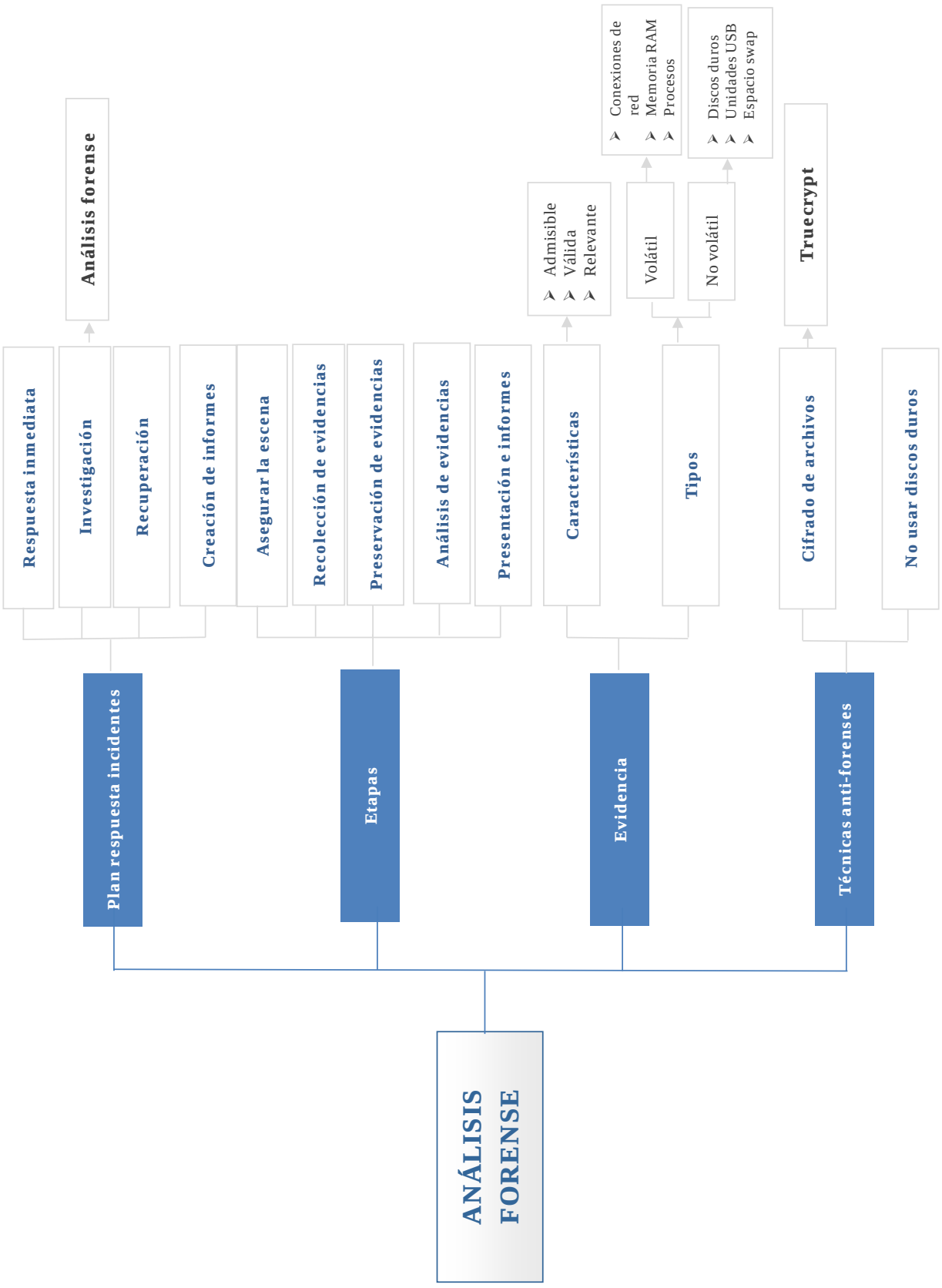
Análisis forense

TEMA 12

Esquema


ANÁLISIS FORENSE
Plan respuesta incidentes
Etapas
Evidencia
Técnicas anti-forenses
Respuesta inmediata
Investigación
Recuperación
Creación de informes
Asegurar la escena
Recolección de evidencias
Preservación de evidencias
Análisis de evidencias
Presentación e informes
Características
Tipos
Cifrado de archivos
No usar discos duros
Análisis forense
Admisible
Válida
Relevante
Volátil
No volátil
Truecrypt
Conexiones de red
Memoria RAM
Procesos
Discos duros
Unidades USB
Espacio swap

Ideas clave

12.1. ¿Cómo estudiar este tema?

El estudio de este tema se realiza a través de los contenidos desarrollados en las **Ideas clave** expuestas a continuación.

En este último tema de la asignatura abordaremos otra de las grandes áreas de la seguridad informática: **el análisis forense**. Comenzaremos definiendo claramente el concepto y delimitando su ámbito de actuación.

Luego analizaremos uno de sus etapas más importantes: **la recolección de evidencias**, donde son necesarias una serie de destrezas y conocimientos, en ocasiones, profundos del sistema operativo atacado. Por supuesto, posteriormente estudiaremos el **proceso de análisis de estas evidencias y las técnicas anti-forenses** más comunes que puedes encontrarte en el ejercicio de tu profesión.

Finalmente, analizaremos un **caso práctico**, en el que entrarán en juego buena parte de los conceptos estudiados en el tema.

12.2. Introducción

Si se trata de ubicar al alumno en la importancia de este tema, no hay más que leer la cantidad de delitos informáticos que ocurren todos los días, y esos delitos, requieren de una investigación forense.

La informática forense tiene tres objetivos principales: el resarcimiento de los daños causados por un ataque informático o criminal, la persecución y procesamiento judicial de los atacantes o criminales y la creación y la aplicación de medidas preventivas para casos similares. La principal meta es la recolección de evidencias y los datos analizables en un proceso forense, es decir, los medios en los que se almacenan los datos o los ficheros donde se registra la actividad del dispositivo a analizar: discos duros, direcciones MAC, documentación relacionada con el caso, logs del sistema, etc.

A pesar de todas las barreras de seguridad que se utilizan para salvaguardar los activos de información, muchas de las cuales hemos ido analizando a lo largo de la asignatura, los incidentes de seguridad se siguen produciendo, por lo que estar preparado para reaccionar ante un ataque es fundamental.

