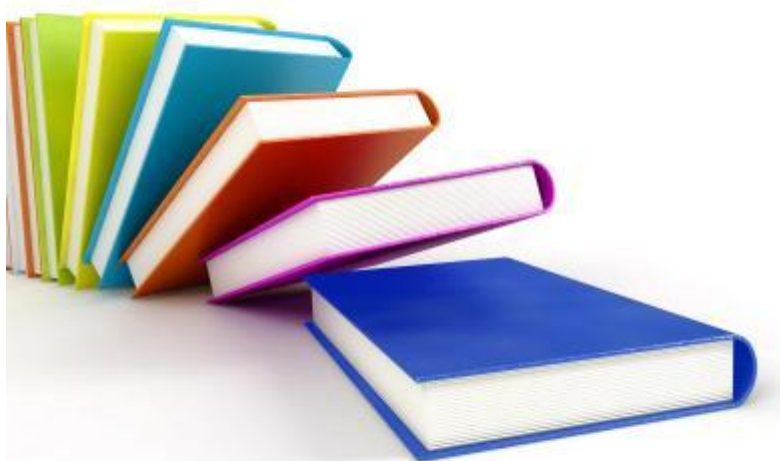


ANEXO B



ANEXO B

Práctica 2. Análisis de una memoria flash para encontrar información relacionada con narcotraficantes

Introducción

La misión es realizar un análisis forense de la imagen de una memoria flash y entregar un informe técnico exhaustivo con todos los pasos seguidos durante la investigación.

Información de referencia

La policía lleva alrededor de 2 meses tras los pasos de una red de narcotraficantes, consiguiendo averiguar que en unos pocos días va a tener lugar uno de los intercambios de cocaína más importantes de los últimos tiempos.

Justo antes de esta entrega la policía logra atrapar a uno de los integrantes de la banda, el cual tenía en su poder una memoria USB, la cual se piensa puede contener información muy valiosa para dar con el paradero de todos los integrantes de la banda y del intercambio que se pretende realizar.

La policía antes de entregar la memoria USB al equipo especialista en informática forense logra sacar información al sospecho:

- Sistema de Archivos: ext2
- Información de utilidad: fecha, hora y lugar de intercambio, todo según el sospechoso se encuentra en la memoria.

El sospechoso también advierte que se tomaron todas las medidas para que la información no pueda ser leída ni evidente a primera vista.

Objetivos del análisis forense

Los objetivos son obtener la siguiente información:

- Fecha y hora de la entrega.
- Nombre del jefe de la banda de narcotraficantes.

- Lugar del intercambio.

Análisis

El primer paso consiste en descargar la imagen proporcionada de la memoria USB, para luego hacer un análisis de integridad de la información mediante la suma de control md5.

```
[root@Forense Imagen]# ls
sdb1.dd  sdb1.md5
[root@Forense Imagen]# md5sum -c sdb1.md5
sdb1.dd: OK
```

Como se puede apreciar al análisis de integridad es correcto, luego se procede a sacar 5 copias de la imagen de la memoria USB, grabar cada copia en un CD, etiquetarlos y almacenarlos.

Una vez verificada la integridad, le sacan copias de la imagen y se procede a crear un caso en Autopsy, el cual ya fue configurado previamente en la máquina donde se realizará el análisis.

```
[root@Forense autopsy-2.24]# ./autopsy
```

```
=====
Autopsy Forensic Browser
http://www.sleuthkit.org/autopsy/
ver 2.24
=====
Evidence Locker: /PracticaForense/Autopsy
Start Time: Tue Nov 23 21:41:45 2010
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
```



Página de inicio de Autopsy

En la ventana de inicio se selecciona NewCase y se ingresa los datos correspondientes.

CREATE A NEW CASE

1. **Case Name:** The name of this investigation. It can contain only letters, numbers, and symbols.

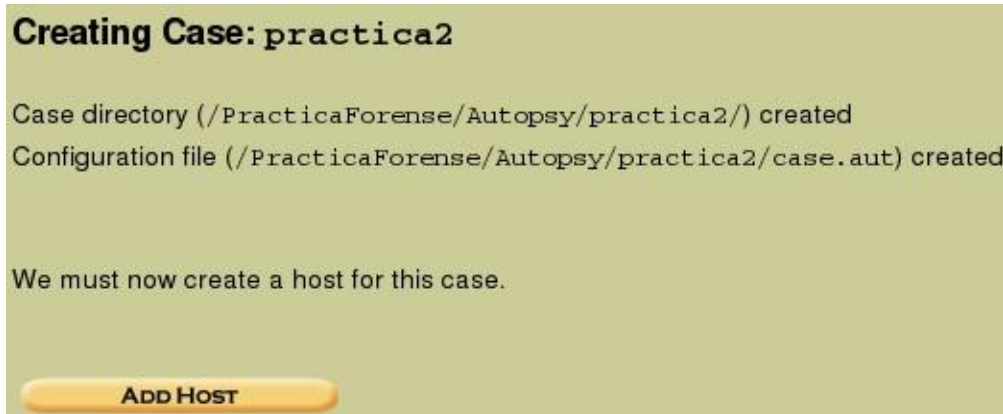
2. **Description:** An optional, one line description of this case.

3. **Investigator Names:** The optional names (with no spaces) of the investigators for this case.

