

IFA. Práctica de laboratorio 03

Hugo Fonseca Díaz
email uo258318@uniovi.es

Escuela de Ingeniería Informática. Universidad de Oviedo.

24 de junio de 2021

1. Ejercicio 1

Se crea el caso en Autopsy con los datos solicitados.

Figura 1: Ejercicio 1: Creación del caso

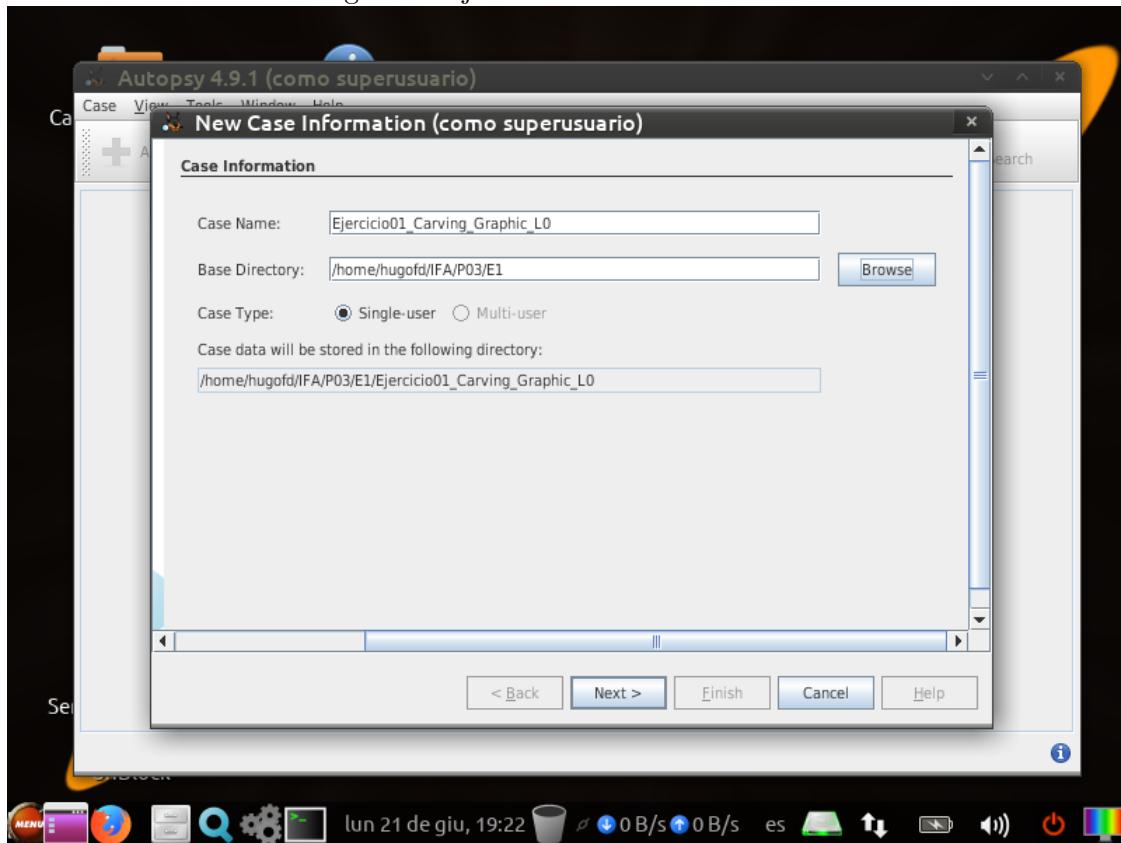
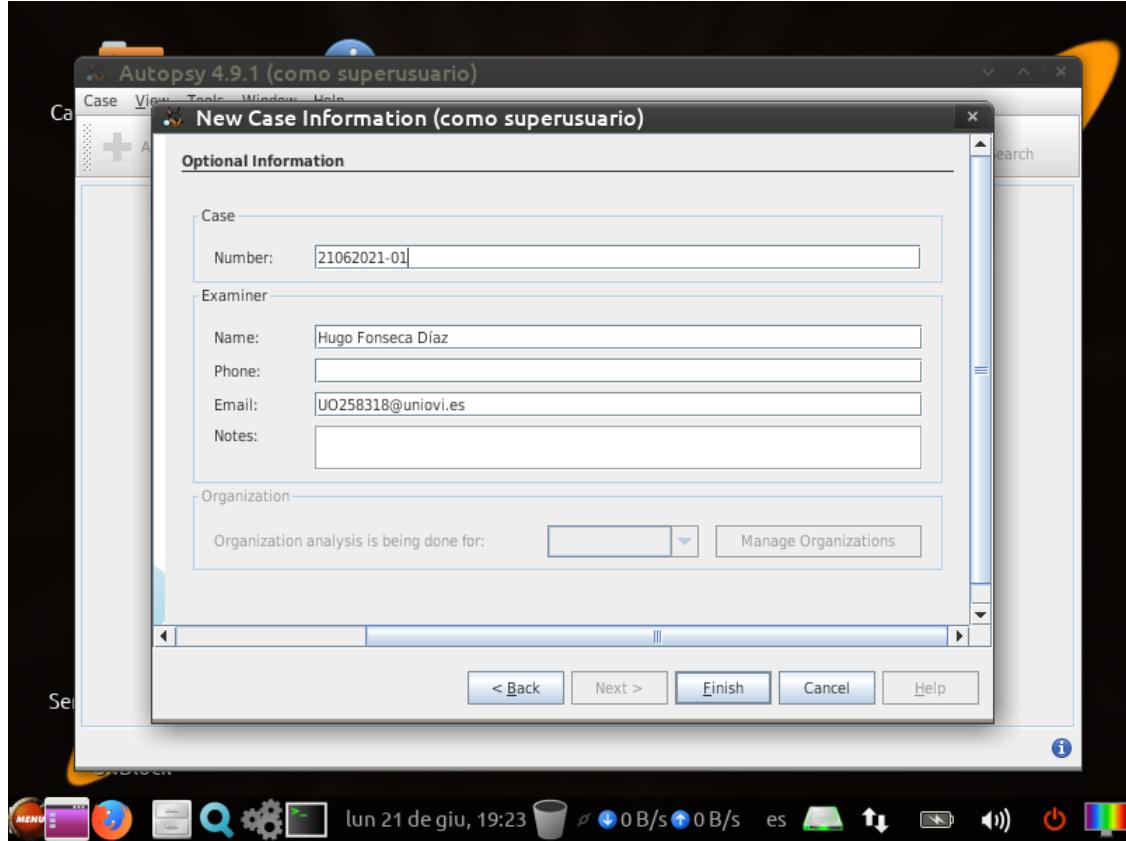
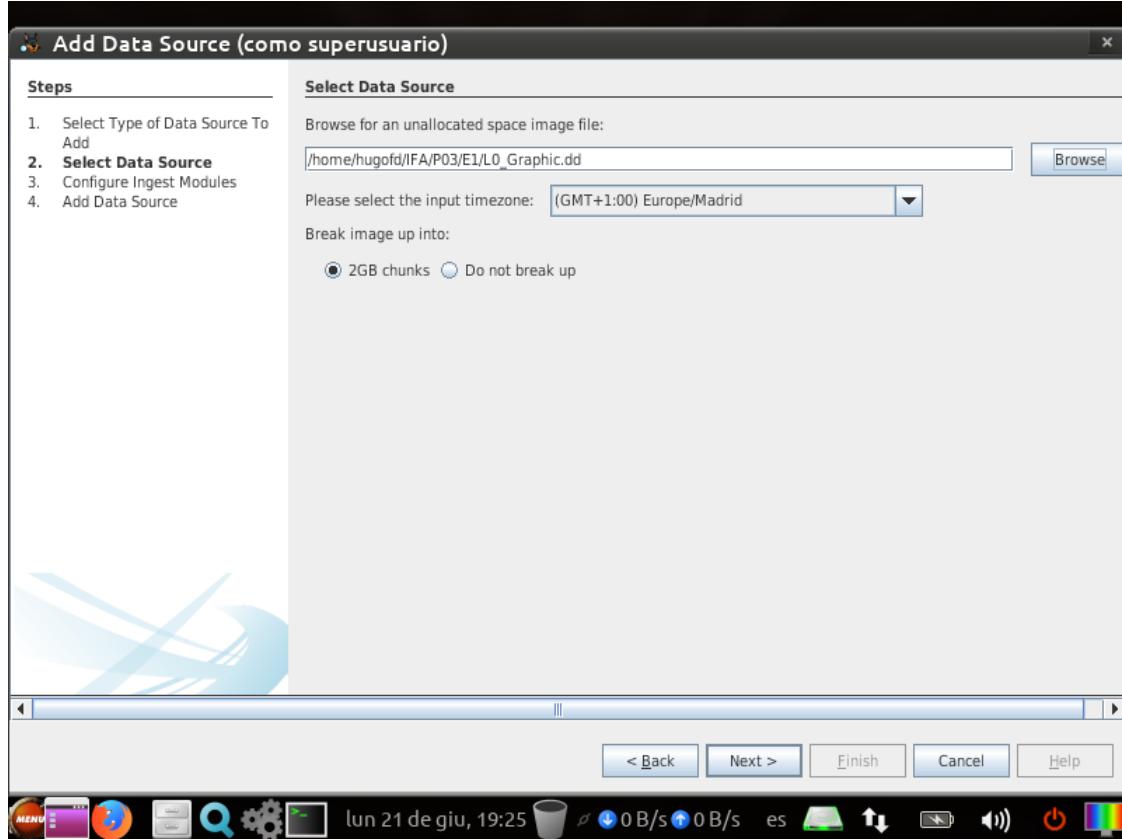


Figura 2: Ejercicio 1: Detalles del examinador



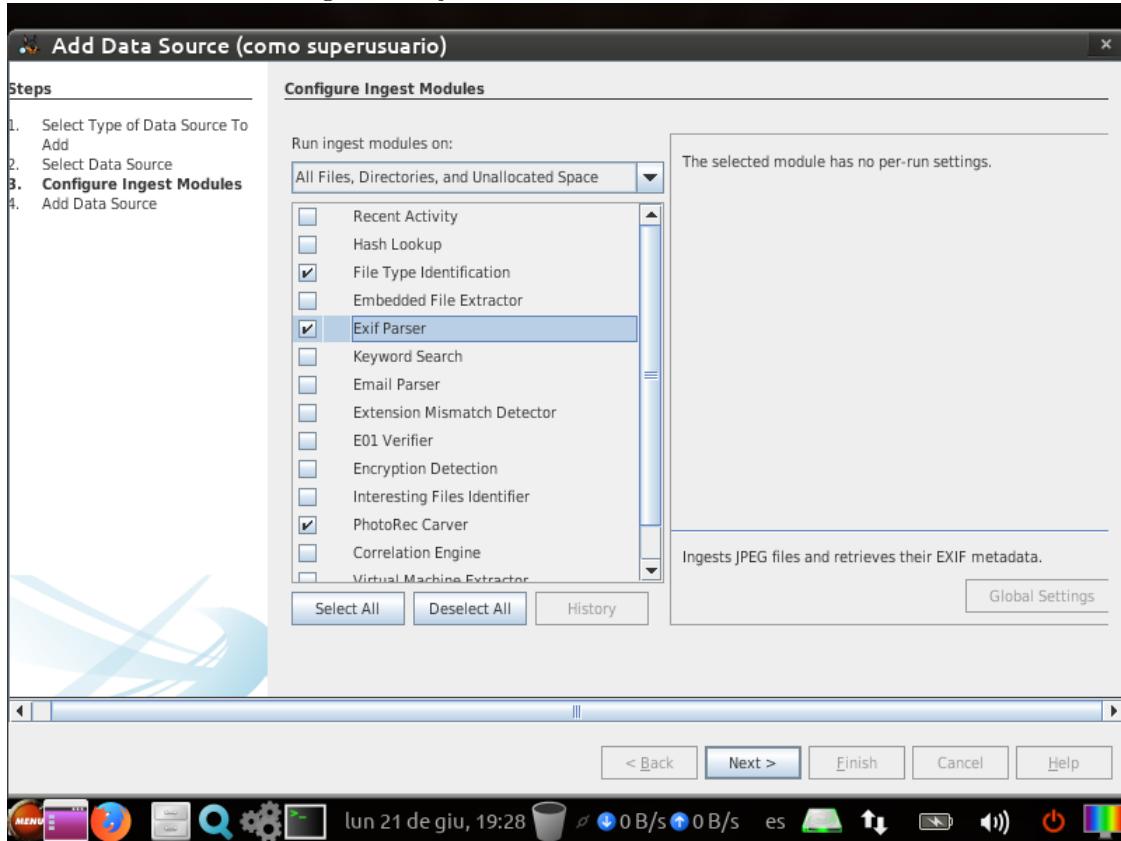
Añadimos la imagen a analizar.

Figura 3: Ejercicio 1: Selección de la imagen



Se seleccionan los módulos de identificación de tipos de fichero, parseador de Exif y *PhotoRec Carver*.

Figura 4: Ejercicio 1: Selección de módulos



Se ejecuta el análisis y se obtienen los resultados con los que se rellenará la tabla.

Figura 5: Ejercicio 1: Resultados del análisis

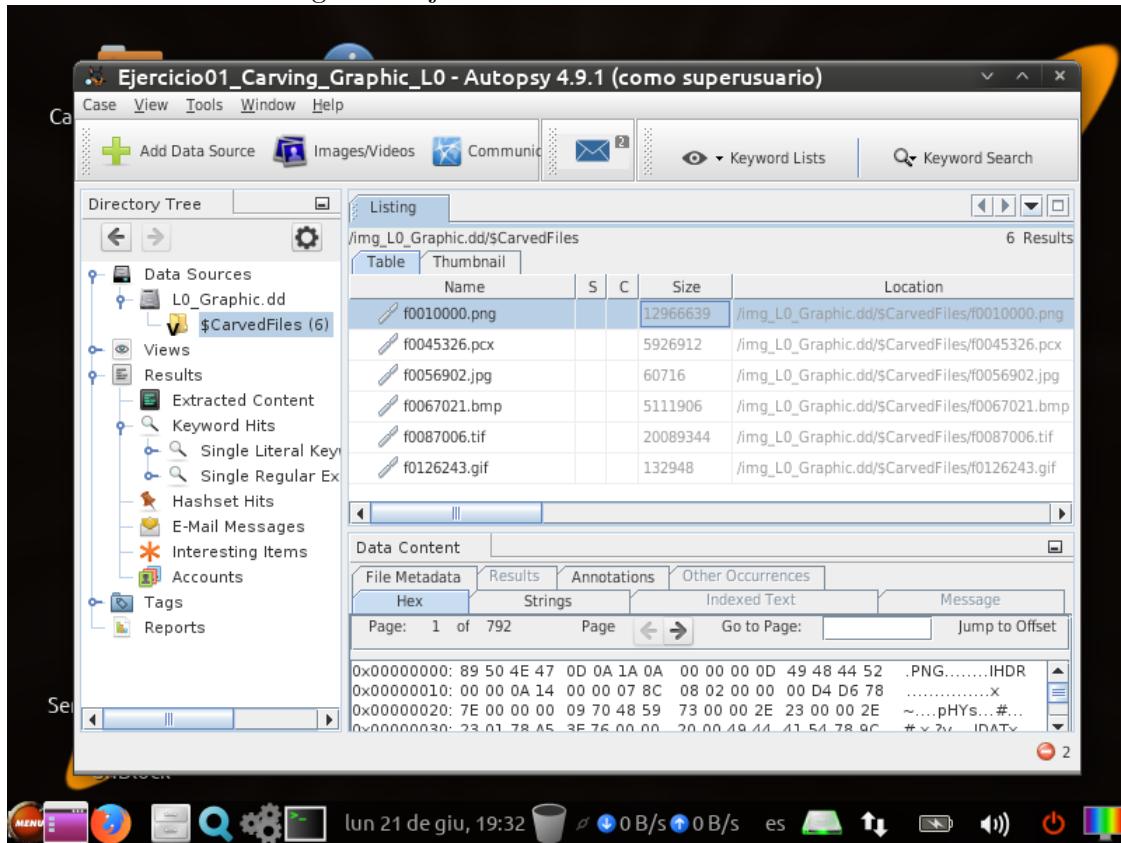
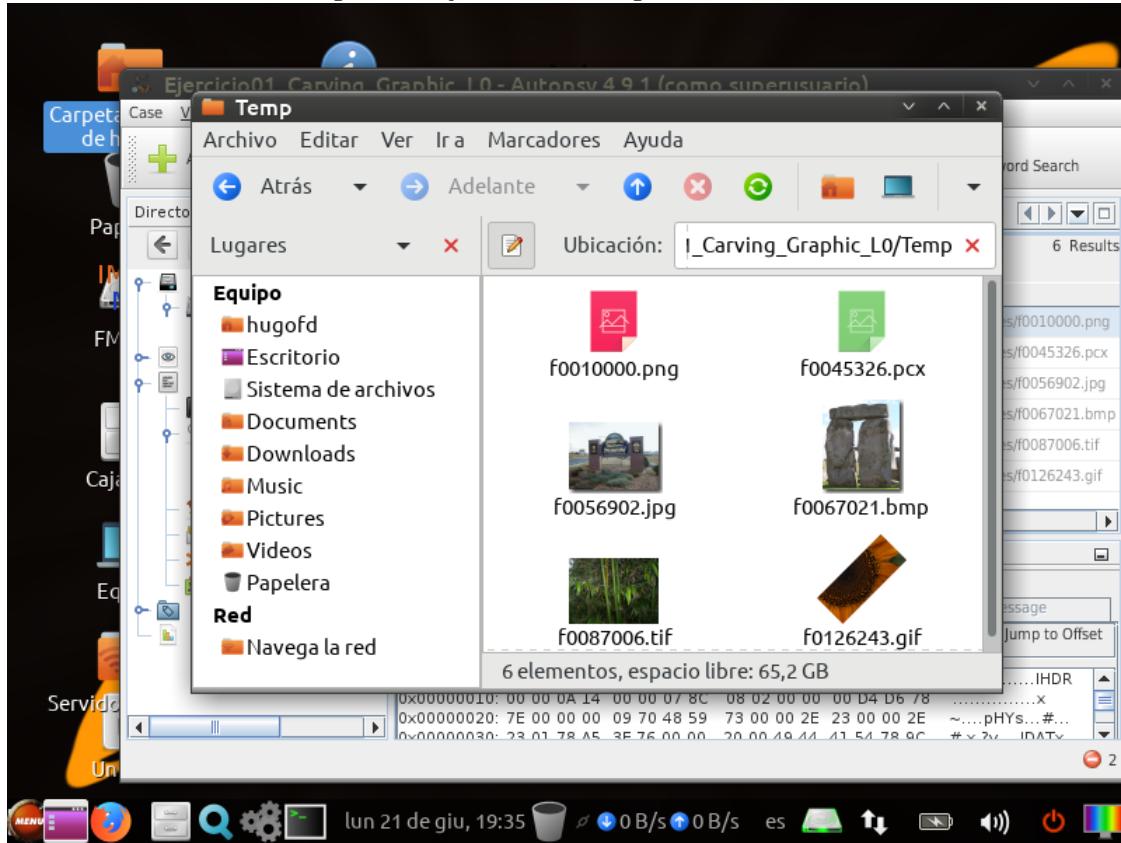


Figura 6: Ejercicio 1: Imágenes obtenidas



Para abrir el archivo con extensión *pcx* se ha utilizado un visor de imágenes online, al no disponer de uno adecuado en el equipo.

| Nombre del fichero en Autopsy | Tamaño del fichero (en Bytes) | Breve descripción imagen visible |
|-------------------------------|-------------------------------|----------------------------------|
| f0010000.png | 12966639 | Flor morada |
| f0045326.pcx | 5926912 | Iglesia y fuente |
| f0056902.jpg | 60716 | Cartel 'Welcome to Moscow' |
| f0067021.bmp | 5111906 | Piedras en forma de Pi |
| f0087006.tif | 20089344 | Cañas de bambú |
| f0126243.gif | 132948 | Girasol |

2. Ejercicio 2

Se crea el caso en Autopsy con los datos solicitados.

Figura 7: Ejercicio 2: Creación del caso

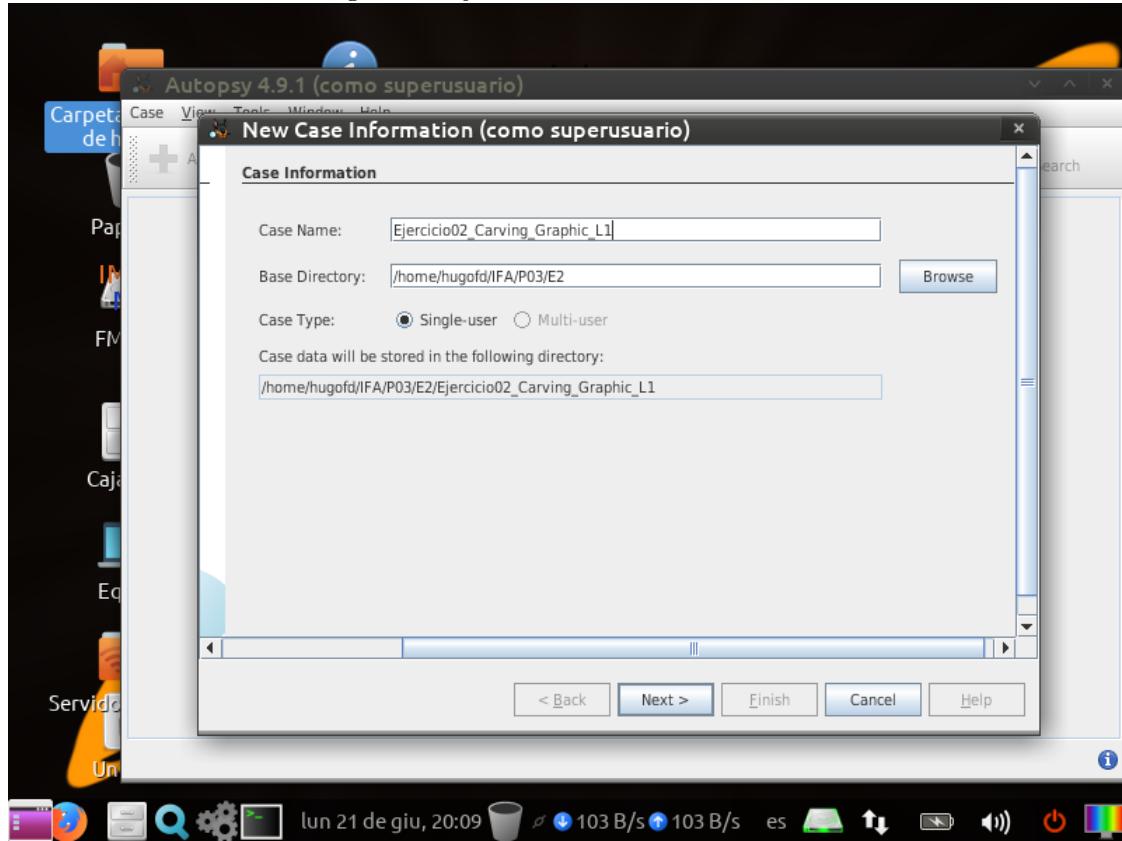
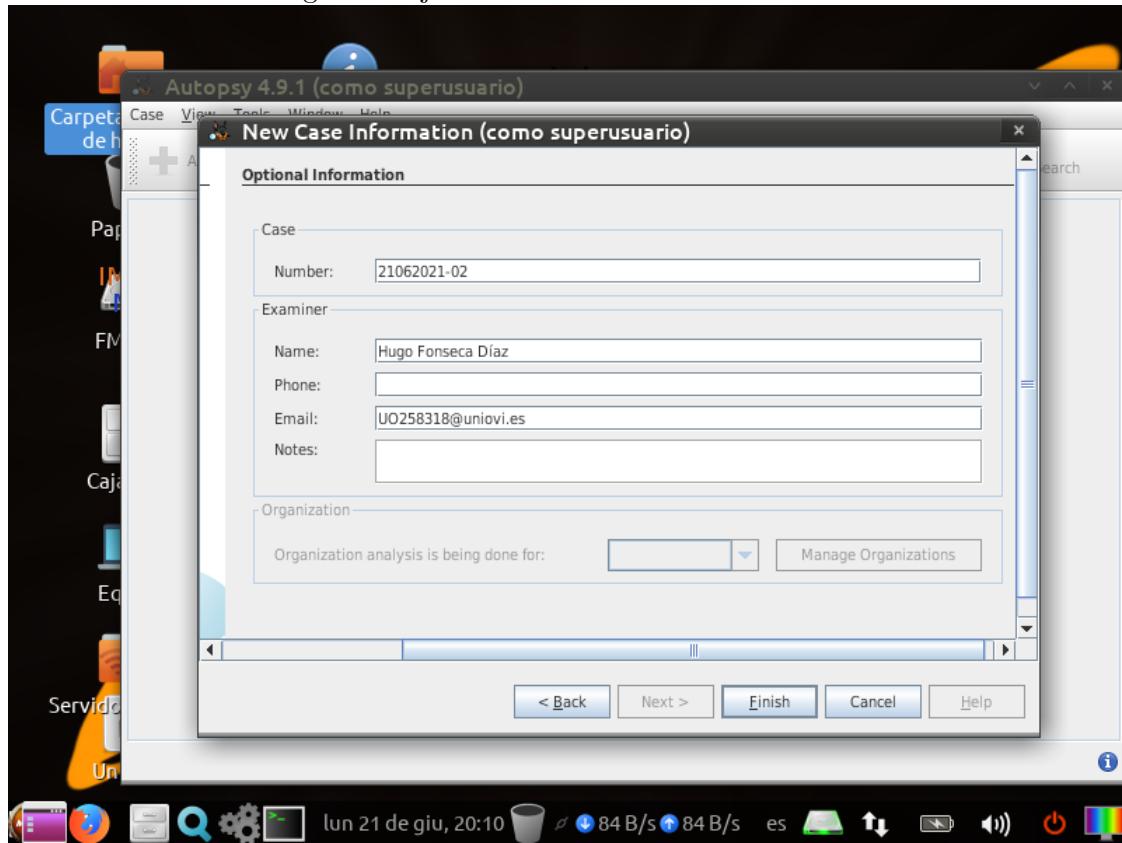
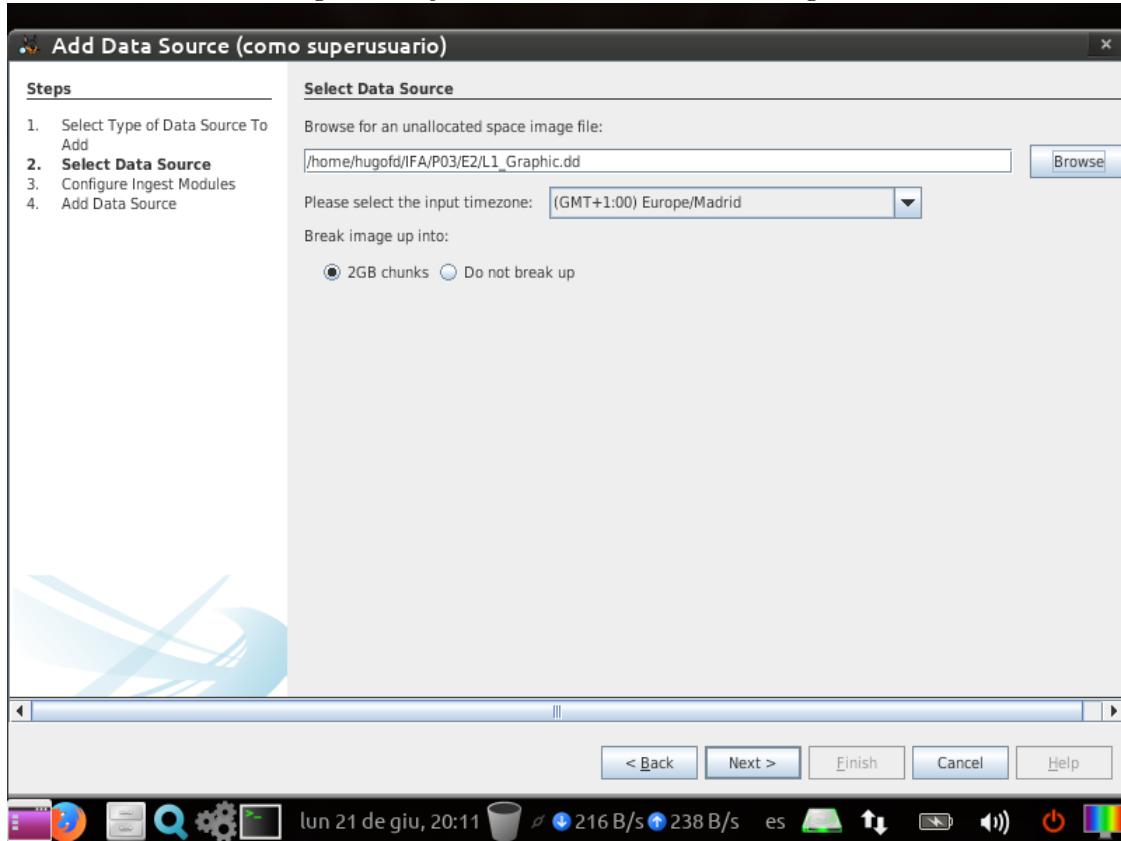


Figura 8: Ejercicio 2: Detalles del examinador



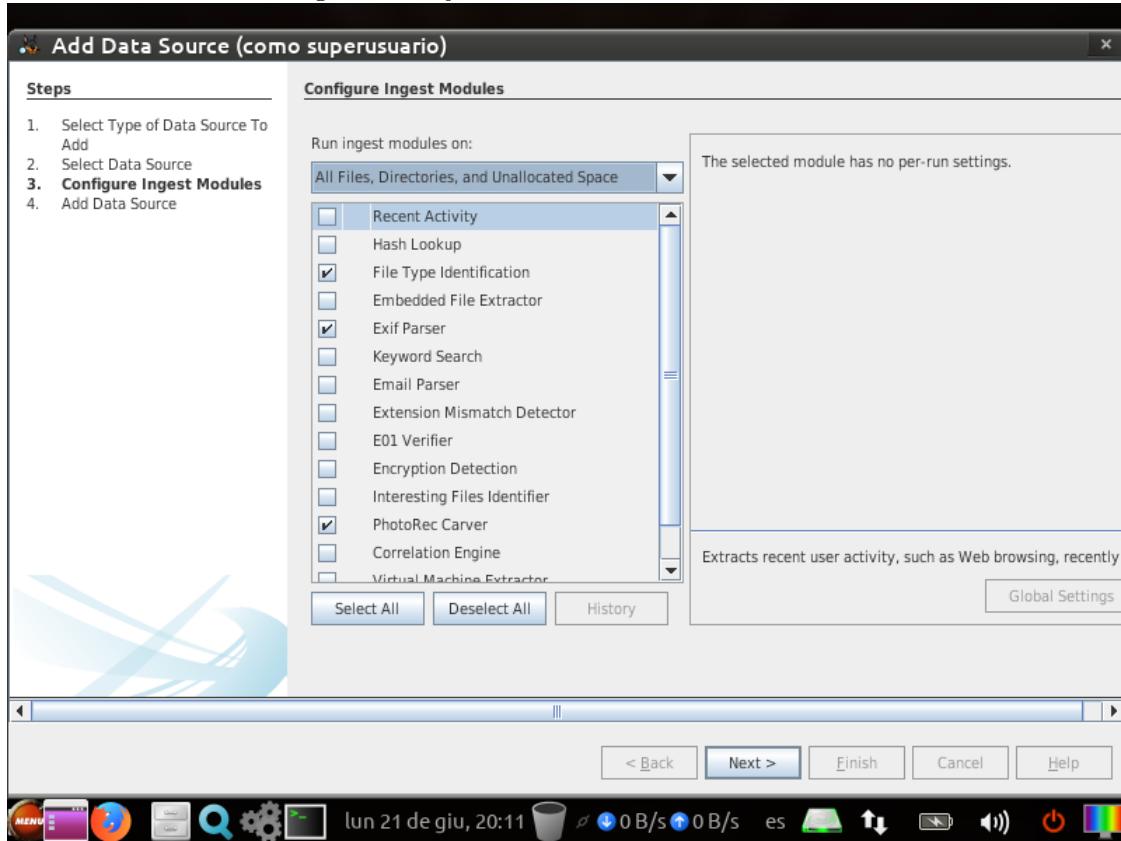
Añadimos la imagen a analizar.

Figura 9: Ejercicio 2: Selección de la imagen



Se seleccionan los módulos de identificación de tipos de fichero, parseador de Exif y *PhotoRec Carver*.

Figura 10: Ejercicio 2: Selección de módulos



Se ejecuta el análisis y se obtienen los resultados con los que se rellenará la tabla.

Figura 11: Ejercicio 2: Resultados del análisis

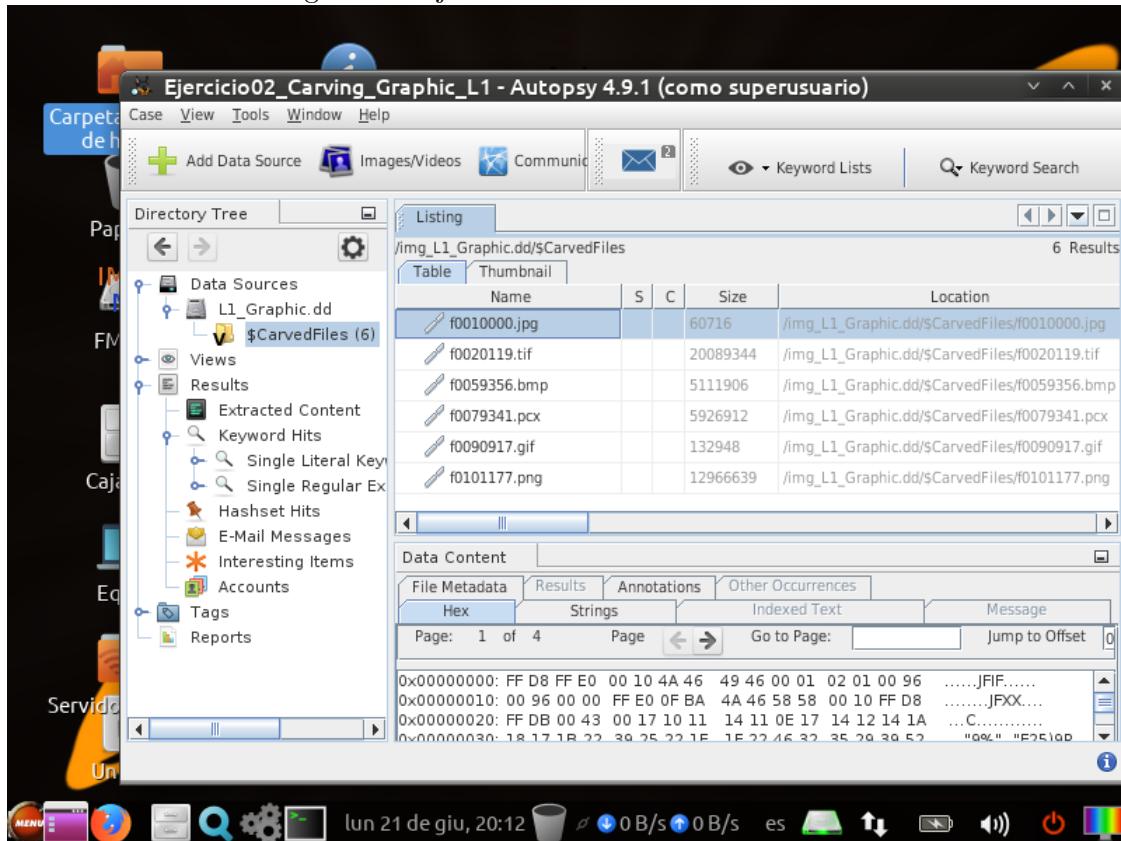
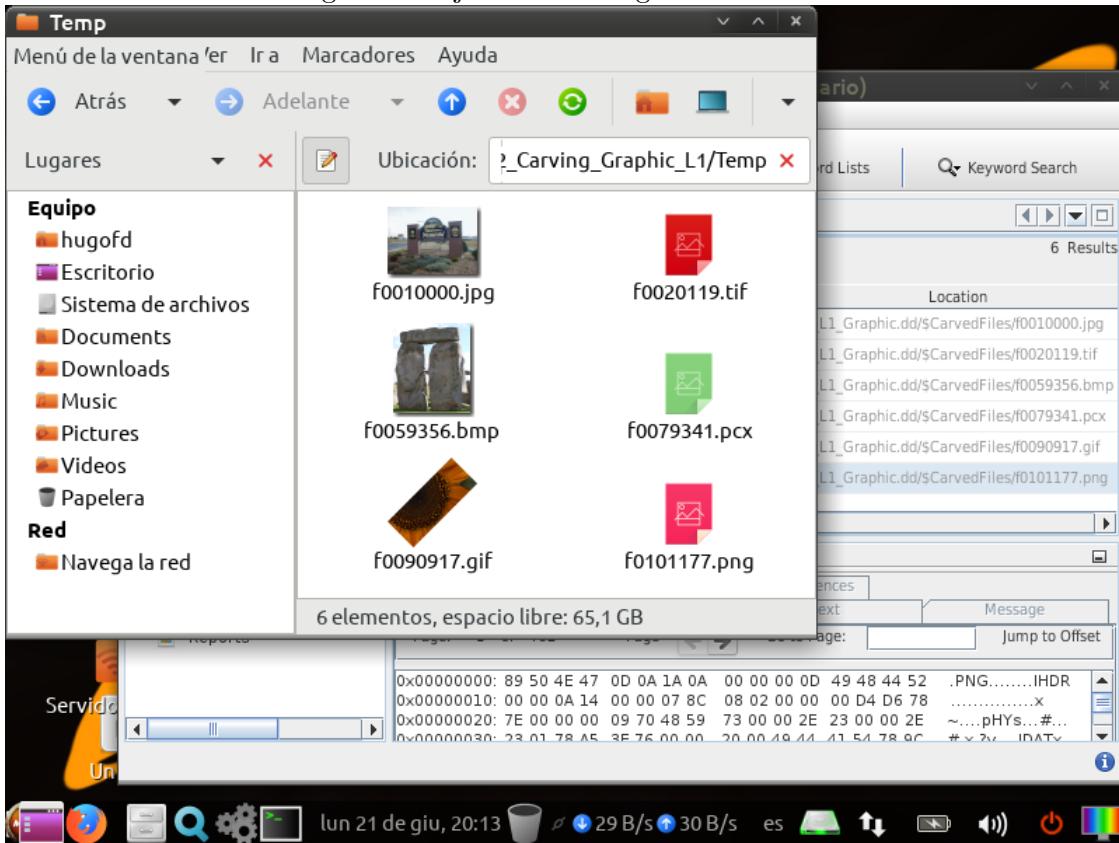


Figura 12: Ejercicio 2: Imágenes obtenidas



Para abrir el archivo con extensión *pcx* se ha utilizado un visor de imágenes online, al no disponer de uno adecuado en el equipo.

| Nombre del fichero en Autopsy | Tamaño del fichero (en Bytes) | Breve descripción imagen visible |
|-------------------------------|-------------------------------|----------------------------------|
| f0010000.jpg | 60716 | Cartel 'Welcome to Moscow' |
| f0020119.tif | 20089344 | Cañas de bambú |
| f0059356.bmp | 5111906 | Piedras en forma de Pi |
| f0079341.pcx | 5926912 | Iglesia y fuente |
| f0090917.gif | 132948 | Girasol |
| f0101177.png | 12966639 | Flor morada |

3. Ejercicio 3

Se crea el caso en Autopsy con los datos solicitados.

