



Swascan
TINEXTA GROUP

BiBi Wiper: analisi malware

www.swascan.com

Sommario

Introduzione.....	3
Analisi statica e malware assessment.....	3
Analisi dinamica e second malware assessment.....	13
Debugging.....	38
IOCs:.....	49
Regola YARA.....	49
CONCLUSIONI:.....	50
Riferimenti:.....	50

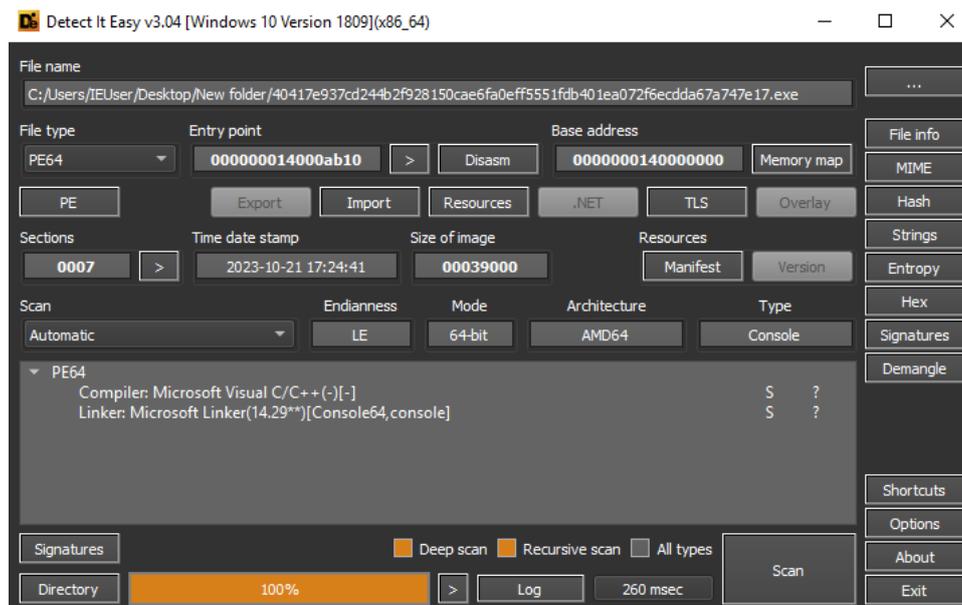
Introduzione

BiBi Wiper è un malware di tipo “distruttivo” utilizzato nel conflitto Israele – Hamas dagli attivisti del gruppo terroristico sunnita. Dal 30 Ottobre 2023 il threat sta infettando anche sistemi operativi Unix, nonostante la variante più utilizzata sia quella Windows, sottoposta ad analisi nel presente articolo.

L’artefatto, similamente a quanto è avvenuto durante la guerra Russo-Ucraina, è stato utilizzato come strumento di guerra ibrida atto ad effettuare azioni distruttive nei confronti delle infrastrutture critiche di Israele, contribuendo di fatto all’offensiva militare e strategica di Hamas. La minaccia, eseguendo una sovrascrittura ed una fase di “locking” dei files (senza però chiederne alcun riscatto) pone BiBi Wiper in una condizione diversa da una minaccia Ransomware. L’unico obiettivo del Wiper è quello di rendere i dati dei target systems non accessibili e non utilizzabili. [0]

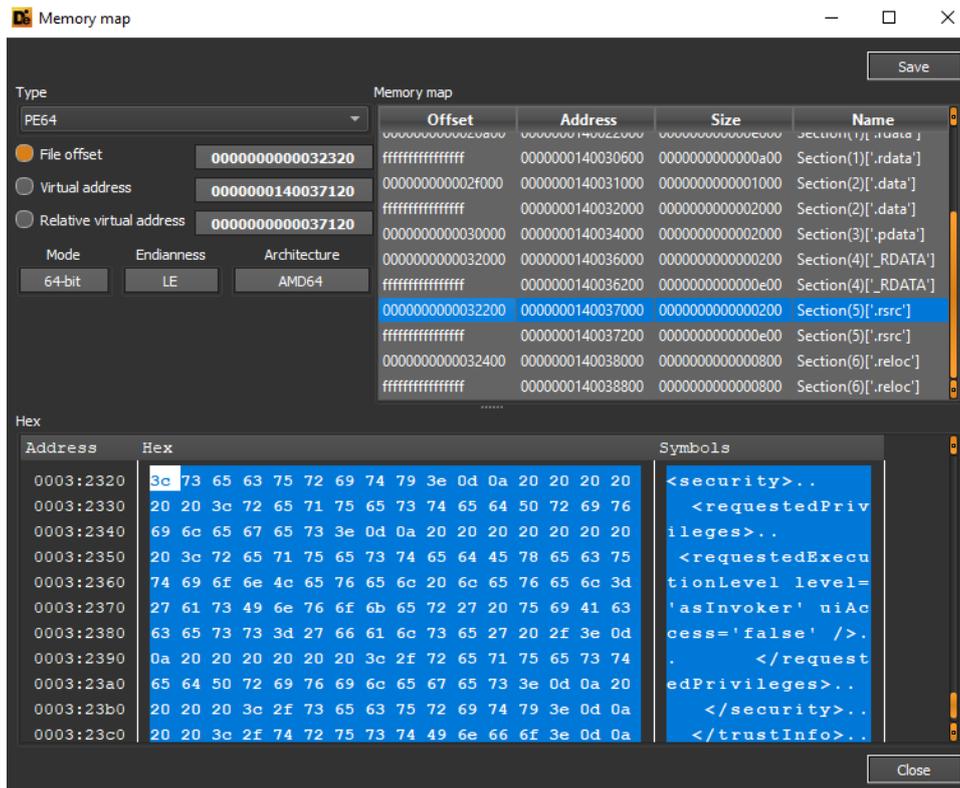
Analisi statica e malware assessment

Il sample analizzato ha come hash **e26bba0304f14ef96beb60376791d32c**, ed è stato sviluppato in C++.



```
File name: C:/Users/IEUser/Desktop/New folder/40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ec
Size: 207872(203.00 kB)
MD5: e26bba0304f14ef96beb60376791d32c
SHA1: 24f6785ca2e82d1d1d61f4cb01d5e753f80445cf
Entropy: 6.19335(not packed)
Operation system: Windows(Vista)
Architecture: AMD64
Mode: 64-bit
Type: Console
Endianness: LE
Entry point(Address): 000000014000ab10
Entry point(Offset): 9f10
Entry point(Relative address): ab10
Entry point(Bytes): 4883ec28e89f0600004883c428e972feffffccccc4883ec284d8b4138488bca498bd1
Entry point(Signature): 4883ec..e8.....4883c4..e9.....cccc4883ec..4d8b41..488bca498bd1
Entry point(Signature)(Rel): 4883ec..e8$$$$$$$$48895c24..55488bec4883ec..488b05.....48bb.....
```

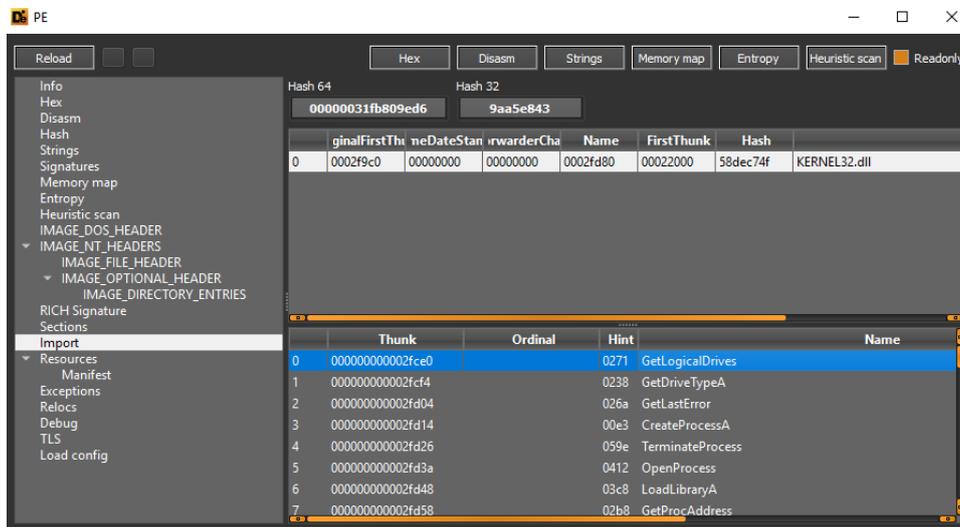
All'interno della sezione `.rsrc` (la quale contiene i dettagli del file di metadati *manifest* ed altre risorse) possiamo evidenziare un'impostazione di execution `"asInvoker"`, pertanto il threat viene lanciato con i medesimi privilegi e permessi di sicurezza del processo parent.



The screenshot shows the Memory map tool interface. The 'File offset' is set to 000000000032320. The 'Memory map' table lists sections, with Section(5) ['.rsrc'] highlighted. Below, the hex data for the .rsrc section is shown, with the following XML snippet visible in the Symbols pane:

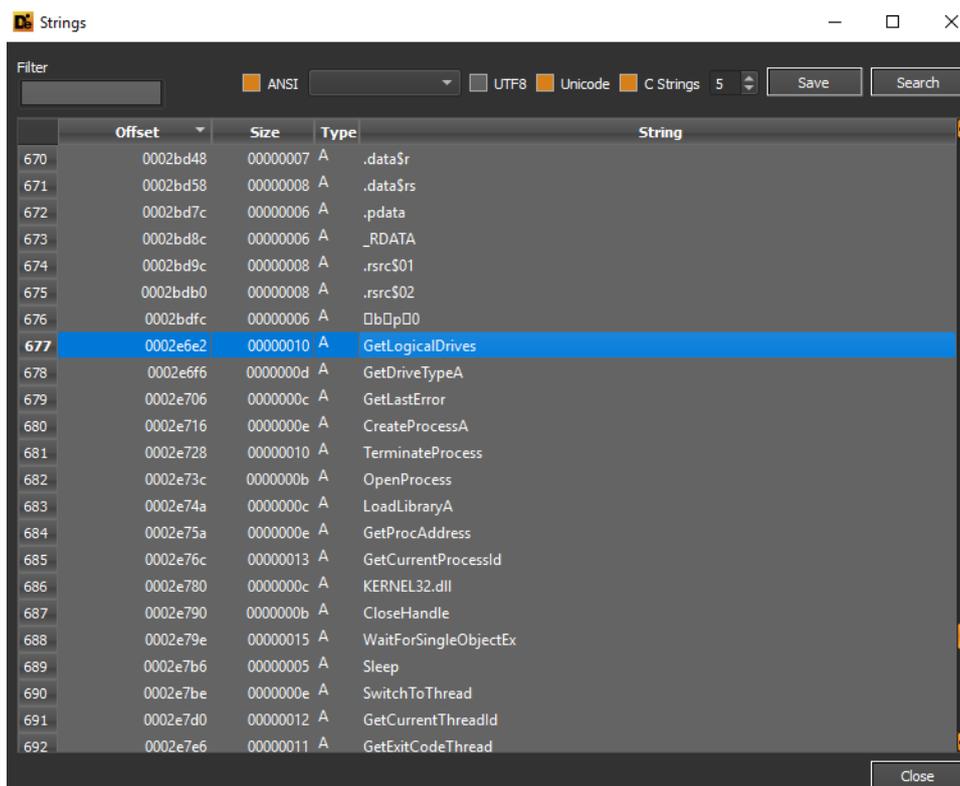
```
<security>..
  <requestedPrivileges>..
    <requestedExecutionLevel level=
      'asInvoker' uiAc
      cess='false' />.
  </request
edPrivileges>..
</security>..
</trustInfo>..
```

Le funzioni importate mediante la libreria *KERNEL32.dll* fanno riferimento a drives enumeration, creazione ed apertura di processi e richiamo di librerie esterne mediante *LoadLibraryA*:



PE Explorer - Import section

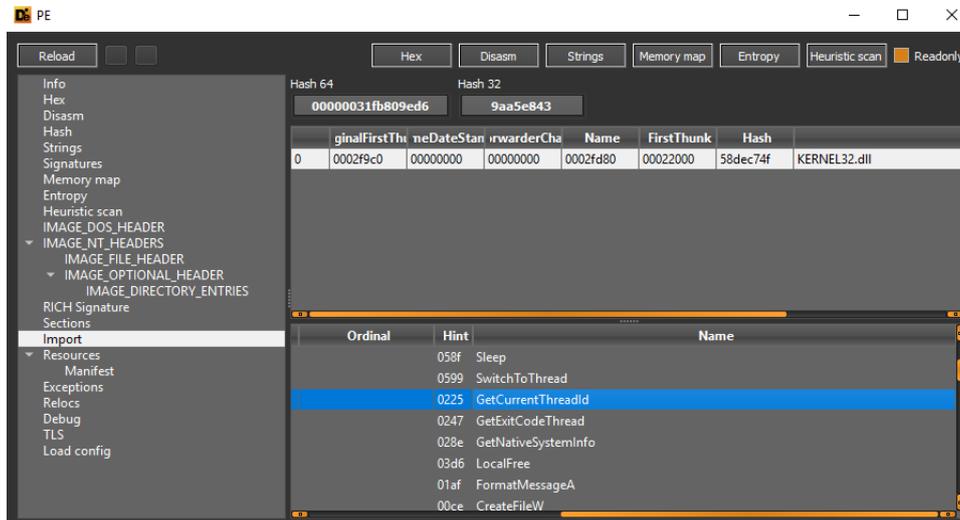
Thunk	Ordinal	Hint	Name
0	000000000002fce0	0271	GetLogicalDrives
1	000000000002fcf4	0238	GetDriveTypeA
2	000000000002fd04	026a	GetLastError
3	000000000002fd14	00e3	CreateProcessA
4	000000000002fd26	059e	TerminateProcess
5	000000000002fd3a	0412	OpenProcess
6	000000000002fd48	03c8	LoadLibraryA
7	000000000002fd58	02b8	GetProcAddress



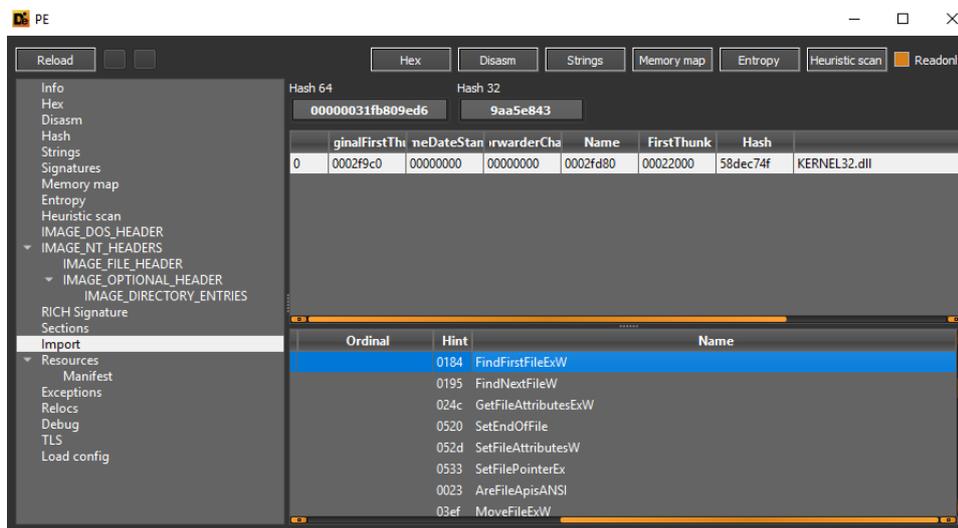
PE Explorer - Strings section

Offset	Size	Type	String
670	0002bd48	A	.data\$r
671	0002bd58	A	.data\$rs
672	0002bd7c	A	.pdata
673	0002bd8c	A	._RDATA
674	0002bd9c	A	.rsrc\$01
675	0002bdb0	A	.rsrc\$02
676	0002bdfc	A	▯▯▯▯▯▯▯▯
677	0002e6e2	A	GetLogicalDrives
678	0002e6f6	A	GetDriveTypeA
679	0002e706	A	GetLastError
680	0002e716	A	CreateProcessA
681	0002e728	A	TerminateProcess
682	0002e73c	A	OpenProcess
683	0002e74a	A	LoadLibraryA
684	0002e75a	A	GetProcAddress
685	0002e76c	A	GetCurrentProcessId
686	0002e780	A	KERNEL32.dll
687	0002e790	A	CloseHandle
688	0002e79e	A	WaitForSingleObjectEx
689	0002e7b6	A	Sleep
690	0002e7be	A	SwitchToThread
691	0002e7d0	A	GetCurrentThreadId
692	0002e7e6	A	GetExitCodeThread

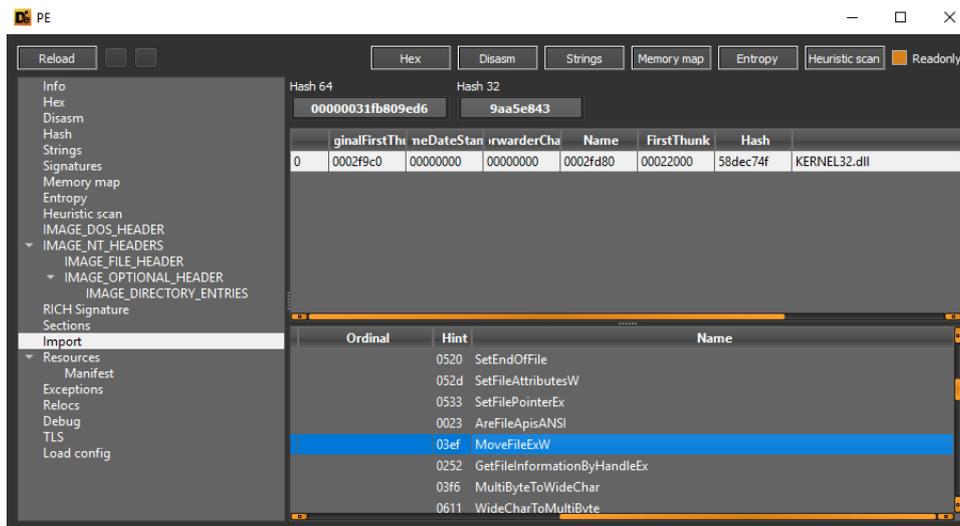
Vengono utilizzate funzioni di threads management per gestire esecuzioni concorrenti:



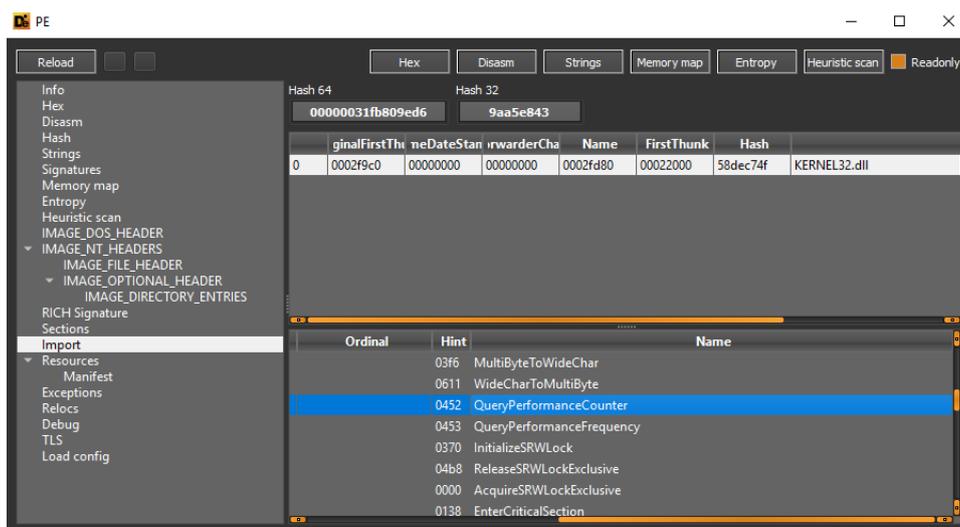
A seguire evidenze di files enumerations loop di ottenimento dei files ed attributi dei medesimi, nonché un puntamento mediante la funzione *SetFilePointerEx*. Quest'ultima è largamente utilizzata da minacce con funzionalità di riferimento a files esterni in quanto permette una gestione più granulare e specifica della location di puntamento.



