

## **Tarea 2.1.2 Recopilación de evidencias Volátiles en Windows**



Trabajo realizado por:  
Samuel Romero de los Reyes

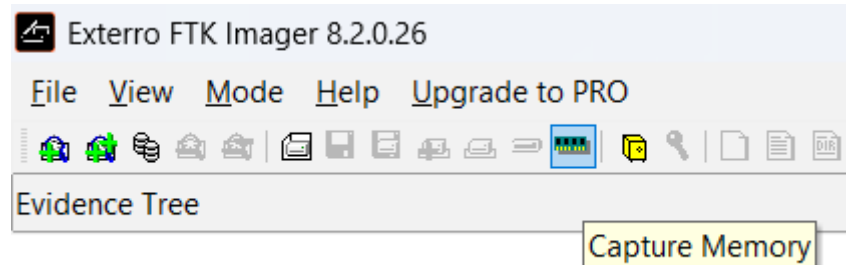
## Índice

<b>1.- Adquisición forense de memoria RAM (cuatro herramientas).....</b>	<b>3</b>
1a.- FTK Imager.....	3
1b.- Magnet RAM Capture.....	4
1c.- DumpIT.....	5
1d.- Winpmem.....	6
<b>2.- Extracción de cadenas y listado de correos electrónicos.....</b>	<b>6</b>
2.1.- Extracción de cadena.....	6
2.2.- Filtrar correos electrónicos ordenados alfabéticamente.....	6
2.3.- Filtrar IPs.....	6

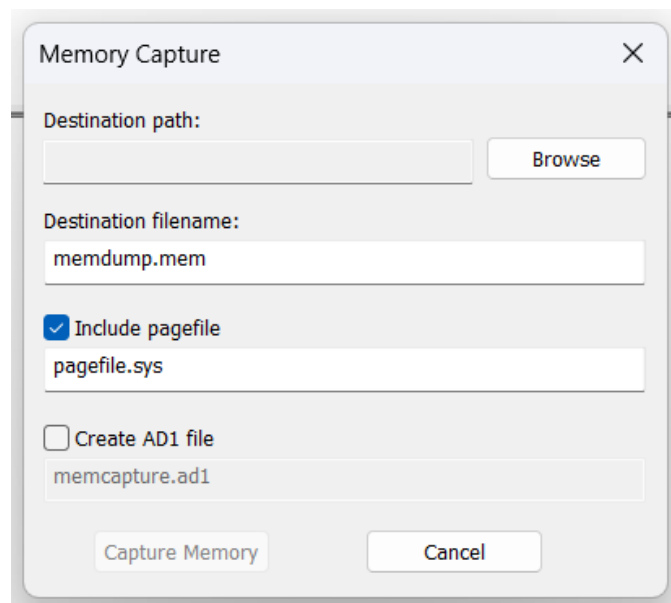
# 1.- Adquisición forense de memoria RAM (cuatro herramientas)

## 1a.- FTK Imager

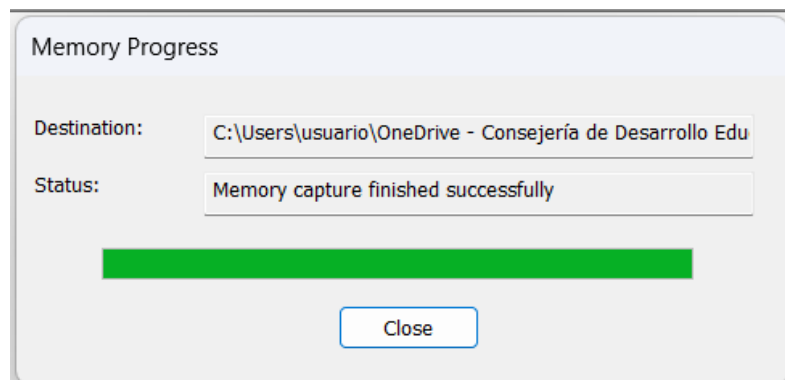
Le damos al dibujo de la RAM para capturar la memoria






Incluimos include pagefile (para obtener una copia completa) y capturamos la memoria.



Esperamos a que termine de cargar.