En este tema final cambiaremos, por tanto, nuestro enfoque, estudiando qué podemos hacer una vez se ha producido un ataque. Esta fase se suele denominar **respuesta a incidentes**, y una de sus etapas más importantes consiste en la investigación del incidente para saber por qué se produjo la intrusión, quién la llevó a cabo y a qué sistemas afectó. Esta investigación se conoce como **análisis forense** y en este tema analizaremos sus características más destacadas.

Un plan de respuesta a incidentes ayuda a estar preparado y a saber cómo se debe actuar una vez se haya identificado un ataque. Constituye un punto clave dentro de los planes de seguridad de la información, ya que mientras que la detección del incidente es el punto que afecta a la seguridad del sistema, la respuesta define cómo debe reaccionar el equipo de seguridad para minimizar los daños y recuperar los sistemas, todo ello garantizando la integridad del conjunto.

El ***plan de respuesta a incidentes*** suele dividirse en **varias fases**, entre las que destacan:

1. **Respuesta inmediata**, para evitar males mayores, como reconfigurar automáticamente las reglas de los cortafuegos o inyectar paquetes de RESET sobre conexiones establecidas.
2. **Investigación**, para recolectar evidencias del ataque que permitan reconstruirlo con la mayor fidelidad posible.
3. **Recuperación**, para volver a la normalidad en el menor tiempo posible y evitar que el incidente se repita de nuevo.
4. **Creación de informes**, para documentar los datos sobre los incidentes y que sirvan como base de conocimiento con posterioridad, para posibles puntos de mejora y como información para todos los integrantes de la organización. De manera adicional, se hacen necesarios los informes por posibles responsabilidades legales que pudieran derivarse.

¿Qué es un análisis forense?

El análisis forense de sistemas **pretende averiguar lo ocurrido durante una intrusión**. Busca dar respuesta a los interrogantes que normalmente envuelven a todo incidente: quién realizó el ataque, qué activos de información se vieron afectados y en qué grado, cuándo tuvo lugar, dónde se originó y contra qué blancos se extendió, cómo fue llevado a cabo y por qué.

Como si de un delito tradicional se tratase, el análisis forense **comprende dos fases**: la primera, la **captura de las evidencias y su protección**; la segunda, el **análisis de las mismas**. Sin embargo, debido a que en los crímenes digitales cada vez resulta más difícil dar respuesta a los seis interrogantes, especialmente quién realizó el ataque, la investigación forense suele centrarse en averiguar qué fue dañado, cómo fue dañado y cómo arreglarlo.
Durante la fase de recolección de evidencias se captura todo aquello que resulte susceptible de posible análisis posterior y que pueda arrojar luz sobre detalles de muestras de un delito. El análisis de la evidencia es la fase más extensa y delicada, ya que requiere poseer conocimientos avanzados para poder interpretar las pruebas incautadas, cuyo volumen puede llegar a ser inmenso.

Dependiendo de la calidad de los datos de registro de actividad se podrá realizar de forma más o menos sencilla el análisis de la evidencia. Igualmente, dependiendo de la información existente se procederá a obtener unos resultados más o menos satisfactorios.

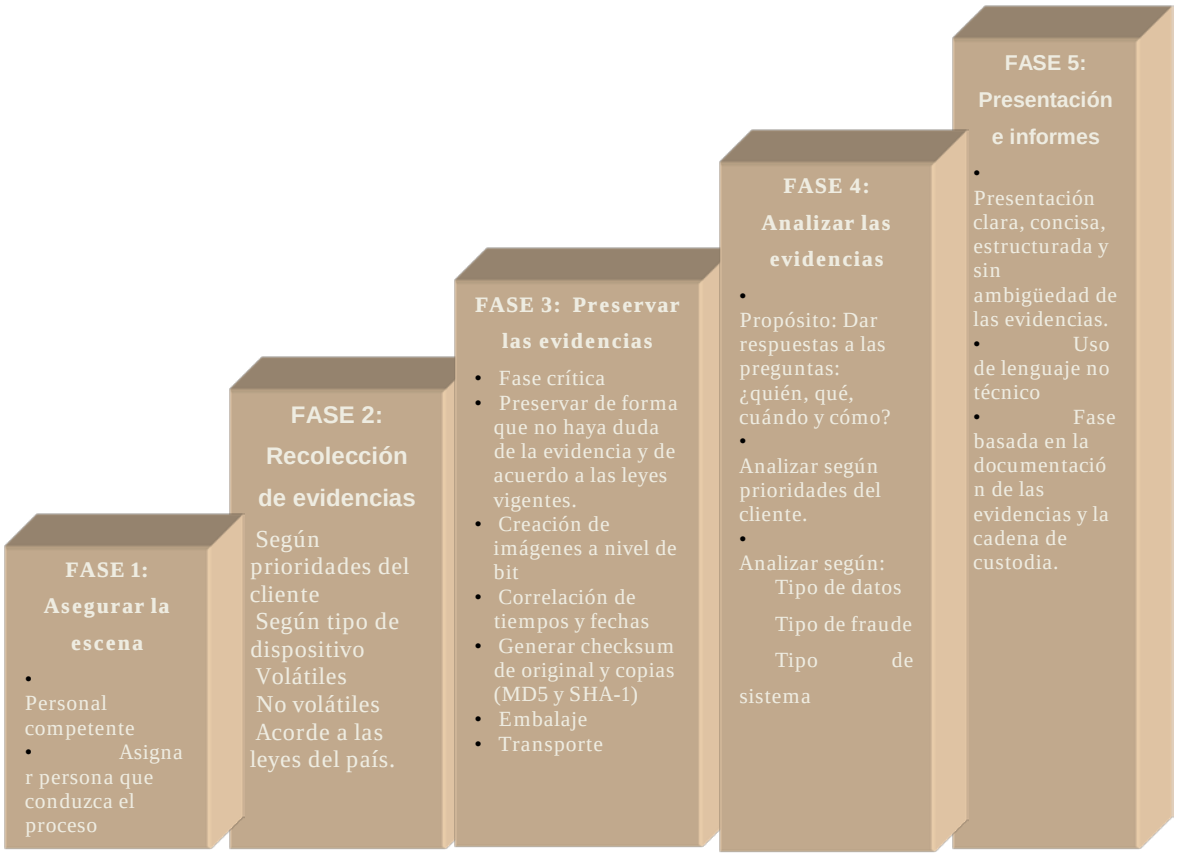


Figura 1. Esquema de las distintas fases del proceso de análisis forense de un sistema

12.3. Recolección de evidencias

El primer paso en todo análisis forense comprende la **captura de la evidencia**. La evidencia será la prueba del delito, la que delatará al intruso. El objetivo de la captura no sólo es la obtención de la misma, sino que para ello se debe proceder con cautela para no modificar pista alguna, ya que la modificación de la prueba varía la evidencia, por lo que puede hacer inútil el análisis posterior, así como su validez en un juicio.

