\CurrentVersion	ProgramFilesDir	C:\Program Files	3
HKLM\Software\Microsoft\Windows\CurrentVersion\ Explorer\User Shell Folders	Common AppData	%ALLUSERSPROFILE%\Application Data	1

HKLM\Software\Policies\Microsoft\Windows\Safer\ CodeIdentifiers	**TransparentEnabled**	**1**	**1**
HKLM\System\CurrentControlSet\Control\ComputerNameActiveComputerName	ComputerName	PC	5
HKLM\System\CurrentControlSet\Control\ MediaProperties\PrivateProperties\Joystick\Winmm	wheel	1	1
HKLM\System\CurrentControlSet\Control\ProductOptions	ProductType	WinNT	1
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	ComSpec	%SystemRoot%\system32\cmd.exe	4
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	FP_NO_HOST_CHECK	NO	4
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	NUMBER_OF_PROCES	1	4
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	OS	Windows_NT	4
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;..JSE;.WSF;.WSH	4
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	PROCESSOR_ARCHIT	X86	4
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	PROCESSOR_IDENTIF	xIE86R Family 6 Model 3 Stepping 3, GenuineIntel	4

HKLM\System\CurrentControlSet\Control\Session Manager\Environment	PROCESSOR_LEVEL	6	4
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	PROCESSOR_REVISIO0N	0303	4
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	Path	SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem	4
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	TEMP	%SystemRoot%\TEMP	4
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	TMP	%SystemRoot%\TEMP	4
HKLM\System\CurrentControlSet\Control\Session Manager\Environment	windir	%SystemRoot%	4
HKLM\System\CurrentControlSet\Control\Terminal Server	TSAppCompat	0	3
HKLM\System\CurrentControlSet\Control\Terminal Server	TSUserEnabled	0	1

RegistryValuesRead:			
Key	**Name**	**Value**	**Times**
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\History	PerUserItem	1	1
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\windows\CurrentVersion\Internet Settings	MigrateProxy	1	1
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\windows\CurrentVersion\Internet Settings	ProxyEnable	0	1
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\windows\CurrentVersion\Internet Settings\Connections	DefaultConnectionSetti	gxs3c0000000300000010000000000000000000000000000000000000000000000400000000	2
HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\windows\CurrentVersion\Internet Settings\Connections	SavedLegacySettings	0x3c00000015000000010000000000000000000000000000000000000000000000400000000	4
HKU\S-1-5-21-842925246-1425521274-308236825-500\VolatileEnvironment	APPDATA	C:\Documents and Settings\Administrator\Application Data	4
HKU\S-1-5-21-842925246-1425521274-308236825-500\VolatileEnvironment	CLIENTNAME	Console	4
HKU\S-1-5-21-842925246-1425521274-308236825-500\VolatileEnvironment	HOMEDRIVE	C:	4
HKU\S-1-5-21-842925246-1425521274-308236825-500\VolatileEnvironment	HOMEPATH	\Documents and Settings\Administrator	4
HKU\S-1-5-21-842925246-1425521274-308236825-500\VolatileEnvironment	HOMESHARE		4
HKU\S-1-5-21-842925246-1425521274-308236825-500\VolatileEnvironment	LOGONSERVER	\\PC	4
HKU\S-1-5-21-842925246-1425521274-308236825-500\VolatileEnvironment	SESSIONNAME	Console	4

MonitoredRegistryKeys:			
Key Name	**Watchsubtree**	**NotifyFilter**	**Count**
HKLM\Software\Microsoft\Tracing\RASAPI32	0	Attributes Change,ValueChange,Security Descriptor Change	2

Tabla27: dwwin.exe-File Activities
Fuente: Anubi, Elaboración: Las autoras

Files Deleted:
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\6F918.dmp **C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\b527_appcompat.txt**
Files Created:
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\6F918.dmp

Files Read:
C:\WINDOWS\win.ini **C:\innocent_s.exe** **PIPE\lsarpc** **c:\autoexec.bat**