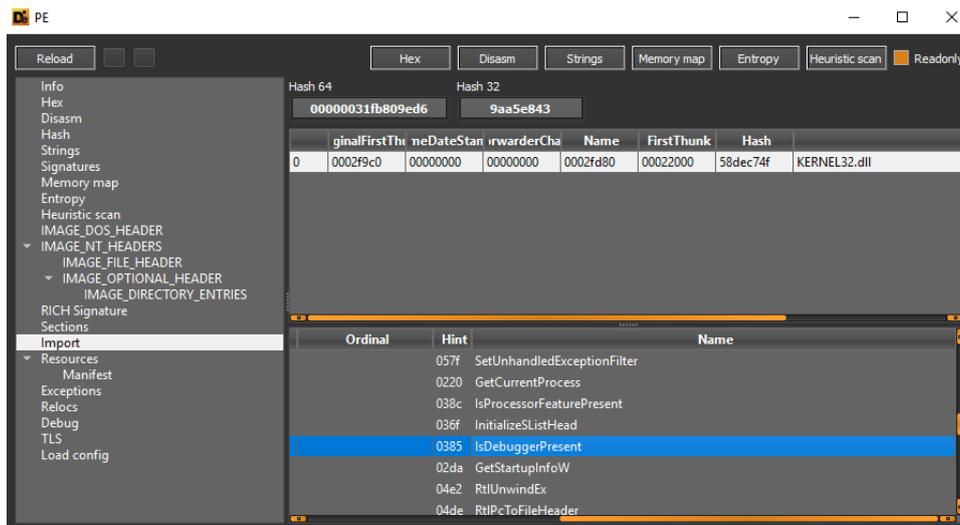
I files vengono rinominati con estensione **.BiBi** dopo che gli stessi sono stati resi inaccessibili attraverso un processo di sovrascrittura:



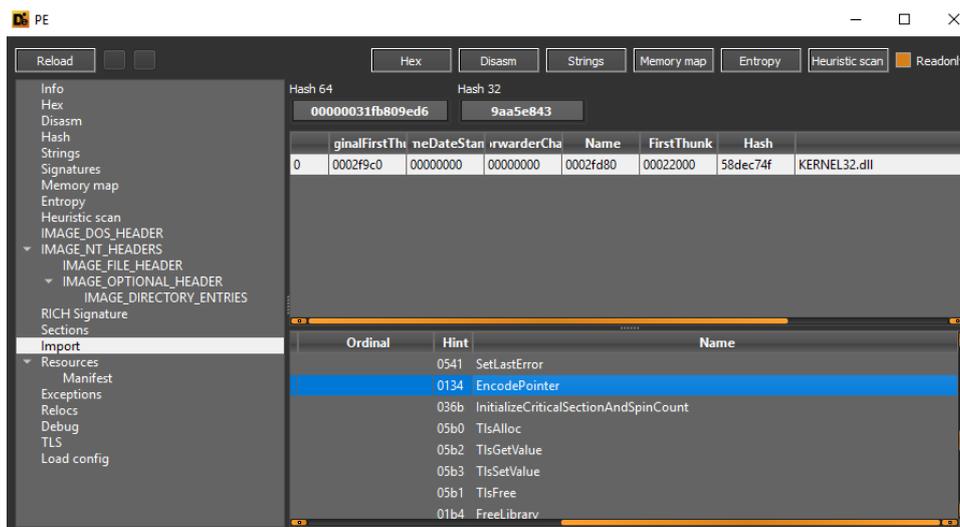
Vengono inoltre ottenuti i dettagli del Performance Counter e della frequenza di esecuzione dei componenti della CPU, tali informazioni possono permettere ad un threat di individuare un eventuale ambiente virtualizzato, come ad esempio macchine virtuali o sandboxes:



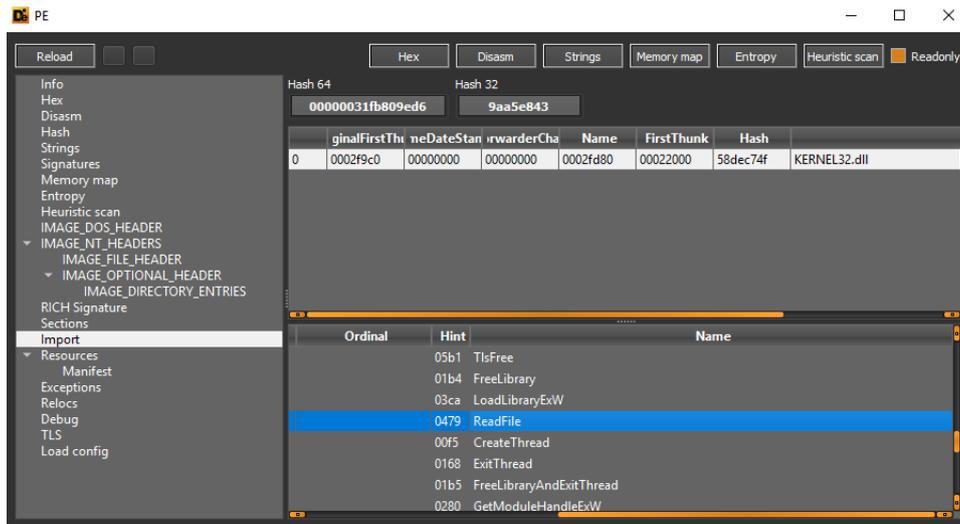
Si noti la funzione di debugger checking *IsDebuggerPresent*, che permette di evitare il monitoraggio e il tracciamento dell'esecuzione del processo stesso attraverso breakpoints e code browsing tools:



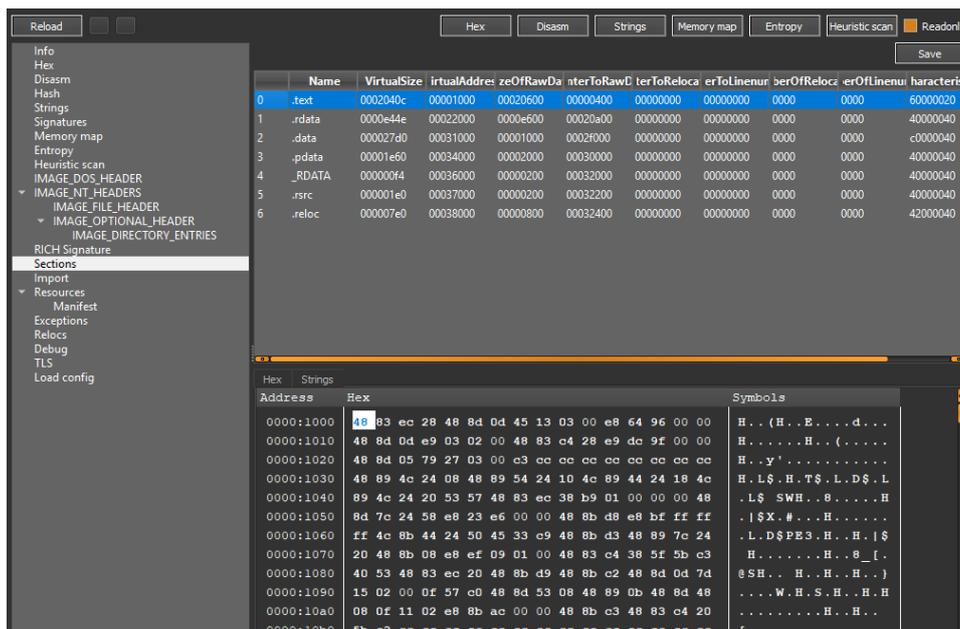
I valori dei puntatori utilizzati vengono codificati mediante il richiamo della funzione *EncodePointer*. I puntatori permettono di far riferimento ad ulteriori variabili ed oggetti all'interno di funzioni eseguite, nel caso specifico non vi è contezza precisa dei valori e attributi riferiti in quanto codificati.

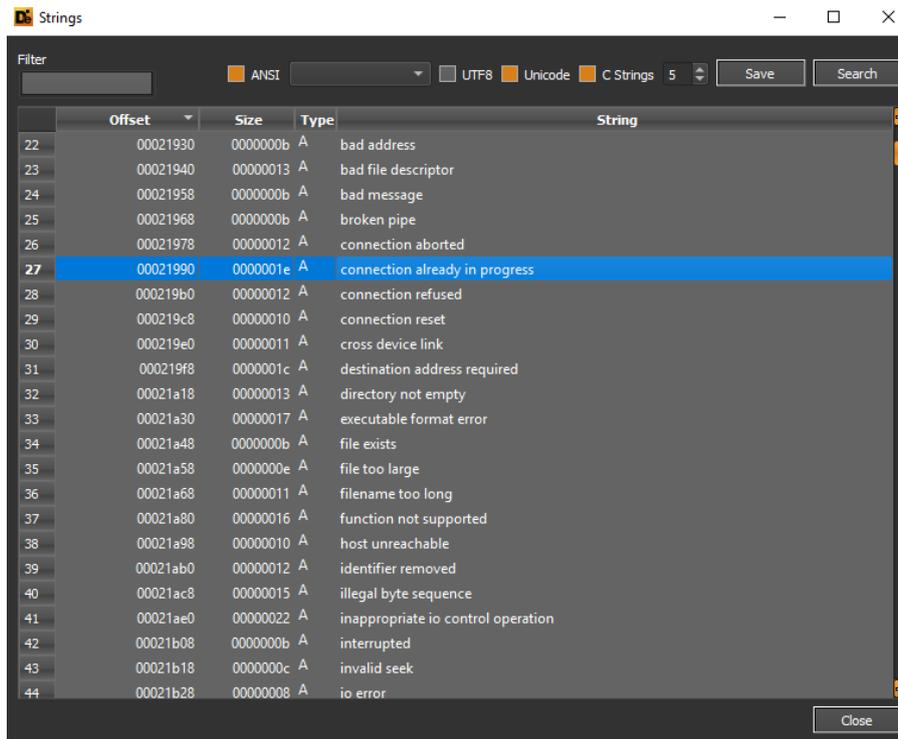


In un contesto concorrente vengono letti i files presi in considerazione:

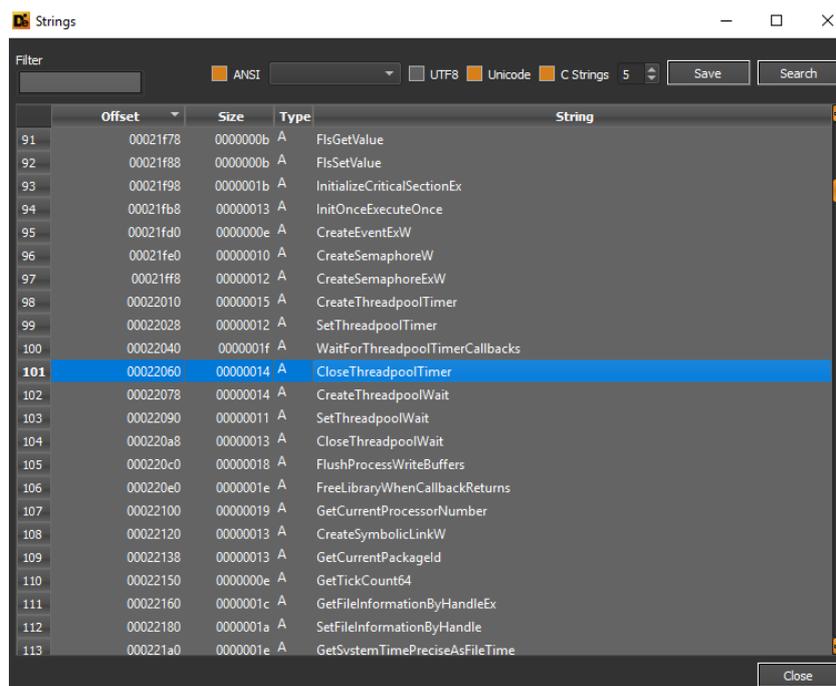


Qui i dettagli delle sezioni del Portable Executable in questione. La sezione principale risulta essere *.text*, la quale contiene le istruzioni direttamente eseguite dalla CPU.

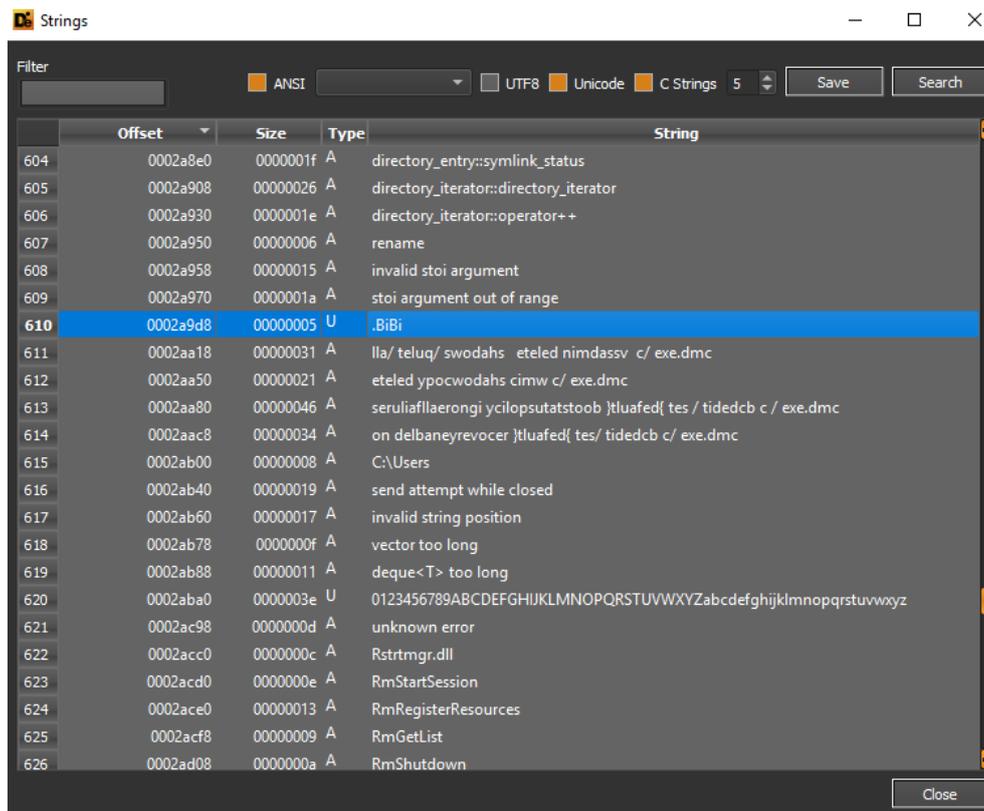




Qui i dettagli di gestione di oggetti concorrenti e resources management, nello specifico multithreading e *semaphores*. Gli oggetti *semaphores* permettono l'utilizzo di risorse con un accesso esclusivo, evitando pertanto un accesso simultaneo da parte di più processi alla medesima risorsa:



Di seguito evidenze relative alla fase di *directory iterator*, estensione .BiBi appesa ai files resi inaccessibili, l'individuazione dei comandi CMD di booting settings al fine di disabilitare il modulo di Windows Automatic Repair ed il controllo di eventuali failures di OS booting. I comandi CMD in questione sono in forma *reversed* (ovvero scritti al rovescio) all'interno delle stringhe estraibili. Vengono inoltre utilizzate istanze di Restart Manager per gestire lo status e la terminazione del processo e vengono eliminate le copie shadow con lo scopo di non permettere il ripristino dei files in modo semplice:



Tramite un processo di text reversing abbiamo ottenuto i seguenti comandi eseguiti:

Input

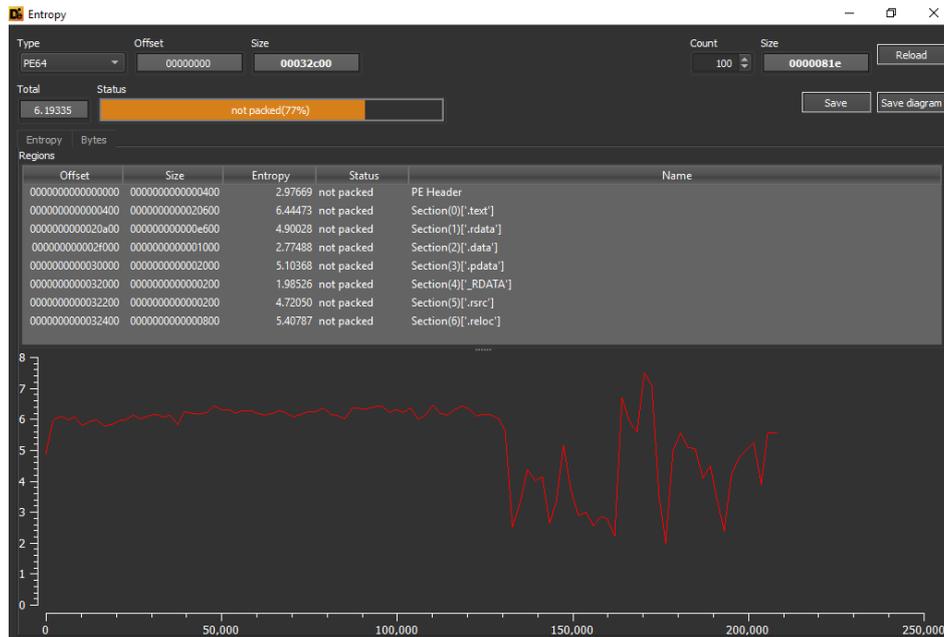
```
lla/ teIuq/ swodahs eteled nimdassv c/ exe.dmc  
eteled ypocwodahs cimw c/ exe.dmc  
seruliafllaerongi ycilopsutatstoob }tluafed{ tes / tidedcb c / exe.dmc  
|on delbaneyrevocer }tluafed{ tes/ tidedcb c/ exe.dmc
```

abc 207 4

Output

```
cmd.exe /c bcdedit /set {default} recoveryenabled no  
cmd.exe / c bcdedit / set {default} bootstatuspolicy ignoreallfailures  
cmd.exe /c wmic shadowcopy delete  
cmd.exe /c vssadmin delete shadows /quiet /all
```

Le sezioni del malware non risultano possedere peculiarità di packing, pertanto i threat actors non hanno provveduto a disporre un'azione di bytes confusing ai fini di rendere più difficoltosa un'eventuale analisi statica dell'artefatto. Tuttavia, come vedremo più avanti, alcuni attributi specifici di comandi eseguiti sono in stato di "text reversed" o in forma codificata.



