Particiones

Para la realización de esta práctica, tuvimos que agregar un disco duro externo, como se muestra en la figura 1 y 2.

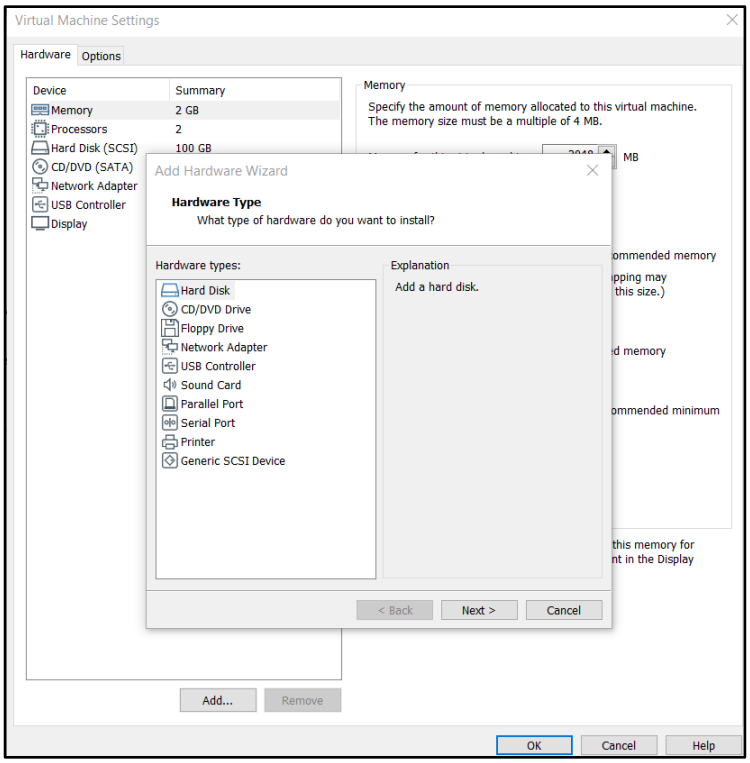

Figura 1. Agregar disco duro externo.

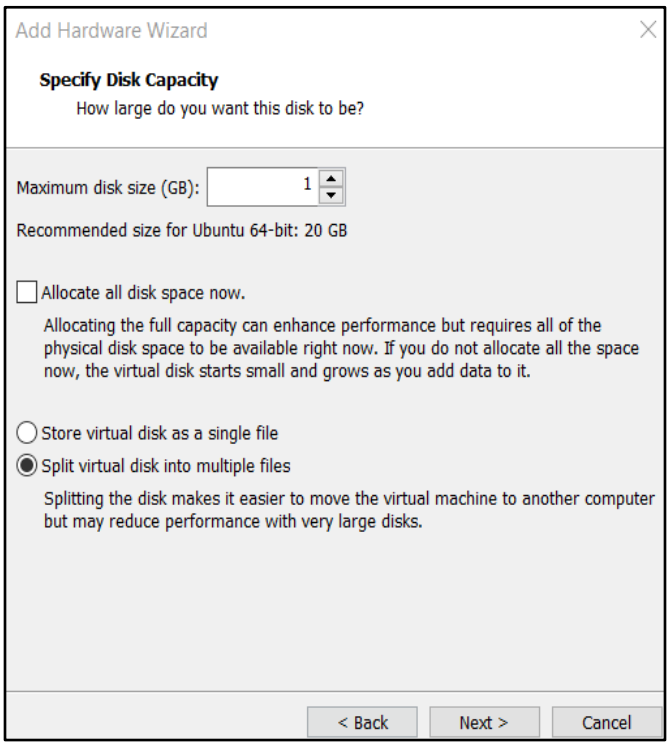

Figura 2. Tamaño del disco duro externo.

En este caso se creó un disco duro externo de 1 Gb., ahora listamos las particiones con el comando *fdisk -l,* como se muestra en la figura 3, observamos que aparecerá vacío.

```
root@ignacio-leal -> /h/sansforensics
# fdisk -l
Disk /dev/sdb: 1 GiB, 1073741824 bytes, 2097152 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/sda: 100 GiB, 107374182400 bytes, 209715200 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xfb53e2d2

Device     Boot     Start       End   Sectors  Size Id Type
/dev/sda1  *         2048 168454143 168452096 80.3G 83 Linux
/dev/sda2       168456190 209713151  41256962 19.7G  5 Extended
/dev/sda5       168456192 209713151  41256960 19.7G 82 Linux swap / Solaris
```

Figura 3. fdisk.

