Forense

OpenWebinars

ÍNDICE

- Iniciación a la informática forense.
- RFC3227.
- Volcados de memoria.
- Algunas herramientas útiles.

INICIACIÓN A LA INFORMÁTICA FORENSE

- El cómputo forense, también llamado informática forense, análisis forense digital o examen forense digital es la aplicación de técnicas científicas y analíticas especializadas a infraestructura tecnológica que permiten identificar, preservar, analizar y presentar datos que sean válidos dentro de un proceso legal.

- Dichas técnicas incluyen reconstruir el bien informático, examinar datos residuales, autenticar datos y explicar las características técnicas del uso aplicado a los datos y bienes informáticos

La obtención de la evidencia: ¿Qué es y cómo se extraen estas pruebas informáticas?

- La prueba o evidencia digital, tiene una gran importancia en el proceso jurídico. Ya sea para un tema civil, penal, laboral, mercantil... A raíz de esta, podemos empezar a realizar una hipótesis y al final, comprendemos lo que ha sucedido realmente.
- Gracias a las pruebas, podemos evidenciar un hecho delictivo o no, que nos lleve hasta la posesión de la verdad. Pero para ello, deberemos pasar un procedimiento para obtener esa prueba.
- Todos los peritos judiciales informáticos deben conocer este proceso de metodología.
- El cual es exigido por nuestras leyes para garantizar que sólo obtendremos la verdad de lo sucedido. Para el derecho, la evidencia digital o electrónica es una certeza perceptible.

- Esta evidencia debe ser manifiesta, evidente u obvia y clara, que nadie pueda refutarla. De ahí, que a veces se oiga a muchos juristas, que ni los testimonios o declaraciones, dejan tan claro un hecho como una prueba. La evidencia es imparcial, no va en contra de nadie y solo evidencia un hecho claro de un suceso acontecido. Mostrando lo que pasó y como ocurrió; e incluso, señalando al responsable o responsables.