Vemos que se ha copiado todo.

	memdump.mem		27/10/2025 19:24	Archivo MEM	36.429.824 ...
	pagefile.sys		27/10/2025 19:24	Archivo de sistema	2.097.152 ...

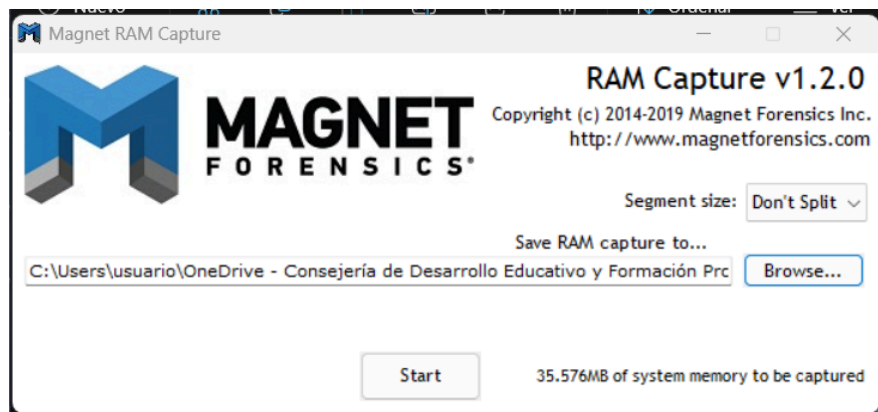
Sacamos el SHA-256 mediante PowerShell.

```
PS C:\Users\usuario\OneDrive - Consejería de Desarrollo Educativo y Formación Profesional\Escritorio\archivosSR> Get-FileHash -Algorithm SHA256 memdump.mem | Format-List

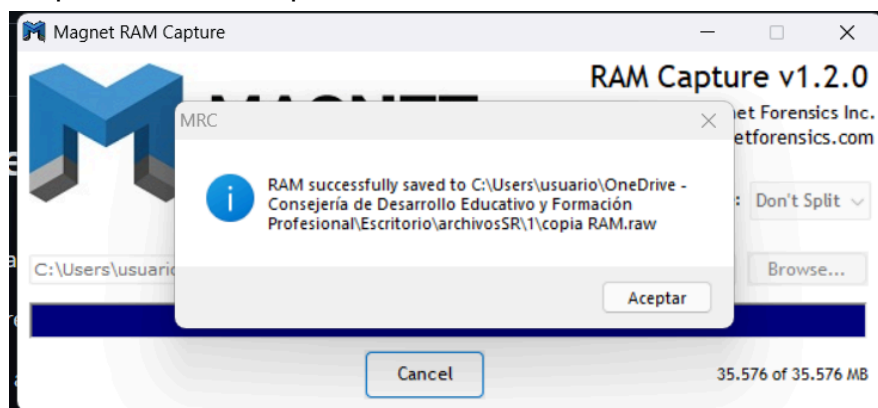
Algorithm : SHA256
Hash      : 0C2B71B650F86ED3F8F457130DC593C37B6F8C51EF6FA9B41D182ACCB8FCD0DC
Path      : C:\Users\usuario\OneDrive - Consejería de Desarrollo Educativo y Formación Profesional\Escritorio\archivosSR\memdump.mem
```

## 1b.- Magnet RAM Capture

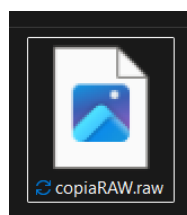
Iniciamos la copia estableciendo la ruta y el nombre.



Esperamos a que termine la copia.



Vemos si está la copia.



