



DIGITAL FORENSICS AND INCIDENT RESPONSE

DIEGO MARTÍN OLEA

Práctica 8

INDICE

Práctica Windows

Práctica Memoria RAM

Práctica Metadatos

Práctica Windows

Tenemos que completar una serie de retos que se encuentran en la página <http://ctf.sancastell.me/challenges> usando las herramientas explicadas en clase y aportar las pruebas y procedimientos de como las hemos obtenido.

El primer reto consiste en obtener el hash de la imagen que tenemos que analizar. Para ello usaremos la máquina virtual de Tsurugi usando Guymager desde la terminal.

Obtenemos esta información

```
Version          : 0.8.13-beta-tsurugi-1
Version timestamp   : 2022-06-23-22.30.49 UTC
Compiled with     : gcc 11.2.0
Using Guymager's own EWF module
libguytools version : 2.1.0
Host name        : tsurugi
Domain name      : (none)
System            : Linux tsurugi 6.0.2-tsurugi #1 SMP PREEMPT_DYNAMIC Sun Oct 16 11:08:13 CEST 2022 x86_64
Linux device      : /home/tsurugi/Downloads/DIEGO MARTIN OLEA/Win10_PC001.vmdk
Device size       : 22991929344 (23.0GB)
Format            : Expert Witness Format, sub-format Guymager - file extension is .Exx
Image meta data
Case number      : Caso_1
Evidence number  : A001
Examiner         : Diego Martin
Description       :
Notes             :
Image path and file name: /home/tsurugi/Downloads/DIEGO MARTIN OLEA/B001.Exx
Info path and file name: /home/tsurugi/Downloads/DIEGO MARTIN OLEA/B001.info
Hash calculation   : MD5, SHA-1 and SHA-256
Source verification : off
Image verification : on
No bad sectors encountered during acquisition.
State: Finished successfully
MD5 hash          : 5ee316b95ad83f67fff1b511c372e2d5
MD5 hash verified source : --
MD5 hash verified image : 5ee316b95ad83f67fff1b511c372e2d5
SHA1 hash          : c407c534116af248c730d3c246f81a6e2d31da1c
SHA1 hash verified source : --
SHA1 hash verified image : c407c534116af248c730d3c246f81a6e2d31da1c
SHA256 hash         : 4446e9c42345a32fa78a8ce20834faa047a3b161eba986f894d2230fcf6b0cbe
SHA256 hash verified source: --
SHA256 hash verified image : 4446e9c42345a32fa78a8ce20834faa047a3b161eba986f894d2230fcf6b0cbe
Image verification OK. The image contains exactly the data that was written.
Acquisition started : 2024-01-05 10:32:12 (ISO format YYYY-MM-DD HH:MM:SS)
Verification started: 2024-01-05 10:42:17
Ended              : 2024-01-05 10:45:54 (0 hours, 13 minutes and 42 seconds)
Acquisition speed  : 36.30 MByte/s (0 hours, 10 minutes and 4 seconds)
Verification speed : 101.05 MByte/s (0 hours, 3 minutes and 37 seconds)
Generated image files and their MD5 hashes
=====
No MD5 hashes available (configuration parameter CalcImageFileMD5 is off)
MD5 Image file
n/a B001.E01
n/a B001.E02
n/a B001.E03
n/a B001.E04
n/a B001.E05
n/a B001.E06
n/a B001.E07
```

Podemos observar como de todos estos datos hemos obtenido el que buscábamos:

SHA256 hash verified image : **4446e9c42345a32fa78a8ce20834faa047a3b161eba986f894d2230fcf6b0cbe**

El siguiente reto consiste en conseguir el nombre de la máquina que estamos analizando.

Para ello primero hemos montado la imagen que tenemos que analizar llamada Win10_PC001.vmdk en Access Data FTK Imager. Esta aplicación lo que nos permite es hacer un arbol de carpetas y ficheros que podemos analizar, leer y nos los estructura.

Los ficheros que nos interesan son los datos volátiles localizados en **%Windir%\System32\Config**:

SAM -> HKEY_LOCAL_MACHINE

SECURITY -> HKEY_LOCAL_MACHINE

SYSTEM -> HKEY_LOCAL_MACHINE

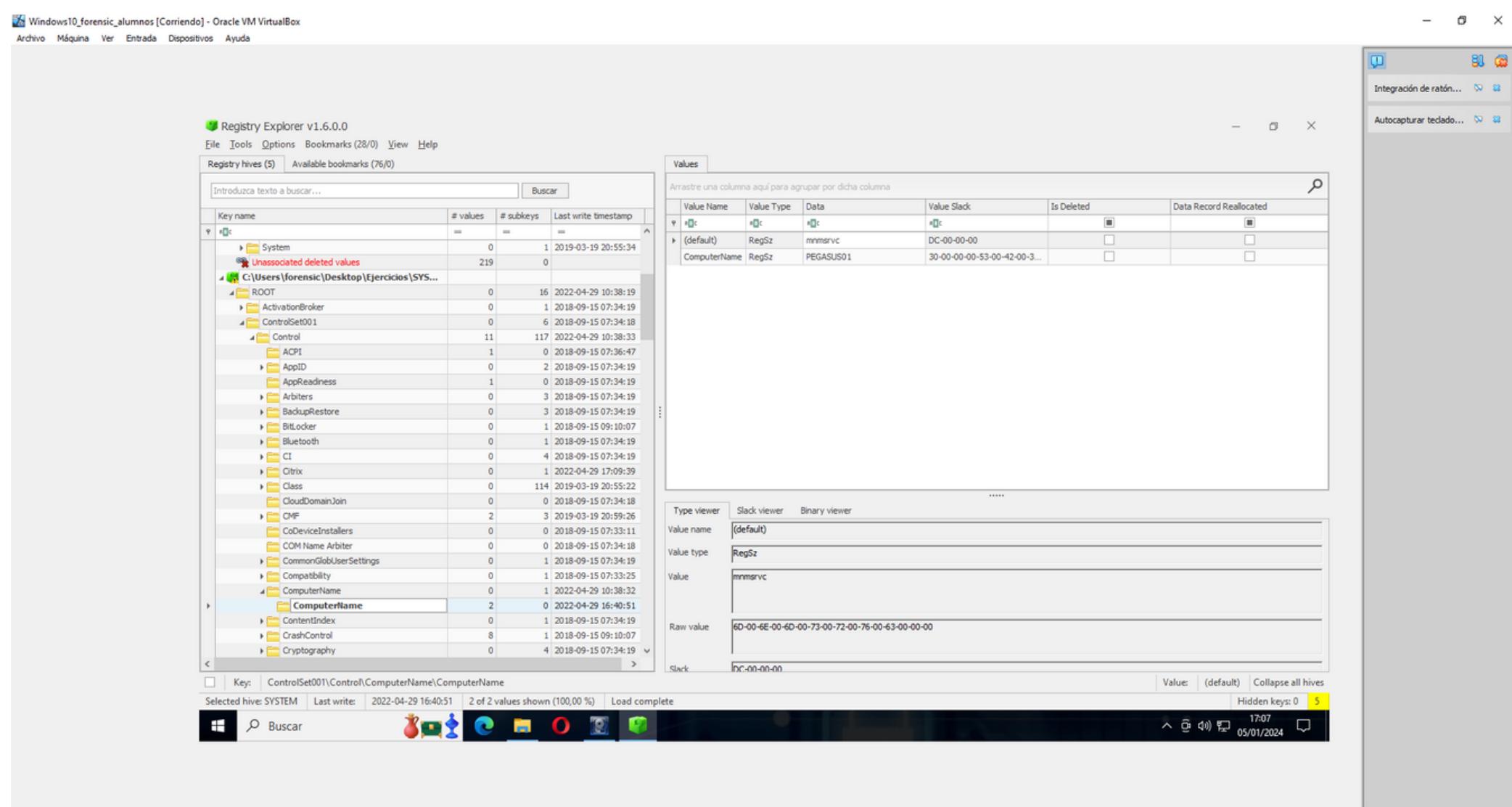
SOFTWARE -> HKEY_LOCAL_MACHINE

DEFAULT -> HKEY_LOCAL_MACHINE

Localizados en **%UserProfile%\{user}\NTUSER.DAT**

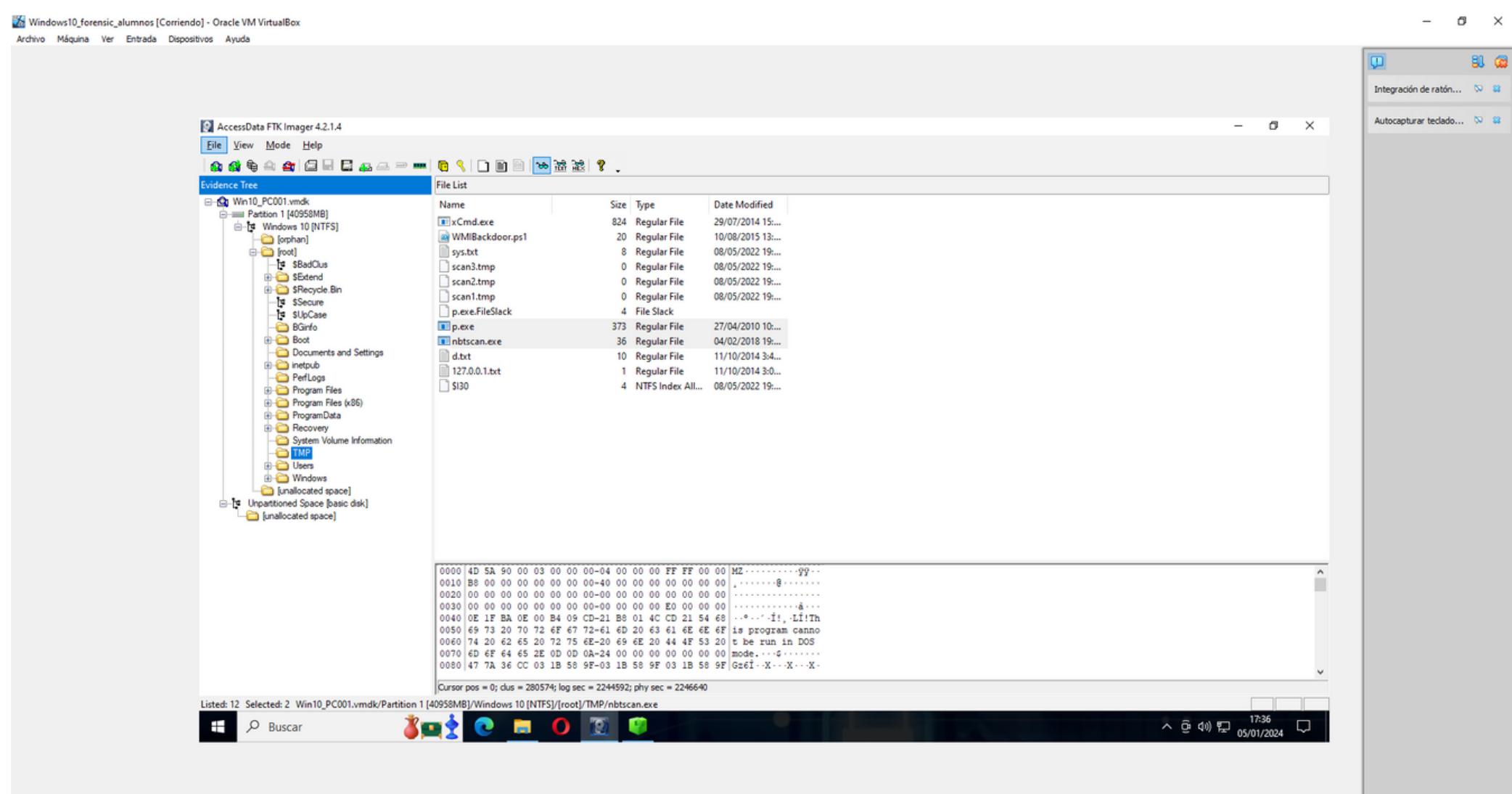
Una vez que los tenemos localizados los extraemos en una carpeta que creamos para guardar estos ficheros. Despues hemos usado la herramienta ZimmermanTools llamada **RegistryExplorer** que es un ejecutable con interfaz gráfica. En este ejecutable introducimos los archivos anteriormente mencionados y nos aparece otro arbol de carpetas y ficheros.

En la ruta de **SYSTEM\ROOT\CONTROLSET001\CONTROL\COMPUTERNAME** la abrimos y nos sale el nombre de **PEGASUS01** que es la máquina que estamos analizando.



Los ficheros maliciosos suelen ser ejecutables .exe que se ejecutan en el sistema y según el tipo pueden realizar unas acciones u otras. Para ello usamos Access Data FTK Imager que nos muestra la MFT. Este archivo almacena la MFT (Master File Table), una tabla que muestra un registro de todo lo que ha ocurrido con el sistema de ficheros del que consta la imagen de disco.