Obtener datos periciales informáticos: camino a la prueba

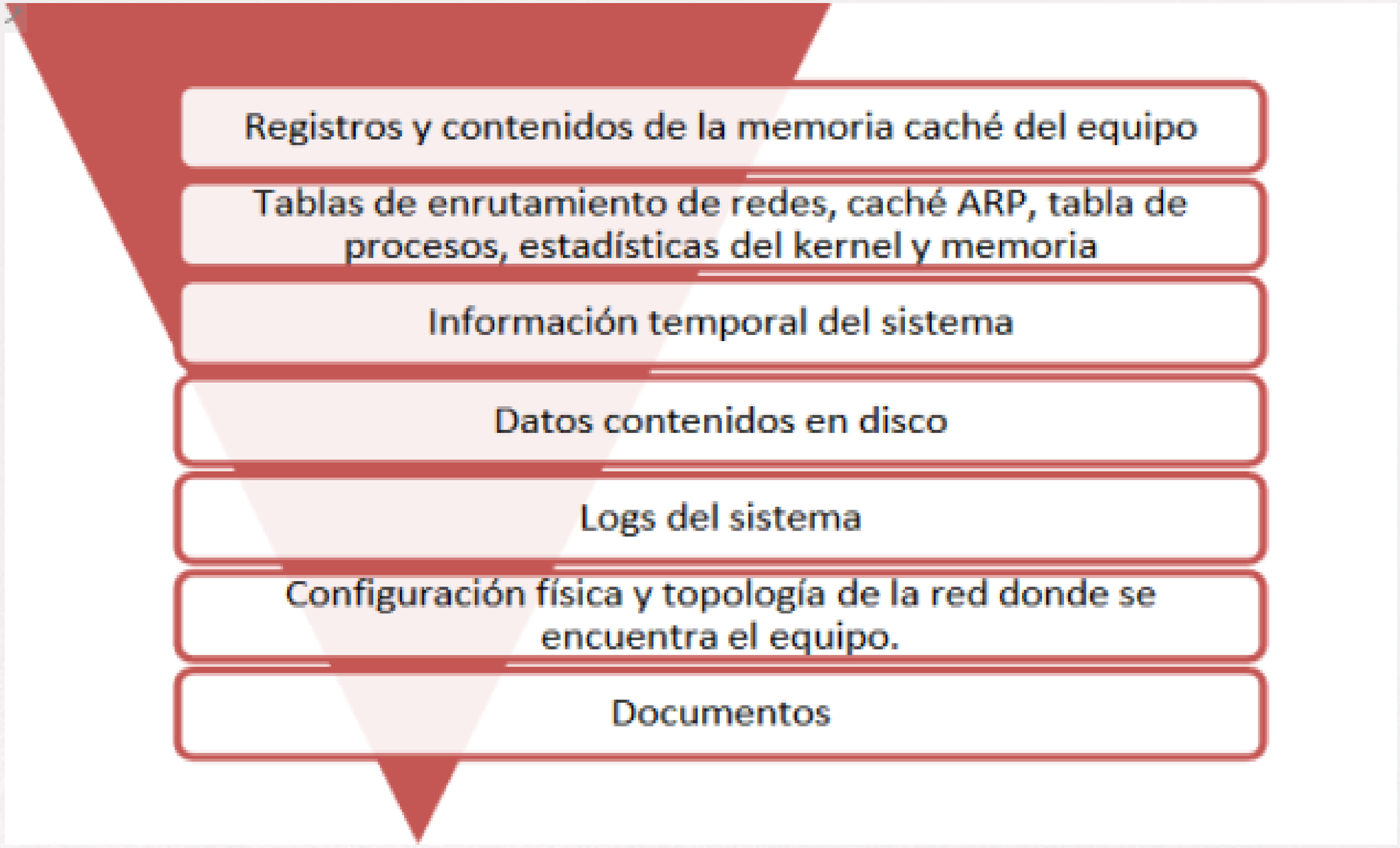
- La evidencia o prueba en las periciales informáticas es un conjunto de datos almacenados. Estos están dentro de un dispositivo electrónico o informático, desde ordenadores, móviles, pendrives... También los datos pueden haber sido transmitidos por un sistema informático. El cual tiene alguna relación con el comedimiento de un delito o acto que compromete este sistema, por donde se ha transmitido. En la ley de Enjuiciamiento Civil, en sus artículos 299 al 386, encontramos las claves que pide la Ley.
- Nos explica e indican cuales son las principales fuentes de pruebas. Una de las fuentes es la denominada dictamen de peritos o prueba pericial. Donde los expertos en una materia, analizan y estudian los componentes para poder extraer la evidencia digital.

RFC 3227: Metodología forense

- **1a fase, asegurar la escena:**
  - Esta fase está más enfocada a casos criminales. No obstante, quizás con menor rigor, es importante en cualquier análisis forense.
  - Es importante recalcar que el investigador no sólo se tiene que centrar en un análisis técnico del equipo o equipos implicados en el incidente, también debe asegurarse que la escena donde se ha producido el incidente no haya sido alterada, desde el descubrimiento del mismo hasta el inicio de su análisis.
  - Todos los implicados en la investigación deben ser conscientes que cualquier acto que lleven a cabo puede comportar unas consecuencias posteriores, así no deben hacer nada que no tengan claro y que pueda alterar los resultados.

- Lo mejor es trabajar consensuando la actuación y anotando todo lo realizado en la escena. Es recomendable realizar fotografías del entorno del equipo para evidenciar el estado original de la escena, identificando así el perímetro de la escena a analizar y protegiéndolo de accesos de personal no autorizado.

- **2a fase , recolección de las evidencias:**
    - **Identificación de las evidencias:**
        - Debido a la volatilidad de los datos y el período de tiempo en el que permanecerán accesibles en el equipo. Debido a estos inconvenientes se deben recolectar de forma ordenada y rápidas las pruebas más volátiles, con un orden de volatilidad de mayor a menor.