Analisi dinamica e second malware assessment

All'interno della funzione **sub_140005530** viene creato un nuovo processo avente come parametri i comandi CMD di *booting modification* (in forma text reversed):

```

sub_140005530 proc near
    bInheritHandles= dword ptr -140h
    dwCreationFlags= dword ptr -138h
    lpEnvironment= qword ptr -130h
    lpCurrentDirectory= qword ptr -128h
    lpStartupInfo= qword ptr -120h
    lpProcessInformation= qword ptr -118h
    var_110= qword ptr -110h
    var_100= qword ptr -100h
    var_F8= qword ptr -0F8h
    StartupInfo= _STARTUPINFOA ptr -0F0h
    ProcessInformation= _PROCESS_INFORMATION ptr -80h
    CommandLine= byte ptr -60h
    var_10= qword ptr -10h
    arg_0= qword ptr 10h

    mov     [rsp-8+arg_0], rbx
    push   rbp
    lea    rbp, [rsp-60h]
    sub    rsp, 160h
    mov    rax, cs: _security_cookie
    xor    rax, rsp
    mov    [rbp+60h+var_10], rax
    xor    ebx, ebx
    mov    [rsp+160h+var_F8], 0Fh
    lea    rdx, a11aTeiuqSwodah ; "lla/ teIuq/ swodahs  eteled nimdassv "...
    mov    [rsp+160h+var_110], rbx
  
```

100.00% (-95, 78) | (744, 415) | 0000495B | 000000014000555B: sub_140005530+2B (Synchronized with Hex View-1)

```

mov     rax, cs:__security_cookie
xor     rax, rsp
mov     [rbp+60h+var_10], rax
xor     ebx, ebx
mov     [rsp+160h+var_F8], 0Fh
lea     rdx, a1laTeiuqSwodah ; "lla/ teIuq/ swodahs eteled nimdassv ..."
mov     [rsp+160h+var_110], rbx
lea     rcx, [rsp+160h+var_110] ; void *
mov     [rsp+160h+var_100], rbx
lea     r8d, [rbx+31h] ; Size
call    sub_140006990
cmp     [rsp+160h+var_F8], 10h
lea     rcx, [rsp+160h+var_110]
mov     rdx, [rsp+160h+var_100]
cmovnb rcx, [rsp+160h+var_110]
add     rdx, rcx
lea     rcx, [rsp+160h+var_110]
cmp     [rsp+160h+var_F8], 10h
cmovnb rcx, [rsp+160h+var_110]
call    __std_reverse_trivially_swappable_1
cmp     [rsp+160h+var_F8], 10h
lea     rcx, [rsp+160h+var_110]
lea     rdx, [rbp+60h+CommandLine]
cmovnb rcx, [rsp+160h+var_110]
sub     rdx, rcx

```

100.00% | (-95, 393) | (732, 400) | 0000495B | 000000014000555B: sub_140005530+2B | (Synchronized with Hex View-1)

Evidenziamo i dettagli di impostazione dei threads e degli attributi di esecuzione del processo stesso:

```

loc_1400055C1:
movzx  eax, byte ptr [rcx]
mov     [rdx+rcx], al
lea     rcx, [rcx+1]
test   al, al
jnz    short loc_1400055C1

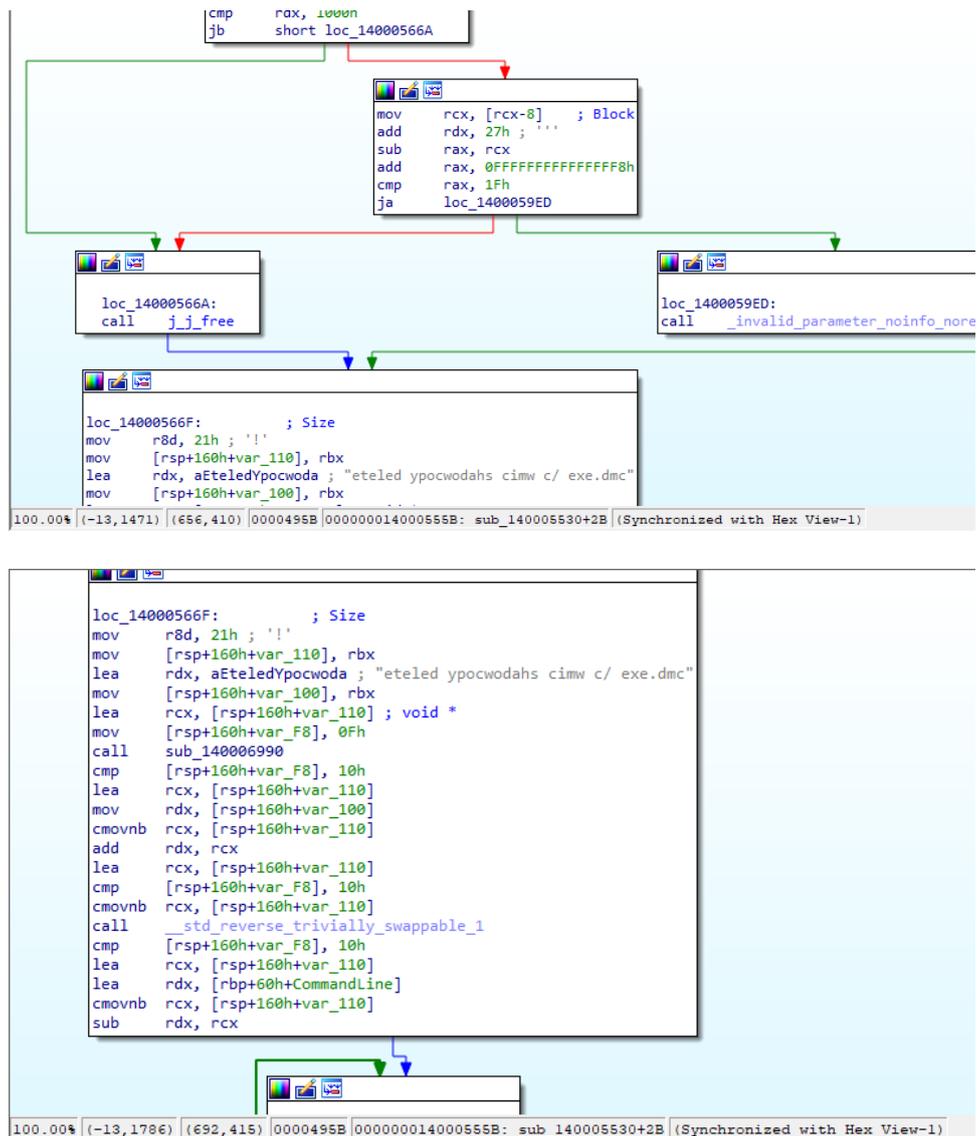
```

```

xorps  xmm0, xmm0
mov     [rsp+160h+StartupInfo.cb], 68h ; 'h'
lea     rax, [rbp+60h+ProcessInformation]
xor     r9d, r9d ; lpThreadAttributes
mov     [rsp+160h+lpProcessInformation], rax ; lpProcessInformation
lea     rdx, [rbp+60h+CommandLine] ; lpCommandLine
lea     rax, [rsp+160h+StartupInfo]
xor     r8d, r8d ; lpProcessAttributes
mov     [rsp+160h+lpStartupInfo], rax ; lpStartupInfo
xor     ecx, ecx ; lpApplicationName
mov     [rsp+160h+lpCurrentDirectory], rbx ; lpCurrentDirectory
mov     [rsp+160h+lpEnvironment], rbx ; lpEnvironment
movups  xmmword ptr [rbp+60h+StartupInfo.dwFillAttribute], xmm0
mov     [rsp+160h+dwCreationFlags], 8000001h ; dwCreationFlags
mov     [rsp+160h+bInheritHandles], ebx ; bInheritHandles
movups  xmmword ptr [rsp+160h+StartupInfo.lpReserved], xmm0
mov     [rbp+60h+StartupInfo.wShowWindow], bx
movups  xmmword ptr [rbp+60h+StartupInfo.lpTitle], xmm0

```

100.00% | (-95, 843) | (788, 413) | 0000495B | 000000014000555B: sub_140005530+2B | (Synchronized with Hex View-1)



Infine, viene richiamata la funzione *CreateProcessA* per la creazione del processo in questione per l'esecuzione dei comandi su citati:

```

movups xmmword ptr [rbp+60h+StartupInfo.lpTitle], xmm0
movups xmmword ptr [rbp+60h+StartupInfo.dwXSize], xmm0
movups xmmword ptr [rbp+60h+StartupInfo.lpReserved2], xmm0
movups xmmword ptr [rbp+60h+StartupInfo.hStdOutput], xmm0
call cs:CreateProcessA
mov rdx, [rsp+160h+var_F8]
cmp rdx, 10h
jnb short loc_14000578E

```

```

mov rcx, [rsp+160h+var_110]
inc rdx
mov rax, rcx
cmp rdx, 1000h
jnb short loc_140005789

```

```

mov rcx, [rcx-8] ; Block
add rdx, 27h ; ''
sub rax, rcx
add rax, 0FFFFFFFFFFFFFFF8h
cmp rax, 1Fh
ja loc_1400059F3

```

```

loc_14000578E: ; Size
mov r8d, 46h ; 'F'
mov [rsp+160h+var_110], rbx
lea rdx, aSeruliafllaero ; "seruliafllaerongi ycilopsutatstooB }tlu"...
mov [rsp+160h+var_100], rbx
lea rcx, [rsp+160h+var_110] ; void *
mov [rsp+160h+var_F8], 0Fh
call sub_140006990
cmp [rsp+160h+var_F8], 10h
lea rcx, [rsp+160h+var_110]
mov rdx, [rsp+160h+var_100]
cmovnb rcx, [rsp+160h+var_110]
add rdx, rcx
lea rcx, [rsp+160h+var_110]
cmp [rsp+160h+var_F8], 10h
cmovnb [rsp+160h+var_110]
call __std_reverse_trivially_swappable_1
cmp [rsp+160h+var_F8], 10h
lea rcx, [rsp+160h+var_110]
lea rdx, [rbp+60h+CommandLine]
cmovnb rcx, [rsp+160h+var_110]
sub rdx, rcx
nop

```

100.00% (-13,2617) (722,413) 0000495B 000000014000555B: sub_140005530+2B (Synchronized with Hex View-1)

100.00% (-5,3182) (706,376) 0000495B 000000014000555B: sub_140005530+2B (Synchronized with Hex View-1)

```

xorps xmm0, xmm0
mov [rsp+160h+StartupInfo.cb], 68h ; 'h'
lea rax, [rbp+60h+ProcessInformation]
xor r9d, r9d ; lpThreadAttributes
mov [rsp+160h+lpProcessInformation], rax ; lpProcessInformation
lea rdx, [rbp+60h+CommandLine] ; lpCommandLine
lea rax, [rsp+160h+StartupInfo]
xor r8d, r8d ; lpProcessAttributes
mov [rsp+160h+lpStartupInfo], rax ; lpStartupInfo
xor ecx, ecx ; lpApplicationName
mov [rsp+160h+lpCurrentDirectory], rbx ; lpCurrentDirectory
mov [rsp+160h+lpEnvironment], rbx ; lpEnvironment
movups xmmword ptr [rbp+60h+StartupInfo.dwFillAttribute], xmm0
mov [rsp+160h+dwCreationFlags], 8000001h ; dwCreationFlags
mov [rsp+160h+bInheritHandles], ebx ; bInheritHandles
movups xmmword ptr [rsp+160h+StartupInfo.lpReserved], xmm0
mov [rbp+60h+StartupInfo.wShowWindow], bx
movups xmmword ptr [rbp+60h+StartupInfo.lpTitle], xmm0
movups xmmword ptr [rbp+60h+StartupInfo.dwXSize], xmm0
movups xmmword ptr [rbp+60h+StartupInfo.lpReserved2], xmm0
movups xmmword ptr [rbp+60h+StartupInfo.hStdOutput], xmm0
call cs:CreateProcessA
mov rdx, [rsp+160h+var_F8]
cmp rdx, 10h
jb short loc_1400058AE

```

100.00% (-6,3738) (689,411) 0000495B 000000014000555B: sub_140005530+2B (Synchronized with Hex View-1)

```

loc_1400058AE: ; Size
mov r8d, 34h ; '4'
mov [rsp+160h+var_110], rbx
lea rdx, a0nDelbaneyrevo ; "on delbaneyrevocer }tluafed{ tes/ tided"...
mov [rsp+160h+var_100], rbx
lea rcx, [rsp+160h+var_110] ; void *
mov [rsp+160h+var_F8], 0Fh
call sub_140006990
cmp [rsp+160h+var_F8], 10h
lea rcx, [rsp+160h+var_110]
mov rdx, [rsp+160h+var_100]
cmovnb rcx, [rsp+160h+var_110]
add rdx, rcx
lea rcx, [rsp+160h+var_110]
cmp [rsp+160h+var_F8], 10h
cmovnb rcx, [rsp+160h+var_110]
call _std_reverse_trivially_swappable_1
cmp [rsp+160h+var_F8], 10h
lea rcx, [rsp+160h+var_110]
lea rdx, [rbp+60h+CommandLine]
cmovnb rcx, [rsp+160h+var_110]
sub rdx, rcx
nop

```

100.00% (17,4531) (805,398) 0000495B 000000014000555B: sub_140005530+2B (Synchronized with Hex View-1)

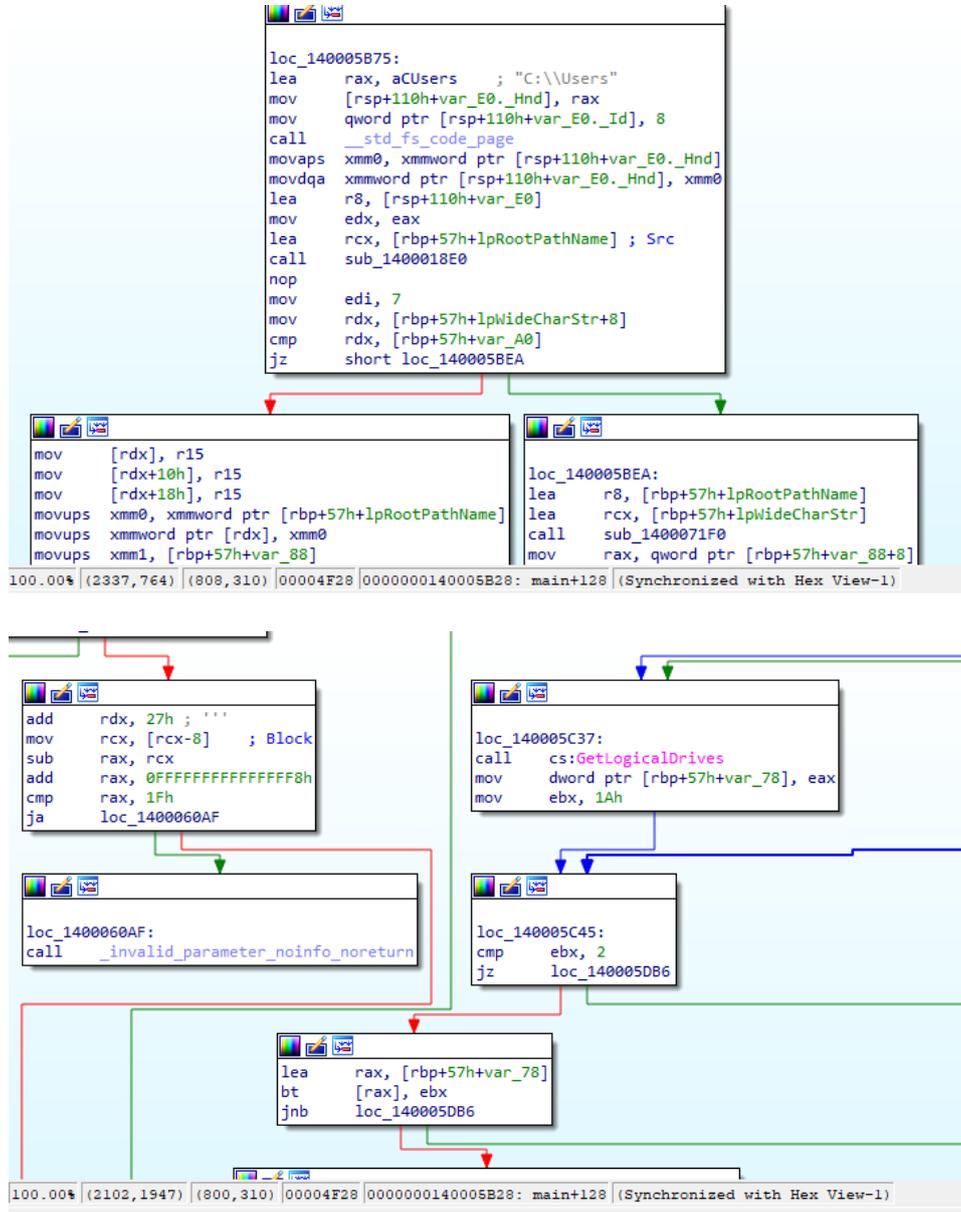
```

xorps xmm0, xmm0
mov [rsp+160h+StartupInfo.cb], 68h ; 'h'
lea rax, [rbp+60h+ProcessInformation]
xor r9d, r9d ; lpThreadAttributes
mov [rsp+160h+lpProcessInformation], rax ; lpProcessInformation
lea rdx, [rbp+60h+CommandLine] ; lpCommandLine
lea rax, [rsp+160h+StartupInfo]
xor r8d, r8d ; lpProcessAttributes
mov [rsp+160h+lpStartupInfo], rax ; lpStartupInfo
xor ecx, ecx ; lpApplicationName
mov [rsp+160h+lpCurrentDirectory], rbx ; lpCurrentDirectory
mov [rsp+160h+lpEnvironment], rbx ; lpEnvironment
movups xmmword ptr [rbp+60h+StartupInfo.dwFillAttribute], xmm0
mov [rsp+160h+dwCreationFlags], 8000001h ; dwCreationFlags
mov [rsp+160h+bInheritHandles], ebx ; bInheritHandles
movups xmmword ptr [rsp+160h+StartupInfo.lpReserved], xmm0
mov [rbp+60h+StartupInfo.wShowWindow], bx
movups xmmword ptr [rbp+60h+StartupInfo.lpTitle], xmm0
movups xmmword ptr [rbp+60h+StartupInfo.dwXSize], xmm0
movups xmmword ptr [rbp+60h+StartupInfo.lpReserved2], xmm0
movups xmmword ptr [rbp+60h+StartupInfo.hStdOutput], xmm0
call cs:CreateProcessA
mov rdx, [rsp+160h+var_F8]
cmp rdx, 10h
jb short loc_1400059CA

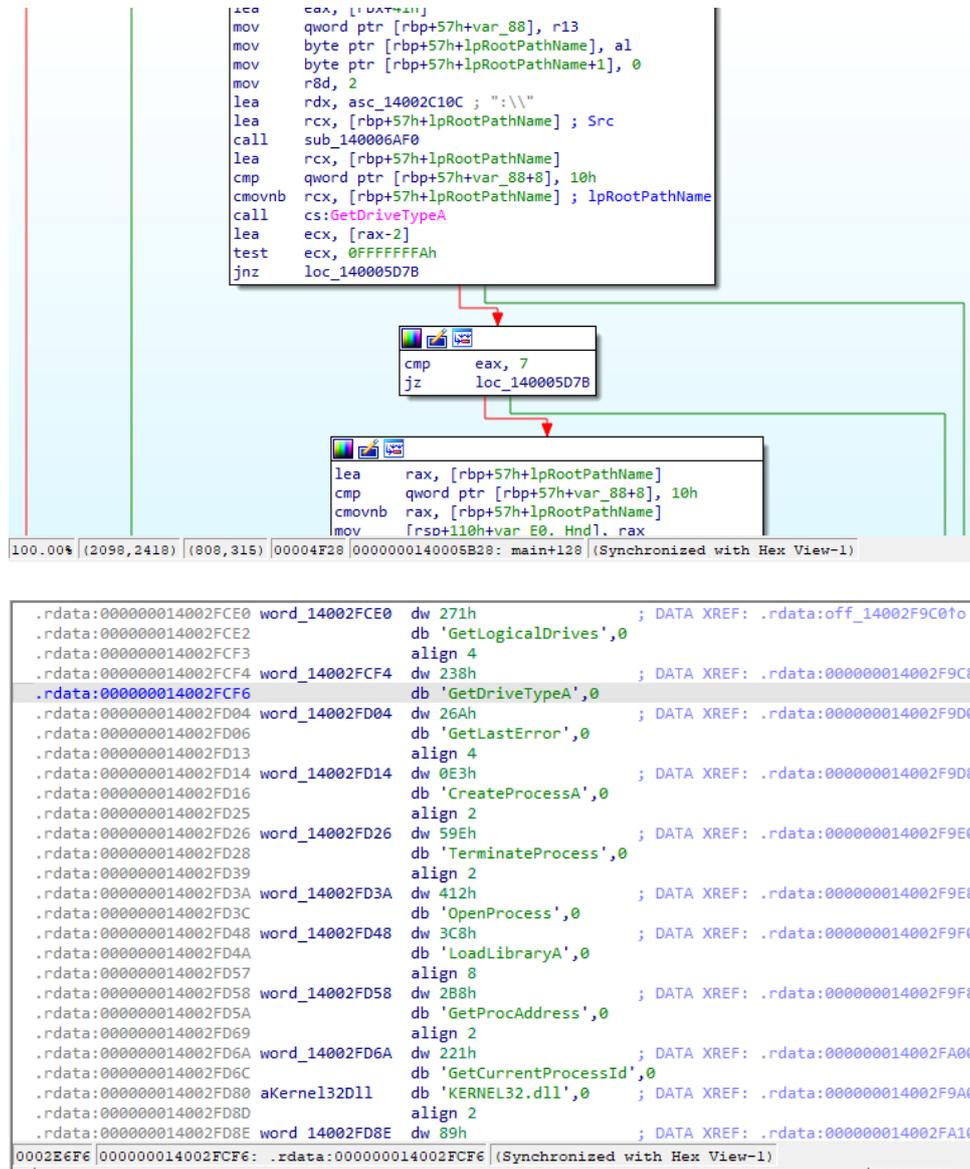
```

100.00% (16,5151) (734,403) 0000495B 000000014000555B: sub_140005530+2B (Synchronized with Hex View-1)

Vengono presi in considerazione i files e dati partendo dalla root folder **C:\\Users**



Successivamente vengono enumerate e classificate le varie tipologie di dischi di sistema:



```

lea     eax, [DAT+11]
mov     qword ptr [rbp+57h+var_88], r13
mov     byte ptr [rbp+57h+lpRootPathName], al
mov     byte ptr [rbp+57h+lpRootPathName+1], 0
mov     r8d, 2
lea     rdx, asc_14002C10C ; "\\\"
lea     rcx, [rbp+57h+lpRootPathName] ; Src
call    sub_140006AF0
lea     rcx, [rbp+57h+lpRootPathName]
cmp     qword ptr [rbp+57h+var_88+8], 10h
cmovnb rcx, [rbp+57h+lpRootPathName] ; lpRootPathName
call    cs:GetDriveTypeA
lea     ecx, [rax-2]
test    ecx, 0FFFFFFFAh
jnz     loc_140005D7B

cmp     eax, 7
jz      loc_140005D7B

lea     rax, [rbp+57h+lpRootPathName]
cmp     qword ptr [rbp+57h+var_88+8], 10h
cmovnb rax, [rbp+57h+lpRootPathName]
mov     [rsi+110h+var_E0.Hnd1.rax]

100.00% (2098,2418) (808,315) 00004F28 | 0000000140006B28: main+128 | (Synchronized with Hex View-1)