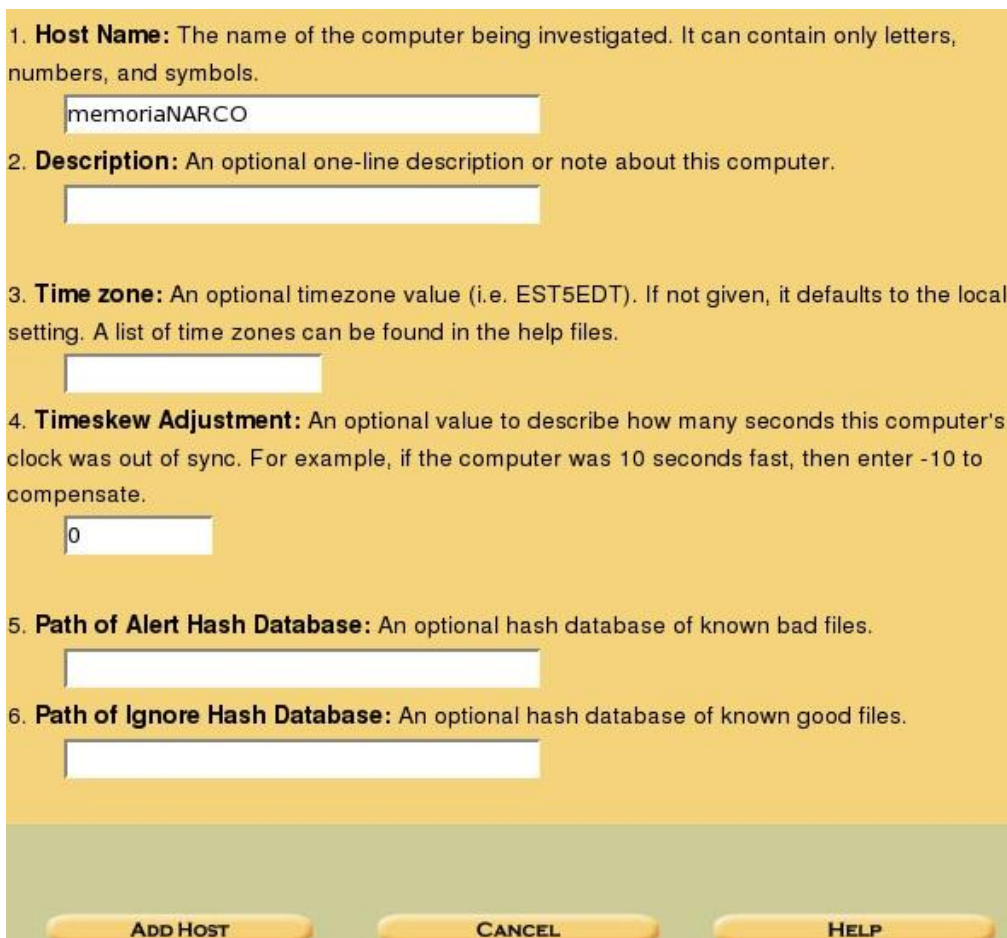
| | | | |
|----|--|----|----------------------|
| a. | <input type="text" value="Omar Almeida Romo"/> | b. | <input type="text"/> |
| c. | <input type="text"/> | d. | <input type="text"/> |
| e. | <input type="text"/> | f. | <input type="text"/> |
| g. | <input type="text"/> | h. | <input type="text"/> |
| i. | <input type="text"/> | j. | <input type="text"/> |

Creación de un caso nuevo

Una vez ingresados los datos el programa crea los directorios donde se almacenará toda la información del caso.



Seleccionamos el botón "Add Host" que nos lleva a la siguiente pantalla de configuración en donde se ingresa la información que se solicite.

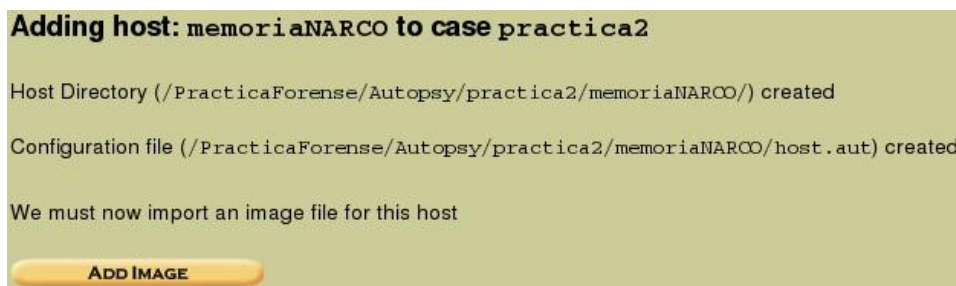


The screenshot shows a configuration dialog box with a yellow background. It contains six numbered fields with descriptions and input boxes:

- 1. Host Name:** The name of the computer being investigated. It can contain only letters, numbers, and symbols.
- 2. Description:** An optional one-line description or note about this computer.
- 3. Time zone:** An optional timezone value (i.e. EST5EDT). If not given, it defaults to the local setting. A list of time zones can be found in the help files.
- 4. Timeskew Adjustment:** An optional value to describe how many seconds this computer's clock was out of sync. For example, if the computer was 10 seconds fast, then enter -10 to compensate.
- 5. Path of Alert Hash Database:** An optional hash database of known bad files.
- 6. Path of Ignore Hash Database:** An optional hash database of known good files.

At the bottom, there are three yellow buttons: "ADD HOST", "CANCEL", and "HELP".

En este caso no se trata de una imagen de un disco duro, por eso se pone ese nombre, en lo que se refiere a la zona horaria no se entregó información al respecto, como si la computadora tenía un desajuste de hora, por eso se asume que es la misma, tampoco se proporciona sobre alguna alerta o algún dato que se tenga que ignorar, también se deja en blanco y se selecciona “Add Host”.



Ahora se añade al caso la imagen de la memoria USB, se pulsa en “Add Image” y a continuación en “Add Image File”.



El programa pide a continuación la ruta de la imagen y el método de importación que se utilizará.

1. Location
 Enter the full path (starting with /) to the image file.
 If the image is split (either raw or EnCase), then enter "*" for the extension.

2. Type
 Please select if this image file is for a disk or a single partition.

Disk Partition

3. Import Method
 To analyze the image file, it must be located in the evidence locker. It can be imported from its current location using a symbolic link, by copying it, or by moving it. Note that if a system failure occurs during the move, then the image could become corrupt.

Symlink Copy Move

NEXT

Seleccionamos "Next" e inmediatamente nos solicita los datos de la imagen, como si se dispone de hash MD5 lo proporcionamos y le pedimos que lo verifique después de importar la imagen.

Image File Details

Local Name: images/sdb1.dd

Data Integrity: An MD5 hash can be used to verify the integrity of the image. (With split images, this hash is for the full image file)

Ignore the hash value for this image.
 Calculate the hash value for this image.
 Add the following MD5 hash value for this image:

Verify hash after importing?