Ahora llenamos de ceros el disco creado, como se muestra en la figura 4.

```
root@ignacio-leal -> /h/sansforensics
# dd if=/dev/zero of=/dev/sdb
dd: writing to '/dev/sdb': No space left on device
2097153+0 records in
2097152+0 records out
1073741824 bytes (1.1 GB, 1.0 GiB) copied, 72.5214 s, 14.8 MB/s
```

Figura 4. Formateo del disco duro externo.

Ahora con el comando fdisk, creamos las particiones, como se muestra en la figura5,

```
root@ignacio-leal -> /h/sansforensics
# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.27.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x99b525d8.
```

Figura 5. Fdisk.

Creamos 3 particiones primarias y una extendida, primero creamos las particiones primarias, como se muestran en las figuras 6 y 7.

```
Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-2097151, default 2048):
Last sector, +sectors or +size{K,M,G,T,P} (2048-2097151, default 2097151): +150M

Created a new partition 1 of type 'Linux' and of size 150 MiB.

Command (m for help): n
Partition type
   p   primary (1 primary, 0 extended, 3 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (2-4, default 2):
First sector (309248-2097151, default 309248):
Last sector, +sectors or +size{K,M,G,T,P} (309248-2097151, default 2097151): +350M

Created a new partition 2 of type 'Linux' and of size 350 MiB.

Command (m for help): n
Partition type
   p   primary (2 primary, 0 extended, 2 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (3,4, default 3):
First sector (1026048-2097151, default 1026048):
Last sector, +sectors or +size{K,M,G,T,P} (1026048-2097151, default 2097151): +100M

Created a new partition 3 of type 'Linux' and of size 100 MiB.

Command (m for help):
```

Figura 6. Particiones primarias.

Ya creadas las tres particiones primarias, ahora creamos la partición extendida.

```
Command (m for help): n
Partition type
   p   primary (3 primary, 0 extended, 1 free)
   e   extended (container for logical partitions)
Select (default e): e

Selected partition 4
First sector (1230848-2097151, default 1230848):
Last sector, +sectors or +size{K,M,G,T,P} (1230848-2097151, default 2097151):

Created a new partition 4 of type 'Extended' and of size 423 MiB.

Command (m for help): n
All primary partitions are in use.
Adding logical partition 5
First sector (1232896-2097151, default 1232896):
Last sector, +sectors or +size{K,M,G,T,P} (1232896-2097151, default 2097151): +175M

Created a new partition 5 of type 'Linux' and of size 175 MiB.

Command (m for help): n
All primary partitions are in use.
Adding logical partition 6
First sector (1593344-2097151, default 1593344):
Last sector, +sectors or +size{K,M,G,T,P} (1593344-2097151, default 2097151):

Created a new partition 6 of type 'Linux' and of size 246 MiB.

Command (m for help):
```

Figura 7. Partición extendida con dos particiones primarias.

La tabla de particiones creada se muestra en la figura 8.

```
Command (m for help): p
Disk /dev/sdb: 1 GiB, 1073741824 bytes, 2097152 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x99b525d8

Device     Boot   Start     End Sectors  Size Id Type
/dev/sdb1          2048  309247  307200  150M 83 Linux
/dev/sdb2        309248 1026047  716800  350M 83 Linux
/dev/sdb3       1026048 1230847  204800  100M 83 Linux
/dev/sdb4       1230848 2097151  866304  423M  5 Extended
/dev/sdb5       1232896 1591295  358400  175M 83 Linux
/dev/sdb6       1593344 2097151  503808  246M 83 Linux
```

Figura 8. Tabla de particiones.

Para que no sean el mismo tipo de particiones, las vamos a cambiar como se muestra en la figura 9.

```
Command (m for help): t
Partition number (1-6, default 6): 2
Partition type (type L to list all types): 6

Changed type of partition 'Linux' to 'FAT16'.

Command (m for help): t
Partition number (1-6, default 6): 3
Partition type (type L to list all types): 82

Changed type of partition 'Linux' to 'Linux swap / Solaris'.

Command (m for help): 6
6: unknown command

Command (m for help): t
Partition number (1-6, default 6): 6
Partition type (type L to list all types): 6

Changed type of partition 'Linux' to 'FAT16'.
```

Figura 9. Cambio de tipo de particiones.

