



**Escuela Técnica Superior de Ingenierías Informática y de Telecomunicaciones**  
**Máster Oficial en Ingeniería Informática**

Curso 2020/2021

# **SIMULACIÓN DE ANÁLISIS FORENSE CON HERRAMIENTAS DE KALI LINUX**

Administración de sistemas y seguridad

**Breve descripción**

**Autores**

Álvaro de la Flor Bonilla y Antonio Manuel Salvat Pérez

**Propiedad Intelectual**

Universidad de Granada

## ÍNDICE DEL PROYECTO

<b>1</b>	<b>Introducción .....</b>	<b>4</b>
1.1	Análisis forense informático .....	4
1.1.1	Definición.....	4
1.1.2	Objetivos .....	4
1.2	Herramientas .....	5
1.2.1	Noticias.....	5
<b>2</b>	<b>Desarrollo .....</b>	<b>6</b>
2.1	Herramientas .....	6
2.1.1	Autopsy .....	6
2.1.2	Binwalk .....	10
2.1.3	Bulk Extractor .....	14
2.1.4	Chkrootkit .....	16
2.1.5	Foremost .....	23
2.1.6	Scalpel.....	26
<b>3</b>	<b>Noticias.....</b>	<b>28</b>
3.1	Binwalk tool .....	28
3.2	Bulk Extractor .....	28
3.3	Scalpel Tool.....	29
3.4	Autopsy tool.....	30
3.5	Chkrootkit .....	30
<b>4</b>	<b>BIBLIOGRAFÍA.....</b>	<b>32</b>

## ÍNDICE DE ILUSTRACIONES

Ilustración 1 - Autopsy pantalla principal .....	6
Ilustración 2 - Ejecución de Autopsy .....	7
Ilustración 3 – Información extraída .....	7
Ilustración 4 - Particiones Autopsy .....	8
Ilustración 5 - Autopsy extensiones.....	9
Ilustración 6 - Autopsy búsqueda por archivo .....	9
Ilustración 7 - Autopsy búsqueda por palabras clave .....	10
Ilustración 8 - Análisis básico binwalk .....	10
Ilustración 9 - Análisis básico binwalk .....	11
Ilustración 10 - Análisis recursivo binwalk .....	11
Ilustración 11 - Binwalk archivos por contenido.....	12
Ilustración 12 - Binwalk extraer tipo determinado .....	12
Ilustración 13 - Binwalk búsqueda de archivo por línea.....	13
Ilustración 14 – Ejecución de bulk_extractor .....	15
Ilustración 15 – Ejecución completa de bulk_extractor.....	15
Ilustración 16 – Archivos extraídos.....	16
Ilustración 17 - chkrootkit instalación .....	17
Ilustración 18 - chkrootkit ejecución .....	17
Ilustración 19 - chkrootkit ejecución cómoda.....	18
Ilustración 20 - chkrootkit obtener resultados .....	18
Ilustración 21 - chkrootkit visualizar resultados .....	19
Ilustración 22 - chkrootkit ejecución automatizada .....	19
Ilustración 23 - instalación rkhunter.....	20
Ilustración 24 - ejecución rkhunter .....	20
Ilustración 25 - rkhunter escaneo librerías .....	21
Ilustración 26 - rkhunter escaneo rootkits.....	21
Ilustración 27 - rkhunter escaneo rootkits, troyanos, malwares .....	22
Ilustración 28 - rkhunter escaneo interfaces de red.....	22
Ilustración 29 - rkhunter resultado.....	23
Ilustración 30 - Foremost instalación.....	24

Ilustración 31 - Foremost obtener IDs .....	25
Ilustración 32 - Foremost analizar unidad .....	25
Ilustración 33 - Foremost reemplazo de archivos.....	26
Ilustración 34 - Foremost resultado en carpeta .....	26
Ilustración 35 – Ejecución de Scalpel.....	27
Ilustración 36 – Archivo resumen .....	27
Ilustración 37 – Vulnerabilidad de Router Cisco.....	28
Ilustración 38 – Cámara IP .....	28
Ilustración 39 – Recopilación de noticias .....	29
Ilustración 40 – Recuperación de archivos.....	29
Ilustración 41 – Borrado de información.....	30
Ilustración 42 – Rootkit de Sony .....	30

# 1 INTRODUCCIÓN

## 1.1 Análisis forense informático

### 1.1.1 Definición

El análisis forense informático es una disciplina que combina en cierta manera tanto el derecho en sí, como por su puesto la informática.

Básicamente, a modo resumen lo que trata de hacer es recopilar y analizar los datos de sistemas informático, redes, comunicaciones inalámbricas, dispositivos de almacenamiento... hasta conseguir elaborar un informe suficientemente sólido para que sea admisible como prueba en un tribunal de justicia.

En el campo de la informática Forense existen diversas etapas que definen la metodología a seguir en una investigación: identificación, preservación o adquisición, análisis y presentación de los resultados. Siguiendo el flujo de las actividades primero identificamos las fuentes a analizar, posteriormente se realiza el análisis para extraer información valiosa, y finalmente se presentan los resultados.

Dentro de la informática forense tenemos varios tipos:

- De sistemas operativos: es el proceso de recuperación útil del sistema operativo, el objetivo es adquirir evidencia contra el autor.
- De redes: se refiere a la recopilación, monitorización y análisis de las actividades de la red para descubrir la fuente de ataques, virus, intrusiones o violaciones de seguridad en la red. También se usa para recopilar pruebas mediante el análisis de datos de tráfico de red para identificar la fuente de un ataque.
- En dispositivos móviles: el proceso forense móvil tiene como objetivo recuperar evidencia digital o datos relevantes de un dispositivo móvil de una forma que conserve la evidencia en una condición forense sólida.
- En la nube o Cloud: actualmente la mayoría de los datos críticos de las empresas se encuentran en la nube. El análisis forense combina la computación en la nube y el análisis forense digital que se centra en la recopilación de información digital de una infraestructura en la nube, esto significa que trabajamos con una colección de recursos informáticos como activos de red, servidores, aplicaciones y cualquier servicio que se brinde.

### 1.1.2 Objetivos

El análisis forense del que hablamos va más allá de intentar localizar posibles fraudes o delitos.

Podemos señalar, por ejemplo, intentar ser capaces de analizar las consecuencias a las que se puede comprometer un sistema mal protegido.

Si el delito ya se ha producido, el analista intentará averiguar quién ha sido el autor de este ataque, así como las causas y la metodología empleada.

En definitiva, localizar las debilidades existentes que permiten localizar vulnerabilidades del sistema, producto afectado o identificar el origen y metodología del ataque.



## **1.2 Herramientas**

En este documento pretendemos realizar un análisis de las principales herramientas empleadas actualmente en procesos de análisis forenses. Para cada una de ellas realizaremos una breve descripción de estas.

Además, para cada una de estas herramientas realizaremos un breve resumen, explicando las principales características y funcionalidades detallando el resultado obtenido al aplicarlas.

### **1.2.1 Noticias**

Respecto a cada una de las distintas herramientas, mostraremos noticias relacionales en las que ha sido factible la utilización de ellas para resolución de casos de especial interés relativamente actuales.

## 2 DESARROLLO

### 2.1 Herramientas

#### 2.1.1 Autopsy

Autopsy es la interfaz gráfica del conjunto The Sleuth Kit que es un conjunto de herramientas open source para el análisis de imágenes de discos multiplataforma.

Cuando abrimos Autopsy la primera tarea es crear un caso nuevo o abrir uno ya existente. Lo siguiente será asociar el origen de datos donde estarían los discos físicos conectados al equipo o una imagen forense que tengamos dentro del equipo. Por último, tendríamos que configurar los módulos a utilizar como vemos en la siguiente imagen:

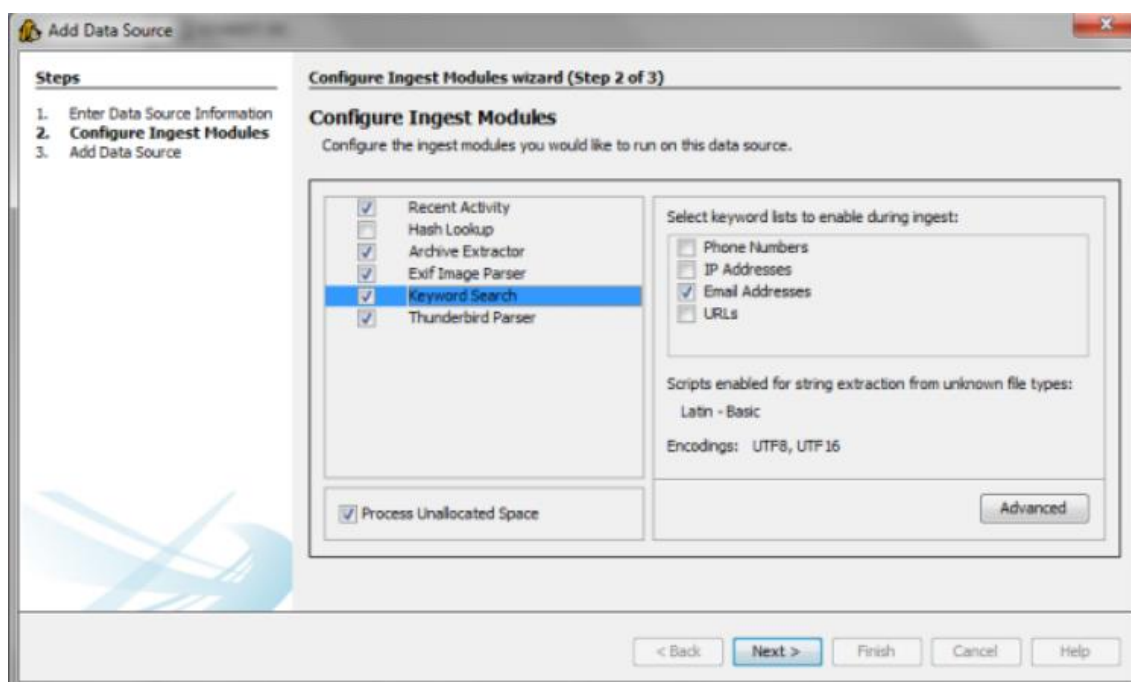


Ilustración 1 - Autopsy pantalla principal

Estos serían los módulos de información relevante:

- **Recent Activity:** extrae la actividad reciente que se ha obtenido en el equipo. Esto incluye los documentos recientemente abiertos, dispositivos conectados, historial web, cookies, y descargas, por ejemplo.
- **Hash Lookup:** nos permite agregar bases de datos con valores de hash para archivos conocidos, como los archivos del sistema operativo o de aplicaciones instaladas.
- **Archive Extractor:** nos permite recuperar archivos eliminados, mediante los metadatos residuales que quedan en el disco, así como también recuperar archivos en espacios no asignados en el disco, mediante la detección de los encabezados de archivos.
- **Exif Image Parser:** permite analizar la información disponible en el encabezado 'Exif' de los archivos de imagen JPEG que se encuentran en el disco. Esto provee información acerca de la cámara con que se tomó la imagen, fecha y hora o la geolocalización, entre otras cosas.

- **Keyword Search:** puede definirse una lista de palabras clave o expresiones regulares a buscar en todo el disco. Como se observa en la imagen anterior, Autopsy ya viene con una lista de expresiones regulares incluidas para búsqueda de números telefónicos, direcciones IP, direcciones de correo electrónico y URL

Una vez seleccionamos los módulos empieza el análisis de forma automática como vemos en esta imagen:

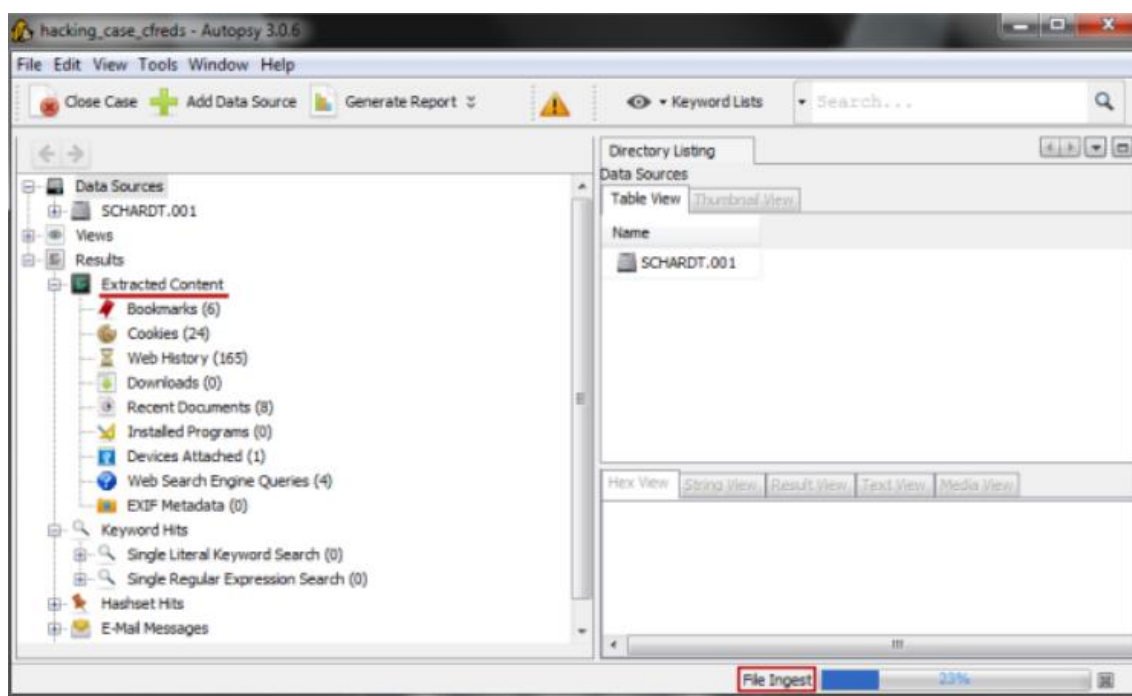


Ilustración 2 - Ejecución de Autopsy

Podemos observar que Autopsy extrae información valiosa para una posible investigación como por ejemplo el historial web, búsquedas web o documentos abiertos recientemente, aquí podemos ver un ejemplo de la extracción del historial de un disco:

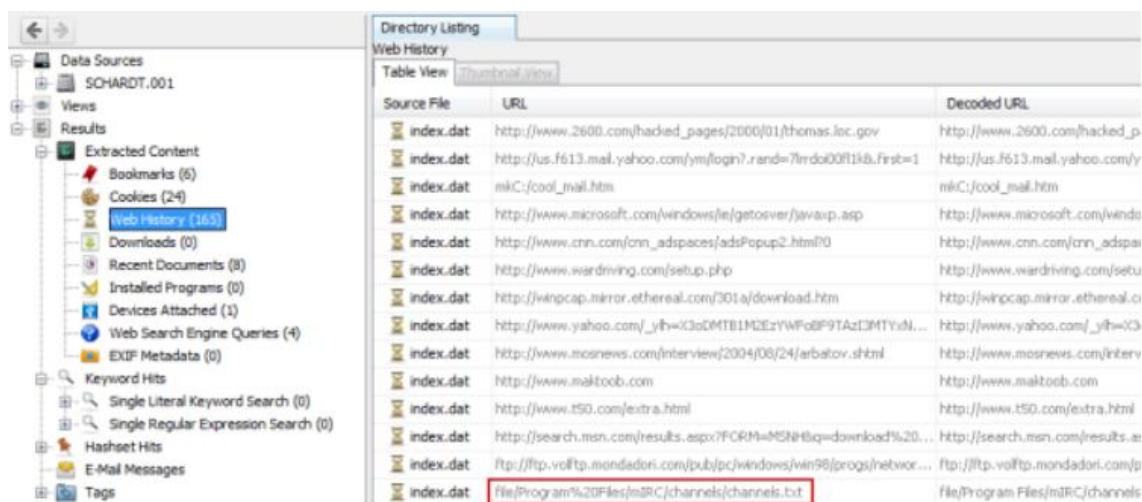


Ilustración 3 – Información extraída



En este caso se ha resaltado el último resultado, el cual lleva a un archivo de mIRC con una lista de canales a los que posiblemente se conectaba el usuario. Además, el sistema de archivos presente en la imagen forense puede ser recorrido de forma jerárquica, pudiendo observar las particiones, así como también los sectores no asignados en el disco:

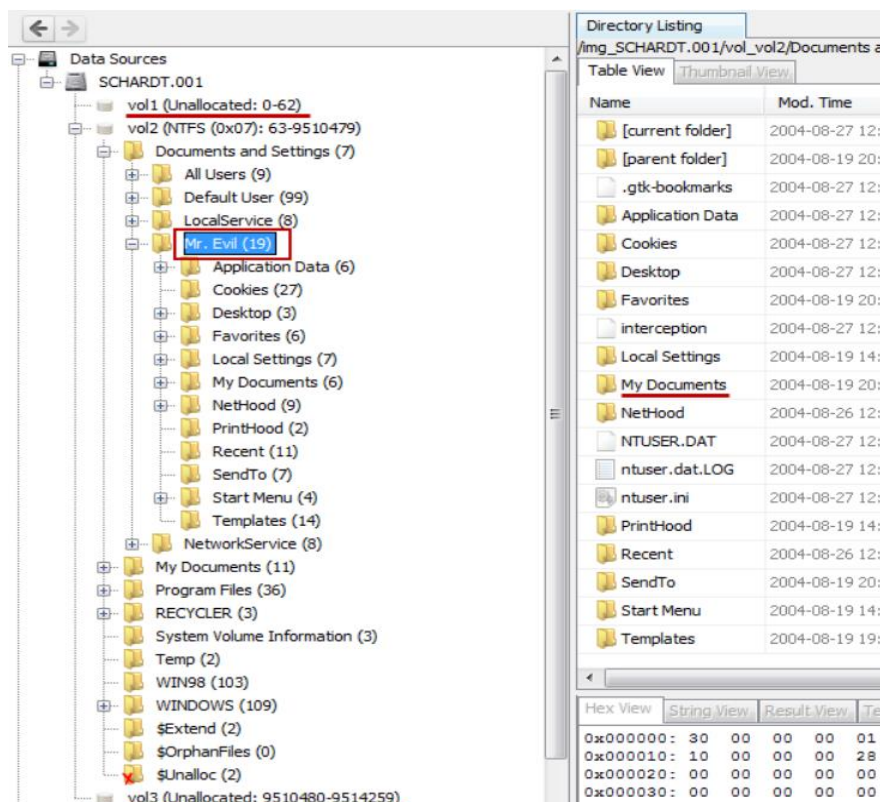


Ilustración 4 - Particiones Autopsy

En la anterior imagen vemos que el sistema de archivos es de tipo NTFS, que parece ser que el sistema operativo del equipo es Windows XP y que existe un usuario que se llama MR. Evil. Tras una inspección rápida del registro de Windows podemos obtener la zona horaria, el nombre mediante el cual se registró el sistema operativo, las aplicaciones instaladas, los documentos del usuario. También podemos ver los sectores no asignados en el disco a los cuales Autopsy puede aplicar la técnica 'carving' para identificar archivos a partir de los encabezados.