Después de abrir las distintas carpetas del apartado root encontramos una llamada TMP donde nos encontramos unos archivos .exe. No sabemos si es lo que buscamos ya que una máquina puede ejecutar diversos .exe para su correcto funcionamiento por lo que analizaremos este tipo de archivos en virustotal para ver si son ejecutables maliciosos o no.



Encontramos estos 3 ejecutables:

- xCmd.exe
- p.exe
- nbtscan.exe

En virus total vemos que están considerados maliciosos e incluso nos salta el propio antivirus de la máquina en cuanto queremos mover los .exe

The screenshot shows the VirusTotal analysis page for the file c9d5dc956841e000bf... (MD5: c9d5dc956841e000bf...). The page indicates 36 security vendors flagged it as malicious. The file is identified as nbtscan-1.0.35.exe. A prominent red warning box from Avast states: "Amenaza resuelta" (Malware resolved), explaining that the file xCmd.exe was moved to quarantine because it was infected by Win32:Malware-gen. The interface includes tabs for DETECTION, DETAILS, RELATIONS, BEHAVIOR, and COMMUNITY.

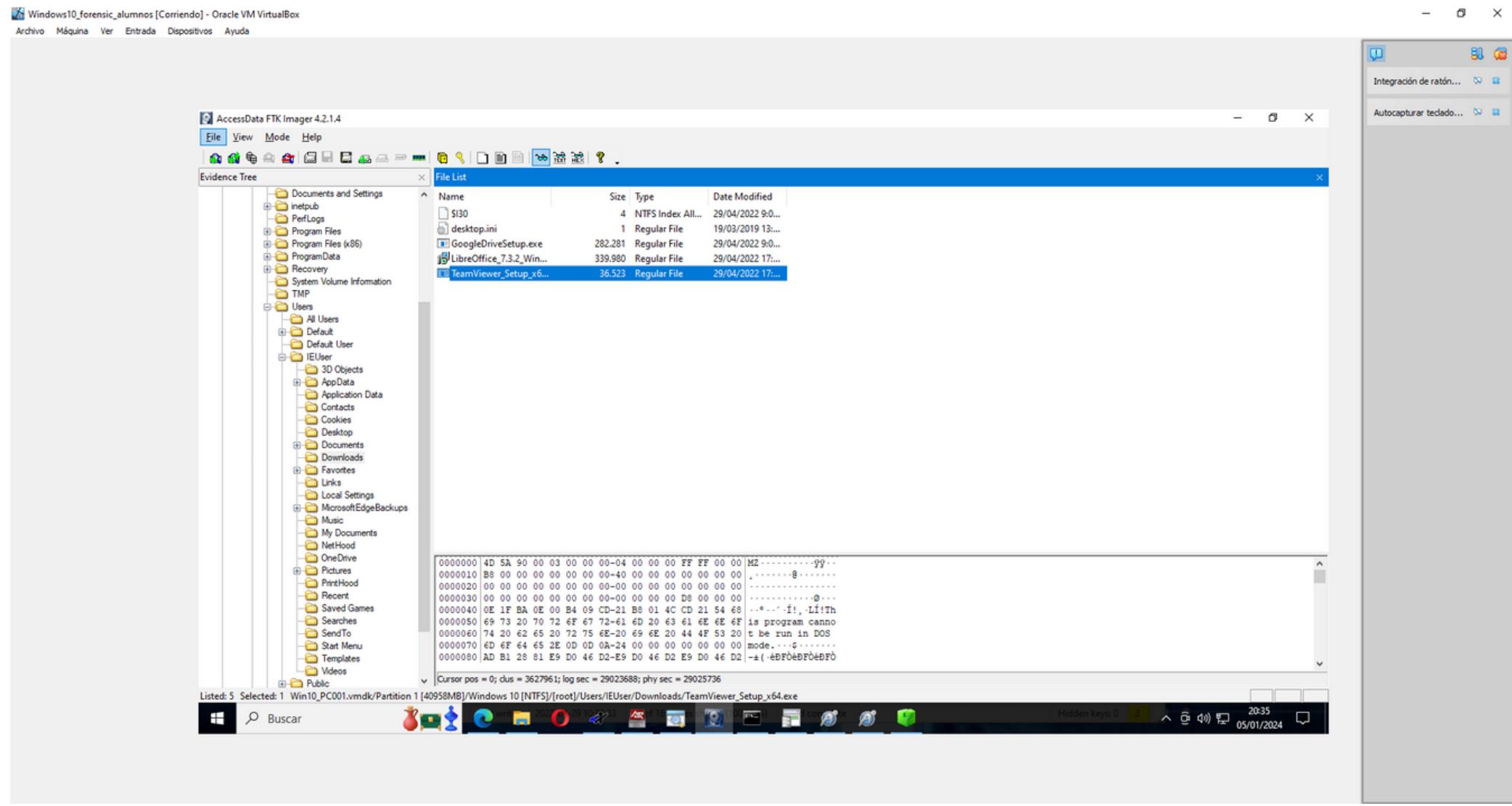
The screenshot shows the AccessData FTK Imager 4.2.1.4 interface. The Evidence Tree panel shows a mounted image file (Win10_PC001.vmdk) with several partitions. The File List panel displays files from the root directory of one partition, including xCmd.exe, WMIBackdoor.ps1, and various log files. A modal dialog from Avast is overlaid on the interface, stating: "Amenaza resuelta" (Malware resolved), "Hemos movido el archivo xCmd.exe.copy0 a la Cuarentena porque estaba infectado por Win32:Malware-gen", and "También podemos protegerle de otros tipos de amenazas". The status bar at the bottom shows the date as 05/01/2024 and the time as 17:45.

Por lo que la carpeta que contiene los ficheros maliciosos es la llamada TMP que se encuentra en **root**.

Ademas nos encontramos el archivo **WMIBackdoor.ps1** que es la powershell maliciosa en la misma carpeta de TMP.

The screenshot shows the FTK Imager interface again, focusing on the File List panel where WMIBackdoor.ps1 is selected. Below the list, a hex dump of the file's content is displayed. The hex dump shows the PowerShell code for the backdoor, starting with EF BB BF (the byte order mark) and continuing with the PowerShell command to establish a WMI backdoor. The status bar at the bottom shows the date as 05/01/2024 and the time as 17:48.

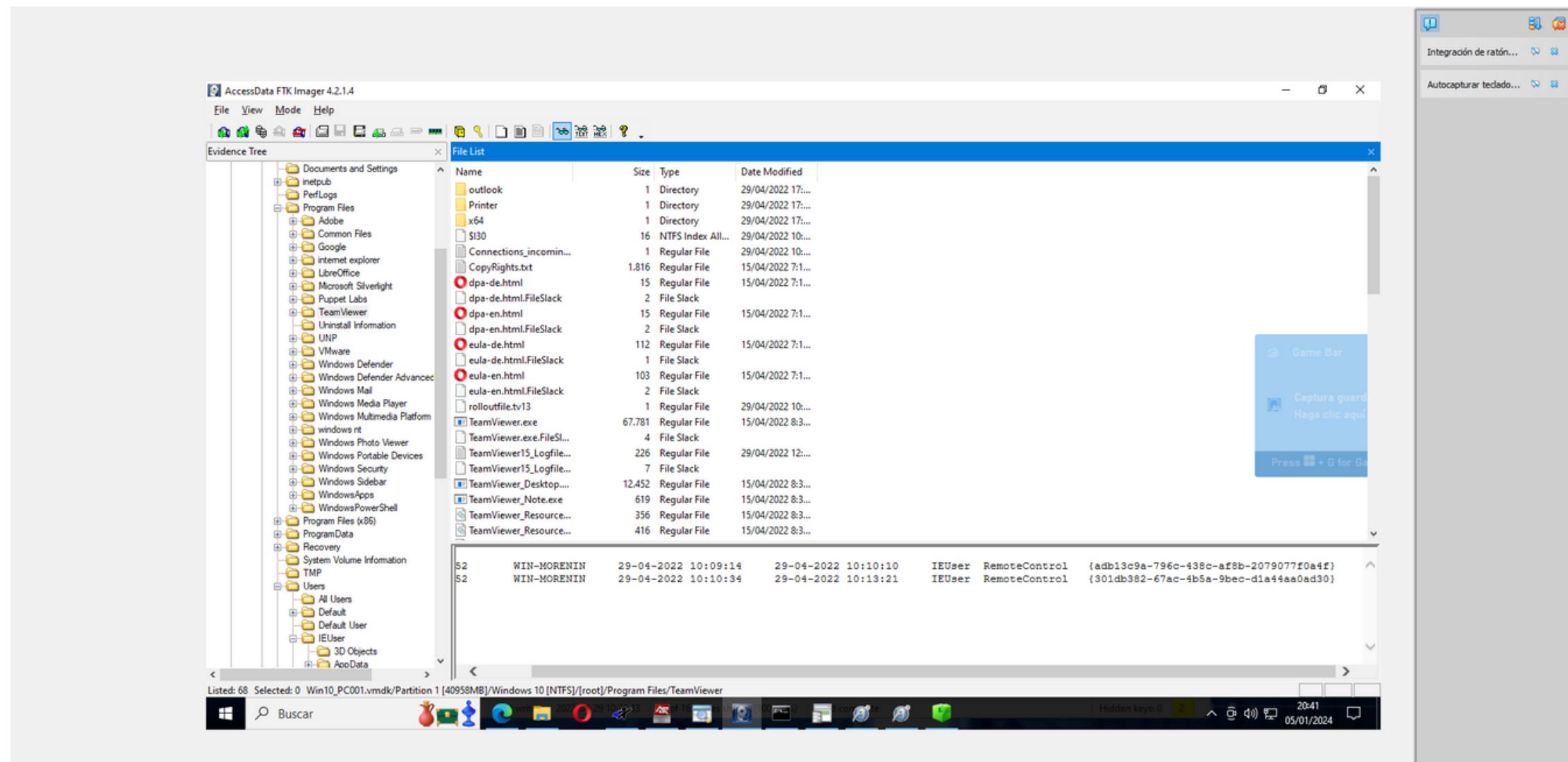
En el reto que nos indica la descarga fichero de control remoto hemos usado distintas herramientas para ver que programa se ha usado. En primer lugar analizando los usuarios de la evidencia nos encontramos distintos usuarios donde encontramos el usuario **IEUSER** que contiene distintas carpetas de documentos usadas. Encontramos la carpeta **Downloads** para ver que archivos o que ficheros ha descargado este usuario y nos encontramos un programa de control remoto llamado **TeamViewer**. En concreto un .exe llamado **TeamViewer_Setup_x64.exe**



Siguiendo esta pista de Teamviewer buscamos en **Programfiles/Teamviewer** y encontramos en el archivo **Connections_incomming.txt** dos conexiones remotas realizadas por IEUSER:

765418952 WIN-MORENIN 29-04-2022 10:09:14 29-04-2022 10:10:10 IEUser RemoteControl {adb13c9a-796c-438c-af8b-2079077f0a4f}

765418952 WIN-MORENIN 29-04-2022 10:10:34 29-04-2022 10:13:21 IEUser RemoteControl {301db382-67ac-4b5a-9bec-d1a44aa0ad30}