Ahora mostramos la tabla de particiones, como se muestra en la figura 10.

```
Command (m for help): p
Disk /dev/sdb: 1 GiB, 1073741824 bytes, 2097152 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x99b525d8

Device     Boot   Start     End Sectors  Size Id Type
/dev/sdb1          2048  309247  307200  150M 83 Linux
/dev/sdb2        309248 1026047  716800  350M  6 FAT16
/dev/sdb3       1026048 1230847  204800  100M 82 Linux swap / Solaris
/dev/sdb4       1230848 2097151  866304  423M  5 Extended
/dev/sdb5       1232896 1591295  358400  175M 83 Linux
/dev/sdb6       1593344 2097151  503808  246M  6 FAT16
```

Figura 10. Tabla de particiones.

Ahora guardamos con el comando *"w"* como se muestra en la figura 11.

```
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

Figura 11. Guardar cambios.

Comprobamos que se guardarán los cambios correctamente como se muestra en la figura 12.

```
root@ignacio-leal -> /h/sansforensics
# fdisk -l /dev/sdb
Disk /dev/sdb: 1 GiB, 1073741824 bytes, 2097152 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x99b525d8

Device     Boot   Start     End Sectors  Size Id Type
/dev/sdb1          2048  309247  307200  150M 83 Linux
/dev/sdb2        309248 1026047  716800  350M  6 FAT16
/dev/sdb3       1026048 1230847  204800  100M 82 Linux swap / Solaris
/dev/sdb4       1230848 2097151  866304  423M  5 Extended
/dev/sdb5       1232896 1591295  358400  175M 83 Linux
/dev/sdb6       1593344 2097151  503808  246M  6 FAT16
root@ignacio-leal -> /h/sansforensics
```

Figura 12. Tabla de particiones.

Ahora analizamos el MBR, como se muestra en la figura 13.

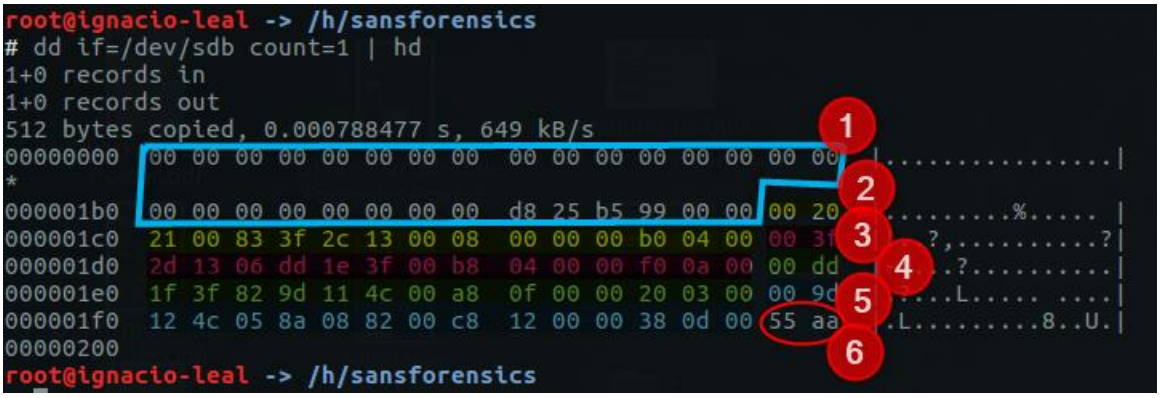

Figura 13. MBR.

1. Código de arranque: 446 bytes.
2. Partición 1: 16 bytes.
3. Partición 2: 16 bytes.
4. Partición 3: 16 bytes.