En un análisis forense, prácticamente **cualquier dato puede resultar una evidencia**, como por ejemplo:

- Una línea de texto en un log de algún dispositivo de red.
- La hora de último acceso a un fichero.
- Una *cookie* en la carpeta de trabajo del navegador.
- Una MAC en una tabla ARP.

- El tiempo desde que se arrancó un sistema.
- DLLs (librerías dinámicas) cargadas en el sistema.
- Usuarios activos. Sesiones activas.
- Un proceso en ejecución.
- Ficheros temporales.
- Papelera de reciclaje.
- Caché de los navegadores.
- Etc...

Como en un proceso judicial tradicional, en un análisis forense informático las evidencias también deben poseer una serie de características para ser válidas. En esencia, la evidencia digital debe ser ***admisible***, ***válida*** y ***relevante***:

- ***Admisible***, es decir, debe ser haber sido obtenida legalmente. Obviamente, no todo vale y no se pueden utilizarse medios ilegales (o simplemente dudosos) para obtener evidencias, por muy importantes que éstas pudieran llegar a ser.
- ***Válida***, lo que significa que debe ser auténtica, fiable y creíble.
- ***Relevante***, es decir, debe estar directa o parcialmente relacionada con el hecho que se investiga.

Recuerda que el objetivo último de todo el proceso es poder demostrar las tres grandes premisas, conocidas por sus siglas, **MOM, medios/oportunidad/motivos.**
Obviamente, se debe proceder a realizar la captura de la evidencia con herramientas que no modifiquen ni el entorno ni la prueba en sí, salvaguardando su integridad. A este proceso se le conoce con el nombre de **cadena de custodia**, que habría que preservar durante todo el proceso, para demostrar que las pruebas obtenidas no han sido manipuladas y puedan tener así validez jurídica.

Suelen distinguirse dos tipos de evidencia. La primer, conocida como **volátil**, comprende la información que desaparece cuando un sistema informático pierde la alimentación eléctrica. Por consiguiente, en esta categoría se incluye tanto la memoria RAM, los procesos activos y usuarios conectados, así como la información de la red y aplicaciones a la escucha en todos los puertos en el momento de la interrupción.

El segundo tipo de evidencia, denominada **permanente o no volátil**, es aquella almacenada en discos duros, unidades USB y, en general, cualquier medio de almacenamiento secundario.

Captura de la evidencia volátil

Dada su fragilidad, y que puede perderse con mucha facilidad, este tipo de evidencia es la primera que debe ser recogida. Por tanto, en la medida de lo posible, **la máquina objeto del análisis no debería ser apagada o reiniciada hasta que se haya completado el proceso**. Si se ha ensayado con anterioridad o es realizado por un especialista, no debería llevar más de unos pocos minutos.

La teoría señala que la herramienta perfecta para este proceso no debería apoyarse en absoluto en el sistema operativo objeto del análisis, pues éste podría haber sido fácilmente manipulado para devolver resultados erróneos. Sin embargo, a pesar de que tales herramientas existen, como ***Tribble***, son herramientas hardware, que necesitan estar instaladas en la máquina **antes** de la intrusión, ataque o análisis de la misma.

Evidentemente, este escenario sólo es factible para máquinas que procesan información especialmente sensible, cuyo hardware puede ser fácilmente controlado. En el resto de casos, la inmensa mayoría, hay que conformarse con utilizar herramientas software y limitar el proceso de recolección de información a los mínimos pasos posibles, con el fin de generar el menor impacto posible sobre la máquina analizada.

Lo ideal sería **hacer uso de un dispositivo de sólo lectura**, como una unidad de CD-ROM o una unidad USB, que contenga las herramientas necesarias para el análisis. Existen varias **distribuciones especializadas en análisis forense**, pero las más conocidas y completas son las siguientes:

http://www.backtrack-linux.org/

http://www.caine-live.net/

Ambas se distribuyen en forma de imagen ISO, de forma que es posible cargar un SO nuevo, absolutamente confiable, para realizar la segunda parte del proceso de recogida de evidencias, que analizaremos a continuación.

Espacio de almacenamiento

Para almacenar las evidencias recogidas, será necesario añadir al sistema analizado algún tipo de almacenamiento externo. Teniendo en cuenta que se está realizando la fase de análisis en vivo y que, por tanto, no es posible apagar el ordenador todavía, existen básicamente dos opciones. La primera **consiste en utilizar una unidad externa, como un disco duro o una memoria USB de suficiente capacidad**. Por otro lado, la segunda opción **implica añadir a la red de la máquina analizada un nuevo sistema, habitualmente un ordenador portátil, en el que poder copiar los datos recogidos**.

El primer método sea quizás el más sencillo y rápido de los dos, pero deja más trazas en el sistema analizado. Por supuesto, también necesita que el sistema cuente con un interfaz USB disponible. Utilizar otra máquina como almacén, por el contrario, tendría el mínimo impacto sobre el sistema analizado. A cambio, complica y enlentece ligeramente el proceso de toma de datos.

En función del tipo de conexión que la máquina analizada tenga a Internet, a través de un módem o de un router, este método podría necesitar cortar la conexión de la misma momentáneamente, lo que provocaría la pérdida de las conexiones activas en el momento del análisis, que, como veremos, es una información de sumo interés.

Memoria principal

El primer tipo de evidencia a recoger es la **memoria RAM**, a pesar de que es habitual que, en muchos procesos forenses, esta reciba poca o ninguna atención. Sin embargo, este tipo de memoria es una fuente muy importante de información, que será irremediablemente perdida en cuanto la máquina sea apagada o reiniciada.

Existen, de hecho, evidencias que en ocasiones sólo podrán ser encontradas en RAM, como los nuevos y sofisticados métodos de infección de ordenadores, que residen únicamente en memoria y no escriben nunca nada al disco duro.