Files Modified:
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\6F918.dmp **PIPE\lsarpc**

File System Control Communication:		
File	**Control Code**	Times
C:\WINDOWS\system32	0x00090028	1
PIPE\lsarpc	0x0011C017	16

Device Control Communication:		
File	**Control Code**	**Times**
\Device\KsecDD	0x00390008	8

MemoryMapped Files:
File Name
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\6F918.dmp **C:\WINDOWS\AppPatch\AcGenral.DLL** **C:\WINDOWS\WinSxS\x86_Microsoft.Windows.Common-Controls_6595b64144ccf1df_6.0.2600.5512_x-ww_35d4ce83\comctl32.dll** **C:\WINDOWS\WindowsShell.Manifest**

C:\WINDOWS\system32\1033\dwintl.dll
C:\WINDOWS\system32\ADVAPI32.dll
C:\WINDOWS\system32\Apphelp.dll
C:\WINDOWS\system32\COMCTL32.DLL
C:\WINDOWS\system32\GDI32.dll
C:\WINDOWS\system32\MSACM32.dll
C:\WINDOWS\system32\MSCTF.dll
C:\WINDOWS\system32\NETAPI32.dll
C:\WINDOWS\system32\PSAPI.DLL
C:\WINDOWS\system32\RASAPI32.DLL
C:\WINDOWS\system32\RPCRT4.dll
C:\WINDOWS\system32\SETUPAPI.dll
C:\WINDOWS\system32\SHELL32.DLL
C:\WINDOWS\system32\SHLWAPI.dll
C:\WINDOWS\system32\Secur32.dll
C:\WINDOWS\system32\ShimEng.dll
C:\WINDOWS\system32\TAPI32.dll
C:\WINDOWS\system32\URLMON.DLL
C:\WINDOWS\system32\USER32.dll
C:\WINDOWS\system32\USERENV.dll
C:\WINDOWS\system32\UxTheme.dll
C:\WINDOWS\system32\VERSION.dll
C:\WINDOWS\system32\WININET.DLL
C:\WINDOWS\system32\WINMM.dll
C:\WINDOWS\system32\WINSTA.dll
C:\WINDOWS\system32\WS2HELP.dll
C:\WINDOWS\system32\WS2_32.dll
C:\WINDOWS\system32\WTSAPI32.dll
C:\WINDOWS\system32\comctl32.dll
C:\WINDOWS\system32\faultrep.dll
C:\WINDOWS\system32\imm32.dll
C:\WINDOWS\system32\kernel32.dll
C:\WINDOWS\system32\msvcrt.dll
C:\WINDOWS\system32\ntdll.dll
C:\WINDOWS\system32\rasman.dll
C:\WINDOWS\system32\riched20.dll
C:\WINDOWS\system32\rtutils.dll
C:\WINDOWS\system32\sensapi.dll
C:\WINDOWS\system32\shfolder.dll
C:\Windows\AppPatch\sysmain.sdb
C:\innocent_s.exe