## Calculamos SHA-256

```
PS C:\Users\usuario\OneDrive - Consejería de Desarrollo Educativo y Formación Profesional\Escritorio\archivosSR\1> Get-FileHash -Algorithm SHA256 copiaRAW.raw | Format-List

Algorithm : SHA256
Hash       : A21C0199EAC26F3EF46FC45B216E41B357417C1B3B2B4400D5D92F09F61E9003
Path       : C:\Users\usuario\OneDrive - Consejería de Desarrollo Educativo y Formación Profesional\Escritorio\archivosSR\1\copiaRAW.raw
```

## 1c.- DumpIT

Ejecutamos el “.exe” y se iniciará a capturar la memoria.

```
All rights reserved.

Thanks for using DumpIt! Always use Microsoft crash dumps!

Destination path:      \??\C:\WINDOWS\system32\40c2bafac22d-20251027-180818.dmp
Computer name:         40c2bafac22d

--> Proceed with the acquisition ? [y/n] y

[+] Information:
Dump Type:             Microsoft Crash Dump

[+] Machine Information:
Windows version:       10.0.26100
MachineId:             DE231F07-751A-EF11-80F3-40C2BAFAC22D
TimeStamp:             134060621020400993
Cr3:                   0x1ae000
KdCopyDataBlock:       0xffffffff805c6d69f34
KdDebuggerData:         0xffffffff805c7601040
KdpDataBlockEncoded:   0xffffffff805c7665c88

Current date/time:     [2025-10-27 (YYYY-MM-DD) 18:08:22 (UTC)]
+ Processing... Done.

Acquisition finished at: [2025-10-27 (YYYY-MM-DD) 18:09:58 (UTC)]
Time elapsed:          1:36 minutes:seconds (96 secs)

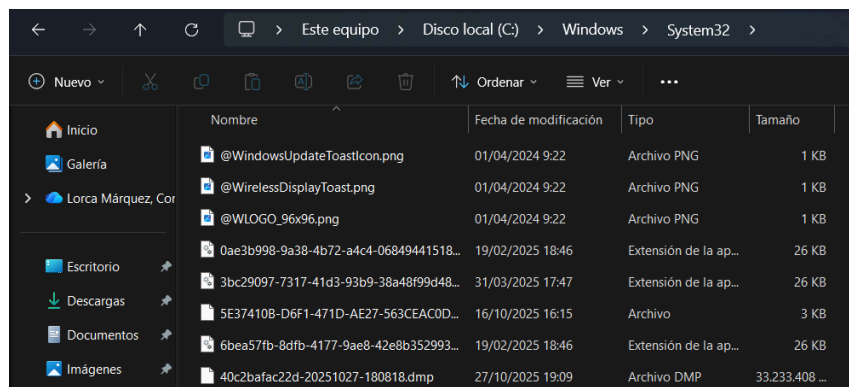
Created file size:     34031009792 bytes (32454 Mb)
Total physical memory size: 32348 Mb

NtStatus (troubleshooting): 0x00000000
Total of written pages: 8281189
Total of inaccessible pages: 0
Total of accessible pages: 8281189

SHA-256: 267106F70E21F7E5A722A21889ED49642AAB609BC1F97B6702F80AF40D2DF586

JSON path:             C:\WINDOWS\system32\40c2bafac22d-20251027-180818.json
```

Vemos que se ha realizado correctamente.



## 1d.- Winpmem

Ejecutamos desde PowerShell.

```
PS C:\Users\usuario\Downloads> .\winpmem_mini_x64_rc2.exe mem.raw
WinPmem64
Extracting driver to C:\Users\usuario\AppData\Local\Temp\pme2A21.tmp
Driver Unloaded.
Deleting C:\Users\usuario\AppData\Local\Temp\pme2A21.tmp
Driver Unloaded.
```

Y ahora obtenemos el SHA-256.

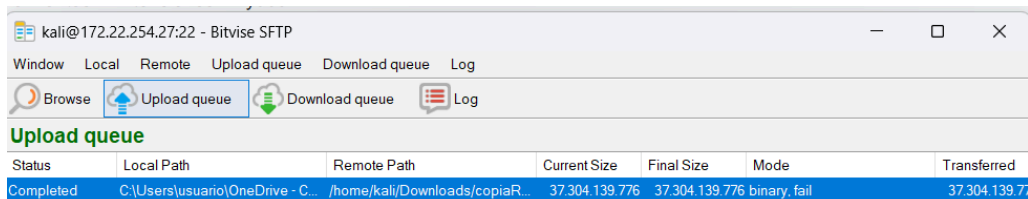
```
PS C:\Users\usuario\Downloads> Get-FileHash -Algorithm SHA256 mem.raw | Format-List

Algorithm : SHA256
Hash      : E3B0C44298FC1C149AFBF4C8996FB92427AE41E4649B934CA495991B7852B855
Path      : C:\Users\usuario\Downloads\mem.raw
```