Registros y contenidos de la memoria caché del equipo
Tablas de enrutamiento de redes, caché ARP, tabla de procesos, estadísticas del kernel y memoria
Información temporal del sistema
Datos contenidos en disco
Logs del sistema
Configuración física y topología de la red donde se encuentra el equipo.
Documentos

- **Recolección de las evidencias:**
    - Con una primera copia realizada y comprobada procedemos a realizar una segunda copia sobre la primera.
    - En este caso también se comprobará que el contenido es idéntico mediante el mismo proceso descrito anteriormente.
    - Teniendo ambas copias entregaremos la primera al secretario judicial o notarlo responsable del caso y nos quedaremos con la segunda para poder trabajar.
    - La segunda copia será nuestra copia de respaldo en todo momento en el laboratorio y no será para trabajar directamente con ella en ningún caso.
    - Para realizar el análisis se deberá realizar una tercera copia, comprobar su integridad y trabajar sobre ella, de tal modo que en caso de cualquier desastre o alteración de los datos siempre tengamos la segunda copia exacta a la original de donde poder volver a realizar otra copia para analizar. Hash SHA-256 mínimo.

- Los RFC,*request for comments*, con documentos que recogen propuestas de expertos en una materia concreta, con el fin de establecer por ejemplo una serie de pautas para llevar a cabo un proceso, la creación de estándares o la implantación de algún protocolo.
- El RFC 3227 es un documento que recoge las directrices para la recopilación de evidencias y su almacenamiento, y un documento que recoge las directrices para la recopilación de evidencias y su almacenamiento, y puede llegar a servir como estándar de facto para la recopilación de información en incidentes de seguridad.

Principios durante la recolección de evidencias:

- Capturar una imagen del sistema tan precisa como sea posible.
- Realizar notas detalladas, incluyendo fechas y horas indicando si se utiliza horario local o UTC.
- Minimizar los cambios en la información que se está recolectando y eliminar los agentes externos que puedan hacerlo.
- En el caso de enfrentarse a un dilema entre recolección y análisis elegir primero recolección y después análisis.
- Recoger la información según el orden de volatilidad (de mayor a menor).
- Tener en cuenta que por cada dispositivo la recogida de información puede realizarse de distinta manera.

- **Orden de volatilidad antes de apagar! (si es posible):**
  - El orden de volatilidad hace referencia al período de tiempo en el que está accesible cierta información. Es por ello que se debe recolectar en primer lugar aquella información que vaya a estar disponible durante el menor período de tiempo, es decir, aquella cuya volatilidad sea mayor.
  - De acuerdo a esta escala se puede crear la siguiente lista en orden de mayor a menor volatilidad:
    - Registros y contenido de la caché.
    - Tabla de enrutamiento, caché ARP, tabla de procesos, estadísticas del kernel, memoria.
    - Información temporal del sistema.
    - Disco.
    - Logs del sistema.
    - Configuración física y tipología de la red.
    - Documentos.

VOLCADOS DE MEMORIA

- Un volcado de memoria, o *memory dump*, en inglés, es una instantánea del estado interno de un programa, ya sea una aplicación en modo usuario o el núcleo del sistema operativo. En otros sistemas se emplea más el término *core dump* con un significado similar o equivalente. El estado interno comprende, entre otros, los valores de los registros "visibles" del procesador y porciones significativas del espacio de direcciones de memoria que incluyen código, datos y pila.
- Los volcados de memoria se pueden clasificar atendiendo al tipo de información capturada o a su extensión (nivel de detalle). Así podemos tener volcados de memoria de una aplicación o del sistema, y volcados pequeños, completos o solamente del núcleo del sistema operativo.

VOLCADO DE MEMORIA EN LINUX

- git clone https://github.com/584ensicsLabs/LiMe.git
- ls -R -l Lime
- uname -r (versión kernel)
- sudo apt-get install make build-essential linux-headersxxxx
- cd Lime/src
- make
- sudo insmod lime-xxxx-generic.ko "path=/home/forense/MemLub1484 format=raw"
- ls -l/home/forense
- Extraer el fichero, firmar
- Visualizar con dick investigator.