Figura 13: Ejercicio 3: Creación del caso

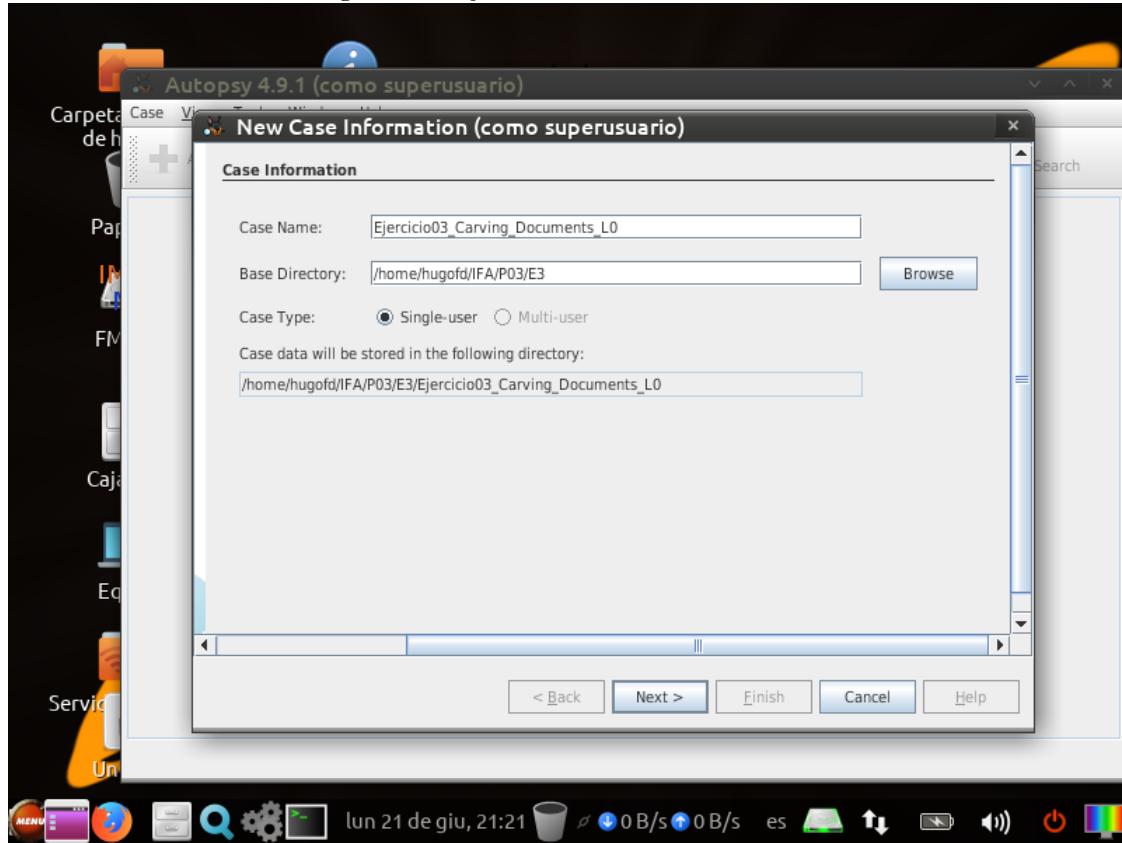
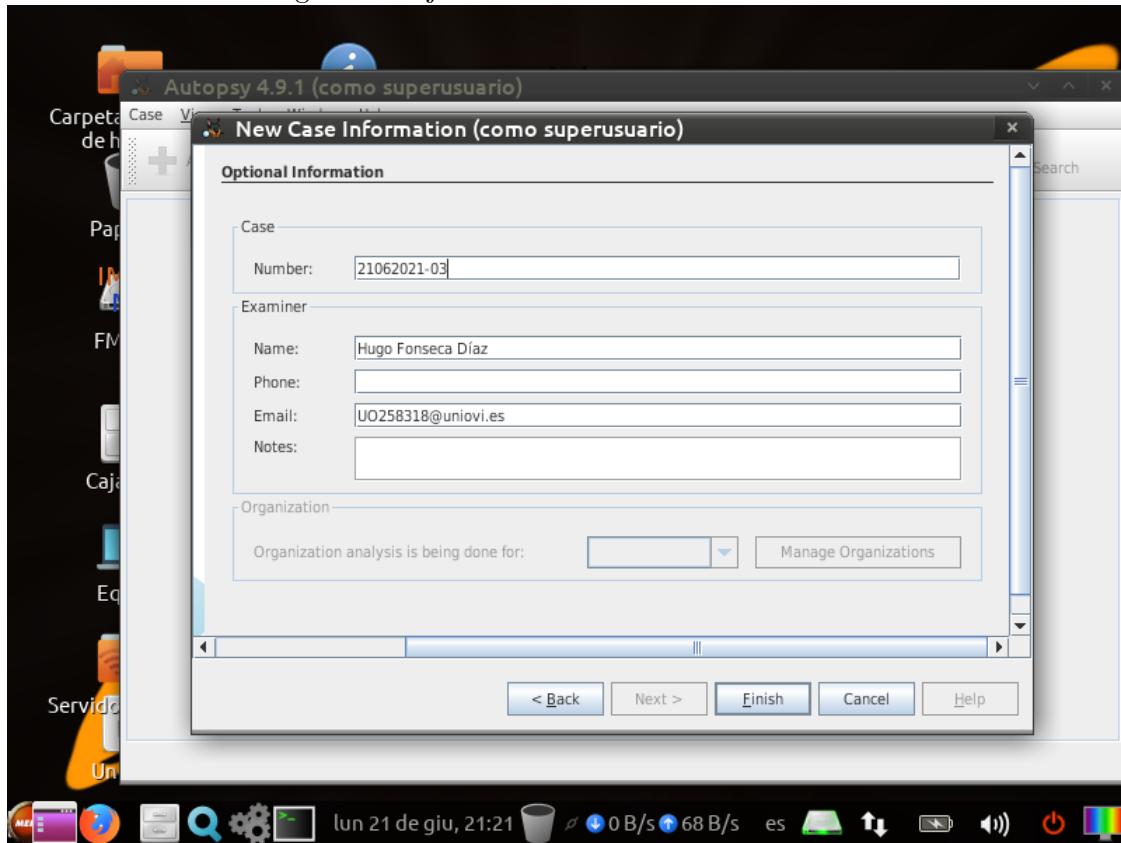
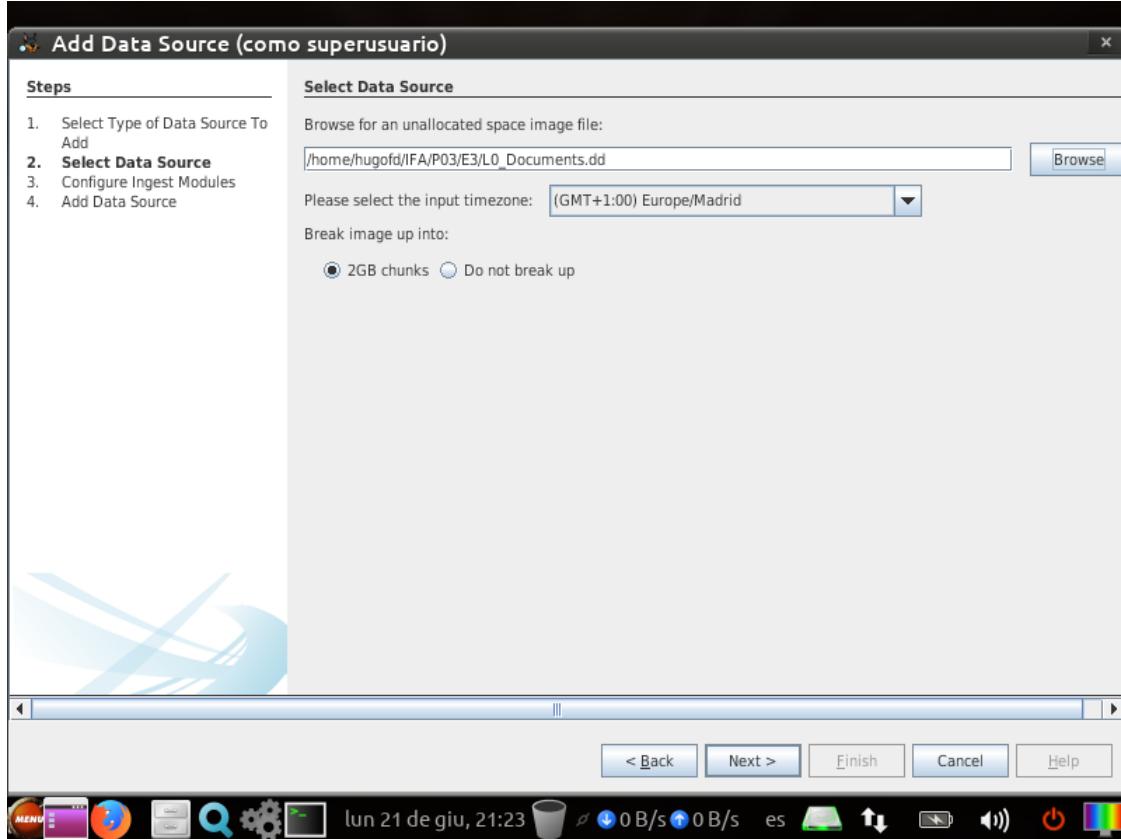


Figura 14: Ejercicio 3: Detalles del examinador



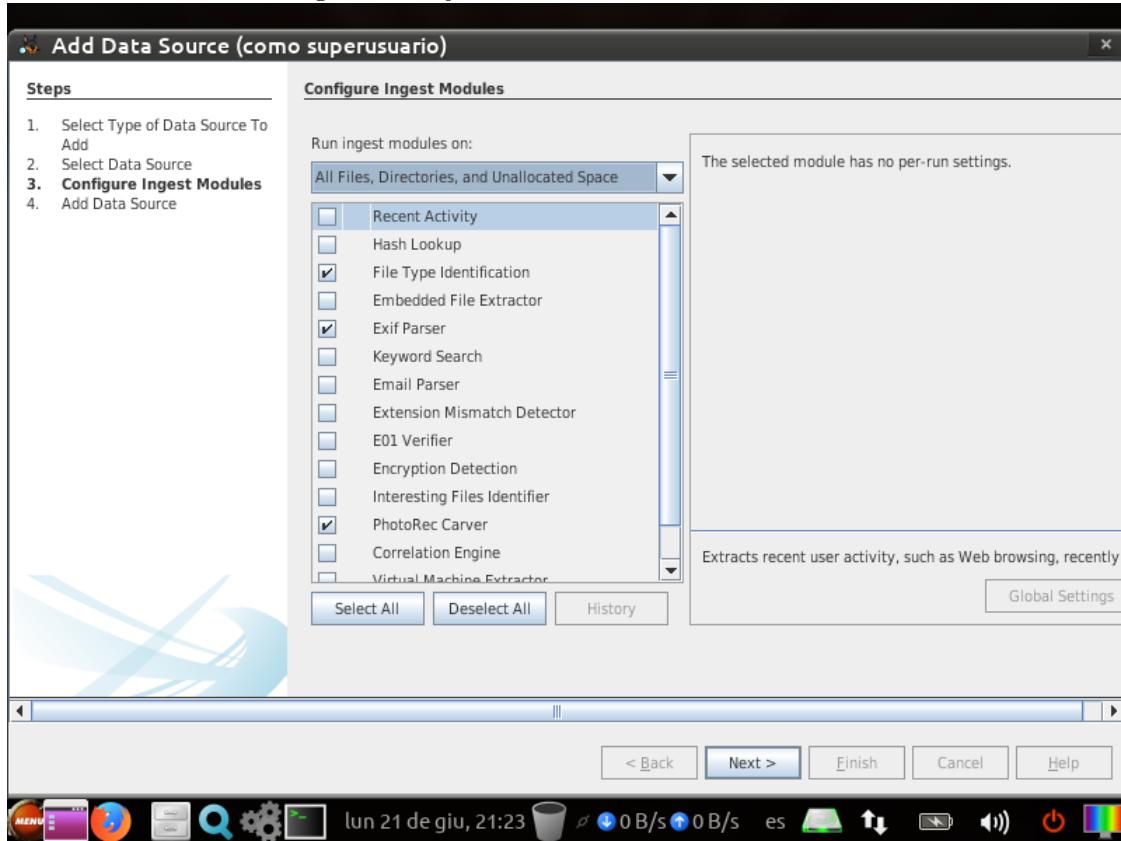
Añadimos la imagen a analizar.

Figura 15: Ejercicio 3: Selección de la imagen



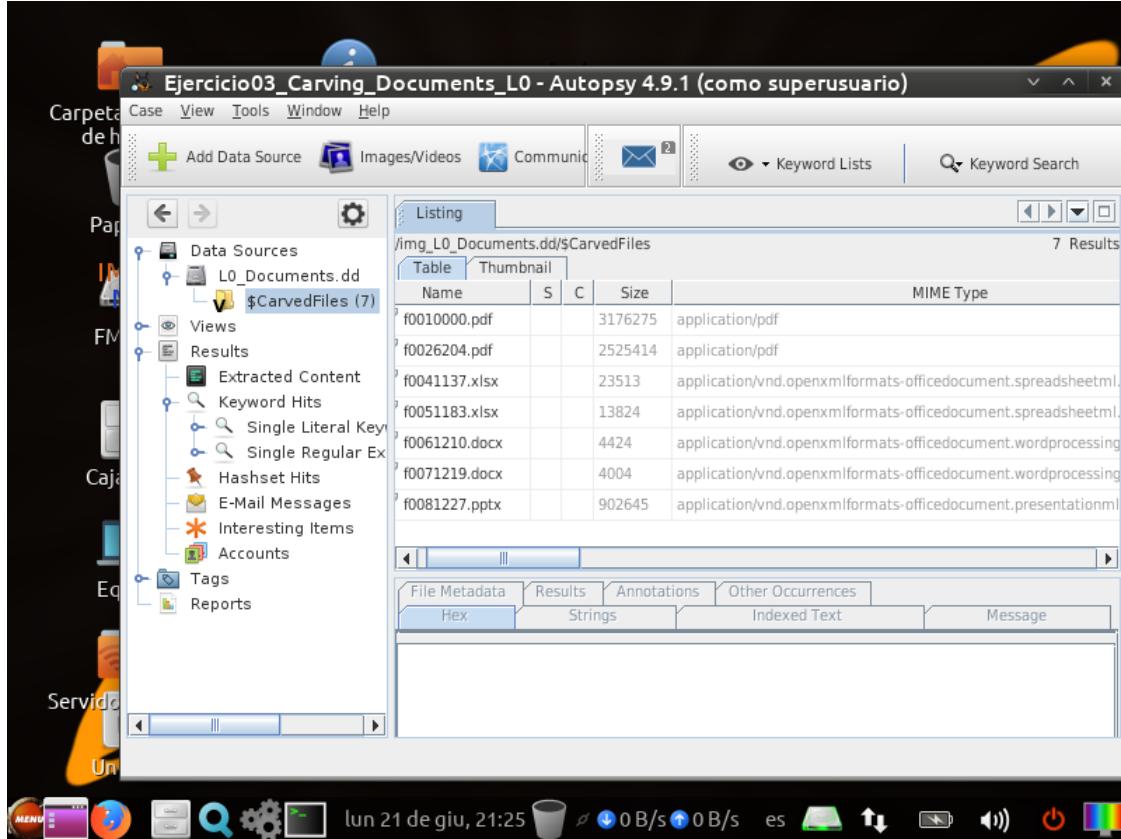
Se seleccionan los módulos de identificación de tipos de fichero, parseador de Exif y *PhotoRec Carver*.

Figura 16: Ejercicio 3: Selección de módulos



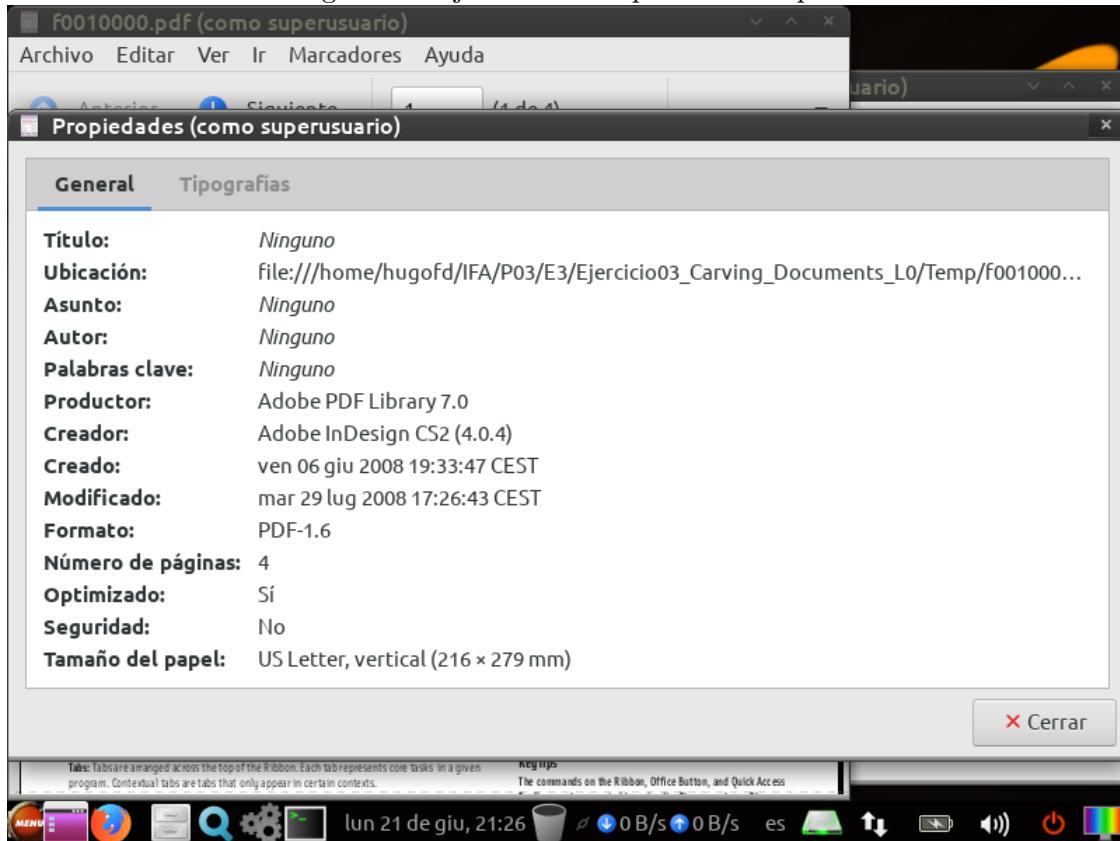
Se ejecuta el análisis y se obtienen los resultados con los que se rellenará la tabla.

Figura 17: Ejercicio 3: Resultados del análisis



Para obtener las fechas se abren los documentos con las aplicaciones externas correspondientes y se busca en sus propiedades.

Figura 18: Ejercicio 3: Propiedades del pdf



| Nombre del fichero en Autopsy | Tamaño del fichero (en Bytes) | Tipo MIME documento | Fecha Creación del documento |
|-------------------------------|-------------------------------|---|------------------------------|
| f0010000.pdf | 3176275 | application/pdf | 2008/06/06 |
| f0026204.pdf | 2525414 | application/pdf | 2008/06/04 |
| f0041137.xlsx | 23513 | application/vnd.openxmlformats-officedocument.spreadsheetml.sheet | 2012/06/13 |
| f0051183.xlsx | 13824 | application/vnd.openxmlformats-officedocument.spreadsheetml.sheet | 2012/07/05 |
| f0061210.docx | 4424 | application/vnd.openxmlformats-officedocument.wordprocessingml.document | Sin especificar |
| f0071219.docx | 4004 | application/vnd.openxmlformats-officedocument.wordprocessingml.document | Sin especificar |
| f0081227.pptx | 902645 | application/vnd.openxmlformats-officedocument.presentationml.presentation | 2010/09/28 |

4. Ejercicio 4

Se crea el caso en Autopsy con los datos solicitados.

Figura 19: Ejercicio 4: Creación del caso

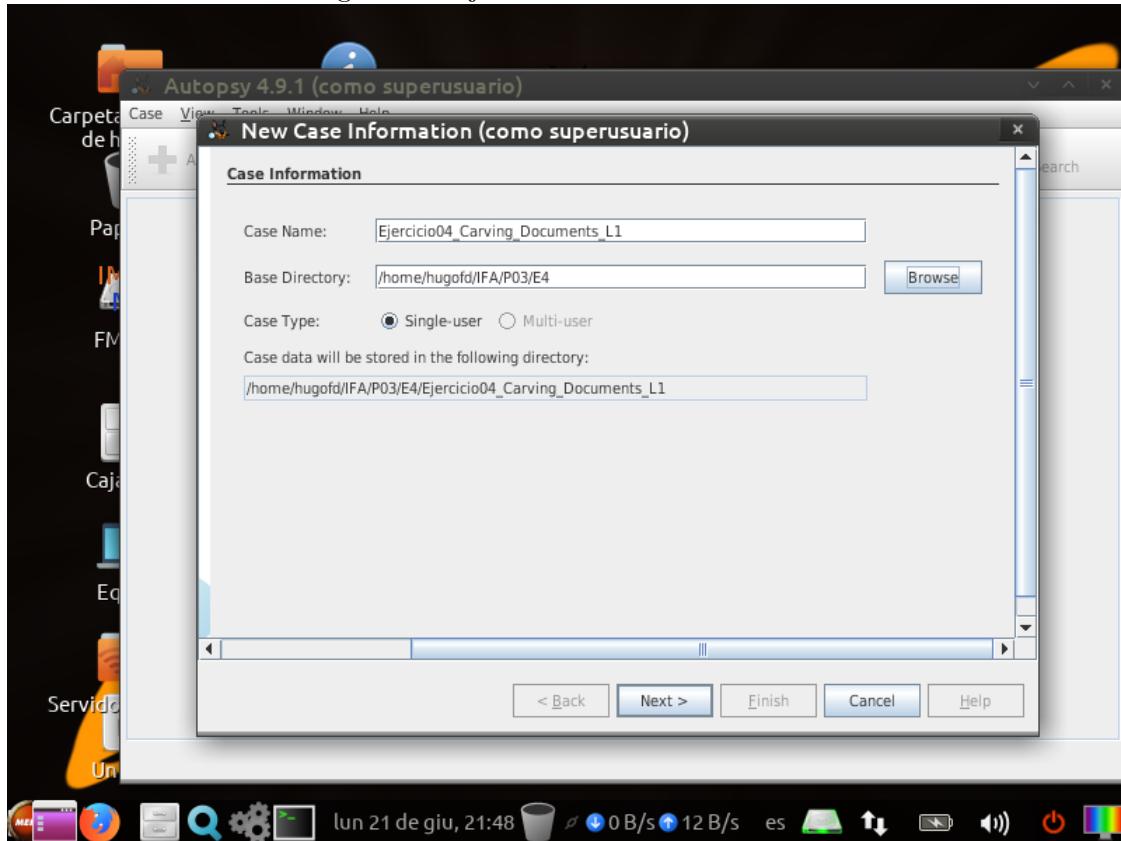
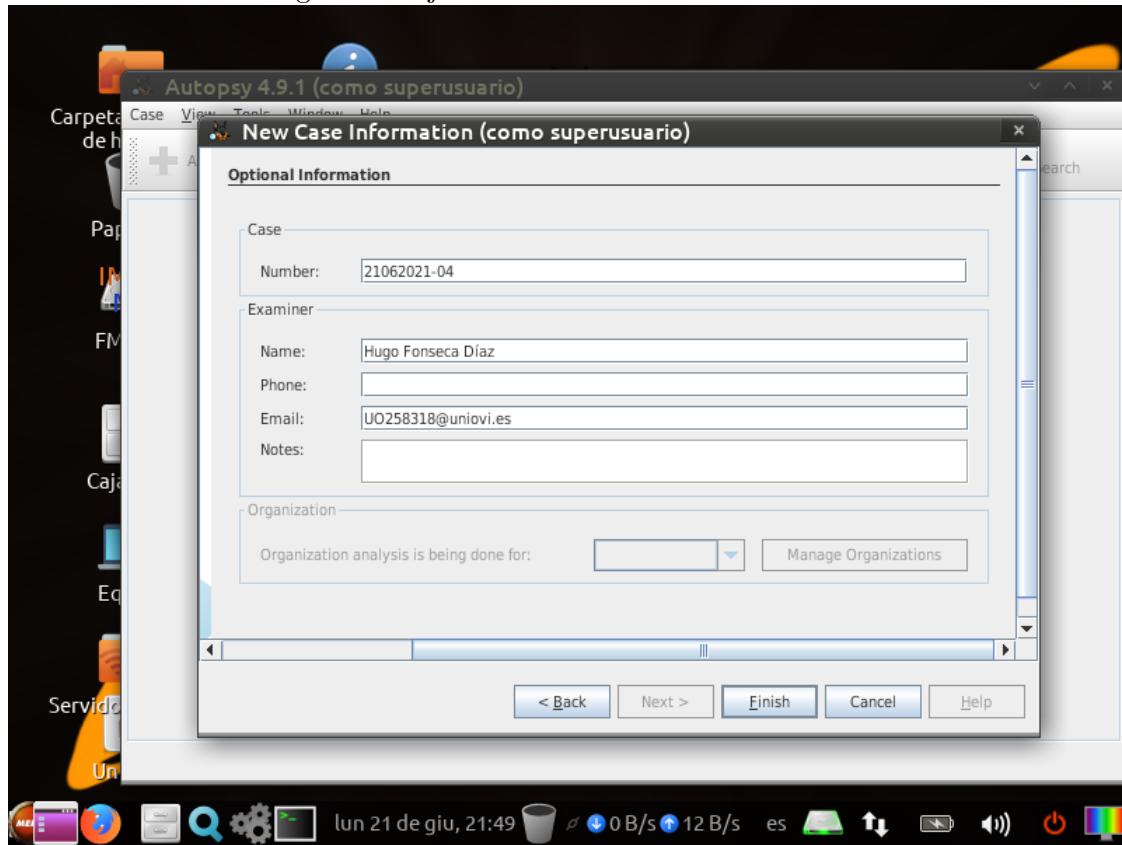
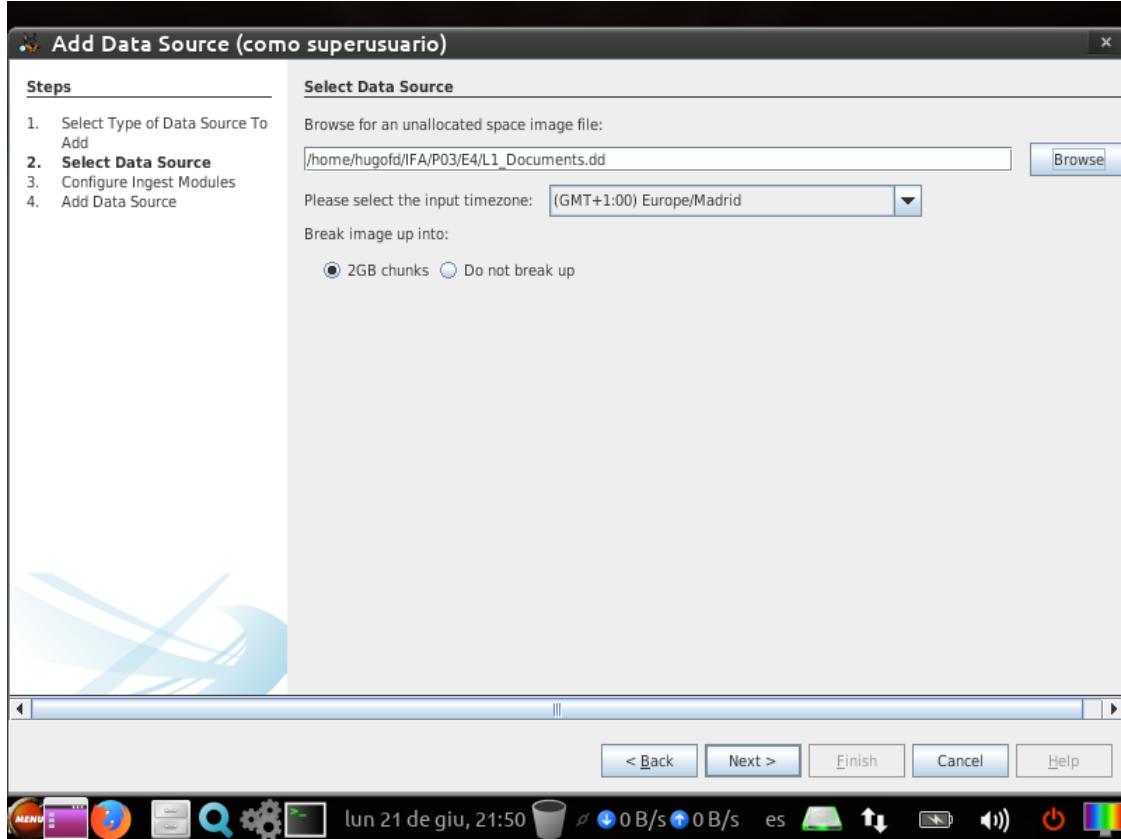


Figura 20: Ejercicio 4: Detalles del examinador



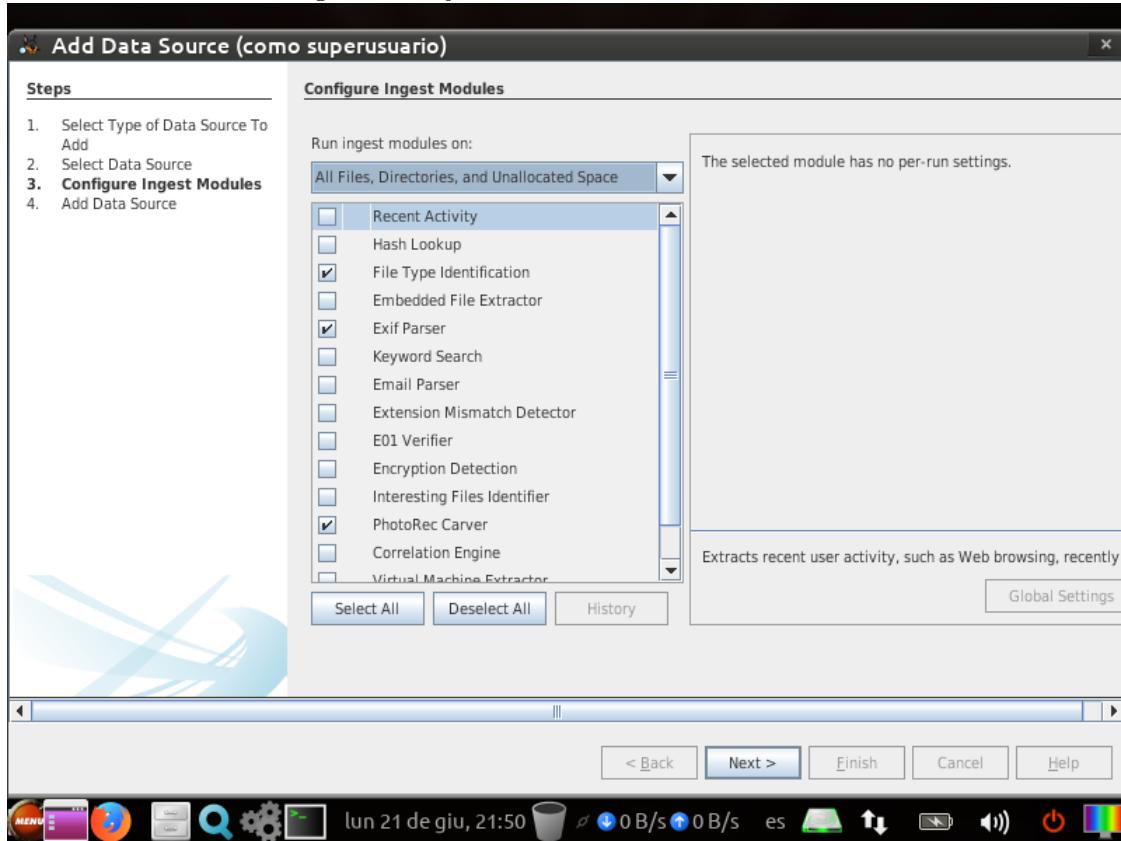
Añadimos la imagen a analizar.

Figura 21: Ejercicio 4: Selección de la imagen



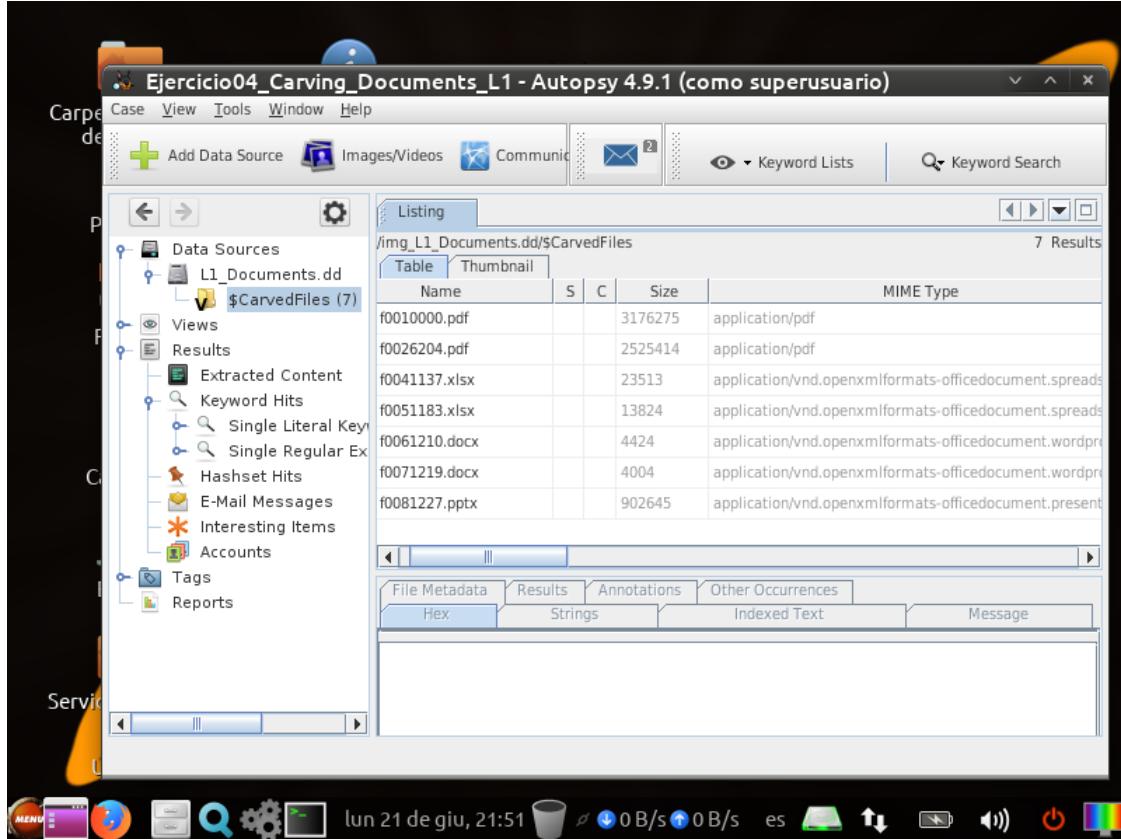
Se seleccionan los módulos de identificación de tipos de fichero, parseador de Exif y *PhotoRec Carver*.

Figura 22: Ejercicio 4: Selección de módulos



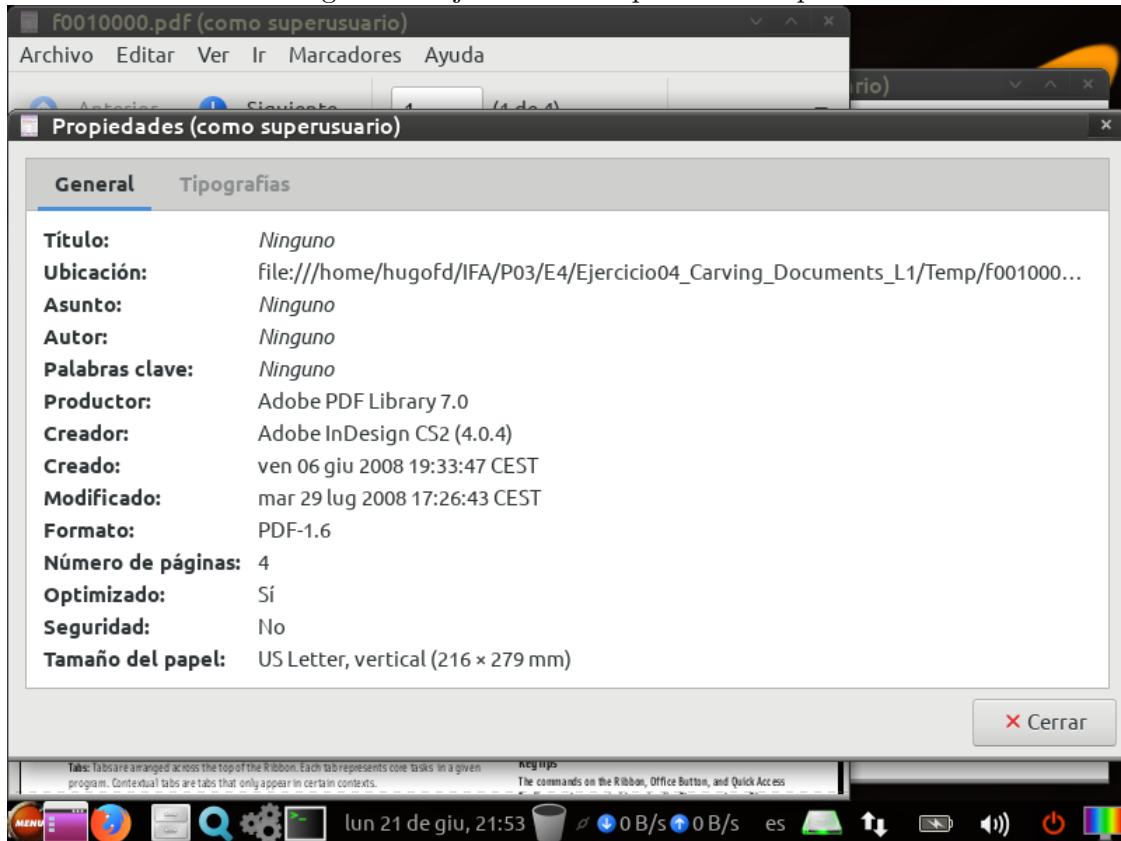
Se ejecuta el análisis y se obtienen los resultados con los que se rellenará la tabla.

Figura 23: Ejercicio 4: Resultados del análisis



Para obtener las fechas se abren los documentos con las aplicaciones externas correspondientes y se busca en sus propiedades.

Figura 24: Ejercicio 4: Propiedades del pdf



| Nombre del fichero en Autopsy | Tamaño del fichero (en Bytes) | Tipo MIME documento | Fecha Creación del documento |
|-------------------------------|-------------------------------|---|------------------------------|
| f0010000.pdf | 3176275 | application/pdf | 2008/06/06 |
| f0026204.pdf | 2525414 | application/pdf | 2008/06/04 |
| f0041137.xlsx | 23513 | application/vnd.openxmlformats-officedocument.spreadsheetml.sheet | 2012/06/13 |
| f0051183.xlsx | 13824 | application/vnd.openxmlformats-officedocument.spreadsheetml.sheet | 2012/07/05 |
| f0061210.docx | 4424 | application/vnd.openxmlformats-officedocument.wordprocessingml.document | Sin especificar |
| f0071219.docx | 4004 | application/vnd.openxmlformats-officedocument.wordprocessingml.document | Sin especificar |
| f0081227.pptx | 902645 | application/vnd.openxmlformats-officedocument.presentationml.presentation | 2010/09/28 |

5. Ejercicio 5

Se crea el caso en Autopsy con los datos solicitados.