```

.rdata:000000014002FCE0	word_14002FCE0	dw 271h	; DATA XREF: .rdata:off_14002F9C0to
.rdata:000000014002FCE2	db 'GetLogicalDrives',0		
.rdata:000000014002FCF3	align 4		
.rdata:000000014002FCF4	word_14002FCF4	dw 238h	; DATA XREF: .rdata:000000014002F9C8
.rdata:000000014002FCF6	db 'GetDriveTypeA',0		
.rdata:000000014002FD04	word_14002FD04	dw 26Ah	; DATA XREF: .rdata:000000014002F9D0
.rdata:000000014002FD06	db 'GetLastError',0		
.rdata:000000014002FD13	align 4		
.rdata:000000014002FD14	word_14002FD14	dw 0E3h	; DATA XREF: .rdata:000000014002F9D8
.rdata:000000014002FD16	db 'CreateProcessA',0		
.rdata:000000014002FD25	align 2		
.rdata:000000014002FD26	word_14002FD26	dw 59Eh	; DATA XREF: .rdata:000000014002F9E0
.rdata:000000014002FD28	db 'TerminateProcess',0		
.rdata:000000014002FD39	align 2		
.rdata:000000014002FD3A	word_14002FD3A	dw 412h	; DATA XREF: .rdata:000000014002F9E8
.rdata:000000014002FD3C	db 'OpenProcess',0		
.rdata:000000014002FD48	word_14002FD48	dw 3C8h	; DATA XREF: .rdata:000000014002F9F0
.rdata:000000014002FD4A	db 'LoadLibraryA',0		
.rdata:000000014002FD57	align 8		
.rdata:000000014002FD58	word_14002FD58	dw 2B8h	; DATA XREF: .rdata:000000014002F9F8
.rdata:000000014002FD5A	db 'GetProcAddress',0		
.rdata:000000014002FD69	align 2		
.rdata:000000014002FD6A	word_14002FD6A	dw 221h	; DATA XREF: .rdata:000000014002FA00
.rdata:000000014002FD6C	db 'GetCurrentProcessId',0		
.rdata:000000014002FD80	aKernel32D11	db 'KERNEL32.dll',0	; DATA XREF: .rdata:000000014002F9A0
.rdata:000000014002FD8D	align 2		
.rdata:000000014002FD8E	word_14002FD8E	dw 89h	; DATA XREF: .rdata:000000014002FA10

0002E6F6 | 000000014002FCF6: .rdata:000000014002FCF6 | (Synchronized with Hex View-1)

Vengono raccolti gli attributi e parametri per procedere con la fase di infection chain, quali il numero di threads, cores CPU, path ed elementi statistici di esecuzione:

```

_140005DC4:      ; Microsoft VisualC v14 64bit runtime
l      unknown_libname_8
      r9d, eax
      rcx, [rbp+57h+lpWideCharStr+8]
      rcx, [rbp+57h+lpWideCharStr]
      rcx, 5
      eax, eax
      edx, edx
      rcx
      eax, 1
/r1    eax, r13d
      r14d, [rax+rax*2]
      r8d, r14d
      edx, r9d
      rcx, aCpuCoresDThrea ; "[+] CPU cores: %d, Threads: %d\n"
l      sub_140001030
      [rbp+57h+var_78], 1388h
      rcx, [rbp+57h+var_78]
l      sub_140007050
      ecx, 8 ; Size
l      ??2@YAPEAX_K@Z ; operator new(unsigned __int64)
      rcx, sub_140005530
      [rax], rcx
      qword ptr [rbp+57h+var_40], rax
      rcx, [rbp+57h+var_50+8]

```

```

lea    rdx, [rbp+57h+Src] ; Src
mov    rcx, rbx ; lpWideCharStr
call   sub_140006090
lea    rdx, [rbp+57h+Src]
cmp    qword ptr [rbp+57h+var_C0+8], 10h
cmovnb rdx, qword ptr [rbp+57h+Src]
lea    rcx, aPathS ; "[+] Path: %s\n"
call   sub_140001030
mov    rdx, qword ptr [rbp+57h+var_C0+8]
cmp    rdx, 10h
jb     short loc_140005B6F

```

100.00% (1860,4714) (803,312) 00004F28 0000000140005B28: main+128 (Synchronized with Hex View-1)

format-string	[+] Stats: %d %d
format-string	[+] Path: %s
format-string	[+] CPU cores: %d, Threads: %d

Abbiamo evidenza della gestione delle sessioni di *Restart Management* con gli attributi di `CurrentProcessID`:

```

movups [rsp+1AF8h+var_58], xmm0
movups [rsp+1AF8h+var_48], xmm0
call   cs:LoadLibraryA
mov    rbx, rax
test   rax, rax
jz     loc_140009088

```

```

lea    rdx, ProcName ; "RmStartSession"
mov    rcx, rax ; hModule
call   cs:GetProcAddress
mov    rdi, rax
test   rax, rax
jz     loc_140009088

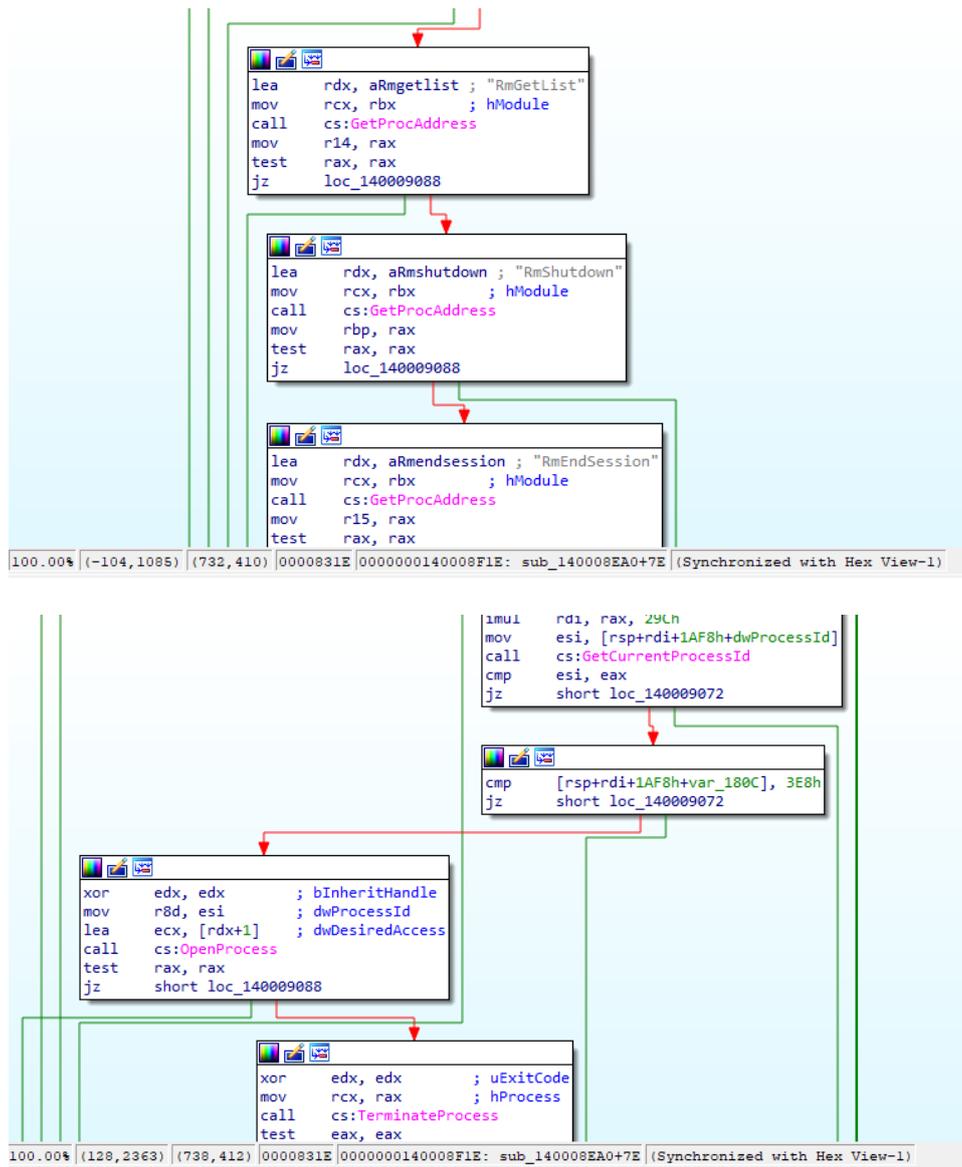
```

```

lea    rdx, aRmregisterreso ; "RmRegisterResources"
mov    rcx, rbx ; hModule
call   cs:GetProcAddress
mov    rsi, rax
test   rax, rax
jz     loc_140009088

```

100.00% (-158,702) (803,310) 0000831E 0000000140008F1E: sub_140008EA0+7E (Synchronized with Hex View-1)



Ai seguenti indirizzi della sezione *.rdata* individuabili da *0000000014002FE48*, sono evidenziabili le funzioni di files looping getting, ad esempio, *FindNextFileW*, *FindFirstFileExW* e *GetFileInformationByHandleEx*. Quest'ultima permette di ottenere dettagli di files specifici all'interno di una fase iterativa:

```

.rdata:000000014002FE2E db 'CreateFileW',0
.rdata:000000014002FE3A word_14002FE3A dw 17Eh ; DATA XREF: .rdata:000000014002FA60
.rdata:000000014002FE3C db 'FindClose',0
.rdata:000000014002FE46 word_14002FE46 dw 184h ; DATA XREF: .rdata:000000014002FA68
.rdata:000000014002FE48 db 'FindFirstFileExW',0
.rdata:000000014002FE59 align 2
.rdata:000000014002FE5A word_14002FE5A dw 195h ; DATA XREF: .rdata:000000014002FA70
.rdata:000000014002FE5C .rdata:000000014002FE5C db 'FindNextFileW',0
.rdata:000000014002FE6A word_14002FE6A dw 24Ch ; DATA XREF: .rdata:000000014002FA78
.rdata:000000014002FE6C db 'GetFileAttributesExW',0
.rdata:000000014002FE81 align 2
.rdata:000000014002FE82 word_14002FE82 dw 520h ; DATA XREF: .rdata:000000014002FA80
.rdata:000000014002FE84 db 'SetEndOfFile',0
.rdata:000000014002FE91 align 2
.rdata:000000014002FE92 word_14002FE92 dw 52Dh ; DATA XREF: .rdata:000000014002FA88
.rdata:000000014002FE94 db 'SetFileAttributesW',0
.rdata:000000014002FEA7 align 8
.rdata:000000014002FEA8 word_14002FEA8 dw 533h ; DATA XREF: .rdata:000000014002FA90
.rdata:000000014002FEAA db 'SetFilePointerEx',0
.rdata:000000014002FEBB align 4
.rdata:000000014002FEBB word_14002FEBB dw 23h ; DATA XREF: .rdata:000000014002FA98
.rdata:000000014002FEBE db 'AreFileApisANSI',0
.rdata:000000014002FECE word_14002FECE dw 3EFh ; DATA XREF: .rdata:000000014002FAA0
.rdata:000000014002FED0 db 'MoveFileExW',0
.rdata:000000014002FEDC word_14002FEDC dw 252h ; DATA XREF: .rdata:000000014002FAA8
.rdata:000000014002FEDE db 'GetFileInformationByHandleEx',0
.rdata:000000014002FEFB align 4
0002E85C|000000014002FE5C: .rdata:000000014002FE5C (Synchronized with Hex View-1)

```

Di seguito ulteriori dettagli presenti all'interno della sezione .rdata inerenti alle funzioni di performance counter querying, ottenimento del timestamp locale per environment execution awareness (individuazione e classificazione dell'ambiente d'esecuzione).

```

.rdata:000000014002FEFE db 'MultiByteToWideChar',0
.rdata:000000014002FF12 word_14002FF12 dw 611h ; DATA XREF: .rdata:000000014002FAB8
.rdata:000000014002FF14 db 'WideCharToMultiByte',0
.rdata:000000014002FF28 word_14002FF28 dw 452h ; DATA XREF: .rdata:000000014002FAC0
.rdata:000000014002FF2A .rdata:000000014002FF2A db 'QueryPerformanceCounter',0
.rdata:000000014002FF42 word_14002FF42 dw 453h ; DATA XREF: .rdata:000000014002FAC8
.rdata:000000014002FF44 db 'QueryPerformanceFrequency',0
.rdata:000000014002FF5E word_14002FF5E dw 370h ; DATA XREF: .rdata:000000014002FAD0
.rdata:000000014002FF60 db 'InitializeSRWLock',0
.rdata:000000014002FF72 word_14002FF72 dw 488h ; DATA XREF: .rdata:000000014002FAD8
.rdata:000000014002FF74 db 'ReleaseSRWLockExclusive',0
.rdata:000000014002FF8C word_14002FF8C dw 0 ; DATA XREF: .rdata:000000014002FAE0
.rdata:000000014002FF8E db 'AcquireSRWLockExclusive',0
.rdata:000000014002FFA6 word_14002FFA6 dw 138h ; DATA XREF: .rdata:000000014002FAE8
.rdata:000000014002FFA8 db 'EnterCriticalSection',0
.rdata:000000014002FFBD align 2
.rdata:000000014002FFBE word_14002FFBE dw 3C4h ; DATA XREF: .rdata:000000014002FAF0
.rdata:000000014002FFC0 db 'LeaveCriticalSection',0
.rdata:000000014002FFD5 align 2
.rdata:000000014002FFD6 word_14002FFD6 dw 36Ch ; DATA XREF: .rdata:000000014002FAF8
.rdata:000000014002FFD8 db 'InitializeCriticalSectionEx',0
.rdata:000000014002FFF4 word_14002FFF4 dw 589h ; DATA XREF: .rdata:000000014002FB00
.rdata:000000014002FFF6 db 'TryEnterCriticalSection',0
.rdata:000000014003000E word_14003000E dw 114h ; DATA XREF: .rdata:000000014002FB08
.rdata:0000000140030010 db 'DeleteCriticalSection',0
.rdata:0000000140030026 word_140030026 dw 2F3h ; DATA XREF: .rdata:000000014002FB10
.rdata:0000000140030028 db 'GetSystemTimeAsFileTime',0
0002E92A|000000014002FF2A: .rdata:000000014002FF2A (Synchronized with Hex View-1)

```

```

.rdata:000000014002FFD5 align 2
.rdata:000000014002FFD6 word_14002FFD6 dw 36Ch ; DATA XREF: .rdata:000000014002FAF8
.rdata:000000014002FFD8 db 'InitializeCriticalSection',0
.rdata:000000014002FFFA word_14002FFFA dw 589h ; DATA XREF: .rdata:000000014002FB00
.rdata:000000014002FFF6 db 'TryEnterCriticalSection',0
.rdata:000000014003000E word_14003000E dw 114h ; DATA XREF: .rdata:000000014002FB08
.rdata:0000000140030010 db 'DeleteCriticalSection',0
.rdata:0000000140030026 word_140030026 dw 2F3h ; DATA XREF: .rdata:000000014002FB10
.rdata:0000000140030028 db 'GetSystemTimeAsFileTime',0
.rdata:0000000140030040 word_140030040 dw 281h ; DATA XREF: .rdata:000000014002FB18
.rdata:0000000140030042 db 'GetModuleHandleW',0
.rdata:0000000140030053 align 4
.rdata:0000000140030054 word_140030054 dw 4D5h ; DATA XREF: .rdata:000000014002FB20
.rdata:0000000140030056 db 'RtlCaptureContext',0
.rdata:0000000140030068 word_140030068 dw 4DCh ; DATA XREF: .rdata:000000014002FB28
.rdata:000000014003006A db 'RtlLookupFunctionEntry',0
.rdata:0000000140030081 align 2
.rdata:0000000140030082 word_140030082 dw 4E3h ; DATA XREF: .rdata:000000014002FB30
.rdata:0000000140030084 db 'RtlVirtualUnwind',0
.rdata:0000000140030095 align 2
.rdata:0000000140030096 word_140030096 dw 5C0h ; DATA XREF: .rdata:000000014002FB38
.rdata:0000000140030098 db 'UnhandledExceptionFilter',0
.rdata:00000001400300B1 align 2
.rdata:00000001400300B2 word_1400300B2 dw 57Fh ; DATA XREF: .rdata:000000014002FB40
.rdata:00000001400300B4 db 'SetUnhandledExceptionFilter',0
.rdata:00000001400300D0 word_1400300D0 dw 220h ; DATA XREF: .rdata:000000014002FB48
.rdata:00000001400300D2 db 'GetCurrentProcess',0
0002E9F6 000000014002FFF6: .rdata:000000014002FFF6 (Synchronized with Hex View-1)

```

Qui ulteriori riferimenti alle funzioni *IsDebuggerPresent* ed *EncodePointer*:

```

.rdata:0000000140030098 db 'UnhandledExceptionFilter',0
.rdata:00000001400300B1 align 2
.rdata:00000001400300B2 word_1400300B2 dw 57Fh ; DATA XREF: .rdata:000000014002
.rdata:00000001400300B4 db 'SetUnhandledExceptionFilter',0
.rdata:00000001400300D0 word_1400300D0 dw 220h ; DATA XREF: .rdata:000000014002
.rdata:00000001400300D2 db 'GetCurrentProcess',0
.rdata:00000001400300E4 word_1400300E4 dw 38Ch ; DATA XREF: .rdata:000000014002
.rdata:00000001400300E6 db 'IsProcessorFeaturePresent',0
.rdata:0000000140030100 word_140030100 dw 36Fh ; DATA XREF: .rdata:000000014002
.rdata:0000000140030102 db 'InitializeSListHead',0
.rdata:0000000140030116 word_140030116 dw 385h ; DATA XREF: .rdata:000000014002
.rdata:0000000140030118 db 'IsDebuggerPresent',0
.rdata:000000014003012A word_14003012A dw 2DAh ; DATA XREF: .rdata:000000014002
.rdata:000000014003012C db 'GetStartupInfoW',0
.rdata:000000014003013C word_14003013C dw 4E2h ; DATA XREF: .rdata:000000014002
.rdata:000000014003013E db 'RtlUnwindEx',0
.rdata:000000014003014A word_14003014A dw 4DEh ; DATA XREF: .rdata:000000014002
.rdata:000000014003014C db 'RtlPcToFileHeader',0
.rdata:000000014003015E word_14003015E dw 468h ; DATA XREF: .rdata:000000014002
.rdata:0000000140030160 db 'RaiseException',0
.rdata:000000014003016F align 10h
.rdata:0000000140030170 word_140030170 dw 541h ; DATA XREF: .rdata:000000014002
.rdata:0000000140030172 db 'SetLastError',0
.rdata:000000014003017F align 20h
.rdata:0000000140030180 word_140030180 dw 134h ; DATA XREF: .rdata:000000014002
.rdata:0000000140030182 db 'EncodePointer',0
.rdata:0000000140030190 word_140030190 dw 368h ; DATA XREF: .rdata:000000014002
0002EAD0 00000001400300D0: .rdata:word_1400300D0 (Synchronized with Hex View-1)

```

L'eseguibile è stato compilato il **21 Ottobre 2023**:

property	value
md5	E26BBA0304F14EF968EB60376791D32C
sha1	24F6785CA2E82D1D1D61F4CB01D5E753F80445CF
sha256	40417E937CD244B2F928150CAE6FA0EFF551FDB401EA072F6ECDDA67A747E17
first-bytes-hex	4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00 B8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
first-bytes-text	M Z @
file-size	207872 (bytes)
entropy	6.193
imphash	7339438F1FA3FBACA1E35B75D7395E40
signature	n/a
entry-point	48 83 EC 28 E8 9F 06 00 00 48 83 C4 28 E9 72 FE FF FF CC CC 48 83 EC 28 4D 8B 41 38 48 8B CA 49 8B
file-version	n/a
description	n/a
file-type	executable
cpu	64-bit
subsystem	console
compiler-stamp	0x65346BC9 (Sat Oct 21 17:24:41 2023)
debugger-stamp	0x65346BC9 (Sat Oct 21 17:24:41 2023)
resources-stamp	0x00000000 (empty)
import-stamp	0x00000000 (empty)
exports-stamp	n/a
version-stamp	n/a
certificate-stamp	n/a

