Vuelvo con este artículo que si bien no tiene mucho de “hacking” propiamente dicho, si que tiene su parte “forense”, que también tiene su encanto. Debido a la suma importancia personal de los datos que contenía el HDD, el poder recuperarlos fue todo un éxito porque además, no se perdió ni un solo documento y es por ello que os lo traslado aquí.

**La historia.**

Hace pocas semanas un familiar cercano me comunicó que su HDD donde tenía todas las fotos, vídeos y demás “timeline” de su familia tanto actual como los no presentes ya, no lo reconocía su equipo. Tras indicarle lo típico “¿has apagado y encendido el disco duro?, ¿has conectado el disco duro a otro ordenador?, ¿has pintado de verde el disco y tirado al río?, etc…” pude diagnosticarle, atrevidamente por teléfono, que había “cascao”.

Cuando coincidimos en una reunión típica familiar (dichos familiares viven bastante lejos de dónde yo) me dieron el regalito del Disco Duro Seagate 7200.11 con firmware SD81 de 500Gb, con 499Gb ocupados de información valiosa.

Mi idea era la que suelo seguir en caso de que me traigan discos duros que suenan a castañuelas y limitado a los cuatro aparatitos que tengo: o intentar dejar el disco duro en una posición donde la mecánica del mismo permita temporalmente acceso (muchas veces he recuperado así algunos datos) o congelando el disco duro (y también he tenido éxito algunas veces).

Pero una simple navegación por San Google (Duck Duck Go más bien) buscando el modelo por si acaso había algún hilo sobre este disco y bingo!, este modelo con algunos firmware distintos al que calza nuestro disco duro a reparar, contiene un fallo de serie que consiste en que el log se corrompe debido a una serie de eventos que registra el propio HDD.



Para que nos entendamos, estos modelos tienen un log de 320 registros el cual, desde que encendemos el disco, comienza a registrar eventos y tal (nada nuevo). El problema viene cuando el puntero que registra la posición del siguiente evento, está desplazado una posición (diseño y tal). Sí, ¿y que pasa? Pues que si coincide que estamos en la posición 319, y se finaliza la escritura en la posición/registro 319, y el puntero se queda en la 320, como apaguemos el disco en ese momento y queramos encenderlo, nos habrá “cascao”.

Se deben dar también unas condiciones concretas de escritura al inicio del registro 320, no solamente la mera situación del puntero es la que produce esta corrupción sino que se tiene que dar unas condiciones de escritura del log.

Con esto, al iniciar el disco duro, detecta una corrupción de memoria y bloqueará el acceso impidiendo así acceder a nuestros datos. Además, todo esto se quedará registrado en la SMART (tecnología para detección de fallos del Disco Duro, anticipar errores del disco, etc…) y por consiguiente, en los siguientes inicios que hagamos nos “bloqueará” el acceso.

**Manos a la obra.**

Bien, puesto en antecedentes y explicando un poco por encima que es lo que le ha pasado al disco, comentar que esto tenía aparentemente, solución.

Concretamente nuestro disco tiene el firmware SD81, que está dentro del “blackseagate” que comentamos. La información que encontré normalmente refería a SD15, el cual cambia alguno de los pasos o datos a introducir por hyperterminal.

**¿Qué se necesita?**

1. Tiempo
2. USB to Serial (UART)
3. Cable alimentación SATA
4. Hyperterminal

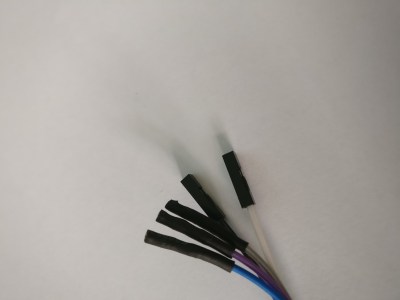
Una vez tengamos estos 4 requisitos, lo demás es bastante sencillo, al menos es la conclusión que sacas cuando lo haces y te funciona (xD).

El dispositivo UART es típico para los que tocan arduino y modifican firmwares de routers, etc.. además su bajo coste (3€ aproximadamente) lo hacen bastante accesible para todos los públicos.



Si aconsejaría que os decantárais por un dispositivo UART que al menos traiga los cables de conexionado largos de 30cm, por ejemplo, para tener maniobrabilidad.

También, tuve problemas con la conexión que traen los cables, porque llevan conectores de plástico duro y para conectarlos a la conexión del HDD tuve que quitarle dichos conectores y tirar de tubo termoretráctil para usarlo de aislante.



*Muestra de 3 cables con conector quitado y en su lugar, termo-retráctil*

Es necesario alimentar el disco duro para proceder con el trabajo. Para ello debes ingeniartelas para poder conectarle de alguna forma un cable SATA (desde una torre de un ordenador cercana, desde una fuente de alimentación de un ordenador, etc…)yo tengo un kit de conexión de discos IDE y SATA por lo que esto no fue problema de negociar o conectar.