Otra función interesante de Autopsy es la que permite agrupar los archivos en categorías, lo que nos permite ver la cantidad de imágenes, audio o documentos presentes en el sistema, y a su vez podemos clasificarlas por extensión:







Directory Listing		
File Types		
Table View Thumbnail View		
Filter Type	File Extensions	Name
 Images (1169)	'.jpg', '.jpeg', '.png', '.psd', '.nef', '.tiff', '.bmp'	
 Videos (34)	'.aaf', '.3gp', '.asf', '.avi', '.m1v', '.m2v', '.m4v', '.mp4', '.mov', '.mpeg'...	
 Audio (146)	'.aiff', '.aif', '.flac', '.wav', '.m4a', '.ape', '.wma', '.mp2', '.mp1', '.mp3', ...	
 Archives (282)	'.zip', '.rar', '.7zip', '.7z', '.arj', '.tar', '.gzip', '.bzip', '.bzip2', '.cab', '.jar'...	
 Documents		TSK_...
 Executable		TSK_...

Ilustración 5 - Autopsy extensiones

Otra técnica que nos da Autopsy para el análisis forense es la búsqueda por palabras claves ya que es poco práctico ir archivo por archivo buscando como vemos en la siguiente imagen:

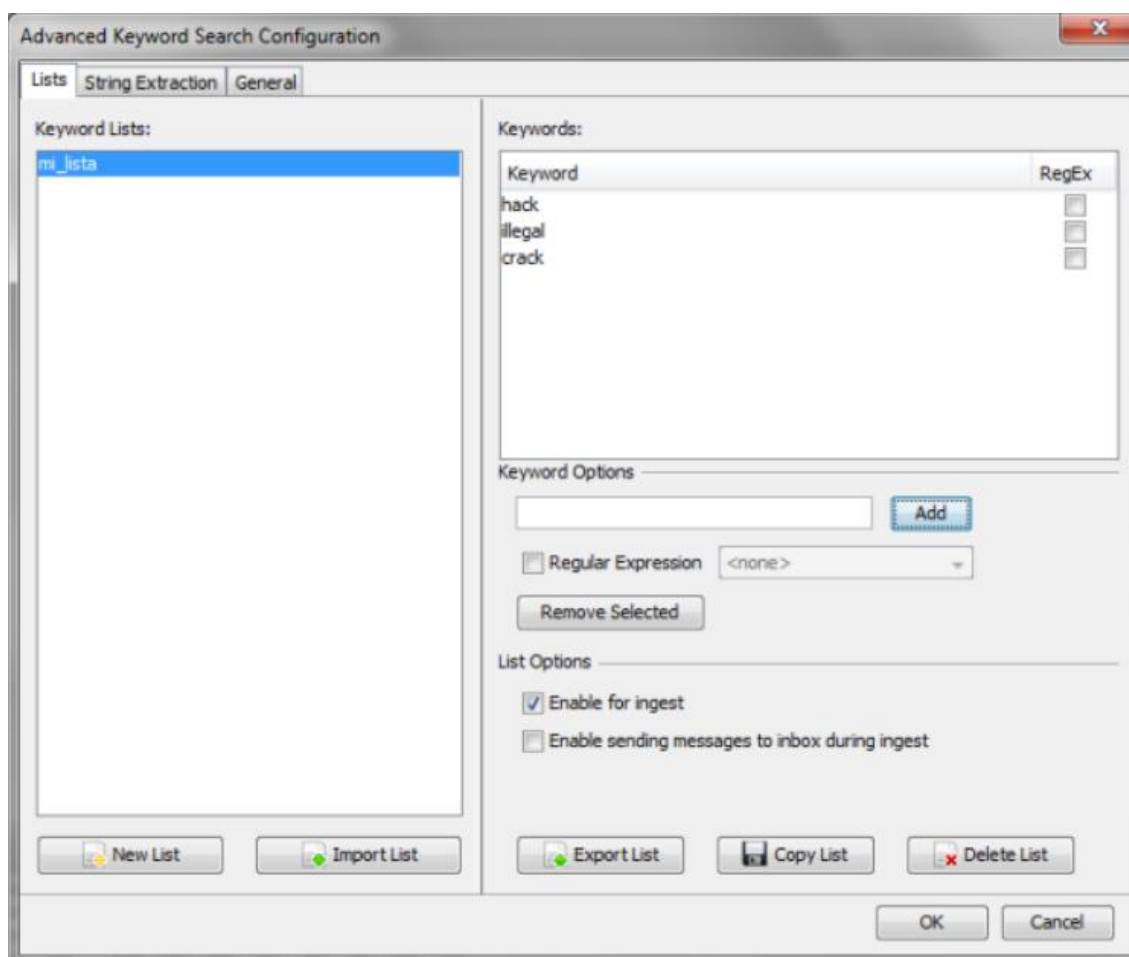


Ilustración 6 - Autopsy búsqueda por archivo

En este caso se ha definido una lista con tres palabras con el objetivo de encontrar evidencia que conecte al individuo con un caso particular de hacking. Luego, el proceso

de indexado realiza la búsqueda de estas palabras construyendo un índice y al finalizar el mismo es posible buscar en forma inmediata cada una de estas palabras clave:

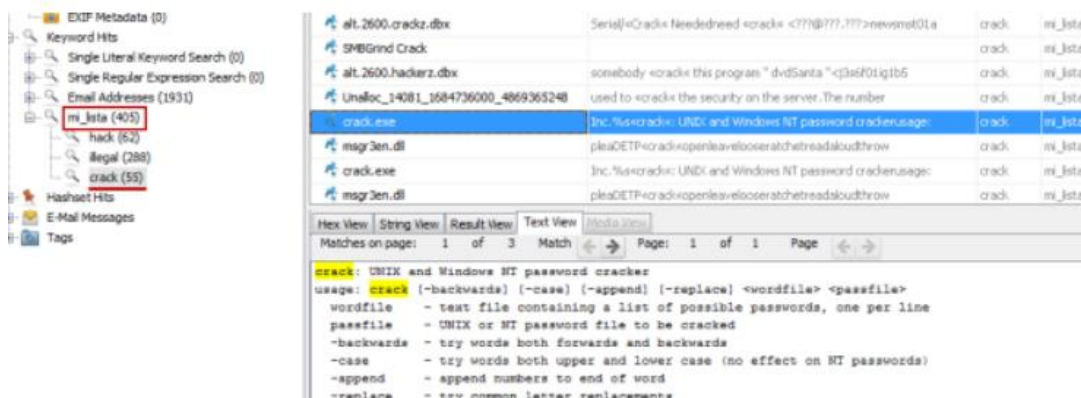


Ilustración 7 - Autopsy búsqueda por palabras clave

En la imagen se observa que hay 55 hits para la palabra crack y entre los resultados se encuentra una herramienta para romper contraseñas. Además, se han encontrado 288 coincidencias para la palabra ilegal, lo cual puede develar información muy útil en el caso de que se recuperen conversaciones, correos electrónicos almacenados o cualquier otro tipo de archivo que conecte al sujeto investigado con actividades ilegales.

Para finalizar este análisis es importante mencionar que los resultados obtenidos pueden ser exportados a documentos HTML, entre otros formatos ofrecidos, para la presentación de lo encontrado en cualquier otro equipo que no cuente con Autopsy. En este post se ha utilizado una imagen forense de prueba del proyecto CFReDS para los análisis realizados; en particular, esta imagen contiene evidencia con la cual se debe probar que el sujeto está involucrado en la captura ilegal de paquetes de red y el robo de contraseñas. Se observa que las posibilidades en el análisis son muy variadas, pudiendo descubrir información oculta en el sistema de archivos, recuperando archivos eliminados, o encontrando palabras clave en chats o correos electrónicos.

### 2.1.2 Binwalk

Binwalk es una herramienta destinada a la extracción de datos de imágenes binarias de firmware (conocido como 'Firmware Hacking').

Este software interactúa directamente con el hardware, por lo que se trata de código de solo-lectura y también otros como por ejemplo los 'routers'.

Para los ejemplos mostrados se ha usado el firmware CT-536B+-A101-302JAZ-C03\_R21.

Ahora veremos las funcionalidades más básicas:

- binwalk -B firmware.bin

```
rekod@rekod-pc:~/Escritorio/CT-536B+-A101-302JAZ-C03_R21$ binwalk firmware.bin
```

DECIMAL	HEXADECIMAL	DESCRIPTION
0	0x0	Broadcom 96345 firmware header, header size: 256, firmware version: "8", board id: "6348GW-11", -CRC32 header che
256	0x100	Squashfs filesystem, big endian, version 2.0, size: 1635961 bytes, 333 inodes, blocksize: 65536 bytes, created: 2

009-08-13 09:26:22

Ilustración 9 - Análisis básico binwalk

Este sería el análisis más básico.

- binwalk -Me firmware.bin

```
rekod@rekod-pc:~/Escritorio/CT-536B+-A101-302JAZ-C03_R21$ binwalk -Me firmware.bin
```

Scan Time: 2017-02-16 16:02:23  
 Target File: /home/rekod/Escritorio/CT-536B+-A101-302JAZ-C03\_R21/firmware.bin  
 MD5 Checksum: 8a2826da67bb932f56107d4b27cd41a9  
 Signatures: 374

DECIMAL	HEXADECIMAL	DESCRIPTION
0	0x0	Broadcom 96345 firmware header, header size: 256, firmware version: "8", board id: "6348GW-11", -CRC32 header che
256	0x100	Squashfs filesystem, big endian, version 2.0, size: 1635961 bytes, 333 inodes, blocksize: 65536 bytes, created: 2

009-08-13 09:26:22




Ilustración 10 - Análisis recursivo binwalk

Con esta opción extraemos de forma recursiva los archivos que están detrás de la imagen binaria, es decir ingeniería inversa.