File System Details

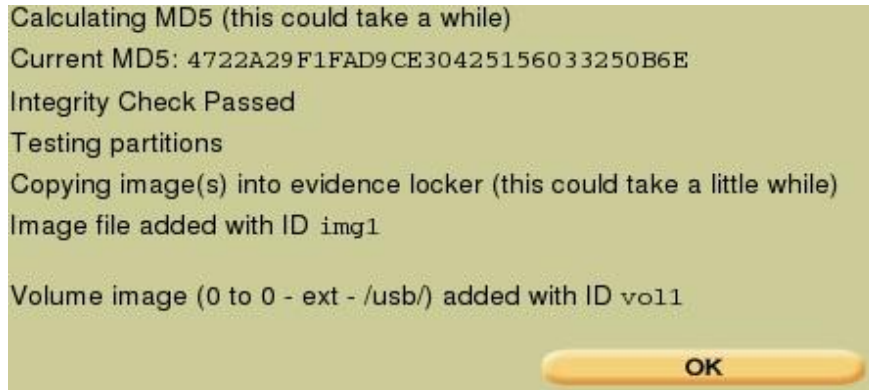
Analysis of the image file shows the following partitions:

Partition 1 (Type: ext2)

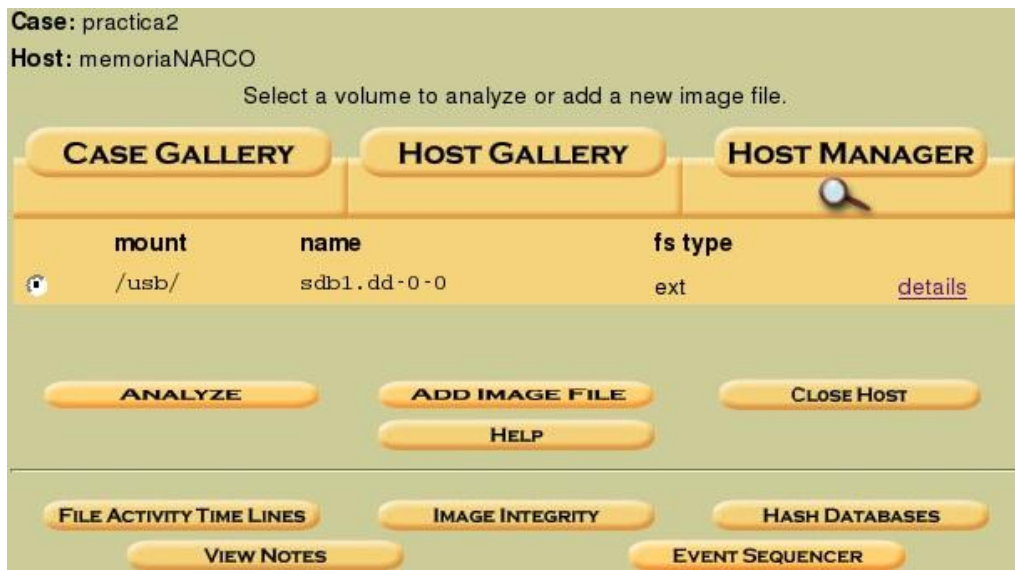
Mount Point: File System Type:

ADD **CANCEL** **HELP**

Se pulsa “Add” para que verifique la imagen.



Si el cálculo y la importación fueron correctos se puede pulsar “OK”, para ir a la ventana principal de gestión de casos, para comenzar ahora si con el análisis de la memoria USB.



Para comenzar con el análisis se pulsa “Analyze” y aparece la página principal de análisis.



Comenzamos ejecutando “Image Details”, donde se recaba información sobre el sistema de archivos, tamaño de bloque, últimas fechas de acceso, espacio libre, tamaño total.

| General File System Details |
|--|
| FILE SYSTEM INFORMATION |
| File System Type: Ext2 |
| Volume Name: |
| Volume ID: 113ed8baafcabc434b5051146a69 |
| Last Written at: Sat Sep 5 12:01:50 2009 |
| Last Checked at: Sat Sep 5 11:59:52 2009 |
| Last Mounted at: Sat Sep 5 12:00:05 2009 |
| Unmounted properly |
| Last mounted on: |
| Source OS: Linux |
| Dynamic Structure |
| Compat Features: Ext Attributes, Resize Inode, Dir Index |
| InCompat Features: Filetype, |
| Read Only Compat Features: Sparse Super, |
| |
| METADATA INFORMATION |
| Inode Range: 1 - 28113 |
| Root Directory: 2 |
| Free Inodes: 28099 |
| |
| CONTENT INFORMATION |
| Block Range: 0 - 112419 |
| Block Size: 1024 |
| Reserved Blocks Before Block Groups: 1 |
| Free Blocks: 107261 |
| |
| BLOCK GROUP INFORMATION |
| Number of Block Groups: 14 |
| Inodes per group: 2008 |
| Blocks per group: 8192 |
| Group: 0: |
| Inode Range: 1 - 2008 |
| Block Range: 1 - 8192 |
| Layout: |
| Super Block: 1 - 1 |
| Group Descriptor Table: 2 - 2 |

Ahora seleccionamos “File Analysis”, nos presenta el listado de los archivos y directorios encontrados.

Current Directory: /usb/

ADD NOTE GENERATE MDS LIST OF FILES

| DEL | Type dir / in | NAME | WRITTEN | ACCESSED | CHANGED | SIZE | UID | GID | META |
|-----|------------------|--------------------------------|------------------------------|------------------------------|------------------------------|-------|-----|-----|-----------------------|
| | d / d | \$OrphanFiles/ | 0000-00-00 00:00:00 (UTC) | 0000-00-00 00:00:00 (UTC) | 0000-00-00 00:00:00 (UTC) | 0 | 0 | 0 | 28113 |
| | d / d | ../ | 2009-09-05 12:01:41 (PDT) | 2009-09-05 12:01:40 (PDT) | 2009-09-05 12:01:41 (PDT) | 1024 | 0 | 0 | 2 |
| | d / d | ./ | 2009-09-05 12:01:41 (PDT) | 2009-09-05 12:01:40 (PDT) | 2009-09-05 12:01:41 (PDT) | 1024 | 0 | 0 | 2 |
| | r / r | jlo.jpg | 2009-09-05 12:00:35 (PDT) | 2009-09-05 12:01:23 (PDT) | 2009-09-05 12:00:35 (PDT) | 43769 | 0 | 0 | 12 |
| | d / d | lost+found/ | 2009-09-05 11:59:52 (PDT) | 2009-09-05 11:59:52 (PDT) | 2009-09-05 11:59:52 (PDT) | 12288 | 0 | 0 | 11 |
| ✓ | r / - | mail | 0000-00-00 00:00:00 (UTC) | 0000-00-00 00:00:00 (UTC) | 0000-00-00 00:00:00 (UTC) | 0 | 0 | 0 | 0 |
| | r / r | script.sh | 2009-09-05 12:01:05 (PDT) | 2009-09-05 12:01:05 (PDT) | 2009-09-05 12:01:05 (PDT) | 10105 | 0 | 0 | 14 |

Nos podemos dar cuenta que existe un archivo denominado “mail” que no pudo ser recuperado, un archivo llamado “jlo.jpg”, otro llamado “script.sh” y un directorio denominado “lost+found”, dentro de este directorio no encontramos datos.

