

Simulación de análisis forense con herramientas de Kali Linux​

Administración de sistemas y seguridad

Propiedad Intelectual

Universidad de Granada

Breve descripción

Autores

Álvaro de la Flor Bonilla y Antonio Manuel Salvat Pérez

**Escuela Técnica Superior de Ingenierías Informática y de Telecomunicaciones**

**Máster Oficial en Ingeniería Informática**

**Curso 2020/2021**

ÍNDICE DEL PROYECTO

[1 Introducción 4](#_Toc69428543)

[1.1 Análisis forense informático 4](#_Toc69428544)

[1.1.1 Definición 4](#_Toc69428545)

[1.1.2 Objetivos 4](#_Toc69428546)

[1.2 Herramientas 5](#_Toc69428547)

[1.2.1 Noticias 5](#_Toc69428548)

[2 Desarrollo 6](#_Toc69428549)

[2.1 Herramientas 6](#_Toc69428550)

[2.1.1 Autopsy 6](#_Toc69428551)

[2.1.2 Binwalk 10](#_Toc69428552)

[2.1.3 Bulk Extractor 14](#_Toc69428553)

[2.1.4 Chkrootkit 16](#_Toc69428554)

[2.1.5 Foremost 23](#_Toc69428555)

[2.1.6 Scalpel 26](#_Toc69428556)

[3 Noticias 28](#_Toc69428557)

[3.1 Binwalk tool 28](#_Toc69428558)

[3.2 Bulk Extractor 28](#_Toc69428559)

[3.3 Scalpel Tool 29](#_Toc69428560)

[3.4 Autopsy tool 30](#_Toc69428561)

[3.5 Chkrootkit 30](#_Toc69428562)

[4 BIBLIOGRAFÍA 32](#_Toc69428563)

íNDICE DE Ilustraciones

[Ilustración 1 - Autopsy pantalla principal 6](#_Toc69428564)

[Ilustración 2 - Ejecución de Autopsy 7](#_Toc69428565)

[Ilustración 3 – Información extraída 7](#_Toc69428566)

[Ilustración 4 - Particiones Autopsy 8](#_Toc69428567)

[Ilustración 5 - Autopsy extensiones 9](#_Toc69428568)

[Ilustración 6 - Autopsy búsqueda por archivo 9](#_Toc69428569)

[Ilustración 7 - Autopsy búsqueda por palabras clave 10](#_Toc69428570)

[Ilustración 8 - Análisis básico binwalk 10](https://uses0-my.sharepoint.com/personal/antsalper4_alum_us_es/Documents/ASS%20(1).docx#_Toc69428571)

[Ilustración 9 - Análisis básico binwalk 11](https://uses0-my.sharepoint.com/personal/antsalper4_alum_us_es/Documents/ASS%20(1).docx#_Toc69428572)

[Ilustración 10 - Análisis recursivo binwalk 11](https://uses0-my.sharepoint.com/personal/antsalper4_alum_us_es/Documents/ASS%20(1).docx#_Toc69428573)

[Ilustración 11 - Binwalk archivos por contenido 12](#_Toc69428574)

[Ilustración 12 - Binwalk extraer tipo determinado 12](#_Toc69428575)

[Ilustración 13 - Binwalk busqueda de archivo por línea 13](#_Toc69428576)

[Ilustración 14 – Ejecución de bulk\_extractor 15](#_Toc69428577)

[Ilustración 15 – Ejecución completa de bulk\_extractor 15](#_Toc69428578)

[Ilustración 16 – Archivos extraidos 16](#_Toc69428579)

[Ilustración 17 - chkrootkit instalación 17](#_Toc69428580)

[Ilustración 18 - chkrootkit ejecución 17](#_Toc69428581)

[Ilustración 19 - chkrootkit ejecución cómoda 18](#_Toc69428582)

[Ilustración 20 - chkrootkit obtener resultados 18](#_Toc69428583)

[Ilustración 21 - chkrootkit visualizar resultados 19](#_Toc69428584)

[Ilustración 22 - chkrootkit ejecución automatizada 19](#_Toc69428585)

[Ilustración 23 - instalación rkhunter 20](#_Toc69428586)

[Ilustración 24 - ejecución rkhunter 20](#_Toc69428587)

[Ilustración 25 - rkhunter escaneo librerías 21](#_Toc69428588)

[Ilustración 26 - rkhunter escaneo rootkits 21](#_Toc69428589)

[Ilustración 27 - rkhunter escaneo rootkits, troyanos, malwares 22](#_Toc69428590)

[Ilustración 28 - rkhunter escaneo interfaces de red 22](#_Toc69428591)

[Ilustración 29 - rkhunter resultado 23](#_Toc69428592)

[Ilustración 30 - Foremost instalación 24](#_Toc69428593)

[Ilustración 31 - Foremost obtener IDs 25](#_Toc69428594)

[Ilustración 32 - Foremost analizar unidad 25](#_Toc69428595)

[Ilustración 33 - Foremost reemplazo de archivos 26](#_Toc69428596)

[Ilustración 34 - Foremost resultado en carpeta 26](#_Toc69428597)

[Ilustración 35 – Ejecución de Scalpel 27](#_Toc69428598)

[Ilustración 36 – Archivo resumen 27](#_Toc69428599)

[Ilustración 37 – Vulnerabilidad de Router Cisco 28](#_Toc69428600)

[Ilustración 38 – Cámara IP 28](#_Toc69428601)

[Ilustración 39 – Recopilación de noticias 29](#_Toc69428602)

[Ilustración 40 – Recuperación de archivos 29](#_Toc69428603)

[Ilustración 41 – Borrado de información 30](#_Toc69428604)

[Ilustración 42 – Rootkit de Sony 30](#_Toc69428605)