- binwalk -Mer firmware.bin

Si quisiéramos realizar lo mismo, pero quedándonos con archivos con contenido.

- binwalk -B -W firmware.bin -f data

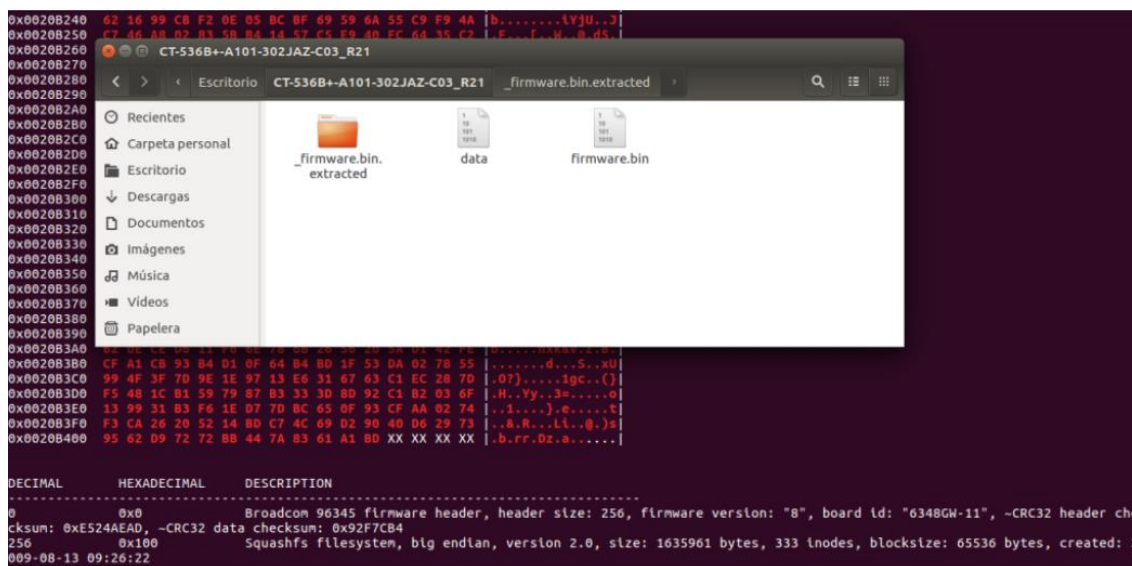


Ilustración 11 - Binwalk archivos por contenido

Sería un volcado de datos en formato hexadecimal del archivo binario a un archivo de nombre 'data'.

- binwalk -B -W -l 100 firmware.bin (mostrará los 100 primeros bytes en formato hex)

Para volcar datos de forma limitada tenemos el comando 'length', el cual se mide en bytes

Si queremos extraer un tipo determinado de datos:

- binwalk -D 'png image:png' firmware.bin
- binwalk -D 'zip archive:zip:unzip' firmware.bin
- binwalk --dd='squashfs:squashfs' firmware.bin

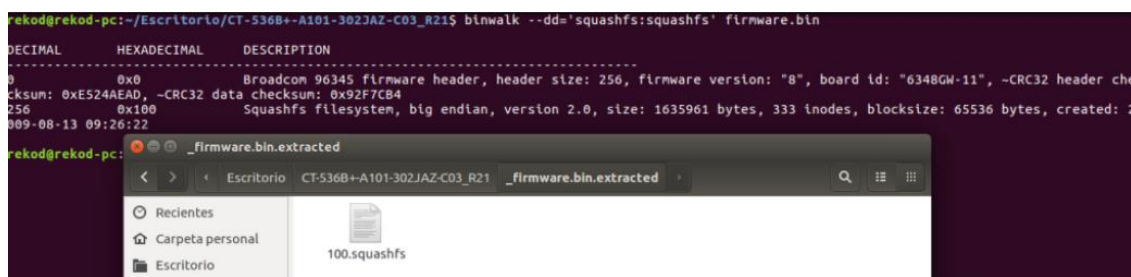


Ilustración 12 - Binwalk extraer tipo determinado

Para buscar un archivo en una línea determinada:

- binwalk -R "\x100" firmware.bin



DECIMAL	HEXADECIMAL	DESCRIPTION
6800	0x1A90	\x100
9503	0x251F	\x100
114128	0x1BDD0	\x100
149630	0x2487E	\x100
295608	0x482B8	\x100
334575	0x51AEF	\x100
342749	0x53ADD	\x100
381473	0x5D221	\x100
390751	0x5F65F	\x100
417841	0x66031	\x100
521492	0x7F514	\x100
530062	0x8168E	\x100
535652	0x82C64	\x100
557497	0x881B9	\x100
601960	0x92F68	\x100
786804	0xC0174	\x100
854863	0xD084F	\x100
1108141	0x10E8AD	\x100
1156613	0x11A605	\x100
1266731	0x13542B	\x100
1283859	0x139713	\x100
1298117	0x13CEC5	\x100
1345370	0x14875A	\x100
1369406	0x14E53E	\x100
1404761	0x156F59	\x100
1428976	0x15CDF0	\x100
1434551	0x15E3B7	\x100
1593234	0x184F92	\x100
1689601	0x19C801	\x100
1702905	0x19FBF9	\x100
1734333	0x1A76BD	\x100
1767031	0x1AF677	\x100
1771740	0x1B08DC	\x100
1808628	0x1B98F4	\x100
1834974	0x1BFFDE	\x100
1899499	0x1CFBEB	\x100
1982464	0x1E4000	\x100
2017824	0x1ECA20	\x100

Ilustración 13 - Binwalk búsqueda de archivo por línea

Para localizar código ejecutable o saber la arquitectura del archivo ejecutable:

- binwalk -A firmware.bin

En el caso de que sepamos que queremos localizar podemos crear un archivo con firmas digitales y usarlo para el escaneo:

- binwalk -m ./file.mgc firmware.bin
- binwalk -m ./file.txt firmware.bin

Si queremos escanear archivos con nombres predeterminados que se encuentran en un archivo:

- binwalk --include='.bin\$' firmware.bin
- binwalk -M -e --include='.bin\$' firmware.bin

Tras ver todo esto podemos ver las grandes posibilidades que nos dan los archivos binarios, de hecho, ahora esta de moda el llamado 'Firmware Hacking' ya que la mayoría de los dispositivos llevan instalado un 'Firmware', esto nos permite desde entrar en 'routers' hasta coches o dispositivos de domótica.

Normalmente se usa en tres escenarios: crear backdoors, exploits o descubrir posibles funcionalidades que no estaban previstas en el dispositivo.

Por ejemplo, si nuestro router estuviera hackeado significaría que posee una backdoor que proporciona un acceso a terceros no autorizados, para saber si esto es así

deberíamos descargar el firmware actual del router, realizar un análisis con la herramienta Binwalk para extraer los datos con:

```
binwalk -dd='squashfs:squashfs' -offset='256' firmware.bin -f data.squashfs
```

Una vez hecho esto buscaríamos en el directorio /etc un archivo llamado 'shadow' o 'passwd'. De manera predeterminada solo debe haber una cuenta root con una contraseña encriptada, si vemos que hay varias cuentas podemos tener por seguro que alguien no autorizado tiene acceso al router.

### 2.1.3 Bulk Extractor

Bulk Extractor es otra de las herramientas más utilizadas para el análisis forense informático. Entre sus principales características destaca que es capaz de extraer información útil sin analizar el sistema de archivos. Es decir, es capaz de analizar una imagen de disco, archivo o un directorio de archivos y extraer su información útil sin necesidad de analizar las estructuras del sistema de archivos.

Lo más curioso de esta herramienta es que permite aplicarla incluso sobre ficheros dañados o que cuenten con datos comprimidos.

En concreto, su uso se centra en intentar extraer toda la información que cuente el sistema de archivos analizado. Alguna de los principales elementos que es capaz de localizar (no todos) son:

- Extraer números de tarjetas de crédito.
- Enlaces a URLs.
- Direcciones IPs.
- Direcciones MAC.
- Correos utilizados.
- Reportes sobre paquetes TCP
- Direcciones telefónicas

El uso de esta herramienta es mucho más sencillo comparada con casos anteriores. Su ejecución se centra en utilizar el siguiente comando, aunque se puede personalizar con la inserción de algunos comandos que permiten la configuración por ejemplo del número de hilos usados, tamaño de los archivos, tiempo máximo...

```
$ bulk_extractor -o <folder_name> <adress>
```

El parámetro "*folder\_name*" indica la carpeta en la que van a ser guardados los archivos que han sido extraídos, mientras que el parámetro "*adress*" hace referencia al sistema de archivos que va a ser analizado.