## 2.- Extracción de cadenas y listado de correos electrónicos

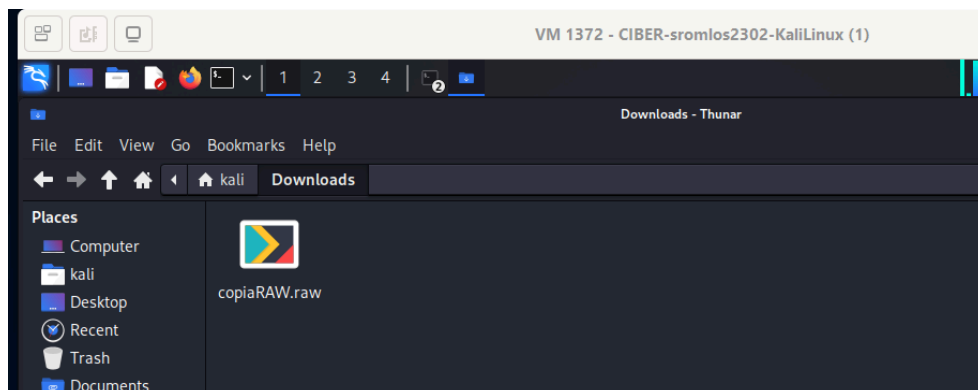
### 2.1.- Extracción de cadena

Me lo pasó para que vaya más a Kali mediante SSH.



Status	Local Path	Remote Path	Current Size	Final Size	Mode	Transferred
Completed	C:\Users\usuario\OneDrive - C...	/home/kali/Downloads/copiaR...	37.304.139.776	37.304.139.776	binary, fail	37.304.139.77

Vemos que lo tenemos en la carpeta de descarga.



## 2.2.- Filtrar correos electrónicos ordenados alfabéticamente

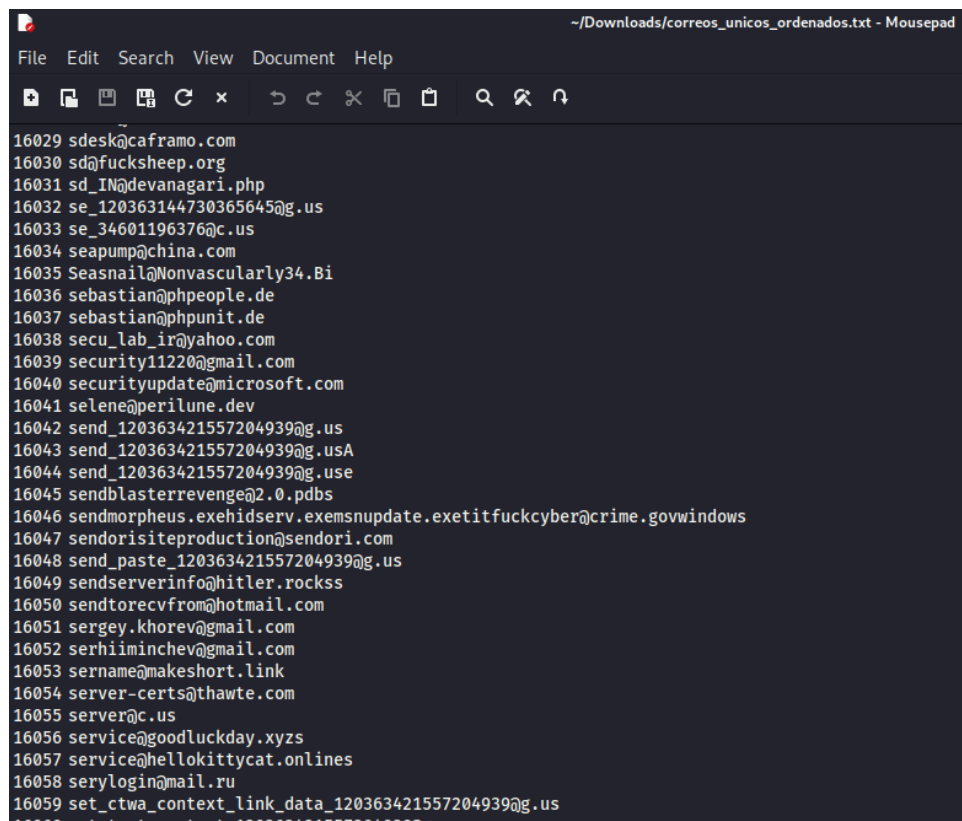
Para filtrar los correos electrónicos, abrimos PowerShell e introducimos el siguiente código:

```
grep -E -o '[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,}' archivoRAM.txt | sort -u > correos_unicos_ordenados.txt
```



```
kali@kali: ~/Downloads
File Actions Edit View Help
(kali@kali)~[~/Downloads]
$ grep -E -o '[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,}' archivoRAM.txt | sort -u > correos_unicos_ordenados.txt
```