# Introducción

## Análisis forense informático

### 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.

### 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.

## 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.

### 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.

# Desarrollo

## Herramientas

### 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:

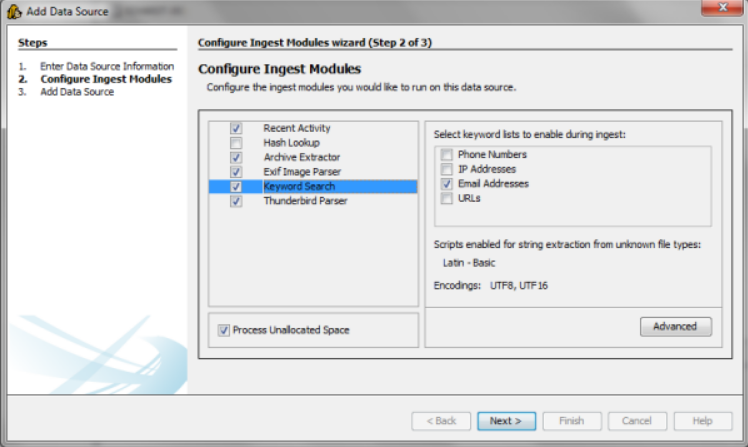


Ilustración - Autopsy pantalla principal

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:

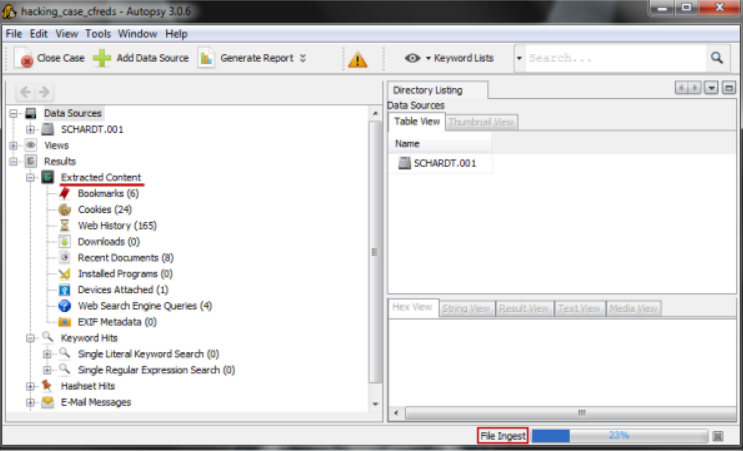


Ilustración - Ejecución de Autopsy

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:

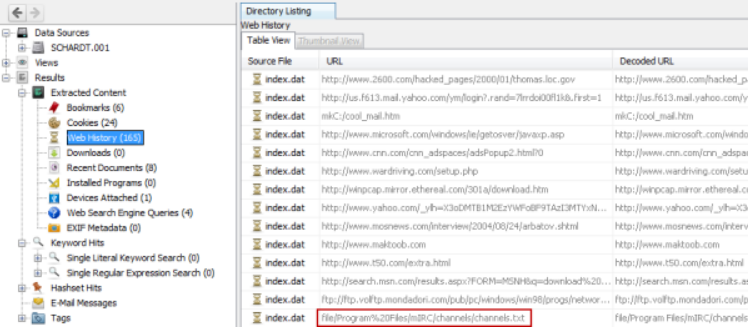


Ilustración – Información extraída

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:

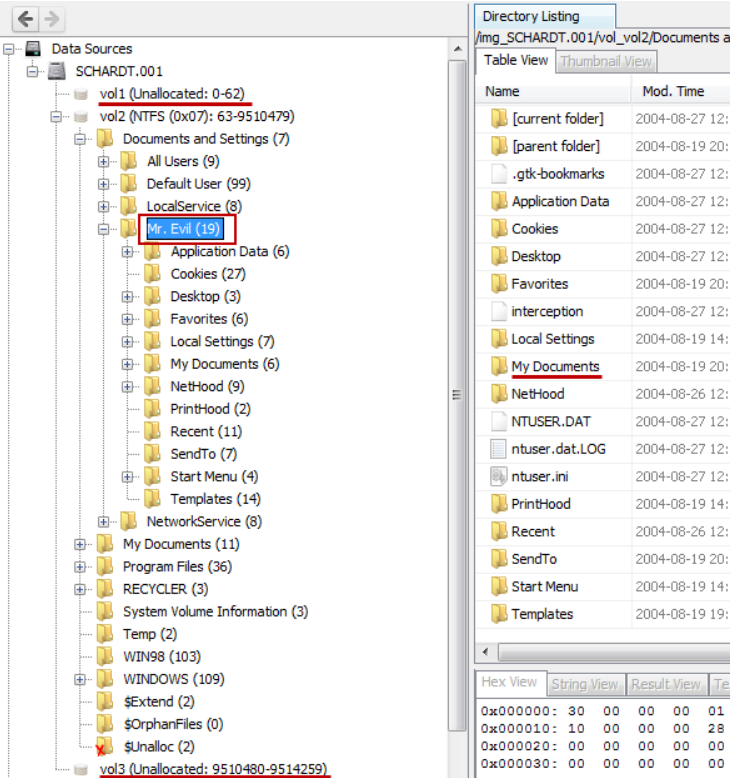


Ilustración - Particiones Autopsy

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:

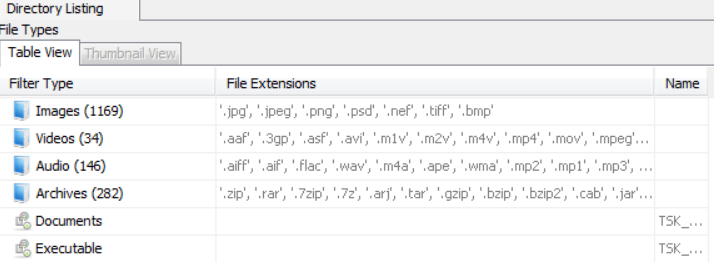


Ilustración - Autopsy extensiones

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:

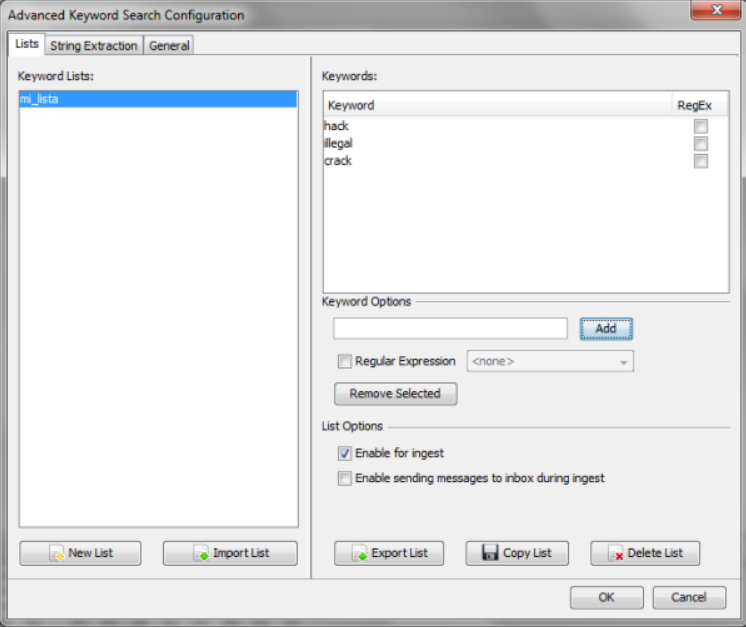


Ilustración - Autopsy búsqueda por archivo

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:

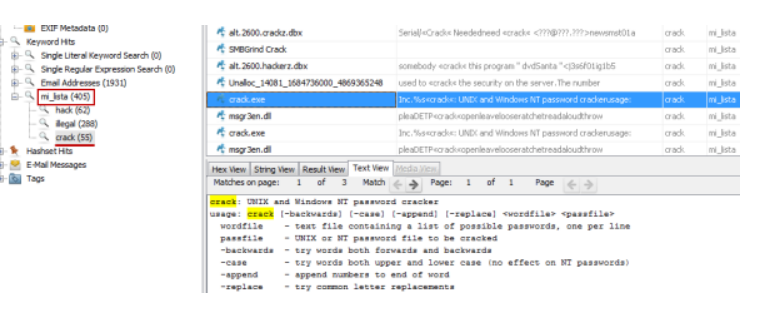


Ilustración - Autopsy búsqueda por palabras clave

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 illegal, 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.

### Binwalk

Binkwalk 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

Ilustración - Análisis básico binwalk

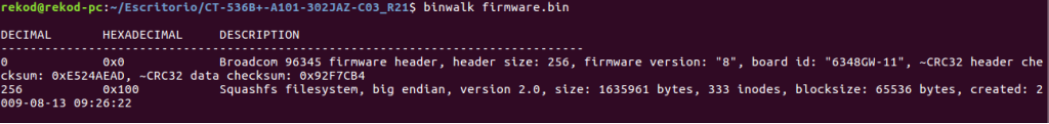


Ilustración - Análisis básico binwalk

Este sería el análisis más básico.

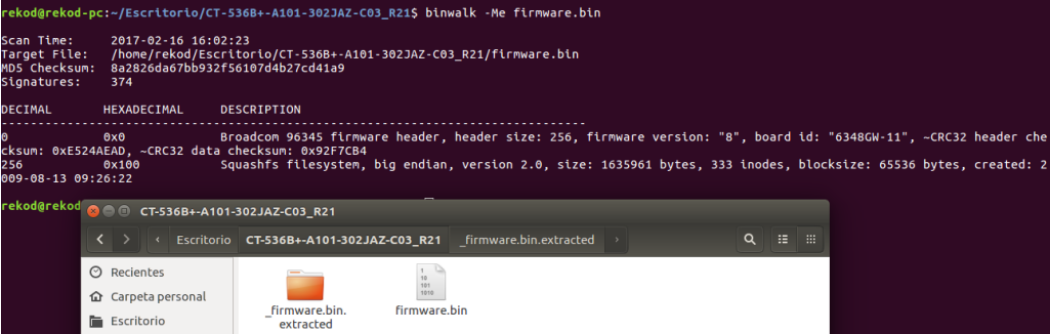
* binwalk -Me firmware.bin

Ilustración - 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

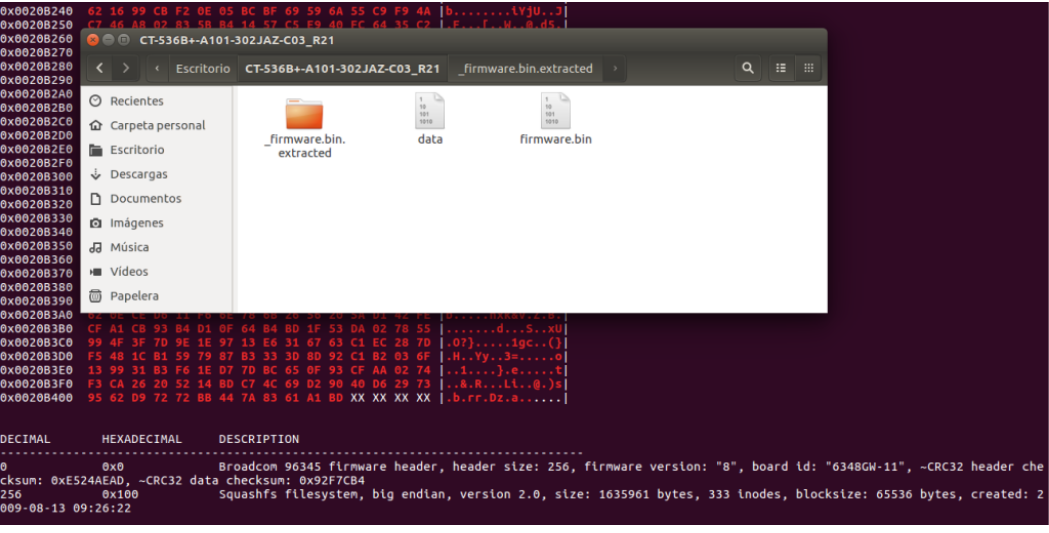


Ilustración - Binwalk archivos por contenido

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 ‘lenght’, 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