A continuación, vamos a realizar un ejemplo práctico en el que mostraremos el resultado de ejecutar esta herramienta sobre el directorio principal del sistema.

```
lecturesnippets@lecturesnippets-ubuntu:~/Desktop$ sudo bulk_extractor -o output /dev/sdb1
bulk_extractor version: 1.3.1
Hostname: lecturesnippets-ubuntu
Input file: /dev/sdb1
Output directory: output
Disk Size: 64422412288
Threads: 1
16:41:20 Offset 0MB (0.00%) Done in n/a at 16:41:19
```

Ilustración 14 – Ejecución de bulk\_extractor

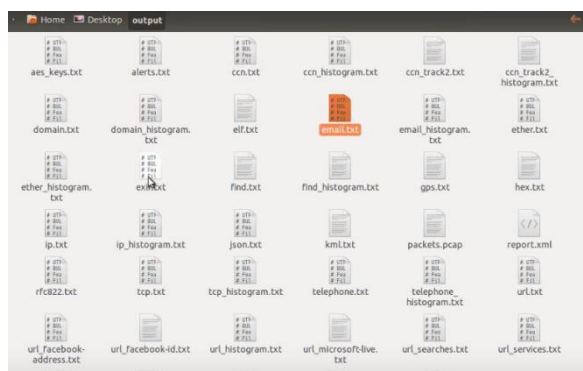
Como podemos ver en la imagen anterior, el comando ha sido ejecutado sobre el directorio “/dev/sdb1” y se le indica que todo archivo extraído se añada a la carpeta “output” que se localiza en la carpeta escritorio del sistema.

```
23:16:10 Offset 64357MB (99.90%) Done in 0:00:23 at 23:16:33
All Data is Read; waiting for threads to finish...
Time elapsed waiting for 1 thread to finish:
(timeout in 60 min .)
All Threads Finished!
Producer time spent waiting: 22724.2 sec.
Average consumer time spent waiting: 0.429018 sec.
*****
** bulk_extractor is probably CPU bound. **
** Run on a computer with more cores **
** to get better performance. **
*****
Phase 2. Shutting down scanners
Phase 3. Creating Histograms
ccn histogram... ccn_track2 histogram... domain histogram...
email histogram... ether histogram... find histogram...
ip histogram... tcp histogram... telephone histogram...
url histogram... url microsoft-live... url services...
url facebook-address... url facebook-id... url searches...

Elapsed time: 2.372e+04 sec.
Overall performance: 2.716 MBytes/sec.
Total email features found: 2043
```

Ilustración 15 – Ejecución completa de bulk\_extractor

Como puede verse en la imagen superior es un proceso bastante complejo y arduo que puede llegar a consumir bastante tiempo, de hecho, en este caso tardó algo más de 23 horas para finalizar completamente.



```
# histogram file version: 1.1
n=63 administrator@www.ac (utf16=63)
n=46 premium-server@thawte.com (utf16=4)
n=21 info@valicert.com (utf16=2)
n=17 administrator@at.at (utf16=17)
n=17 support@accessdata.com (utf16=17)
n=14 cps-requests@verisign.com
n=9 administrator@cl.at (utf16=9)
n=5 administrator@marketing.ac (utf16=5)
n=5 rico@ricostacruz.com (utf16=5)
n=4 txtadministrator@www.ac (utf16=4)
n=3 administrator@addthis.com
n=3 jeffsmith@redmond.corp.microsoft.com (utf16=3)
n=3 vshubin@ntdev.microsoft.com
n=2 administrator@www.ge (utf16=2)
n=2 eay@cryptsoft.com
n=2 gtk-devel-list@gnome.org
n=2 info@prof-uis.com (utf16=2)
n=2 meishui981@gmail.com
n=1 l.txtadministrator@at (utf16=1)
n=1 administrator@adobe.de (utf16=1)
n=1 administrator@ac.at (utf16=1)
n=1 administrator@ac.bi (utf16=1)
n=1 administrator@ad.ad (utf16=1)
n=1 administrator@ox-d.ad (utf16=1)
```



## Ilustración 16 – Archivos extraídos

En las dos imágenes anteriores se muestra el proceso final, en el que podemos ver cada uno de los archivos finales extraídos en el caso de la izquierda, y la muestra del archivo de correos utilizados en el caso de la derecha.

Nos ha sorprendido mucho su potencial. De hecho, hemos leído en la documentación que es capaz de extraer números de tarjetas bancarias a partir de un análisis de lectura RAM.

### 2.1.4 Chkrootkit

Esta herramienta nos permite localizar 'rootkits', estos serían herramientas maliciosas que garantizan acceso o privilegios a un sistema. Una característica importante de las 'rootkits' es que ocultan su presencia muy bien. En algunos casos podrían llegar incluso a corromper el 'kernel' de un sistema o incluso el hardware del equipo.

Si detectamos un 'rootkit' se asume que se debe reinstalar el sistema incluso en algunos casos extremos reemplazar el hardware. Otro problema es la cantidad de falsos positivos que da esta herramienta por lo que se suele usar junto a la herramienta 'rkhunter'.

Chkrootkit incluye las siguientes herramientas:

- Chkrootkit: programa principal que examina los binarios del sistema operativo en busca de modificaciones hechas por 'rootkits' para saber si un código fue cambiado.
- Ifpromisc.c: examina si las interfaces están en modo promiscuo, es decir, si una interfaz de red está en este modo puede ser usada por un posible atacante o un software malicioso para capturar tráfico de la red.
- chklastlog.c: esta herramienta revisa lo que fue eliminado de lastlog. Lastlog es un comando que nos muestra información sobre los logins del sistema. Una rootkit podría modificarlos para evitar su detección.
- Chkwtmp.c: es similar al anterior, este script examina el archivo wtmp que también tiene información sobre los logins del sistema, por lo que esta herramienta busca modificaciones en este archivo.
- check\_wtmpx.c: igual que el anterior pero solo para sistemas Unix Solaris.
- chkproc.c: busca indicios de troyanos en LKM (Loadable Kernel Modules).

- `chkdirs.c`: es similar al anterior, pero examina troyanos dentro de los módulos del Kernel.

Ahora veremos cómo detectar rootkits. Para instalarlo usaremos el siguiente comando:

***\$ apt install chkrootkit -y***

```
root@montsegur:/home/linuxlat# apt install chkrootkit -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libmicrodns0 libminizip1 libmpv1 libqt5test5 libqt5webengine-data
  libqt5webengine5 libqt5webenginecore5 libre2-5 linux-image-4.19.0-6-amd64
  python-apt qml-module-qt-labs-platform qml-module-qtwebchannel
  qml-module-qtwebengine
Use 'apt autoremove' to remove them.
The following NEW packages will be installed:
  chkrootkit
0 upgraded, 1 newly installed, 0 to remove and 37 not upgraded.
Need to get 293 kB of archives.
After this operation, 956 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian buster/main amd64 chkrootkit amd64 0.52-3+b10
[293 kB]
Fetched 293 kB in 1s (220 kB/s)
Preconfiguring packages ...
Selecting previously unselected package chkrootkit.
(Reading database ... 376058 files and directories currently installed.)
Preparing to unpack .../chkrootkit_0.52-3+b10_amd64.deb ...
Unpacking chkrootkit (0.52-3+b10) ...
Setting up chkrootkit (0.52-3+b10) ...
```

Ilustración 17 - chkrootkit instalación

Una vez instalado para ejecutarlo solo necesitamos ejecutar el siguiente comando:

***\$ sudo chkrootkit***

```
Searching for anomalies in shell history files...      nothing found
Checking `asp'...                                    not infected
Checking `bindshell'...                              not infected
Checking `lkm'...                                     chkproc: nothing de
tected
chkdirs: nothing detected
Checking `rexedcs'...                                 not found
Checking `sniffer'...                                 lo: not promisc and
no packet sniffer sockets
wlp3s0: PACKET SNIFFER(/usr/sbin/dhclient[14526], /usr/sbin/wpa_supplicant[773]
, /usr/sbin/wpa_supplicant[773])
Checking `w55808'...                                  not infected
Checking `wted'...                                    chkwtmpt: nothing de
leted
Checking `scalper'...                                 not infected
Checking `slapper'...                                 not infected
Checking `z2'...                                       chklastlog: nothing
deleted
Checking `chkutmp'...                                  The tty of the fol
lowing user process(es) were not found
in /var/run/utmp !
! RUID      PID TTY    CMD
! linuxhi+  14958 pts/0  bash
! linuxhi+  14964 pts/0  su
```

Ilustración 18 - chkrootkit ejecución

En la captura podemos ver como los scripts explicados anteriormente se ejecutan uno a uno.

Para tener una vista más cómoda podemos ejecutar:

***\$ sudo chkrootkit / less***

```
ROOTDIR is '/'
Checking `amd'... not found
Checking `basename'... not infected
Checking `biff'... not found
Checking `chfn'... not infected
Checking `chsh'... not infected
Checking `cron'... not infected
Checking `crontab'... not infected
Checking `date'... not infected
Checking `du'... not infected
Checking `dirname'... not infected
Checking `echo'... not infected
Checking `egrep'... not infected
Checking `env'... not infected
Checking `find'... not infected
Checking `fingerd'... not found
Checking `gpm'... not found
Checking `grep'... not infected
Checking `hdparm'... not infected
Checking `su'... not infected
Checking `ifconfig'... not infected
Checking `inetd'... not infected
Checking `inetdconf'... not found
:
```

Ilustración 19 - chkrootkit ejecución cómoda

También podemos ver los resultados ejecutando:

***\$ sudo chkrootkit > resultados***

```
root@montsegur:/home/linuxlat# sudo chkrootkit > resultados
root@montsegur:/home/linuxlat#
```

Ilustración 20 - chkrootkit obtener resultados

Y si quisiéramos ver los resultados ejecutamos:

***\$ less resultados***

```
ROOTDIR is '/'
Checking `amd'... not found
Checking `basename'... not infected
Checking `biff'... not found
Checking `chfn'... not infected
Checking `chsh'... not infected
Checking `cron'... not infected
Checking `crontab'... not infected
Checking `date'... not infected
Checking `du'... not infected
Checking `dirname'... not infected
Checking `echo'... not infected
Checking `egrep'... not infected
Checking `env'... not infected
Checking `find'... not infected
Checking `fingerd'... not found
Checking `gpm'... not found
Checking `grep'... not infected
Checking `hdparm'... not infected
Checking `su'... not infected
Checking `ifconfig'... not infected
:
```

Ilustración 21 - chkrootkit visualizar resultados

También podemos configurarlo para automatizarlo para que realice los escaneos editando el archivo de configuración `/etc/chkrootkit.conf` con por ejemplo el editor nano. Para que sea automático debemos darle a la variable `'RUN_DAILY'` a `'true'`.

```
GNU nano 3.2 /etc/chkrootkit.conf
RUN_DAILY="true"
RUN_DAILY_OPTS="-q"
DIFF_MODE="false"

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell
```

Ilustración 22 - chkrootkit ejecución automatizada

Ahora veremos como se podría hacer con la herramienta `'rkhunter'`, que suele usarse complementariamente para descartar falsos positivos.