Figura 25: Ejercicio 5: Creación del caso

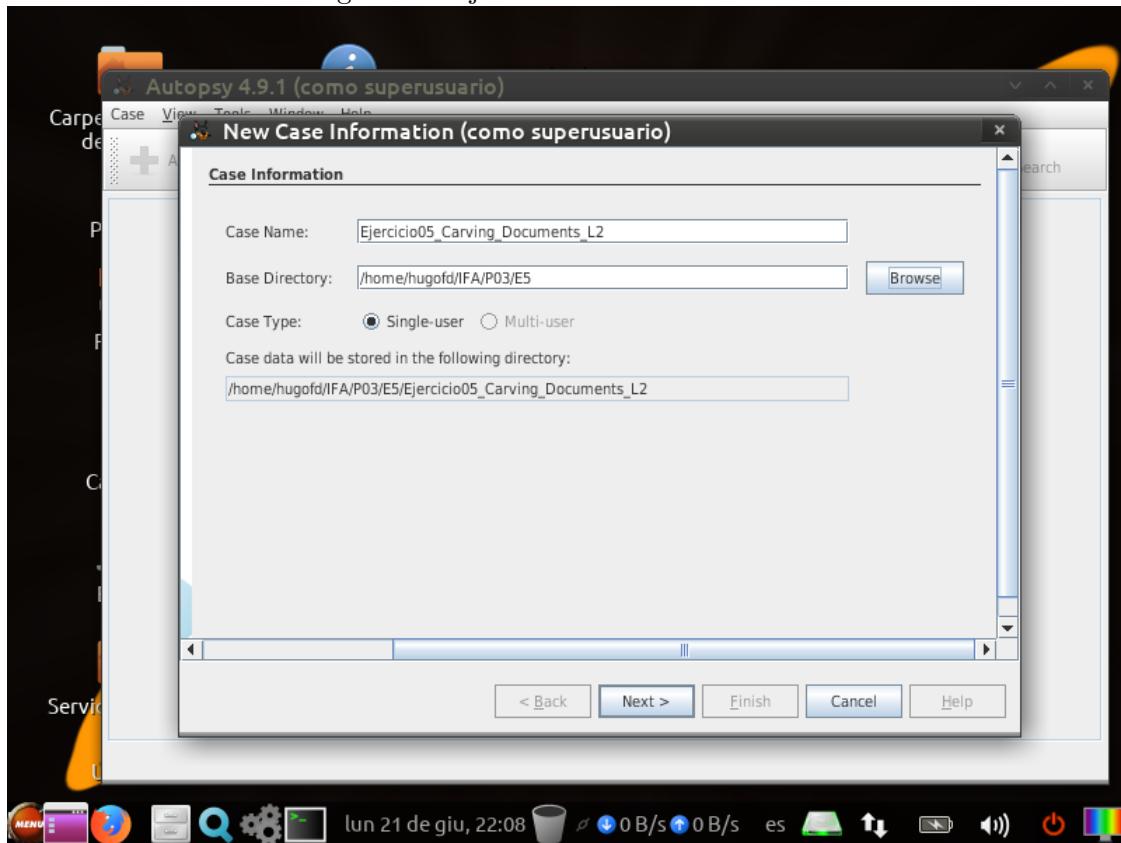
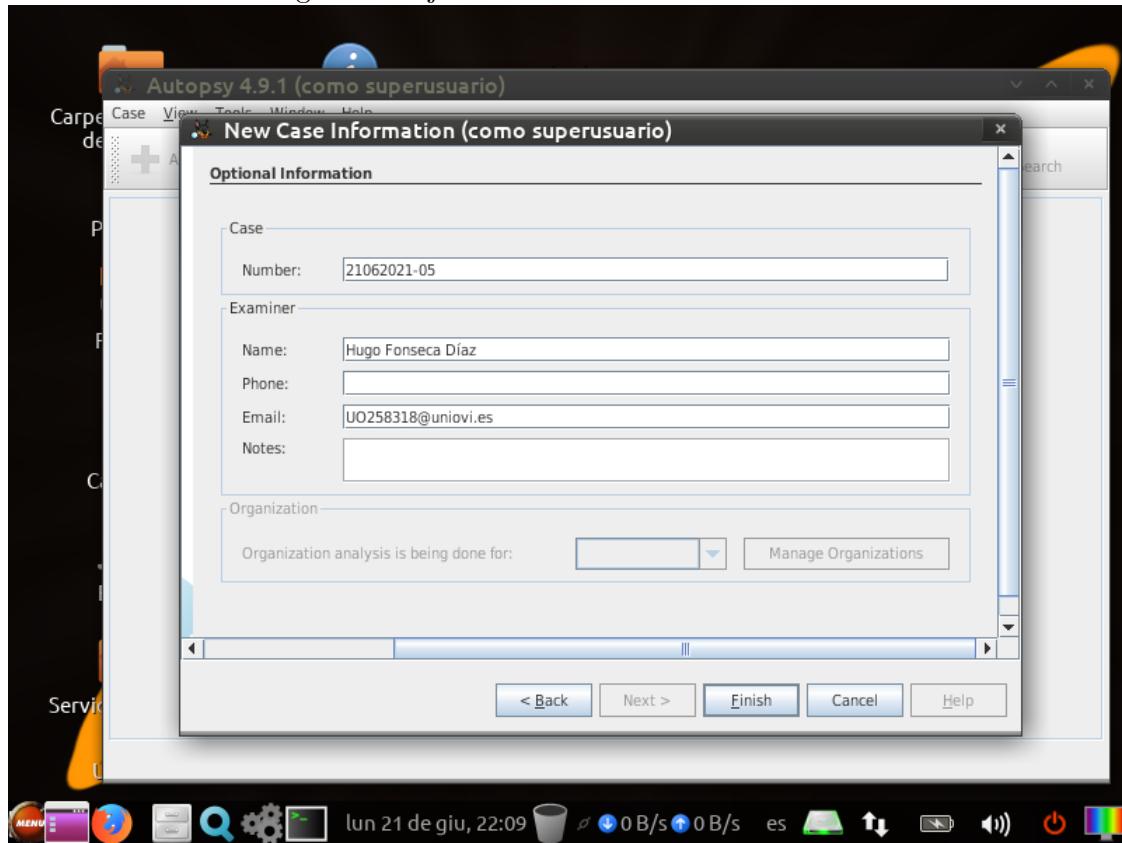
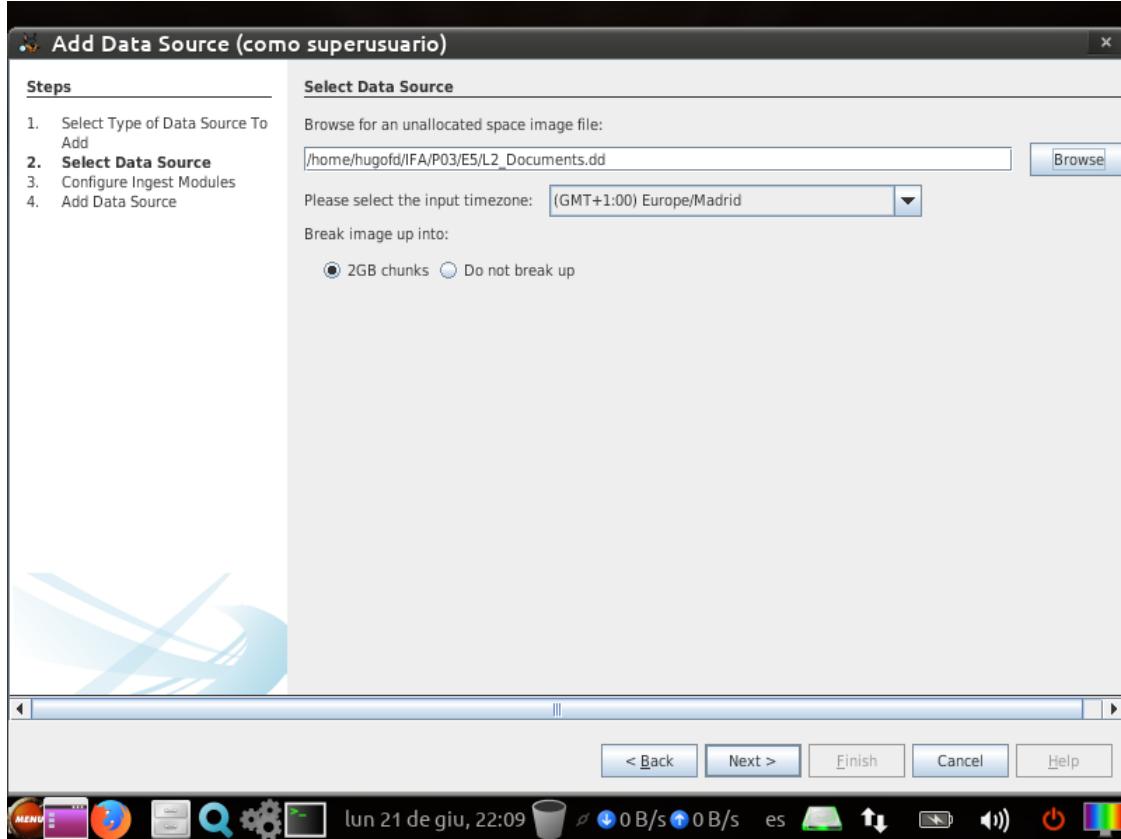


Figura 26: Ejercicio 5: Detalles del examinador



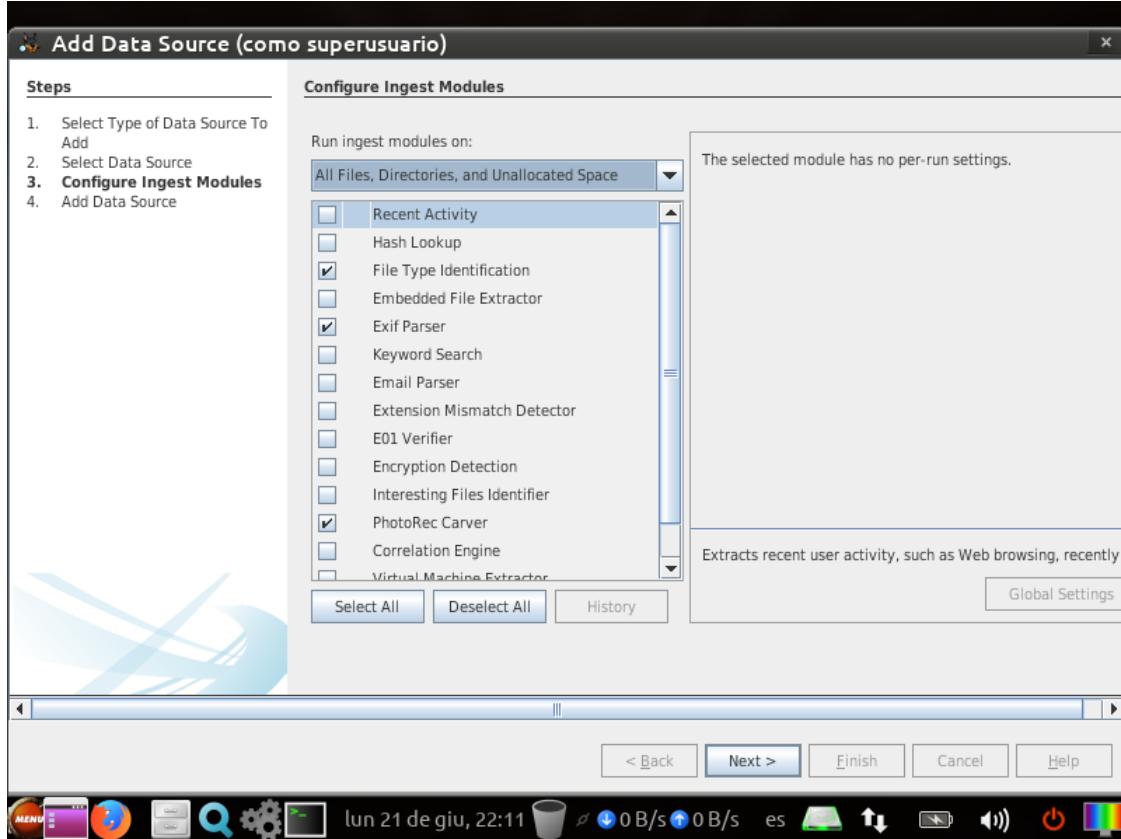
Añadimos la imagen a analizar.

Figura 27: Ejercicio 5: Selección de la imagen



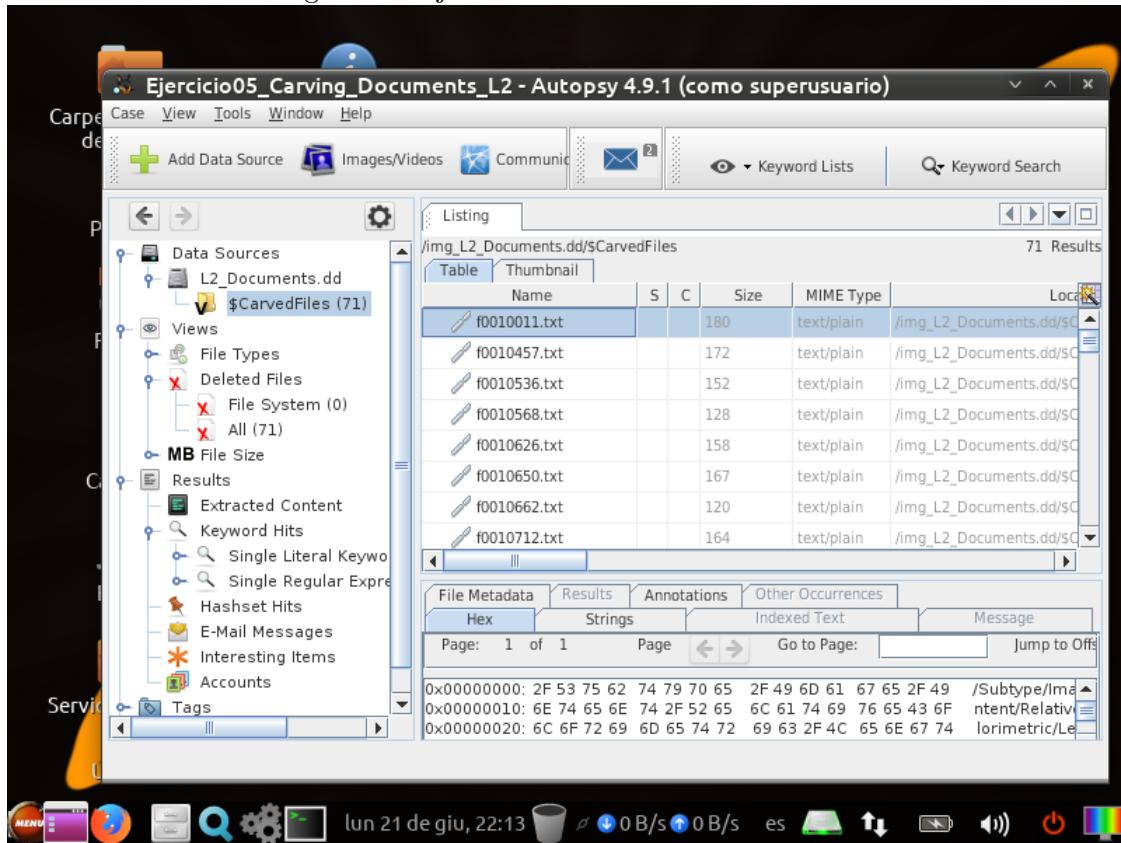
Se seleccionan los módulos de identificación de tipos de fichero, parseador de Exif y *PhotoRec Carver*.

Figura 28: Ejercicio 5: Selección de módulos



Se ejecuta el análisis y se obtienen los resultados con los que se responderá a las preguntas.

Figura 29: Ejercicio 5: Resultados del análisis



- a) Hay 71 falsos positivos.
- b) Todos son de tipo texto plano.

Esto puede deberse a que Autopsy no haya sido capaz de recuperar los archivos con sus verdaderos tipos MIME y los fragmentos de esos archivos sean tratados como texto plano.

6. Ejercicio 6

Se crea el caso en Autopsy con los datos solicitados.

Figura 30: Ejercicio 6: Creación del caso

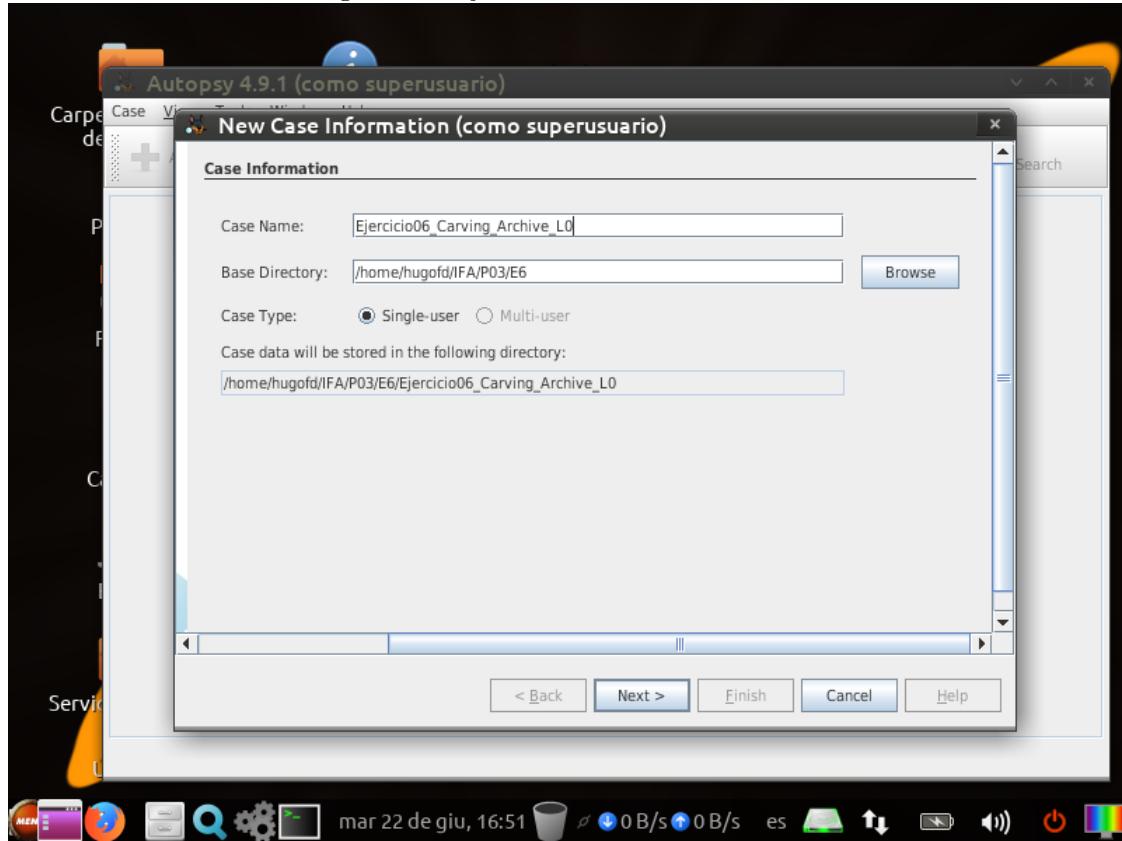
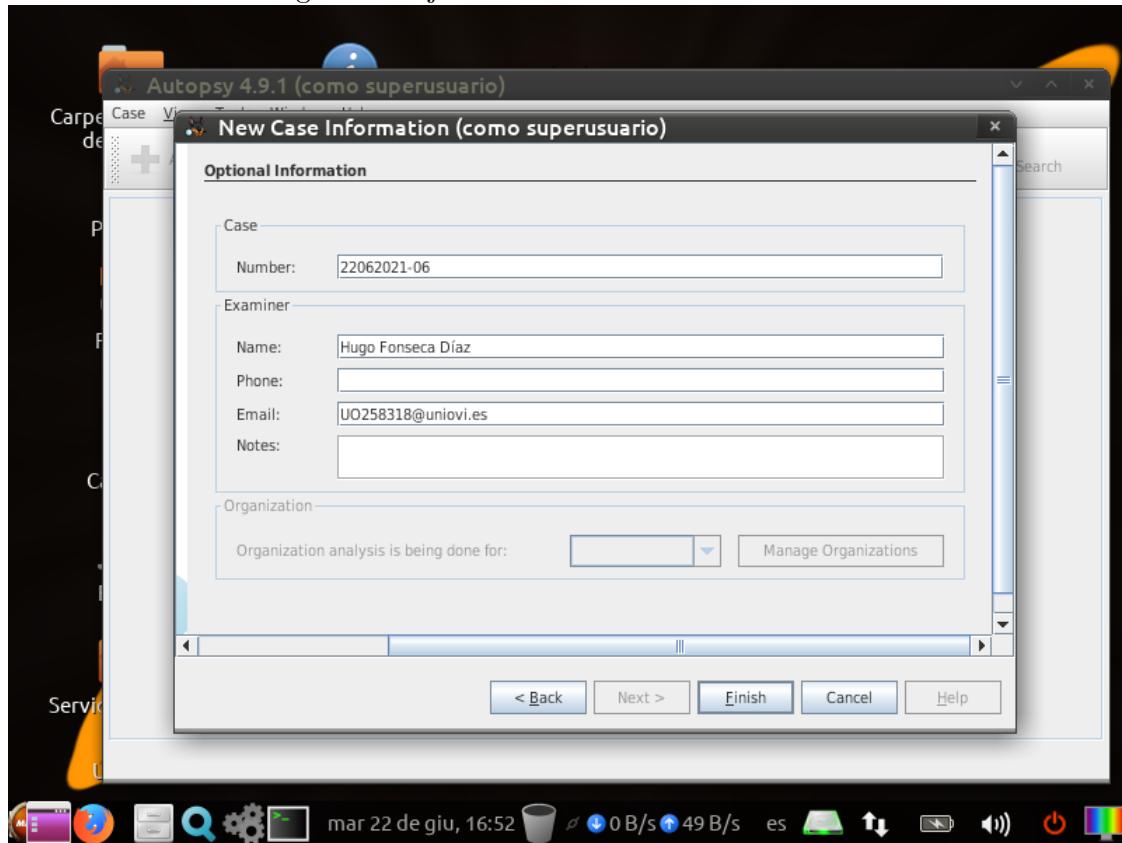
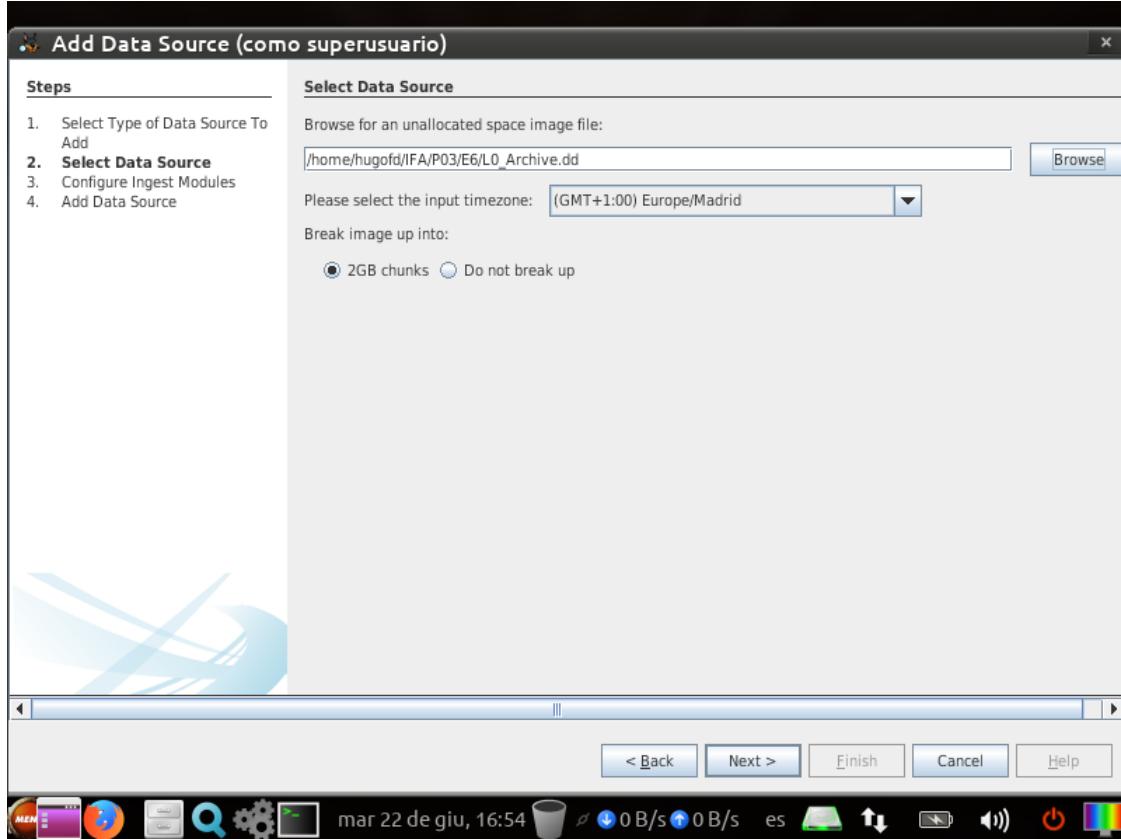


Figura 31: Ejercicio 6: Detalles del examinador



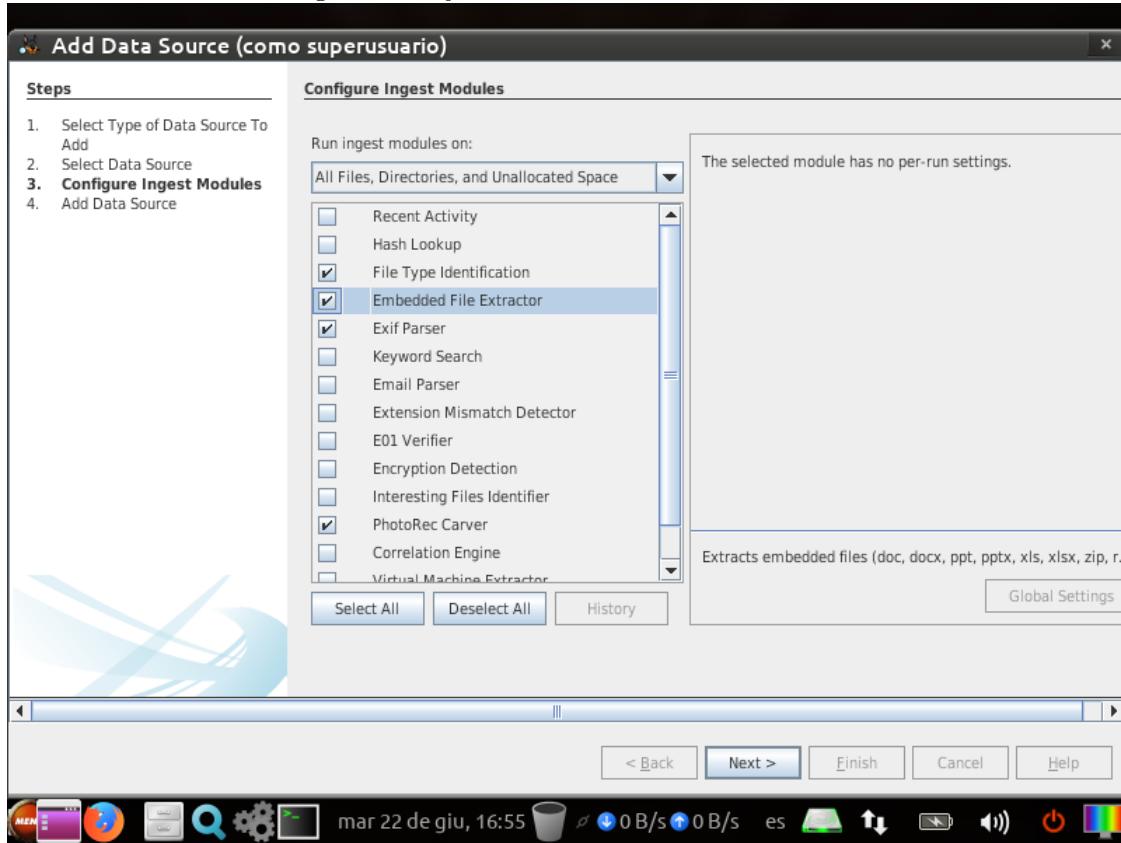
Añadimos la imagen a analizar.

Figura 32: Ejercicio 6: Selección de la imagen



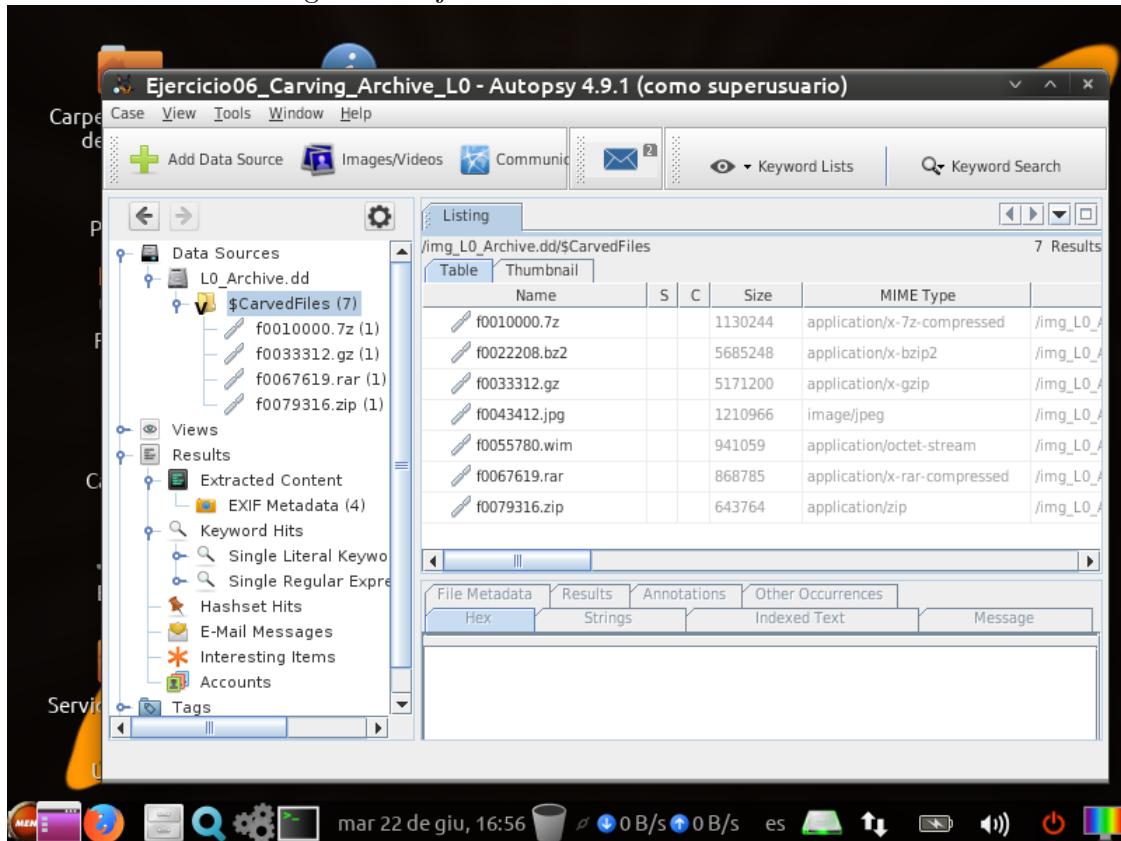
Se seleccionan los módulos de identificación de tipos de fichero, parseador de Exif, *PhotoRec Carver* y el módulo de extracción de ficheros.

Figura 33: Ejercicio 6: Selección de módulos



Se ejecuta el análisis y se obtienen los resultados con los que se rellenará la tabla.

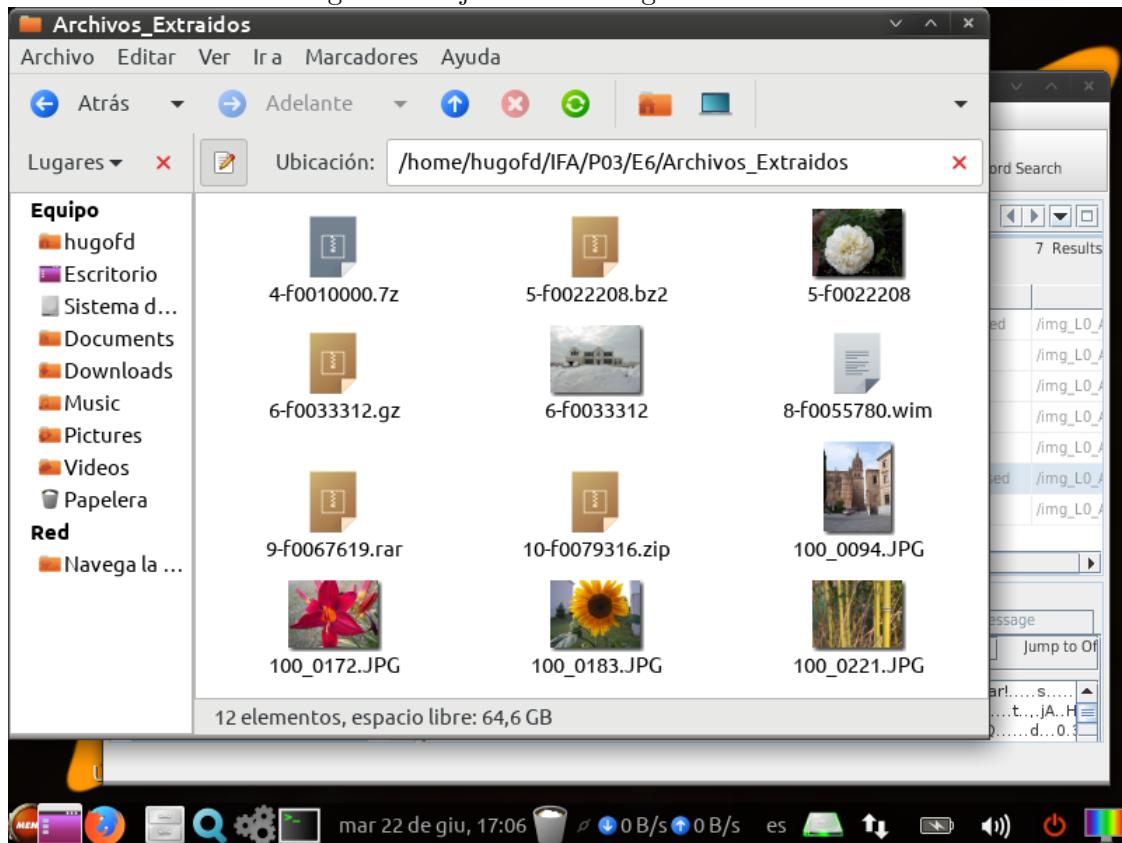
Figura 34: Ejercicio 6: Resultados del análisis



| Nombre del fichero en Autopsy | Tamaño del fichero (en Bytes) | Tipo MIME |
|-------------------------------|-------------------------------|------------------------------|
| f0010000.7z | 1130244 | application/x-7z-compressed |
| f0022208.bz2 | 5685248 | application/x-bzip2 |
| f0033312.gz | 5171200 | application/x-gzip |
| f0055780.wim | 941059 | application/octet-stream |
| f0067619.rar | 868785 | application/x-rar-compressed |
| f0079316.zip | 643764 | application/zip |

Se extraen las imágenes de los ficheros comprimidos.

Figura 35: Ejercicio 6: Imágenes extraídas



Se ejecuta la herramienta *exiftool* para obtener los datos que se usan a la hora de rellenar la siguiente tabla.

Figura 36: Ejercicio 6: Resultado del comando *exiftool* con una de las imágenes extraídas

```

hugofd@IFA-AU-HugoFonsecaDiaz: ~/IFA/P03/E6/Archivos_Extraidos
Archivo Editar Ver Buscar Terminal Ayuda
Flash : Off
Resolution : 1
Protect : 0
Cont Take : 0
Color Mode : 1
F Number : 8.0
Zoom : x1.0
Macro : Off
Light S : 0
Exposure Compensation : 0
Camera Type : SR86
Serial Number : #00000001
Version : v86-77U
ID : EPSON DIGITAL STILL CAMERA
Pic Len : 46681
Thm Len : 2826
Tag Q : 86
Tag R : 289
Tag B : 312
S0 : 10e,0,3d2,6dac,39a,2d06,43b2,2010000,
d970003,64f0490,0,0,66,11010000,2d432407,0,0,0,190c0000,2a021d81
:

```

| Nombre del fichero en Autopsy | Fecha y hora de la imagen | Dispositivo con el que se tomó la imagen | Breve descripción de la imagen |
|-------------------------------|---------------------------|--|--------------------------------|
| 6-f0033312 | 2003/02/18 10:46:51 | SR86 EPSON DIGITAL STILL CAMERA | Casa con nieve |
| 5-f0022208 | 2004/05/31 15:03:51 | KODAK DX4530 ZOOM DIGITAL CAMERA | Flor blanca |
| 100_0094.jpg | 2004/06/19 04:52:06 | KODAK DX4530 ZOOM DIGITAL CAMERA | Iglesia / Catedral |
| 100_0172.jpg | 2004/07/02 19:42:41 | KODAK DX4530 ZOOM DIGITAL CAMERA | Flor rojiza |
| 100_0183.jpg | 2004/07/05 19:57:36 | KODAK DX4530 ZOOM DIGITAL CAMERA | Girasol |
| 100_0221.jpg | 2004/08/28 07:32:22 | KODAK DX4530 ZOOM DIGITAL CAMERA | Cañas bambú |

7. Ejercicio 7

Se crea el caso en Autopsy con los datos solicitados.