Tabla 28: dwwin..Exe-ProcessActivities
Fuente: Anubi, Elaboración: Las autoras

ForeignMemoryRegionsRead:
Process: C:\innocent_s.exe

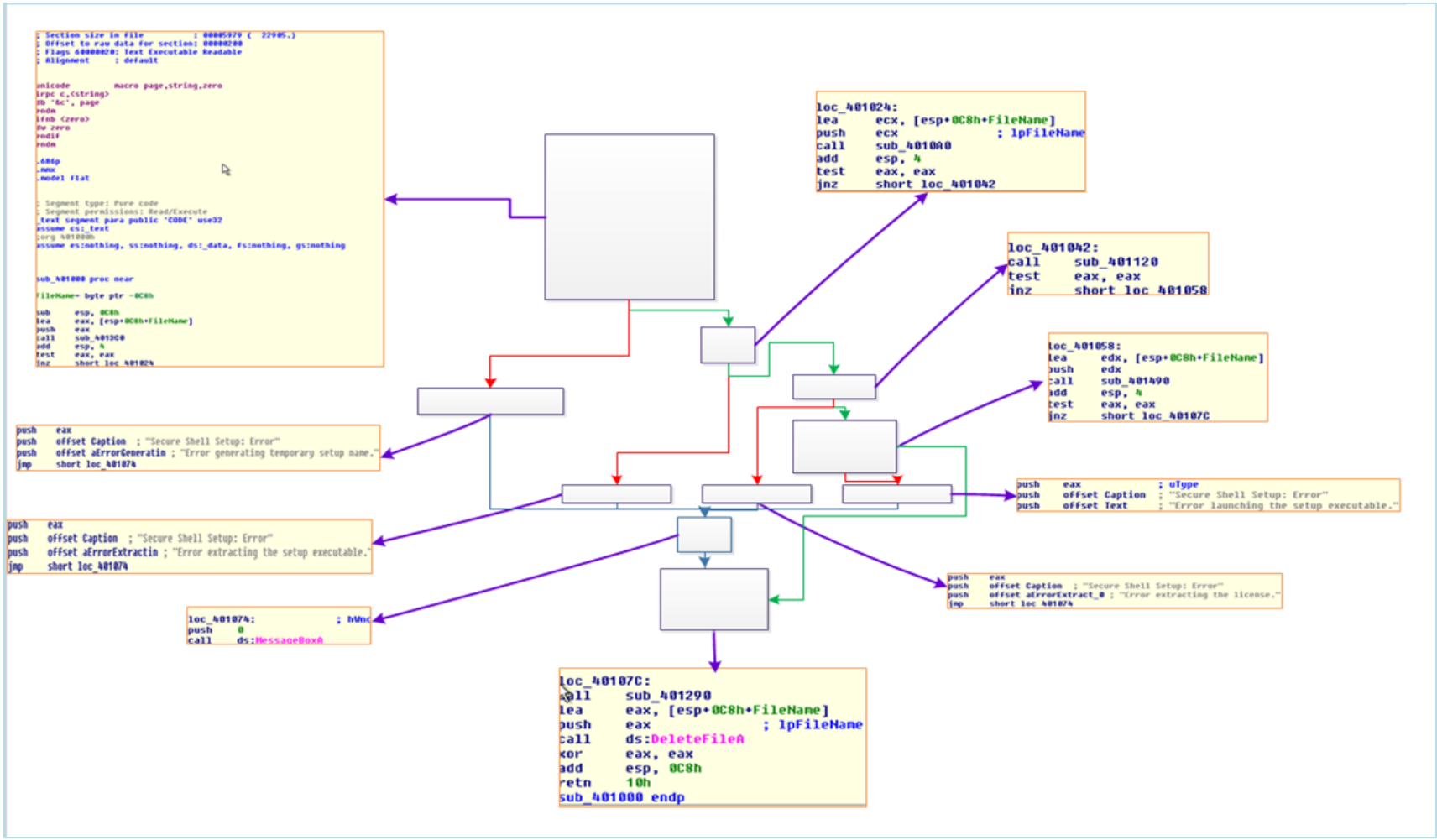


Figura 78: Diagramas de IDA
Fuente: Programa de IDA, Elaboración: Las autoras

GLOSARIO

TIC: Las tecnologías de la información y la comunicación, agrupan los elementos y las técnicas usados en el tratamiento y la transmisión de la Información.

IT: Tecnologías de la Información (TI) es la rama de ingeniería que se ocupa del uso de la informática y las telecomunicaciones para almacenar, recuperar y transmitir información.

Hashes: Usan algoritmos criptográficos para crear un mensaje resumido de los datos y representarlos como una pieza relativamente pequeña de datos

Malware: son programas que contiene código malicioso para infectar el sistema de nuestra computadora.

RFC 1244: La RFC 1244 define Política de Seguridad como: "una declaración de intenciones de alto nivel que cubre la seguridad de los sistemas informáticos y que proporciona las bases para definir y delimitar responsabilidades para las diversas actuaciones técnicas y organizativas que se requerirán.