5. Partición 4: 16 bytes.
6. Fin MBR: 2 bytes.

La tabla MBR reserva 16 bytes para definir una partición, es posible especificar hasta cuatro particiones en la MBR. Una de esas particiones puede ser de tipo extendida.

- Byte 0: Indica si la partición está activa (0x80) o no (0x00).
- Byte 1: Indica el cabezal donde inicia la partición.
- Byte 2-3: Indican el sector y cilindro donde inicia la partición.
- Byte 4: Indica el tipo de partición.
- Byte 5: Indica el cabezal donde finaliza la partición.

- Byte 6-7: Indican el sector y cilindro donde finaliza la partición.
- Byte 8-11: Indican la distancia entre sectores, desde la tabla de particiones al primer sector de la partición (sector de inicio).
- Byte 12-15: Número de sectores en la partición (longitud de la partición).

Ahora mostraremos en cuales bytes de nuestra tabla MBR corresponden a cada uno, empezando a mostrar el byte que indica si la partición esta activa, como se muestra en la figura 14.

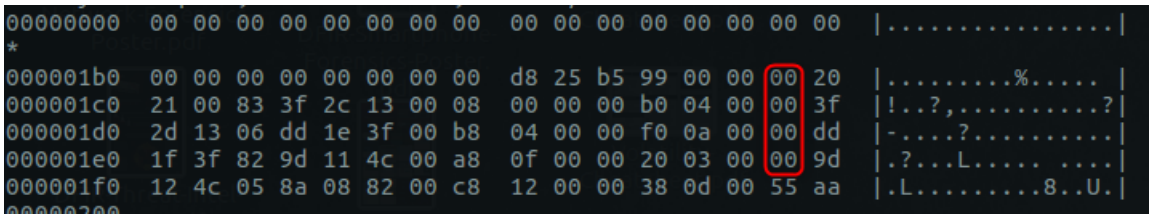

Figura 14. Byte que indica si la partición esta activa.

Ahora en la figura 15, mostramos el byte que indica el cabezal donde inicia la partición.

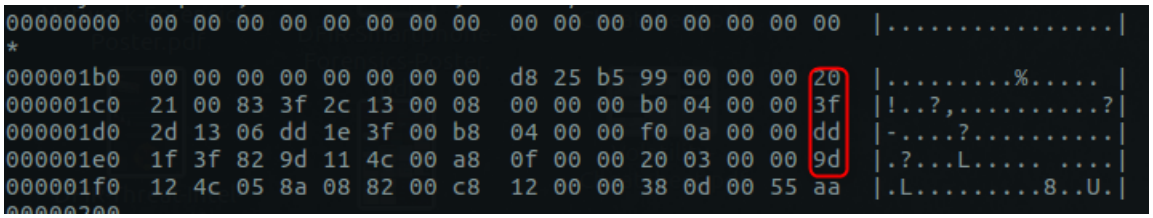

Figura 15. Byte que indica el cabezal donde inicia la partición.

En la figura 16, mostramos los bytes que indican el sector y cilindro donde inicia la partición.

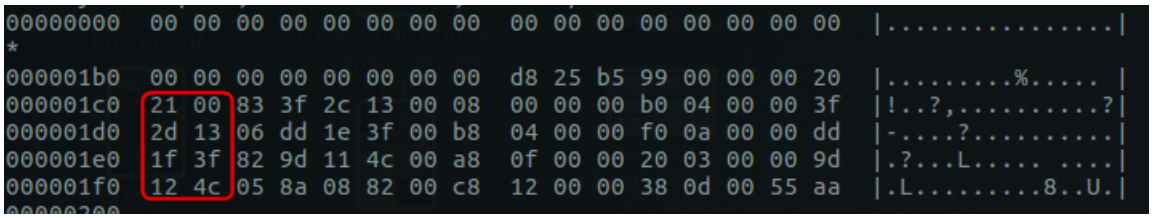

Figura 16. Bytes que indican el sector y cilindro donde inicia la partición

En la figura 17, mostramos el byte que indica el tipo de partición.

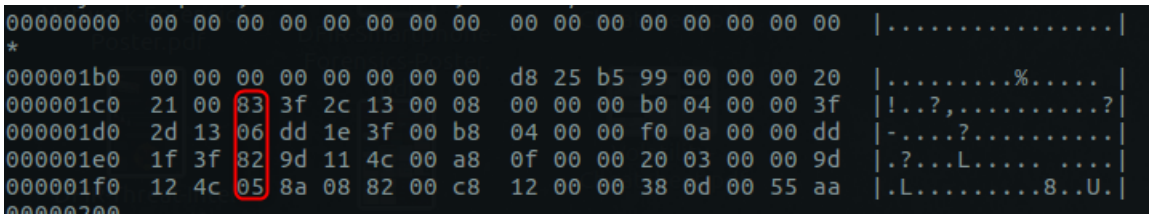

Figura 17. Byte que indica el tipo de partición

En la figura 18, mostramos el byte que indica el cabezal donde finaliza la partición.

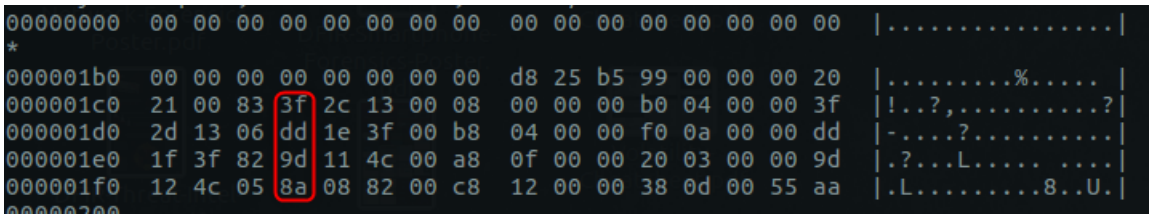