VOLCADO DE MEMORIA DE WINDOWS

- **Uso de volatility**

Descargar volatility:

1. volatiXXXXX imageinfo -f *imagen.raw* **(imatge)**
2. volatiXXXXX - profile= *suggest* pslist -f*destination memory* **(processos)**
3. volatiXXXXX -profile= *suggest* kdbgscan -f*detination memory* **(kernel debug scan)**
4. volatiXXXXX -profile= *suggest* kpcrscan -f*detination memory* **(kernel procesor region scan)**
5. volatiXXXXX -profile= *suggest* psscan -f*detination memory* **(list services)**

1. volatiXXXX -profile= *suggest* dlllist -f *detination memory* **(llistar dll)**
2. volatiXXXXX -profile= *suggest* dlllist -p 2484 -f*detination memory* **(llistar dll per procés)**
3. volatiXXXXX -profile= *suggest* dlldump -D*destination directory* -f *detination memory* **(volcado de DLL, creamos antes un directorio)**

- Podemos ver las dll dentro del directorio
- Incluso podemos volcar archivos DLL de procesos específicos si descubrimos que puede haberse ejecutado un proceso malicioso.
- Del mismo modo, podemos volcar archivos DLL de un proceso oculto utilizando su dirección de desplazamiento como se muestra a continuación.

- Aquí hay una lista de todos los procesos ocultos una vez más. Ahora hemos utilizado la dirección de desplazamiento para smss.exe, que es 0x024f1020, y volcamos las DLL en la carpeta llamada Oculto.

1. **volatiXXXXX -profile= *suggest* pstree -f*destination memory* (processos por arbol)**
2. **volatiXXXXX -profile= *suggest* consoles -f*detination memory* (consola)**

- Este complemento se utiliza para encontrar los diversos comandos escritos localmente o remotamente a través de puertas traseras

1. volatiXXXXX -profile= *suggest* hivescan -f*destination memory* **(direcciones físicas de las colmenas de memoria)**
2. volatiXXXXX -profile= *suggest* hivelist -f*destination memory* **(direcciones virtuales de las colmenas de memoria)**
3. volatiXXXXX -profile= *suggest* svcscan -f*detination memory* **(servicios en funcionamiento)**
4. volatiXXXXX -profile= *suggest* handles -f*destination memory* **(manejadores)**

- Para mostrar los identificadores abiertos en un proceso, use el comando de identificadores. Esto se aplica a archivos, claves de registro, mutexes, canalizaciones con nombre, eventos, estaciones de ventana, escritorios, subprocesos y todos los demás tipos de objetos ejecutivos asegurables.

  volatiXXXXX -profile= *suggest* getsids -f*detination memory* **(secure id)**

- Para ver los SID (identificadores de seguridad) asociados con un proceso, sirve para identificar procesos que han escalado maliciosamente los privilegios.

  volatiXXXXX -profile= *suggest* cmdscan -f*detination memory* **(console escaner)**

- El complemento cmdscan busca en la memoria de csrss.exe en XP/2003/Vista/2008 y conhost.exe en Windows 7 los comandos que los atacantes ingresaron a través de una consola (cmd.exe).

  volatiXXXXX -profile= *suggest* envars -f*detination memory* **(variables de entorno de proceso)**

  volatiXXXXX -profile= *suggest* verinfo -f*detination memory* **(archvos PE)**

- Para mostrar la información de la versión incrustada en los archivos PE, use el comando verinfo. No todos los archivos PE tienen información de versión, y muchos autores de malware falsifican que incluya datos falsos, pero este comando puede ser muy útil para identificar binarios y hacer correlaciones con otros archivos.

  volatiXXXXX -profile= *suggest* memmap -f*detination memory* **(mapa de memoria)**