Como primer paso verificamos el archivo “jlo.jpg” y se puede advertir que no se reconoce como un archivo de imagen, ya que presenta una cabecera de solo ceros.

| DEL | Type dir / in | NAME | WRITTEN | ACCESSED | CHANGED | SIZE | UID | GID | META |
|-----|------------------|--------------------------------|------------------------------|------------------------------|------------------------------|-------|-----|-----|-----------------------|
| | d / d | \$OrphanFiles/ | 0000-00-00 00:00:00 (UTC) | 0000-00-00 00:00:00 (UTC) | 0000-00-00 00:00:00 (UTC) | 0 | 0 | 0 | 28113 |
| | d / d | ../ | 2009-09-05 12:01:41 (PDT) | 2009-09-05 12:01:40 (PDT) | 2009-09-05 12:01:41 (PDT) | 1024 | 0 | 0 | 2 |
| | d / d | ./ | 2009-09-05 12:01:41 (PDT) | 2009-09-05 12:01:40 (PDT) | 2009-09-05 12:01:41 (PDT) | 1024 | 0 | 0 | 2 |
| | r / r | jlo.jpg | 2009-09-05 12:00:35 (PDT) | 2009-09-05 12:01:23 (PDT) | 2009-09-05 12:00:35 (PDT) | 43769 | 0 | 0 | 12 |

ASCII (display - report) * Hex (display - report) * ASCII Strings (display - report) * Export * Add Note

File Type: data

Hex Contents Of File: /usb/jlo.jpg

```

00000000: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000010: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000030: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000040: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000050: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000060: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000070: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000080: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000090: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000A0: 0000 0000 0000 0000 0000 0000 0000 0000 .....

```

Procedemos a revisar los metadatos pulsando sobre el valor "12" que aparece en la columna "Meta".

```
Pointed to by file:
/usb/j1o.jpg

File Type:
data

MD5 of content:
f380d7c0196a63be4fc7c3a6b3719e61

SHA-1 of content:
82f55d4c72e3295e18155f133d44841c8e401e5b

Details:

inode: 12
Allocated
Group: 0
Generation Id: 1076238465
uid / gid: 0 / 0
mode: rw-r--r--
size: 43769
num of links: 1
inode: 12
Allocated
Group: 0
Generation Id: 1076238465
uid / gid: 0 / 0
mode: rw-r--r--
size: 43769
num of links: 1

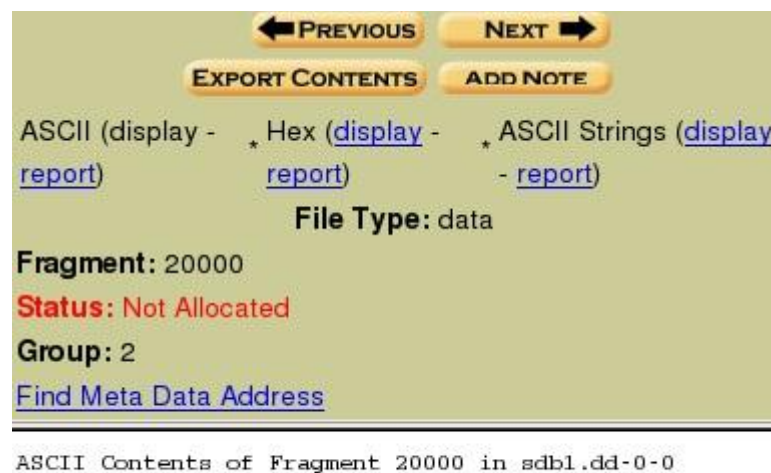
Inode Times:
Accessed: Sat Sep 5 12:01:23 2009
File Modified: Sat Sep 5 12:00:35 2009
Inode Modified: Sat Sep 5 12:00:35 2009

Direct Blocks:
20000 7682 7683 7684 7685 7686 7687 7688
7689 7690 7691 7692 7694 7695 7696 7697
7698 7699 7700 7701 7702 7703 7704 7705
7706 7707 7708 7709 7710 7711 7712 7713
7714 7715 7716 7717 7718 7719 7720 7721
7722 7723 7724

Indirect Blocks:
7693
```

Siempre que se analice los metadatos se debe verificar que el tamaño del archivo coincida con el número total de bloques asignados.