Figura 18. Byte que indica el cabezal donde finaliza la partición.

En la figura 19, mostramos los bytes que indican el sector y cilindro donde finaliza la partición.

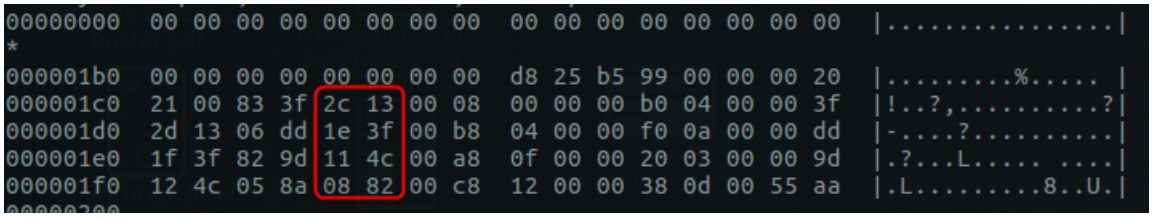

Figura 19. Bytes que indican el sector y cilindro donde finaliza la partición.

En la figura 20, mostramos los bytes que indican la distancia entre sectores, desde la tabla de particiones al primer sector de la partición (sector de inicio).

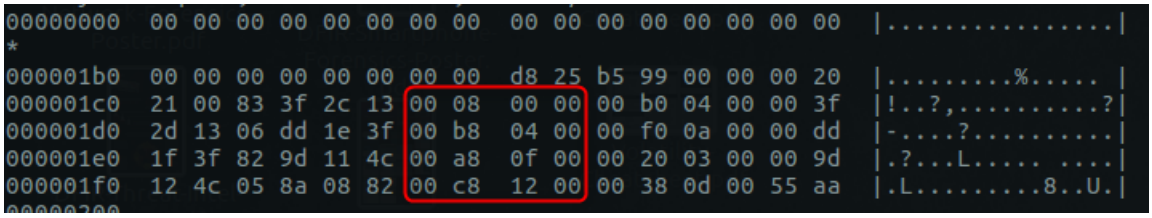

Figura 20. Bytes que indican la distancia entre sectores, desde la tabla de particiones al primer sector de la partición (sector de inicio).

En la figura 21, mostramos los bytes que indican el número de sectores en la partición (longitud de la partición).

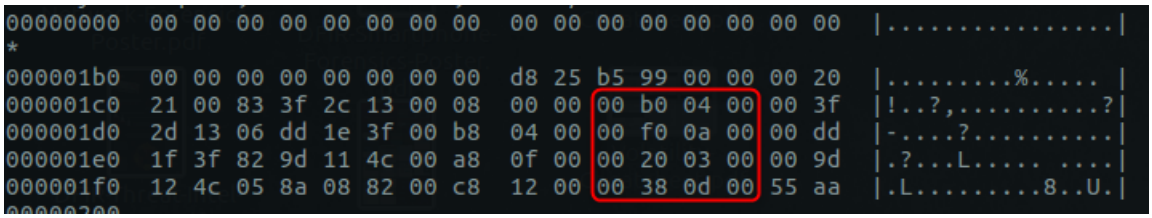

Figura 21. Bytes que indican el número de sectores en la partición (longitud de la partición).

Ahora solo basta conocer las particiones extendidas, como se muestra en la figura 22.

Figura 22. Tabla MBR, de las particiones extendidas.

1. Código de arranque: 446 bytes.
2. Partición 1: 16 bytes.
3. Partición 2: 16 bytes.
4. Partición 3: 16 bytes.
5. Partición 4: 16 bytes.
6. Fin MBR: 2 bytes.

En este caso observamos que solamente las particiones 1 y 2, están definidas y eso concuerda con el particionamiento que realizamos previamente, la posición de los bytes indica lo mismo que en este caso.

Ahora mostraremos en cuales bytes de nuestra tabla MBR corresponden a cada uno, empezando a mostrar el byte que indica si la partición esta activa, como se muestra en la figura 23.

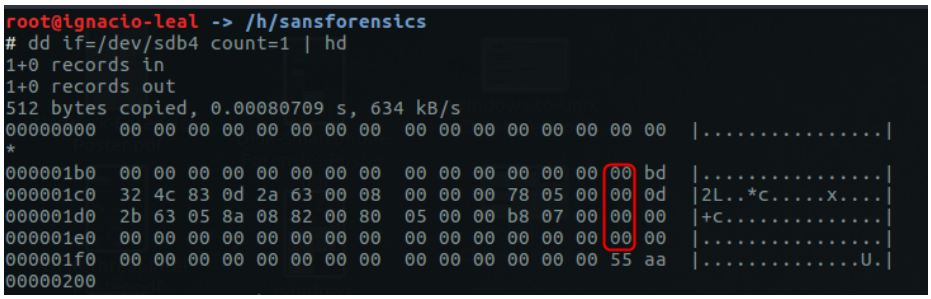

Figura 23. Byte que indica si la partición esta activa.

Ahora en la figura 24, mostramos el byte que indica el cabezal donde inicia la partición.