Figura 37: Ejercicio 7: Creación del caso

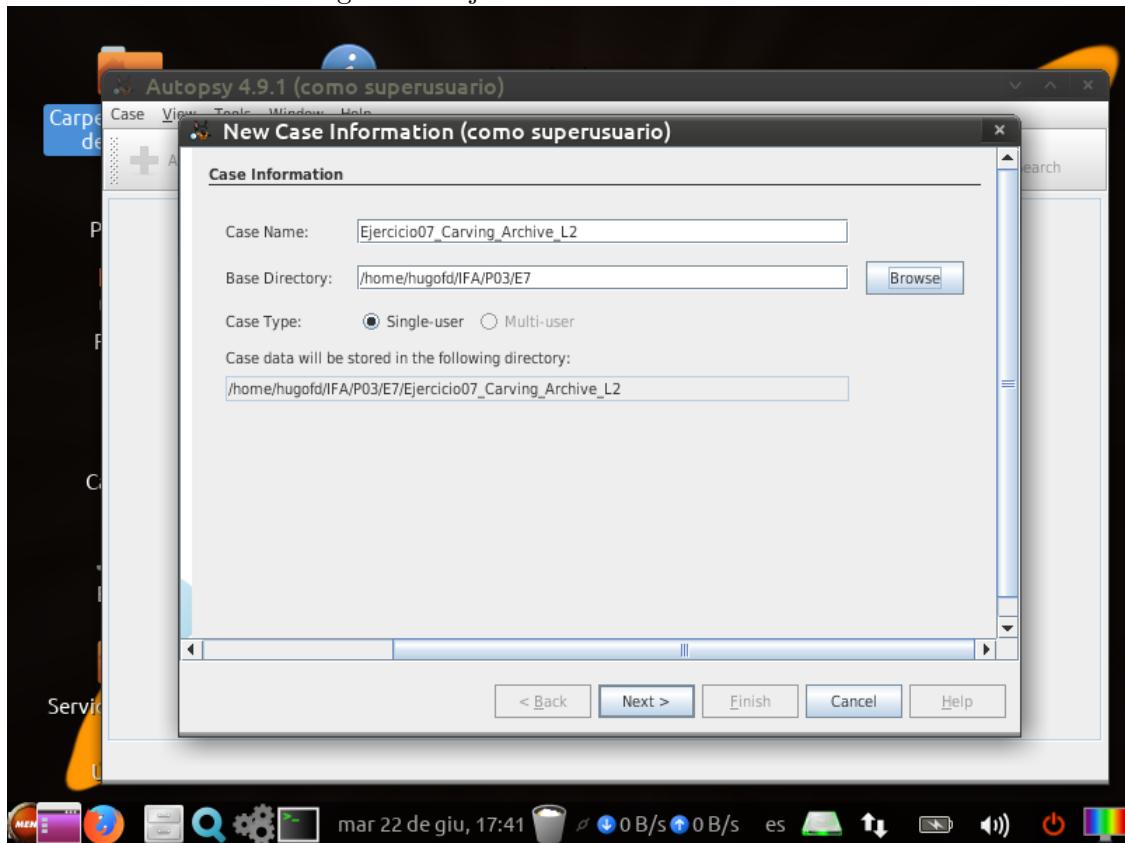
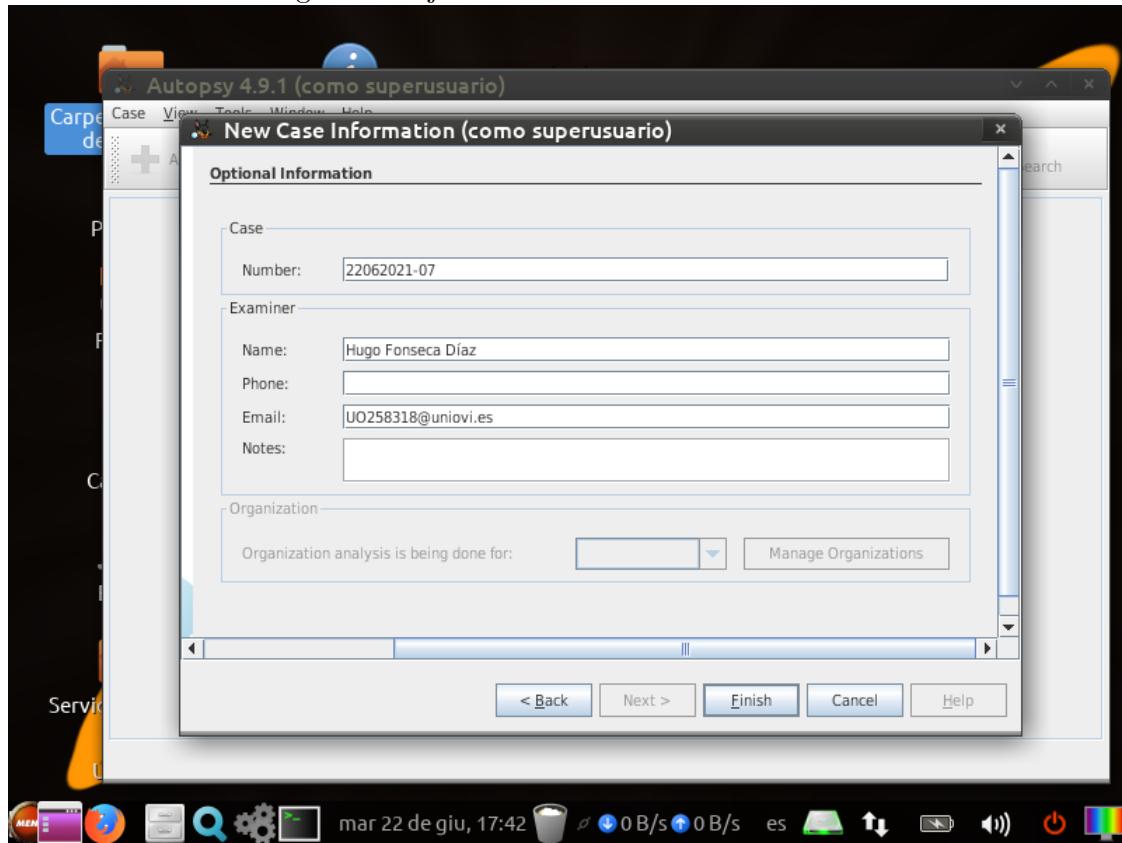
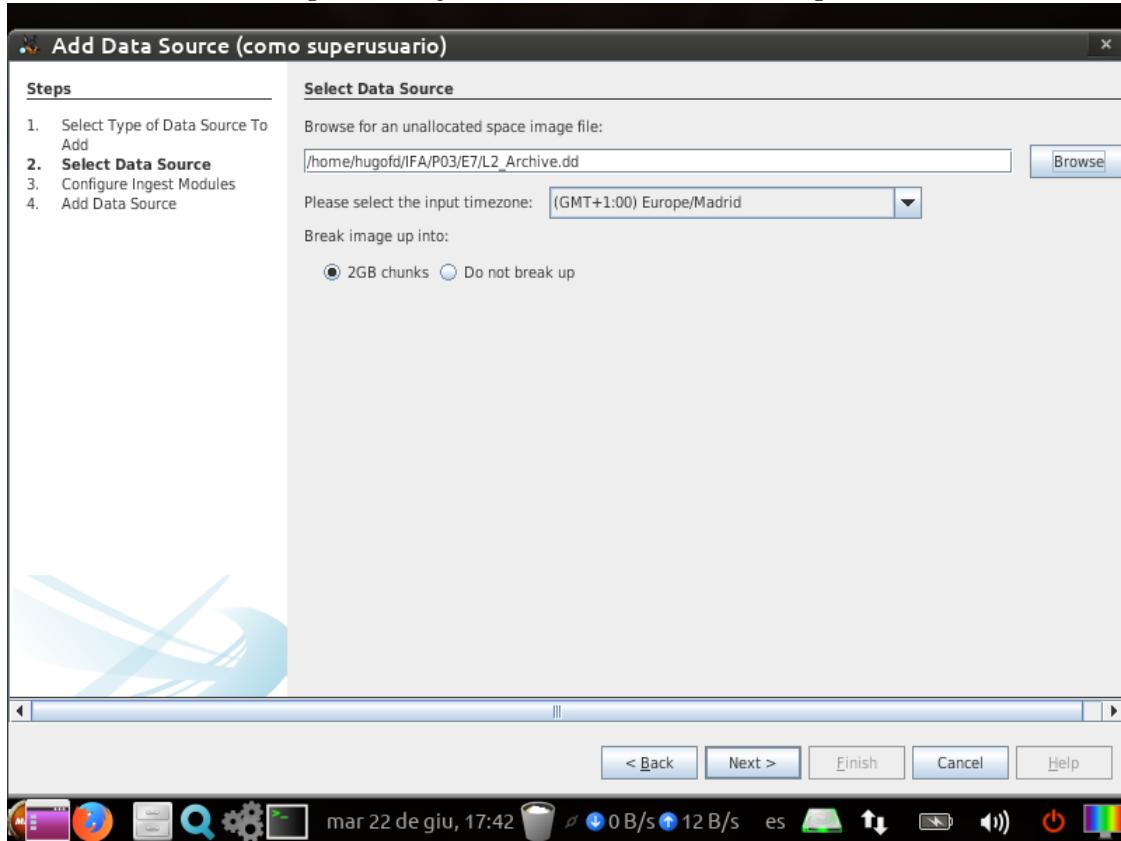


Figura 38: Ejercicio 7: Detalles del examinador



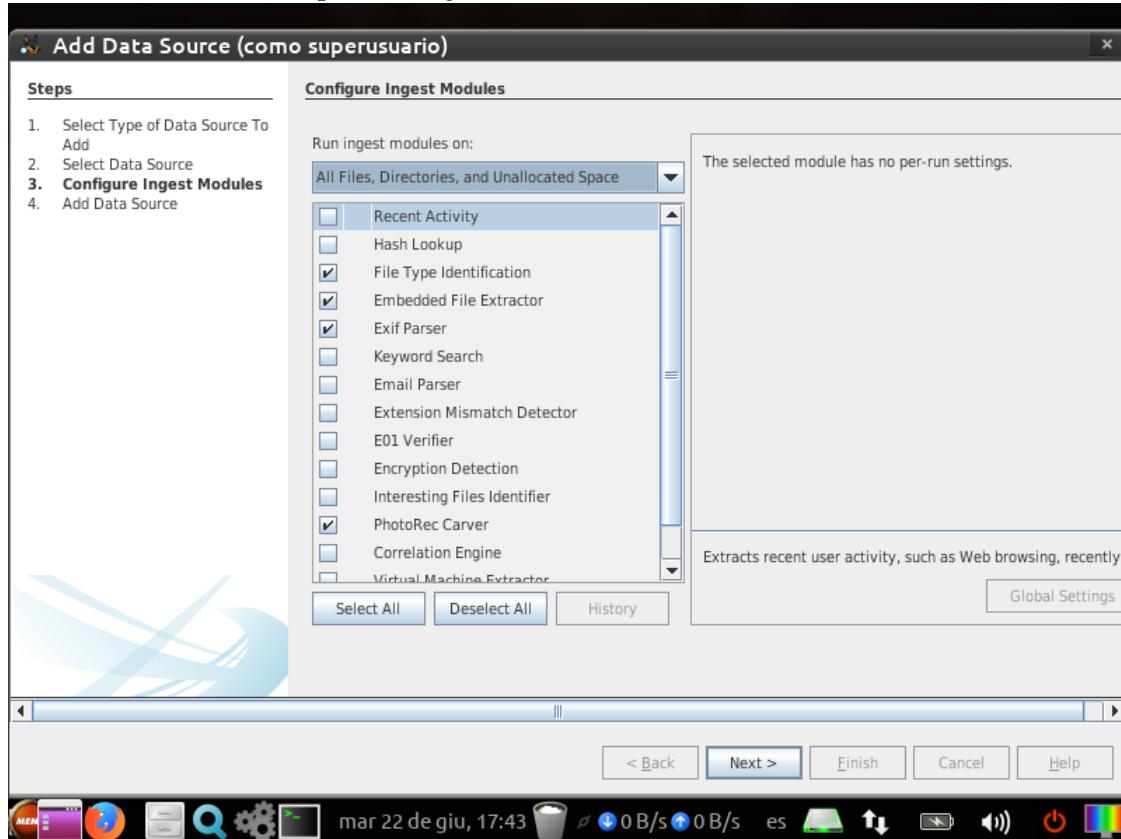
Añadimos la imagen a analizar.

Figura 39: Ejercicio 7: Selección de la imagen



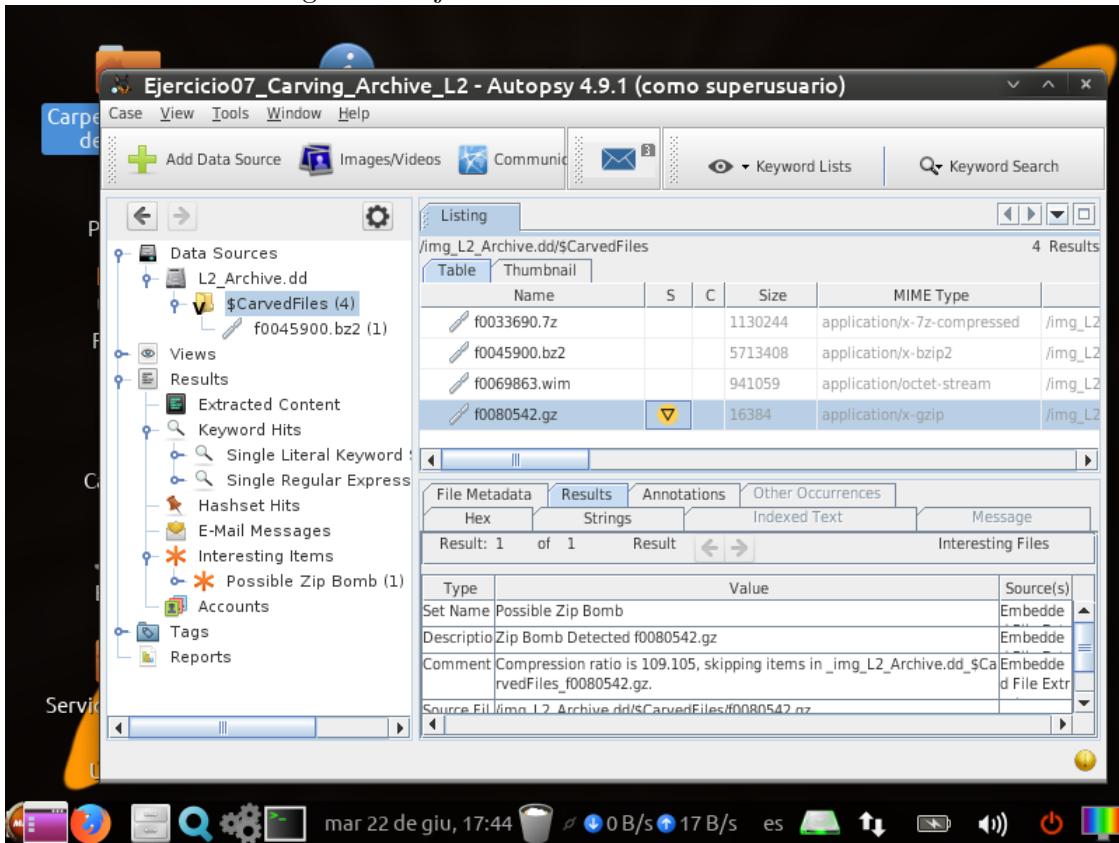
Se seleccionan los módulos de identificación de tipos de fichero, parseador de Exif, *PhotoRec Carver* y el módulo de extracción de ficheros.

Figura 40: Ejercicio 7: Selección de módulos



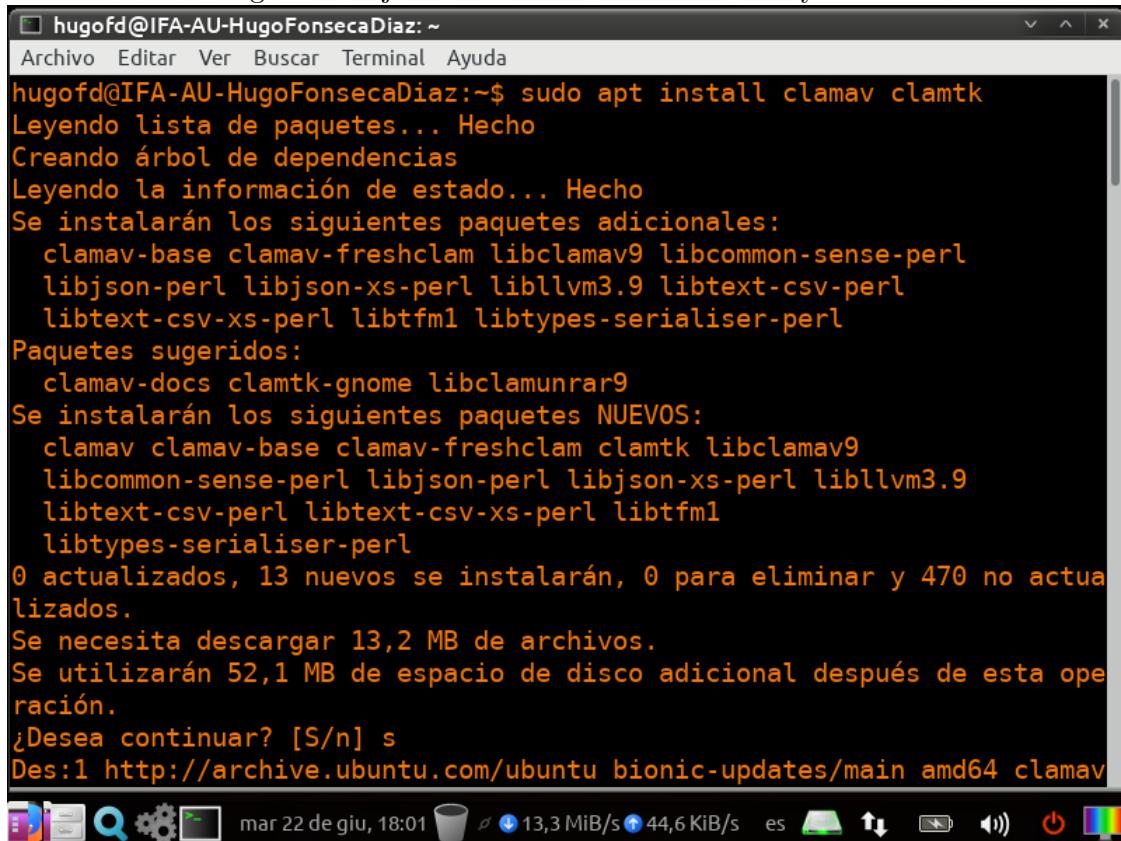
Se ejecuta el análisis y se observa que Autopsy ha detectado una posible bomba zip entre uno de los ficheros comprimidos.

Figura 41: Ejercicio 7: Resultados del análisis



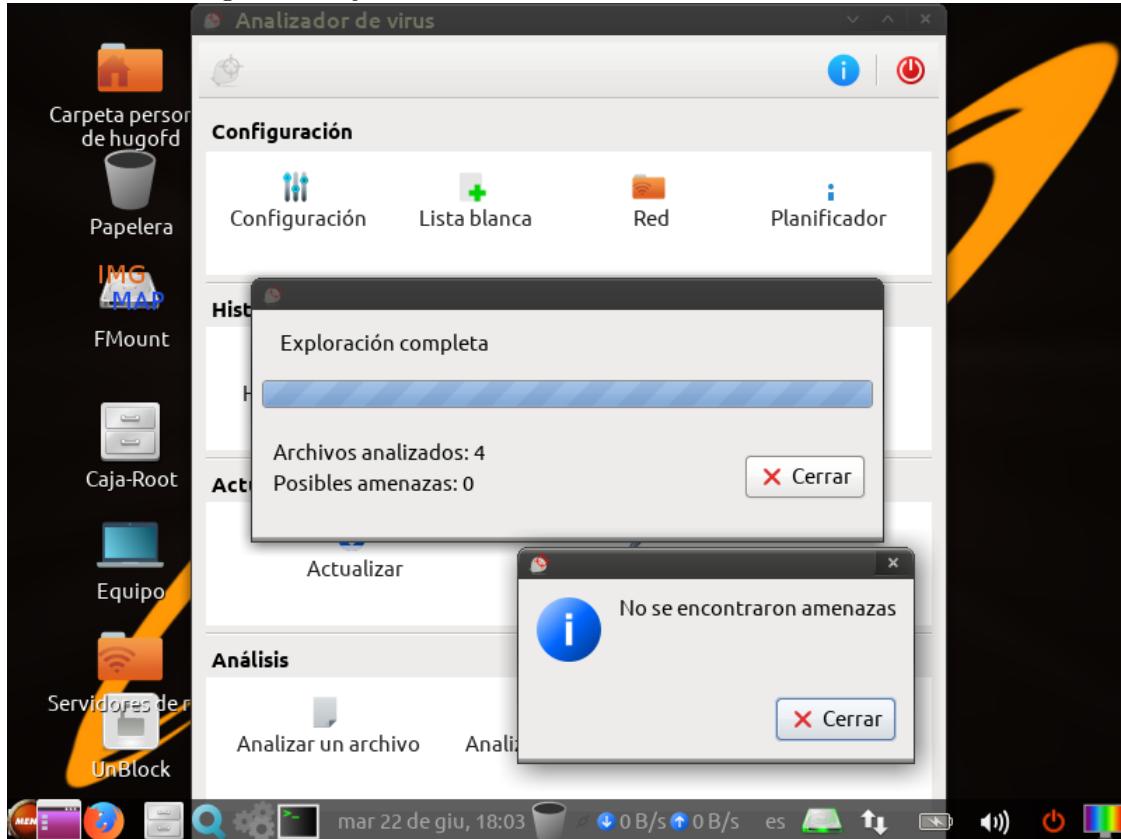
Se instalan los paquetes *clamav* y *clamtk* y se analiza la carpeta donde se extrajeron los ficheros comprimidos.

Figura 42: Ejercicio 7: Instalación de *clamav* y *clamtk*



```
hugofd@IFA-AU-HugoFonsecaDiaz: ~
Archivo Editar Ver Buscar Terminal Ayuda
hugofd@IFA-AU-HugoFonsecaDiaz:~$ sudo apt install clamav clamtk
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  clamav-base clamav-freshclam libclamav9 libcommon-sense-perl
    libjson-perl libjson-xs-perl libllvm3.9 libtext-csv-perl
    libtext-csv-xs-perl libtfm1 libtypes-serialiser-perl
Paquetes sugeridos:
  clamav-docs clamtk-gnome libclamunrar9
Se instalarán los siguientes paquetes NUEVOS:
  clamav clamav-base clamav-freshclam clamtk libclamav9
    libcommon-sense-perl libjson-perl libjson-xs-perl libllvm3.9
    libtext-csv-perl libtext-csv-xs-perl libtfm1
    libtypes-serialiser-perl
0 actualizados, 13 nuevos se instalarán, 0 para eliminar y 470 no actualizados.
Se necesita descargar 13,2 MB de archivos.
Se utilizarán 52,1 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 clamav
[...]
```

Figura 43: Ejercicio 7: Resultados del análisis del antivirus



- a) El antivirus no detecta nada, pero Autopsy si que notificó que uno de los ficheros podía ser una bomba zip. Este es un ataque que comprime con una alta proporción una gran cantidad de datos en un archivo comprimido de pocos datos. Sirve para inutilizar los programas que descomprimen dicho fichero, normalmente se busca inutilizar un antivirus, para luego ejecutar otro tipo de malware.
- b) Bomba zip.

8. Ejercicio 8

Se crea el caso en Autopsy con los datos solicitados.

Figura 44: Ejercicio 8: Creación del caso

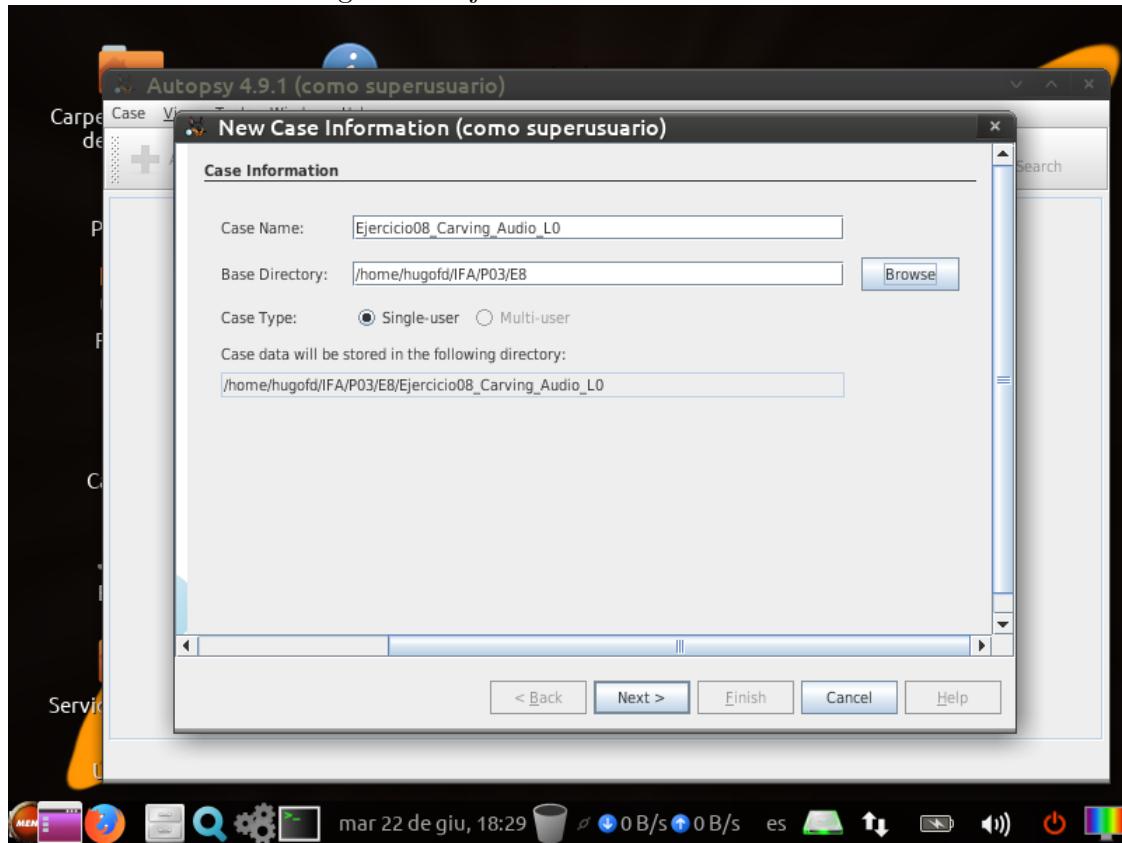
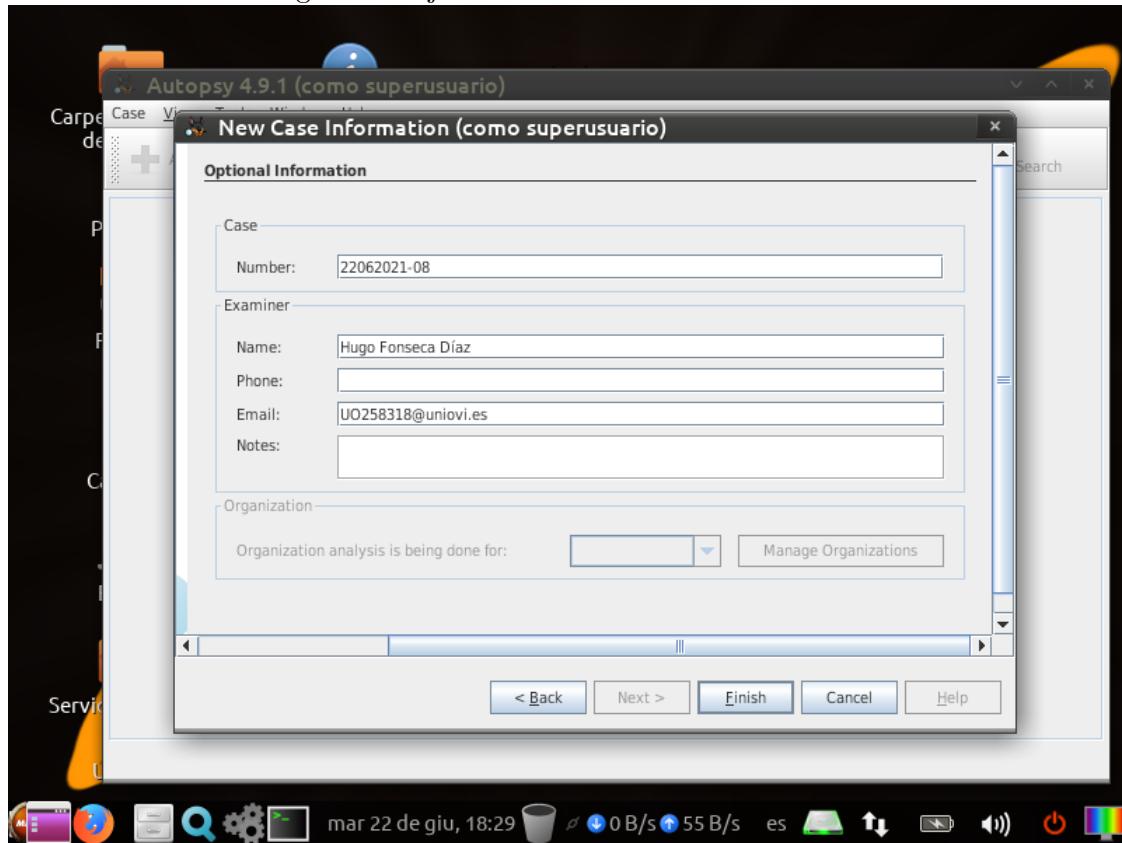
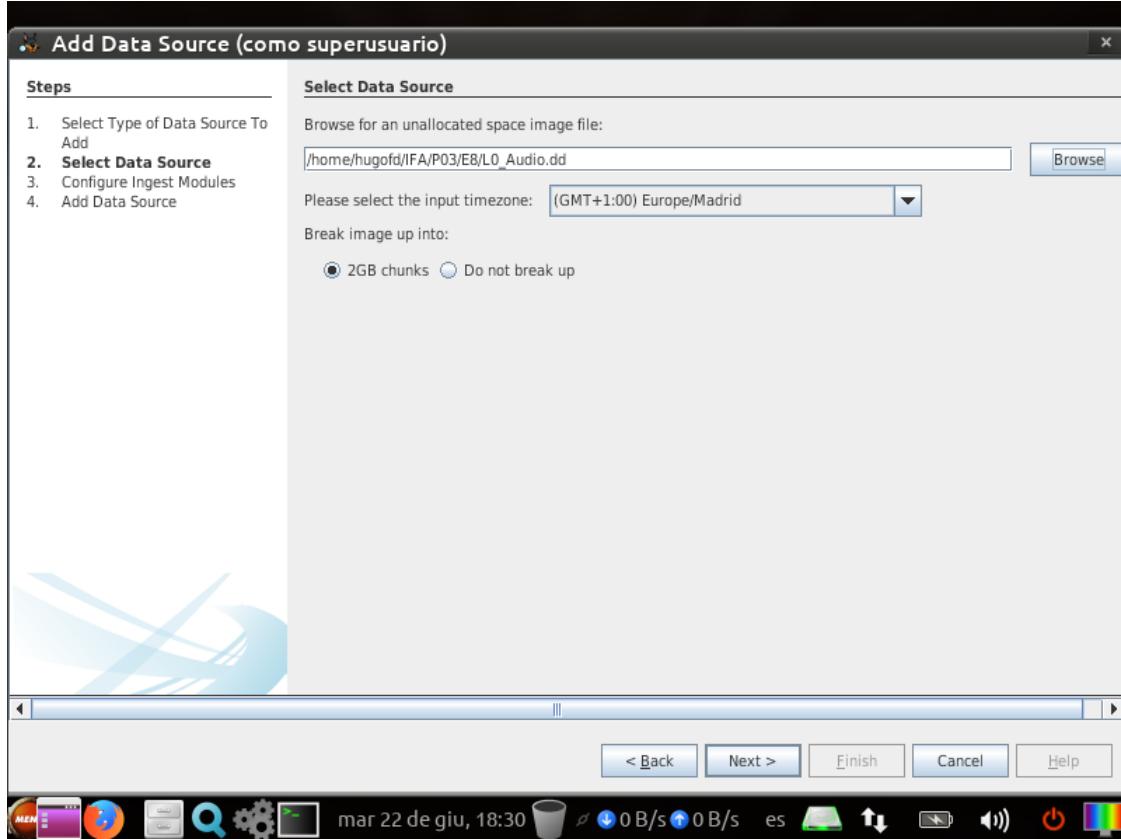


Figura 45: Ejercicio 8: Detalles del examinador



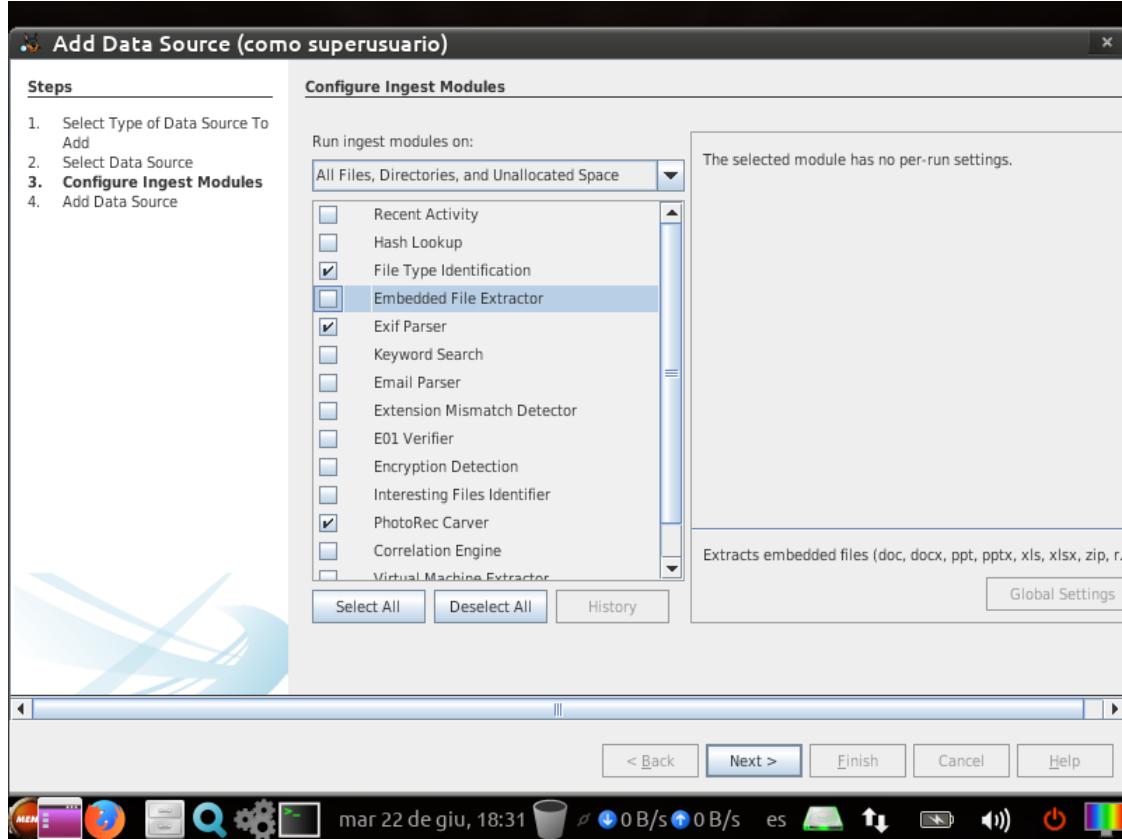
Añadimos la imagen a analizar.

Figura 46: Ejercicio 8: Selección de la imagen



Se seleccionan los módulos de identificación de tipos de fichero, parseador de Exif y *PhotoRec Carver*.

Figura 47: Ejercicio 8: Selección de módulos



Para llenar la tabla se usarán los datos obtenidos al ejecutar el análisis de Autopsy y mediante el uso de la herramienta *MediaInfo*.

Figura 48: Ejercicio 8: Resultados del análisis

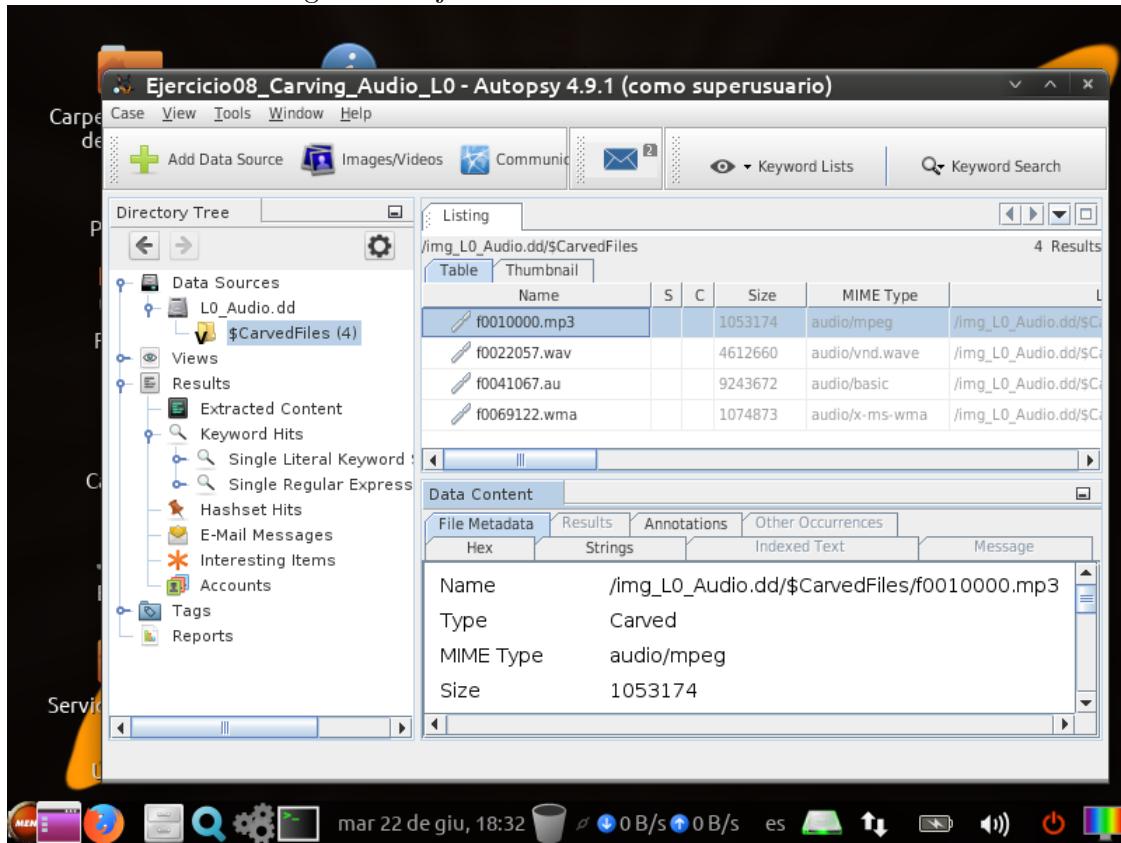
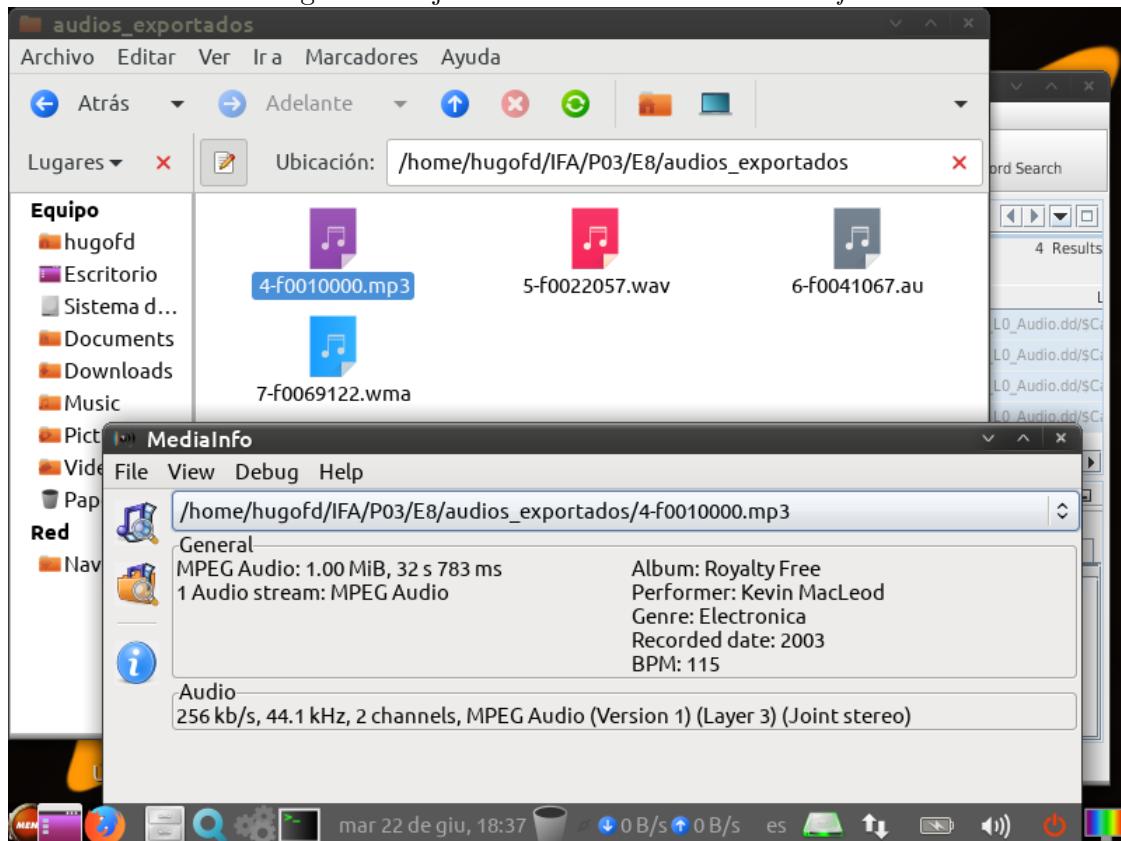


Figura 49: Ejercicio 8: Herramienta *MediaInfo*



| Nombre del fichero en Autopsy | Tamaño del fichero (en Bytes) | Tipo MIME | Autor | Género | Duración | Tasa de Muestreo |
|-------------------------------|-------------------------------|----------------|--------------|-------------|-----------|------------------|
| f0010000.mp3 | 1053174 | audio/mpeg | Kevin McLeod | Electronica | 32s 783ms | 44.1kHz |
| f0022057.wav | 4612660 | audio/vnd.wave | - | - | 26s 148ms | 44.1kHz |
| f0041067.au | 9243672 | audio/basic | - | - | 3min 29s | 44.1kHz |
| f0069122.wma | 1074873 | audio/x-ms-wma | - | (80) | 1min 5s | 44.1kHz |

9. Ejercicio 9

Se crea el caso en Autopsy con los datos solicitados.