NIST: por sus siglas en inglés, National Institute of Standards and Technologies una agencia de la Administración de Tecnología del Departamento de Comercio de los Estados Unidos.

CompTIA: Computing Technology Industry Association es una organización sin fines de lucro dedicada a la certificación de profesionales para la industria de tecnologías de información.

EC-COUNCIL: El Consejo Internacional de Consultores de Comercio Electrónico s conocido principalmente como un organismo de certificación profesional.

Peritos forenses: es un profesional dotado de conocimientos especializados y reconocidos, a través de sus estudios superiores, que suministra información u opinión fundada a los tribunales de justicia sobre los puntos litigiosos que son materia de su dictamen. Existen dos tipos de peritos, los nombrados judicialmente y los propuestos por una o ambas partes (y luego aceptados por el juez o el fiscal), y ambos ejercen la misma influencia en el juicio.

TCP/IP: es un conjunto de protocolos diseñados para las redes de área amplia WAN. El protocolo TCP/IP está conformado por un modelo de cuarto capas: interface de red, red, transporte y aplicación.

Hacking Ético: es una nueva ética surgida de y aplicada a las comunidades virtuales o cibercomunidades, aunque no exclusivamente, que se resumen en el acceso libre a la información y en que la informática puede mejorar la calidad de vida de las personas-- han constituido la base de la mayor parte de definiciones que se han elaborado posteriormente.

ANSI: ANSI es una organización privada sin fines de lucro, que permite la estandarización de productos, servicios, procesos, sistemas y personal en Estados Unidos. Además, ANSI se coordina con estándares internacionales

para asegurar que los productos estadounidenses puedan ser usados a nivel mundial.

Caine: Computer Aided Investigative Environment, es una distribución live CD para realizar análisis forense.

Open Source: es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de poder acceder al código, que a las cuestiones éticas y morales las cuales se destacan en el software libre.

Datapipe: DataPipe es una útil aplicación que realiza acciones de búsqueda y reemplazo dentro de bases de datos específicas.

Telnet: El término TELNET se refiere a la conexión remota a un equipo de modo terminal remoto con una máquina en la que estamos autorizados.

SSH: es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a través de una red. SSH nos permite copiar datos de forma segura (tanto ficheros sueltos como simular sesiones FTP cifradas), gestionar claves RSA para no escribir claves al conectar a los dispositivos y pasar los datos de cualquier otra aplicación por un canal seguro tunelizado mediante SSH.

FTP: es un protocolo de red para la transferencia entre sistemas conectados a una red TCP (Transmisión Control Protocol), basado en la arquitectura cliente-servidor.

Exploit: Un Exploit es un programa o código que "explota" una vulnerabilidad del sistema o de parte de él para aprovechar esta deficiencia en beneficio del creador del mismo.

John the Ripper: es un programa multiplataforma de criptografía que aplica fuerza bruta o utilizando diccionarios para descifrar contraseñas. Es capaz de romper varios algoritmos de cifrado o hash.

Buffer: es un espacio de memoria, en el que se almacenan datos para evitar que el programa o recurso que los requiere, ya sea hardware o software, se quede sin datos durante una transferencia.

Dll:" es un archivo que contiene funciones que se pueden llamar desde aplicaciones u otras Dll. Los desarrolladores utilizan las Dll para poder reciclar el código y aislar las diferentes tareas.

Md5sum: es un programa originario de los sistemas Unix, la función de hash devuelve un valor que es prácticamente único para cada archivo, con la particularidad que una pequeña variación en el archivo provoca una salida totalmente distinta, lo que ayuda a detectar si el archivo sufrió alguna variación. Es una herramienta de seguridad que sirve para verificar la integridad de los datos.