El siguiente paso consistiría en **obtener información sobre todos los procesos activos en el sistema, junto con los puertos y ficheros que cada uno de ellos tienen abiertos**.

El siguiente conjunto de información interesante son las conexiones de red activas y puertos TCP/UDP abiertos, además del entorno de red de la máquina, como su tabla de rutas y ARP.

Afortunadamente, las dos distribuciones forenses mencionadas cuentan con herramientas que recolectan esta información de forma automática, acelerando el proceso, ya que el tiempo es un factor crítico durante una intrusión, y reduciendo la posibilidad de cometer errores.

Por ejemplo, en un entorno *Windows* podríamos utilizar un comando similar a éste para realizar la copia de la memoria RAM:

```
dd.exe if=\\.\PhysicalMemory bs=4096 of=\temp\memory.img
```

Y éste para un entorno *Linux*:

```
dd if=/proc/mem bs=1024 of=/Physicalmemory.img
```

Existe también otra herramienta muy interesante para analizar la memoria de un sistema «vivo», recién comprometido, llamada **pmdump 1** (Disponible en: http://ntsecurity.nu/toolbox/pmdump).

Veamos su salida tipo:

```
>pmdump -list
pmdump 1.2 - (c) 2002, Arne Vidstrom (arne.vidstrom@ntsecurity.nu)
              - http://ntsecurity.nu/toolbox/pmdump/
    0 - System idle process
    8 - System
  224 - SMSS.EXE
…
…
 1640 - mdm.exe
 1324 - vmware.exe
 1768 - CMD.EXE
 1644 - pmdump.exe
> pmdump 1768 cmd.mem
```

Como puedes ver, esta aplicación lista todos los procesos corriendo en un sistema, de forma que pueda identificarse fácilmente aquellos sospechosos de ser los culpables de la intrusión. Podrías pensar, en este punto, que se obtendría la misma información con las herramientas del SO, y no es necesario utilizar así una aplicación. Pero recuerda que, tras una intrusión, no deberías confiar en ninguna de las salidas (listado de procesos corriendo, ficheros en disco, etc.) que proporcione el SO, pues todas estas podrían estar manipuladas por software instalado durante el ataque.

Esta aplicación puede, además, realizar una acción muy interesante, mostrada en la última línea del cuadro anterior: volcar a disco el contenido en RAM de un proceso vivo. De esta forma podremos tomar «muestras» de procesos sospechosos para analizarlo, después, con calma.

Para ver este proceso en acción realiza los siguientes **pasos sencillos**:

1. Abre el Notepad y escribe cualquier frase en él. ¡No grabes ni cierres el fichero!
2. Busca el ID del proceso del Notepad con el comando anterior:

```
>pmdump –list
```

3. Genera un volcado de la memoria del proceso:

```
>pmdump ID_proceso notepad.mem
```

4. Examina lo encontrado en busca de la frase que escribiste. ¿Qué encuentras?

```
>strings notepad.mem | grep32 “TEXTO”
```

Búsqueda de cadenas

La búsqueda de cadenas imprimibles, aquellas cuyos caracteres tienen códigos ASCII que no corresponden a comandos, es una técnica básica pero flexible y muy poderosa.

Se lleva a cabo con el comando «strings», como en el siguiente ejemplo:

```
>strings particion1.dd | more
```

El número de cadenas devueltas de esta forma es, normalmente, muy alto y resulta difícil de analizar manualmente. Por eso esta técnica es más apropiada cuando se sabe qué buscar. Por ejemplo, si se desea encontrar todas las apariciones de cuentas de correo de Hotmail, puede realizarse una búsqueda similar a la siguiente:

```
>strings particion1.dd | grep “hotmail.com”
```

En el **Anexo A** puede encontrarse una lista completa de patrones correspondiente a cadenas potencialmente útiles en un análisis forense, cómo direcciones de correo, direcciones IP, teléfonos, etc...

Captura de la evidencia de disco

Una vez asegurada la evidencia volátil, puede realizarse, si es necesario, la desconexión del sistema de la red o su apagado. Tras ello, se iniciará la **segunda fase del proceso de recolección**, **cuyo objetivo son ahora los discos duros**. Para ello, el primer paso sería siempre realizar una copia de los mismos, pues nunca debe trabajarse sobre los datos originales.

Para este propósito pueden utilizarse cualquiera de las distribuciones mencionadas en el apartado anterior, que incluyen varias buenas herramientas. Otra opción consiste en el **uso de una utilidad software específica para el clonado de discos**, como *Norton Ghost*, o su equivalente libre, *Ghost for Linux*, que puede encontrarse en http://sourceforge.net/projects/g4l.

Para mantener la cadena de custodia, los discos clonados deberían ser etiquetados con la fecha y hora de la copia, el nombre de la persona que llevó a cabo el proceso y un resumen criptográfico de cada disco. Para evitar posibles problemas, es recomendable utilizar el hash *SHA1*, debido a las recientes vulnerabilidades descubiertas en el conocido *MD5*, que ya han sido utilizadas como triquiñuelas legales en algún juicio.

Análisis del historial de comandos

Para facilitar la repetición de comandos largos, el shell bash almacena los últimos 500 comandos escritos en el fichero `~/.bash_history`. Cada usuario con una cuenta en el sistema tiene este fichero en su directorio de trabajo.

Como puede suponerse, ésta es una fuente de información potencialmente muy valiosa, pues Linux es un sistema operativo en el que muchas operaciones siguen realizándose desde la línea de comandos por parte de los usuarios.

Para encontrar todos los ficheros de historial presentes en el sistema puede utilizarse el siguiente comando:

```
>find / -name .bash_history
```

Para visualizar los contenidos del historial basta con listar el contenido de dicho fichero, o utilizar el comando «history», que muestra el historial del usuario actual. La salida de este comando será similar a la siguiente:

```
...
280 find / -name .bash_history
281 less /home/oscar/.bash_history
282 less /home/oscar/.bashrc
283 less /home/oscar/.bash_logout
284 less /home/oscar/.bashrc
285 less /root/.bash_history
286 ls -l /home/oscar/.bash_history
287 history
288 man history
289 man bash_history
...
```

Hay que tener en cuenta, sin embargo, las siguientes **consideraciones acerca de esta técnica**:

- Las líneas almacenadas en el historial están ordenadas temporalmente, de forma que las que aparecen primero son más antiguas, pero no puede saberse cuándo fue tecleado cada comando.
- Un mismo comando puede ser tecleado múltiples veces y sólo aparecerá una en el historial pues, por defecto, las líneas duplicadas no se almacenan.
- El comportamiento normal del historial hace que sólo se almacenen los comandos al ejecutar 'exit', o salir convenientemente, de forma que si no es así, los comandos no se almacenarán.