Gli indicatori di maggior interesse inerenti al sample fanno riferimento perlopiù a caratteristiche di files management, environment ed hardware information discovery, services management ed execution ed external function calling:

indicator (37)	detail	level
The file references string(s)	type: blacklist, count: 36	1
The count of libraries is suspicious	count: 1	1
The file imports symbol(s)	type: blacklist, count: 21	1
The file contains a blacklist section	section: _RDATA	1
The time-stamp of the compiler is suspicious	year: 2023	2
The time-stamp of a directory is suspicious	directory: debug, stamp: Sat Oct 21 17:24:41 2023	2
The file checksum is invalid	checksum: 0x00000000	3
The file references a group of API	type: synchronization, count: 38	3
The file references a group of API	type: execution, count: 68	3
The file references a group of API	type: file, count: 38	3
The file references a group of API	type: reckoning, count: 14	3
The file references a group of API	type: services, count: 13	3
The file references a group of API	type: storage, count: 4	3
The file references a group of API	type: diagnostic, count: 10	3
The file references a group of API	type: dynamic-library, count: 16	3
The file references a group of API	type: memory, count: 16	3
The file references a group of API	type: exception, count: 8	3
The file references a group of API	type: console, count: 12	3
The file references a group of hint	type: file, count: 8	3
The file references a group of hint	type: format-string, count: 3	3
The file references a group of hint	type: utility, count: 1	3
The file references a group of hint	type: rtti, count: 22	3
The file references a group of hint	type: function, count: 1	3

Qui si elencano ulteriori attributi in merito al Portable Executable, inclusa la file signature:

property	value	detail
compiler-stamp	0x65346BC9	Sat Oct 21 17:24:41 2023
size-of-optional-header	0x00F0	240 bytes
signature	0x00004550	PE00
machine	0x8664	Amd64
sections	0x0007	7
pointer-symbol-table	0x00000000	0x00000000
number-of-symbols	0x00000000	0x00000000
processor-32bit	0x00000000	false
system-image	0x00000000	false
executable	0x00000002	true
dynamic-link-library	0x00000000	false
debug-stripped	0x00000000	false
line-stripped-from-file	0x00000000	false
local-symbols-stripped-from-file	0x00000000	false
relocation-stripped	0x00000000	false
large-address-aware	0x00000020	true
uniprocessor	0x00000000	false
bytes-of-machine-words-reversed-Low	0x00000000	false
bytes-of-machine-words-reversed-Hi	0x00000000	false
media-run-from-swap	0x00000000	false
network-run-from-swap	0x00000000	false

Nelle sezioni dell'artefatto si evincono i valori legati ai coefficienti d'entropia e l'entrypoint (ovvero l'indirizzo iniziale d'esecuzione) della sezione `.text` (istruzioni CPU) all'indirizzo `0x0000AB10`:

property	value	value	value
name	.text	.rdata	.data
md5	47086F913C767FB79FF63FEA...	FC9155991D99E81FAA884AE...	F4D28948DD21F61F2911F50...
entropy	6.445	4.900	2.775
file-ratio (99.51%)	63.79 %	28.33 %	1.97 %
raw-address	0x00000400	0x00020A00	0x0002F000
raw-size (206848 bytes)	0x00020600 (132608 bytes)	0x0000E600 (58880 bytes)	0x00001000 (4096 bytes)
virtual-address	0x0000000040001000	0x0000000040022000	0x0000000040031000
virtual-size (211262 bytes)	0x0002040C (132108 bytes)	0x0000E44E (58446 bytes)	0x000027D0 (10192 bytes)
entry-point	0x0000AB10	-	-
characteristics	0x60000020	0x40000040	0xC0000040
writable	-	-	x
executable	x	-	-
shareable	-	-	-
discardable	-	-	-
initialized-data	-	x	x
uninitialized-data	-	-	-
unreadable	-	-	-
self-modifying	-	-	-
virtualized	-	-	-
file	n/a	n/a	n/a

value	value	value	value
.pdata	.RDATA	.rsrc	.reloc
E969B76C781BFBAFF75A595...	91DDDA35C6D0A6CCD1D39...	2D5EB1E7989B77F5C38C725...	CC5B904DECA074980130772...
5.104	1.982	4.718	5.407
3.94 %	0.25 %	0.25 %	0.99 %
0x00030000	0x00032000	0x00032200	0x00032400
0x00002000 (8192 bytes)	0x00002000 (512 bytes)	0x00002000 (512 bytes)	0x00000800 (2048 bytes)
0x0000000040034000	0x0000000040036000	0x0000000040037000	0x0000000040038000
0x00001E60 (7776 bytes)	0x000000F4 (244 bytes)	0x000001E0 (480 bytes)	0x000007E0 (2016 bytes)
-	-	-	-
0x40000040	0x40000040	0x40000040	0x42000040
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	x
x	x	x	x
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
n/a	n/a	n/a	n/a

Sono presenti diverse funzioni classificabili come meritevoli di attenzione: *CreateProcessA*, *OpenProcess*, *SwitchToThread*, *GetCurrentThreadId*, *GetNativeSystemInfo*, *FindFirstFileExW*, *FindNextFileW*, *MoveFileExW* e *SetFileAttributesW*.

functions (99)	blacklist (21)	type (1)	ordinal (0)	library (1)
CreateProcessA	x	implicit	-	kernel32.dll
TerminateProcess	x	implicit	-	kernel32.dll
OpenProcess	x	implicit	-	kernel32.dll
GetCurrentProcessId	x	implicit	-	kernel32.dll
SwitchToThread	x	implicit	-	kernel32.dll
GetCurrentThreadId	x	implicit	-	kernel32.dll
GetNativeSystemInfo	x	implicit	-	kernel32.dll
FindFirstFileExW	x	implicit	-	kernel32.dll
FindNextFileW	x	implicit	-	kernel32.dll
SetFileAttributesW	x	implicit	-	kernel32.dll
MoveFileExW	x	implicit	-	kernel32.dll
GetFileInformationByHandleEx	x	implicit	-	kernel32.dll
QueryPerformanceFrequency	x	implicit	-	kernel32.dll
RtlLookupFunctionEntry	x	implicit	-	kernel32.dll
RtlPcToFileHeader	x	implicit	-	kernel32.dll
RaiseException	x	implicit	-	kernel32.dll
FreeLibraryAndExitThread	x	implicit	-	kernel32.dll
GetModuleHandleExW	x	implicit	-	kernel32.dll
WriteFile	x	implicit	-	kernel32.dll
GetEnvironmentStringsW	x	implicit	-	kernel32.dll
SetEnvironmentVariableW	x	implicit	-	kernel32.dll

Qui alcune stringhe di information ed attributes gathering, nonché l'estensione appesa ai files .BiBi.

format-string	[+] Stats: %d %d
format-string	[+] Path: %s
format-string	[+] CPU cores: %d, Threads: %d
file	dj.H
file	Rstrtmgr.dll
file	KERNEL32.dll
file	kernel32.dll
file	mscoree.dll
file	.exe
file	.dll
file	.sys
dos-message	!This program cannot be run in DOS mode,
-	oBYwo
-	oRich

	zu-za
	\r\n
	CONOUTS
	.BiBi
	0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz
	\r\n
	\r\n
	\r\n
	\r\n5

Il debugger timestamp risulta essere datato anch'esso il **21 Ottobre 2023**:

property	value
md5	7AD0B3E52BDC0524CC523484FD471772
sha1	107DE9369E5B8D11496BA77ED21CBC8AD9908FA0
sha256	302217B570AC70A8BD2D75279D478D731FF02BD211514B77A0ED5FB2C7EF644D
size	968 (bytes)
format	PGO
debugger-stamp	0x65346BC9 (Sat Oct 21 17:24:41 2023)
path	n/a

Si notino le seguenti evidenze legate alla fase di PE assessment e delle sezioni incluse, tra cui anche le *VirtualSizes* (le dimensioni delle sezioni nel momento in cui vengono mappate in memoria):

Stud_PE editing : "40417e937cd244b2f928150cae6fa0eff555..."

File Edit Tools Help

c:\users\ieuser\desktop\new folder\40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6e

Headers Dos Sections Functions Resources Signature F

HEADERS (Coff+Optional)

0000AB10	EntryPoint (rva)
00009F10	EntryPoint (raw)
0000000140000000	ImageBase
00039000	Size of Image
00001000	Sections Alignment
00000200	File Alignment
0007	Number of sections
0022	Characteristics

DATA DIRECTORY

	RVA	Size	Raw
Import Table	0002F994	00000028	0002E394
Export Table	00000000	00000000	00000000
Data Dir :	IMAGE_DIR_ENTRY_RESOURCE		
GoHex ++	00037000	000001E0	00032200

Basic HEADERS tree view in hexeditor SAVE to file

Visit Stud_PE Forum <- News Here Test' it Rva<=>Raw File Compare OK

Stud_PE editing : "40417e937cd244b2f928150cae6fa0eff555..."

File Edit Tools Help

c:\users\ieuser\desktop\new folder\40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6e

Headers Dos Sections Functions Resources Signature F

No	Name	VirtualSize	VirtualOffset	RawSize	RawOffset	Characteri...
01	.text	0002040C	00001000	00020600	00000400	60000020
02	.rdata	0000E44E	00022000	0000E600	00020A00	40000040
03	.data	000027D0	00031000	00001000	0002F000	C0000040
04	.pdata	00001E60	00034000	00002000	00030000	40000040
05	._RDATA	000000F4	00036000	00000200	00032000	40000040
06	.rsrc	000001E0	00037000	00000200	00032200	40000040
07	.reloc	000007E0	00038000	00000800	00032400	42000040

Visit Stud_PE Forum <- News Here Test' it Rva<=>Raw File Compare OK

Si evidenziano riferimenti relativi all'importazione di diverse funzioni principali di drives enumeration, performance counter information gathering e puntamento di files:

Stud_PE editing : "40417e937cd244b2f928150cae6fa0eff555..."

File Edit Tools Help

c:\users\ieuser\desktop\new folder\40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6e

Headers Dos Sections **fx** Functions Resources Signature F

Imported Functions Exported Functions

- f() AreFileApisANSI ord:35 rva2iat: 000220D8
- f() MoveFileExW ord:1007 rva2iat: 000220E0
- f() GetFileInformationByHandleEx ord:594 rva2iat: 000220E8
- f() MultiByteToWideChar ord:1014 rva2iat: 000220F0
- f() WideCharToMultiByte ord:1553 rva2iat: 000220F8
- f() **QueryPerformanceCounter** ord:1106 rva2iat: 00022100
- f() QueryPerformanceFrequency ord:1107 rva2iat: 00022108
- f() InitializeSRWLock ord:880 rva2iat: 00022110
- f() ReleaseSRWLockExclusive ord:1208 rva2iat: 00022118
- f() AcquireSRWLockExclusive ord:0 rva2iat: 00022120

Found No Exports !

Show Imp OriginalFirstThunk FirstThunk Show Exp

[Visit Stud PE Forum](#) < News Here Test' it Rva<=>Raw File Compare OK

Stud_PE editing : "40417e937cd244b2f928150cae6fa0eff555..."

File Edit Tools Help

c:\users\ieuser\desktop\new folder\40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6e

Headers Dos Sections **fx** Functions Resources Signature F

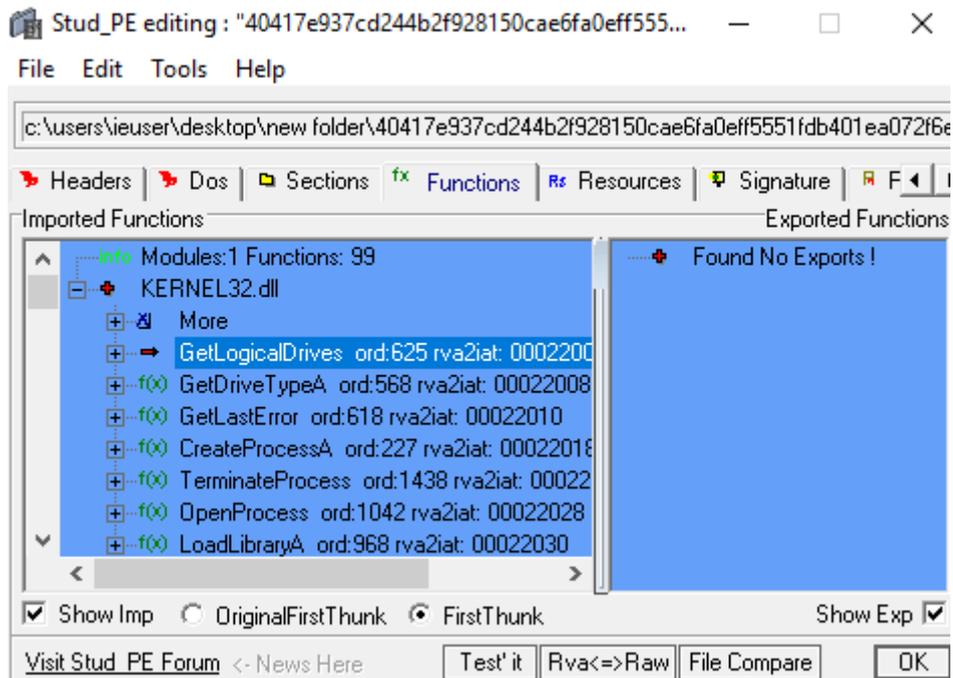
Imported Functions Exported Functions

- f() FormatMessageA ord:431 rva2iat: 00022090
- f() CreateFileW ord:206 rva2iat: 00022098
- f() FindClose ord:382 rva2iat: 000220A0
- f() FindFirstFileExW ord:388 rva2iat: 000220A8
- f() FindNextFileW ord:405 rva2iat: 000220B0
- f() GetFileAttributesExW ord:588 rva2iat: 000220B8
- f() SetEndOfFile ord:1312 rva2iat: 000220C0
- f() SetFileAttributesW ord:1325 rva2iat: 000220C8
- f() **SetFilePointerEx** ord:1331 rva2iat: 000220D0
- f() AreFileApisANSI ord:35 rva2iat: 000220D8

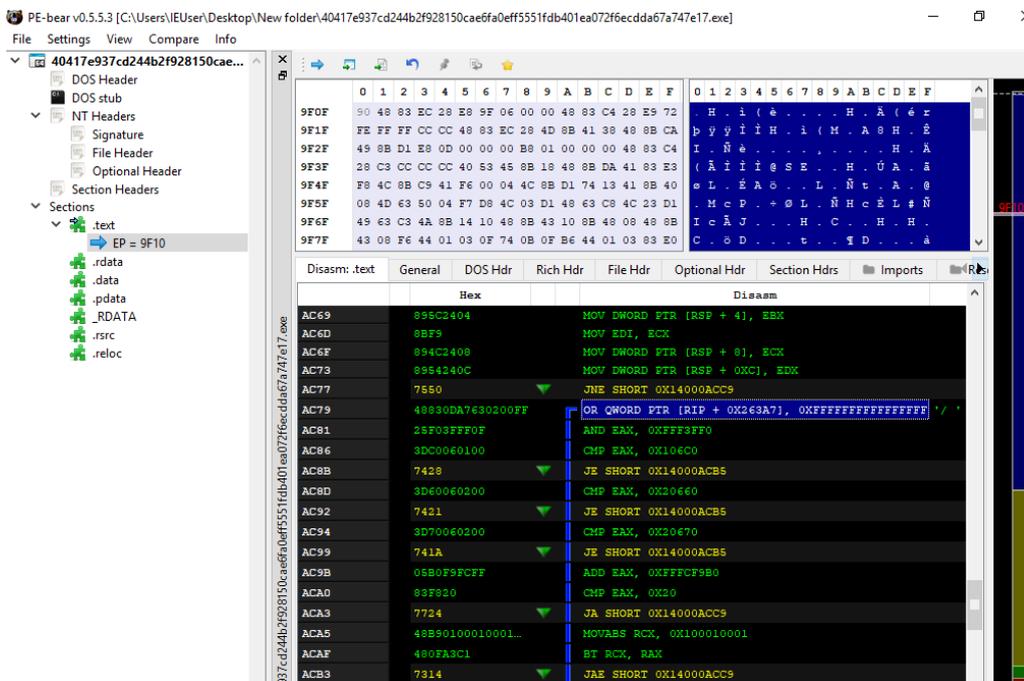
Found No Exports !

Show Imp OriginalFirstThunk FirstThunk Show Exp

[Visit Stud PE Forum](#) < News Here Test' it Rva<=>Raw File Compare OK



All'interno della sezione `.text` è riscontrabile l'utilizzo dell'operatore `OR` in merito all'attributo ***QWORD PTR [RIP + 0x263A7]***. L'operazione di `OR` logica viene effettuata con l'elemento esadecimale ***0xFFFFFFFFFFFFFFFF***, il quale rappresenta un errore di **read access violation**.



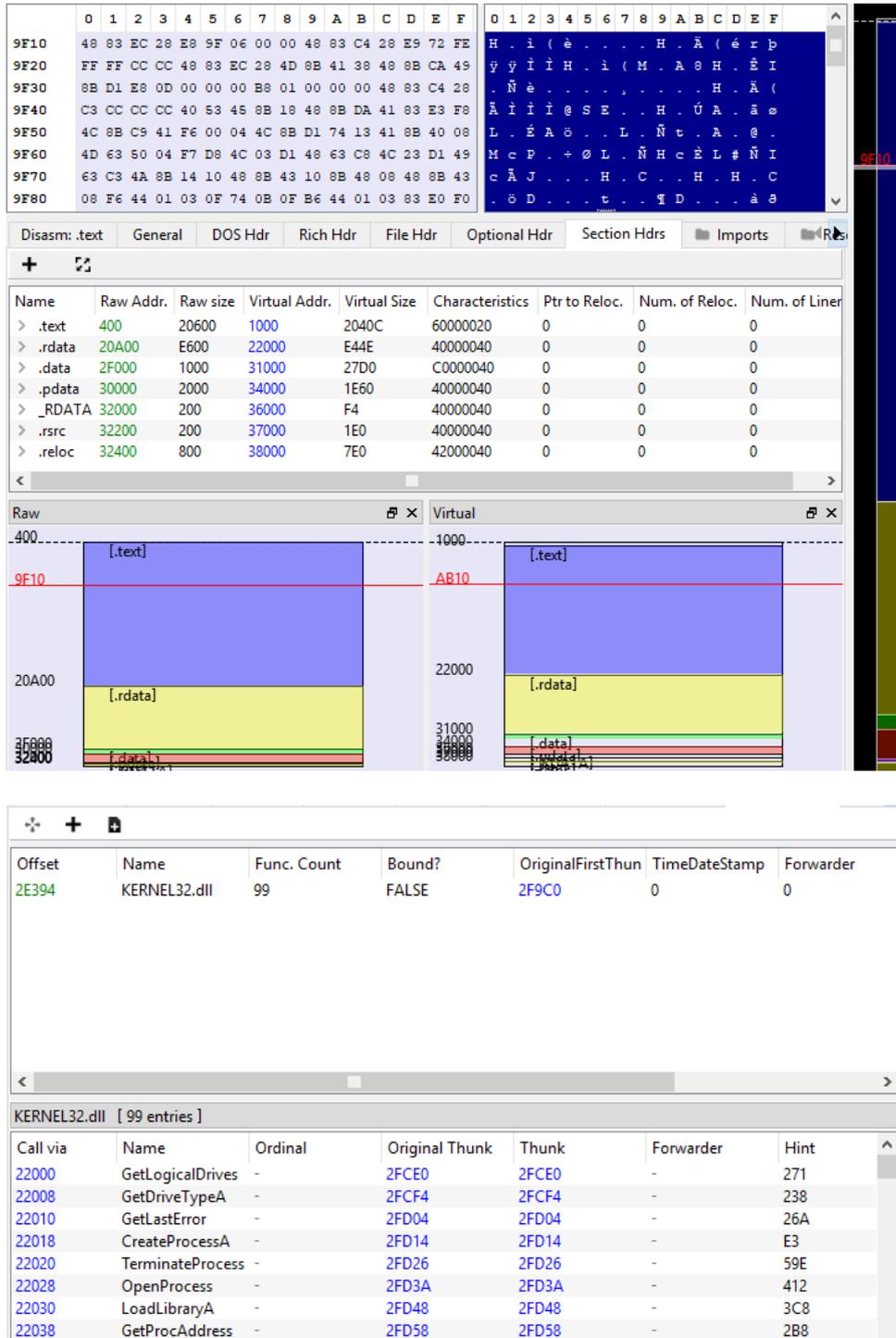
All'interno dell'ultima pagina del PE vi sono 90 bytes:

Offset	Name	Value
0	Magic number	5A4D
2	Bytes on last page of file	90
4	Pages in file	3
6	Relocations	0
8	Size of header in paragraphs	4
A	Minimum extra paragraphs needed	0
C	Maximum extra paragraphs needed	FFFF
E	Initial (relative) SS value	0
10	Initial SP value	B8
12	Checksum	0
14	Initial IP value	0
16	Initial (relative) CS value	0
18	File address of relocation table	40
1A	Overlay number	0
1C	Reserved words[4]	0, 0, 0, 0
24	OEM identifier (for OEM information)	0
26	OEM information; OEM identifier specific	0
28	Reserved words[10]	0, 0, 0, 0, 0, 0, 0, 0, 0, 0
3C	File address of new exe header	100

L'*Import Address Table* (elemento che contiene gli indirizzi delle librerie DLL esterne importate) possiede una dimensione che si attesta a **320**:

Offset	Name	Value	Value
		8000	TerminalServer aware
160	Size of Stack Reserve	100000	
168	Size of Stack Commit	1000	
170	Size of Heap Reserve	100000	
178	Size of Heap Commit	1000	
180	Loader Flags	0	
184	Number of RVAs and Sizes	10	
	Data Directory	Address	Size
188	Export Directory	0	0
190	Import Directory	2F994	28
198	Resource Directory	37000	1E0
1A0	Exception Directory	34000	1E60
1A8	Security Directory	0	0
1B0	Base Relocation Table	38000	7E0
1B8	Debug Directory	2C340	38
1C0	Architecture Specific Data	0	0
1C8	RVA of GlobalPtr	0	0
1D0	TLS Directory	2C500	28
1D8	Load Configuration Directory	2C380	138
1E0	Bound Import Directory in headers	0	0
1E8	Import Address Table	22000	320
1F0	Delay Load Import Descriptors	0	0
1F8	.NET header	0	0

Qui le dimensioni delle sezioni:



The image displays a debugger interface with three main components:

- Memory Dump:** Shows a hex dump of memory starting at address 9F10. The dump contains ASCII characters, including "H. i (è . . . H. Ä (é r b" and "ÿ ÿ Ì Ì H. i (M. A Ø H. È I".
- Section List:** A table listing sections with their raw and virtual addresses and sizes.