Figura 50: Ejercicio 9: Creación del caso

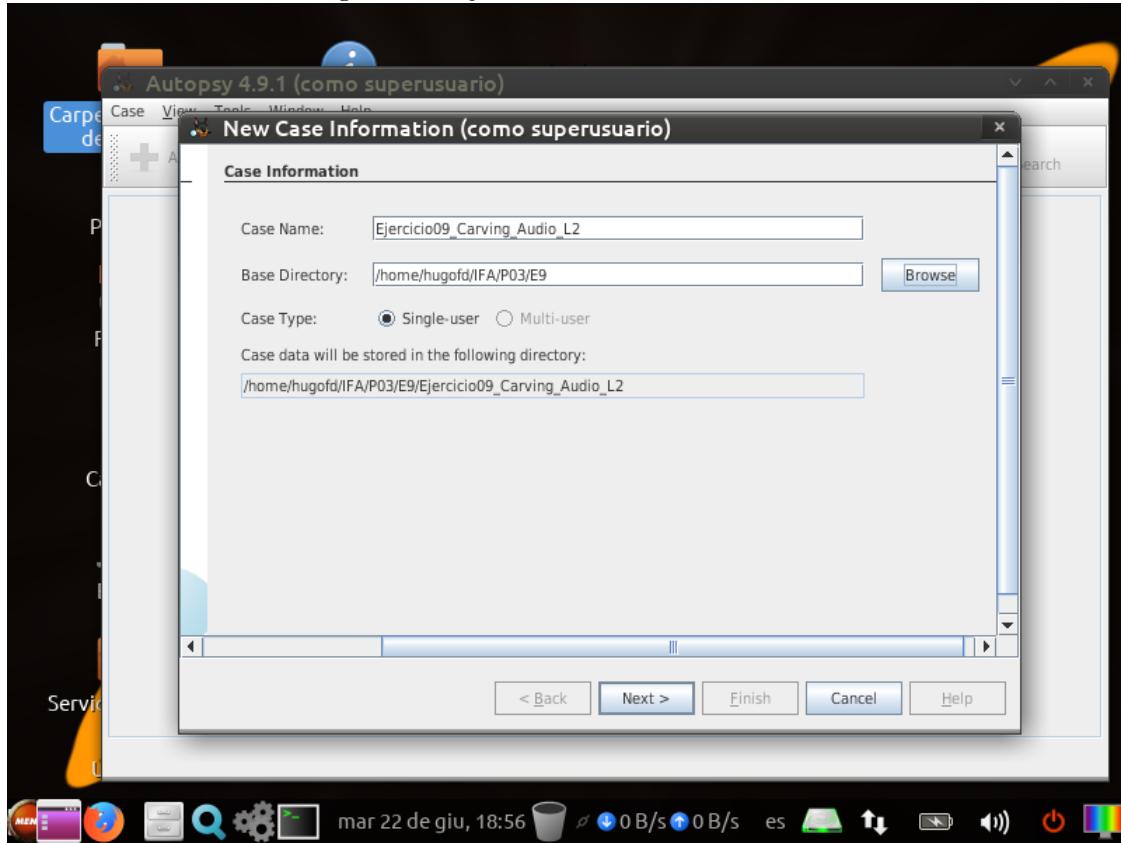
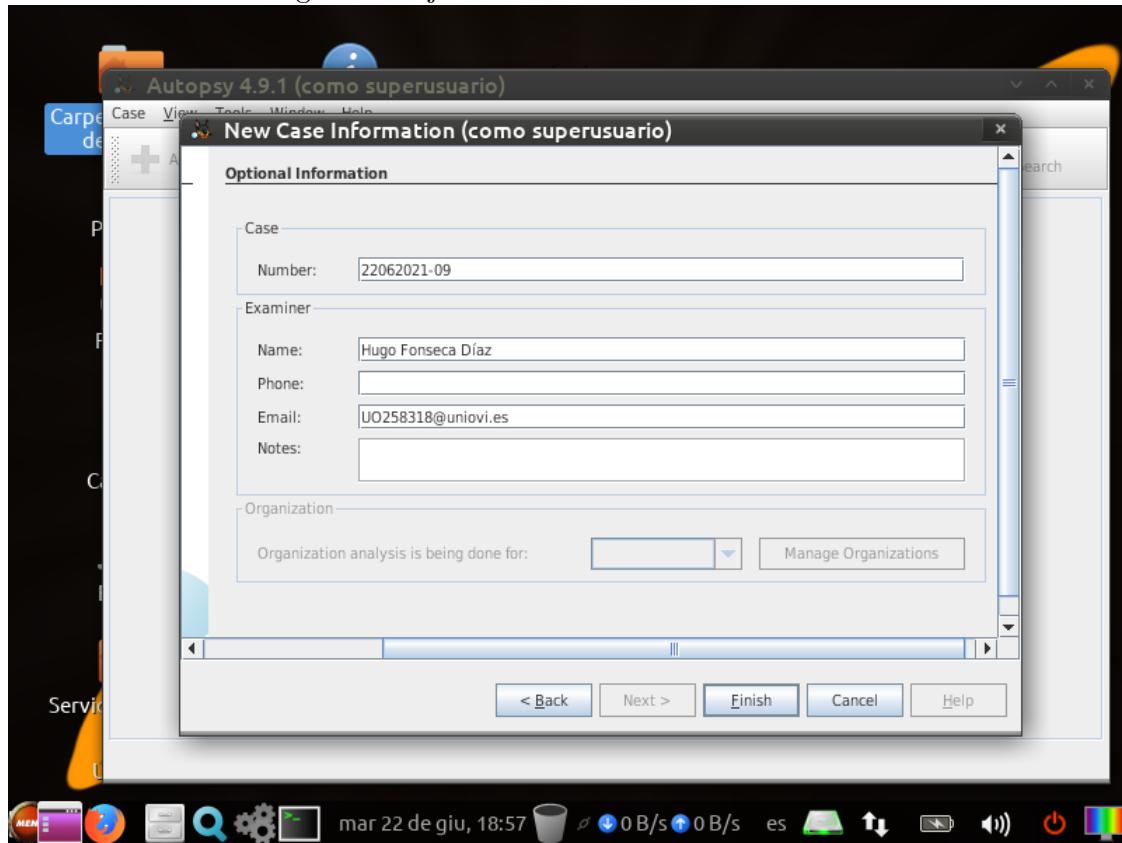
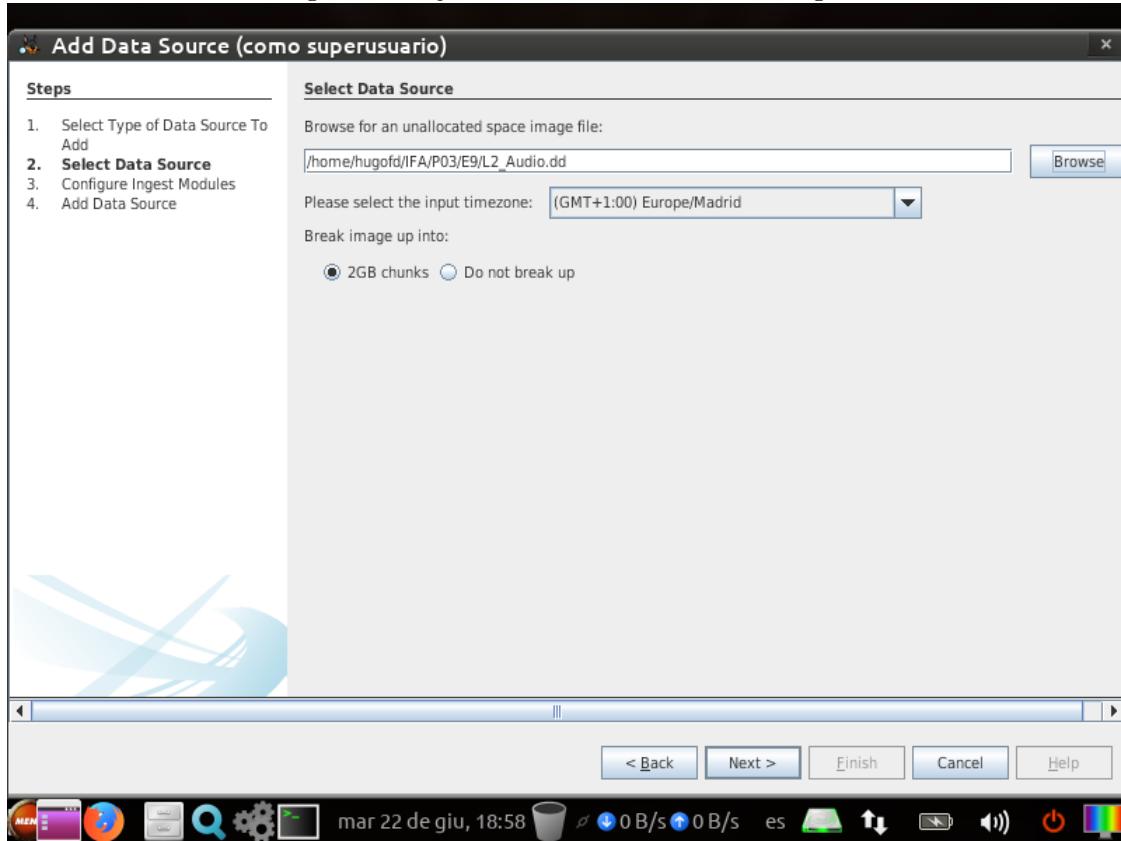


Figura 51: Ejercicio 9: Detalles del examinador



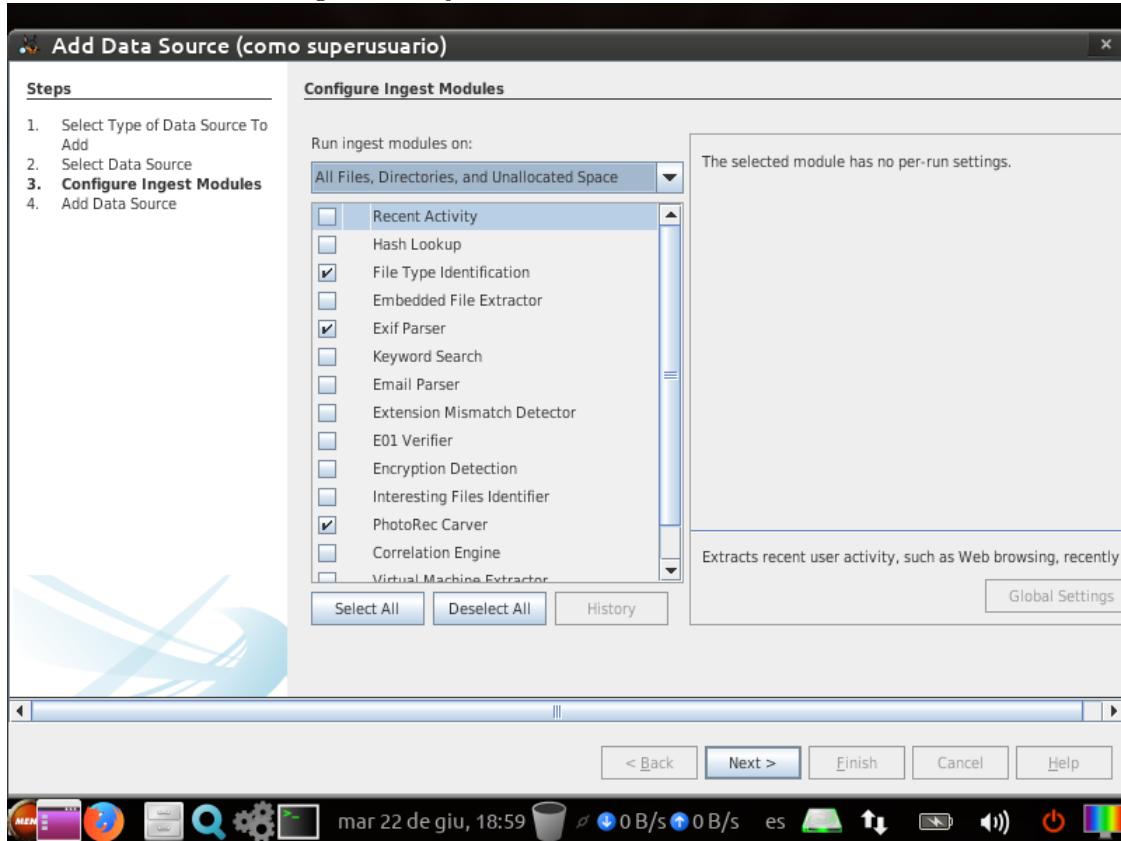
Añadimos la imagen a analizar.

Figura 52: Ejercicio 9: Selección de la imagen



Se seleccionan los módulos de identificación de tipos de fichero, parseador de Exif y *PhotoRec Carver*.

Figura 53: Ejercicio 9: Selección de módulos



Para llenar la tabla se usarán los datos obtenidos al ejecutar el análisis de Autopsy y mediante el uso de la herramienta *MediaInfo*.

Figura 54: Ejercicio 9: Resultados del análisis

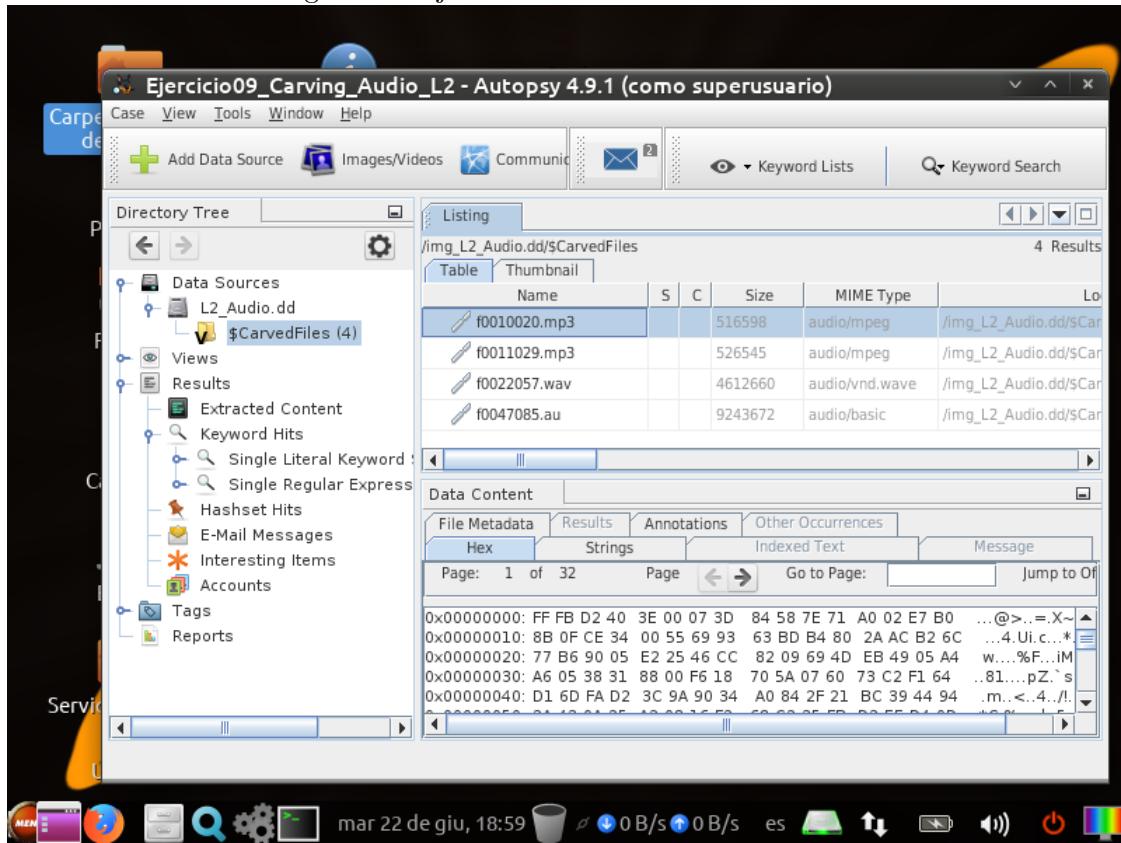
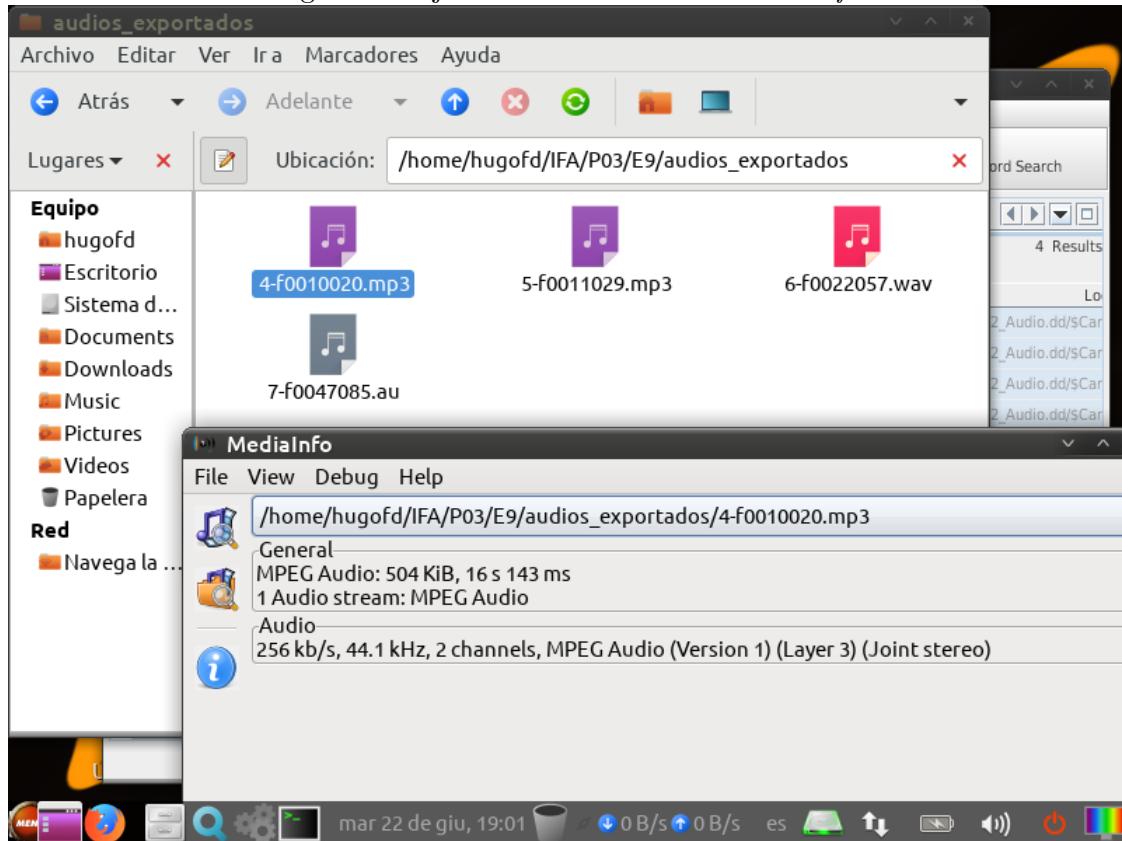


Figura 55: Ejercicio 9: Herramienta *MediaInfo*



| Nombre del fichero en Autopsy | Tamaño del fichero (en Bytes) | Tipo MIME | Autor | Género | Duración | Tasa de Muestreo |
|-------------------------------|-------------------------------|----------------|--------------|-------------|-----------|------------------|
| f0010020.mp3 | 516598 | audio/mpeg | - | - | 16s 143ms | 44.1kHz |
| f0011029.mp3 | 526545 | audio/mpeg | Kevin McLeod | Electronica | 16s 326ms | 44.1kHz |
| f0022057.wav | 4612660 | audio/vnd.wave | - | - | 26s 148ms | 44.1kHz |
| f0047085.au | 9243672 | audio/basic | - | - | 3min 29s | 44.1kHz |

Se puede observar que los dos ficheros mp3 son dos fragmentos del fichero mp3 del ejercicio 8.

10. Ejercicio 10

Se crea el caso en Autopsy con los datos solicitados.

Figura 56: Ejercicio 10: Creación del caso

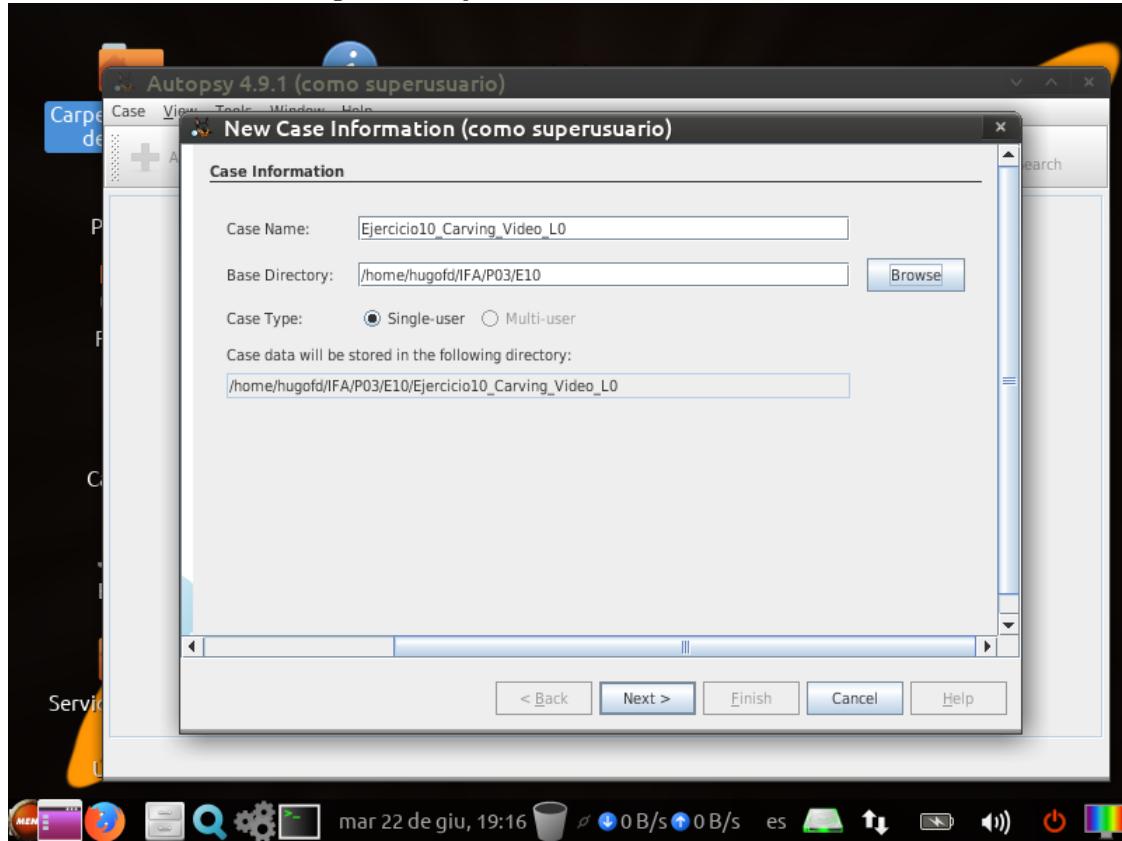
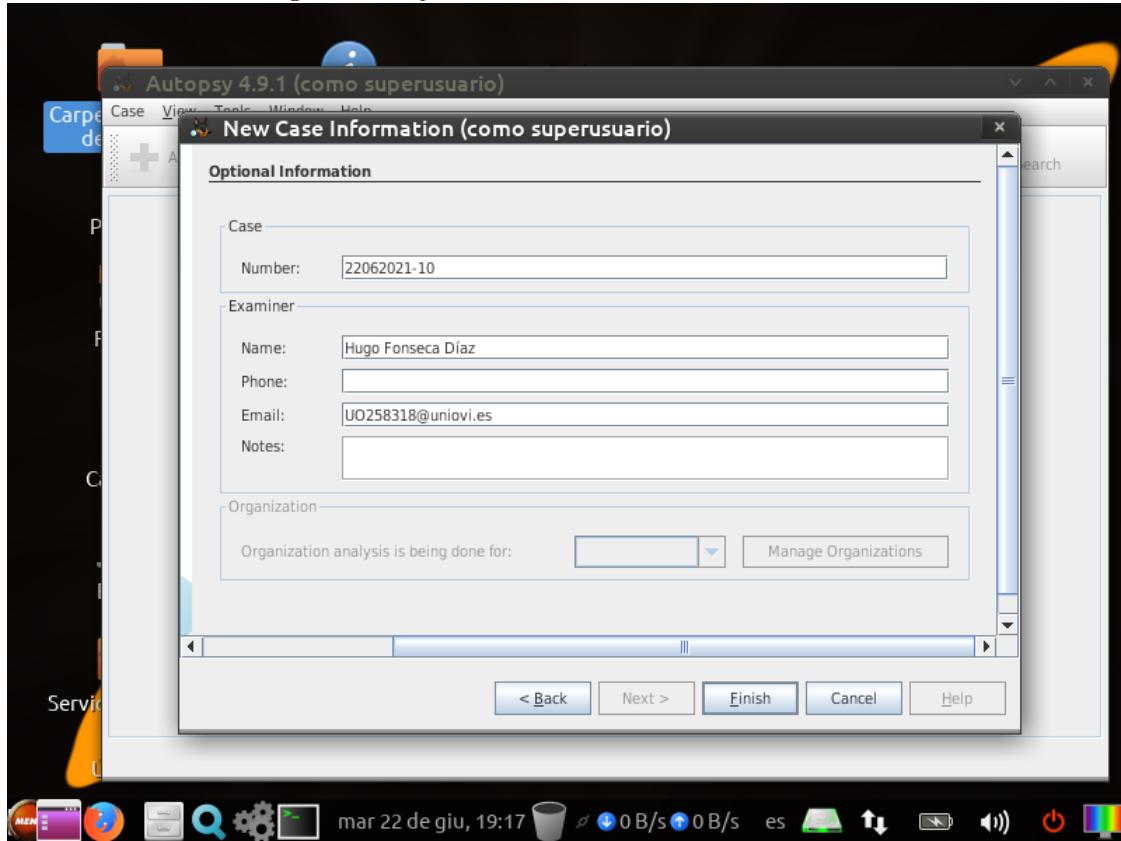
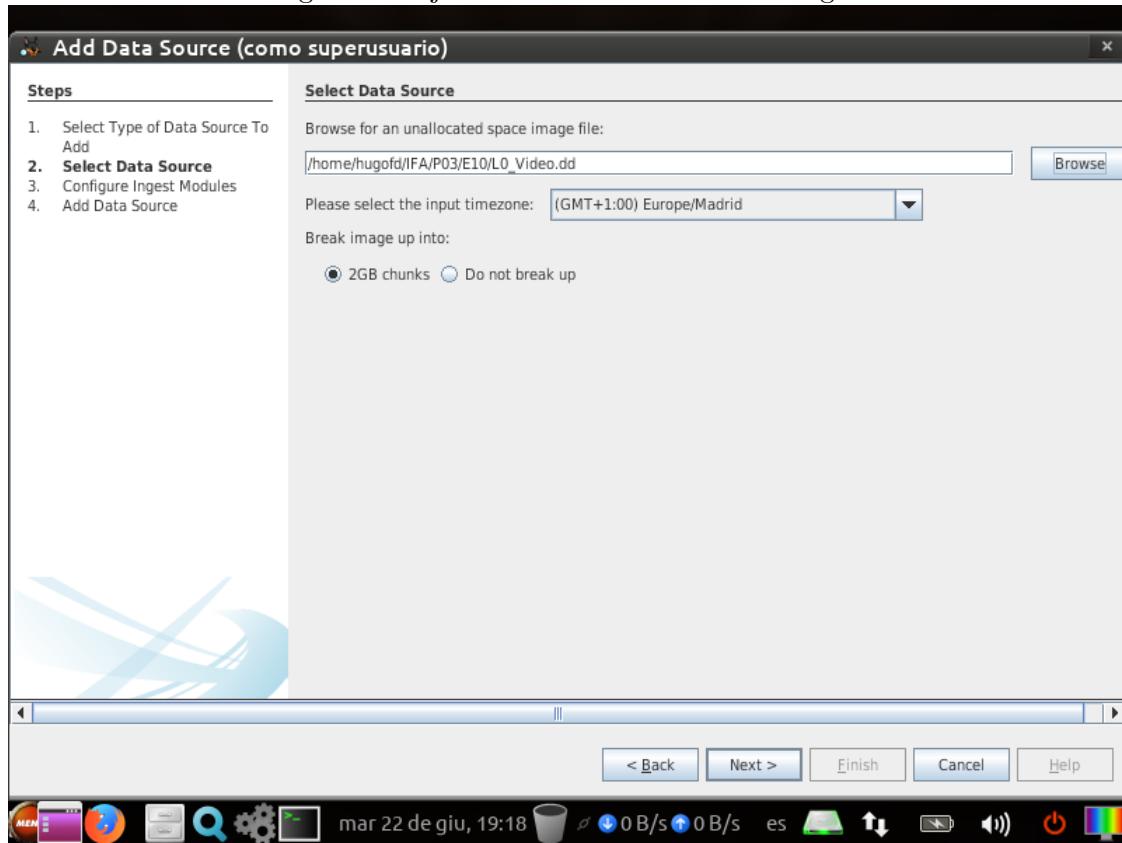


Figura 57: Ejercicio 10: Detalles del examinador



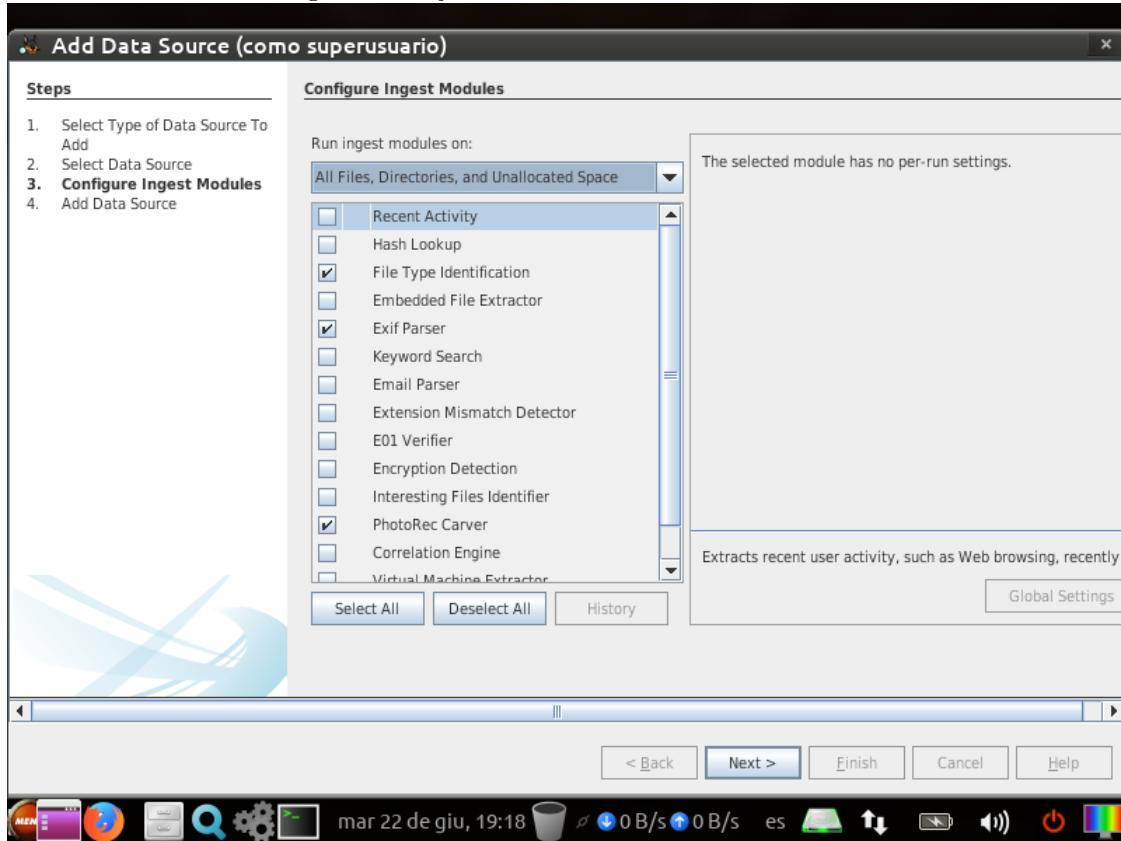
Añadimos la imagen a analizar.

Figura 58: Ejercicio 10: Selección de la imagen



Se seleccionan los módulos de identificación de tipos de fichero, parseador de Exif y *PhotoRec Carver*.

Figura 59: Ejercicio 10: Selección de módulos



Para llenar la tabla se usarán los datos obtenidos al ejecutar el análisis de Autopsy y mediante el uso de las herramientas *MediaInfo* y *FileInfo*.

Figura 60: Ejercicio 10: Resultados del análisis

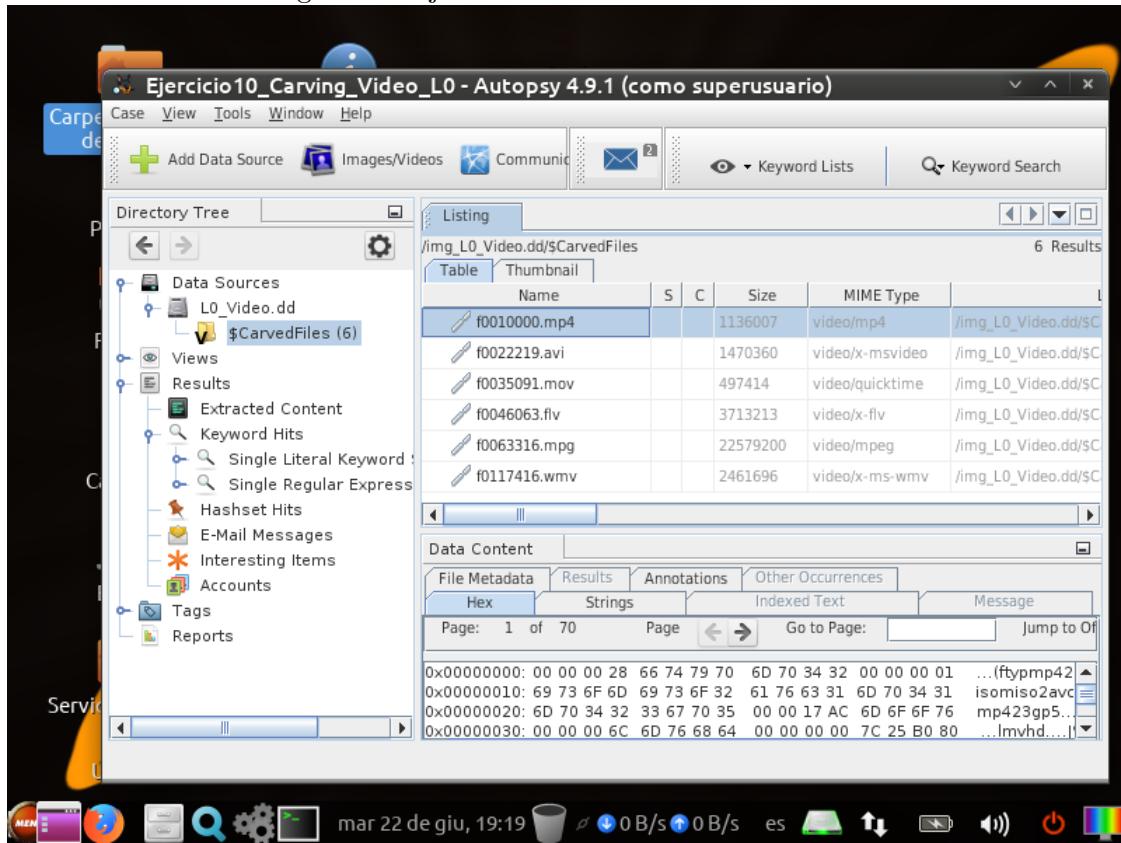
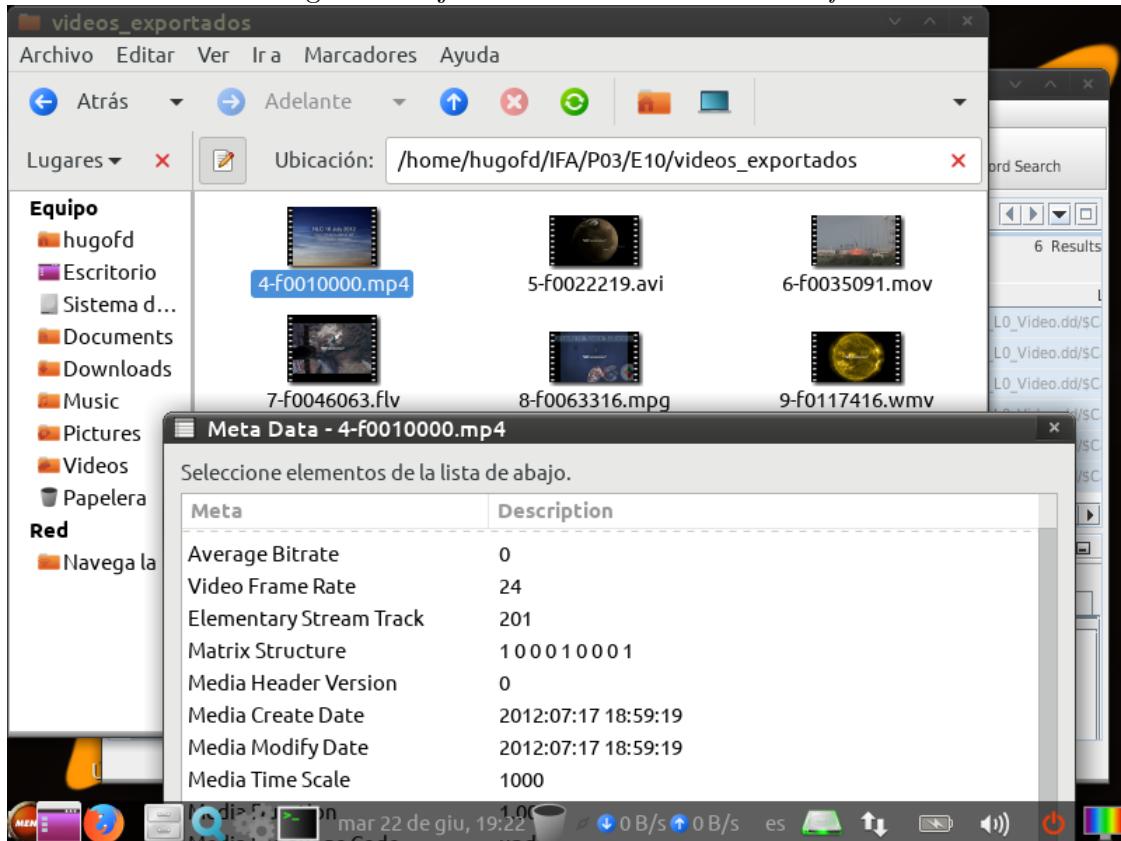


Figura 61: Ejercicio 10: Herramienta *FileInfo*

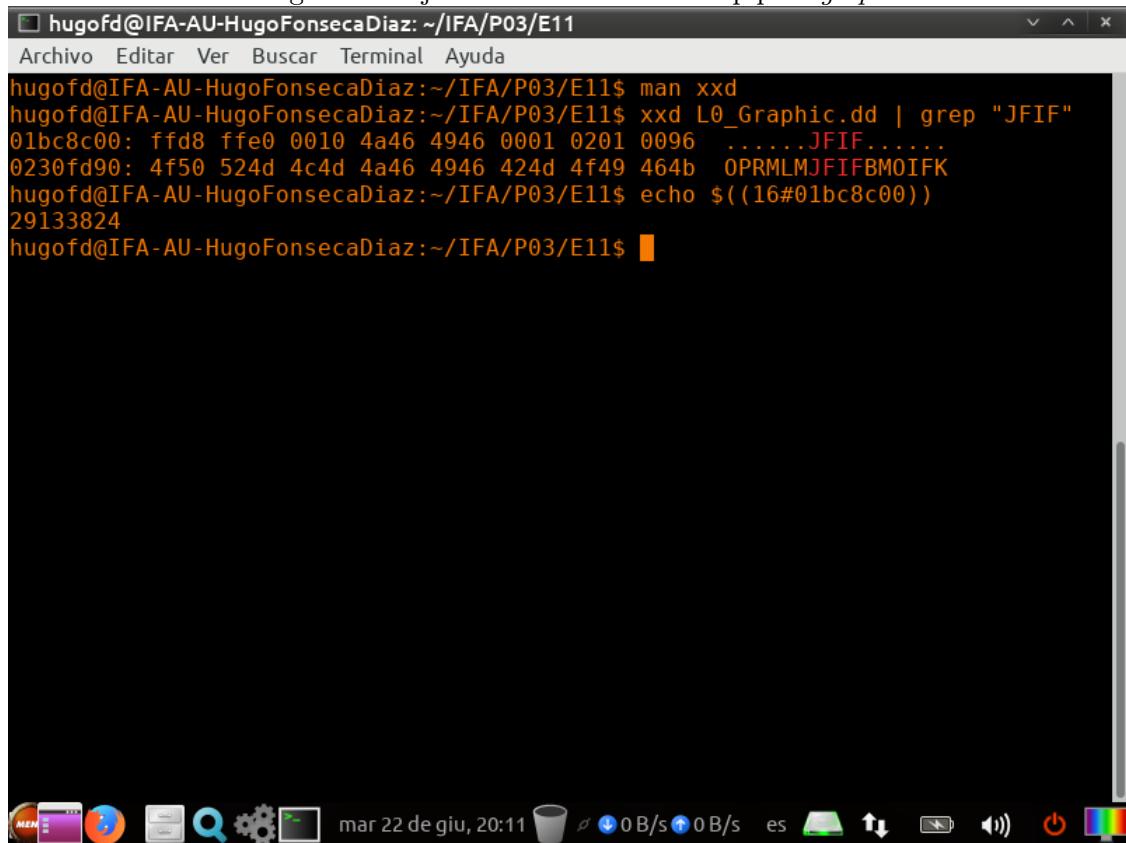


| Nombre del fichero en Autopsy | Tamaño del fichero (en Bytes) | Tipo MIME | FPS | Resolución | Tasa Muestreo | Duración | Fecha |
|-------------------------------|-------------------------------|-----------------|-------|------------|---------------|----------|---------------------|
| f001000.mp4 | 1136007 | video/mp4 | 24 | 512x340 | 24 bit depth | 16.38s | 2012/07/17 18:59:19 |
| f0022219.avi | 1470360 | video/x-msvideo | 30 | 512x288 | 44kHz | 10.13s | - |
| f0035091.mov | 497414 | video/quicktime | 29.97 | 512x288 | 24 bit depth | 12.15s | 2012/08/02 13:19:44 |
| f0046063.flv | 3713213 | video/x-flv | 30 | 640x480 | 44kHz | 26.07s | - |
| f0063316.mpg | 22579200 | video/mpeg | 29.97 | 512x288 | 44kHz | 19.79s | - |
| f0117416.wmv | 2461696 | video/x-ms-wmv | 29.97 | 512x288 | 44kHz | 25.53s | 2012/08/16 12:10:37 |

11. Ejercicio 11

Para realizar la primera parte del ejercicio se utilizará el comando `xxd` junto al comando `grep`. Empezamos buscando la cadena *JFIF* en la imagen del ejercicio. Con esta búsqueda se sacará el offset en hexadecimal, y se convertirá a decimal en *bash*.