Búsqueda de archivos ocultos

Una técnica muy común en *Linux*, utilizada habitualmente por *rootkits*, es **crear archivos y directorios ocultos**, que no se muestran con las opciones normales de comandos como «ls» o el explorador gráfico de archivos.

Por supuesto, ocultos no significa en absoluto inaccesibles, por lo que resultan fáciles de localizar. Para ello puede ejecutarse:

```
>/usr/bin/find / -name ".*" -exec ls -la {} \;
```

Sin embargo, la lista de ficheros ocultos puede ser larga. De hecho, habitualmente lo es, ya que existen ciertos tipos de archivo que el propio sistema operativo oculta, como ficheros de configuración, normalmente por razones estéticas.

Por ejemplo, en una instalación ***Ubuntu*** por defecto existen más de 1000 archivos legítimos de este tipo. Para agilizar el análisis, la solución pasa por el análisis diferencial. Para ello se genera un fichero de base, que contenga todos los ficheros ocultos normales que se encuentran en una distribución Linux recién instalada. Este fichero puede generarse con el mismo comando anterior.

Después bastaría buscar aquellos ficheros que no están en el fichero base que hemos creado. Para esto puede ejecutarse el siguiente comando:

```
diff ficheros_ocultos.txt ficheros_ocultos_base.txt
```

Espacio de swap

Para que un ordenador pueda ejecutar programas de tamaño superior al de la memoria principal, todos los sistemas operativos modernos utilizan una técnica llamada ***swapping***. Esta técnica consiste en almacenar porciones de la memoria principal en el disco duro, cuando éstas no se están utilizando, y recuperarlas cuando sean necesarias.

Además, mucha de la memoria utilizada por una aplicación típica sólo se utiliza durante su inicialización, y no se vuelve a usar. El sistema, entonces, puede desplazar a disco esta memoria y así liberar espacio para otras aplicaciones.

Partición y fichero de swap

Linux implementa el espacio de swap de dos formas: utilizando particiones y ficheros. La partición de swap es una partición del disco dedicada exclusivamente a esta tarea y ningún otro fichero puede residir allí. Por otro lado, el fichero de swap es un fichero normal, que reside en el sistema de ficheros, junto a otros archivos y datos.

Para comprobar cuánto espacio de swap tiene una máquina, puede utilizarse el comando ***swapon***.

Interés forense

El interés forense de este espacio de swap es muy alto, puesto que **suele almacenar información muy valiosa**. En general, la información aquí almacenada se encuentra desordenada y habitualmente, incompleta. Sin embargo, pueden encontrarse URL's visitadas, aunque hayan éstas hayan sido intencionadamente eliminadas por el usuario, contenido de páginas Web visitadas, correos electrónicos enviados o recibidos y, en general, cualquier información que haya fluido alguna vez por el sistema operativo.

La ventaja, desde el punto de vista forense, del espacio de swap es que no **se elimina ni se borra cuando el sistema operativo se reinicia o apaga**. Por tanto, puede encontrarse información mucho tiempo después de que ésta haya sido generada. En general, es la información más longeva que puede encontrarse en un sistema.

Protección de la partición de swap

Aunque no es un hábito habitual ni extendido, **existen técnicas para proteger la información remanente de la partición de swap**. La principal es el cifrado de la misma, lo que imposibilita su posterior análisis si sólo se tiene acceso al sistema operativo apagado.

Por el contrario, si se puede acceder a la máquina encendida, la medida anterior es inútil, pues todos los sistemas de cifrado disponibles guardan una versión no cifrada de la partición, para que pueda ser utilizada por el sistema operativo. Otra técnica que podría utilizarse es la **eliminación de la información de la partición en cada reinicio del sistema**. Nuevamente, sólo resulta efectiva si no se tiene acceso al sistema operativo funcionando. Para este proceso pueden utilizarse programas como ***shred***, e incluirse en los scripts de arranque y apagado del sistema:

```
> shred -n1 -v /dev/hda6
```

12.4. Análisis de las evidencias

Sorprendentemente, a pesar de la creciente importancia que está cobrando la **forensia digital**, no hay definida una metodología clara del análisis de las evidencias recogidas. Aunque es cierto que, en parte, es debido a la inherente dificultad del proceso, que se presta poco a una procedimentación exhaustiva. Como consecuencia, la fase de análisis del proceso forense sigue siendo en buena medida un proceso manual, que requiere de especialistas con mucha experiencia.

A esta situación ayuda también la escasez de herramientas de análisis, hasta el punto de que sólo pueden considerarse un par de ellas dignas de mención. Por un lado, la conocida ***EnCase***, muy completa, aunque de precio realmente elevado, lo que la convierte en una opción sólo para grandes corporaciones, gobiernos o cuerpos de seguridad. Por otro lado, una alternativa libre y gratuita, el conjunto de dos aplicaciones, ***The Sleuth Kit*** y ***Autopsy***, disponibles ambas en http://www.sleuthkit.org/.

Para realizar el análisis, lo ideal sería contar con un pequeño **«kit de forensía»**, que puede estar compuesto por elementos tan sencillos como un par de discos duros portátiles, con capacidad suficiente para clonar otros discos (alrededor de 400GB sería lo recomendable), una memoria USB y un juego de CD's con las distribuciones forenses mencionadas, listos para ser utilizados rápidamente.

En este punto, es posible actuar de dos maneras. La primera consiste en **trabajar sobre los datos en bruto, tal y como fueron recogidos**. Esto, aparte de hacer el proceso lento y tedioso, limita mucho las posibilidades de actuación, a poco más que una búsqueda de cadenas.

La segunda, mucho más recomendable, consiste en **«arrancar» la copia, para poder trabajar en el entorno original del sistema analizado**. Por supuesto, el sistema operativo no arrancará fácilmente si, simplemente, nos limitamos a conectar el disco duro clonado a una nueva máquina. Para facilitar esta tarea, existen herramientas específicas, como ***Live View*** (disponible en http://liveview.sourceforge.net) y que crea una ***máquina virtual Vmware*** de una imagen forense en bruto previamente tomada o de un disco físico.