Por último necesitamos HYPERTERMINAL. Los que tengáis XP ya podéis actualizaros a otro sistema más moderno o tirar de Linux asi que cuando hagáis eso (:P), si os decantáis por Windows tendréis que descargaros Hyperterminal (<https://onedrive.live.com/?cid=b98f3d302358df4a&id=B98F3D302358DF4A!222>) para poder hacer lo que voy a mostrar. También podéis usar Putty.

**Proceso.**

1. Conectar UART al equipo y comprobar que están los drivers OK.

Esto se hace en el administrador de dispositivos. Si os pasa como a mí, que lo instalé en un Windows 10, el UART que tengo no me iniciaba. Tuve que recurrir a unos drivers de 2008 para que me funcionara. Este fue el paso que más dolor de cabeza me dio aunque parezca mentira x D.

https://i2.wp.com/hacking-etico.com/wp-content/uploads/2017/04/drivers.png?resize=319%2C42&ssl=1

*Muestra de como****NO****debe aparecer el UART en nuestro sistema.*

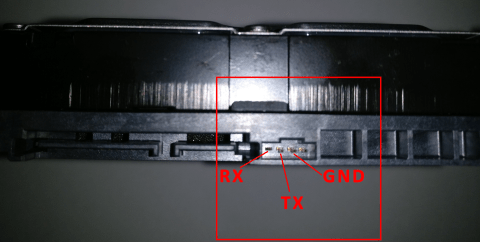
1. Conectar cableado al disco duro.

Importante esto:

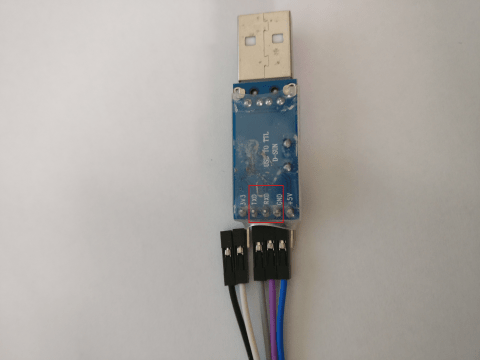
– El conector de transmisión de datos, TX, debe ir conectado a de Recepción de datos, RX, del Disco.

– El conector de recepción de datos del UART, RX, debe ir conectado al de Transmisión de datos del HDD, TX.

– La masa del UART conectada a la masa del HDD.



*Muestra pines de conexión en HDD*



*Muestra de pines de conexión en USB.*

1. Conectar UART al USB del equipo. Con todos los cables ya enchufados.
2. Conectar cable de alimentación al HDD.
3. Crear conexión Hyperterminal con las siguientes características:
   1. Velocidad de transmisión: 38400bps
   2. Bit de datos: 8 bits
   3. Bit de paridad: No
   4. Bits de parada: 1 bit
   5. Control de flujo por hardware: No.
4. Introducción de comandos

Este paso es fundamental. Ojito que este proceso es solo para el SD81, a mí es el que me ha funcionado y desconozco si funcionará o si dañaría el disco duro en otros firmwares por lo que **cuidado con usarlo en otro firmware**.

Comentar que F3 2> o F3 1> es la consola y nos dice en que “nivel” estamos, por lo que los comandos que introducimos son los que están después del >

* F3 2>**U**

Mostrará algo así:

*Spin Up Complete*

*Elapsed Time 5.921 secs*

* F3 2>**/1**
* F3 1>**N1**
* F3 1>
* F3 T>m0,2,2,,,,,22

**Es muy importante teclearlo como está, con la m en minúscula, respetando las comas, sin espacios, etc…**

Veremos algo así como:

*Max Wr Retries = 00, Max Rd Retries = 00, Max ECC T-Level = 14, Max Certify Rewrite Retries = 00C8*

User Partition Format Successful – Elapsed Time 0 mins 00 secs

* F3 T>

Ahora podremos desconectar la alimentación del HDD, y desconectar nuestro UART. Posteriormente, tras esperar algunos segundos, podremos conectar nuestro HDD a algún equipo con adaptador o a una torre, para comprobar si hemos “obrado el milagro” y poder poner los datos a salvo en otro disco duro.

Existe una especie de parche/actualización de Seagate para solucionar este bug en varias versiones afectadas SDxx pero lo más seguro inicialmente es proceder a trasladar la información a otro HDD.

PD: Aclarar que en otros hilos en Internet, utilizan la desconexión de la placa del disco y paran el motor (tras aislarlo con una tarjeta o similar) con comandos (Z) y luego conectan y tal. Esto es para comprobar si nuestro disco duro tiene ese tipo de error (aparecería algo similar a esto: LED:000000CC FAddr:0025A1E1), al menos, a mí el proceso me ha funcionado “saltándome” estos pasos de “aislar” el motor del disco duro de la placa y luego conectar.

Extra:

* La instrucción “**/2″** nos sitúa en el nivel 2, lo que nos da privilegios para realizar los pasos correspondientes.
* Si tecleamos “**/1″**, iremos al nivel 1.
* Para borrar la SMART, previamente introduciendo “**/1**“, utilizamos comando **N1**.
* **U**: Arranca motor de giro del Disco. (Caso de tener que aislar temporalmente la PCB del motor del disco)
* **Z**: Para motor de giro del Disco.(Caso de tener que aislar temporalmente la PCB del motor del disco)