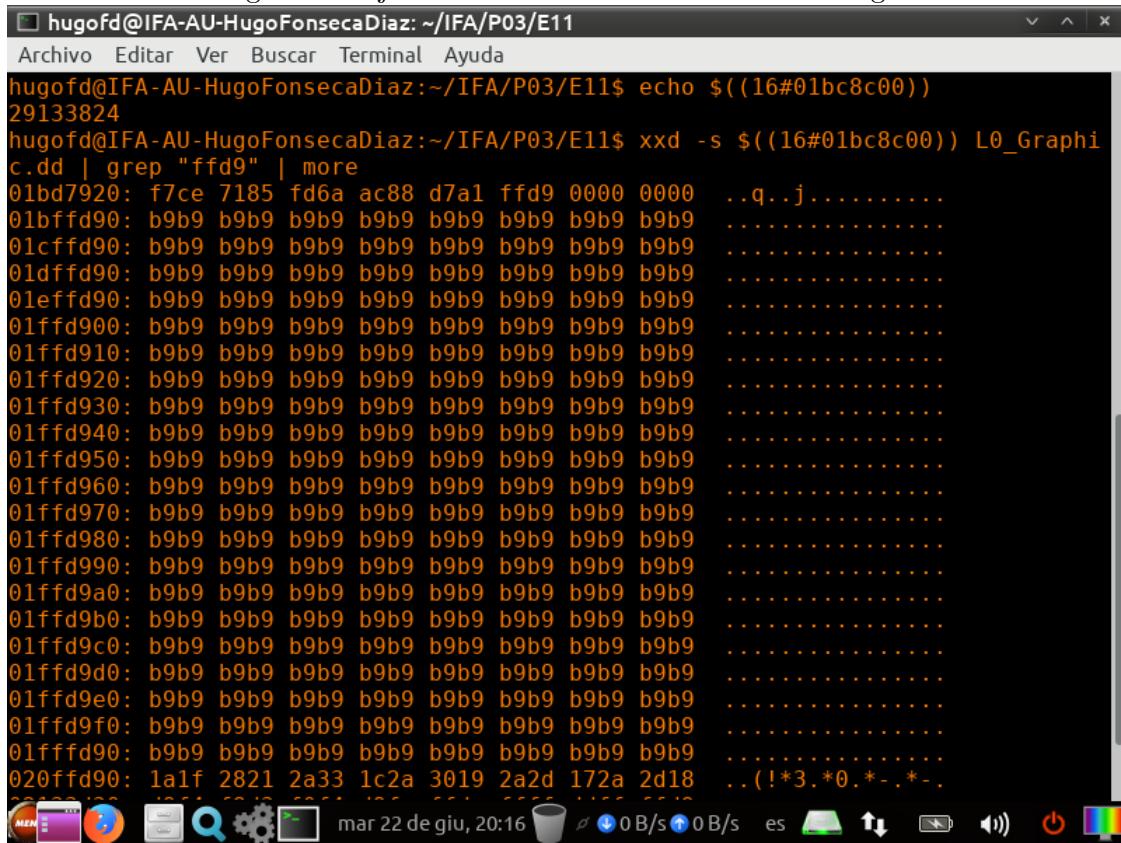
Figura 62: Ejercicio 11: `xxd` con una pipe a `grep`



```
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E11$ man xxd
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E11$ xxd L0_Graphic.dd | grep "JFIF"
01bc8c00: ffd8 ffe0 0010 4a46 4946 0001 0201 0096 .....JFIF.....
0230fd90: 4f50 524d 4c4d 4a46 4946 424d 4f49 464b OPRMLMJFIFBMOIFK
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E11$ echo $((16#01bc8c00))
29133824
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E11$
```

Una vez obtenido el offset, se buscará el final de la imagen. Para ello se buscará con `grep` la cadena *ffd9* pasándole al comando `xxd` el offset obtenido previamente.

Figura 63: Ejercicio 11: Buscando el final de la imagen



The screenshot shows a terminal window titled "hugofd@IFA-AU-HugoFonsecaDiaz: ~/IFA/P03/E11". The window contains the following command and its output:

```
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E11$ echo $((16#01bc8c00))
29133824
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E11$ xxd -s $((16#01bc8c00)) L0_Graphi
c.dd | grep "ffd9" | more
01bd7920: f7ce 7185 fd6a ac88 d7a1 ffd9 0000 0000  ..q..j.....
01bffd90: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01cffd90: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01dfffd90: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01efffd90: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd900: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd910: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd920: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd930: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd940: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd950: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd960: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd970: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd980: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd990: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd9a0: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd9b0: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd9c0: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd9d0: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd9e0: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffd9f0: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
01ffffd90: b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 b9b9 .....
020ffd90: 1a1f 2821 2a33 1c2a 3019 2a2d 172a 2d18 ..(!*3.*0.*-*.-.
```

The terminal window has a menu bar with "Archivo", "Editar", "Ver", "Buscar", "Terminal", and "Ayuda". The status bar at the bottom shows the date and time as "mar 22 de giu, 20:16" and network activity as "0 B/s ↑ 0 B/s ↓ es".

Una vez se ha encontrado el offset del final de la imagen, se le suma el desplazamiento, en este caso 10 bytes, y se calcula el tamaño restando el offset final del inicial.

Figura 64: Ejercicio 11: Calculando tamaño de imagen

The screenshot shows a terminal window titled "hugofd@IFA-AU-HugoFonsecaDiaz: ~/IFA/P03/E11". The window contains the following text:

```
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E11$ echo $((16#01bd7920))
29194528
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E11$ echo "Le sumamos 10 bytes de desp
lazamiento"
Le sumamos 10 bytes de desplazamiento
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E11$ python
Python 2.7.15rc1 (default, Nov 12 2018, 14:31:15)
[GCC 7.3.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> 29194528 + 10
29194538
>>> 29194538 - 29133824
60714
>>> print "Tamaño de la imagen: 60714 bytes"
Tamaño de la imagen: 60714 bytes
>>> █
```

At the bottom of the terminal window, there is a standard Linux system tray with icons for network, battery, volume, and other system status indicators.

Ahora se utiliza el comando `dd` con la información obtenida previamente y se comprueba que la imagen extraída es la del cartel que pone 'Welcome to Moscow'.

Figura 65: Ejercicio 11: Carving de la imagen

The screenshot shows a Linux desktop environment. At the top, there is a terminal window titled "hugofd@IFA-AU-HugoFonsecaDiaz: ~/IFA/P03/E11". The terminal displays the command "dd if=L0_Graphic.dd of=imgCarving.jpg count=60714 skip=29133824" and its execution results, including byte counts and speeds. Below the terminal is a file viewer window titled "imgCarving.jpg (como superusuario)". The viewer shows a photograph of a "Welcome to MOSCOW" sign made of brick. The sign features the text "Welcome to MOSCOW" in large letters, "HOME OF THE BEARS" in smaller letters below it, and "University of Idaho" on the left side. The desktop interface includes a dock with various icons at the bottom.

12. Ejercicio 12

Se crea el caso en Autopsy con los datos solicitados.

Figura 66: Ejercicio 12: Creación del caso

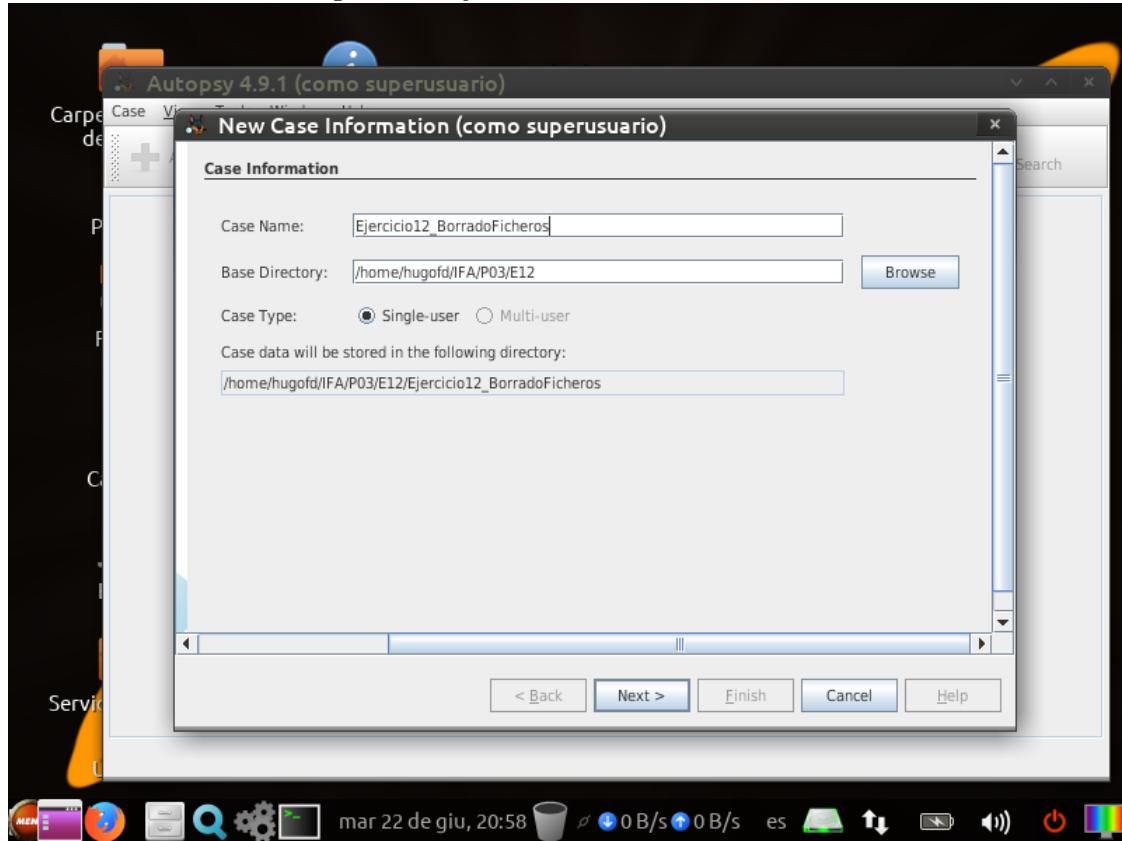
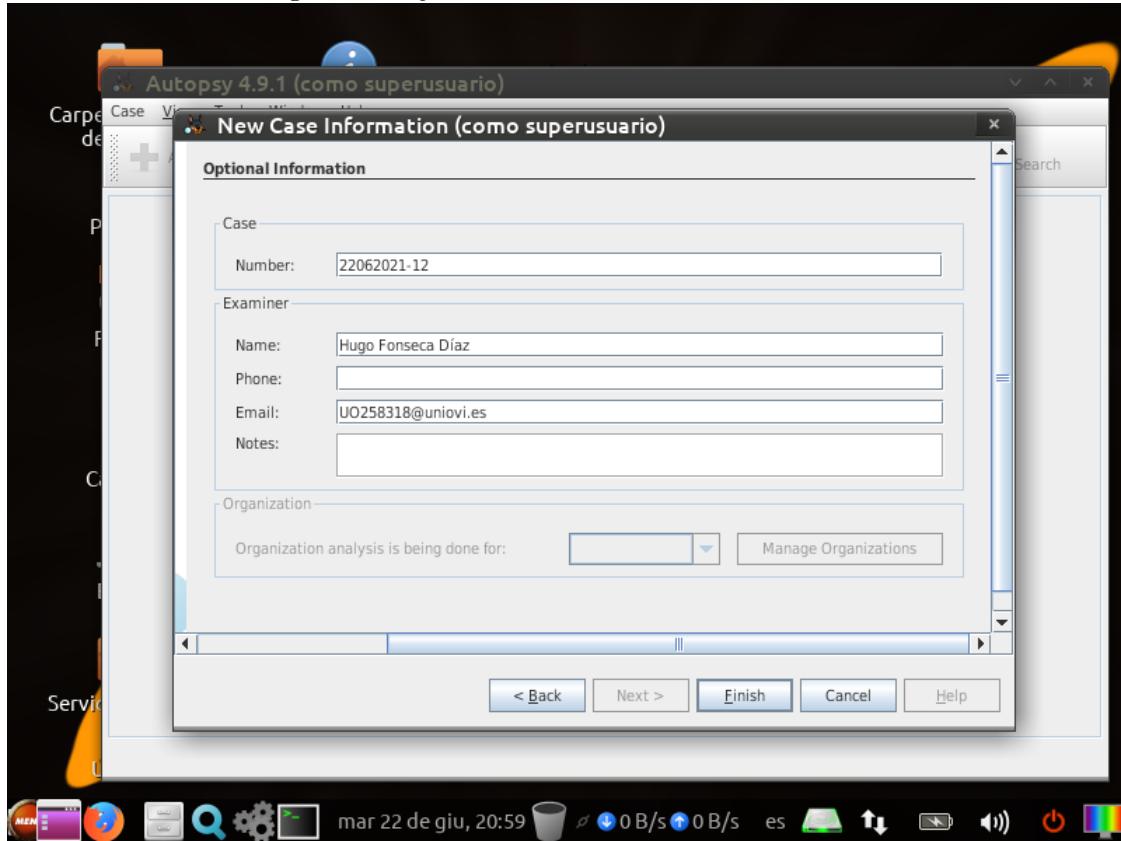
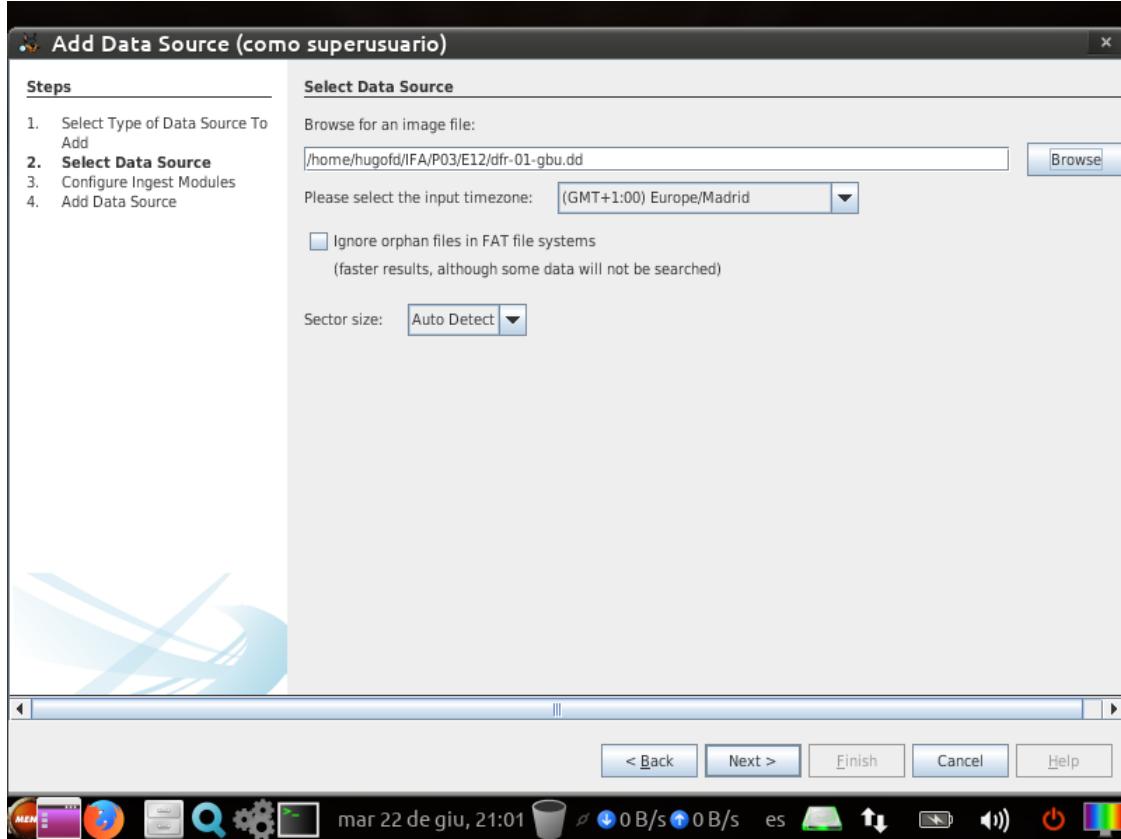


Figura 67: Ejercicio 12: Detalles del examinador



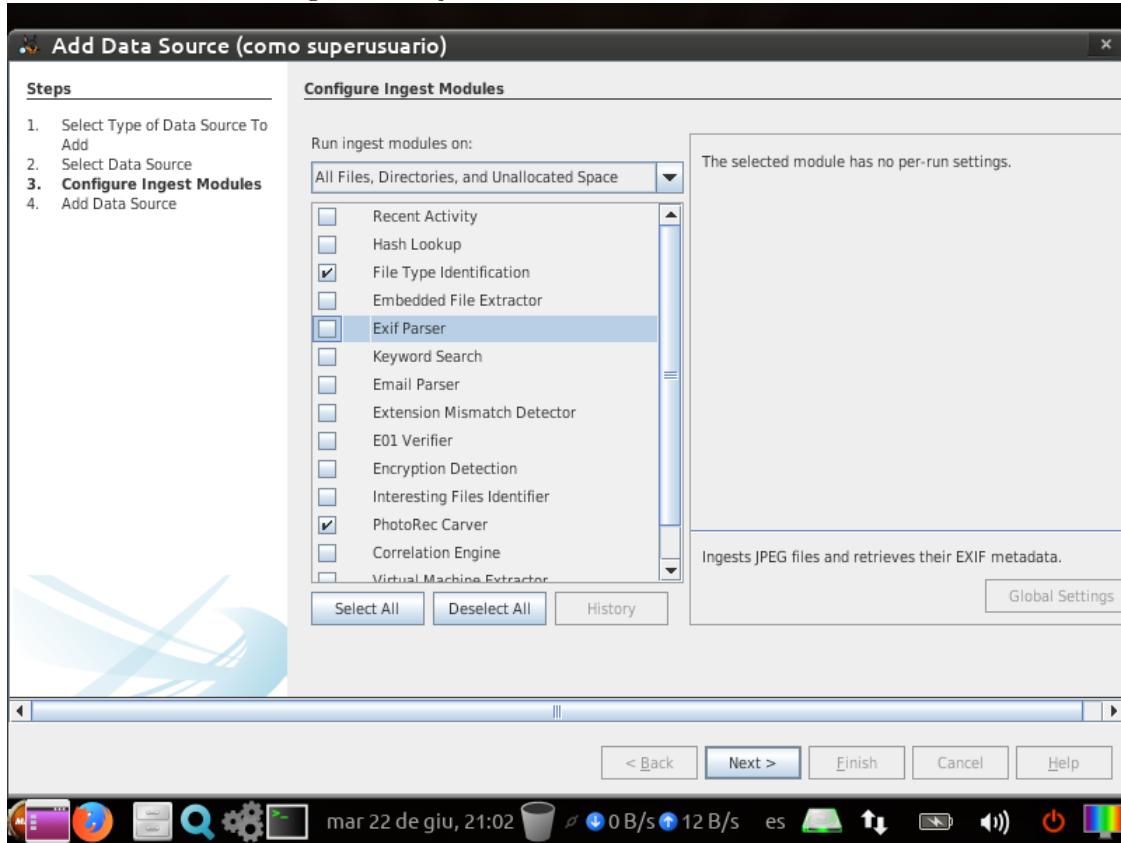
Añadimos la imagen a analizar.

Figura 68: Ejercicio 12: Selección de la imagen



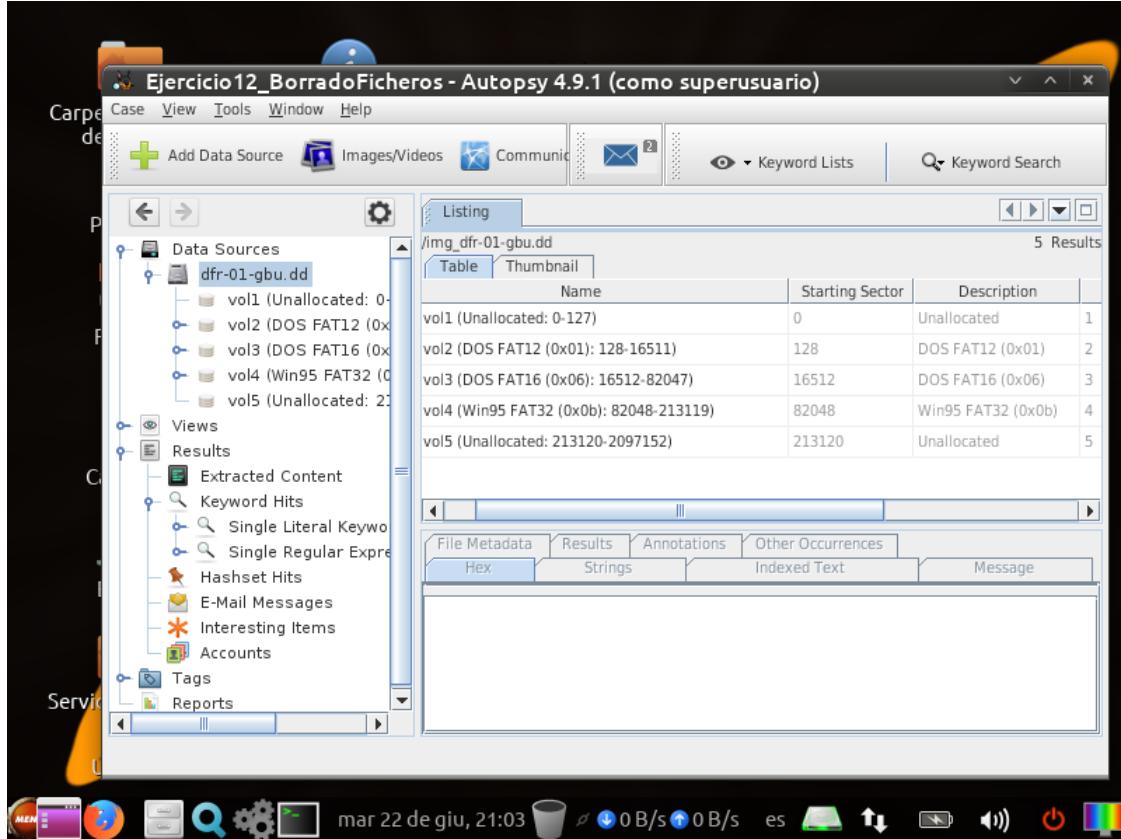
Se seleccionan los módulos de identificación de tipos de fichero y *PhotoRec Carver*.

Figura 69: Ejercicio 12: Selección de módulos



Se obtienen los resultados del análisis con los que se responderán a las diferentes cuestiones del ejercicio.

Figura 70: Ejercicio 12: Resultados del análisis

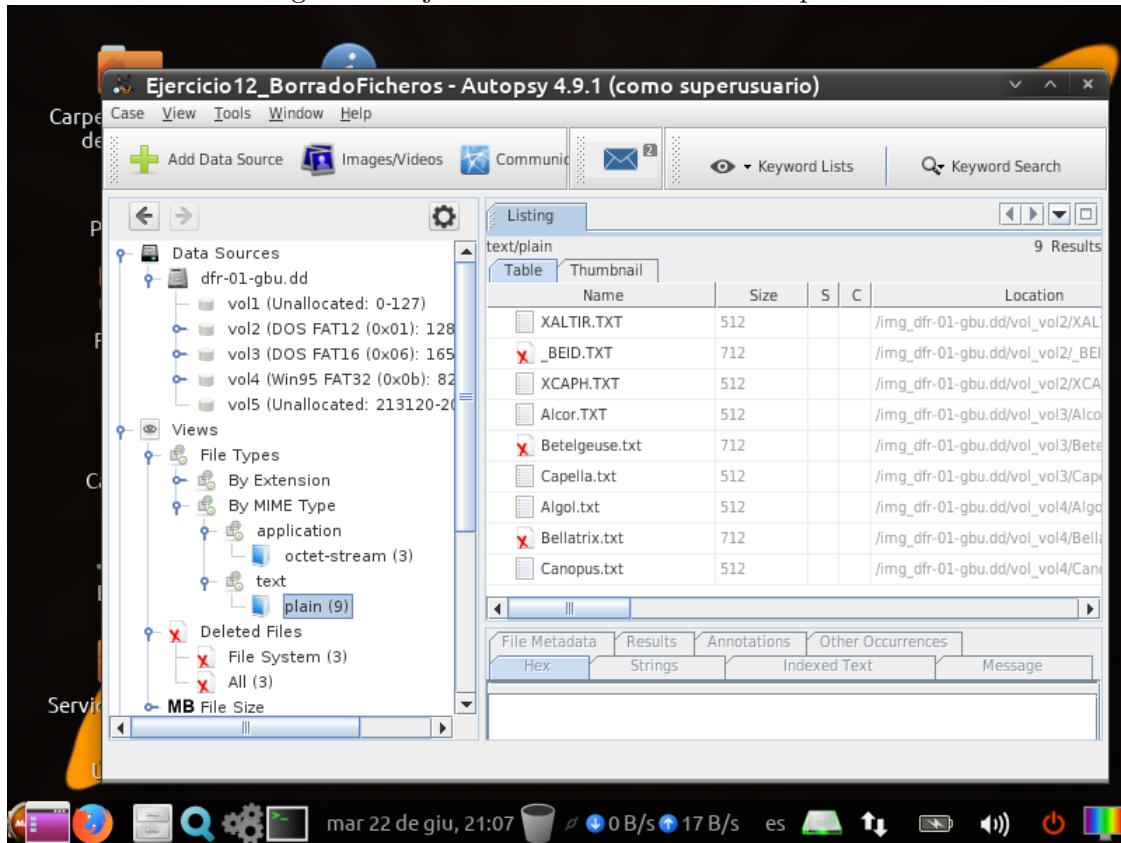


a)

| Número partición | Sector comienzo | Sector finalización | Tipo Sistema de Ficheros |
|------------------|-----------------|---------------------|--------------------------|
| 1 | 0 | 127 | Unallocated |
| 2 | 128 | 16511 | DOS FAT12 |
| 3 | 16512 | 82047 | DOS FAT16 |
| 4 | 82048 | 213119 | Win95 FAT32 |
| 5 | 213120 | 2097152 | Unallocated |

b) Para responder a esta cuestión se observan los resultados de la pestaña 'Views'.

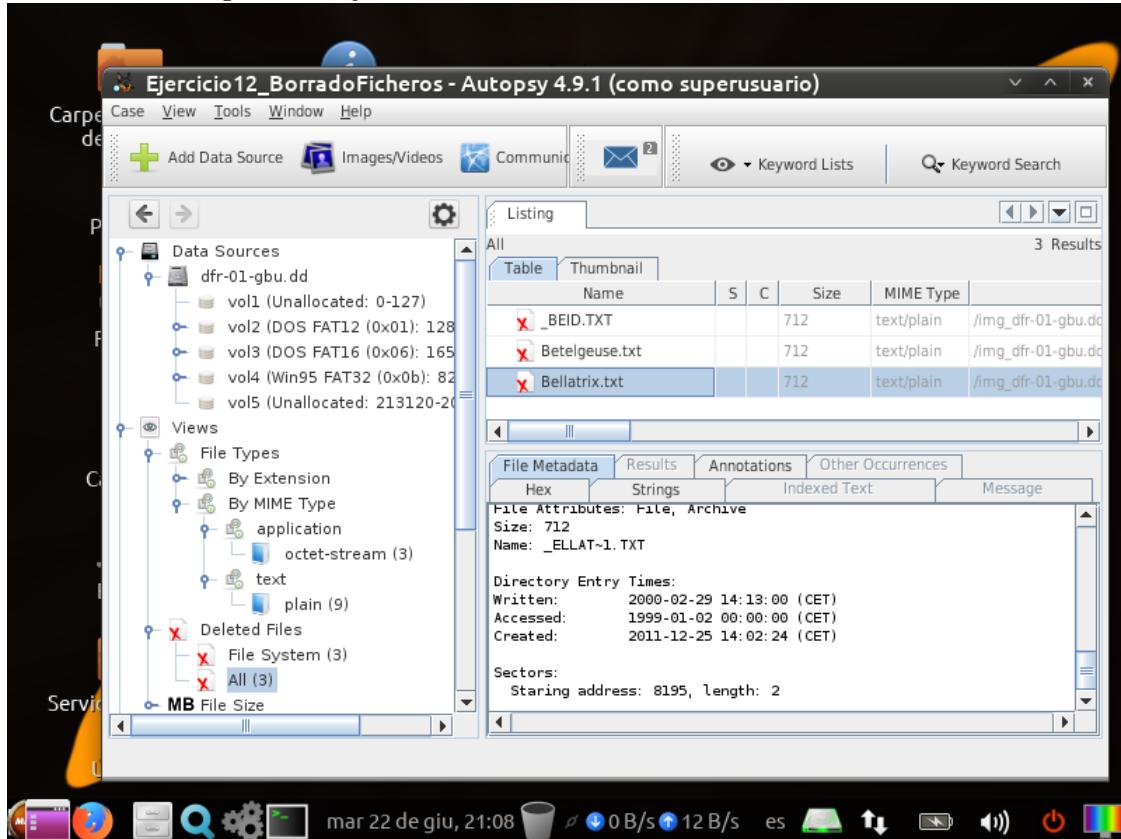
Figura 71: Ejercicio 12: Ficheros de texto plano



Se puede ver que hay 9 ficheros de texto plano, 3 de ellos borrados.

- c) Para llenar esta tabla se miran los metadatos que muestra Autopsy de cada archivo borrado.

Figura 72: Ejercicio 12: Metadatos de los ficheros borrados



| Nombre | Tamaño | Partición | Sector relativo | Acceso (GMT) | Modificación (GMT) | Creación (GMT) |
|----------------|--------|-----------|-----------------|------------------------|------------------------|------------------------|
| _BEID.txt | 712 | vol 2 | 170 | 1999/01/01 23:00:00 | 2000/02/29 13:11:00 | 2011/12/25 13:02:22 |
| Betelgeuse.txt | 712 | vol 3 | 546 | 1999/01/01 23:00:00 | 2000/02/29 13:12:00 | 2011/12/25 13:02:24 |
| Bellatrix.txt | 712 | vol 4 | 8195 | 1999/01/01 23:00:00 | 2000/02/29 13:13:00 | 2011/12/25 13:02:24 |

- d) Se muestran a continuación las líneas de tiempo de los tres ficheros borrados, en el filtro de la parte izquierda de la captura se observa el fichero actual.

Figura 73: Ejercicio 12: Línea temporal de *Bellatrix.txt*

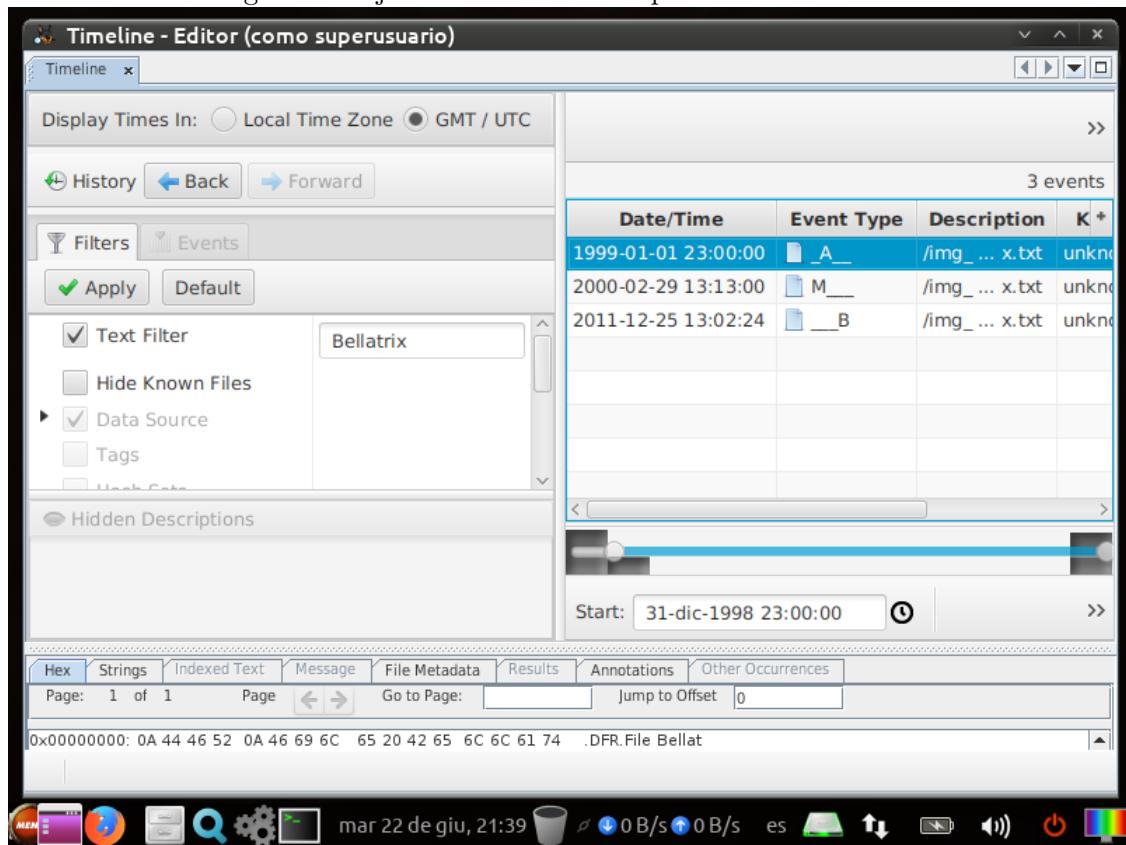


Figura 74: Ejercicio 12: Línea temporal de `_BEID.txt`

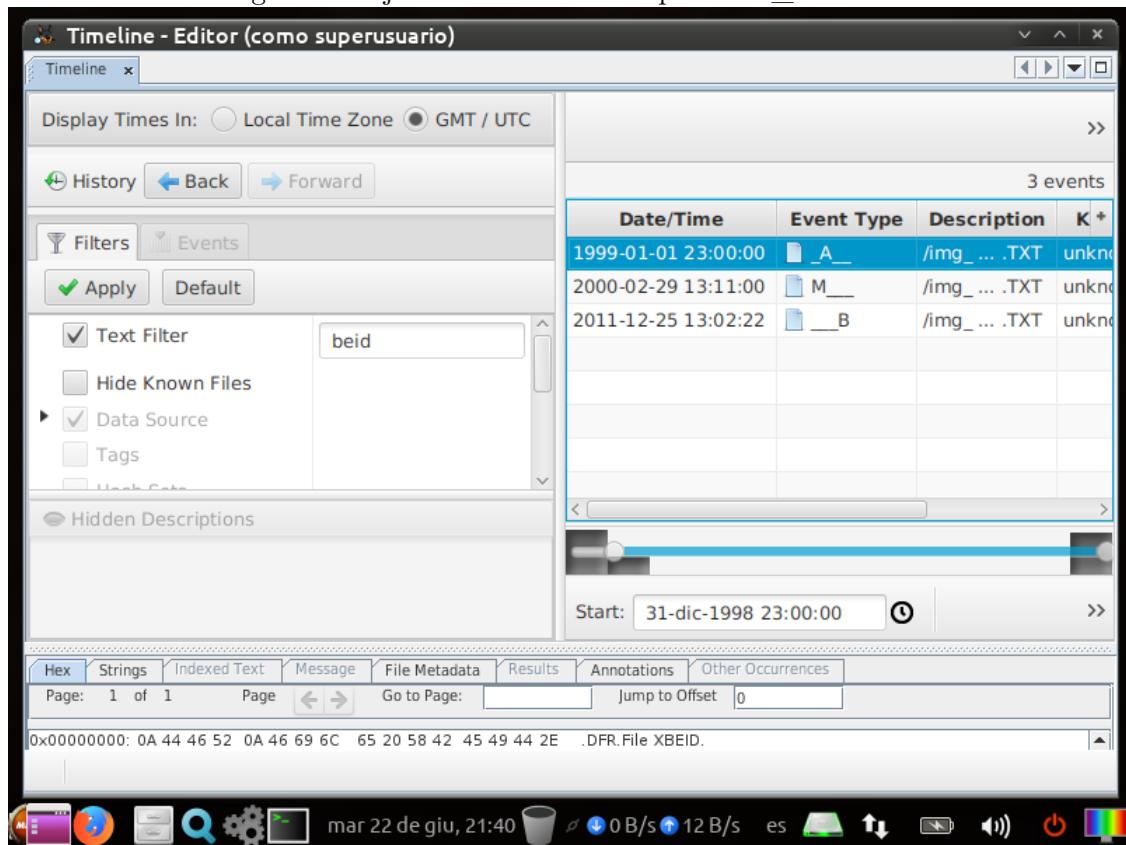
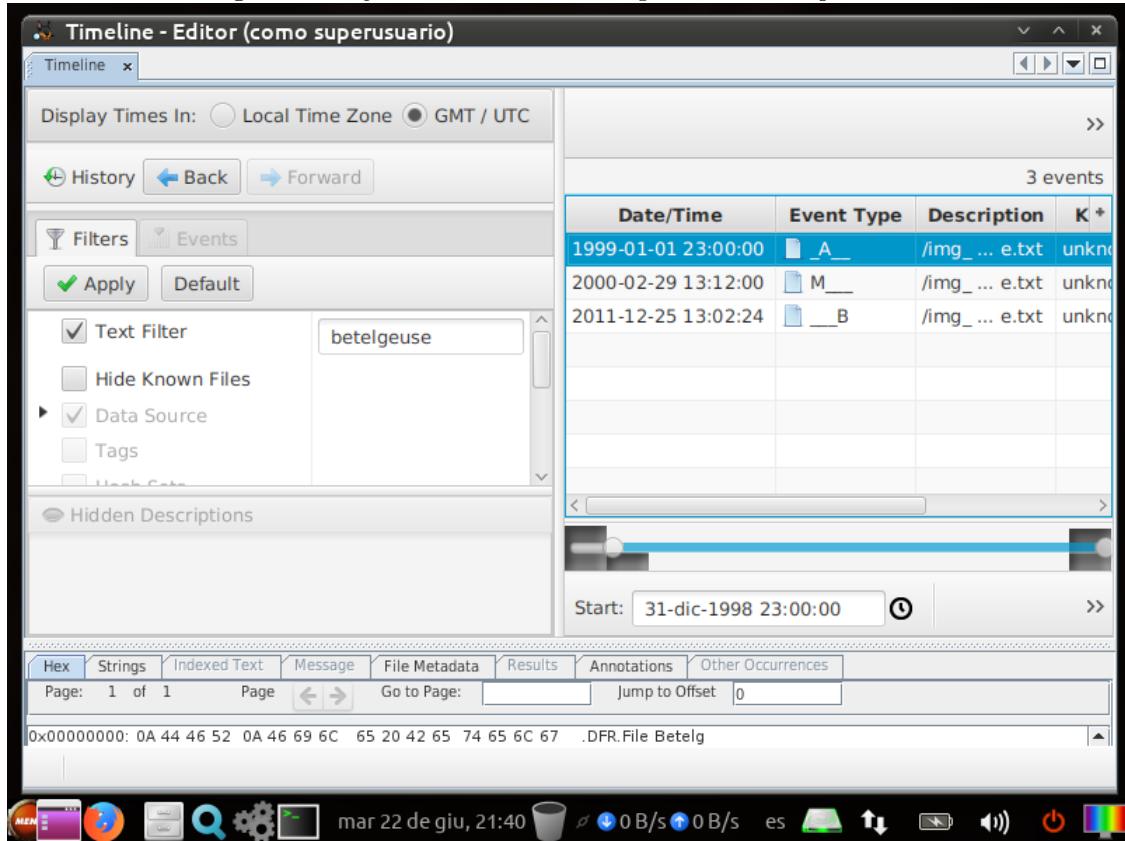


Figura 75: Ejercicio 12: Línea temporal de *Betelgeuse.txt*



13. Ejercicio 13

Se crea el caso en Autopsy con los datos solicitados.