Vemos el “.txt”.



```
~/Downloads/correos_unicos_ordenados.txt - Mousepad
File Edit Search View Document Help
16029 sdesk@caframo.com
16030 sd@fucksheep.org
16031 sd_IN@devanagari.php
16032 se_120363144730365645@g.us
16033 se_34601196376@c.us
16034 seapump@china.com
16035 Seasnail@Nonvascularly34.Bi
16036 sebastian@ppeople.de
16037 sebastian@phpunit.de
16038 secu_lab_ir@yahoo.com
16039 security11220@gmail.com
16040 securityupdate@microsoft.com
16041 selene@perilune.dev
16042 send_120363421557204939@g.us
16043 send_120363421557204939@g.usA
16044 send_120363421557204939@g.usE
16045 sendblasterrevenge@2.0.pdbs
16046 sendmorpheus.exehidserv.exemsnupdate.exetitfuckcyber@crime.govwindows
16047 sendorisiteproduction@sendori.com
16048 send_paste_120363421557204939@g.us
16049 sendserverinfo@hitler.rockss
16050 sendtorecvfrom@hotmail.com
16051 sergey.khorev@gmail.com
16052 serhiimincev@gmail.com
16053 sername@makeshort.link
16054 server-certs@thawte.com
16055 server@c.us
16056 service@goodluckday.xyzs
16057 service@hellokittycat.onlines
16058 serylogin@mail.ru
16059 set_ctwa_context_link_data_120363421557204939@g.us
16060 set_text_content_120363421557204939@g.us
```

## 2.3.- Filtrar IPs

Para filtrar las IPs, abrimos PowerShell e introducimos el siguiente código:

```
grep -oP '(?:25[0-5]|2[0-4]\d|[01]?\d?\d)\.(?:25[0-5]|2[0-4]\d|[01]?\d?\d)' archivoRAM.txt | sort -u > ips_ipv4_unicas.txt
```



```
(kali@kali)~[~/Downloads]
$ grep -oP '(?:25[0-5]|2[0-4]\d|[01]?\d?\d)\.(?:25[0-5]|2[0-4]\d|[01]?\d?\d)' archivoRAM.txt | sort -u > ips_ipv4_unicas.txt
```

Y ahora vemos el “.txt”.

```
~/Downloads/ips_ipv4_unicas.txt - Mousepad
File Edit Search View Document Help
[Icons] [Icons] [Icons] [Icons] [Icons] [Icons] [Icons] [Icons] [Icons] [Icons] [Icons] [Icons]
1 0.0.0.0
2 0.0.0.00
3 00.0.0.0
4 0.0.0.000
5 000.0.0.0
6 00.00.00.00
7 0.0.0.01
8 00.002.88.11
9 0.0.0.04
10 0.0.0.064
11 0.0.0.1
12 0.00.2.4
13 0.0.0.26
14 0.0.026.0
15 0.0.027.0
16 0.0.027.1
17 0.0.028.0
18 0.0.0.46
19 0.0.050.0
20 0.0.0.8
21 0.0.1.0
22 0.0.10.0
23 0.0.1.000
24 0.0.100.0
25 00.100.1.1
26 00.100.1.25
27 00.100.1.3
28 00.1.00.2
29 001.018.02.02
30 0.01.02.07
```