BIBLIOGRAFÍA

[1] Estrada de Flete A., Sorando de Silva H., Graterol N., Rodríguez M., Planificación de la investigación cuantitativa, http://www.monografias.com/trabajos61/planificacion-investigacion-cuantitativa/planificacion-investigacion-cuantitativa2.shtml

[2] Softpedia, Descripción de DataPipe, http://www.softpedia.es/programa-DataPipe-8368.html

[3] Wikipedia, Secure Shell, http://es.wikipedia.org/wiki/Secure_Shell

[4] Access Data, FTK, http://www.accessdata.com/products/digital-forensics/ftk

[5] Segu-info, Exploit, http://www.segu-info.com.ar/malware/exploit.htm

[6] Wikipedia, Telnet, http://es.wikipedia.org/wiki/Telnet

[7] Wordpress, Jhon the Ripper, http://cursoredlocal.wordpress.com/2011/01/30/john-the-ripper-windows-v1-7-6/

[8] Microsoft Technet, FTP, http://technet.microsoft.com/es-ec/library/bb490910.aspx

[9] Sve & Julian, Que son las DLL, http://www.svetlian.com/dll/articulos_descripcion_dll.htm

[10] Forensic control limited, Free computer forensic tolls, http://forensiccontrol.com/resources/free-software/

[11] Wikipedia, Análisis estático de Software, http://es.wikipedia.org/wiki/An%C3%A1lisis_est%C3%A1tico_de_software

[12] Wikipedia, Análisis dinámico de Software, http://es.wikipedia.org/wiki/An%C3%A1lisis_din%C3%A1mico_de_software

[13] Yago Jesus, SecuritybyDefault.com, Reversing Malware Tales: Safe Debugging, http://www.securitybydefault.com/2012/06/reversing-malware-tales-safe-debugging.html

[14] Daboweb Team, Herramientas de interpretación de capturas de red, http://www.daboweb.com/2012/06/15/herramientas-para-la-interpretacion-de-capturas-de-red-910-parte-3/

[15] InfoSpyware, Que son los Malware, http://www.infospyware.com/articulos/que-son-los-malwares/

[16] ElHacker.net, Compilación herramientas análisis y desinfección malware, http://blog.elhacker.net/2013/02/aio-2013-compilacion-herramientas-desinfectar-malware-monitorizar-securizar-windows.html

[17] Informaticaforense.org, Informática forense en un mundo digital, http://www.informaticaforense.org/2011/08/29/informatica-forense-en-un-mundo-digital/

[18] Wikipedia, OllyDbg, http://es.wikipedia.org/wiki/OllyDbg

[19] Microsoft Technet, Process Explorer, http://technet.microsoft.com/en-us/sysinternals/bb896653.aspx

[20] Anubis, Anubis: Analyzing Unknown Binaries, http://anubis.iseclab.org/

[21] Set Staff, Definición de IDA, http://www.set-ezine.org/index.php?num=33&art=17

[22] Gallardo Luis, FTP activo vs FTP pasivo, http://lgallardo.com/2009/06/23/ftp-activo-vs-ftp-pasivoftp-active-vs-passive/

[23] Redes y Seguridad, Clasificación de Software Malicioso, http://www.redesyseguridad.es/clasificacion-del-software-malicioso/

[24] Servicios de Información Tecnológica (SIT), Definición de tipos de virus, http://sit.gda.itesm.mx/definiciones.html

[25] López O., Amaya H., León R., Acosta B., INFORMÁTICA FORENSE : GENERALIDADES, ASPECTOS TÉCNICOS Y HERRAMIENTAS, http://www.criminalistaenred.com.ar/Informatica_F.html

[26] El Software Libre, Tutorial de Linux caine, http://software-libre-if.blogspot.com/p/tutorial-de-linux-caine.html

[27] Wikipedia, Computo Forense, http://es.wikipedia.org/wiki/C%C3%B3mputo_forense

[28] Segu.Info News, Información y Herramientas de Análisis Forense, http://blog.segu-info.com.ar/2011/02/informacion-y-herramientas-de-analisis.html#axzz2OMj4y8rp