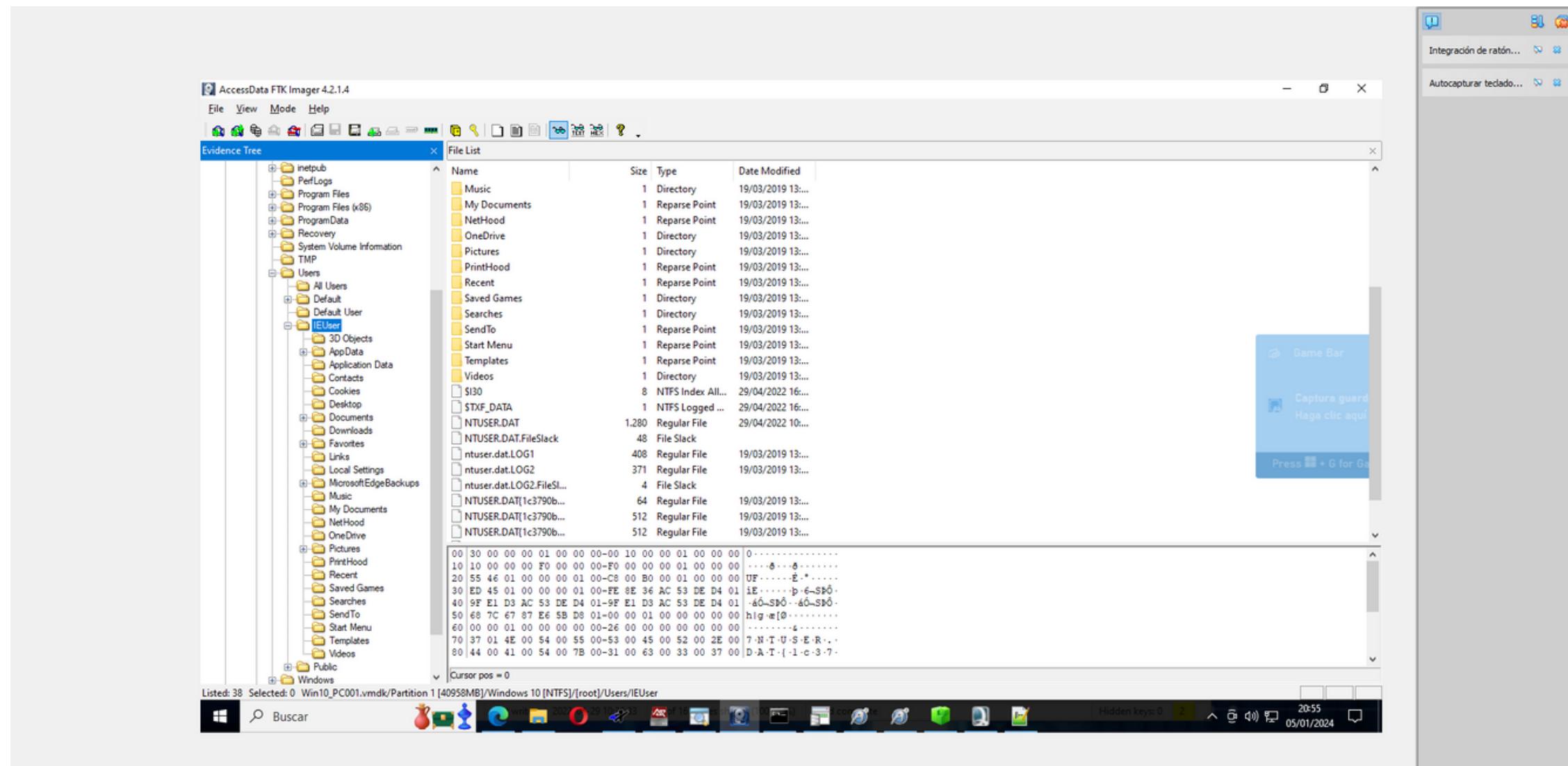


Y obtenemos la ID desde la que se conecta IEUser que es 765418952

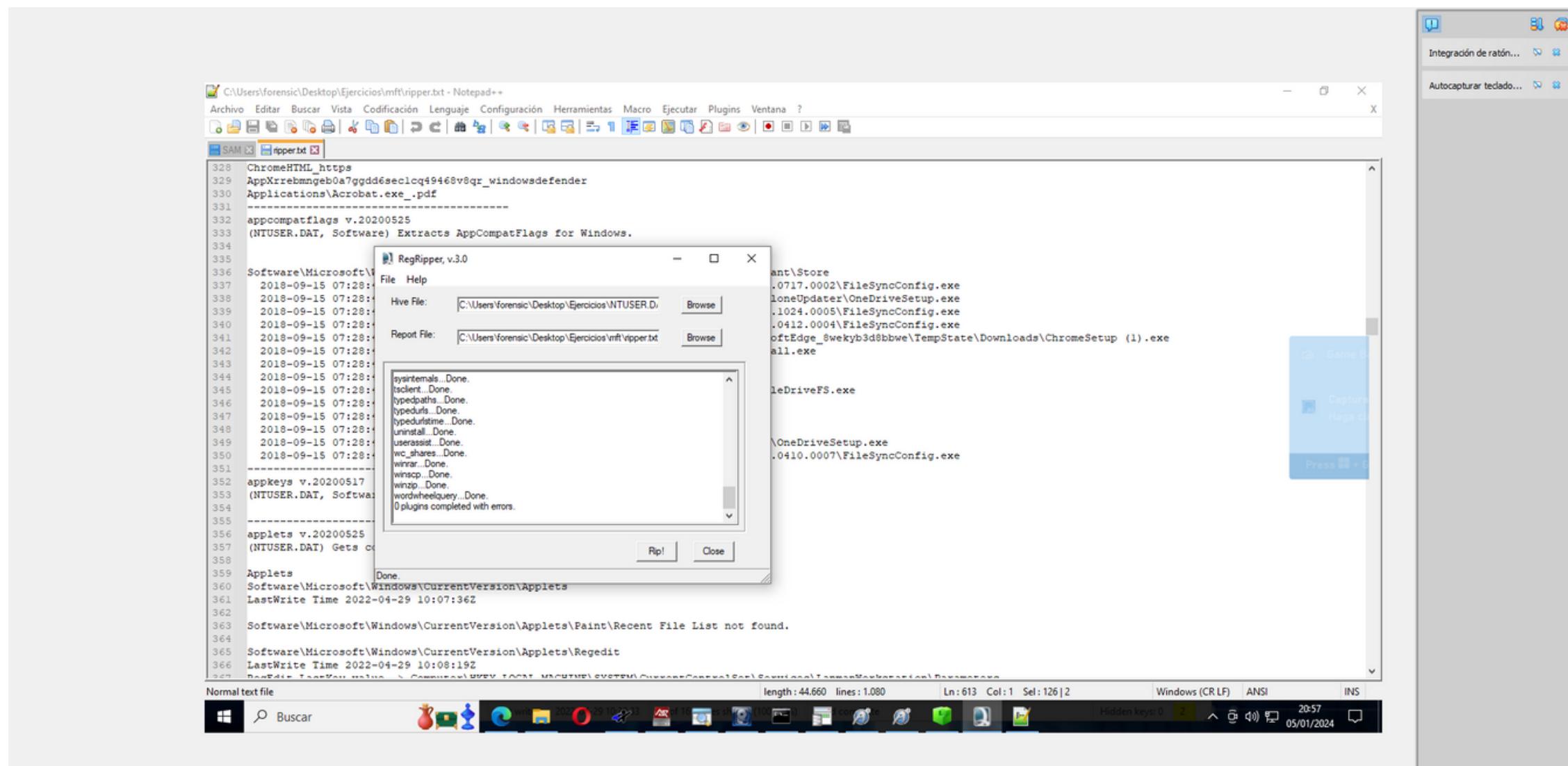
Por lo que hemos encontrado la fecha en la se ejecutó el programa:

29-04-2022

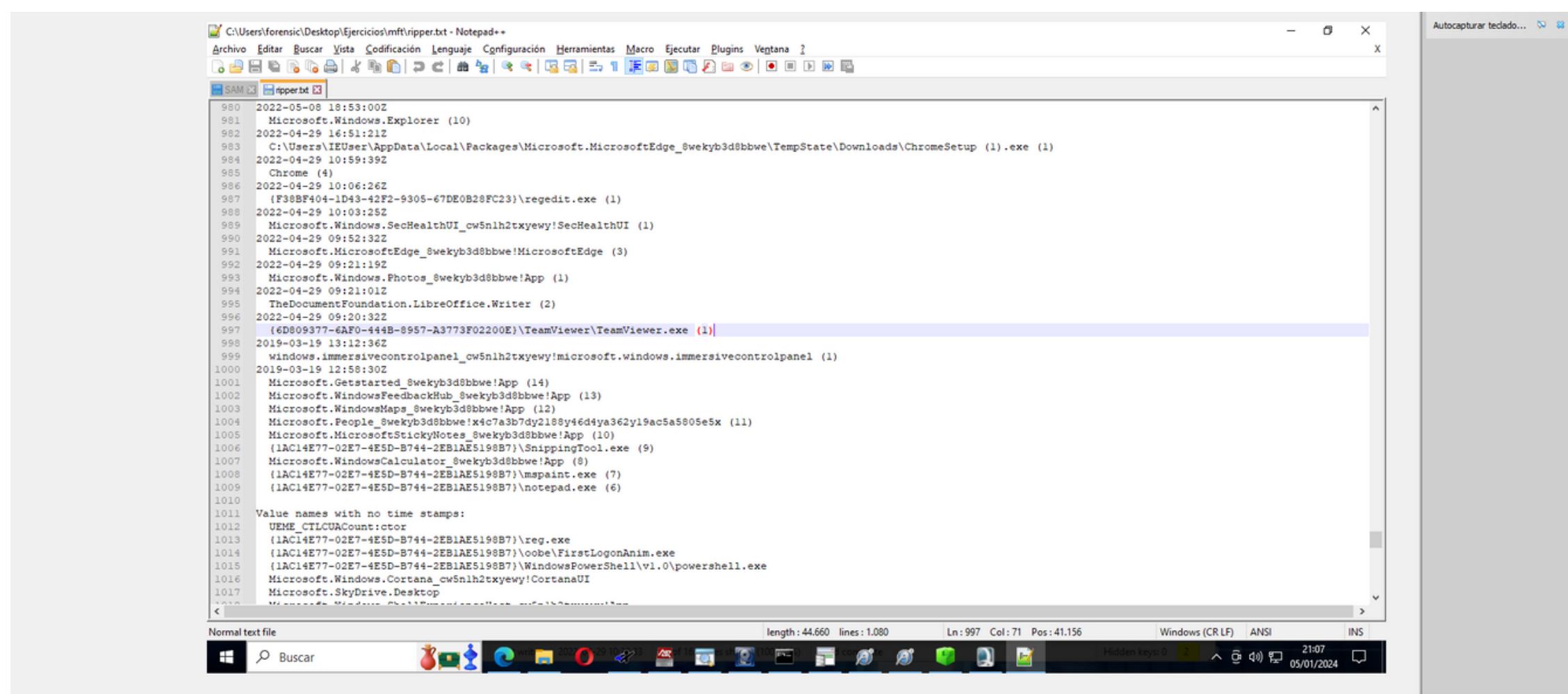
Usando el archivo llamado **NTUSER.DAT** sacado de la ruta **ROOT/USERS/IEUSER**



Extraemos el archivo y lo guardamos en una carpeta. Despues usamos la herramienta llamada **RegRipper** donde analizamos el archivo **NTUSER.DAT**



y nos da un output.txt que guardamos en nuestra carpeta. Una vez que obtenemos el .txt le damos a editar con notepad para ver el contenido del .txt . Leyendo el archivo encontramos el proceso de cuando instaló el Teamviewer y los pasos de lo que fue ejecutando el usuario IEUSER.



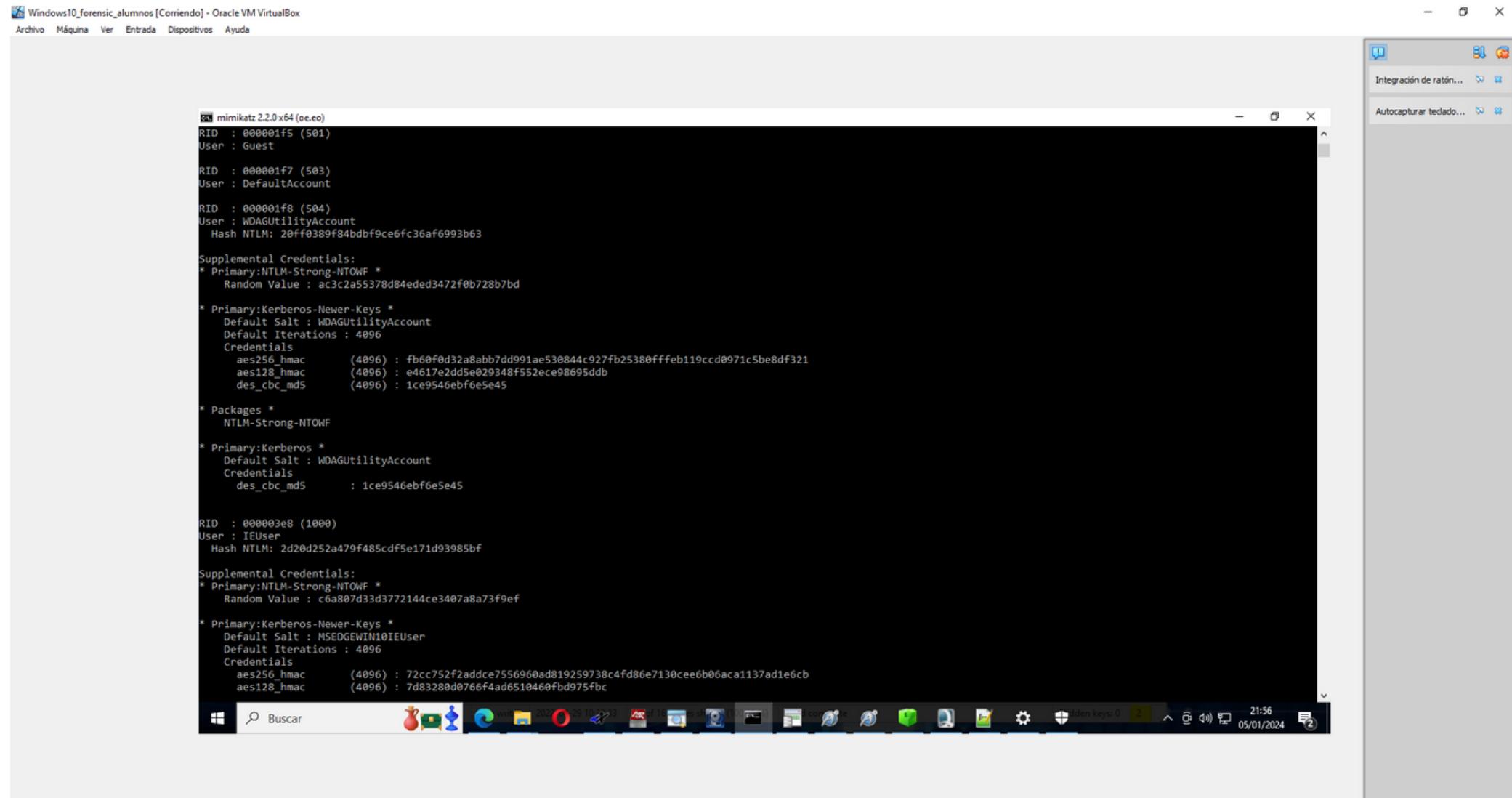
Y encontramos que fue el dia 2022-04-29 cuando instaló el programa.