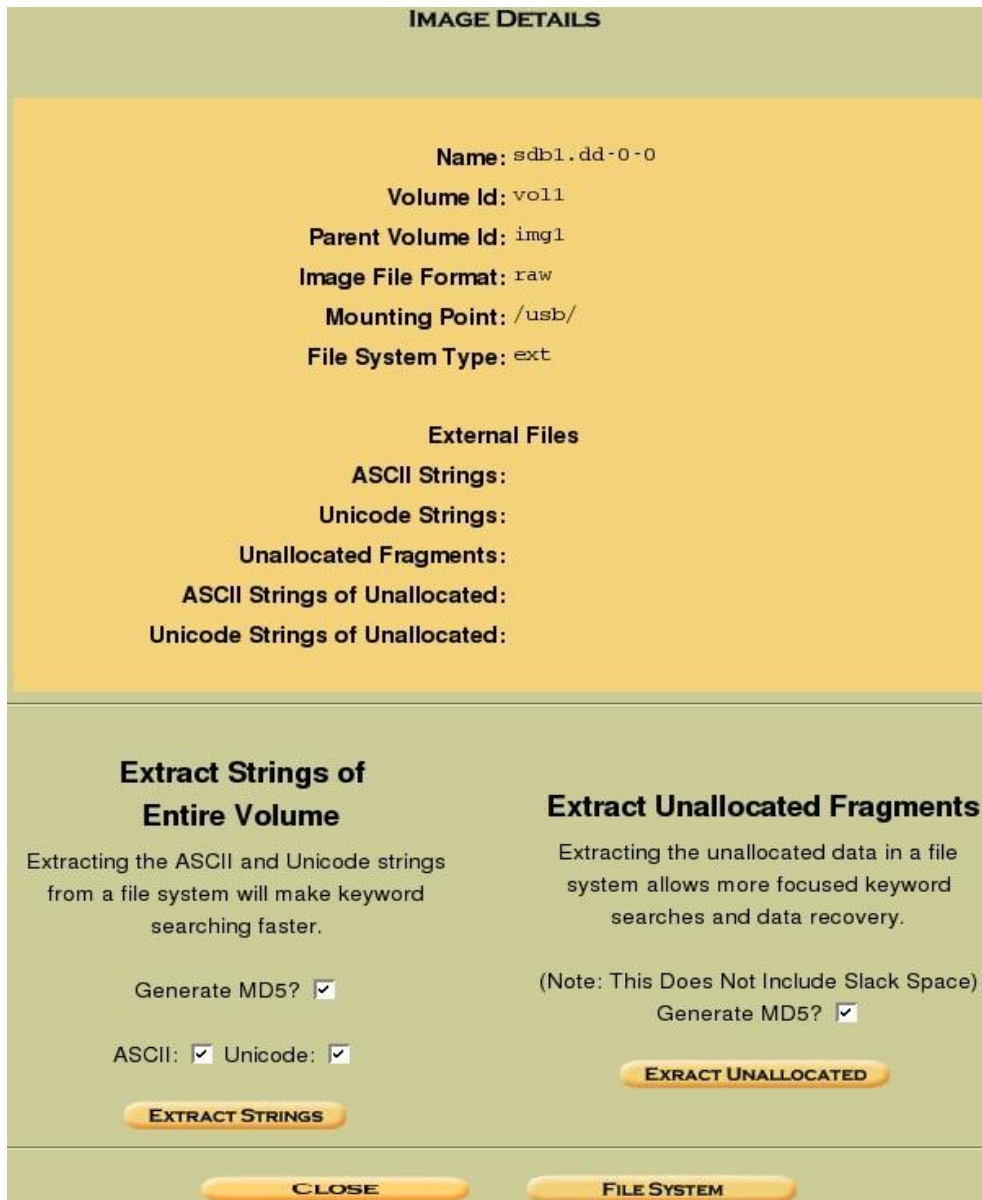
A simple vista llama la atención el bloque 20000, que no concuerda con la secuencia del resto de bloques y el bloque 7693 que rompe la secuencia de asignación de bloques. Pulsando sobre el bloque 20000 se puede verificar que se trata de un espacio no asignado y está vacío.



Como se menciona con anterioridad en el Anexo A si se tratase de una imagen esta debe comenzar con cualquiera de las siguientes cabeceras:

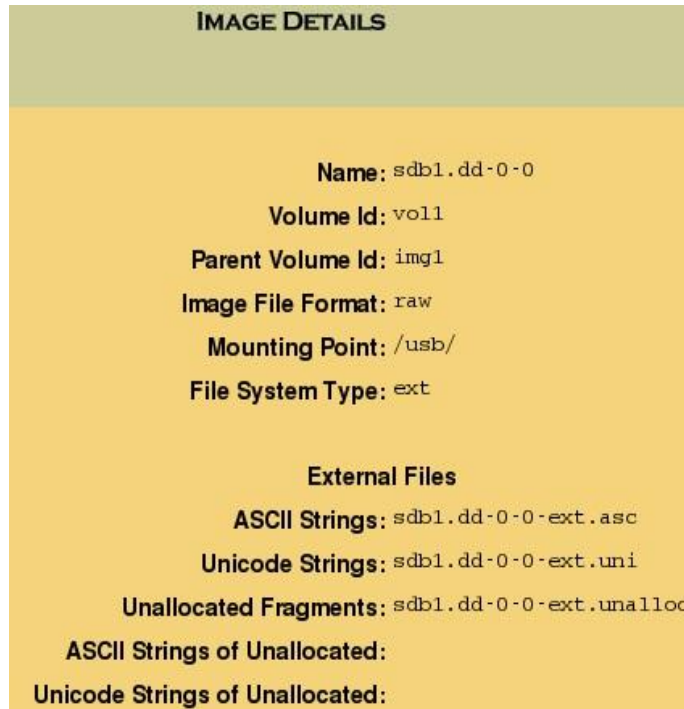
- FF D8 FF E0
- XX XX
- 4^a 46ÿØÿà..JFIF

Lo siguiente es hacer una búsqueda de estas cabeceras, para ello es necesario extraer las cadenas ASCII y UNICODE. Para lograr esto se debe escoger el botón “Close” en el menú superior, y se vuelve a la pantalla “Host Manager”. Se selecciona el enlace “details”



Lo que hará a continuación es extraer las cadenas de texto ASCII y Unicode¹ para acelerar las búsquedas que se tengan que realizar. Primero se hará en el sistema de archivos y a continuación en los fragmentos no asignados pulsando primero sobre “Extract Strings” y a continuación sobre “Extract Unallocated”.

¹ “El Estándar Unicode es un estándar de codificación de caracteres diseñado para facilitar el tratamiento informático, transmisión y visualización de textos de múltiples lenguajes y disciplinas técnicas además de textos clásicos de lenguas muertas. El término Unicode proviene de los tres objetivos perseguidos: universalidad, uniformidad y unicidad”.(Tomado de Wikipedia)



Una vez realizado esto regresamos a la pantalla principal de análisis y seleccionamos “Keyword Search”.



Ingresamos “JFIF²” en el cuadro de búsqueda y se hace clic en “search” y en el resultado de la derecha se escoge “HEX”.

² El formato de intercambio de archivos JPEG (JFIF) es un archivo de imagen en formato estándar.