Figura 24. Byte que indica el cabezal donde inicia la partición.

En la figura 25, mostramos los bytes que indican el sector y cilindro donde inicia la partición.

Figura 25. Bytes que indican el sector y cilindro donde inicia la partición

En la figura 26, mostramos el byte que indica el tipo de partición.

Figura 26. Byte que indica el tipo de partición

En la figura 27, mostramos el byte que indica el cabezal donde finaliza la partición.

Figura 27. Byte que indica el cabezal donde finaliza la partición.

En la figura 28, mostramos los bytes que indican el sector y cilindro donde finaliza la partición.

Figura 28. Bytes que indican el sector y cilindro donde finaliza la partición.

En la figura 29, mostramos los bytes que indican la distancia entre sectores, desde la tabla de particiones al primer sector de la partición (sector de inicio).

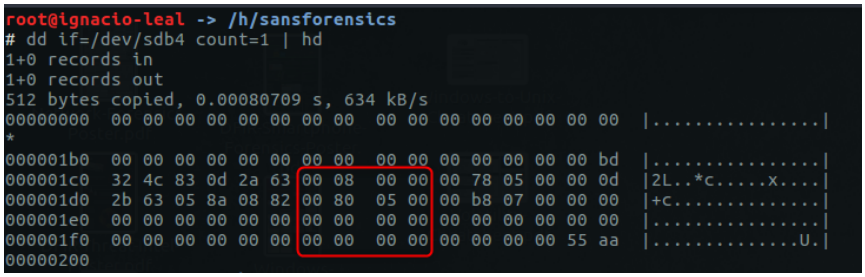

Figura 29. Bytes que indican la distancia entre sectores, desde la tabla de particiones al primer sector de la partición (sector de inicio).

En la figura 30, mostramos los bytes que indican el número de sectores en la partición (longitud de la partición).

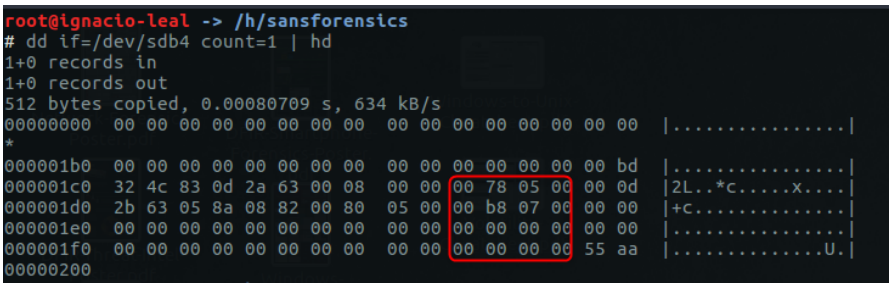

Figura 30. Bytes que indican el número de sectores en la partición (longitud de la partición).

Las particiones 3 y 4 están vacías.

Conclusiones

La realización de esta práctica me permitió conocer como realizar particiones desde línea de comandos, lo cuál es de gran utilidad para formatear discos y cambiar el tipo de partición, es indispensable para un analista forense conocer el formato de la tabla MBR, ya que al conocerlo podemos identificar información relevante del disco.

Referencias

https://maslinux.es/comando-fdisk-para-administrar-particiones-de-disco-en-gnulinux/