Para conseguir la contraseña del usuario usaremos la herramienta mimikatz descargada de

<https://github.com/gentilkiwi/mimikatz/releases/tag/2.2.0-20220919>

Usamos el comando **C:\Herramientas\x64>mimikatz.exe -h** para ejecutar desde la consola la aplicación y automáticamente introducimos el comando **lsadump::sam /system:C:\Users\forensic\Desktop\Ejercicios\SYSTEM /sam:C:\Users\forensic\Desktop\Ejercicios\SAM**

La ruta varía según donde este guardado el archivo SYSTEM y SAM que sacamos de la MFT. Una vez ejecutado el comando navegamos por la consola y encontramos que el usuario IEUser tiene de contraseña el hash: Hash NTLM: **2d20d252a479f485cdf5e171d93985bf**



Al obtener el hash podemos usar una página en nuestro navegador llamada <https://crackstation.net/> donde introducimos el hash. Al introducirlo nos sale que la contraseña que corresponde a ese hash es qwerty por lo que habriamos obtenido la contraseña con la que accede.

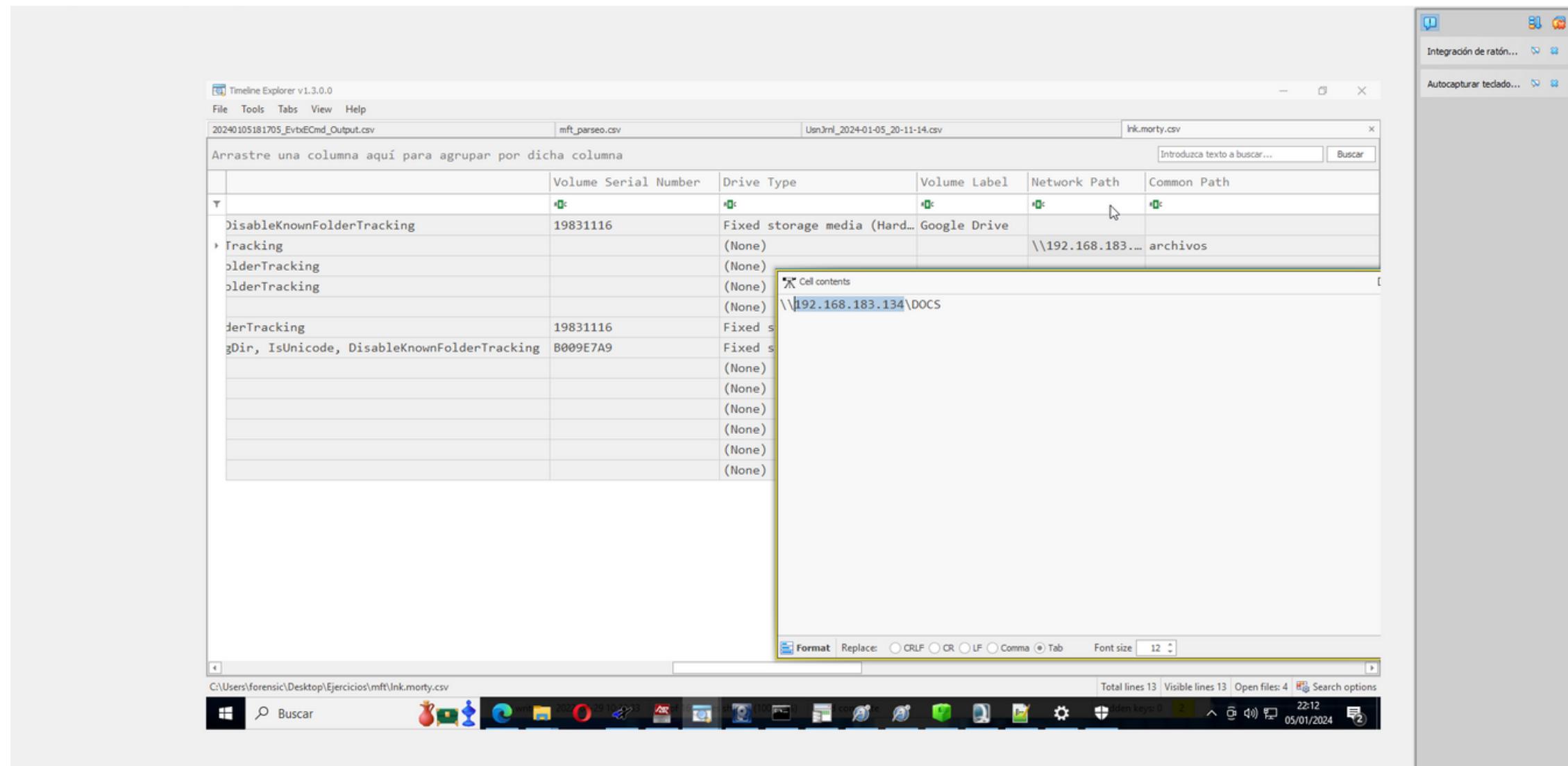
Hash	Type	Result
2d20d252a479f485cdf5e171d93985bf	NTLM	qwerty

Color Codes: Exact match, Partial match, Not found.

Para conseguir la IP desde la que se conecta otro usuario debemos usar los LNK. Para ello usamos la herramienta de **LECcmd.exe** ejecutandola en consola introduciendo este comando:

```
C:\Herramientas\02_ZimmermanTools>LECcmd.exe -d
"C:\Users\forensic\Desktop\Ejercicios\Kape\E\Users\IEUser\AppData\Roaming\Microsoft\Windows\Recent" --csv
C:\Users\forensic\Desktop\Ejercicios\mft --csvf lnk.morty.csv
```

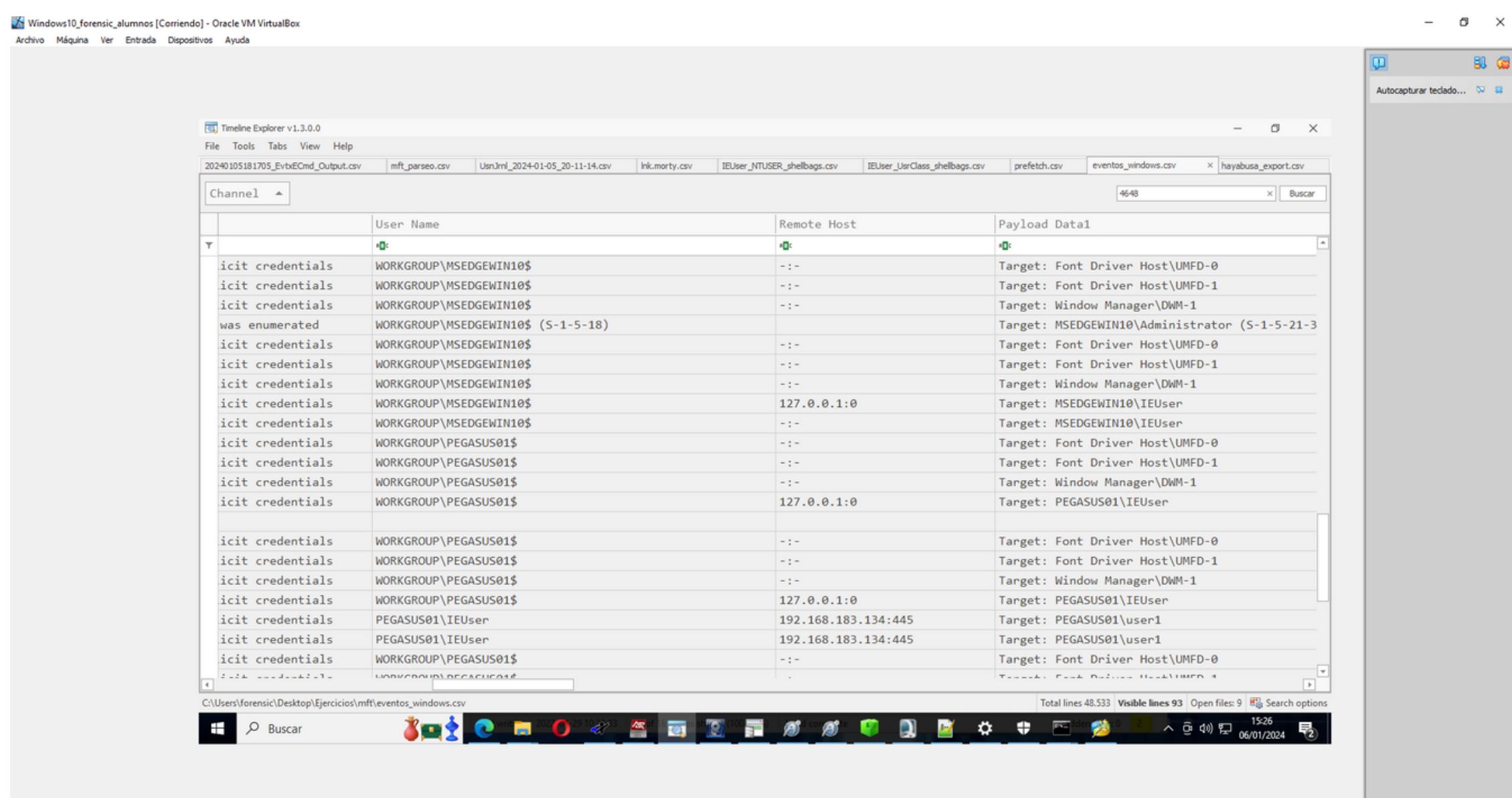
Nos crea un archivo csv que saca de la ruta del usuario que hemos encontrado en la MSF de IEUser. Una vez que hemos obtenido el csv lo leemos con Timeline Explorer y nos encontramos la IP al analizar el csv.