Searching for ASCII: Done
Saving: Done
1 hits- [link to results](#)

Searching for Unicode: Done
Saving: Done
0 hits

[New Search](#)

1 occurrence of JFIF was found

Search Options:
ASCII
Case Sensitive

Fragment 7681 ([Hex](#) - [Ascii](#))
1: 6 (JFIF)

JFIF was not found

Search Options:
Unicode
Case Sensitive

PREVIOUS NEXT
EXPORT CONTENTS ADD NOTE

ASCII ([display - report](#)) * Hex ([display - report](#)) * ASCII Strings ([display - report](#))
File Type: JPEG image data, JFIF standard 1.02

Fragment: 7681
Status: Allocated
Group: 0
[Find Meta Data Address](#)

Hex Contents of Fragment **7681** in sdbl.dd-0-0

| | | | | | | |
|-----|----------|----------|----------|----------|-------|--------------|
| 0 | ffd8ffe0 | 00104a46 | 49460001 | 02000001 | | JF IF..... |
| 16 | 00010000 | ffe00008 | 4f636164 | 3030ffdb | | Ocad 00..... |
| 32 | 00840004 | 04040608 | 06080808 | 08080808 | | |
| 48 | 08080b0a | 0a0a0a0a | 0e0a0a0a | 0b100e11 | | |
| 64 | 11100e10 | 0f12141a | 16121318 | 130f1016 | | |
| 80 | 1f17181b | 1bd1d1d1 | 11162022 | 1f1c221a | | |
| 96 | 1c1d1c01 | 05101020 | 20202020 | 20204040 | | ..@.. |
| 112 | 40404080 | 80808080 | 80808080 | 80808080 | | |
| 128 | 80808080 | 80808080 | 80808080 | 80808080 | | |
| 144 | 80808080 | 80808080 | 80808080 | 80808080 | | |
| 160 | 80808080 | ffc00011 | 0801c201 | 5e030111 | | ^..... |
| 176 | 00021101 | 031101ff | c400a300 | 00020203 | | |
| 192 | 01010100 | 00000000 | 00000000 | 05060407 | | |
| 208 | 02030801 | 00090100 | 02030101 | 00000000 | | |
| 224 | 00000000 | 00000002 | 03000104 | 05061000 | | |

Se encuentra "JFIF" una sola vez y en el bloque 7681, si se recuerda la estructura de bloques se puede observar que se trata del bloque que hacía falta, el paso siguiente es extraer los bloques para ver si tienen sentido.

Para lograr esto se selecciona "Data Unit" en la parte superior, y en la casilla "Fragment Number" se introduce el valor 7681, en la casilla "Number of Fragments" el valor 12 y se hace clic en "View", con todo esto lo que se pretende hacer es extraer desde el bloque 7681 hasta el 79692 y luego desde el 7694 hasta el 7724, es decir evitando el bloque 7693.

Fragment Number:

Number of Fragments:

Fragment Size: 1024

Address Type:

Lazarus Addr:

[VIEW](#)

[ALLOCATION LIST](#)

[LOAD UNALLOCATED](#)

PREVIOUS NEXT
EXPORT CONTENTS ADD NOTE

ASCII ([display - report](#)) * Hex ([display - report](#)) * ASCII Strings ([display - report](#))
File Type: JPEG image data, JFIF standard 1.02

Fragments: 7681-7692
Status: Allocated
Group: 0

ASCII Contents of Fragments 7681-7692 in sdbl.dd-0-0

```

.....JFIF.....Ocad00.....
.....".....@.....
7E.....i:n.....t.A.m)n=08.D...I.r)...B...:~%]&.....<j.../..b.Gg...?..H...riQ.....l_
..
>...jA...p...0H_v.m...o...LP...q...4Gf?(?.(Oje...G.6.../...q...n.n.H...OH..?..y..l.r{.....a..i.3.E
.....lQ.....#-Or.....>...j^..O.....7.....5TZ.....I.....@.....cmD.....{}>...[I.....$.r.Z4..6...AV3.[
t.....|...7.$...4...).G.....u6.D...x...x...
=.....2.?.....c...l...
.....i\.....@?|...S...5...E..7..l.<q&.....$.b..i..Fn.....'0a...Z.....3-c.=..|.9?..&l..".....
..h\.....I.o.v...9...>.Z[...r...0]n...j;...i...#aT...am$.+W.16e.U...9ell...
.:#.C...;.>7...#

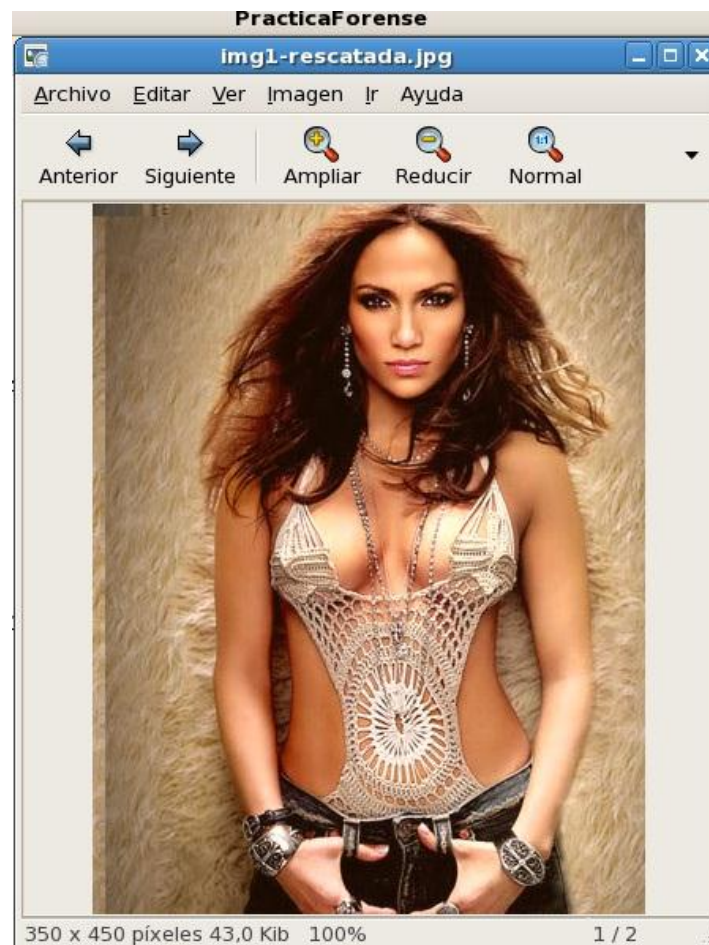
```

Se hace clic en “Export Contents” y se guarda el archivo como “vol1-Fragment7681.raw”, luego hacemos lo mismo para el siguiente fragmento, con los valores 7694 y 31, se guarda el segundo archivo como “vol1-Fragment7694.raw”.