Figura 76: Ejercicio 13: Creación del caso

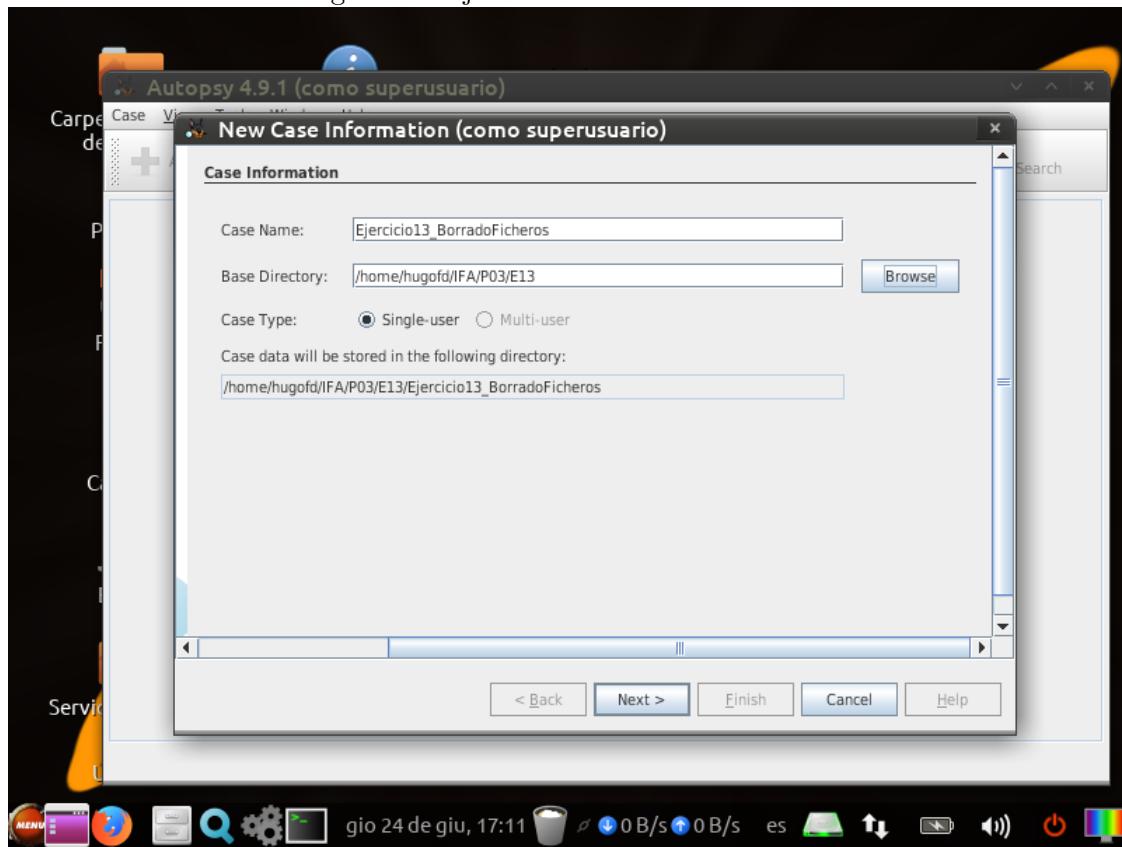
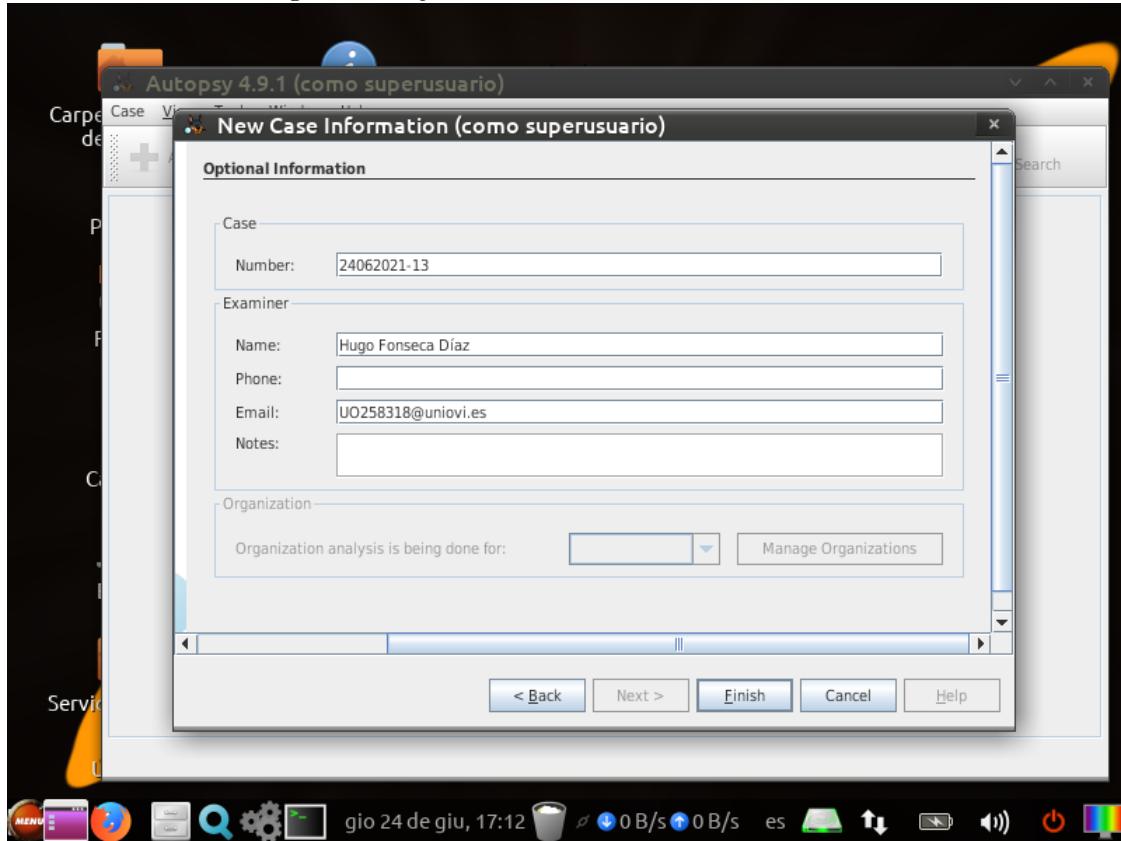
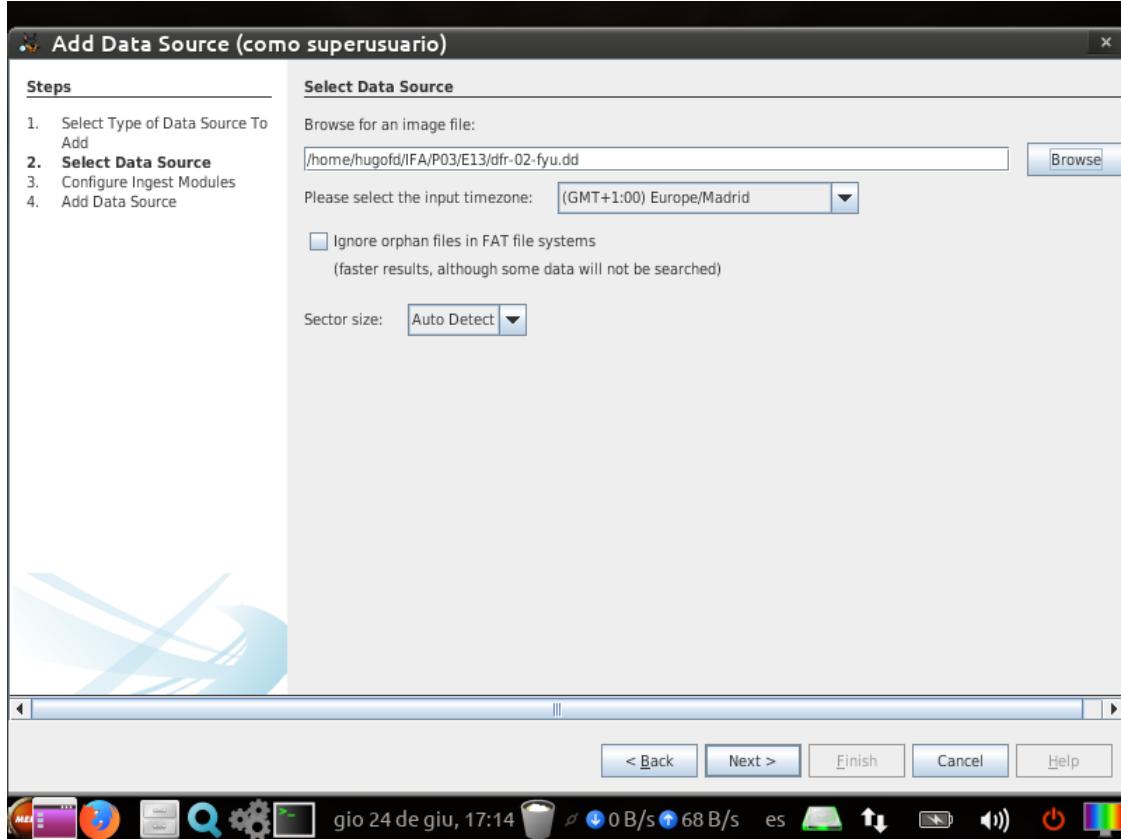


Figura 77: Ejercicio 13: Detalles del examinador



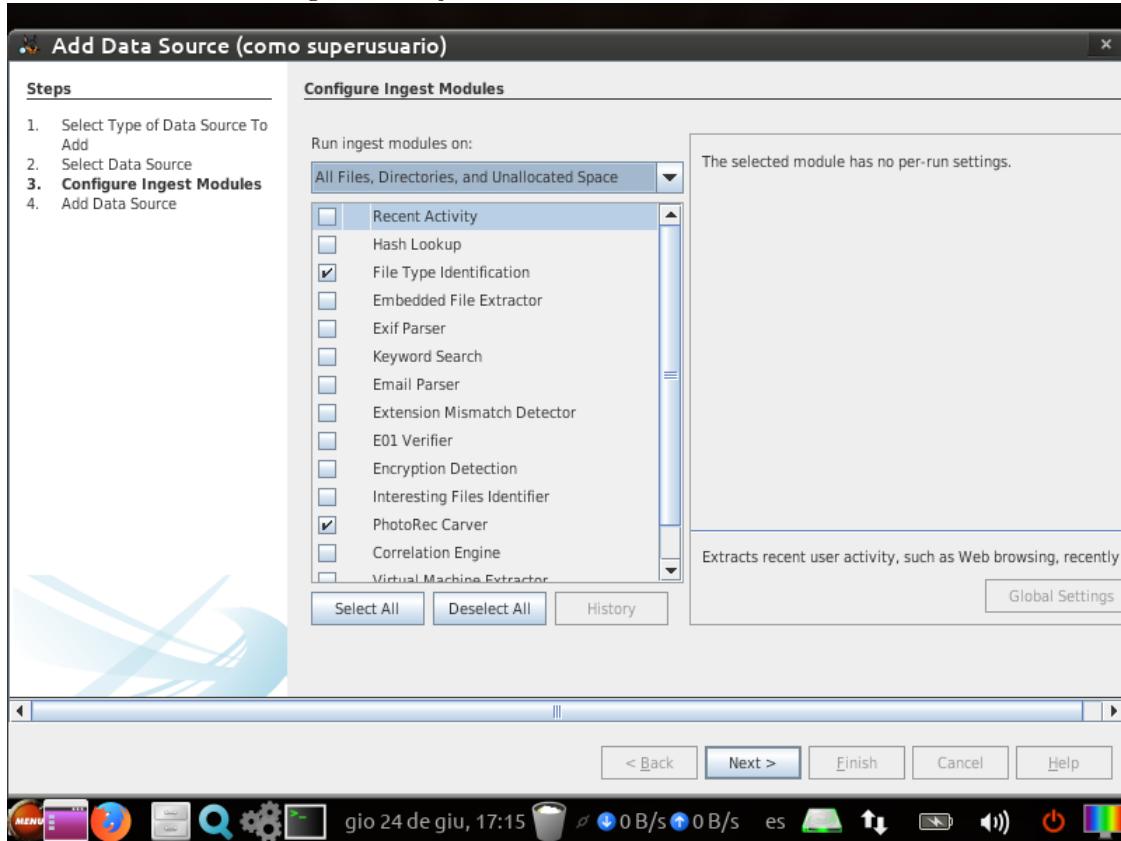
Añadimos la imagen a analizar.

Figura 78: Ejercicio 13: Selección de la imagen



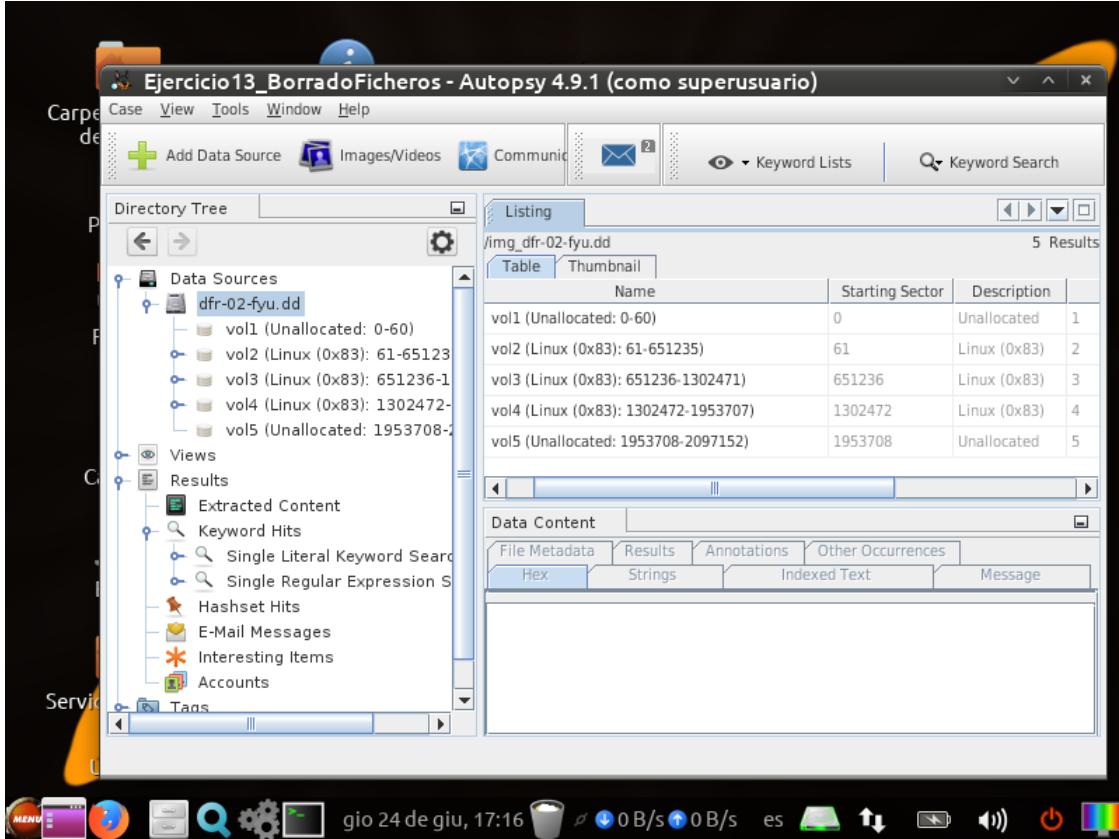
Se seleccionan los módulos de identificación de tipos de fichero y *PhotoRec Carver*.

Figura 79: Ejercicio 13: Selección de módulos



Se obtienen los resultados del análisis con los que se responderán a las diferentes cuestiones del ejercicio.

Figura 80: Ejercicio 13: Resultados del análisis



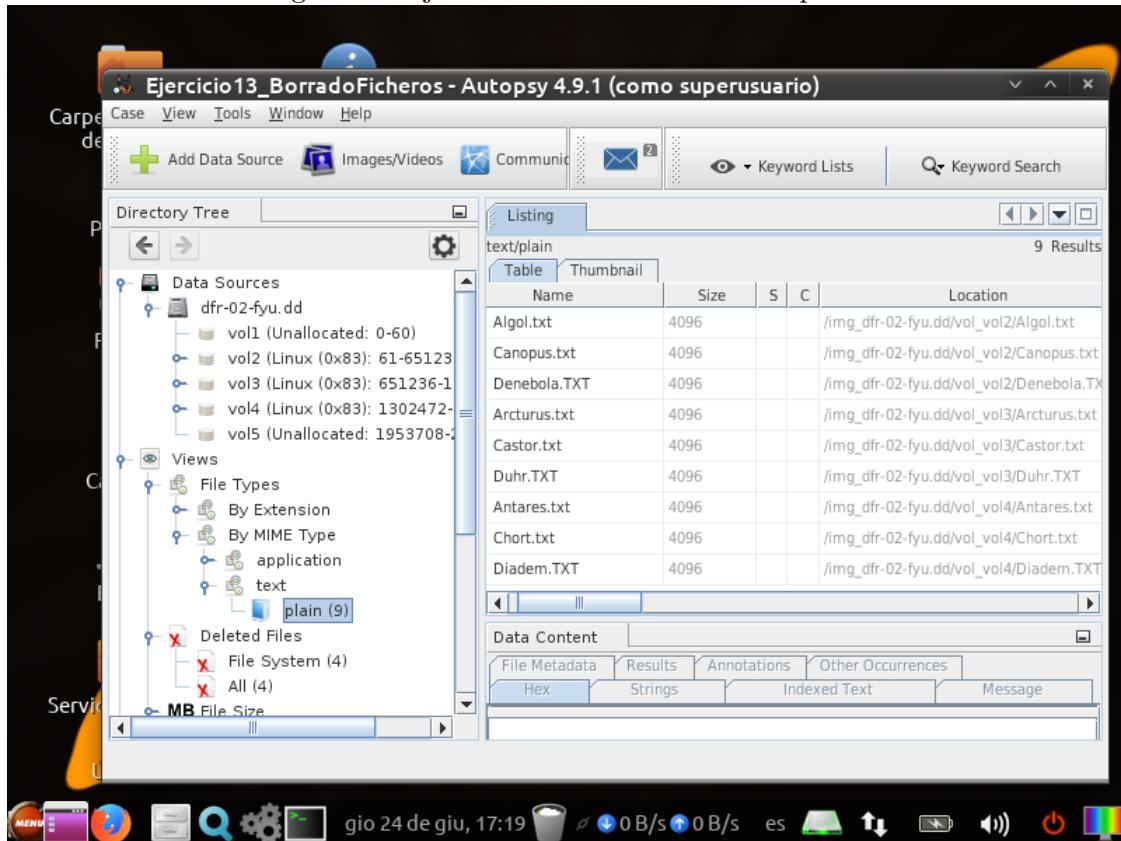
a)

TBD CAMBIAR TABLA!

| Número partición | Sector comienzo | Sector finalización | Tipo Sistema de Ficheros |
|------------------|-----------------|---------------------|--------------------------|
| 1 | 0 | 127 | Unallocated |
| 2 | 128 | 16511 | DOS FAT12 |
| 3 | 16512 | 82047 | DOS FAT16 |
| 4 | 82048 | 213119 | Win95 FAT32 |
| 5 | 213120 | 2097152 | Unallocated |

b) Para responder a esta cuestión se observan los resultados de la pestaña 'Views'.

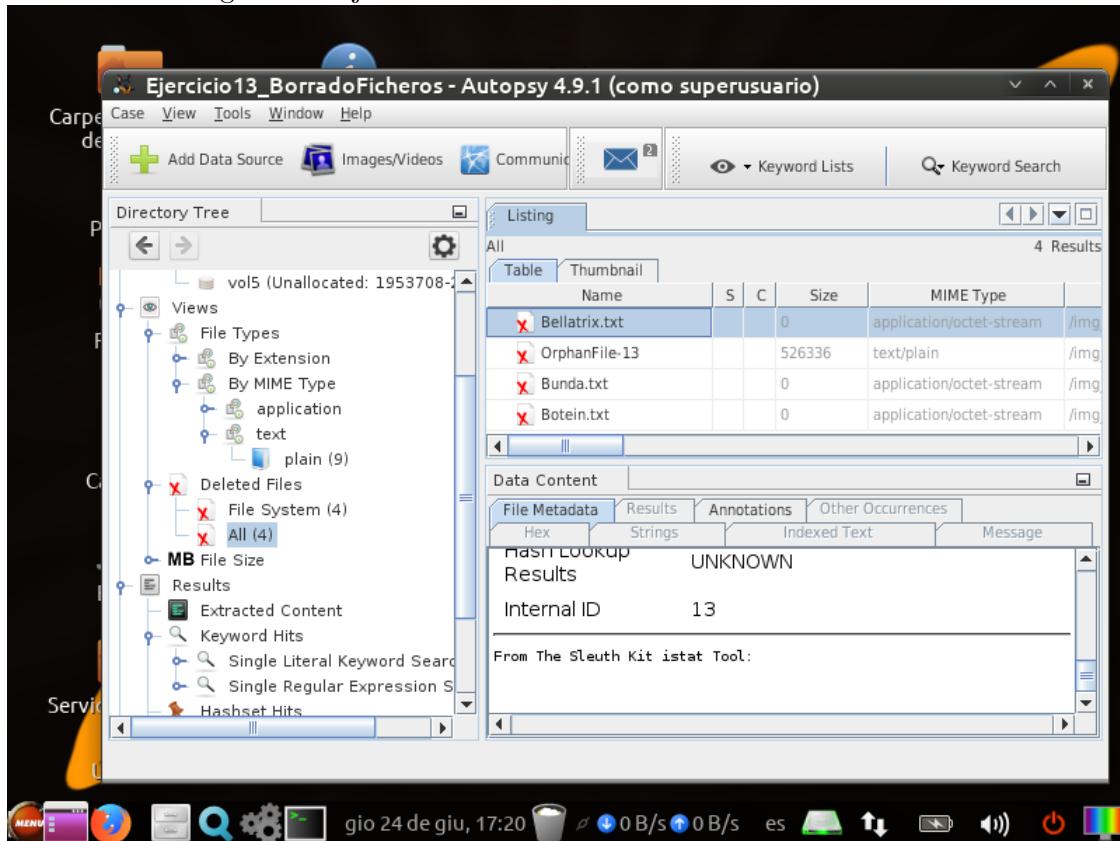
Figura 81: Ejercicio 13: Ficheros de texto plano



Se puede ver que hay 9 ficheros de texto plano. Hay 4 ficheros adicionales borrados, uno llamado Orphan-Files, el cual es autogenerado por Autopsy, y tres ficheros con extensión txt pero cuyos tipos MIME no son texto plano.

c) Para llenar esta tabla se miran los metadatos que muestra Autopsy de cada archivo borrado.

Figura 82: Ejercicio 13: Metadatos de los ficheros borrados



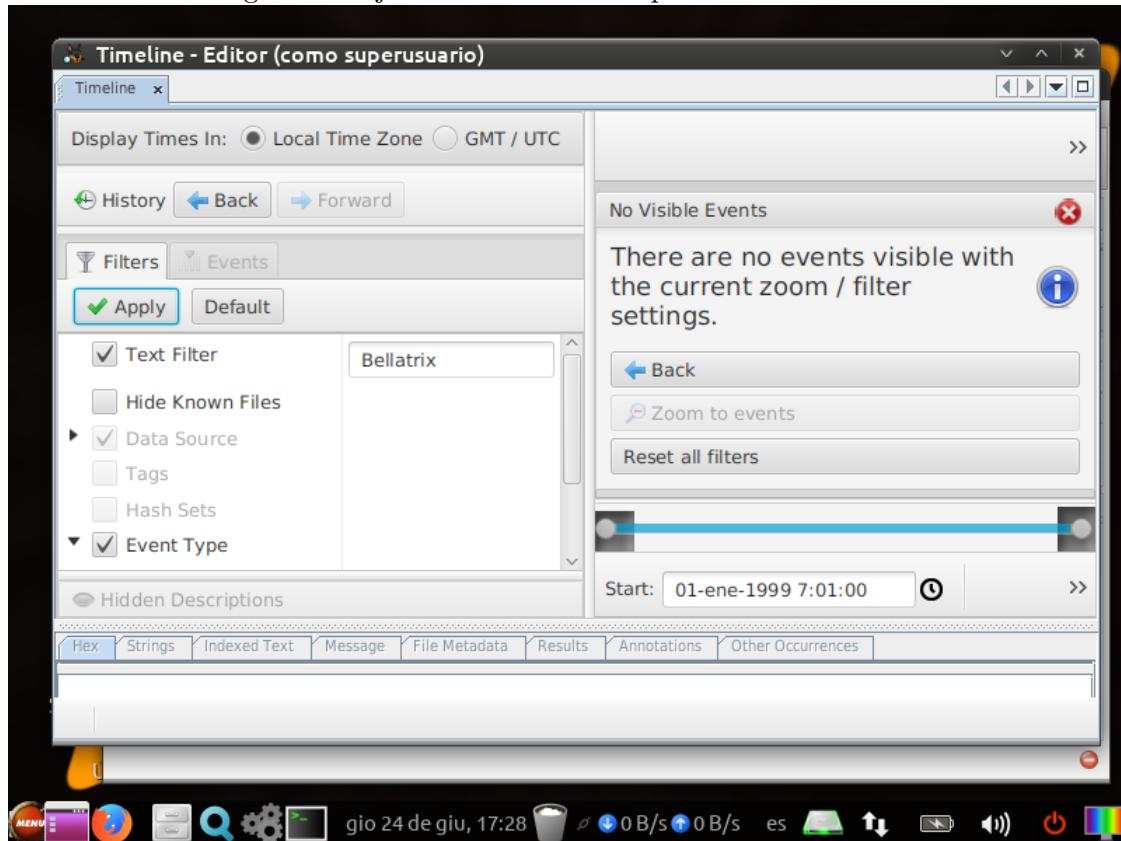
Como se puede observar hay menos metadatos sobre los ficheros borrados que en el ejercicio anterior, por lo que habrá secciones de la tabla sin rellenar.

TBD CAMBIAR TABLA!

| Nombre | Tamaño | Partición | Sector relativo | Acceso (GMT) | Modificación (GMT) | Creación (GMT) |
|----------------|--------|-----------|-----------------|------------------------|------------------------|------------------------|
| _BEID.txt | 712 | vol 2 | 170 | 1999/01/01 23:00:00 | 2000/02/29 13:11:00 | 2011/12/25 13:02:22 |
| Betelgeuse.txt | 712 | vol 3 | 546 | 1999/01/01 23:00:00 | 2000/02/29 13:12:00 | 2011/12/25 13:02:24 |
| Bellatrix.txt | 712 | vol 4 | 8195 | 1999/01/01 23:00:00 | 2000/02/29 13:13:00 | 2011/12/25 13:02:24 |

- d) Se muestran a continuación las líneas de tiempo de los tres ficheros borrados, en el filtro de la parte izquierda de la captura se observa el fichero actual.

Figura 83: Ejercicio 13: Línea temporal de *Bellatrix.txt*



Se observa que no hay datos para *Bellatrix.txt*

Figura 84: Ejercicio 13: Línea temporal de *Bunda.txt*

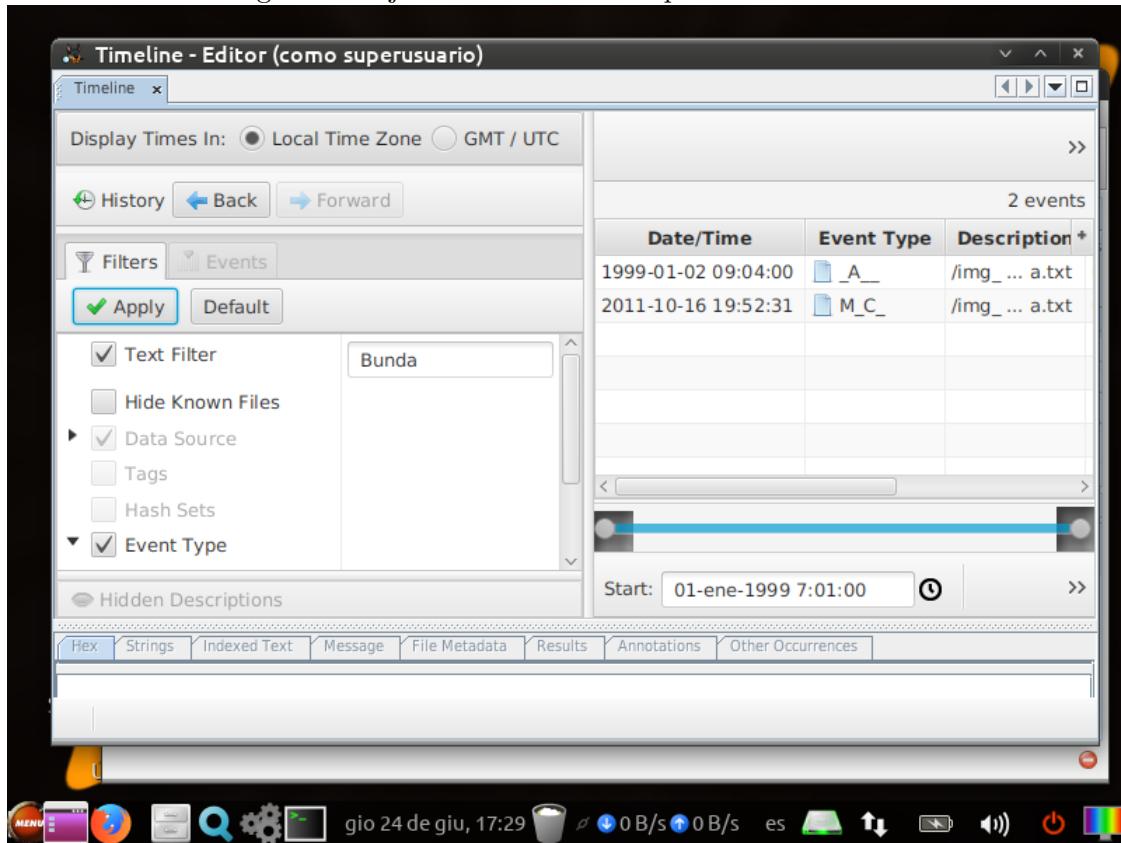
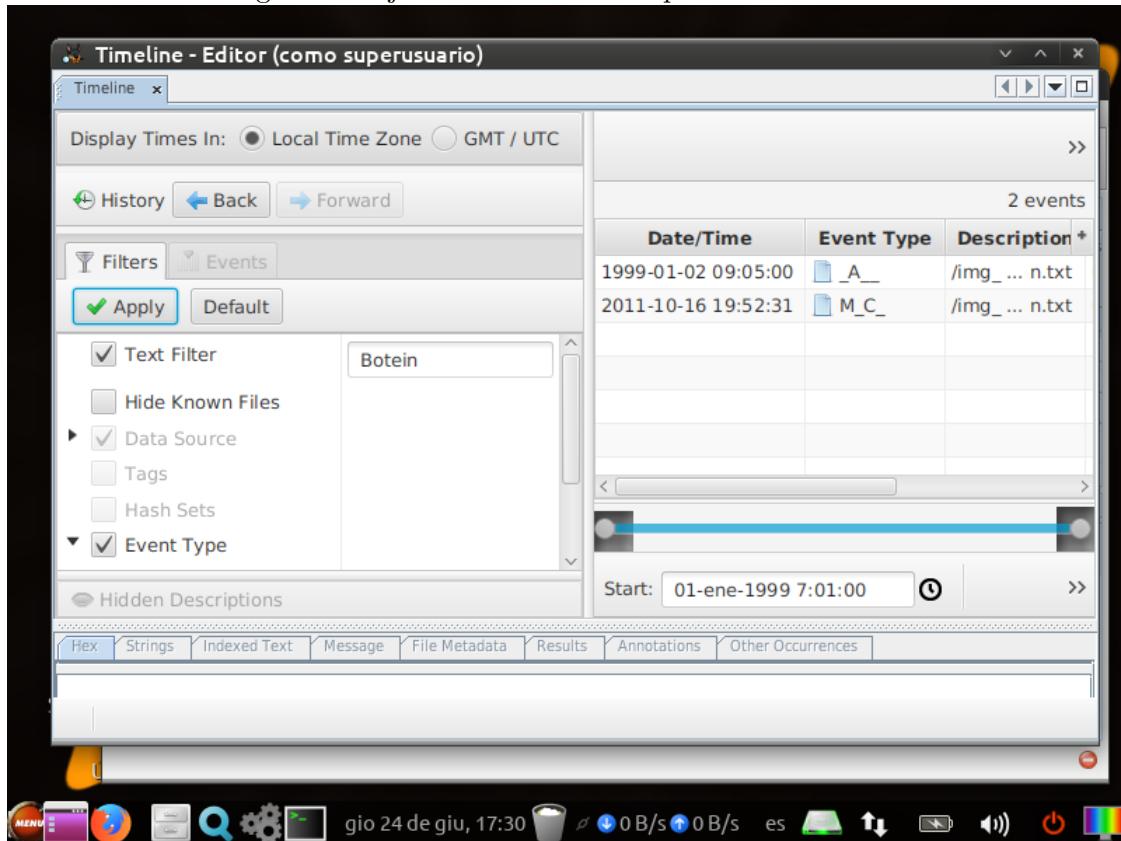


Figura 85: Ejercicio 13: Línea temporal de *Botein.txt*



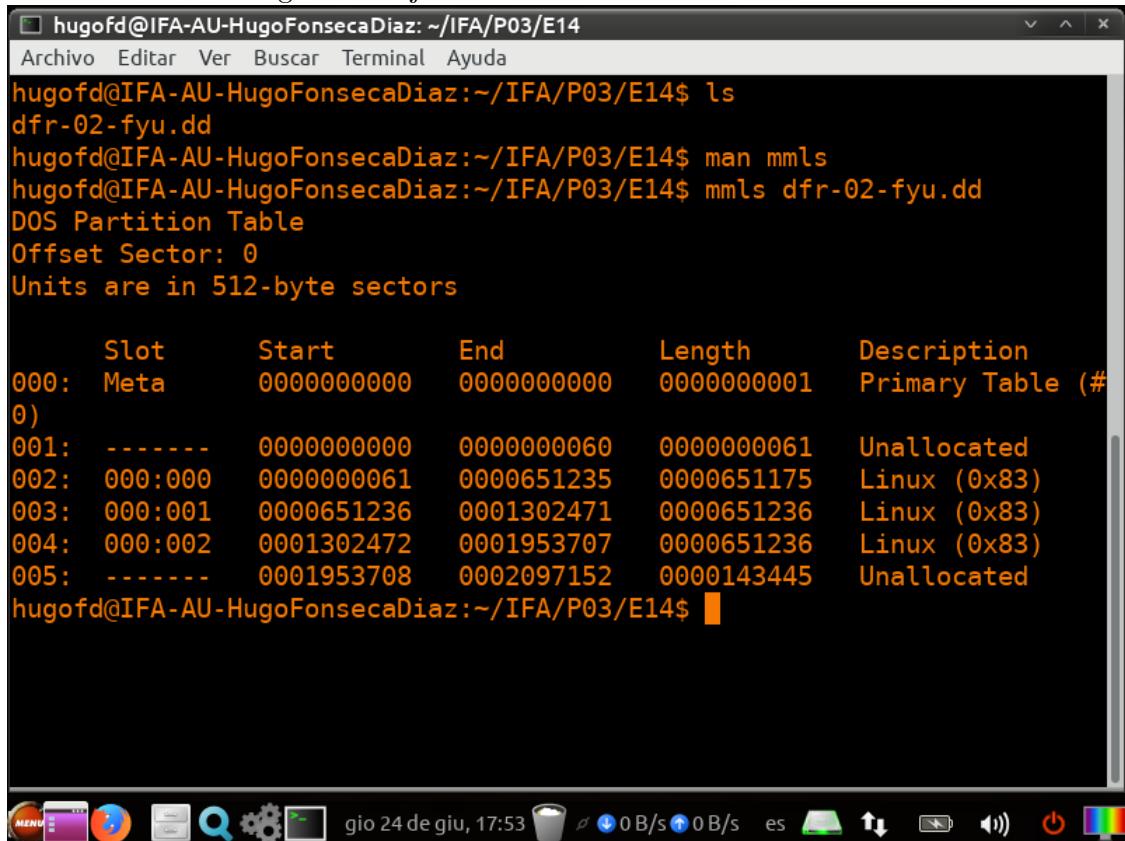
Para *Bunda.txt* y *Botein.txt* sí que se recuperan datos.

14. Ejercicio 14

Se responde a continuación a las diferentes cuestiones planteadas por el ejercicio.

- Se utiliza el comando `mmls`, que lista las particiones con sus sectores de inicio y fin, entre otros datos.

Figura 86: Ejercicio 14: Salida del comando *mmls*



The screenshot shows a terminal window with the following content:

```
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ ls
dfr-02-fyu.dd
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ man mmls
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ mmls dfr-02-fyu.dd
DOS Partition Table
Offset Sector: 0
Units are in 512-byte sectors

      Slot      Start        End        Length     Description
000: Meta    0000000000  0000000000  0000000001  Primary Table (#0)
001: -----  0000000000  0000000060  0000000061  Unallocated
002: 000:000  0000000061  0000651235  0000651175  Linux (0x83)
003: 000:001  0000651236  0001302471  0000651236  Linux (0x83)
004: 000:002  0001302472  0001953707  0000651236  Linux (0x83)
005: -----  0001953708  0002097152  0000143445  Unallocated
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$
```

The terminal window has a menu bar with Archivo, Editar, Ver, Buscar, Terminal, Ayuda. The desktop environment icons at the bottom include MENU, Home, Task Manager, Search, Settings, and others.

- b) Sí, la información es consistente entre ambas herramientas.
- c) Se usa el comando **fsstat**, con la flag *t* para mostrar solo el tipo de partición y la flag *o* para pasarle al comando el sector donde comienza la partición.

Figura 87: Ejercicio 14: Salida del comando *fsstat* para las diferentes particiones

The screenshot shows a terminal window titled "hugofd@IFA-AU-HugoFonsecaDiaz: ~/IFA/P03/E14". The window contains the following text:

```
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ mmls dfr-02-fyu.dd
DOS Partition Table
Offset Sector: 0
Units are in 512-byte sectors

   Slot      Start        End      Length     Description
000: Meta    0000000000  0000000000  0000000001 Primary Table (#0)
001: -----  0000000000  0000000060  0000000061 Unallocated
002: 000:000  0000000061  0000651235  0000651175 Linux (0x83)
003: 000:001  0000651236  0001302471  0000651236 Linux (0x83)
004: 000:002  0001302472  0001953707  0000651236 Linux (0x83)
005: -----  0001953708  0002097152  0000143445 Unallocated

hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ man fsstat
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ fsstat -t -o 61 dfr-02-fyu.dd
ext2
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ fsstat -t -o 651236 dfr-02-fyu.dd
ext3
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ fsstat -t -o 1302472 dfr-02-fyu.dd
ext4
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$
```

The terminal window has a menu bar with Archivo, Editar, Ver, Buscar, Terminal, Ayuda. The status bar at the bottom shows the date and time (gio 24 de giu, 17:59), battery level (0 B/s), signal strength (0 B/s), and system status (es). The desktop icons visible at the bottom include MENU, a purple folder, a red circular icon, a grey document, a magnifying glass, a gear, and a square.

- d) Se utiliza el comando **f1s** que recibe como argumentos, entre otros, el comienzo del sector de la partición que se quiere analizar.

Figura 88: Ejercicio 14: Salida del comando `f1s` con las flags *ro*

The screenshot shows a terminal window titled "hugofd@IFA-AU-HugoFonsecaDiaz: ~/IFA/P03/E14". The window contains the following text:

```
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ man f1s
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ mm1s dfr-02-fyu.dd
DOS Partition Table
Offset Sector: 0
Units are in 512-byte sectors

      Slot      Start        End        Length     Description
000: Meta    0000000000  0000000000  0000000001 Primary Table (#0)
001: -----  0000000000  0000000060  0000000061 Unallocated
002: 000:000  0000000061  0000651235  0000651175 Linux (0x83)
003: 000:001  0000651236  0001302471  0000651236 Linux (0x83)
004: 000:002  0001302472  0001953707  0000651236 Linux (0x83)
005: -----  0001953708  0002097152  0000143445 Unallocated
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ f1s -o 1302472 -r dfr-02-fyu.dd
d/d 11: lost+found
r/r 12: Antares.txt
r/r * 13:      Botein.txt
r/r 14: Chort.txt
r/r 15: Diadem.TXT
V/V 81601:      $OrphanFiles
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$
```

The terminal window has a dark background and light-colored text. Below the window, there is a dock with various icons and system status information.

- e) Se usa ahora el comando `f1s` con las flags *dFrO*, *d* muestra solo elementos borrados, *F* muestra solo ficheros, *r* es para que la búsqueda sea recursiva y *o* para introducir el comienzo del sector de la partición.