Name	Raw Addr.	Raw size	Virtual Addr.	Virtual Size	Characteristics	Ptr to Reloc.	Num. of Reloc.	Num. of Liner
> .text	400	20600	1000	2040C	60000020	0	0	0
> .rdata	20A00	E600	22000	E44E	40000040	0	0	0
> .data	2F000	1000	31000	27D0	C0000040	0	0	0
> .pdata	30000	2000	34000	1E60	40000040	0	0	0
> _RDATA	32000	200	36000	F4	40000040	0	0	0
> .rsrc	32200	200	37000	1E0	40000040	0	0	0
> .reloc	32400	800	38000	7E0	42000040	0	0	0
- Symbol Table:** Shows the symbol table for KERNEL32.dll.

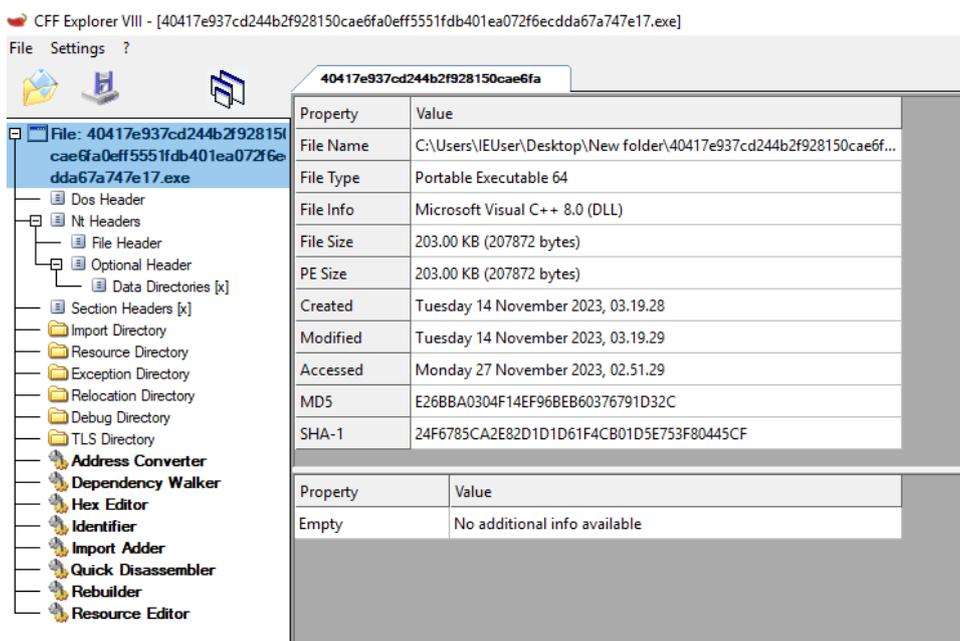
Offset	Name	Func. Count	Bound?	OriginalFirstThun	TimeDateStamp	Forwarder
2E394	KERNEL32.dll	99	FALSE	2F9C0	0	0

Il timestamp di debugging risale al **22 Ottobre 2023**:

Offset	Name	Value	Meaning
2AD40	Characteristics	0	
2AD44	TimeStamp	65346BC9	Sunday, 22.10.2023 00:24:41 UTC
2AD48	MajorVersion	0	
2AD4A	MinorVersion	0	
2AD4C	Type	D	POGO
2AD50	SizeOfData	3C8	
2AD54	AddressOfRaw...	2CFF4	
2AD58	PointerToRawD...	2B9F4	

Offset	Name	Value
--------	------	-------

L'eseguibile è stato compilato in **DllCharacteristics 8160** (relativo all'applicazione di ASLR e high entropy del PE ai fini della protezione da exploits rendendo di fatto randomici gli indirizzi delle funzioni richiamate e porzioni di memoria fondamentali utilizzate dal processo stesso).



CFF Explorer VIII - [40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.exe]

File Settings ?

40417e937cd244b2f928150cae6fa

Property	Value
File Name	C:\Users\IEUser\Desktop\New folder\40417e937cd244b2f928150cae6f...
File Type	Portable Executable 64
File Info	Microsoft Visual C++ 8.0 (DLL)
File Size	203.00 KB (207872 bytes)
PE Size	203.00 KB (207872 bytes)
Created	Tuesday 14 November 2023, 03.19.28
Modified	Tuesday 14 November 2023, 03.19.29
Accessed	Monday 27 November 2023, 02.51.29
MD5	E26BBA0304F14EF968EB60376791D32C
SHA-1	24F6785CA2E82D1D1D61F4CB01D5E753F80445CF

Property	Value
Empty	No additional info available

40417e937cd244b2f928150cae6fa				
Member	Offset	Size	Value	Meaning
SizeOfInitializedData	00000120	Dword	00013A00	
SizeOfUninitializedData	00000124	Dword	00000000	
AddressOfEntryPoint	00000128	Dword	0000AB10	.text
BaseOfCode	0000012C	Dword	00001000	
ImageBase	00000130	Qword	0000000140000000	
SectionAlignment	00000138	Dword	00001000	
FileAlignment	0000013C	Dword	00000200	
MajorOperatingSystemVers...	00000140	Word	0006	
MinorOperatingSystemVers...	00000142	Word	0000	
MajorImageVersion	00000144	Word	0000	
MinorImageVersion	00000146	Word	0000	
MajorSubsystemVersion	00000148	Word	0006	
MinorSubsystemVersion	0000014A	Word	0000	
Win32VersionValue	0000014C	Dword	00000000	
SizeOfImage	00000150	Dword	00039000	
SizeOfHeaders	00000154	Dword	00000400	
Checksum	00000158	Dword	00000000	
Subsystem	0000015C	Word	0003	Windows Console
DllCharacteristics	0000015E	Word	8160	Click here
SizeOfStackReserve	00000160	Qword	000000000100000	
SizeOfStackCommit	00000168	Qword	0000000000001000	
SizeOfHeapReserve	00000170	Qword	000000000100000	
SizeOfHeapCommit	00000178	Qword	0000000000001000	
LoaderFlags	00000180	Dword	00000000	
NumberOfRvaAndSizes	00000184	Dword	00000010	

Diverse funzioni di files management ed importazione di librerie esterne sono contenute all'interno del dump esadecimale (*WriteFile*, *LoadLibraryExW* e *GetFileType*) del Portable Executable:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Ascii
0002EA60	43	6F	6E	74	65	78	74	00	DC	04	52	74	6C	4C	6F	6F	Context.00 RtlLoc
0002EA70	6B	75	70	46	75	6E	63	74	69	6F	6E	45	6E	74	72	79	kupFunctionEntry
0002EA80	00	00	E3	04	52	74	6C	56	69	72	74	75	61	6C	55	6E	..0 RtlVirtualUn
0002EA90	77	69	6E	64	00	00	C0	05	55	6E	68	61	6E	64	6C	65	wind..A0 Unhandle
0002EAA0	64	45	78	63	65	70	74	69	6F	6E	46	69	6C	74	65	72	dExceptionFilter
0002EAB0	00	00	7F	05	53	65	74	55	6E	68	61	6E	64	6C	65	64	..0 SetUnhandled
0002EAC0	45	78	63	65	70	74	69	6F	6E	46	69	6C	74	65	72	00	ExceptionFilter.
0002EAD0	20	02	47	65	74	43	75	72	72	65	6E	74	50	72	6F	63	.. GetCurrentProc
0002EAE0	65	73	73	00	8C	03	49	73	50	72	6F	63	65	73	73	6F	ess.0 IsProcesso
0002EAF0	72	46	65	61	74	75	72	65	50	72	65	73	65	6E	74	00	rFeaturePresent.
0002EB00	6F	03	49	6E	69	74	69	61	6C	69	7A	65	53	4C	69	73	c0 InitializeSLis
0002EB10	74	48	65	61	64	00	85	03	49	73	44	65	62	75	67	67	tHead.0 IsDebugg
0002EB20	65	72	50	72	65	73	65	6E	74	00	DA	02	47	65	74	53	erPresent.0 GetS
0002EB30	74	61	72	74	75	70	49	6E	66	6F	57	00	E2	04	52	74	tartupInfoW.0 Rtl
0002EB40	6C	55	6E	77	69	6E	64	45	78	00	DE	04	52	74	6C	50	lUnwindEx.0 RtlP
0002EB50	63	54	6F	46	69	6C	65	48	65	61	64	65	72	00	68	04	cToFileHeader.h0
0002EB60	52	61	69	73	65	45	78	63	65	70	74	69	6F	6E	00	00	RaiseException..
0002EB70	41	05	53	65	74	4C	61	73	74	45	72	72	6F	72	00	00	A0 SetLastError..
0002EB80	34	01	45	6E	63	6F	64	65	50	6F	69	6E	74	65	72	00	40 EncodePointer..
0002EB90	6B	03	49	6E	69	74	69	61	6C	69	7A	65	43	72	69	74	k0 InitializeCrit
0002EBA0	69	63	61	6C	53	65	63	74	69	6F	6E	41	6E	64	53	70	icalSectionAndSp
0002EBB0	69	6E	43	6F	75	6E	74	00	B0	05	54	6C	73	41	6C	6C	inCount.0 TlsAll
0002EBC0	6F	63	00	00	B2	05	54	6C	73	47	65	74	56	61	6C	75	oc..0 TlsGetValu
0002EBD0	65	00	B3	05	54	6C	73	53	65	74	56	61	6C	75	65	00	e.0 TlsSetValue.
0002EBE0	B1	05	54	6C	73	46	72	65	65	00	B4	01	46	72	65	65	.0 TlsFree.0 Free
0002EBF0	4C	69	62	72	61	72	79	00	CA	03	4C	6F	61	64	4C	69	Library.0 LoadLi
0002EC00	62	72	61	72	79	45	78	57	00	00	79	04	52	65	61	64	braryExW..y0 Read
0002EC10	46	69	6C	65	00	00	F5	00	43	72	65	61	74	65	54	68	File..0 CreateTh
0002EC20	72	65	61	64	00	00	68	01	45	78	69	74	54	68	72	65	read..h0 ExitThre
0002EC30	61	64	00	00	B5	01	46	72	65	65	4C	69	62	72	61	72	ad..0 FreeLibrar
0002EC40	79	41	6E	64	45	78	69	74	54	68	72	65	61	64	00	00	yAndExitThread..
0002EC50	80	02	47	65	74	4D	6F	64	75	6C	65	48	61	6E	64	6C	GetModuleHandl
0002EC60	65	45	78	57	00	00	CA	01	47	65	74	43	50	49	6E	66	eExW..0 GetCPInf
0002EC70	6F	00	DC	02	47	65	74	53	74	64	48	61	6E	64	6C	65	o.0 GetStdHandle
0002EC80	00	00	25	06	57	72	69	74	65	46	69	6C	65	00	7D	02	..0 WriteFile.}
0002EC90	47	65	74	4D	6F	64	75	6C	65	46	69	6C	65	4E	61	6D	GetModuleFileNam
0002ECA0	65	57	00	00	67	01	45	78	69	74	50	72	6F	63	65	73	eW..q0 ExitProces
0002ECB0	73	00	DF	01	47	65	74	43	6F	6D	6D	61	6E	64	4C	69	s.0 GetCommandLi
0002ECC0	6E	65	41	00	E0	01	47	65	74	43	6F	6D	6D	61	6E	64	neA.0 GetCommand
0002ECD0	4C	69	6E	65	57	00	05	02	47	65	74	43	6F	6E	73	6F	LineW.0 GetConso
0002ECE0	6C	65	4D	6F	64	65	00	00	76	04	52	65	61	64	43	6F	leMode..0 ReadCo
0002ECF0	6E	73	6F	6C	65	57	00	00	58	02	47	65	74	46	69	6C	nsoleW..X GetFil
0002ED00	65	54	79	70	65	00	51	03	48	65	61	70	41	6C	6C	6F	eType.0 HeapAllo

Qui la risorsa del manifest file, ove si evincono i privilegi d'esecuzione e permessi di sicurezza:

Configuration Files	1 - [lang: 1033]
	<pre><?xml version='1.0' encoding='UTF-8' standalone='yes?> <assembly xmlns='urn:schemas-microsoft-com:asm.v1' manifestVersion='1.0'> <trustInfo xmlns='urn:schemas-microsoft-com:asm.v3'> <security> <requestedPrivileges> <requestedExecutionLevel level='asInvoker' uiAccess='false' /> </requestedPrivileges> </security> </trustInfo> </assembly></pre>

All'interno della funzione **fun_1400f4a4** si nota il richiamo di **QueryPerformanceCounter** al fine di monitorare l'utilizzo del Performance Counter ed individuare un'eventuale esecuzione all'interno di un ambiente virtualizzato:

```
__asm__("movups [rdx], xmm0");
fun_1400bd34(&rdx->f8, &rcx->f8, r8, r9b);
return rcx;
}

int64_t fun_1400f4a4() {
void* rsp1;
uint64_t rax2;
uint64_t v3;
uint64_t rax4;
uint64_t v5;

rsp1 = reinterpret_cast<void*>(reinterpret_cast<int64_t>(__zero_stack_offset()) - 40);
rax2 = reinterpret_cast<uint64_t>(QueryPerformanceFrequency(reinterpret_cast<int64_t>(rsp1) + 48));
if (! reinterpret_cast<int32_t>(&rax2) || ((rax2 = reinterpret_cast<uint64_t>(QueryPerformanceCounter(reinterpret_cast<int64_t>(r
    rax4 = 0xffffffffffff;
    g140032c00 = 0xffffffffffff;
} else {
    g140032c00 = v3;
    rax4 = v5;
}
g140032c08 = rax4;
return 0;
}

void fun_140003bbc() {
}

void fun_1400041d4() {
}

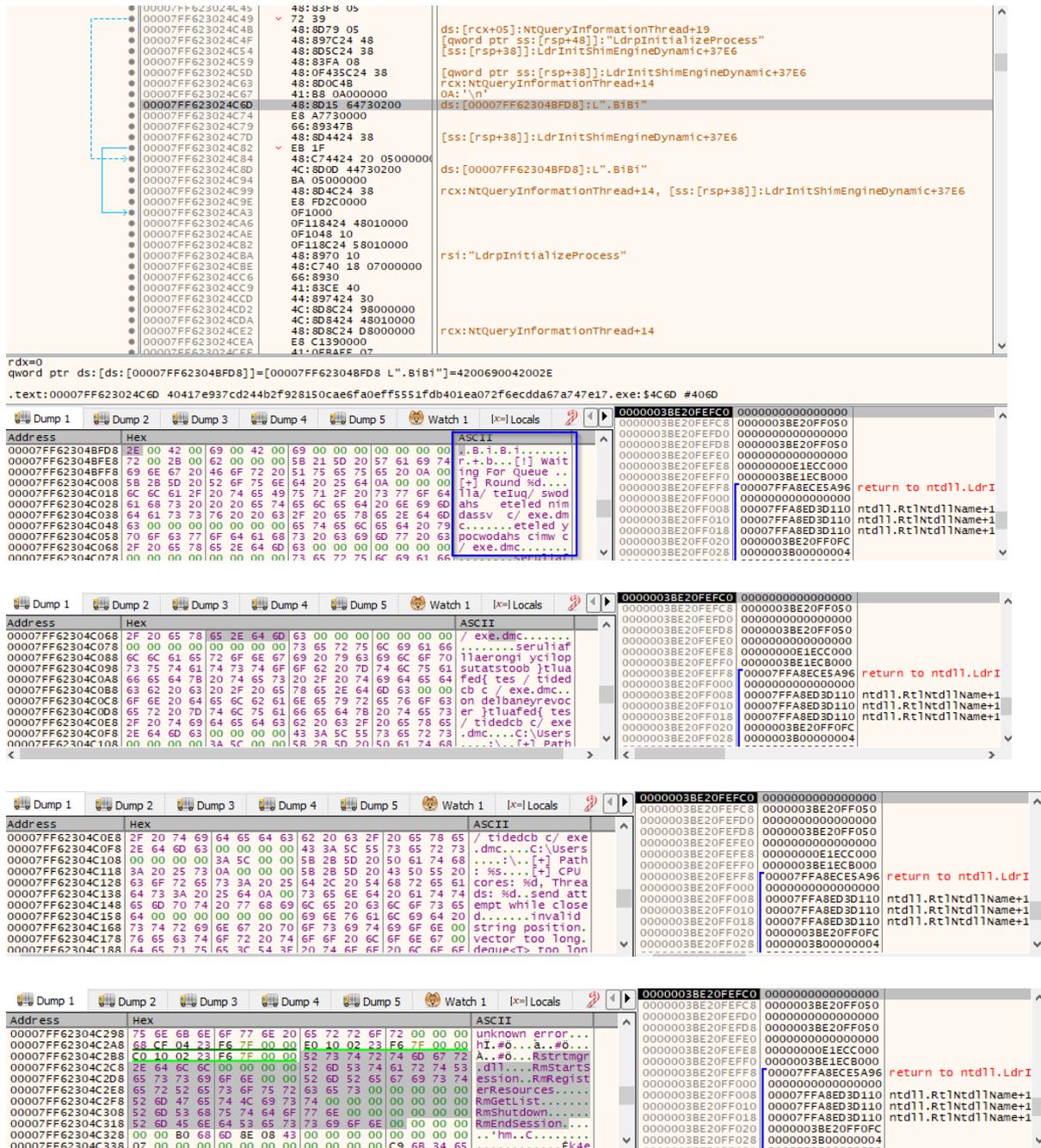
struct s276 {
int64_t f0;
void** f8;
};

struct s277 {
signed char[8] pad8;
void** f8;
};

struct s276* fun_1400066e0(struct s276* rcx, struct s277* rdx, void** r8, unsigned char r9b) {
__asm__("xorps xmm0, xmm0");
rcx->f0 = 0x140022610;
__asm__("movups [rdx], xmm0");
fun_1400bd34(&rdx->f8, &rcx->f8, r8, r9b);
}
```

Debugging

Effettuando una sessione di debugging possiamo riscontrare l'estensione .BiBi appesa ai files resi inaccessibili e stringhe di logging delle esecuzioni multithreading e dei comandi di Windows boot setting:



The screenshot displays a debugger interface with three main panels:

- Thread List:** Shows a list of threads with their IDs and names. A thread with ID `00007FF623048FD8` is highlighted, with its name `ds:[00007FF623048FD8]:L\".BiBi\"`.
- Memory Dump:** Shows a dump of memory at address `000003BE20FF00`. The dump is organized into columns for Address, Hex, and ASCII. The ASCII column contains the text `return to ntdll.LdrI`.
- Disassembly:** Shows the assembly instructions for the selected thread. The instruction `lea rdx, [ds:[00007FF623048FD8]=420069004200E]` is highlighted, which corresponds to the `lea` instruction mentioned in the text.