**`$ apt install rkhunter -y`**

```
root@montsegur:~# apt install rkhunter -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer
required:
  libmicrodns0 libminizip1 libmpv1 libqt5test5 libqt5webengine-data
  libqt5webengine5 libqt5webenginecore5 libre2-5
  linux-image-4.19.0-6-amd64 python-apt qml-module-qt-labs-platform
  qml-module-qtwebchannel qml-module-qtwebengine
Use 'apt autoremove' to remove them.
The following NEW packages will be installed:
  rkhunter
0 upgraded, 1 newly installed, 0 to remove and 37 not upgraded.
Need to get 256 kB of archives.
After this operation, 1,107 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian buster/main amd64 rkhunter all 1.4.
6-5 [256 kB]
Fetched 256 kB in 1s (324 kB/s)
Preconfiguring packages ...
```

Ilustración 23 - instalación rkhunter

Una vez lo tenemos instalando para realizar el escaneo ejecutamos:

***\$ rkhunter --check***

```
root@montsegur:~# rkhunter --check
[ Rootkit Hunter version 1.4.6 ]

Checking system commands...

Performing 'strings' command checks
  Checking 'strings' command [ OK ]

Performing 'shared libraries' checks
  Checking for preloading variables [ None found ]
  Checking for preloaded libraries [ None found ]
  Checking LD_LIBRARY_PATH variable [ Not found ]

Performing file properties checks
  Checking for prerequisites [ OK ]
  /usr/bin/awk [ OK ]
  /usr/bin/basename [ OK ]
  /usr/bin/bash [ OK ]
  /usr/bin/cat [ OK ]
  /usr/bin/chattr [ OK ]
```

Ilustración 24 - ejecución rkhunter

El primer paso de Rootkit Hunter es analizar los archivos binarios del sistema, librerías y strings.

```
/usr/sbin/nologin [ OK ]
/usr/sbin/pwck [ OK ]
/usr/sbin/rmmod [ OK ]
/usr/sbin/route [ OK ]
/usr/sbin/rsyslogd [ OK ]
/usr/sbin/runlevel [ Warning ]
/usr/sbin/sulogin [ OK ]
/usr/sbin/sysctl [ OK ]
/usr/sbin/useradd [ OK ]
/usr/sbin/userdel [ OK ]
/usr/sbin/usermod [ OK ]
/usr/sbin/vipw [ OK ]
/usr/sbin/unhide [ OK ]
/usr/sbin/unhide-linux [ OK ]
/usr/sbin/unhide-posix [ OK ]
/usr/sbin/unhide-tcp [ OK ]
/usr/lib/systemd/systemd [ Warning ]
[Press <ENTER> to continue]
```

Ilustración 25 - rkhunter escaneo librerías

En las anteriores capturas se revisaron los binarios y librerías, una vez presionamos 'Enter' pasará a intentar detectar rootkits.

```
[Press <ENTER> to continue]
Checking for rootkits...

Performing check of known rootkit files and directories
55808 Trojan - Variant A [ Not found ]
ADM Worm [ Not found ]
AjaKit Rootkit [ Not found ]
Adore Rootkit [ Not found ]
aPa Kit [ Not found ]
Apache Worm [ Not found ]
Ambient (ark) Rootkit [ Not found ]
Balaor Rootkit [ Not found ]
BeastKit Rootkit [ Not found ]
beX2 Rootkit [ Not found ]
BOBKit Rootkit [ Not found ]
cb Rootkit [ Not found ]
CiNIK Worm (Slapper.B variant) [ Not found ]
Danny-Boy's Abuse Kit [ Not found ]
Devil Rootkit [ Not found ]
```

Ilustración 26 - rkhunter escaneo rootkits

Si volvemos a presionar 'Enter' pasará a analizar más rootkits, troyanos y malwares.



```
Performing additional rootkit checks
  Suckit Rootkit additional checks           [ OK ]
  Checking for possible rootkit files and directories [ None found ]
  Checking for possible rootkit strings       [ None found ]

Performing malware checks
  Checking running processes for suspicious files [ None found ]
  Checking for login backdoors                  [ None found ]
  Checking for sniffer log files                [ None found ]
  Checking for suspicious directories          [ None found ]
  Checking for suspicious (large) shared memory segments [ Warning ]
  Checking for Apache backdoor                 [ Not found ]

Performing Linux specific checks
  Checking loaded kernel modules               [ OK ]
  Checking kernel module names                [ OK ]

[Press <ENTER> to continue]
```

Ilustración 27 - rkhunter escaneo rootkits, troyanos, malwares

Nuevamente si apretamos 'Enter' analizará las interfaces de red y los puertos conocidos posiblemente usados por troyanos y backdoors.

```
Checking the network...

Performing checks on the network ports
  Checking for backdoor ports                 [ None found ]

Performing checks on the network interfaces
  Checking for promiscuous interfaces         [ None found ]

Checking the local host...

Performing system boot checks
  Checking for local host name                [ Found ]
  Checking for system startup files           [ Found ]
  Checking system startup files for malware   [ None found ]

Performing group and account checks
  Checking for passwd file                   [ Found ]
  Checking for root equivalent (UID 0) accounts [ None found ]
  Checking for passwordless accounts         [ None found ]
```

Ilustración 28 - rkhunter escaneo interfaces de red

Finalmente pulsando por última vez 'Enter' nos revelará información del proceso y la ruta del archivo con los resultados ubicados en /var/log/rkhunter.log

```
File properties checks...
  Files checked: 149
  Suspect files: 16

Rootkit checks...
  Rootkits checked : 477
  Possible rootkits: 18

Applications checks...
  All checks skipped

The system checks took: 9 minutes and 48 seconds

All results have been written to the log file: /var/log/rkhunter.log

One or more warnings have been found while checking the system.
Please check the log file (/var/log/rkhunter.log)

root@montsegur:~#
```

Ilustración 29 - rkhunter resultado

### 2.1.5 Foremost

Foremost es un programa de datos que tiene el objetivo de recuperar archivos eliminados en Linux. Una de sus ventajas es que podemos usarlo para recuperar archivos de diferentes formatos. Esta herramienta ejecuta una búsqueda de tipo forense en el disco duro para realizar la recuperación.

Esta herramienta fue desarrollada por la Oficina de Investigadores Especiales de la Fuerza Aérea de los Estados Unidos junto con el Centro de Estudios e Investigación de Seguridad de Sistemas de Información, por lo que podemos ver la relevancia que tiene.

Cuando se borra un archivo del sistema y lo envías a la papelera permanecerá hasta que lo vacíes. Pero vaciarla no significa que los archivos se van para siempre, sino que siguen ya que el sistema solo elimina los metadatos y deja los datos inferiores con el fin de sobrescribirlos. Hay muchas posibilidades de recuperar un archivo, pero no siempre con un 100% de calidad e integridad.

Foremast copia y analiza el disco duro para detectar archivos ocultos y luego se aloja esa información de forma temporal usando la memoria del equipo y buscará sus coincidencias para encontrar un archivo integral.

Foremost puede recuperar archivos con los siguientes formatos: jpg, gif, png, bmp, avi, tiff, mp4, exe, mpg, wav, asf, wma, mp3, fws, riff, wmv, mov, pdf, ole, doc, docx, xls, xlsx, ppt, pptx, zip, rar, html, cpp, java, art, pst, ost, dbx, idx, mbx, wpc, pgp, txt, rpm, dat, etc.

La sintaxis es la siguiente:

***\$ foremost (-v / -V - -h / -T / -Q / -q / -a / -w / -d) (-t (tipo)) (-s (bloques)) (-k (tamaño)) (-b (tamaño)) (-c (archivo)) (-o (dir)) (-i (archivo))***

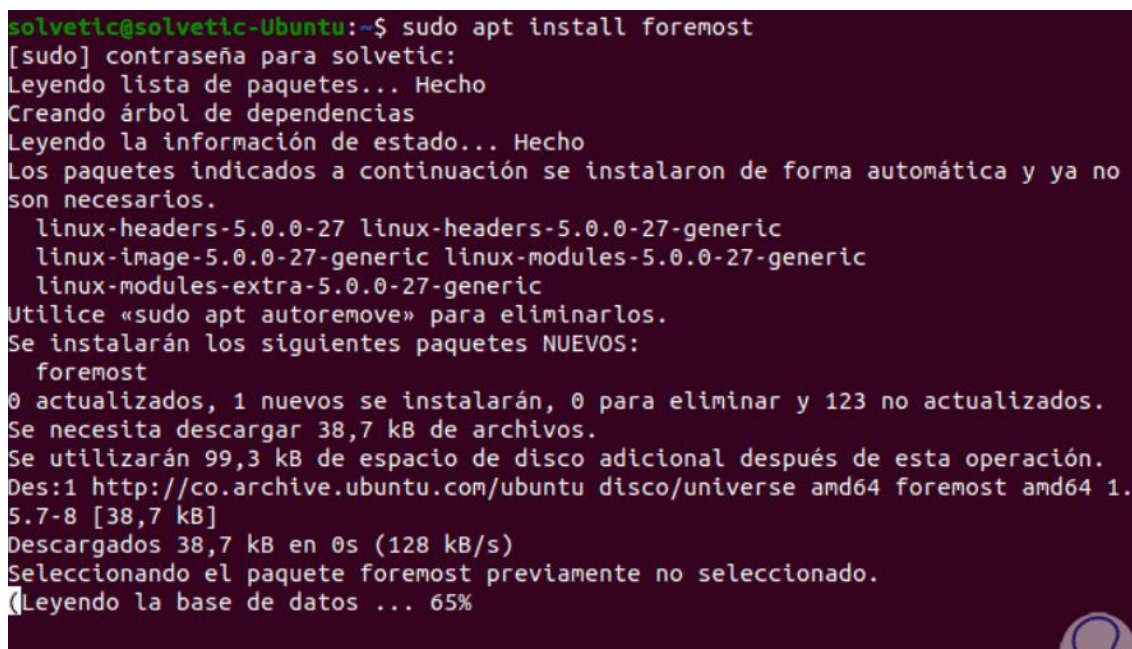


Los parámetros son los siguientes:

- V: despliega los derechos de Copyright y la información del objeto.
- t: especifica el tipo de archivo.
- d: activa la detección indirecta de bloques.
- i: permite especificar el archivo de salida.
- a: escribe todos los encabezados y no detecta errores.
- w: solo escribe en el archivo auditado pero no escribe en los demás archivos del sistema.
- o: define la salida del archivo.
- c: establece la configuración del archivo.
- q: habilita el modo rápido.
- Q: habilita el modo silencio.
- v: activa el modo verbose para obtener mejores detalles.

Para instalar Foremost ejecutamos el siguiente comando:

***\$ sudo apt install foremost***



```
solvetic@solvetic-Ubuntu:~$ sudo apt install foremost
[sudo] contraseña para solvetic:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  linux-headers-5.0.0-27 linux-headers-5.0.0-27-generic
  linux-image-5.0.0-27-generic linux-modules-5.0.0-27-generic
  linux-modules-extra-5.0.0-27-generic
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes NUEVOS:
  foremost
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 123 no actualizados.
Se necesita descargar 38,7 kB de archivos.
Se utilizarán 99,3 kB de espacio de disco adicional después de esta operación.
Des:1 http://co.archive.ubuntu.com/ubuntu disco/universe amd64 foremost amd64 1.5.7-8 [38,7 kB]
Descargados 38,7 kB en 0s (128 kB/s)
Seleccionando el paquete foremost previamente no seleccionado.
(Leyendo la base de datos ... 65%
```

Ilustración 30 - Foremost instalación

Ahora veremos cómo usar la herramienta, lo primero será conocer la ID de la unidad por lo que ejecutamos:

**\$ df -h**

```
solvetic@solvetic-Ubuntu:~$ df -h
Filesystem      Tamaño Usados  Disp Uso% Montado en
udev            2,0G    0      2,0G   0% /dev
tmpfs           395M    1,4M   393M   1% /run
/dev/sda1       49G     20G    27G  43% /
tmpfs           2,0G    0      2,0G   0% /dev/shm
tmpfs           5,0M    4,0K    5,0M   1% /run/lock
tmpfs           2,0G    0      2,0G   0% /sys/fs/cgroup
/dev/loop1      3,8M    3,8M    0 100% /snap/gnome-system-monitor/100
/dev/loop2      1,0M    1,0M    0 100% /snap/gnome-logs/73
/dev/loop0      55M     55M    0 100% /snap/core18/1144
/dev/loop3      54M     54M    0 100% /snap/core18/941
/dev/loop5      3,8M    3,8M    0 100% /snap/gnome-system-monitor/77
/dev/loop4      43M     43M    0 100% /snap/gtk-common-themes/1313
/dev/loop6      4,2M    4,2M    0 100% /snap/gnome-calculator/406
/dev/loop7      4,3M    4,3M    0 100% /snap/gnome-calculator/501
/dev/loop8      15M     15M    0 100% /snap/gnome-characters/254
/dev/loop10     15M     15M    0 100% /snap/gnome-characters/317
/dev/loop11     36M     36M    0 100% /snap/gtk-common-themes/1198
/dev/loop9      90M     90M    0 100% /snap/core/6673
/dev/loop14     1,0M    1,0M    0 100% /snap/gnome-logs/61
/dev/loop12     90M     90M    0 100% /snap/core/7713
/dev/loop13     152M    152M    0 100% /snap/gnome-3-28-1804/31
/dev/loop15     150M    150M    0 100% /snap/gnome-3-28-1804/71
tmpfs           395M    32K    395M   1% /run/user/1000
```

Ilustración 31 - Foremost obtener IDs

Por ejemplo, podemos seleccionar /dev/sda1 para realizar la búsqueda allí, ahora intentaremos rescatar archivos .docx, para ello ejecutamos lo siguiente:

**\$ foremost -v -t docx -i /dev/sda1 -o ~/recovery/**

Al ejecutar esto comenzará el análisis en esa unidad.

```
solvetic@solvetic-Ubuntu:~$ foremost -v -t docx -i /dev/sda1 -o ~/recovery/
Foremost version 1.5.7 by Jesse Kornblum, Kris Kendall, and Nick Mikus
Audit File

Foremost started at Mon Oct  7 15:33:24 2019
Invocation: foremost -v -t docx -i /dev/sda1 -o /home/solvetic/recovery/
Output directory: /home/solvetic/recovery
Configuration file: /etc/foremost.conf
Processing: stdin
|-----
File: stdin
Start: Mon Oct  7 15:33:24 2019
Length: Unknown

Num      Name (bs=512)      Size      File Offset      Comment
```

Ilustración 32 - Foremost analizar unidad

Cuando la búsqueda finaliza los archivos que se han recuperado estarán en la carpeta precedida del parámetro -o, allí podremos reemplazar el tipo de archivo por el que queremos.

```

solvetic@solvetic-Ubuntu:~$ foremost -v -t png -i /dev/sdc1/ -o ~/solvetic/
Foremost version 1.5.7 by Jesse Kornblum, Kris Kendall, and Nick Mikus
Audit File

Foremost started at Mon Oct 7 15:38:17 2019
Invocation: foremost -v -t png -i /dev/sdc1/ -o /home/solvetic/solvetic/
Output directory: /home/solvetic/solvetic
Configuration file: /etc/foremost.conf
Processing: stdin
|-----|
File: stdin
Start: Mon Oct 7 15:38:17 2019
Length: Unknown

Num      Name (bs=512)      Size      File Offset      Comment

```

Ilustración 33 - Foremost reemplazo de archivos

El proceso tardará según el tamaño de la unidad y el número de archivos. Automáticamente creará una carpeta en el directorio 'Home' con el nombre que se ha especificado.

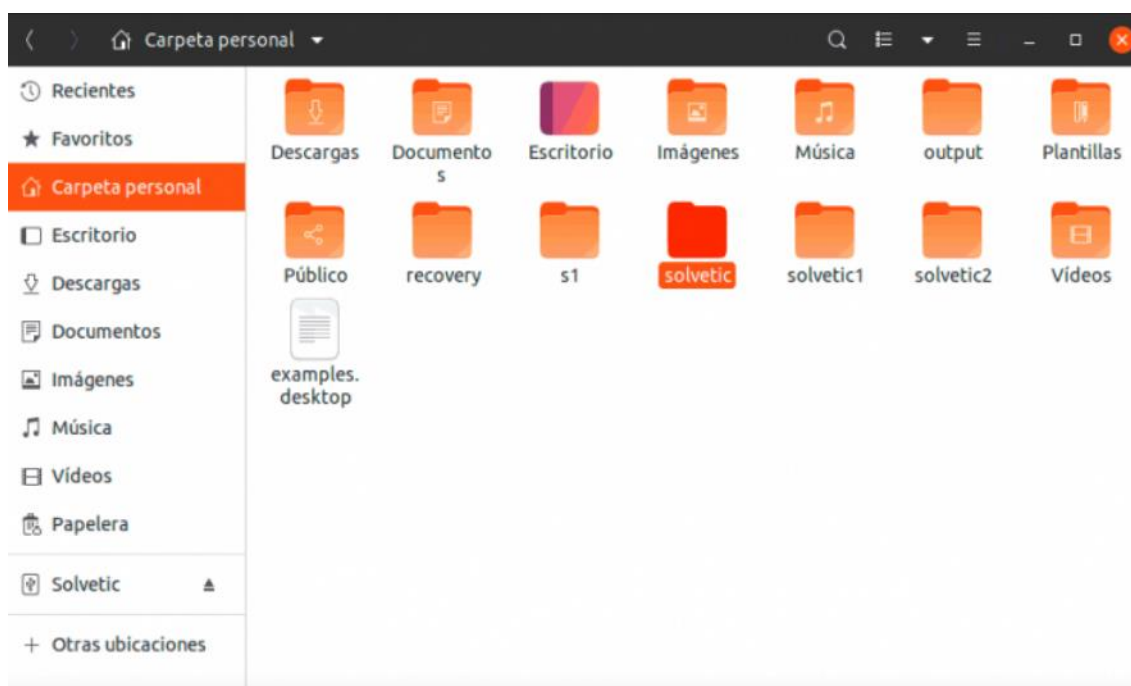


Ilustración 34 - Foremost resultado en carpeta

### 2.1.6 Scalpel

Es una herramienta que permite recuperar archivos. Utiliza una técnica que se basa en los encabezados, pies de páginas y estructuras internas de los mismos, que le facilita acceder al a de la base de bloques dónde están los archivos borrados, identificarlos y recuperarlos casi al instante, siendo así una herramienta bastante útil para la investigación forense digital.