Esta herramienta resulta realmente recomendable, pues facilitará enormemente el trabajo. Para asegurar que el proceso de análisis no manipula de ninguna forma la evidencia original, la máquina virtual puede marcarse como de sólo lectura.

Por supuesto, en el interior de esta máquina virtual, podemos ejecutar cualquiera de nuestras distribuciones forenses favoritas y empezar realmente el análisis. Éste debe comprender, como mínimo, los siguientes pasos.

Recuperación de archivos borrados

Su dificultad, sin embargo, varía mucho de un SO a otro. Por ejemplo, cuando un fichero es borrado en una máquina *Linux* que utilice un sistema de ficheros *ext2*, el sistema operativo se limita a marcar los bloques de datos como *libres*. Sin embargo, los enlaces a los bloques de datos todavía están disponibles y, por tanto, es posible e incluso relativamente sencillo recuperar un archivo borrado, siempre y cuando estos bloques no hayan sido asignados ya a otro archivo.

Este proceso es muy similar al de *Windows*, siendo la lista de inodos equivalente a la *FAT*. Por la misma razón, la probabilidad de recuperar correctamente un archivo decrece con el tiempo transcurrido y la carga del sistema.

Sin embargo, con *ext3* el proceso es bastante más complicado. La razón radica en que este sistema de ficheros realiza un paso adicional en el borrado de un archivo: elimina también el tamaño del fichero y los enlaces a los bloques de datos. De esta forma, es muy difícil determinar dónde estaban los datos originales y a qué fichero pertenecían.

En cualquier caso, el proceso, sobre todo en entornos *Windows*, suele recuperar bastante información eliminada. Este paso es especialmente importante, pues un intruso tratará siempre de ocultar sus huellas borrando aquella información más comprometedora. Por ejemplo, los logs del sistema, el primer sitio dónde comenzar el análisis.

Otro punto de mucho interés, que también suele pasarse por alto, es el espacio de intercambio o de swap. El interés forense de este espacio, puesto que suele almacenar información muy valiosa, a pesar de que, en general, la información aquí almacenada se encuentra desordenada y, habitualmente, incompleta.

Sin embargo, pueden encontrarse URL's visitadas, aunque hayan estas hayan sido intencionadamente eliminadas por el usuario, contenido de páginas Web visitadas, correos electrónicos enviados o recibidos y, en general, cualquier información que haya fluido alguna vez por el sistema operativo.

La ventaja, desde el punto de vista forense, del espacio de swap es que no se elimina ni se borra cuando el sistema operativo se reinicia o apaga. Por tanto, puede encontrarse información mucho tiempo después de que ésta haya sido generada. En general, es la información más longeva que puede encontrarse en un sistema, incluso con meses de antigüedad. En entornos Microsoft, este espacio de swap puede encontrarse en el archivo C:\pagefile.sys.

Afortunadamente estas tareas pueden ser realizadas de forma semi-automática con las herramientas The Sleuth Kit y Autopsy. Incluyen, además, la posibilidad de gestionar los casos, generar una reconstrucción temporal del ataque o la generación de informes directamente en formato PDF o HTML.

12.5. Técnicas antiforenses

A continuación se analizarán algunas de las **técnicas antiforenses** más importantes que podrían ser utilizadas.

No usar un disco duro

Sin duda, una de las más efectivas y seguras consiste en **no dejar rastros analizables**. Para ello, basta con utilizar, por ejemplo, un CD autoarrancable, que contiene un sistema operativo completo y utilizable. Al ejecutarse completamente en memoria RAM, no quedan evidencias que puedan ser posteriormente recogidas y analizadas.

Cifrado con criptografía fuerte

Otra técnica sería el **cifrado de ciertos archivos o, incluso particiones o discos duros completos, con algoritmos fuertes**. «Fuertes» significa que no contienen vulnerabilidades criptográficas conocidas y, por tanto, no son atacables en este sentido.

Existen, por el contrario, multitud de protocolos propietarios, de mala calidad, que pueden ser fácilmente descifrados, como el de algunos teclados inalámbricos o discos duros. Sin embargo, si se utilizan protocolos estándares, como AES, el criptoanálisis no es posible, por lo que el único recurso sería realizar ataques por diccionario o fuerza bruta contra la contraseña.

Algunas de las herramientas más conocidas que podrían utilizarse para este cifrado en Linux serían GPG, equivalente a PGP, para archivos y TrueCrypt para particiones y discos completos.

TrueCrypt

TrueCrypt es una herramienta multiplataforma, *Linux* y *Windows*, que sirve para crear particiones y volúmenes cifrados. Estas particiones se cifran y descifran en tiempo real, de forma que los datos no permanecen descifrados más que unos instantes en memoria RAM. El análisis forense de unos de estos volúmenes es, en principio, imposible, debido a que los algoritmos de cifrado utilizados son robutos (AES).

Existe, además, una dificultad añadida en este análisis. Este software tiene la capacidad de crear volúmenes cifrados negables, utilizando *esteganografía*. Un volumen negable es un volumen cifrado oculto cuya propia existencia no puede ser descubierta. Suelen ser utilizados para evitar situaciones en las que el usuario es obligado a revelar la contraseña de acceso.
Estos volúmenes negables se crean en el interior de un volumen cifrado normal. Incluso cuando éste está montado y en uso, es imposible probar que existe un volumen oculto en su interior, porque el espacio libre se llena siempre con datos aleatorios y, por tanto, el volumen oculto no puede ser distinguido de éstos.

La contraseña del volumen oculto es, por supuesto, diferente a la del volumen visible.
Una técnica habitual sería almacenar en éste último información no sensible, cuya revelación no suponga un compromiso. De esta forma, si es forzado a revelar la contraseña, sólo se tendrá acceso a esta información.

En la **Figura 1** puede observarse un esquema del proceso.