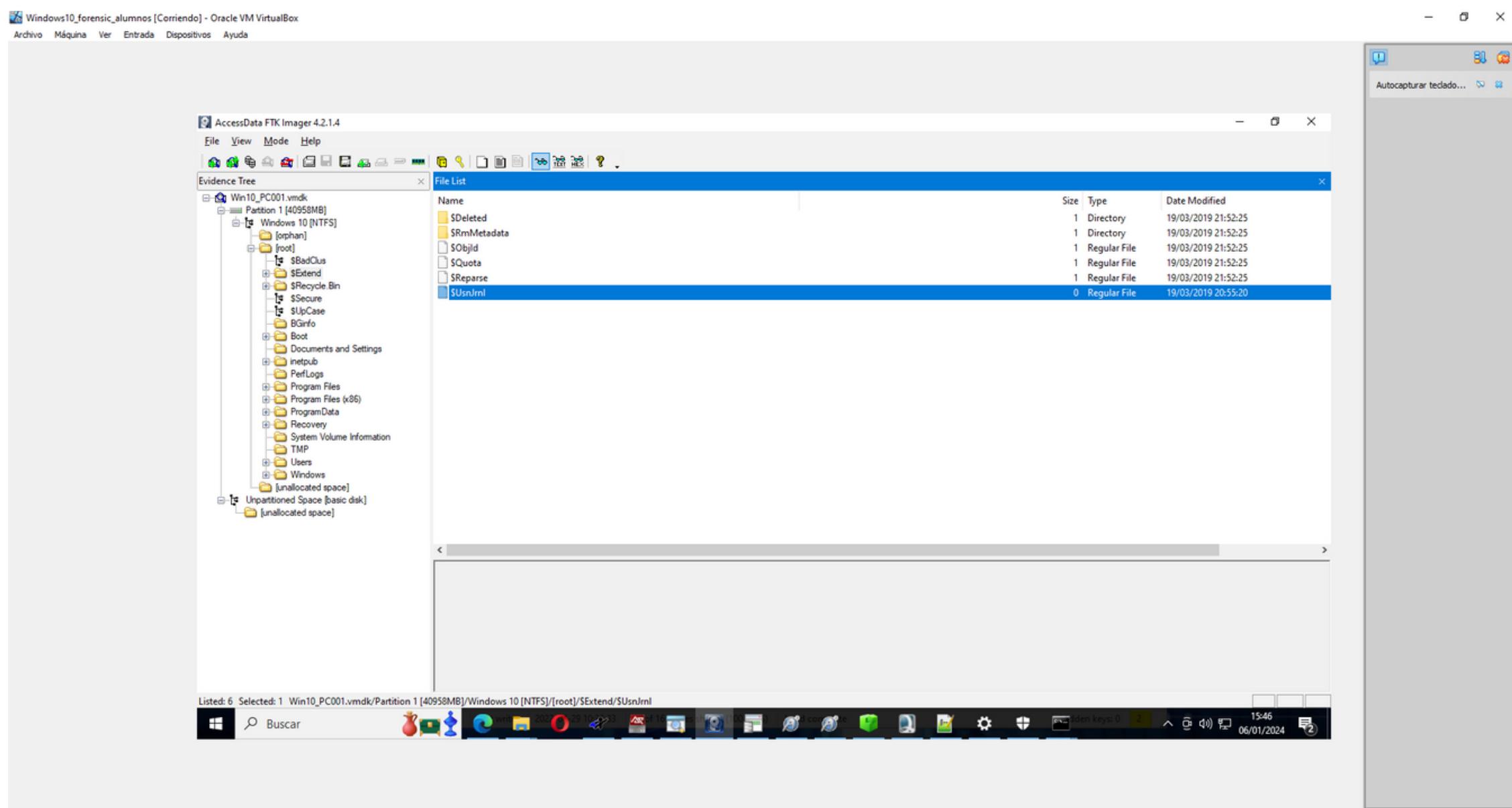
Para conseguir el puerto desde donde se conecta la **ip 192.168.183.134** hemos comprobado los eventos de windows que se encuentran en la ruta "**%System32%\Winevt\Log**" que ya tenemos en nuestro sistema ya que Kape los obtuvo. Por lo que con el comando:

```
C:\Herramientas\02_ZimmermanTools\EvtxECmd>EvtxECmd.exe -d
C:\Users\forensic\Desktop\Ejercicios\Kape\E\Windows\System32\winevt\logs --csvf
C:\Users\forensic\Desktop\Ejercicios\mft --csv eventos_windows.csv
```

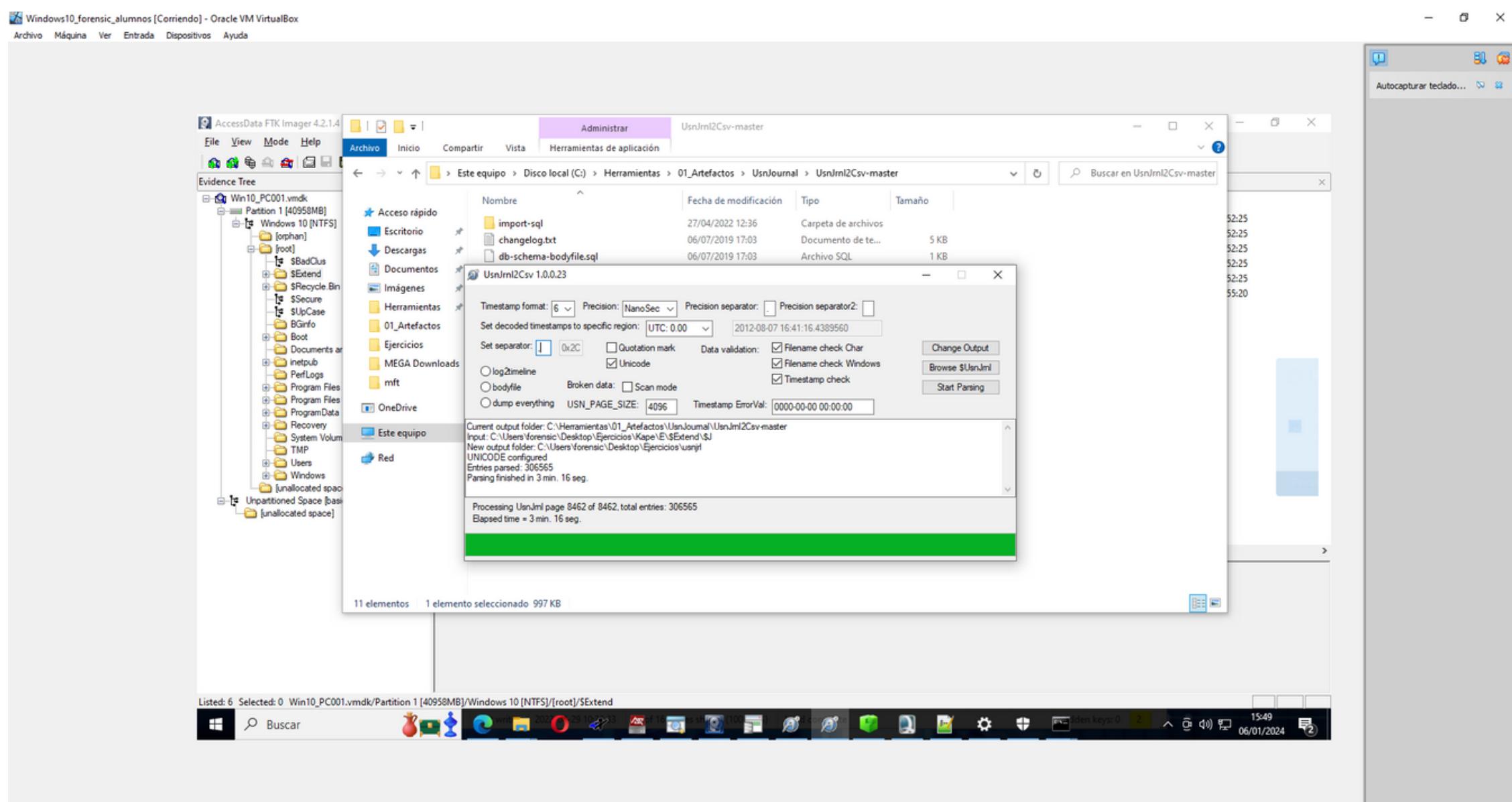
Usamos la herramienta **EvtxECmd** para crear un archivo .csv con todos los eventos de windows. Posteriormente abrimos el archivo csv con TimelineExplorer y buscamos por número de identificador el evento 4648 y ahí encontramos entre los eventos la IP junto con el puerto al que se conecta que es el 445



El archivo .zip borrado es el llamado ***cosas.zip***. Lo encontramos después de usar el archivo extraído de la MFT Usrjrn1 obtenido de la ruta ***root/extend***



Una vez que tenemos el fichero Usrjrn1 usamos la herramienta ***UsnJrn12Csv64.exe*** que es un ejecutable y elegimos el archivo con salida -csv



Una vez que ya tenemos el archivo csv guardado abrimos la herramienta de TimeLine Explore y filtramos los .zip para ver que archivos se borraron. Ahí encontramos el .zip y lo que se hizo con ellos.

The screenshot shows the Timeline Explorer interface with a search bar at the top containing "Introduzca texto a buscar..." and a search button "Buscar". Below the search bar is a table header with columns "Tag", "Offset", and "File Name". A filter bar below the table has "File Name Contiene .zip" selected. The main table lists numerous entries, all of which contain ".zip" in their file names. The first few entries are:

- 3 0x011E6648 chocolatey.zip
- 4 0x011E66A0 chocolatey.zip
- 5 0x011E66F8 chocolatey.zip
- 7 0x020B8A00 cosas.zip
- 8 0x020B8A50 cosas.zip
- 9 0x020B8AA0 cosas.zip
- 0 0x020B8AF0 cosas.zip
- 1 0x020B8B40 cosas.zip
- 2 0x020B8B90 cosas.zip
- 3 0x020B8BE0 cosas.zip
- 4 0x020B8C30 cosas.zip
- 5 0x020B8C80 cosas.zip
- 6 0x020B8CD0 cosas.zip
- 2 0x0210E3E0 cosas.zip
- 0 0x015E8560 elementary.zip
- 1 0x015E85C0 elementary.zip
- 2 0x015E8618 elementary.zip
- 3 0x015E8670 elementary.zip
- 4 0x015E86C0 elementary.zip
- 0 0x012E5700 images_breeze.zip

The status bar at the bottom indicates "Total lines 306.565 | Visible lines 161 | Open files: 10 | 15:50 | 06/01/2024".

The screenshot shows the Timeline Explorer interface with a search bar at the top containing "Introduzca texto a buscar..." and a search button "Buscar". Below the search bar is a table header with columns "▲ USN", "Timestamp", and "Reason". A filter bar below the table has "File Name Contiene .zip" selected. The main table lists numerous file system events, many of which involve ".zip" files. The first few entries are:

- 111044168 2022-04-29 08:23:09.1712131 BASIC_INFO_CHANGE
- 111044256 2022-04-29 08:23:09.1712131 BASIC_INFO_CHANGE+CLOSE
- 111044344 2022-04-29 08:23:09.1712131 CLOSE+FILE_DELETE
- 126585344 2022-05-08 19:03:34.4535258 FILE_CREATE
- 126585424 2022-05-08 19:03:34.4535258 DATA_EXTEND+FILE_CREATE
- 126585504 2022-05-08 19:03:34.4688982 CLOSE+DATA_EXTEND+FILE_CREATE
- 126585584 2022-05-08 19:03:34.4688982 BASIC_INFO_CHANGE
- 126585664 2022-05-08 19:03:34.4688982 BASIC_INFO_CHANGE+CLOSE
- 126585744 2022-05-08 19:03:34.5627126 RENAME_OLD_NAME
- 126585824 2022-05-08 19:03:34.5627126 RENAME_NEW_NAME
- 126585904 2022-05-08 19:03:34.5627126 CLOSE+RENAME_NEW_NAME
- 126585984 2022-05-08 19:03:34.5627126 SECURITY_CHANGE
- 126586064 2022-05-08 19:03:34.5627126 CLOSE+SECURITY_CHANGE
- 126936032 2022-05-08 19:14:07.6185235 RENAME_OLD_NAME
- 115246440 2022-04-29 08:32:57.9812555 FILE_CREATE
- 115246528 2022-04-29 08:32:57.9812555 DATA_EXTEND+FILE_CREATE
- 115246616 2022-04-29 08:32:57.9812555 DATA_EXTEND+DATA_OVERWRITE+FILE_CREATE
- 115246704 2022-04-29 08:32:57.9812555 BASIC_INFO_CHANGE+DATA_EXTEND+DATA_OVERWRITE+FILE_CREATE
- 115246792 2022-04-29 08:32:57.9812555 BASIC_INFO_CHANGE+CLOSE+DATA_EXTEND+DATA_OVERWRITE+FILE_CREATE
- 112088832 2022-04-29 08:32:44.9192675 FILE_CREATE

The status bar at the bottom indicates "Total lines 306.565 | Visible lines 161 | Open files: 10 | 15:50 | 06/01/2024".