Figura 89: Ejercicio 14: Salida del comando *fls* con las flags *dFrO*

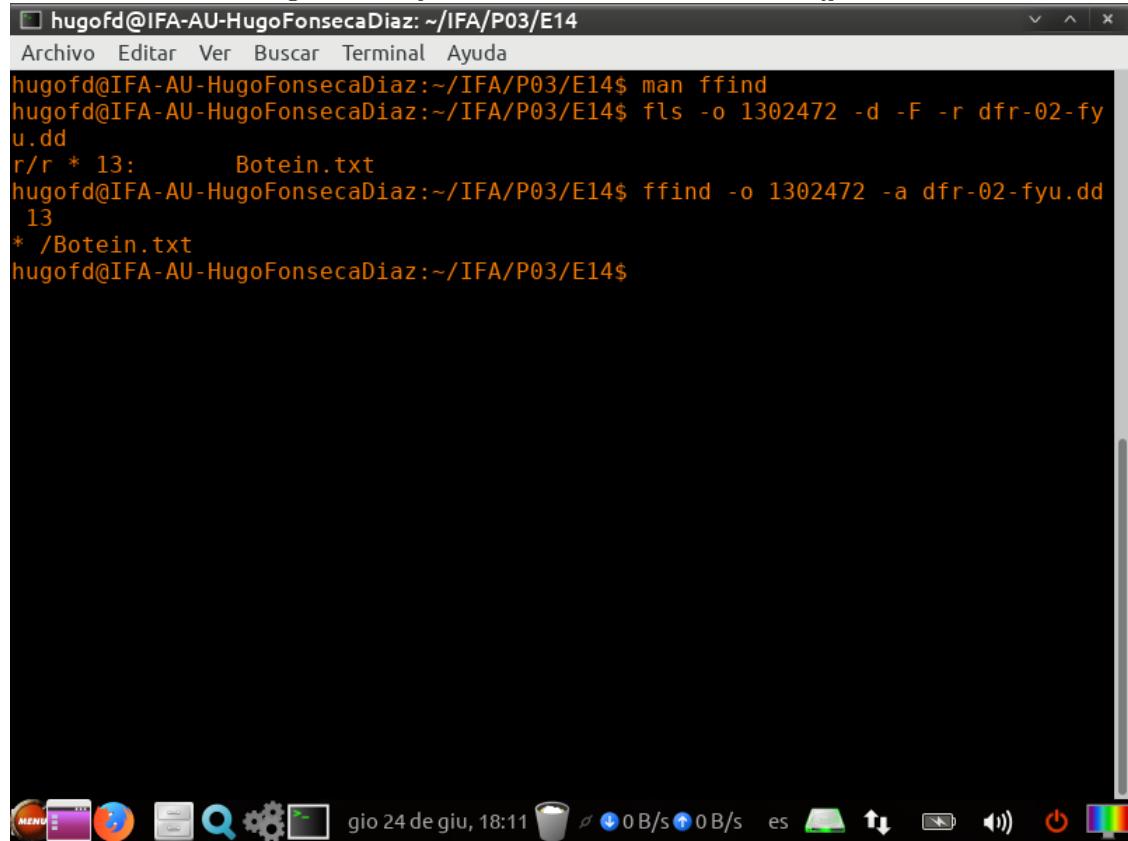
The screenshot shows a terminal window titled "hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14". The window contains the following text:

```
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ man fls
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ fls -o 1302472 -d -F -r dfr-02-fy
u.dd
r/r * 13:      Botein.txt
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ █
```

The terminal is running on a desktop environment, as evidenced by the taskbar icons at the bottom, which include a menu, a file manager, a browser, a terminal, a search function, system settings, and a power button.

- f) Se utiliza el comando **ffind** con las flags *oa*, *o* para introducir el comienzo del sector de la partición y *a* para buscar todos los ficheros asociados. Se le pasa al comando el inodo del elemento que se está buscando, en este caso el 13.

Figura 90: Ejercicio 14: Salida del comando *ffind*



The screenshot shows a terminal window titled "hugofd@IFA-AU-HugoFonsecaDiaz: ~/IFA/P03/E14". The window contains the following text:

```
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ man ffind
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ fls -o 1302472 -d -F -r dfr-02-fy
u.dd
r/r * 13:      Botein.txt
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ ffind -o 1302472 -a dfr-02-fyu.dd
 13
* /Botein.txt
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$
```

The terminal window has a dark background and light-colored text. At the bottom, there is a dock with various icons, including a trash can, a search bar, and system status indicators like battery level and signal strength.

g) Se usa el comando *istat* pasandole como argumento el comienzo del sector de la partición y el inodo a buscar.

Figura 91: Ejercicio 14: Salida del comando *istat* para el inodo 13

The screenshot shows a terminal window titled "hugofd@IFA-AU-HugoFonsecaDiaz: ~/IFA/P03/E14". The window contains the following text:

```
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ man istat
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ fls -o 1302472 -d -F -r dfr-02-fy
u.dd
r/r * 13:      Botein.txt
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ istat -o 1302472 dfr-02-fyu.dd 13
inode: 13
Not Allocated
Group: 0
Generation Id: 2392951179
uid / gid: 0 / 0
mode: rrw-----
Flags: Extents,
size: 0
num of links: 0

Extended Attributes (Block: 4386)
security.selinux=unconfined_u:object_r:file_t:s0

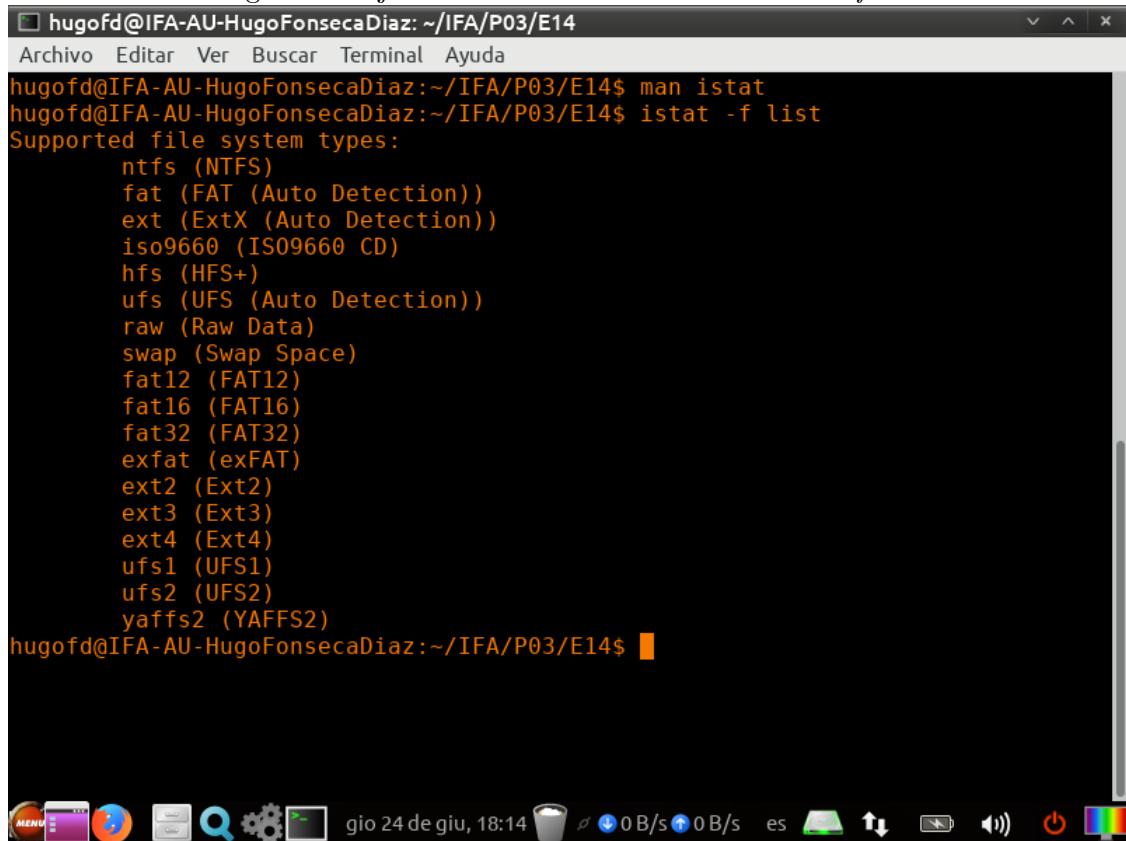
Inode Times:
Accessed: 1999-01-02 09:05:00 (CET)
File Modified: 2011-10-16 19:52:31 (CEST)
Inode Modified: 2011-10-16 19:52:31 (CEST)
Deleted: 2011-10-16 19:52:31 (CEST)

Direct Blocks:
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$
```

The terminal window has a dark background with light-colored text. At the bottom, there is a standard Linux desktop dock with icons for various applications like a menu, file manager, browser, terminal, and system settings. The date and time "gio 24 de giu, 18:13" are also visible at the bottom.

h) Se usa el comando *istat* con la flag *f* y el argumento *list*.

Figura 92: Ejercicio 14: Salida del comando *istat -f list*

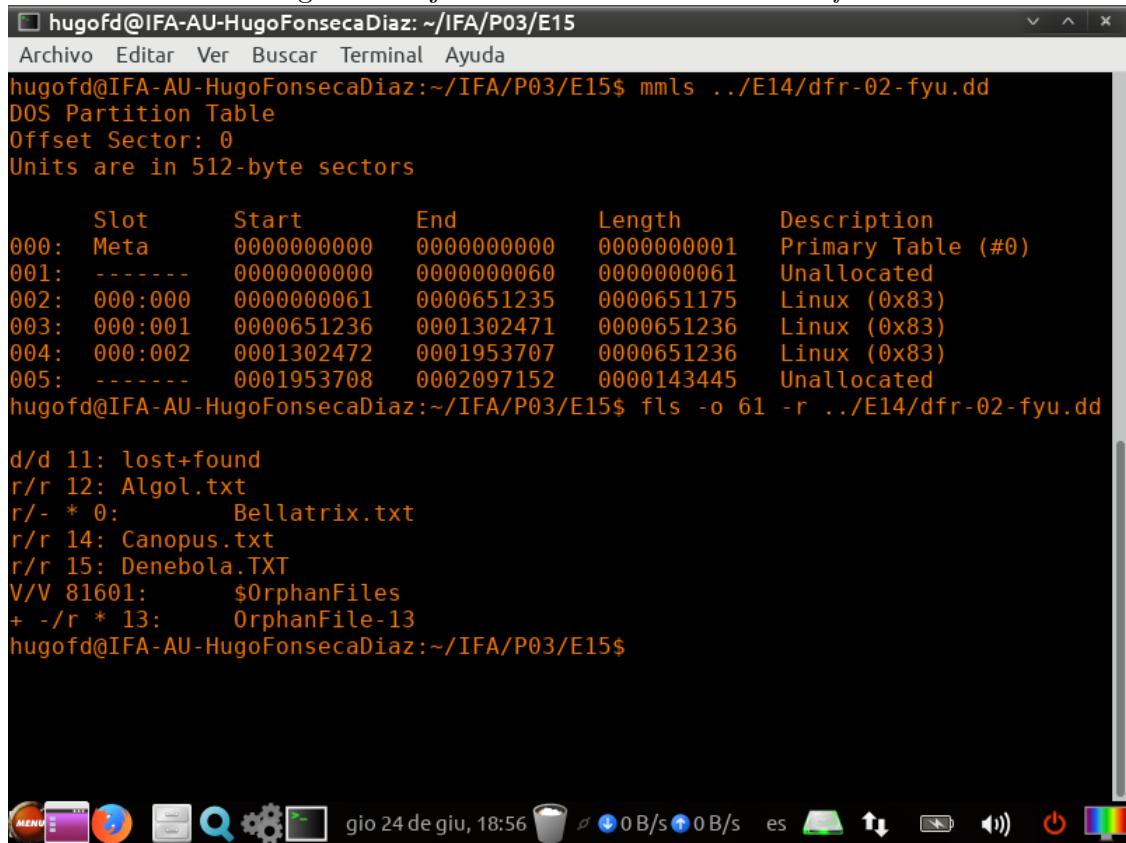


```
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ man istat
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$ istat -f list
Supported file system types:
    ntfs (NTFS)
    fat (FAT (Auto Detection))
    ext (ExtX (Auto Detection))
    iso9660 (ISO9660 CD)
    hfs (HFS+)
    ufs (UFS (Auto Detection))
    raw (Raw Data)
    swap (Swap Space)
    fat12 (FAT12)
    fat16 (FAT16)
    fat32 (FAT32)
    exfat (exFAT)
    ext2 (Ext2)
    ext3 (Ext3)
    ext4 (Ext4)
    ufs1 (UFS1)
    ufs2 (UFS2)
    yaffs2 (YAFFS2)
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E14$
```

15. Ejercicio 15

Se busca obtener el inodo del fichero huérfano de la partición dos. Para ello, se utiliza el comando **f1s** con las flags *ro*, *r* para que la búsqueda sea recursiva y *o* para recibir el sector de inicio de la partición.

Figura 93: Ejercicio 15: Salida del comando *fls*



The screenshot shows a terminal window titled "hugofd@IFA-AU-HugoFonsecaDiaz: ~/IFA/P03/E15". The window contains the following text:

```
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E15$ mmls ../E14/dfr-02-fyu.dd
DOS Partition Table
Offset Sector: 0
Units are in 512-byte sectors

   Slot      Start        End      Length     Description
000: Meta    0000000000  0000000000  0000000001 Primary Table (#0)
001: -----  0000000000  0000000060  0000000061 Unallocated
002: 000:000  0000000061  0000651235  0000651175 Linux (0x83)
003: 000:001  0000651236  0001302471  0000651236 Linux (0x83)
004: 000:002  0001302472  0001953707  0000651236 Linux (0x83)
005: -----  0001953708  0002097152  0000143445 Unallocated

hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E15$ fls -o 61 -r ../E14/dfr-02-fyu.dd

d/d 11: lost+found
r/r 12: Algol.txt
r/- * 0: Bellatrix.txt
r/r 14: Canopus.txt
r/r 15: Denebola.TXT
V/V 81601: $OrphanFiles
+ -/r * 13: OrphanFile-13
hugofd@IFA-AU-HugoFonsecaDiaz:~/IFA/P03/E15$
```

The terminal window has a dark background and light-colored text. At the bottom, there is a toolbar with various icons and system status information.

Una vez se tiene el inodo del fichero huérfano, en este caso el inodo 13, se usa el comando *icat* con las flags *or*, *o* para indicar el sector de comienzo y *r* para recuperar el fichero en caso de que este borrado. Se le indica al comando el inodo a buscar y se redirige la salida del comando a un fichero de texto.

Figura 94: Ejercicio 14: Salida del comando `icat` redirigida a un fichero de texto plano

16. Ejercicio 16

Se crea el caso en Autopsy con los datos solicitados.

Figura 95: Ejercicio 16: Creación del caso

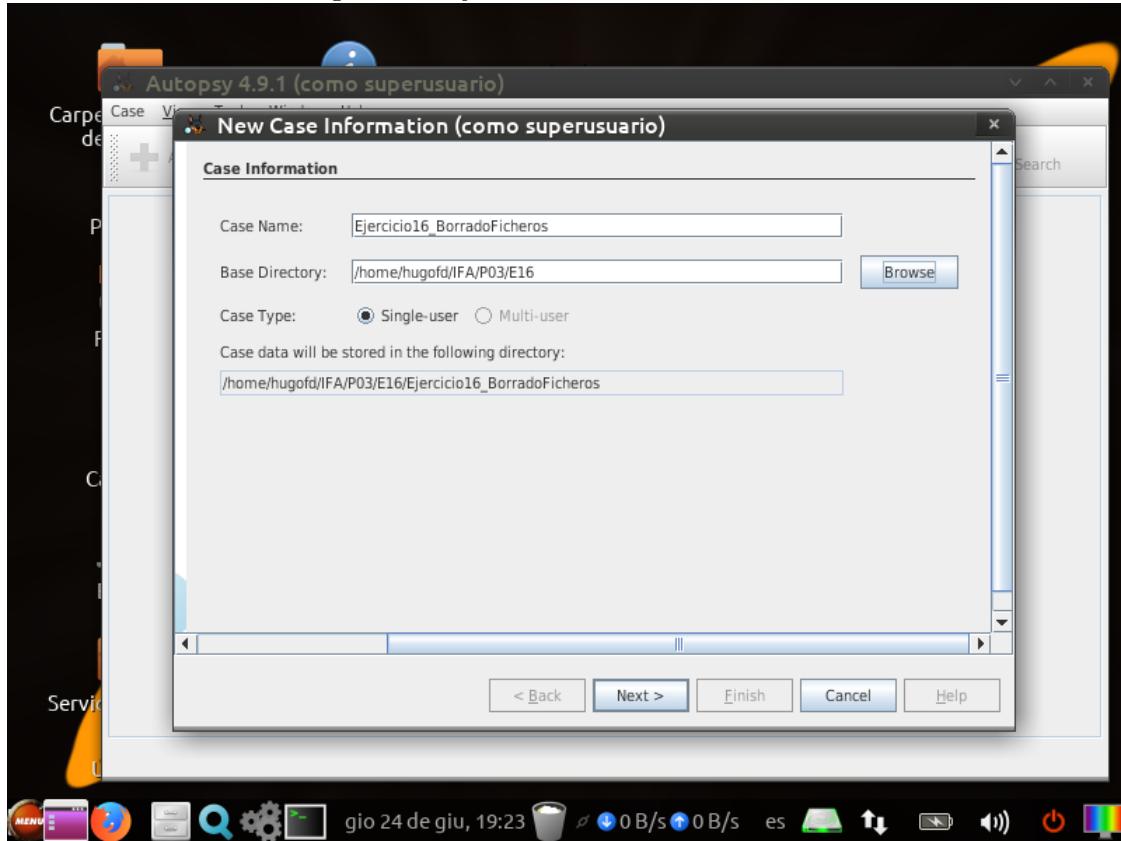
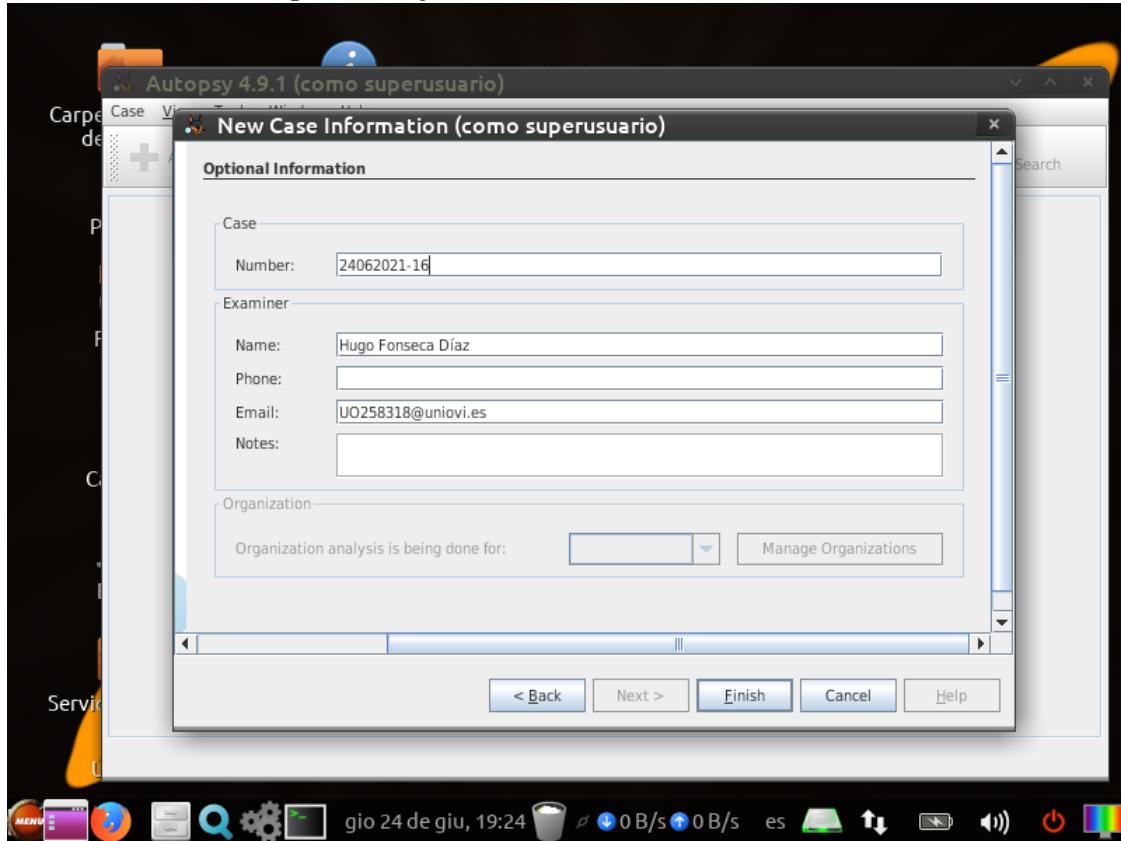
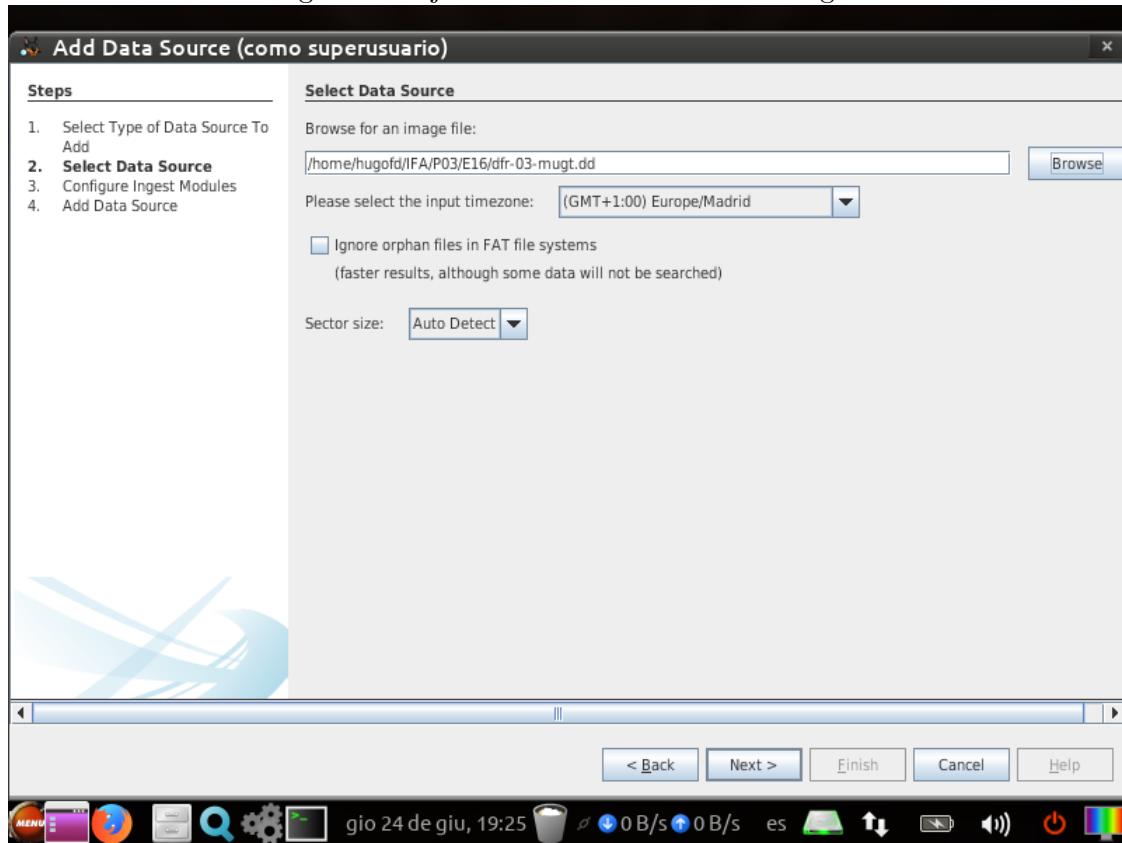


Figura 96: Ejercicio 16: Detalles del examinador



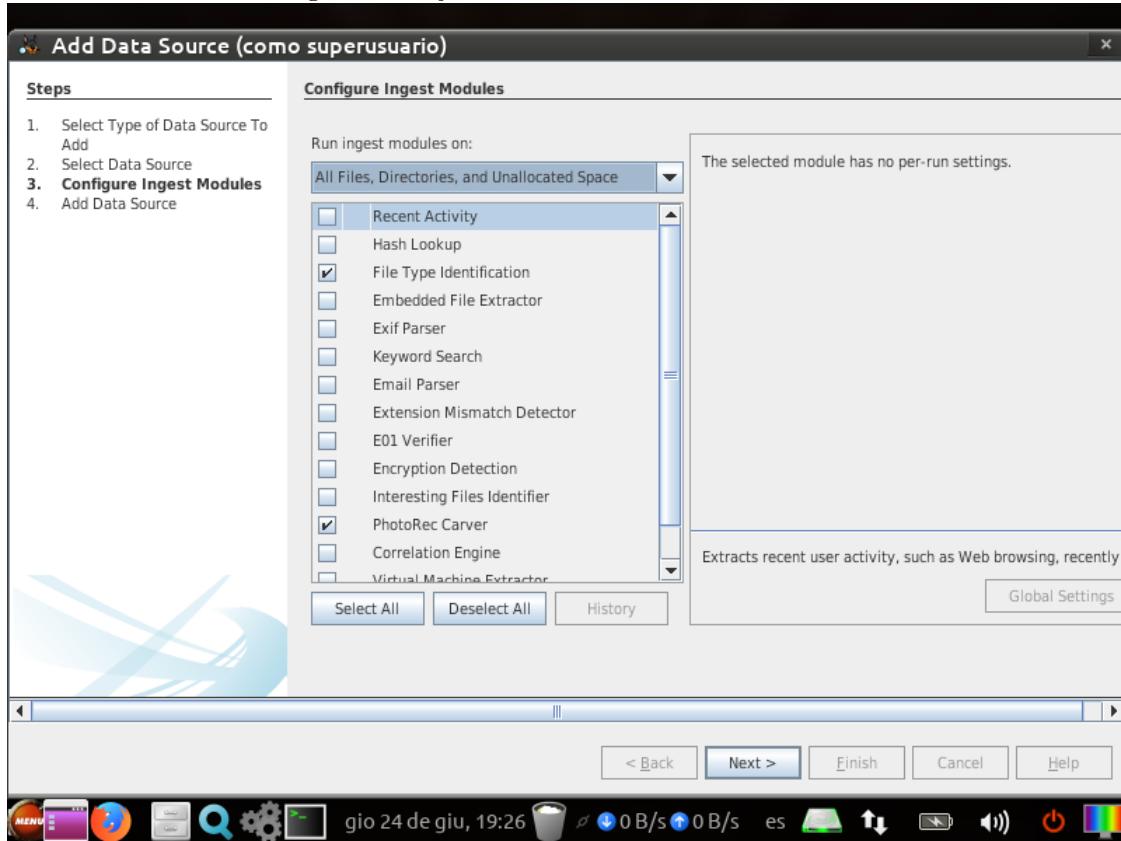
Añadimos la imagen a analizar.

Figura 97: Ejercicio 16: Selección de la imagen



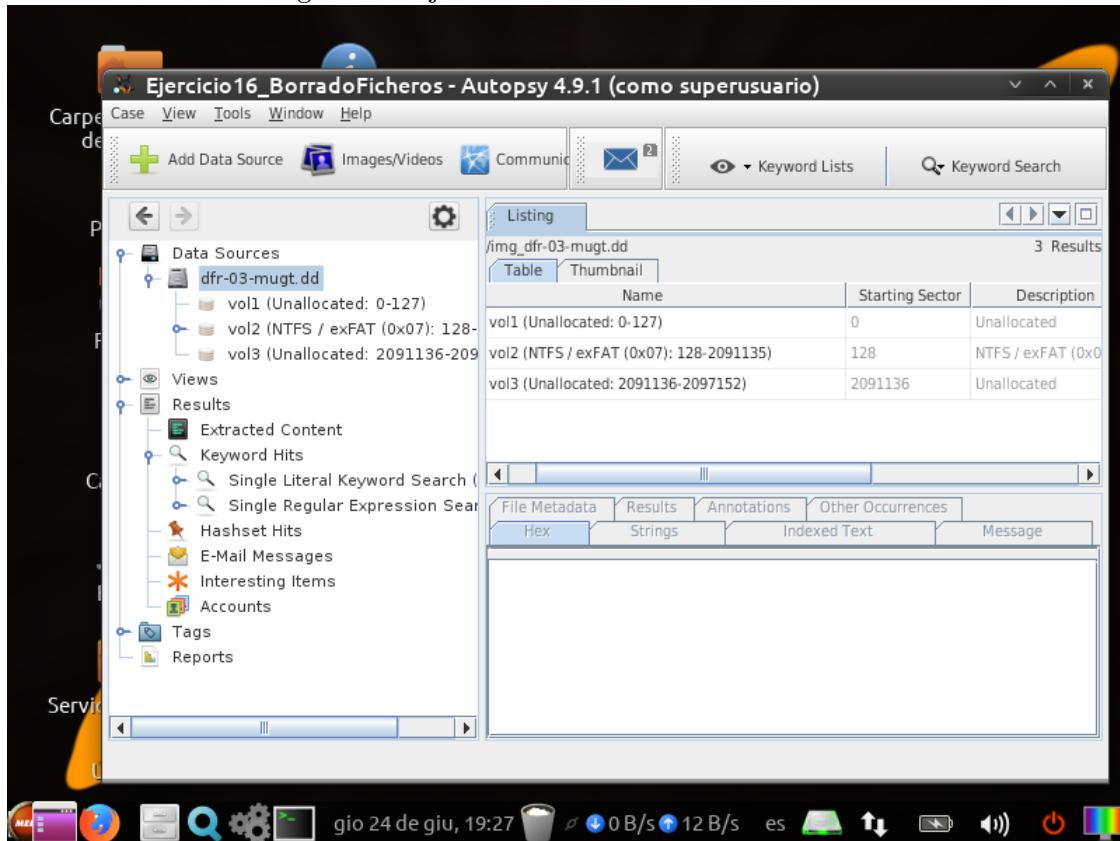
Se seleccionan los módulos de identificación de tipos de fichero y *PhotoRec Carver*.

Figura 98: Ejercicio 16: Selección de módulos



Se obtienen los resultados del análisis con los que se responderán a las diferentes cuestiones del ejercicio.

Figura 99: Ejercicio 16: Resultados del análisis

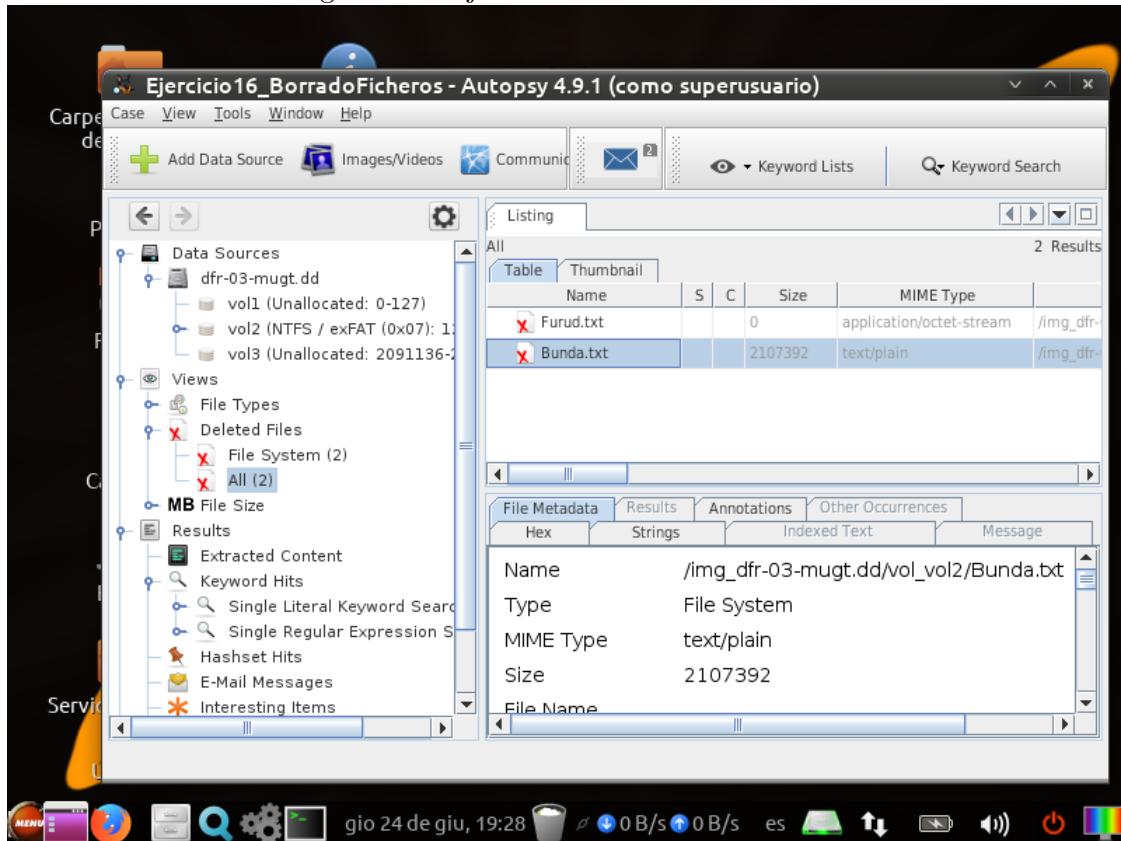


TBD CAMBIAR TABLA!

| Número partición | Sector comienzo | Sector finalización | Tipo Sistema de Ficheros |
|------------------|-----------------|---------------------|--------------------------|
| 1 | 0 | 127 | Unallocated |
| 2 | 128 | 16511 | DOS FAT12 |
| 3 | 16512 | 82047 | DOS FAT16 |
| 4 | 82048 | 213119 | Win95 FAT32 |
| 5 | 213120 | 2097152 | Unallocated |

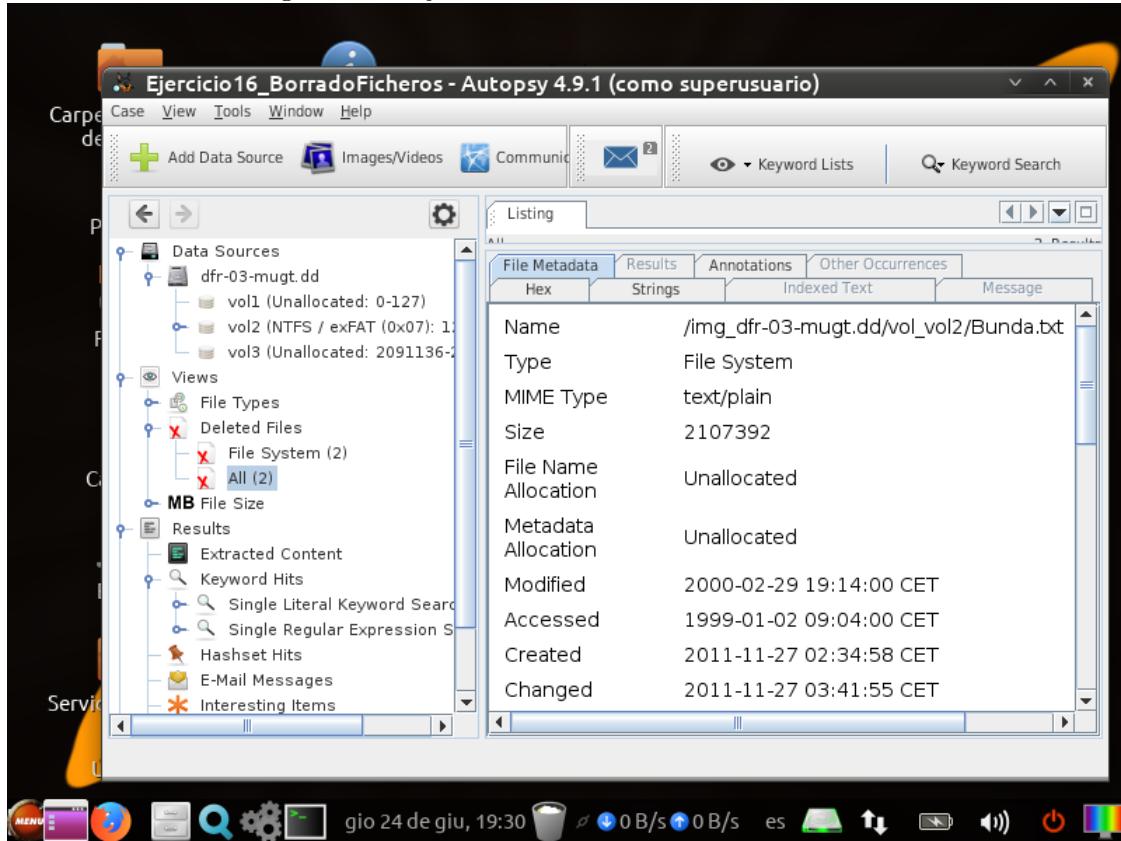
- a) Hay dos ficheros borrados pero solo *Bunda.txt* tiene el tipo MIME texto plano.

Figura 100: Ejercicio 16: Ficheros borrados



b) Se observan los metadatos del fichero *Bunda.txt*.

Figura 101: Ejercicio 16: Metadatos de *Bunda.txt*

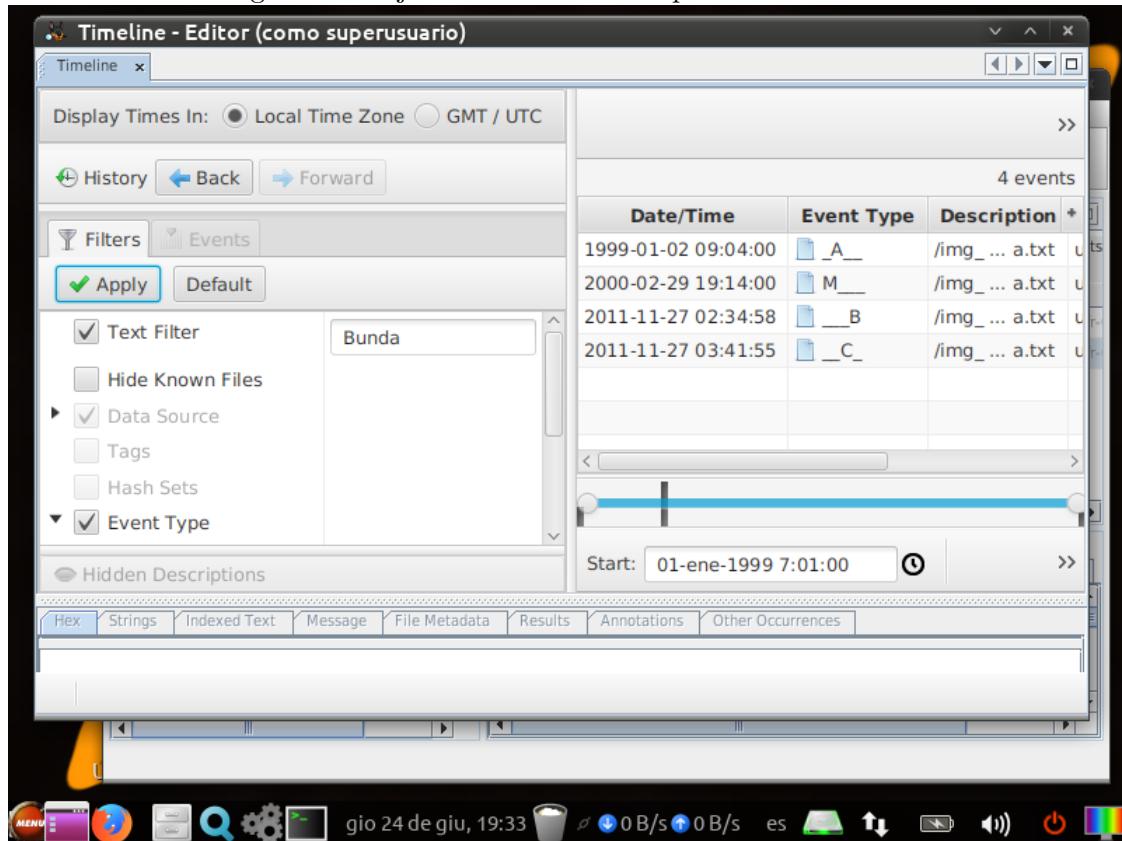


TBD CAMBIAR TABLA!

| Nombre | Tamaño | Partición | Sector relativo | Acceso (GMT) | Modificación (GMT) | Creación (GMT) |
|----------------|--------|-----------|-----------------|---------------------|---------------------|---------------------|
| _BEID.txt | 712 | vol 2 | 170 | 1999/01/01 23:00:00 | 2000/02/29 13:11:00 | 2011/12/25 13:02:22 |
| Betelgeuse.txt | 712 | vol 3 | 546 | 1999/01/01 23:00:00 | 2000/02/29 13:12:00 | 2011/12/25 13:02:24 |
| Bellatrix.txt | 712 | vol 4 | 8195 | 1999/01/01 23:00:00 | 2000/02/29 13:13:00 | 2011/12/25 13:02:24 |

c) Se muestra a continuación la línea temporal de *Bunda.txt*

Figura 102: Ejercicio 16: Línea temporal de *Bunda.txt*



Referencias