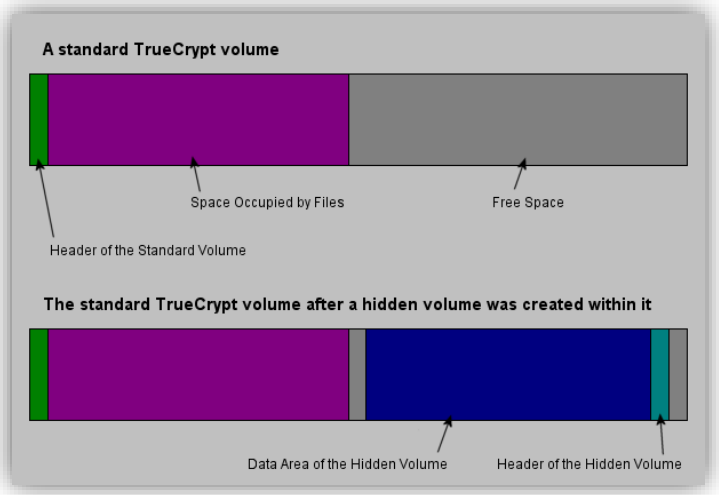


Figura 2. Partición esteganográficamente oculta

Archivos ZIP con clave

El algoritmo de cifrado utilizado por el conocido software de compresión PKZIP es vulnerable a un ataque criptoanalítico llamado **ataque por texto claro conocido**. De esta forma es posible recuperar cualquier clave, de cualquier longitud, en unos pocos minutos. La contrapartida es que es necesario conocer de antemano una pequeña parte de la información cifrada (13 bytes como mínimo).

Esta necesidad, en general, es solucionable. Los contenidos de un archivo ZIP, aunque éste esté cifrado, pueden listarse sin problemas. De esta forma es posible conocer los nombres de los archivos. Si es posible averiguar o intuir sólo 13 bytes de la información cifrada, entonces puede aplicarse el método.

Recuperación de la contraseña

Para realizar este método es necesario utilizar **la herramienta «pkcrack»**, disponible en el CDROM del curso. Como se ha comentado, **este ataque necesita conocer de antemano una porción de la información contenida en el archivo cifrado**. Esto puede parecer un contrasentido, pero puede ser muy útil en ocasiones. Imaginemos que el fichero ZIP contiene múltiples archivos, por ejemplo fotografías y que ha sido posible recuperar una de ellas. En este escenario sería posible recuperar el resto de imágenes, sin importar lo compleja que sea la contraseña de cifrado.

El procedimiento es como sigue:

1. En el punto de partida se cuenta con un archivo cifrado llamado 'imagenes.zip' y uno de los ficheros que contiene, llamado «img_0131.jpg».

2. Se genera un fichero ZIP, de nombre «imagenes_descifradas.zip» con la imagen:

```
> zip img_des.zip img_0131.jpg
```

3. Extraemos de este fichero recién creado la porción comprimida correspondiente a la imagen y renombramos el fichero a «descifrada.jpg»:

```
> extract img_des.zip img_0131.jpg
> mv img_0131.jpg descifrada.jpg
```

4. Realizamos la misma operación con la versión cifrada:

```
> extract imagenes.zip img_0131.jpg
> mv img_0131.jpg cifrada.jpg
```

5. Utilizamos el programa «pkcrack» para, finalmente, recuperar la contraseña:

```
>pkcrack -c cifrada.jpg -p descifrada.jpg
```

Este proceso puede recuperar cualquier contraseña, independientemente de su complejidad, en un par de minutos. Si no se dispone de un *fichero en claro* que también esté contenido en el ZIP, el proceso se vuelve más complejo y lento, pero aún puede llevarse a cabo.

12.6. Caso de estudio práctico

En este apartado veremos un caso práctico, de un ataque real. Analizaremos cómo se procedió, de forma que tengamos una visión global de todas las técnicas y aspectos que hemos visto en el tema.

Caso práctico:

Como responsable de seguridad IT de una empresa, te informan de que se ha detectado que se ha accedido a una máquina de otro empleado para sustraer un documento confidencial, sospechándose de alguien de la propia empresa.

Estos son los pasos para realizar el análisis informático que se siguieron en este caso, tal como explica **David Barroso**, experto en análisis forense de *S21sec*:

1. Premisa inicial y toma de contacto:

- Conversación con los relacionados en el caso, con el objetivo de tener una idea clara de lo que ha ocurrido y planificar la investigación.
- La premisa esencial de trabajo debe ser documentar todas las acciones realizadas, puesto que de los resultados obtenidos (evidencias y documentación) dependerá su fiabilidad y validez ante terceros.

2. Se inicia la investigación con la copia del disco duro del sospechoso.

- Acompañado de un representante de los trabajadores y de los responsables del departamento, se requisa el ordenador del sospechoso para, delante de los responsables, realizar una copia íntegra, bit a bit, del disco duro.
- Esta copia servirá para la investigación, y el ordenador requisado será guardado para su custodia en la caja fuerte de la empresa o en un lugar que garantice su integridad. ¡Esto es importante para garantizar la cadena de custodia!
- Si se dispone de otros medios de almacenamiento externo, como CD o discos USB, se requisan también para su posterior análisis.

3. Análisis del disco duro.

- Se buscan indicios de virus, troyanos o gusanos, así como herramientas de ataque, con la idea de realizar una foto del ordenador y saber para qué se usa realmente, así como el nivel del usuario.
- A continuación se localiza y analiza la información que ha sido borrada.
- Seguidamente, se realiza una escala temporal (timeline de nuestras acciones y timeline en base a los eventos del sistema) lo más exacta y detallada posible.

- Conociendo aproximadamente la fecha del suceso, se reconstruye el escenario segundo a segundo: **acceso a ficheros, creación o borrado de ficheros, ejecución de comandos, accesos a páginas de Internet...**
- Como se trata de un robo de información confidencial, se busca en todo el disco, tanto en archivos existentes, como en archivos borrados y espacio no utilizado.

4. Análisis de logs

- Revisión y análisis de todos los logs o huellas de actividad, de la máquina y de todos los elementos relacionados, tales como cortafuegos, IDS, impresoras y servidores.

5. Cadena de custodia

- Una vez que hemos obtenido las pruebas, es fundamental documentarlas y mantener lo que se denomina cadena de custodia, que garantiza el origen de las pruebas, imprescindibles si hay juicio.

6. Presentación de resultados a los responsables

- Acciones legales/disciplinarias si fuera necesario

Accede el artículo con las indicaciones de David Barroso a través del aula virtual o desde la siguiente dirección web:
http://elpais.com/diario/2006/01/19/ciberpais/1137641068_850215.html

Lo + recomendado

No dejes de leer...