Con las dos partes se intenta ver cuál es el contenido, para esto se utiliza el comando cat, que también se había utilizado en el anexo A.

```
[root@Forense PracticaForense]# cat voll-Fragment* > img1-rescatada.jpg
[root@Forense PracticaForense]# ls
afflib-3.6.4          firefox              libewf-20100226.tar.gz
afflib-3.6.4.tar.gz  imagen              sleuthkit-3.2.0
Autopsy              imagen-dd.tar.gz    sleuthkit-3.2.0.tar.gz
autopsy-2.24         img1-rescatada.jpg voll-Fragment7681.raw
autopsy-2.24.tar.gz libewf               voll-Fragment7694.raw
```

Y se obtiene la siguiente imagen:



Por lo general y como ya se vio en el Anexo A, este tipo de imágenes puede ser utilizada para ocultar información. Para encontrar la forma más básica de ocultación hacemos clic en el enlace “Report” de la sección “ASCII Strings” y se revisa el archivo generado en busca de cualquier información valiosa.

CONTENT

```
JFIF
Ocad00
@@@@@
!"1A
!AQa"2Bq
,Z6C
< pw=uj1&5632
m)n=@8
->%]&
b,Gg
4Gf?(?
(Oje
1;{\
#-Or
n1Pi
]u!S
)am$
{ahg
LF=3E
$OH~
$`=
u$"f
,r9"
e4!E
!Ydj
u:dt
~-=mrz
V1KK
uUQ$
```

Antes de continuar investigando no nos podemos olvidar del otro archivo que se encuentra en la imagen de la memoria.


```
MD5 of content:
1eb4046ac4bfd52821c5395967fd2667 -

SHA-1 of content:
f7e4de06937412cff6617390593273d9fecb4860 -

Details:

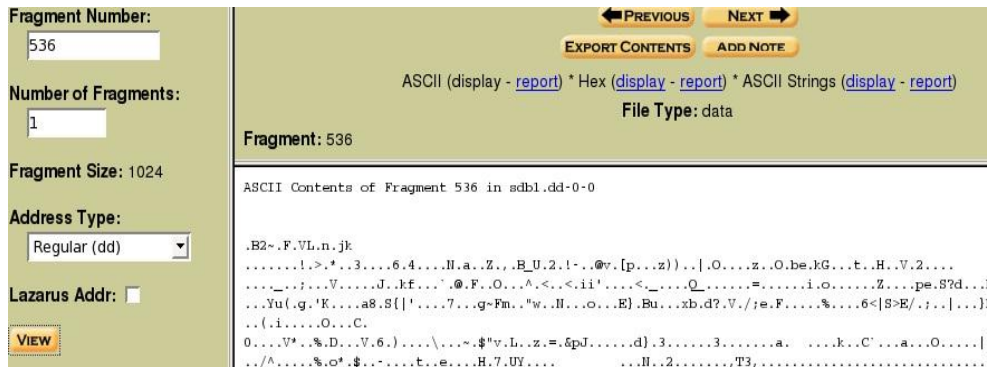
inode: 14
Allocated
Group: 0
Generation Id: 1076238467
uid / gid: 0 / 0
mode: rw-r--r--
size: 10105
num of links: 1

Inode Times:
Accessed: Sat Sep 5 12:01:05 2009
File Modified: Sat Sep 5 12:01:05 2009
Inode Modified: Sat Sep 5 12:01:05 2009

Direct Blocks:
527 528 529 530 531 532 533 534
535 0
```

Si se analizan los bloques nuevamente se encuentra algo fuera de lugar porque la secuencia en lugar de ser 536 es 0, si se hace clic sobre el 0 para ver el contenido y este se encuentra vacío. Pero si regresamos a "Data Unit" y buscamos el bloque 536 si se encuentra datos.

```
Fragment: 0
-----
ASCII Contents of Fragment 0 in sdbl.dd-0-0
```

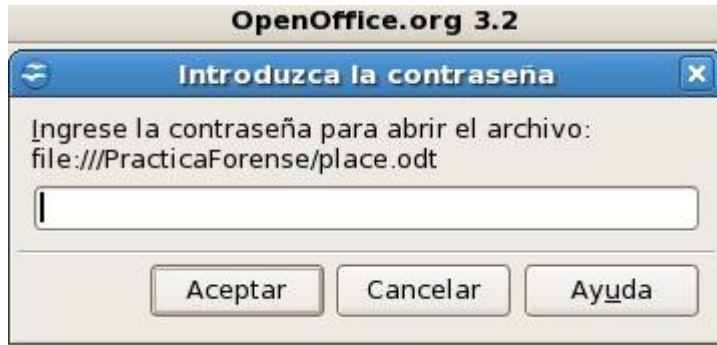


Ahora se procede a extraer los archivos con el mismo procedimiento descrito para la imagen y se obtiene 2 archivos.

```
[root@Forense PracticaForense]# ls
afflib-3.6.4          Imagen          sleuthkit-3.2.0.tar.gz
afflib-3.6.4.tar.gz  imagen.dd.tar.gz  voll-Fragment0.gz
Autopsy             img1-rescatada.jpg  voll-Fragment527.gz
autopsy-2.24        libewf          voll-Fragment7681.raw
autopsy-2.24.tar.gz  libewf-20100226.tar.gz  voll-Fragment7694.raw
firefox             sleuthkit-3.2.0
[root@Forense PracticaForense]# gzip -d voll-Fragment0.gz
gzip: voll-Fragment0.gz: not in gzip format
[root@Forense PracticaForense]# gzip -d voll-Fragment527.gz
[root@Forense PracticaForense]# ls
afflib-3.6.4          Imagen          sleuthkit-3.2.0.tar.gz
afflib-3.6.4.tar.gz  imagen.dd.tar.gz  voll-Fragment0.gz
Autopsy             img1-rescatada.jpg  voll-Fragment527
autopsy-2.24        libewf          voll-Fragment7681.raw
autopsy-2.24.tar.gz  libewf-20100226.tar.gz  voll-Fragment7694.raw
firefox             sleuthkit-3.2.0
```