Qui un esempio di istruzione `lea` che copia il valore esadecimale dell'attributo contenente il comando di eliminazione delle copie shadow all'interno del registro `rdx`:

```

40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.00007FF62302555B
lea rdx,qword ptr ds:[7FF62304C018]; ds:[00007FF62304C018]:"11a/teIuq/swodahs eteled nimdassv c/ exe.dmc"
mov qword ptr ss:[rsp+50],rbx; [qword ptr ss:[rsp+50]]:"LdrpInitializeProcess"
lea rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:"LdrpInitializeProcess"
mov qword ptr ss:[rsp+60],rbx
lea r8d,qword ptr ds:[rbx+31]
CALL 40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.7FF623026990
cmp qword ptr ss:[rsp+68],10
lea rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:"LdrpInitializeProcess"
mov rdx,qword ptr ss:[rsp+60]
cmovae rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [qword ptr ss:[rsp+50]]:"LdrpInitializeProcess"
add rdx,rcx; rcx:NtQueryInformationThread+14
lea rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:"LdrpInitializeProcess"
cmp qword ptr ss:[rsp+68],10
cmovae rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [qword ptr ss:[rsp+50]]:"LdrpInitializeProcess"
CALL 40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.7FF623029E34
cmp qword ptr ss:[rsp+68],10
lea rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:"LdrpInitializeProcess"
cmovae rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [qword ptr ss:[rsp+50]]:"LdrpInitializeProcess"
sub rdx,rcx; rcx:NtQueryInformationThread+14

```

```

40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.00007FF6230255C1
movzx eax,byte ptr ds:[rcx]; byte ptr ds:[rcx]:NtQueryInformationThread+14
mov byte ptr ds:[rdx+rcx],al; byte ptr ds:[rdx+rcx]:NtQueryInformationThread+14
lea rcx,qword ptr ds:[rcx+1]; rcx:NtQueryInformationThread+14, ds:[rcx+1]:NtQueryInformationThread+15
test al,al
jne 40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.7FF623025551

```

rdx=0
qword ptr ds:[ds:[00007FF62304C018]]=[00007FF62304C018 "11a/teIuq/swodahs eteled nimdassv c/ exe.dmc"]=496574202F616C6C
.txt:00007FF62302555B 40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.exe:5555B #495B

Address	Hex	ASCII	Comment
00007FF62302555B	48 8D 15 B6 6A 02 00 48 89 5C 24 50 48 8D 4C 24	H...H...\$PH.L\$	
00007FF623025568	50 48 89 5C 24 60 44 8D 43 31 E8 16 14 00 00 48	PH...\$D.Cie...H	
00007FF623025578	83 7C 24 68 10 48 8D 4C 24 50 48 8B 54 24 60 48	.\$h.H.L\$PH.T\$H	
00007FF623025588	0F 43 4C 24 50 48 03 01 48 8D 4C 24 50 48 83 7C	.CL\$PH.Nh.L\$PH.I	
00007FF623025598	24 68 10 48 0F 43 4C 24 50 E8 8B 48 00 00 48 83	.\$h.H.CL\$P\$.H...H	
00007FF6230255A8	7C 24 68 10 48 8D 4C 24 50 48 8D 55 00 00 48 0F	.\$h.H.L\$PH.U.H.C	
00007FF6230255B8	4C 24 50 48 2B D1 0F B6 01 88 04 0A 48 8D 49 01	L\$PH.N...H.I.	
00007FF6230255C8	84 C0 75 F2 0F 57 C0 C7 44 24 70 68 00 00 00 48	.Aud.WACD\$ph...H	
00007FF6230255D8	8D 45 E0 45 33 C9 48 89 44 24 48 8D 55 00 48	.EBE\$Eh.D\$Hh.U.H	
00007FF6230255E8	8D 44 24 70 45 33 C0 48 89 44 24 40 33 C9 48 89	.D\$P\$E\$AH.D\$E\$Eh.	
00007FF6230255F8	8C 24 38 48 8E 8F 24 3A 0E 11 4E 8D 87 24 3A 0E	.\$e\$u.L\$E.C\$E\$E\$	

```

40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.00007FF62302555B
mov r8d,21; 21:!!'
mov qword ptr ss:[rsp+50],rbx; [qword ptr ss:[rsp+50]]:"LdrpInitializeProcess"
lea rcx,qword ptr ds:[7FF62304C050]; ds:[00007FF62304C050]:"eteled ypcwodahs c1mw c/ exe.dmc"
mov qword ptr ss:[rsp+60],rbx
lea rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:"LdrpInitializeProcess"
mov qword ptr ss:[rsp+68],F
CALL 40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.7FF623026990
cmp qword ptr ss:[rsp+68],10
lea rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:"LdrpInitializeProcess"
mov rdx,qword ptr ss:[rsp+60]
cmovae rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [qword ptr ss:[rsp+50]]:"LdrpInitializeProcess"
add rdx,rcx; rcx:NtQueryInformationThread+14
lea rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:"LdrpInitializeProcess"
cmp qword ptr ss:[rsp+68],10
cmovae rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [qword ptr ss:[rsp+50]]:"LdrpInitializeProcess"
CALL 40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.7FF623029E34
cmp qword ptr ss:[rsp+68],10
lea rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:"LdrpInitializeProcess"
lea rdx,qword ptr ss:[rbp]
cmovae rcx,qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [qword ptr ss:[rsp+50]]:"LdrpInitializeProcess"
sub rdx,rcx; rcx:NtQueryInformationThread+14

```

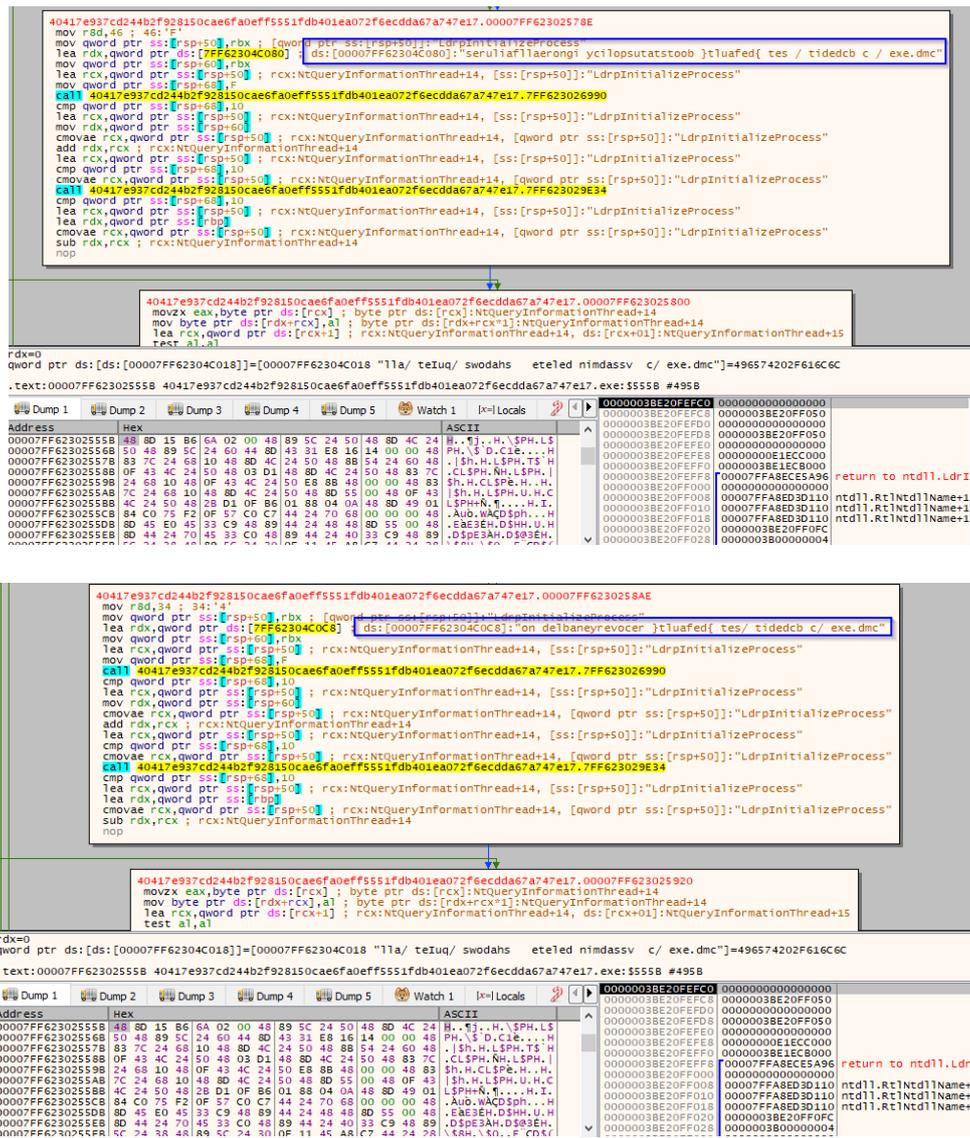
```

40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.00007FF6230256E0
movzx eax,byte ptr ds:[rcx]; byte ptr ds:[rcx]:NtQueryInformationThread+14
mov byte ptr ds:[rdx+rcx],al; byte ptr ds:[rdx+rcx]:NtQueryInformationThread+14
lea rcx,qword ptr ds:[rcx+1]; rcx:NtQueryInformationThread+14, ds:[rcx+1]:NtQueryInformationThread+15
test al,al
jne 40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.7FF6230256E0

```

rdx=0
qword ptr ds:[ds:[00007FF62304C018]]=[00007FF62304C018 "11a/teIuq/swodahs eteled nimdassv c/ exe.dmc"]=496574202F616C6C
.txt:00007FF62302555B 40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17.exe:5555B #495B

Address	Hex	ASCII	Comment
00007FF62302555B	48 8D 15 B6 6A 02 00 48 89 5C 24 50 48 8D 4C 24	H...H...\$PH.L\$	
00007FF623025568	50 48 89 5C 24 60 44 8D 43 31 E8 16 14 00 00 48	PH...\$D.Cie...H	
00007FF623025578	83 7C 24 68 10 48 8D 4C 24 50 48 8B 54 24 60 48	.\$h.H.L\$PH.T\$H	
00007FF623025588	0F 43 4C 24 50 48 03 01 48 8D 4C 24 50 48 83 7C	.CL\$PH.Nh.L\$PH.I	
00007FF623025598	24 68 10 48 0F 43 4C 24 50 E8 8B 48 00 00 48 83	.\$h.H.CL\$P\$.H...H	
00007FF6230255A8	7C 24 68 10 48 8D 4C 24 50 48 8D 55 00 00 48 0F	.\$h.H.L\$PH.U.H.C	
00007FF6230255B8	4C 24 50 48 2B D1 0F B6 01 88 04 0A 48 8D 49 01	L\$PH.N...H.I.	
00007FF6230255C8	84 C0 75 F2 0F 57 C0 C7 44 24 70 68 00 00 00 48	.Aud.WACD\$ph...H	
00007FF6230255D8	8D 45 E0 45 33 C9 48 89 44 24 48 8D 55 00 48	.EBE\$Eh.D\$Hh.U.H	
00007FF6230255E8	8D 44 24 70 45 33 C0 48 89 44 24 40 33 C9 48 89	.D\$P\$E\$AH.D\$E\$Eh.	
00007FF6230255F8	8C 24 38 48 8E 8F 24 3A 0E 11 4E 8D 87 24 3A 0E	.\$e\$u.L\$E.C\$E\$E\$	



The screenshot displays a debugger window with assembly code and a memory dump. The assembly code includes instructions such as:

```

mov r8d, 46; 46:'
mov qword ptr ss:[rsp+50], rdx; [qword ptr ss:[rsp+50]]:'LdrpInitializeProcess
lea rcx, qword ptr ds:[FF62304C080]; ds:[00007FF62304C080]:'serul1ar1aerong1 ycl1opsutastob0 t]luafed{ tes / tidedcb c / exe.dmc"
mov qword ptr ss:[rsp+60], rdx
lea rcx, qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:'LdrpInitializeProcess
mov qword ptr ss:[rsp+68], 10
lea rcx, qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:'LdrpInitializeProcess
mov rdx, qword ptr ss:[rsp+60]
cmovae rcx, qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [qword ptr ss:[rsp+50]]:'LdrpInitializeProcess
lea rcx, qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:'LdrpInitializeProcess
cmp qword ptr ss:[rsp+68], 10
cmovae rcx, qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [qword ptr ss:[rsp+50]]:'LdrpInitializeProcess
CALL 4041E937CD244B2F928150CAE6FA0EFF551FDB401EA072F6ECDDA67A747E17.7FF62302E934
cmp qword ptr ss:[rsp+68], 10
lea rcx, qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [ss:[rsp+50]]:'LdrpInitializeProcess
lea rdx, qword ptr ss:[rbp]
cmovae rcx, qword ptr ss:[rsp+50]; rcx:NtQueryInformationThread+14, [qword ptr ss:[rsp+50]]:'LdrpInitializeProcess
sub rdx, rcx; rcx:NtQueryInformationThread+14
nop

```

The memory dump shows hex and ASCII values, with a highlighted instruction 'return to ntdll.LdrI'.

Le classificazioni OSINT dell'artefatto sottoposto a disamina fanno riferimento alla firma "Trojan.Win.BiBiWiper.C5541532":

57 / 72

57 security vendors and 2 sandboxes flagged this file as malicious

40417e937cd244b2f928150cae6fa0eff551fdb401ea072f6ecd8a67a747e17

bibi.exe

Size: 203.00 KB | Last Analysis Date: a moment ago

peexe 64bits checks-cpu-name

Community Score

DETECTION DETAILS RELATIONS BEHAVIOR COMMUNITY 16

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label: trojan.stealerbibi | Threat categories: trojan ransomware | Family labels: stealer bibi wiper

Security vendors' analysis

AhnLab-V3	Trojan.Win.BiB.Wiper.C5541532	Alibaba	Trojan.PSW.Win32/Stealer.174fc9b9
ALYac	Trojan.Agent.Wiper	Antiy-AVL	Trojan.Win64.Filecoder
Arcabit	Trojan.Generic.D42E85A7	Avast	Win32:BiBi-B [Wpr]
AVG	Win32:BiBi-B [Wpr]	Avira (no cloud)	TR/FileCoder.rqgcg
BitDefender	Trojan.GenericKD.70157735	Bkav Pro	W64.AI.DetectMalware

Qui le identificazioni di alcune regole IDS che fanno riferimento ad operazioni ICMP e Ping:

Crowdsourced IDS rules

- Matches rule PROTOCOL-ICMP PING Windows at Snort registered user ruleset
↳ misc-activity
- Matches rule PROTOCOL-ICMP Unusual PING detected at Snort registered user ruleset
↳ successful-recon-limited
- Matches rule PROTOCOL-ICMP PING at Snort registered user ruleset
↳ misc-activity
- Matches rule PROTOCOL-ICMP Echo Reply at Snort registered user ruleset
↳ misc-activity

Ecco un esempio di malicious detonation che prende in considerazione i files e li rende in accessibili aggiungendo anche l'estensione finale .bibi ed un attributo numerico di riferimento.

Activity Summary

-  F:\connect.avi
F:\xcsvwuz7h.bibi1
-  F:\dashBorder_192.bmp
F:\nu4nrzybhd.bibi1
-  F:\delete.avi
F:\kkesj8mktm.bibi1
-  F:\toolbar.bmp
F:\bjn9ahsxc.bibi1

Files Dropped

- + C:\Users\Default User\Application Data\Microsoft\Windows\SendTo\CVticrMb7U.BiBi3 (copy)
- + C:\Users\Default User\Application Data\Microsoft\Windows\SendTo\mji6rELaVg.BiBi3 (copy)
- + C:\Users\Default User\l7fVcM7kEH.BiBi2 (copy)
- + C:\Users\Default User\Local Settings\Microsoft\Windows\Shell\j8RbPMfTmV.BiBi4 (copy)
- + C:\Users\Default User\M9k5iX5yCh.BiBi2 (copy)
- + C:\Users\Default User\SendTo\UMGBLwzpGG.BiBi4 (copy)
- + C:\Users\Default User\guKHsGa1zb.BiBi2 (copy)
- + C:\Users\Default User\bl7Y4cu5z.BiBi2 (copy)
- + C:\Users\Default User\vs2Werllgv.BiBi2 (copy)
- + C:\Users\Default\5f915EmUbN.BiBi1

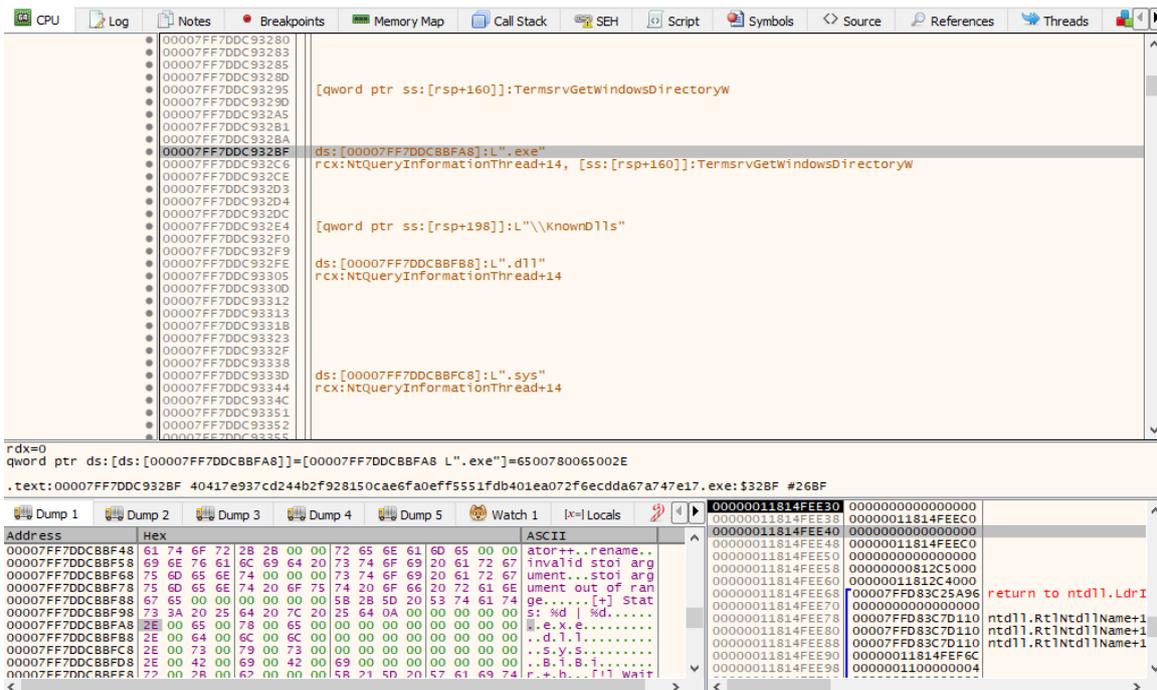
Nello screenshot sottostante la gestione degli attributi di sistema contestualmente all'ottenimento dei files da sottoporre a sovrascrittura rilevati dall'esecuzione di un ciclo **while**:

```

}
*reinterpret_cast<int32_t*>(&rbx417) = *reinterpret_cast<int32_t*>(&rbx417) - 1;
*reinterpret_cast<int32_t*>(&rbx417 + 4) = 0;
if (*reinterpret_cast<int32_t*>(&rbx417) < 0)
    break;
}
addr_140005afd_492:
rbx417 = v428;
rdi429 = v410;
while (rbx417 != rdi429) {
    fun_140006b90(rbx417, reinterpret_cast<uint64_t*>(rbp396) + 0xffffffffffff87, r8_403);
    rdx430 = reinterpret_cast<void**>(reinterpret_cast<uint64_t*>(rbp396) + 0xffffffffffff87);
    if (0) {
        rdx430 = v424;
    }
    rcx404 = reinterpret_cast<void**>("[+] Path: %s\n");
    fun_140001030("[+] Path: %s\n", rdx430, r8_403, 0, "[+] Path: %s\n", rdx430, r8_403, 0);
    rsp413 = reinterpret_cast<void**>(reinterpret_cast<uint64_t*>(rsp413) - 8 + 8 - 8 + 8);
    if (!1) {
        rdx425 = reinterpret_cast<void**>(8);
        rcx404 = v424;
        rax431 = rcx404;
        if (1)
            goto addr_140005b6a_498;
        rdx425 = reinterpret_cast<void**>(47);
        rcx404 = *reinterpret_cast<void**>(rcx404 + 0xffffffffffff88);
        if (reinterpret_cast<unsigned char*>(rax431) - reinterpret_cast<unsigned char*>(rcx404) + 0xffffffffffff88 > 31)
            goto addr_1400060c7_500;
        addr_140005b6a_498:
        fun_14000a87c(rcx404, rdx425, rcx404, rdx425);
        rsp413 = reinterpret_cast<void**>(reinterpret_cast<uint64_t*>(rsp413) - 8 + 8);
    }
    rbx417 = rbx417 + 32;
}
eax432 = fun_1400090b8(rcx404);
*reinterpret_cast<int32_t*>(&r9_433) = eax432;
*reinterpret_cast<int32_t*>(&r9_433 + 4) = 0;
*reinterpret_cast<int32_t*>(&rax434) = eax432;
*reinterpret_cast<int32_t*>(reinterpret_cast<int64_t*>(&rax434) + 4) = 0;
rax435 = rax434 / reinterpret_cast<uint64_t*>(reinterpret_cast<int64_t*>(reinterpret_cast<unsigned char*>(v410) - v43)
if (*reinterpret_cast<int32_t*>(&rax435) < reinterpret_cast<int32_t*>(1)) {
    *reinterpret_cast<uint32_t*>(&rax435) = 1;
    *reinterpret_cast<int32_t*>(reinterpret_cast<int64_t*>(&rax435) + 4) = 0;
}
}