Práctica memoria RAM

Para esta práctica realizaremos una obtención de memoria RAM de un sistema operativo de Windows. En primer lugar debemos descargar la herramienta que vamos a utilizar para crear la memoria. Para ello descargamos en [**C:\Herramientas>winpmem_mini_x64_rc2.exe nombre_del_fichero.mem**](https://github.com/Velocidex/WinPmem el programa en nuestro ordenador.. Una vez que tengamos el ejecutable abriremos una consola en nuestro sistema y ejecutaremos con permisos de administrador la copia de la memoria RAM con el comando</p></div><div data-bbox=)

El archivo que obtendremos es el llamado **windows_ram.mem**

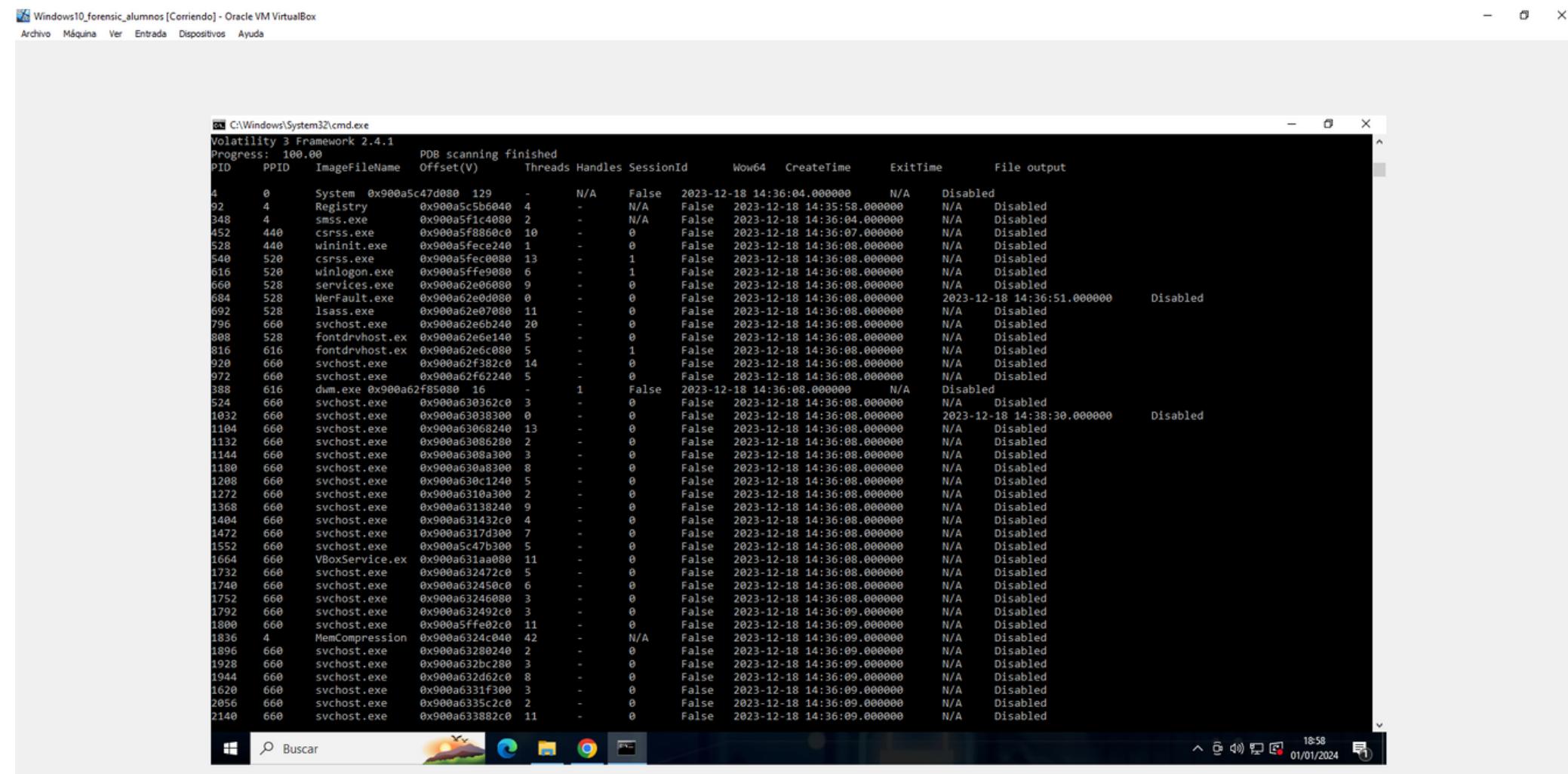
Después de hacer esto descargamos la herramienta volatility <https://www.volatilityfoundation.org/releases-vol3> en nuestro sistema.

Esta herramienta la usaremos para ejecutar una serie de comandos en nuestra adquisición de memoria RAM. El volatility está compuesto de distintos plugings que nos ofrece diversa información en cuanto a procesos, información etc. Usamos el comando

C:\Herramientas\volatility3-2.4.1\volatility3-2.4.1>python vol.py -f

C:\Herramientas\windows_ram.mem windows.pslist.PsList

Para ejecutar el volatility sobre nuestra adquisición y obtener los procesos de la máquina.

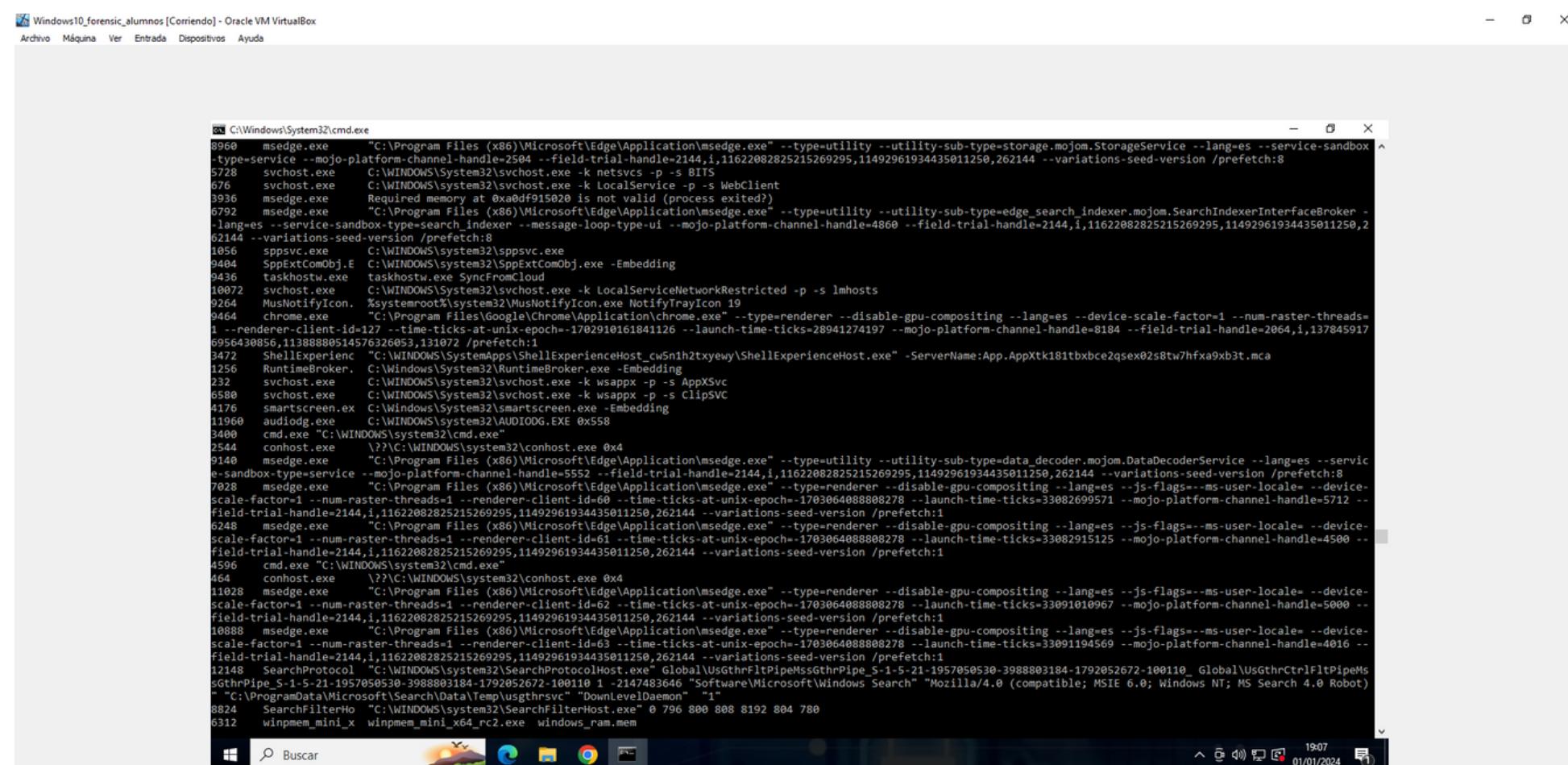


Usamos el comando

C:\Herramientas\volatility3-2.4.1\volatility3-2.4.1>python vol.py -f

C:\Herramientas\windows_ram.mem windows.cmdline.

CmdLine para ver el historial de CMD.



Usando el comando

C:\Herramientas\volatility3-2.4.1\volatility3-2.4.1>python vol.py -f

C:\Herramientas\windows_ram.mem windows.modules.

Modules vemos las liberías kernel del sistema.

```
C:\Windows\System32\cmd.exe
Volatility 3 Framework 2.4.1
Progress: 100.00      PDB scanning finished
Offset Base  Size Name          Path   File output
0x000a5c44e610 0xf08728200000 0x10456000 ntoskrnl.exe \SystemRoot\system32\ntoskrnl.exe    Disabled
0x000a5c44e5e0 0xf08725240000 0x6000 hal.dll \SystemRoot\system32\hal.dll    Disabled
0x000a5c44de10 0xf08725250000 0xb000 kdcm.dll \SystemRoot\system32\kd.dll    Disabled
0x000a5c44ea20 0xf08725210000 0x28000 mupdate.dll \SystemRoot\system32\mupdate_AuthenticAMD.dll Disabled
0x000a5c44ebd0 0xf08725290000 0x6e000 CLFS.SYS \SystemRoot\System32\drivers\CLFS.SYS Disabled
0x000a5c44ed80 0xf08725260000 0x27000 tm.sys \SystemRoot\System32\drivers\tm.sys Disabled
0x000a5c45a050 0xf08725300000 0x1a000 PSHED.dll \SystemRoot\System32\PSHED.dll Disabled
0x000a5c45a210 0xf08725230000 0xc0000 BOOTVID.dll \SystemRoot\System32\BOOTVID.dll Disabled
0x000a5c45a3c0 0xf08725450000 0xd0000 FLTMGR.SYS \SystemRoot\System32\drivers\FLTMGR.SYS Disabled
0x000a5c45a580 0xf08725470000 0xd3000 msrpc.sys \SystemRoot\System32\drivers\msrpc.sys Disabled
0x000a5c45a750 0xf0872524c0000 0x29000 ksecdi.sys \SystemRoot\System32\drivers\ksecdi.sys Disabled
0x000a5c45a850 0xf0872524c000 0x10000 clipsp.sys \SystemRoot\System32\drivers\clipsp.sys Disabled
0x000a5c45a910 0xf08725560000 0xe0000 cmicext.sys \SystemRoot\System32\drivers\cmicext.sys Disabled
0x000a5c45a9a0 0xf08725570000 0x11000 msad.dll \SystemRoot\System32\drivers\msad.dll Disabled
0x000a5c45a9c0 0xf08725590000 0xc0000 ntosext.sys \SystemRoot\System32\drivers\ntosext.sys Disabled
0x000a5c45b010 0xf0872a2c0000 0x80000 CI.dll \SystemRoot\System32\CI.dll    Disabled
0x000a5c45b1e0 0xf0872a2cf0000 0xb0000 cng.sys \SystemRoot\System32\drivers\cng.sys Disabled
0x000a5c45b3a0 0xf0872a2db0000 0xd1000 Wdf01000.sys \SystemRoot\System32\drivers\Wdf01000.sys Disabled
0x000a5c45b4e0 0xf087255a0000 0x13000 WDFLDR.SYS \SystemRoot\System32\drivers\WDFLDR.SYS Disabled
0x000a5c45b570 0xf087255d0000 0x11000 WppRecorder.sys \SystemRoot\System32\drivers\WppRecorder.sys Disabled
0x000a5c45b730 0xf087255c0000 0x1f000 SleepstudyHelper.sys \SystemRoot\System32\drivers\SleepstudyHelper.sys Disabled
0x000a5c45b8f0 0xf0872a9e0000 0x26000 acpix.sys \SystemRoot\System32\Drivers\acpix.sys Disabled
0x000a5c45ba00 0xf08725f00000 0xd0000 msseccore.sys \SystemRoot\System32\Drivers\msseccore.sys Disabled
0x000a5c45bca0 0xf0872a9e0000 0x10000 spagent.sys \SystemRoot\System32\Drivers\spagent.sys Disabled
0x000a5c45b200 0xf0872a8e0000 0xc0000 CPTL.SYS \SystemRoot\System32\drivers\CPTL.SYS Disabled
0x000a5c45c010 0xf08725600000 0xc0000 WHILIB.SYS \SystemRoot\System32\drivers\WHILIB.SYS Disabled
0x000a5c45c370 0xf0872a7fd0000 0xb0000 intelpep.sys \SystemRoot\System32\drivers\intelpep.sys Disabled
0x000a5c45c530 0xf0872b0400000 0x17000 WindowsTrustedRT.sys \SystemRoot\System32\drivers\WindowsTrustedRT.sys Disabled
0x000a5c45c6f0 0xf0872b060000 0xb0000 IntelTA.sys \SystemRoot\System32\drivers\IntelTA.sys Disabled
0x000a5c45c8a0 0xf0872b0700000 0xb0000 WindowsTrustedRTProxy.sys \SystemRoot\System32\drivers\WindowsTrustedRTProxy.sys Disabled
0x000a5c45ca20 0xf0872b080000 0x14000 pcw.sys \SystemRoot\System32\drivers\pcw.sys Disabled
0x000a5c45c2a0 0xf0872b0a0000 0xb0000 msisadvr.sys \SystemRoot\System32\drivers\msisadvr.sys Disabled
0x000a5c45cdd0 0xf0872b0b0000 0x77000 pci.sys \SystemRoot\System32\drivers\pci.sys Disabled
0x000a5c45d010 0xf0872b130000 0x15000 vdrroot.sys \SystemRoot\System32\drivers\vdrroot.sys Disabled
0x000a5c45d1c0 0xf0872b150000 0x2f000 pdc.sys \SystemRoot\System32\drivers\pdc.sys Disabled
0x000a5c45d200 0xf0872b160000 0x19000 pcfmg.sys \SystemRoot\System32\drivers\pcfmg.sys Disabled
0x000a5c45d770 0xf0872b1e0000 0x31000 partmgr.sys \SystemRoot\System32\drivers\partmgr.sys Disabled
0x000a5c45d850 0xf0872b290000 0x19000 volmgr.sys \SystemRoot\System32\drivers\volmgr.sys Disabled
0x000a5c45d540 0xf0872b2b0000 0x63000 volmgrx.sys \SystemRoot\System32\drivers\volmgrx.sys Disabled
0x000a5c46ba10 0xf0872b320000 0x1e000 mountmgr.sys \SystemRoot\System32\drivers\mountmgr.sys Disabled
```

Usamos el comando

C:\Herramientas\volatility3-2.4.1\volatility3-2.4.1>python vol.py -f

C:\Herramientas\windows_ram.mem windows.registry.printkey.