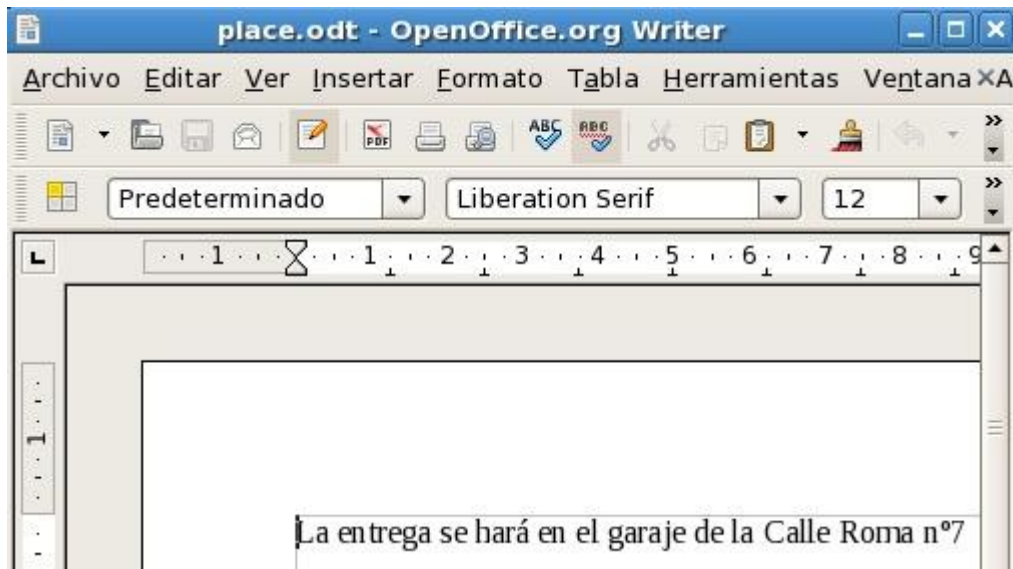
El primer archivo no se puede descomprimir, pero el segundo si y contiene un archivo de tipo "ODT", cuando se trata de ver el contenido sale la siguiente ventana:



Ante este caso es necesario revisar nuevamente todas las cadenas de texto que se obtuvieron en busca de algún dato que nos pueda servir como clave. Efectivamente en la imagen extraída se encuentra el texto:

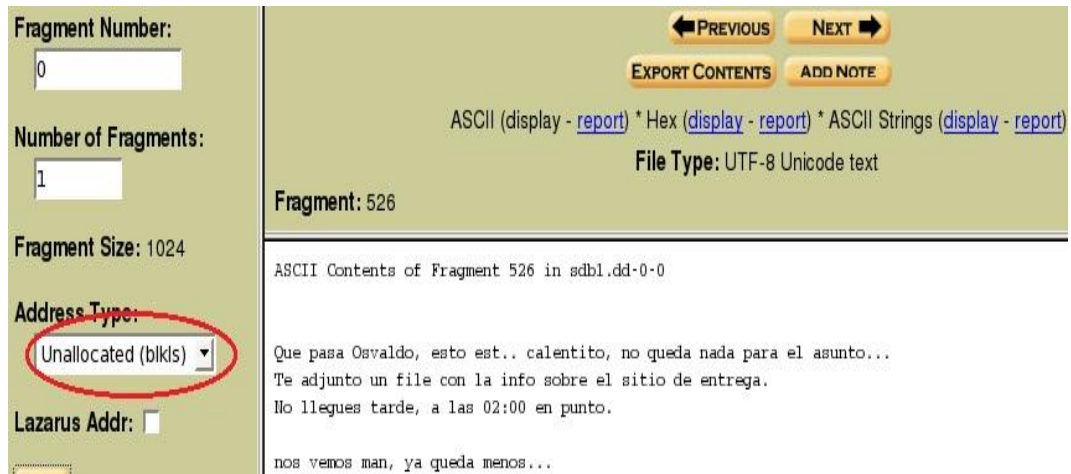
```
#####  
!"1A  
!AQa"2Bq  
,Z6C  
pw=uj1&5632  
m)n=@8  
->%] &
```

Parece ser el texto **"pw=uj1&5632"**, se prueba si es el password y se obtiene lo siguiente:



Por primera vez desde que se comenzó con el análisis parece que se encontró un dato valioso para la investigación, ahora si regresamos

a la página principal podemos hacer una búsqueda en el resto de la memoria, incluyendo las partes no legibles.



The screenshot shows a web-based interface for analyzing memory fragments. On the left, there are input fields for 'Fragment Number' (0), 'Number of Fragments' (1), 'Fragment Size' (1024), 'Address Type' (Unallocated (blks)), and 'Lazarus Addr'. On the right, there are navigation buttons (PREVIOUS, NEXT), action buttons (EXPORT CONTENTS, ADD NOTE), and a file type indicator (File Type: UTF-8 Unicode text). The main content area displays the ASCII contents of fragment 526, which is a text message in Spanish.

Fragment Number: 0
Number of Fragments: 1
Fragment Size: 1024
Address Type: Unallocated (blks)
Lazarus Addr:

Fragment: 526
File Type: UTF-8 Unicode text

ASCII Contents of Fragment 526 in sdbl.dd-0-0

Que pasa Osvaldo, esto est.. calentito, no queda nada para el asunto...
Te adjunto un file con la info sobre el sitio de entrega.
No llegues tarde, a las 02:00 en punto.

nos vemos man, ya queda menos...

Con este último dato se ha logrado obtener todas las respuestas a los objetivos planteados inicialmente:

- Fecha y hora de la entrega: **2009-09-05 / 02:00**
- Nombre del jefe de la banda de narcotraficantes: **Osvaldo**
- Lugar del intercambio: **En el garaje de la calle Roma nº 7**

Por favor no olvidar que todos estos resultados deben ser documentados y etiquetados de tal forma que se pueda luego obtener una línea de tiempo que nos indique los pasos que se siguieron de forma ordenada, incluso nos ayuda si es necesario regresar hacia algún paso previo.

Con este resultado la unidad de informática forense de la policía está contribuyendo para que puedan dismantelar la banda de narcotraficantes.