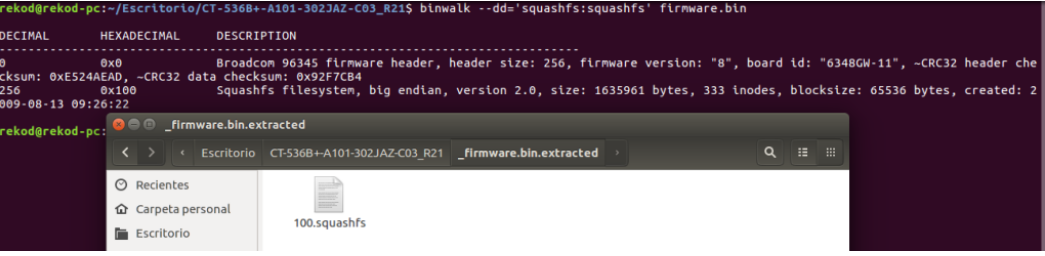


Ilustración - Binwalk extraer tipo determinado

Bara buscar un archivo en una línea determinada:

* binwalk -R “\x100” firmware.bin

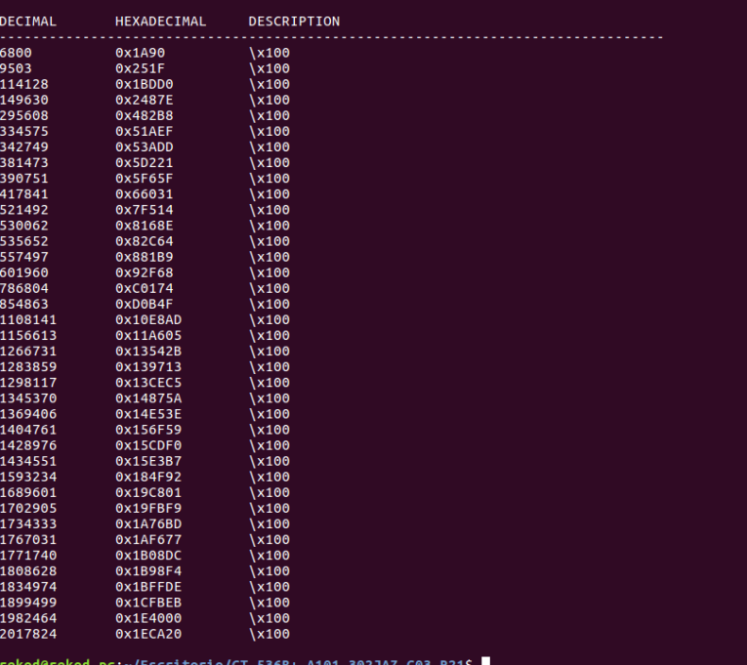


Ilustración - Binwalk busqueda de archivo por línea

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 –finclude=’\.bin$’ firmware.bin
* binwalk -M -e –finclude=’\.bin$’ 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.

### 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.

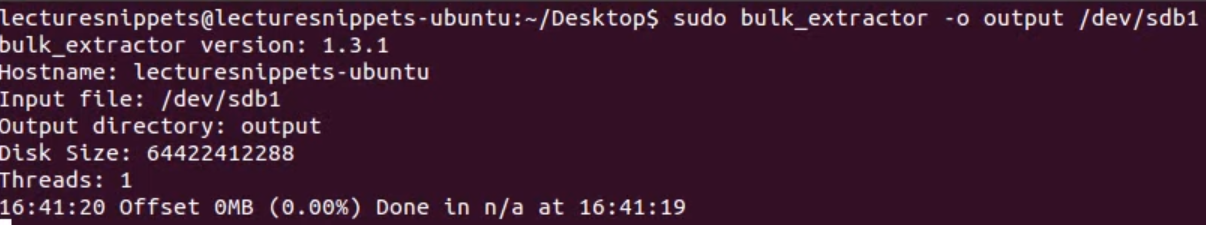


Ilustración – Ejecución de bulk\_extractor

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.

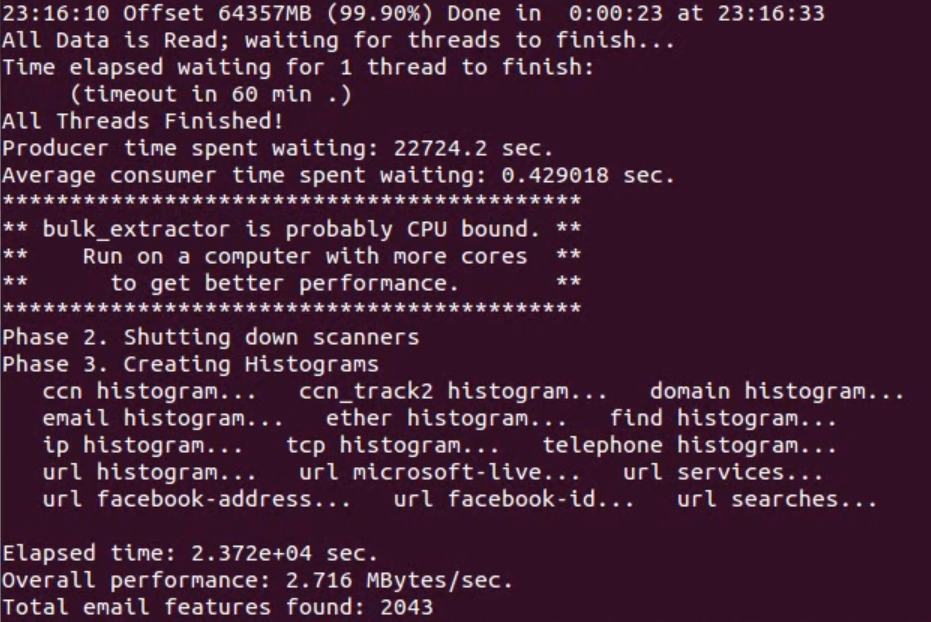


Ilustración – Ejecución completa de bulk\_extractor

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.