Una vez más, su uso es muy sencillo, y únicamente utiliza la siguiente orden:





## 3 NOTICIAS

### 3.1 Binwalk tool

Respecto a esta herramienta hemos encontrado numerosas noticias donde es muy factible que haya sido usada para explotar y localizar las posibles vulnerabilidades.

Un ejemplo puede ser alguna de las numerosas vulnerabilidades que han sido publicadas por Cisco.

#### Bugs in Cisco RV132W and RV134W routers

[CVE-2021-1287 Detail](#)

by NIST March 17, 2021

"A vulnerability in the web-based management interface of Cisco RV132W ADSL2+ Wireless-N VPN Routers and Cisco RV134W VDSL2 Wireless-AC VPN Routers could allow an authenticated, remote attacker to execute arbitrary code on an affected device or cause the device to restart unexpectedly. The vulnerability exists because the web-based management interface does not properly validate user-supplied input.... A successful exploit could allow the attacker to execute arbitrary code as the root user .... " The attacker needs to be authenticated to the device before they can exploit the flaw. Fixes are available.

- [Cisco Small Business RV132W and RV134W Routers Management Interface Remote Command Execution and Denial of Service Vulnerability](#) by Cisco March 17, 2021.
- [Cisco Plugs Security Hole in Small Business Routers](#) by Lindsey O'Donnell for ThreatPost March 17, 2021

Ilustración 37 – Vulnerabilidad de Router Cisco

En concreto se permitía hacer usos de comandos "root" y, por tanto, dejar totalmente expuesto todo el sistema.

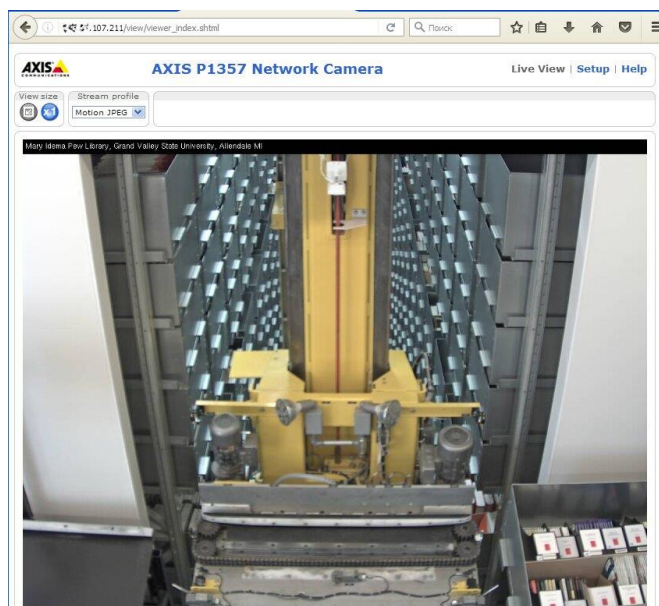


Ilustración 38 – Cámara IP

Algo que nos puede resultar más familiar es el hackeo de cámaras IP. En la mayoría de los casos se hace un estudio de su firmware creando un directorio de tuplas usuario/contraseña que permiten realizar un fácil ataque si se tiene un mínimo de información del sujeto o empresa al que se va a atacar.

Por ejemplo, en la imagen de arriba se puede ver como se ha sido capaz de acceder al sistema de vigilancia y control de una empresa privada.

### 3.2 Bulk Extractor

En la mayoría de los delitos que se comenten en la red cometen el error de no enmascarar su IPs. Además, suelen ir acompañados de actuaciones en grupo que comparten este problema.



Ilustración 39 – Recopilación de noticias

Podemos señalar el caso de la pornografía infantil que se caracteriza porque suele estar conformada por un grupo de delincuentes, normalmente sin conocimientos informáticos profundos, que localizan, crean y comparten este contenido ilegal entre ellos.

Al momento de localizar uno de estos integrantes, suele ser normal que sea bastante sencillo tirar de la rama y ser capaces de detener a cada uno de ellos.

### 3.3 Scalpel Tool

¿Cómo puede ser planteada esta herramienta en el análisis forense?

## Recuperados los archivos informáticos borrados sobre los sumarios relacionados con Jesús Gil

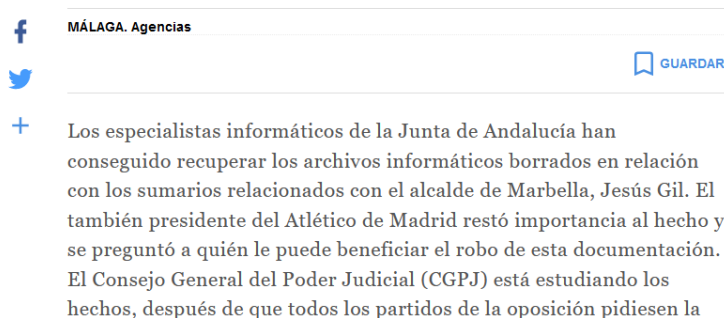


Ilustración 40 – Recuperación de archivos

Pues lo sorprendente es que ya desde 2001 se conocen técnicas similares a esta. En la imagen podemos ver como se fue capaz de recuperar toda la información que había sido borrada referente al caso.

Evidentemente las técnicas han mejorado muchísimo. Aun así, sigue siendo uno de los principales pilares a la hora de realizar análisis forenses informáticos.

### 3.4 Autopsy tool

En este caso hacemos referencia al software clave de utilización en el análisis forense español.

David del Olmo tuvo un caso de un empleado directivo de una gran empresa tecnológica que iba a ser despedido. “Se negaba entregar los medios tecnológicos que le proporcionó la empresa (portátil y disco externo)”, por lo que la empresa empieza a sospechar que está realizando proyectos para otra de la competencia”.

Para intentar corroborar esta tesis, se cita al empleado en una notaría mediante un burofax, para que el notario pudiera dar fe de lo que allí iba a ocurrir. “El representante legal de la empresa preguntó a este empleado si hacía entrega de todos los datos y si había borrado algo”. ¿La respuesta delante del notario? Que no había borrado nada.

Sin embargo, y una vez realizada la clonación del disco y la posterior recuperación de datos, “obtuve un “timeline” del usuario donde conseguí demostrar que el usuario durante la madrugada previa había borrado archivos de la empresa y correos electrónicos”. De hecho, se había levantado a las 5 de la mañana para proceder a todo esta eliminación de archivos antes de presentarse en la notaría donde, probablemente, no se esperaba la presencia de un perito.



David del Olmo

Ilustración 41 – Borrado de información

En la noticia se hace referencia a una situación vivida por el analista forense David del Olmo. En resumen, David analizó el equipo de un alto directivo que estaba realizando un filtrado de información y trabajos para empresas rivales.

Tras las sospechas, al analizar el equipo la mayor parte de la información había sido borrada, sin embargo, David fue capaz de obtener un informe de las últimas acciones realizadas, señalando el continuado borrado de información que se había realizado días antes.

### 3.5 Chkrootkit

#### Lo que hizo Sony

El 31 de octubre del 2005, el experto en seguridad informática Mark Russinovich publicó su descubrimiento en [su blog](#) sobre un software espía, conocido como rootkit, que se había instalado secretamente en su ordenador. Dedujo que el rootkit estaba conectado con el reproductor de música que venía incluido en CDs de música de Sony. El programa oculto rootkit se usaba para espiar a los usuarios y sus hábitos de escucha, compartiendo esta información con Sony, así como evitaba la lectura [del disco](#) por parte de terceros.

En el proceso de espionaje, el rootkit [creaba fallos adicionales de seguridad](#) que abrían las puertas para otros ataques peores. Incluso si los usuarios detectaban el rootkit, desinstalarlo de forma segura sin dañar la máquina era otro problema.

El rootkit se cargó en un total de [aproximadamente 25 millones de CDs](#) e [infectó más de 550.000 redes en más de cien países, incluyendo miles de redes militares y de defensa de los EE.UU.](#)

Pero el presidente de Sony BMG, Thomas Hesse, desestimó totalmente el problema, y declaró [textualmente](#) “La mayoría de la gente, creo, ni siquiera sabe lo que es un Rootkit, así que ¿por qué han de preocuparse?”. La prensa publicó lo que Sony estaba haciendo de forma secreta a la propiedad privada de los usuarios y Sony se vio forzada a pagar [numerosos procesos judiciales](#) y recuperar la confianza de los usuarios tan pronto como fuese posible.

Ilustración 42 – Rootkit de Sony



En la noticia superior se muestra como Sony introduzco código malicioso sin alertar al usuario capaz de crear fallos de seguridad, abrir puertas traseras y analizar posibles acciones ilegales sobre sus productos.



## 4 BIBLIOGRAFÍA

- <https://www.welivesecurity.com/la-es/2015/04/15/5-fases-analisis-forense-digital/>
- <https://rekodbyte.wordpress.com/2017/02/18/binwalk/>
- <https://linux.lat/como-detectar-rootkits-con-chkrootkit-y-rkhunter/>
- <https://www.solvetic.com/tutoriales/article/7900-como-usar-foremost-linux-y-recuperar-archivos-borrados/>
- <https://ciberseguridad.blog/las-mejores-herramientas-hacking/>
- <https://www.welivesecurity.com/la-es/2013/08/12/en-que-consiste-analisis-forense-de-informacion/>
- <https://protecciondatos-lopd.com/empresas/informatica-forense/>
- <https://duartecarito.wixsite.com/eportafolioforense/single-post/2015/05/18/fases-en-la-informatica-forense#:~:text=FASE%20DE%20IDENTIFICACION&text=En%20esta%20etapa%20de%20la,duraci%C3%B3n%20y%20detalles%20del%20mismo.>
- <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/39681/6/cgervillarTFM1214memoria.pdf>