- El comando memmap le muestra exactamente qué páginas residen en la memoria, dado un proceso específico DTB (o kernel DTB si usa este complemento en el proceso inactivo o del sistema). Le muestra la dirección virtual de la página, el desplazamiento físico correspondiente de la página y el tamaño de la página. La información de mapa generada por este complemento proviene del método *getavailableadresses* del espacio de direcciones subyacente.

  volatiXXXXX -profile= *suggest* vadwalk -f*detination memory* (procesos de los drivers)

ALGUNAS HERRAMIENTAS ÚTILES

- Para garantizar que estamos hablando de evidencias judiciales debe haber un conjunto de varios factores. Uno de ellos es el conocimiento de los peritos judiciales informáticos como la aplicación de las herramientas forenses. Las técnicas que se utilizan para la metodología, se puede extraer cualquier evidencia que nos señale una acción o suceso. Los peritos aportan una experiencia en una materia, en este caso, es la ciencia de la informática. Con este saber podemos ayudar a la justicia o a cualquier ciudadano u organismo que nos solicite ayuda.
- Existen varias herramientas para la obtención de evidencias:
    - Clonadores de discos (hw y sw).
    - Distros live forensic.
    - Scripts y api's extracción de información.

ANÁLISIS DE RED

- **Snort**: Sistema de detección de intrusiones basados en red. Contiene un "engine" propio de detección. Trabaja mediante filtros predeterminados o pre-programados.
- **Wireshark**: Análisis de protocolos de red y tráfico de la misma.
- **Nmap**: Analizador de puertos y servicios, también utilizado en auditorias. Permite mucha integración.
- **Xplico**: Es un framework forense para realizar tareas de análisis de datos recopilados en capturas de red.

Tratamiento de unidades de disco:

- **Dcdd3**: permite copiar grandes imágenes de discos en partes más pequeñas para un traslado y posterior análisis más cómodo.
- **Mount manager:** permite detectar, montar y desmontar, examinar y administrar unidades de almacenamiento conectadas a disco duro.
- **Ghost y GuyManager:** Clonado de discos bit a bit, rápido y eficaz.

GESTIÓN Y ANÁLISIS DE MEMORIA

- **Volatility**: Permite hacer volcados de memoria de máquinas con sistemas perativos Windows, Linux, Mac OSX e incluso Android. Trabaja con versiones tanto de 32 como de 64 bits.
- **RedLine**: Captura de memoria y análisis mediante entorno gráfico.
- **Memoryze**: Permite la captura de memoria RAM en equipos con sistemas operativos Windows y OSX.

ANÁLISIS DE APLICACIONES Y SERVICIOS

- **OfficeMalScanner**: permite escanear archivos de la suitOffice, en busca de códigos maliciosos.
- **OllyDbg**: Carga y permite debugar DLLS's y escaneo de todo tipo de archivos.
- **Radare**: permite aplicar ingeniería inversa para analizar código de una aplicación maliciosa que se ha ejecutado.
- **Process explorer**: Muestra información de los procesos que hay abiertos en una máquina.
- **PDFStreamDumper**: Análisis de código malicioso dentro de archivos de PDF.

FRAMEWORK Y SUITES

- **DEFT**: permite el análisis forense de terminales móviles y dispositivos con Android o iOS. Contiene herramientas útiles para el análisis de ficheros de diferentes tipos, búsqueda de rookits, virus, malware, etc.

- **Caine**: un entorno de trabajo orientado a completar las fases de un análisis forense, una interfaz gráfica bastante amigable y un proceso semiautomático para generar informes a partir de los resultados obtenidos.

- **Autopsy**: es un conjunto de aplicaciones muy útiles para el análisis forense. Permite búsquedas de palabras sobre todo el equipo. Analiza el registro del sistema operativo Windows. Extrae datos EXIF de las imágenes, permite visualizar miniaturas de las mismas así como clasificación de los archivos del sistema por tipo, entre muchas otras opciones y herramientas.

- KailLinux, Parrot, BackTrack, Backbox, Santoku, BugTraq, Pentoo, Nst...