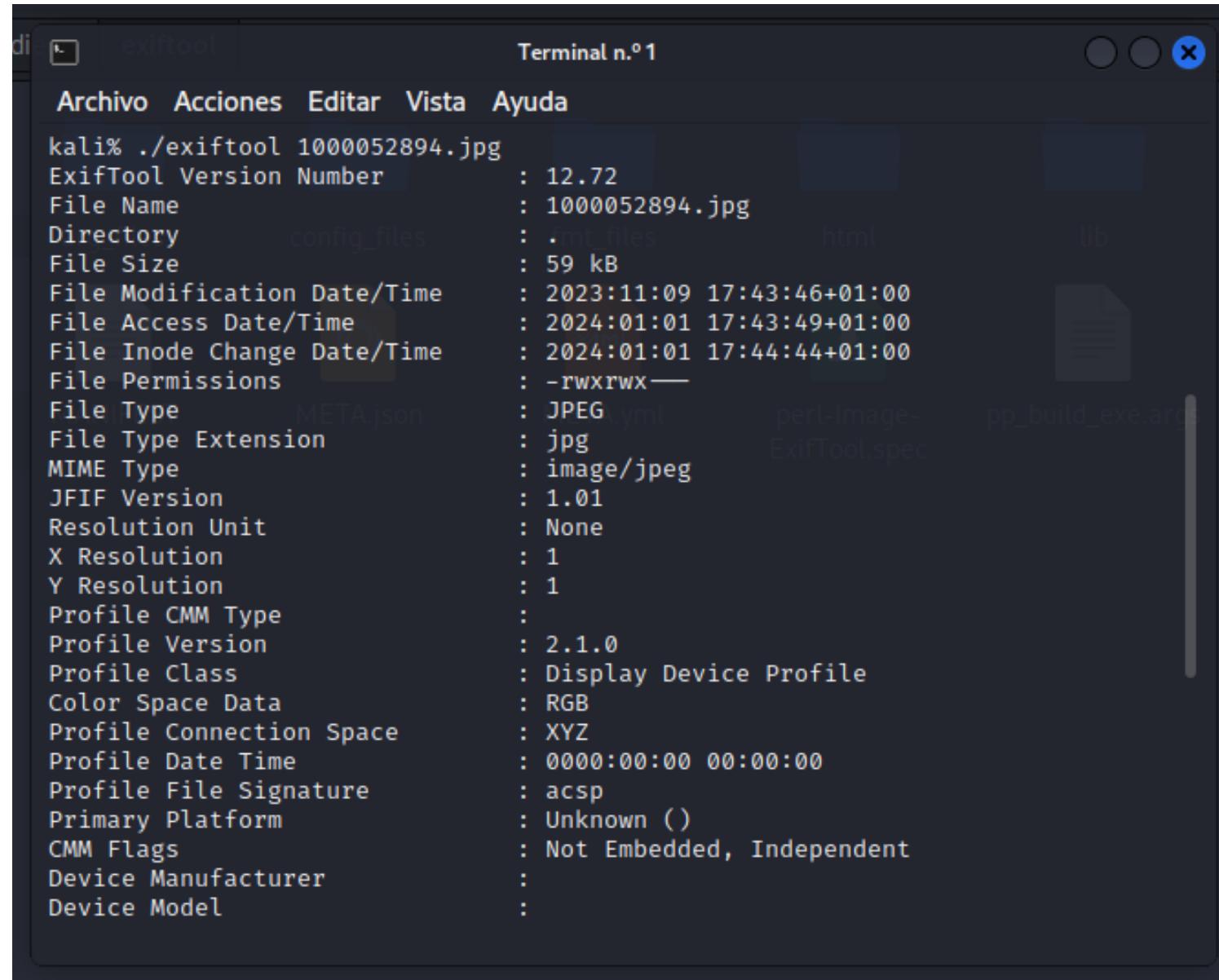
PrintKey para obtener claves que se encuentren en el registro.

```
C:\Windows\System32\cmd.exe
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .wma False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .wmv False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .wpl False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .wl False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .wrl False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .wsb False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .w3f False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .xml False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .xvid False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .xxe False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .z False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .zip False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .zipx False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat .zpl False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat ActivatableClasses False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppID False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat appinstaller.oauth2 False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat Applications False
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppX0en2kacdmv8ydhmtbeayjp27223q6 F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppX0j688mrddm2gsn5y1q8jpxstfsxk7s F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppX0resaq7rSembh496ke39yqc1atfhjr F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppX0t69n30jztar4a12pv0h1x91e8jsacr F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppX0yfn1fhybwjxhemdkvyy841hdgqqy4y F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppX17qce2ahnjz5whsj2zk524farwxv F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppXiaekbsnxkswh8hk1pb4f1s3d2rfbnyb F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppXlapmywg4z9t3tk3nrrn9y0ntjcscg9675 F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppXlyprbth8spa55n2rc6zbh6tcikr1tfsh F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppXzhxyjtgv6vynpg4pmxk7vdgzkp98h F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppX2jm25qtomp2qstv333vv5meik5bf4b F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppX333jbhpaq2hjpyd097614409g4kz15cw F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppX3cx04417yba9f9z7fem54fc937697n6k F
2024-01-01 16:32:41.000000 0xcc058cce0000 Key \??\C:\Users\forensic\AppData\Local\Microsoft\Windows\UsrClass.dat AppX3p914qnpgw4hwj856jw2y286v7d4qnzh F
```

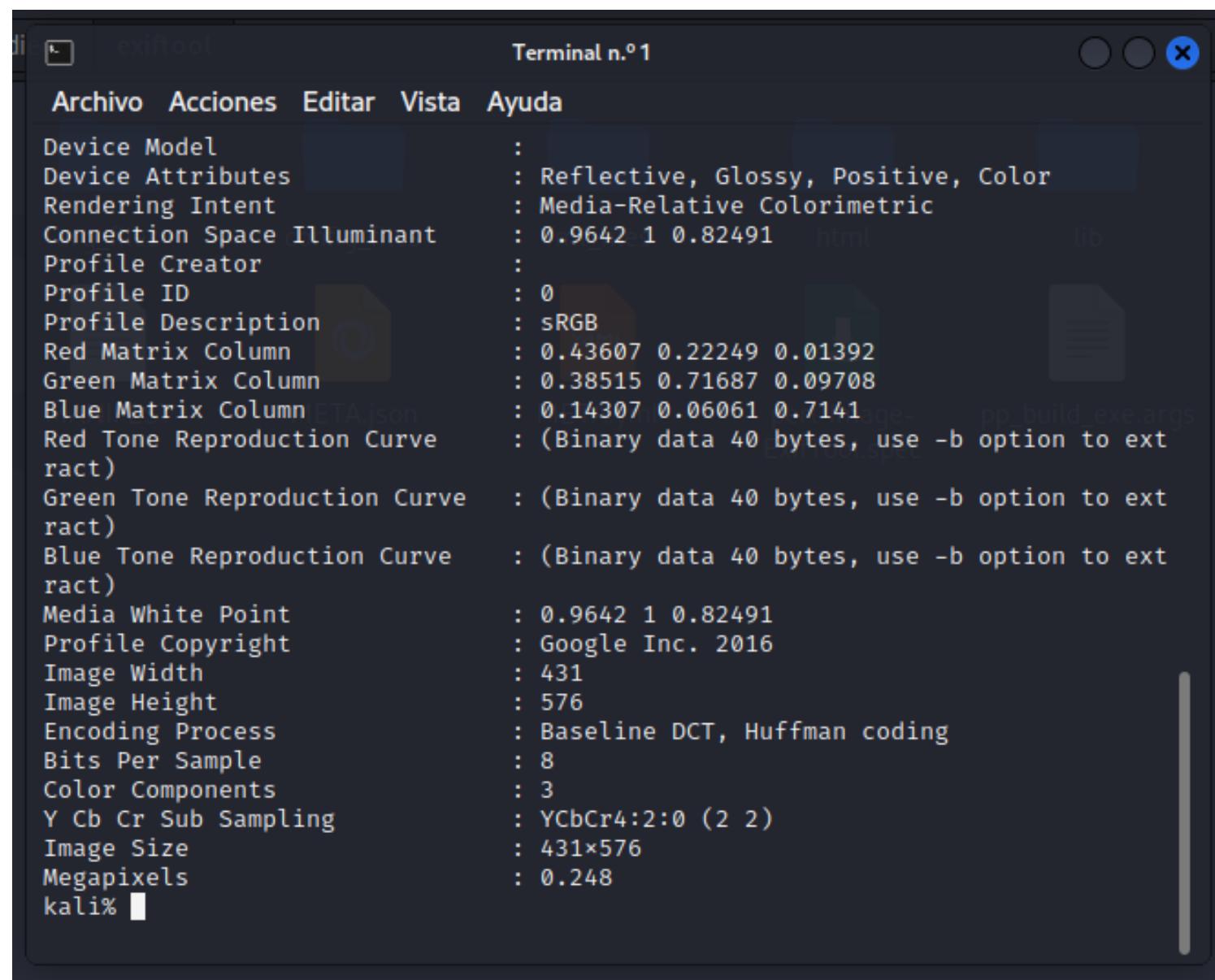
Práctica Metadatos.

Para realizar este ejercicio hemos usado la herramienta de exiftool que hemos descargado haciendo un gitclone <https://github.com/exiftool/exiftool> en nuestra máquina virtual Linux.

En esta máquina virtual se encuentra la foto que queremos analizar y usando la herramienta hemos obtenido una serie de metadatos.



```
Terminal n.º 1
Archivo Acciones Editar Vista Ayuda
kali% ./exiftool 1000052894.jpg
ExifTool Version Number : 12.72
File Name : 1000052894.jpg
Directory : config_files/html/lib
File Size : 59 kB
File Modification Date/Time : 2023:11:09 17:43:46+01:00
File Access Date/Time : 2024:01:01 17:43:49+01:00
File Inode Change Date/Time : 2024:01:01 17:44:44+01:00
File Permissions : -rwxrwx—
File Type : JPEG
File Type Extension : jpg
MIME Type : image/jpeg
JFIF Version : 1.01
Resolution Unit : None
X Resolution : 1
Y Resolution : 1
Profile CMM Type :
Profile Version : 2.1.0
Profile Class : Display Device Profile
Color Space Data : RGB
Profile Connection Space : XYZ
Profile Date Time : 0000:00:00 00:00:00
Profile File Signature : acsp
Primary Platform : Unknown ()
CMM Flags : Not Embedded, Independent
Device Manufacturer :
Device Model :
```



```
Terminal n.º 1
Archivo Acciones Editar Vista Ayuda
Device Model :
Device Attributes : Reflective, Glossy, Positive, Color
Rendering Intent : Media-Relative Colorimetric
Connection Space Illuminant : 0.9642 1 0.82491
Profile Creator :
Profile ID : 0
Profile Description : sRGB
Red Matrix Column : 0.43607 0.22249 0.01392
Green Matrix Column : 0.38515 0.71687 0.09708
Blue Matrix Column : 0.14307 0.06061 0.7141
Red Tone Reproduction Curve : (Binary data 40 bytes, use -b option to extract)
Green Tone Reproduction Curve : (Binary data 40 bytes, use -b option to extract)
Blue Tone Reproduction Curve : (Binary data 40 bytes, use -b option to extract)
Media White Point : 0.9642 1 0.82491
Profile Copyright : Google Inc. 2016
Image Width : 431
Image Height : 576
Encoding Process : Baseline DCT, Huffman coding
Bits Per Sample : 8
Color Components : 3
YCbCr Sub Sampling : YCbCr4:2:0 (2 2)
Image Size : 431x576
Megapixels : 0.248
kali%
```

El objetivo del ejercicio es ver la modificación de estos metadatos al enviar esta imagen a través de aplicaciones o servicios de mensajería y como se modifican. Para ello enviaremos esta imagen por correo electrónico, discord, whatsapp y la descargaremos para analizarla de nuevo.