Buenas prácticas de recolección de evidencias (RFC 3227)

A pesar de su importancia, el análisis forense es uno de esos campos de la informática que sigue estando poco procedimentado. Este documento muestra un intento de estandarizar el proceso de recolección de evidencias, recogido en un *RFC (Request For Comments)*.

Accede al documento a través del aula virtual o desde la siguiente dirección web:
http://www.faqs.org/rfcs/rfc3227.html

No dejes de ver...

Autopsy live computer forensic practical

En este vídeo se explica el funcionamiento del sofware forense Autopsia para análisis forense informático.

Accede al vídeo a través del aula virtual o desde la siguiente dirección web:
https://www.youtube.com/watch?v=Smy4mj293GE

Informática forense y seguridad

Interesante charla de un experto en Informática Forense, Peritaje y Seguridad, presentada el 21 de Marzo de 2013 dentro del Ciclo de Conferencias de la Escuela Superior de Informática de la Universidad de Castilla-La Mancha.

Accede al video a través del aula virtual o desde la siguiente dirección web:
https://www.youtube.com/watch?v=UhumXfZedM0

Técnicas anti-forense

En este caso vemos la contrapartida del análisis forense: las técnicas para evitarlo. Las presenta un miembro del Cuerpo Nacional de Polícia, en el marco de las conferencias DISI de la UPM, lo que la hace más interesante.

Accede al video a través del aula virtual o desde la siguiente dirección web:
https://www.youtube.com/watch?v=-j-yoBiqFAQ

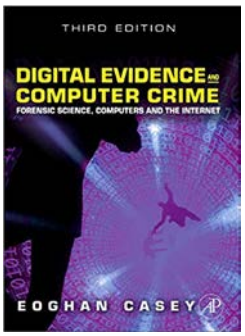
+ Información

A fondo

Digital Evidence and Computer Crime

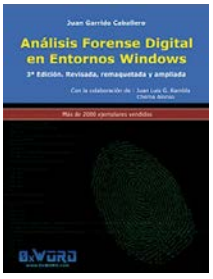
Casey, E. (2011). *Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet*. Editorial Academic Press. ISBN: 978-0123742681

Este libro proporciona el conocimiento necesario para descubrir y utilizar las evidencias digitales de manera efectiva en cualquier tipo de investigación.



Análisis Forense Digital en Entornos Windows

Garrido, J. (2010). *Análisis Forense Digital en Entornos Windows*. Ediciones 0xWORD. ISBN: 978-84-616-0392-3.



Este libro, ameno y bien escrito, hace un repaso exhaustivo sobre las técnicas de análisis forense en entornos *Windows*. Es especialmente interesante porque tiene un buen listado de herramientas que pueden ser utilizadas en un análisis forense.

Test

1. Llegas al escenario de un delito informático como experto en la materia. Allí el secretario judicial te insta a que le indiques como va a ser tu proceder en cuanto al tratamiento de las evidencias de un análisis forense. Tu contestación es:

A. Preservación – Análisis - Recolección
B. Recolección – Análisis - Preservación
C. Recolección – Preservación - Análisis
D. Análisis – Recolección - Preservación

2. El objetivo de la cadena de custodia es:

A. Demostrar que los ejecutores del análisis forense están suficientemente capacitados para realizarlo
B. Demostrar que el sistema ha sido realmente atacado y cómo
C. Demostrar que las evidencias no han sido manipuladas durante la fase de recolección
D. Demostrar que las evidencias no han sido manipuladas durante todo el desarrollo del análisis forense

3. Las evidencias volátiles son aquellas que:

A. Son frágiles y pueden modificarse fácilmente sin querer
B. Son difíciles de conseguir, puesto que suelen estar ocultas
C. Se pierden cuando el equipo pierde la alimentación eléctrica
D. Se encuentran en el disco duro del equipo

4. Para que una evidencia digital pueda ser aceptada judicialmente debe ser admisible, válida y relevante. Pero, ¿qué significan exactamente estos aspectos?

A. Admisible: obtenida legalmente, válida: auténtica, fiable y creíble y relevante: relacionada directamente con el hecho
B. Admisible: auténtica, fiable y creíble, válida: relacionada directamente con el hecho y relevante: obtenida legalmente
C. Admisible: relacionada directamente con el hecho, válida: obtenida legalmente y relevante: auténtica, fiable y creíble
D. Admisible: auténtica, fiable y creíble, válida: obtenida legalmente y relevante: relacionada directamente con el hecho

5. Señala ejemplos de evidencias no volátiles:

A. Sesiones TCP activas en el momento de la intrusión

B. Procesos en ejecución

C. Contenido de la memoria RAM

D. Hora de último acceso a un fichero

6. Las dos herramientas de análisis de referencia, de código abierto y propietaria, respectivamente, son:

A. Autopsy y nmap

B. Ethereal y Autopsy

C. Encase y Autopsy

D. Autopsy y Encase

7. Una técnica anti-forense sencilla y efectiva es:

A. Borrar todas las huellas generadas antes de apagar el ordenador

B. Borrar el historial del navegador

C. Eliminar los logs del sistema operativo, de forma que no puedan reconstruirse las acciones realizadas

D. Arrancar el ordenador desde un CD o USB, de forma que no queden evidencias en el disco duro

8. TrueCrypt es una herramienta de:

A. Cifrado de particiones y volúmenes cifrados

B. Análisis de evidencias forenses

C. Búsqueda de evidencias forenses ocultas o borradas

D. Generación de informes de análisis forenses

9. ¿Cómo es aconsejable realizar la recolección inicial de evidencias?

A. En solitario, para evitar que alguien pueda manipular las evidencias

B. Acompañados de algún responsable de la entidad, para poder realizar la copia inicial de discos duros, etc.

C. Acompañados del sospechoso de la acción, para evitar que pueda manipular evidencias

D. Acompañados exclusivamente del personal técnico, pues el resto no entenderá nuestras acciones

10. Es importante documentar cada paso realizado durante el proceso de análisis, con el fin de:

A. Recordar las acciones realizadas para poder elaborar un buen informe
B. Poder demostrar que se ha mantenido la cadena de custodia
C. Aprender de lo sucedido, con el fin de que no vuelva a suceder
D. Ninguna de las anteriores