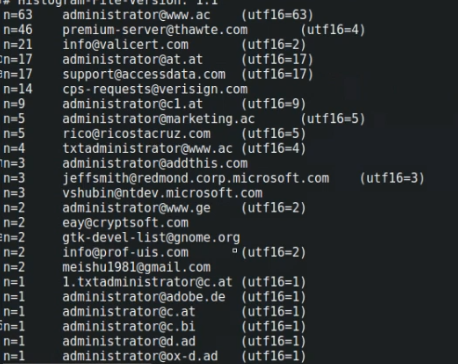
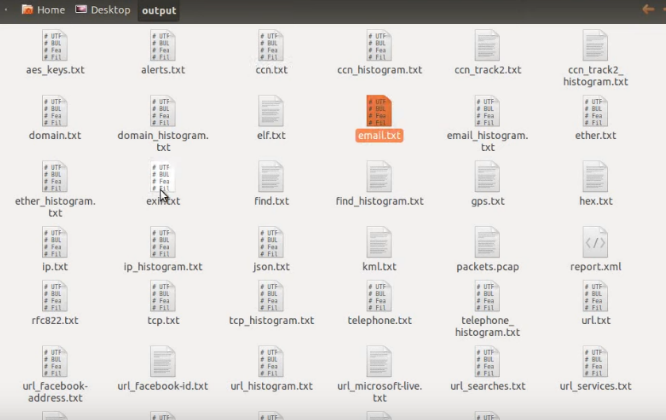


Ilustración – Archivos extraidos

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.

### 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**

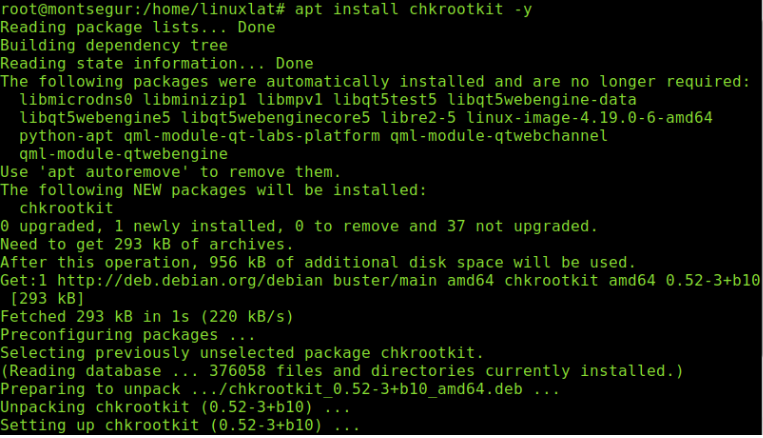


Ilustración - chkrootkit instalación

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

**$ sudo chkrootkit**

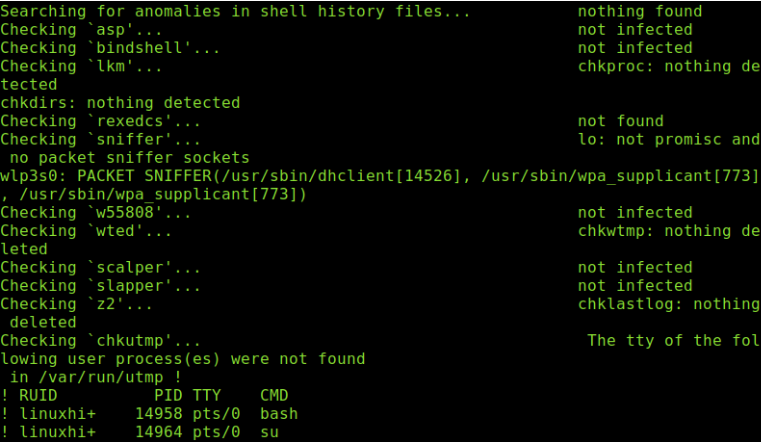


Ilustración - chkrootkit ejecución

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**

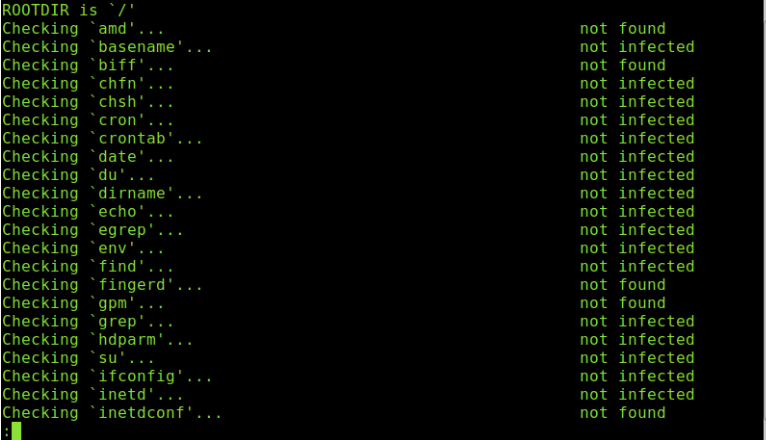


Ilustración - chkrootkit ejecución cómoda

También podemos ver los resultados ejecutando:

**$ sudo chkrootkit > resultados**

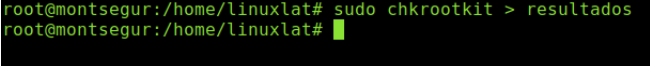


Ilustración - chkrootkit obtener resultados

Y si quisiéramos ver los resultados ejecutamos:

**$ less resultados**

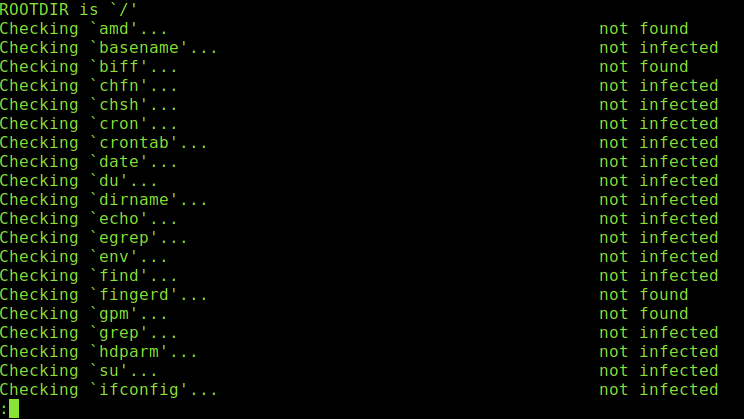


Ilustración - chkrootkit visualizar resultados

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’.

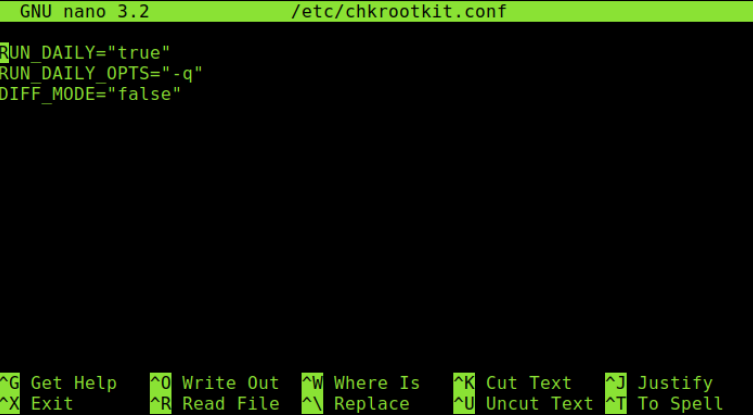


Ilustración - chkrootkit ejecución automatizada

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

**$ apt install rkhunter -y**

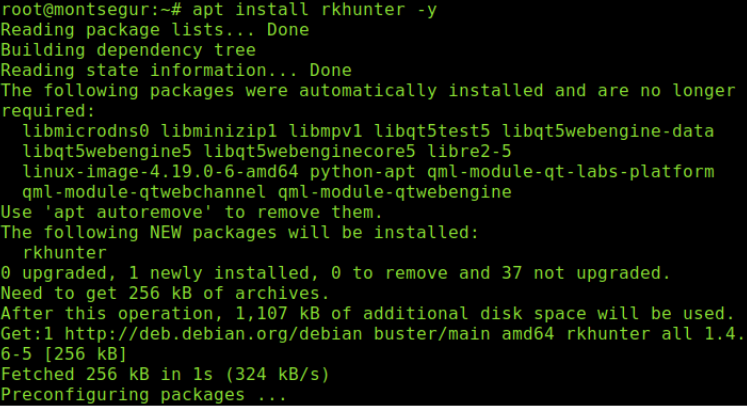


Ilustración - instalación rkhunter

Una vez lo tenemos instalando para realizar el escaneo ejecutamos:

**$ rkhunter –check**

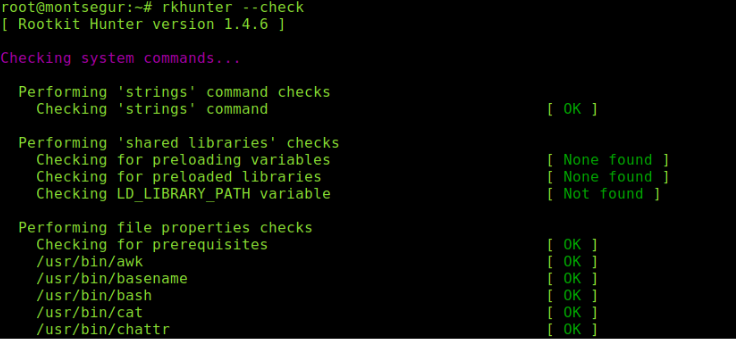


Ilustración - ejecución rkhunter

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

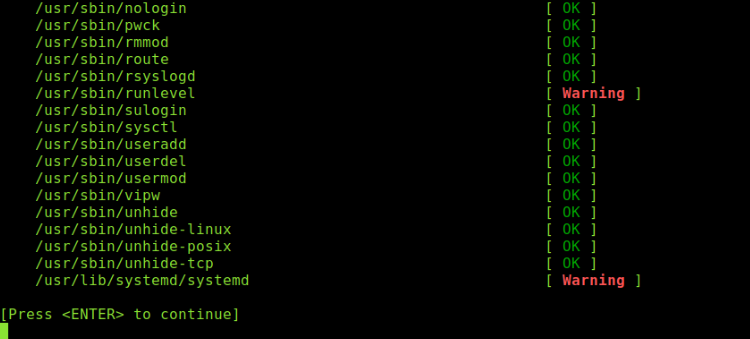


Ilustración - rkhunter escaneo librerías

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

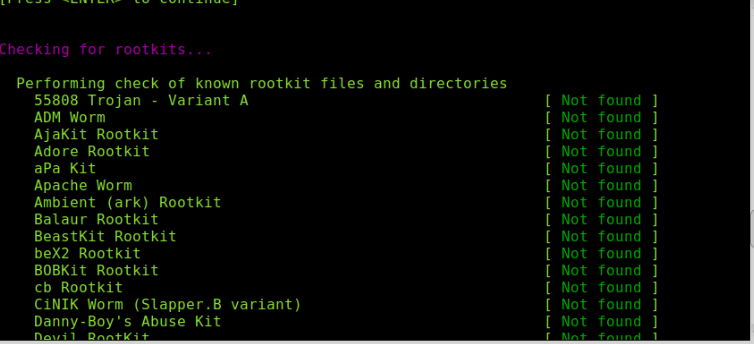


Ilustración - rkhunter escaneo rootkits

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

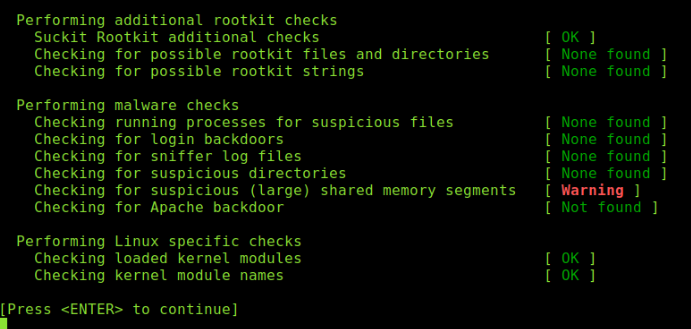


Ilustración - rkhunter escaneo rootkits, troyanos, malwares

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

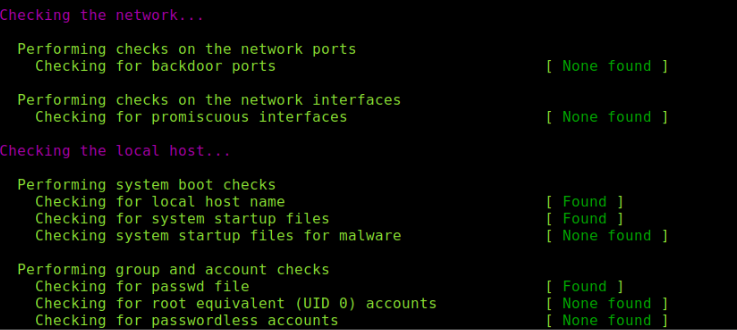


Ilustración - rkhunter escaneo interfaces de red

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

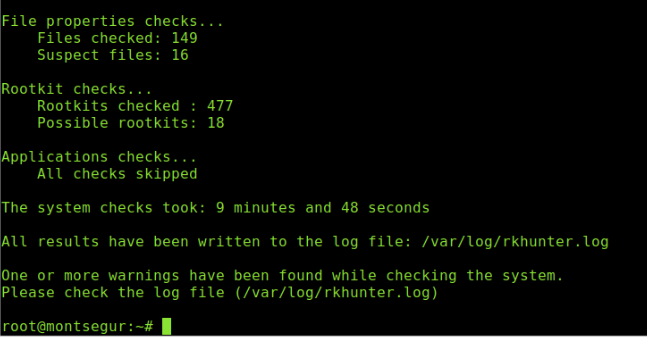


Ilustración - rkhunter resultado

### 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 Foremast ejecutamos el siguiente comando:

**$ sudo apt install foremost**

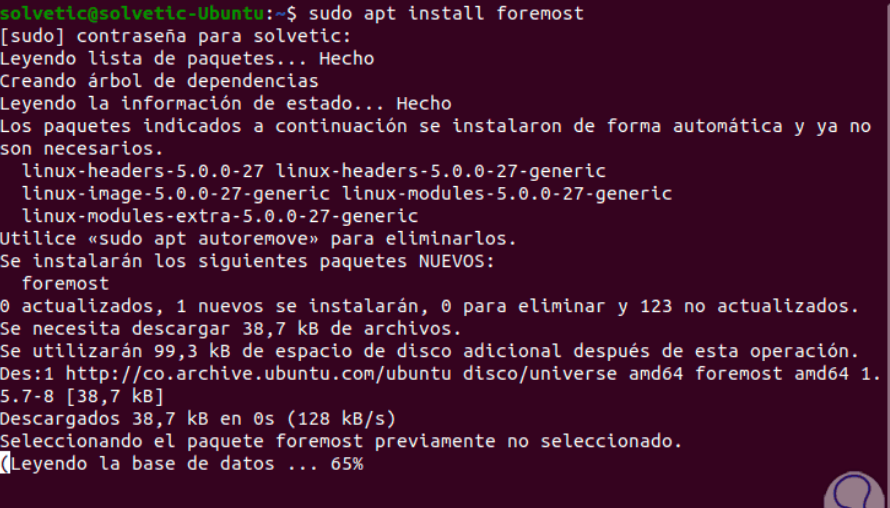


Ilustración - Foremost instalación

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

**$ df -h**

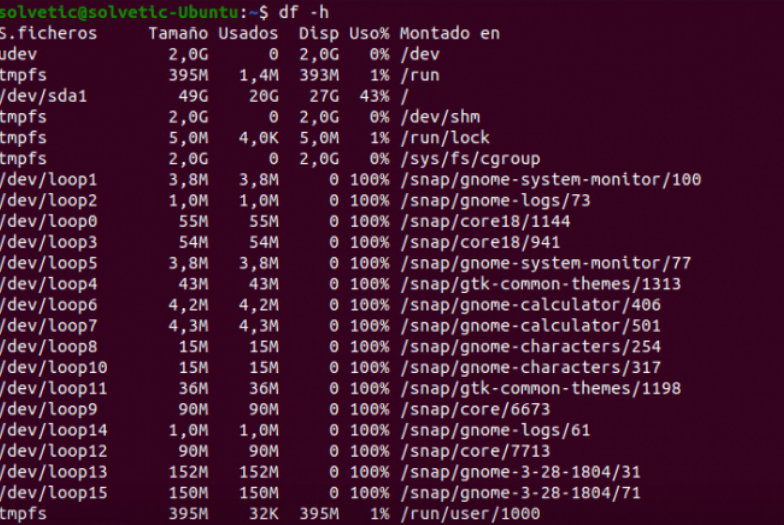


Ilustración - Foremost obtener IDs

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.