Aquí podemos observar usando el comando **exiftool -Sort nombredelaimagen.jpg** los datos que nos da la imagen habiéndola descargado del discord:

El objetivo del ejercicio es ver la modificación de estos metadatos al enviar esta imagen a través de aplicaciones o servicios de mensajería y como se modifican. Para ello enviaremos esta imagen por correo electrónico, discord, whatsapp y la descargaremos para analizarla de nuevo.

Aquí podemos observar usando el comando **exiftool -Sort nombrede la imagen.jpg** los datos que nos da la imagen descargada del discord:

```

kali% exiftool -Sort 1000052894.jpg
Bits Per Sample : 8
Blue Matrix Column : 0.14307 0.06061 0.7141
Blue Tone Reproduction Curve : (Binary data 40 bytes, use -b option to extract)
CMM Flags : Not Embedded, Independent
Color Components : 3
Color Space Data : RGB
Connection Space Illuminant : 0.9642 1 0.82491
Device Attributes : Reflective, Glossy, Positive, Color
Device Manufacturer :
Device Model :
Directory :
Encoding Process : Baseline DCT, Huffman Coding
ExifTool Version Number : 12.57
File Access Date/Time : 2024:01:01 18:00:00+01:00
File Inode Change Date/Time : 2024:01:01 18:05:59+01:00
File Modification Date/Time : 2024:01:01 17:59:00+01:00
File Name : 1000052894.jpg
File Permissions : -rwxrwx—
File Size : 6.6 kB
File Type : JPEG
File Type Extension : jpg
Image Height : 576
Image Size : 431x576
Image Width : 431
MIME Type : image/jpeg
Media White Point : 0.9642 1 0.82491
Megapixels : 0.032
Primary Platform : Unknown ()
Profile CMM Type :
Profile Class : Display Device Profile
Profile Connection Space : XYZ
Profile Copyright : Google Inc. 2016
Profile Creator :
Profile Date Time : 2016:01:01 00:00:00+00:00
Profile Description : sRGB
Profile File Signature : acsp
Profile ID : 0
Profile Version : 2.1.0
Red Matrix Column : 0.43607 0.22249 0.01392
Red Tone Reproduction Curve : (Binary data 40 bytes, use -b option to extract)
Resolution Unit : None
Rendering Intent : Media-Relative Colorimetric
YCbCr Sub Sampling : YCbCr4:2:0 (2 2)
kali% 1000052894.jpg

```

Instagram:

```

kali% exiftool -Sort 414759423_338426422381294_2178880924228607223_n.jpg
Bits Per Sample : 8
Color Components : 3
Current IPTC Digest : 1b8b5837b423bf5968a4748243b4b9d5
Directory :
Encoding Process : Progressive DCT, Huffman coding
ExifTool Version Number : 12.57
File Access Date/Time : 2024:01:01 18:04:48+01:00
File Inode Change Date/Time : 2024:01:01 18:04:25+01:00
File Modification Date/Time : 2024:01:01 18:04:24+01:00
File Name : 414759423_338426422381294_2178880924228607223_n.jpg
File Permissions : -rwxrwx—
File Size : 6.6 kB
File Type : JPEG
File Type Extension : jpg
Image Height : 206
Image Size : 154x206
Image Width : 154
JFIF Version : 1.01
MIME Type : image/jpeg
Megapixels : 0.032
Resolution Unit : None
Special Instructions : FBMD0a000a6f010000bd03000033070000ed0700009
X Resolution : 1
YCbCr Sub Sampling : YCbCr4:2:0 (2 2)
Y Resolution : 1
kali%

```

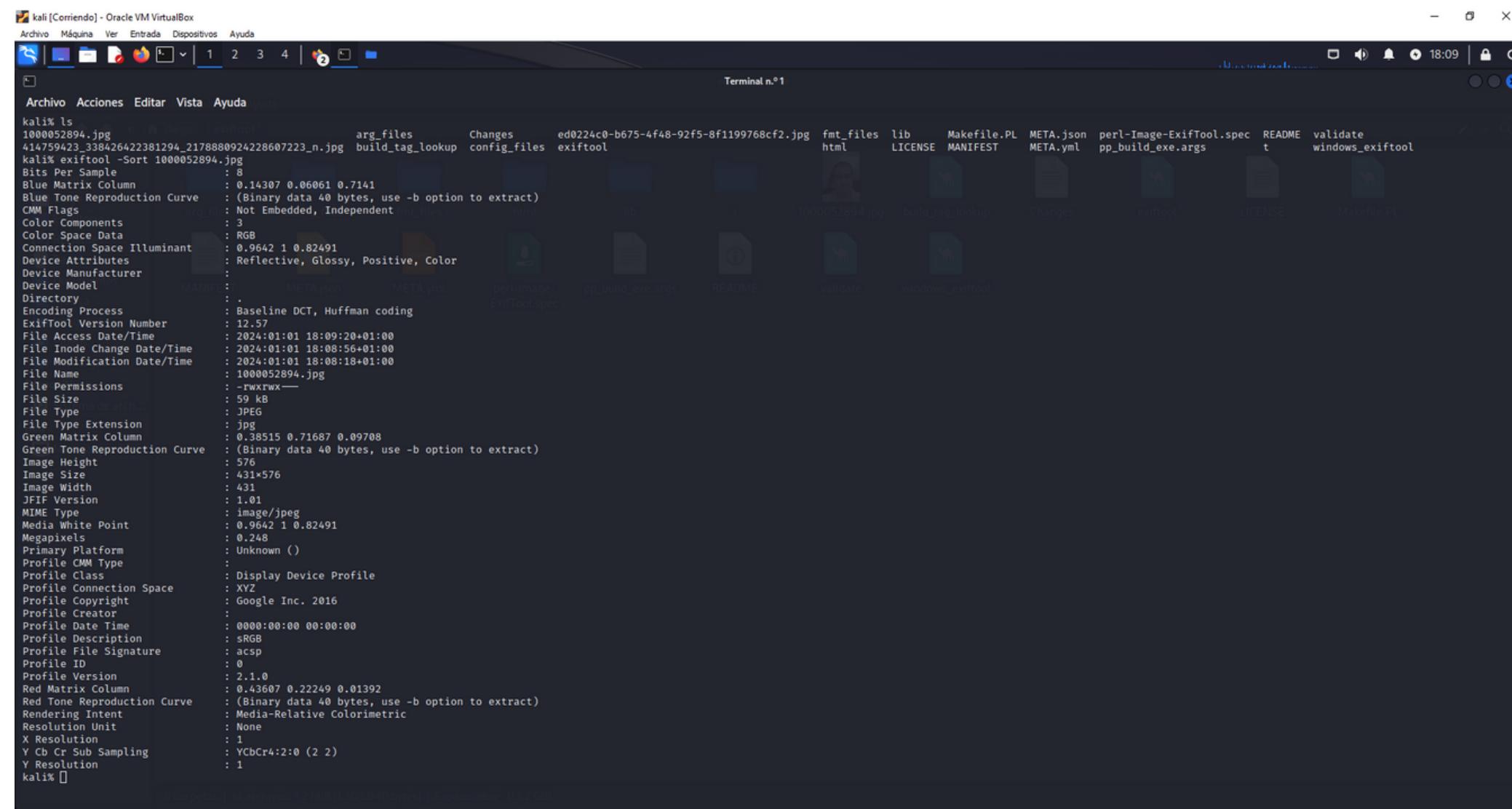
Whatsapp:

```

kali% windows_exiftool -Sort ed0224c0-b675-4f48-92f5-8f1199768cf2.jpg
Bits Per Sample : 8
Blue Matrix Column : 0.14307 0.06061 0.7141
Blue Tone Reproduction Curve : (Binary data 40 bytes, use -b option to extract)
CMM Flags : Not Embedded, Independent
Color Components : 3
Color Space Data : RGB
Connection Space Illuminant : 0.9642 1 0.82491
Device Attributes : Reflective, Glossy, Positive, Color
Device Manufacturer :
Device Model :
Directory :
Encoding Process : Baseline DCT, Huffman coding
ExifTool Version Number : 12.57
File Access Date/Time : 2024:01:01 18:06:33+01:00
File Inode Change Date/Time : 2024:01:01 18:05:36+01:00
File Modification Date/Time : 2024:01:01 18:05:55+01:00
File Name : ed0224c0-b675-4f48-92f5-8f1199768cf2.jpg
File Permissions : -rwxrwx—
File Size : 62 kB
File Type : JPEG
File Type Extension : jpg
Green Matrix Column : 0.38515 0.71687 0.09708
Green Tone Reproduction Curve : (Binary data 40 bytes, use -b option to extract)
Image Height : 576
Image Size : 431x576
Image Width : 431
JFIF Version : 1.01
MIME Type : image/jpeg
Media White Point : 0.9642 1 0.82491
Megapixels : 0.248
Primary Platform : Unknown ()
Profile CMM Type :
Profile Class : Display Device Profile
Profile Connection Space : XYZ
Profile Copyright : Google Inc. 2016
Profile Creator :
Profile Date Time : 2016:01:01 00:00:00
Profile Description : sRGB
Profile File Signature : acsp
Profile ID : 0
Profile Version : 4.3.0
Red Matrix Column : 0.43607 0.22249 0.01392
Red Tone Reproduction Curve : (Binary data 40 bytes, use -b option to extract)
Resolution Unit : None
Rendering Intent : Media-Relative Colorimetric
YCbCr Sub Sampling : YCbCr4:2:0 (2 2)
Y Resolution : 1
kali%

```

Gmail:



```
kali ls
1000052894.jpg arg_files Changes ed0224c0-b675-4f48-92f5-8f1199768cf2.jpg fmt_files lib LICENSE Makefile.PL META.json perl-Image-ExifTool.spec README validate windows_ExifTool
414759423_38426422381294_2178880924228607223_n.jpg build_tag_lookup config_files exiftool html META.yml pp_build_exe.args t windows_ExifTool
kali% exiftool -Sort 1000052894.jpg
Bits Per Sample : 8
Blue Matrix Column : 0.14307 0.06061 0.7141
Blue Tone Reproduction Curve : (Binary data 40 bytes, use -b option to extract)
Color Flags : Direct, Independent
Color Components : 3
Color Space Data : RGB
Connection Space Illuminant : 0.9642 1 0.82491
Device Attributes : Reflective, Glossy, Positive, Color
Device Manufacturer :
Device Model :
Directory :
Embedding Process : Baseline DCT, Huffman coding
ExifTool Version Number : 12.57
File Access Date/Time : 2024:01:01 18:09:20+01:00
File Inode Change Date/Time : 2024:01:01 18:08:56+01:00
File Modification Date/Time : 2024:01:01 18:08:18+01:00
File Name : 1000052894.jpg
File Permissions : -rwxrw-
File Size : 59 kB
File Type : JPEG
File Type Extension : JPG
Green Matrix Column : 0.38515 0.71687 0.09708
Green Tone Reproduction Curve : (Binary data 40 bytes, use -b option to extract)
Image Height : 576
Image Size : 431x576
Image Width : 431
JFIF Version :
MIME Type : image/jpeg
Media White Point : 0.9642 1 0.82491
Megapixels : 0.248
Primary Platform :
Profile CMW Type :
Profile Class : Display Device Profile
Profile Connection Space : XYZ
Profile Copyright : Google Inc. 2016
Profile Creator :
Profile Date Time : 0000:00:00 00:00:00
Profile Description :
Profile File Signature :
Profile ID :
Profile Version : 2.1.0
Red Matrix Column : 0.42507 0.22249 0.01392
Red Tone Reproduction Curve : (Binary data 40 bytes, use -b option to extract)
Rendering Intent : Media-Relative Colorimetric
Resolution Unit : None
X Resolution : 1
YCbCr Sub Sampling : YCbCr4:2:0 (2 2)
Y Resolution : 1
kali%
```

Como se puede comprobar cada servicio de mensajería elimina, modifica o añade datos distintos a la misma foto.