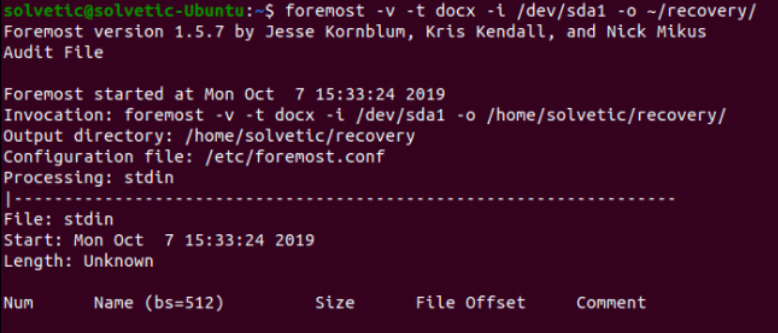


Ilustración - Foremost analizar unidad

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.

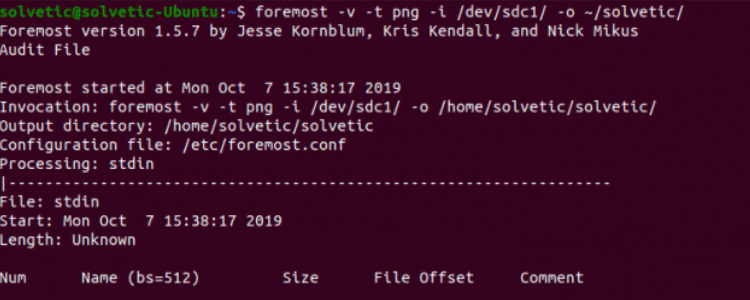


Ilustración - Foremost reemplazo de archivos

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.

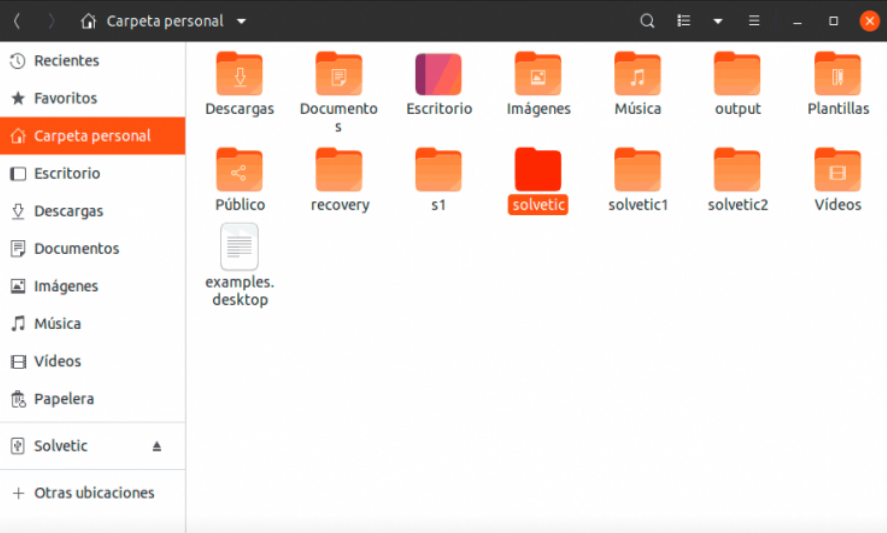


Ilustración - Foremost resultado en carpeta

### 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:

$ scalpel -c /etc/scalpel/scalpel.conf <device\_directory> -o <output>

En cuanto *“device\_directory”* indica el directorio de archivos donde se quiere realizar el análisis. En el caso de *“output”* se indica el directorio en el que se quiere que se guarden los archivos recuperados.

Texto

Descripción generada automáticamente

Ilustración – Ejecución de Scalpel

Una vez completado el proceso, junto con los archivos extraídos se añade otro resumen indicado que se ha recuperado.

Tabla

Descripción generada automáticamente

Ilustración – Archivo resumen

En la imagen anterior puede ver como se indica que se han recuperado cuatro archivos, en este caso cuatro imágenes del sistema de archivos *“./sdb”*.

# Noticias

## 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.

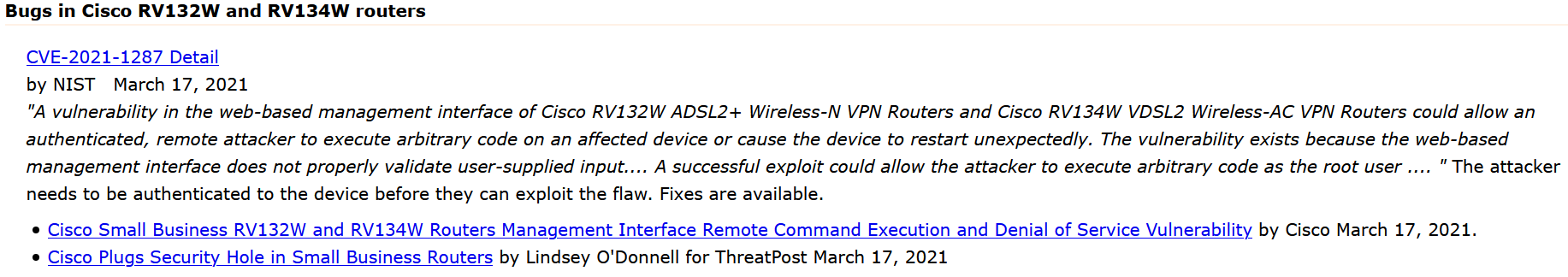


Ilustración – Vulnerabilidad de Router Cisco

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

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Ilustración – 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.

## 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 – 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.

## Scalpel Tool

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



Ilustración – 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.

## Autopsy tool

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



Ilustración – 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.

## Chkrootkit



Ilustración – 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.

# 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>