```

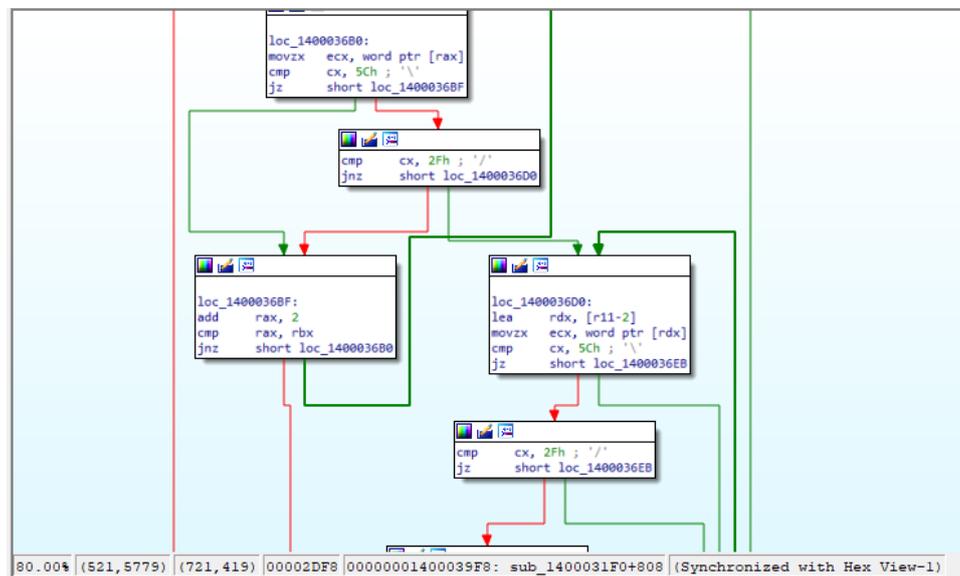
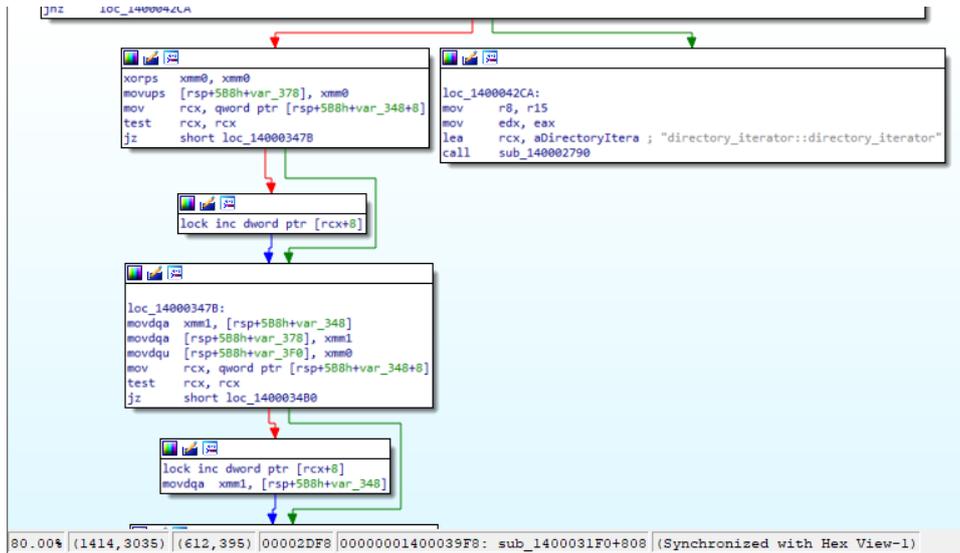
Vengono "saltati" i seguenti tipi di files durante l'esecuzione del malware: **.exe**, **.dll** e **.sys**.

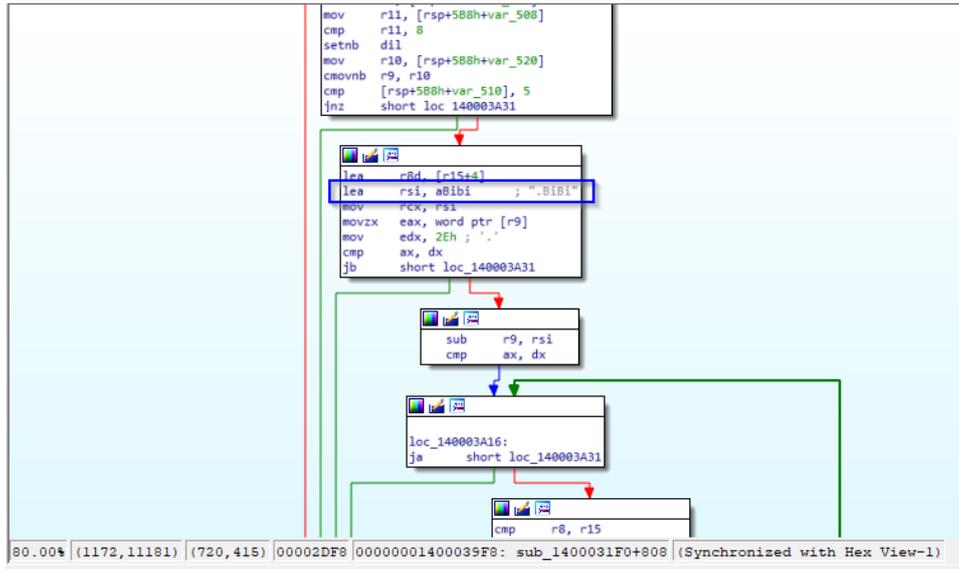


The screenshot shows a debugger window with the following content:

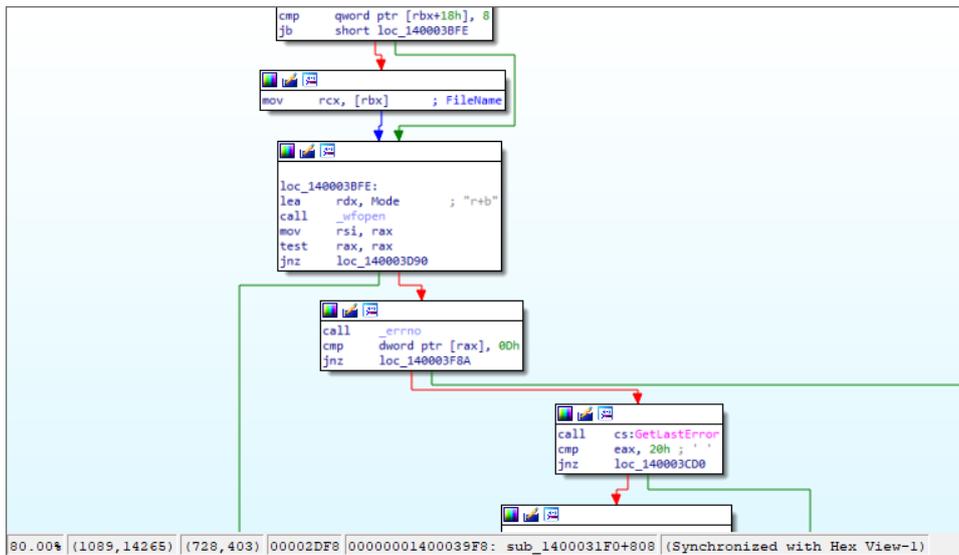
- Memory Dump:**
 - Address: 00007FF7DDC93280 to 00007FF7DDC93355
 - Hex: 61 74 6F 72 28 28 00 00 72 65 6E 61 6D 65 00 00
 - ASCII: ator+..rename..
 - Address: 00007FF7DDC93288 to 00007FF7DDC93355
 - Hex: 69 6E 76 61 6C 69 64 20 73 74 6F 69 20 61 72 67
 - ASCII: invalid stoi arg
 - Address: 00007FF7DDC9328B to 00007FF7DDC93355
 - Hex: 75 6D 65 6E 74 20 6F 75 74 20 6F 66 20 61 72 67
 - ASCII: ument... stoi arg
 - Address: 00007FF7DDC9328E to 00007FF7DDC93355
 - Hex: 67 65 00 00 00 00 00 58 28 5D 20 53 74 61 74
 - ASCII: ge.....[+] Stat
 - Address: 00007FF7DDC93291 to 00007FF7DDC93355
 - Hex: 73 3A 20 25 64 20 7C 20 25 64 0A 00 00 00 00
 - ASCII: s: %d | %d.....
 - Address: 00007FF7DDC93294 to 00007FF7DDC93355
 - Hex: 2E 00 65 00 78 00 6C 00 00 00 00 00 00 00
 - ASCII: .e.x.e.....
 - Address: 00007FF7DDC93297 to 00007FF7DDC93355
 - Hex: 2E 00 64 00 6C 00 6C 00 00 00 00 00 00 00
 - ASCII: .d.t.l.....
 - Address: 00007FF7DDC9329A to 00007FF7DDC93355
 - Hex: 2E 00 73 00 79 00 73 00 00 00 00 00 00 00
 - ASCII: .s.y.s.....
 - Address: 00007FF7DDC9329D to 00007FF7DDC93355
 - Hex: 2E 00 42 00 69 00 42 00 69 00 00 00 00 00
 - ASCII: .B.I.B.I.....
 - Address: 00007FF7DDC932A2 to 00007FF7DDC93355
 - Hex: 72 00 28 00 62 00 00 58 21 5D 20 57 61 69 74
 - ASCII: r.+h.....[] Wait
- Disassembly:**
 - Address: 00007FF7DDC93280 to 00007FF7DDC93285: [qword ptr ss:[rsp+160]]:TermsrvGetWindowsDirectory
 - Address: 00007FF7DDC9328F to 00007FF7DDC93294: ds:[00007FF7DDC8BF8]:L".exe"
 - Address: 00007FF7DDC93297 to 00007FF7DDC932A2: [qword ptr ss:[rsp+198]]:L"\knowndlls"
 - Address: 00007FF7DDC9329A to 00007FF7DDC9329F: ds:[00007FF7DDC8BF8]:L".dll"
 - Address: 00007FF7DDC932A2 to 00007FF7DDC932A7: ds:[00007FF7DDC8BF8]:L".sys"
- Registers:**
 - rdx=0
 - qword ptr ds:[ds:[00007FF7DDC8BF8]:L".exe"]=[00007FF7DDC8BF8 L".exe"]=6500780065002E
- Watch:**
 - 00000011814FEE30: 0000000000000000
 - 00000011814FEE38: 00000011814FEEC0
 - 00000011814FEE40: 0000000000000000
 - 00000011814FEE48: 00000011814FEEC0
 - 00000011814FEE50: 0000000000000000
 - 00000011814FEE58: 0000000812C5000
 - 00000011814FEE60: 00000011812C4000
 - 00000011814FEE68: 00007FFD83C25A96
 - 00000011814FEE70: 0000000000000000
 - 00000011814FEE78: 00007FFD83C7D110
 - 00000011814FEE80: 00007FFD83C7D110
 - 00000011814FEE88: 00007FFD83C7D110
 - 00000011814FEE90: 00000011814F5F6C
 - 00000011814FEE98: 0000001100000004

All'interno della funzione **sub_1400031F0** viene effettuata l'iterazione delle directories di sistema, una volta identificati i files da rendere inaccessibili essi vengono sovrascritti in parte con un pattern randomico generato ed inserito all'interno della stream evidenziabile nella funzione **sub_1400048D0**. Dopo aver effettuato l'azione di sovrascrittura, i files presi in considerazione vengono rinominati con l'estensione appesa **.BiBi** ed una cifra specifica.

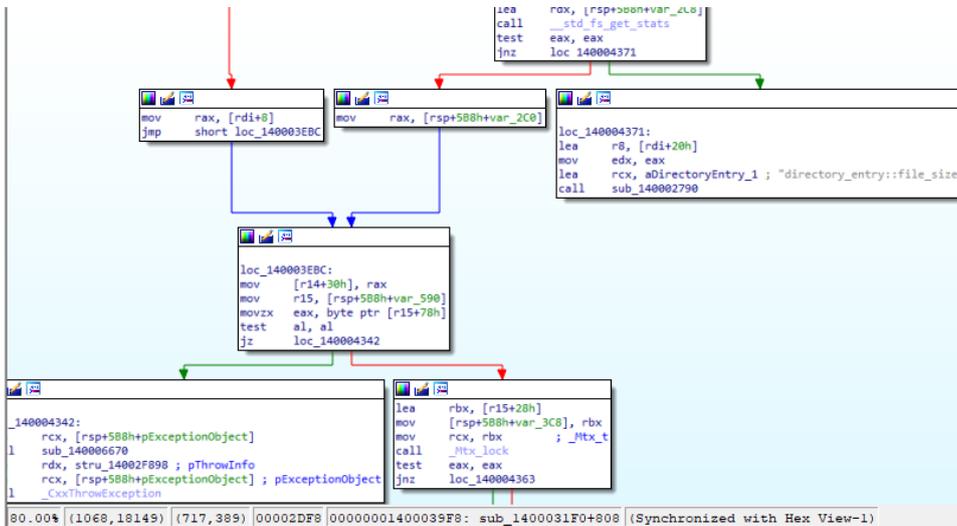
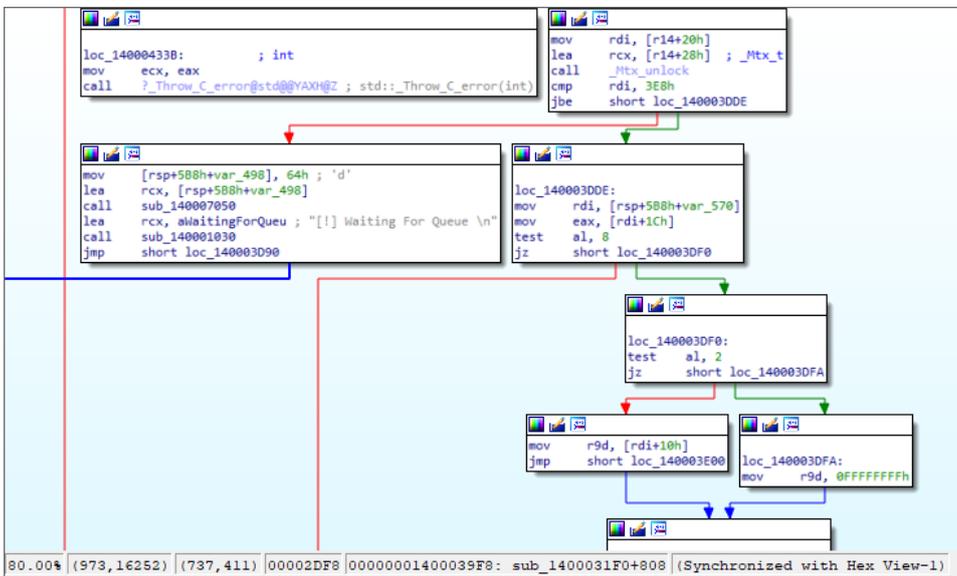
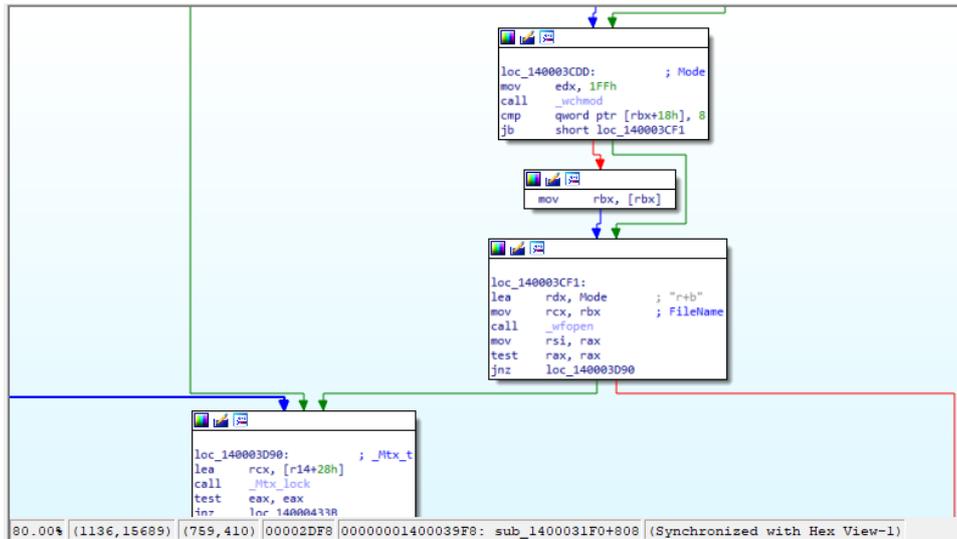




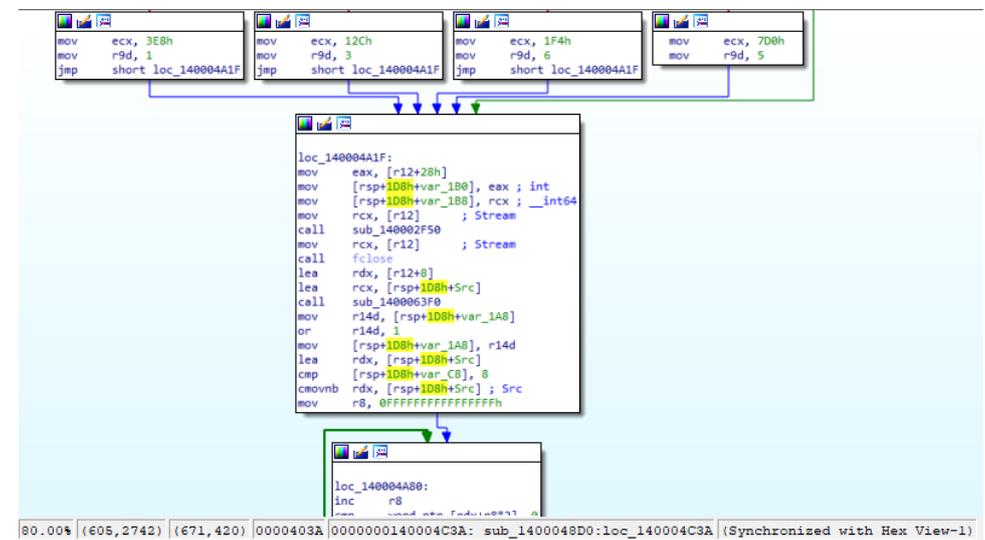
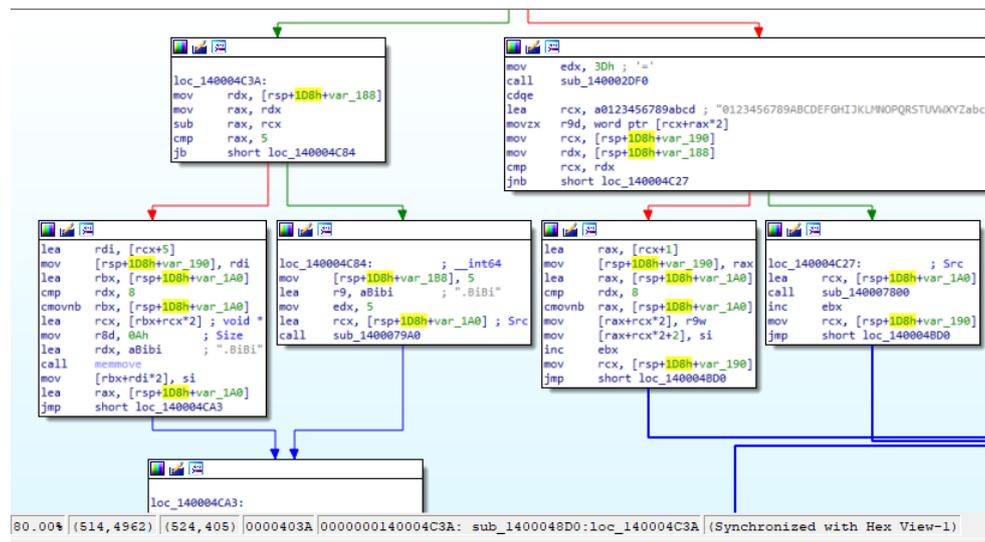
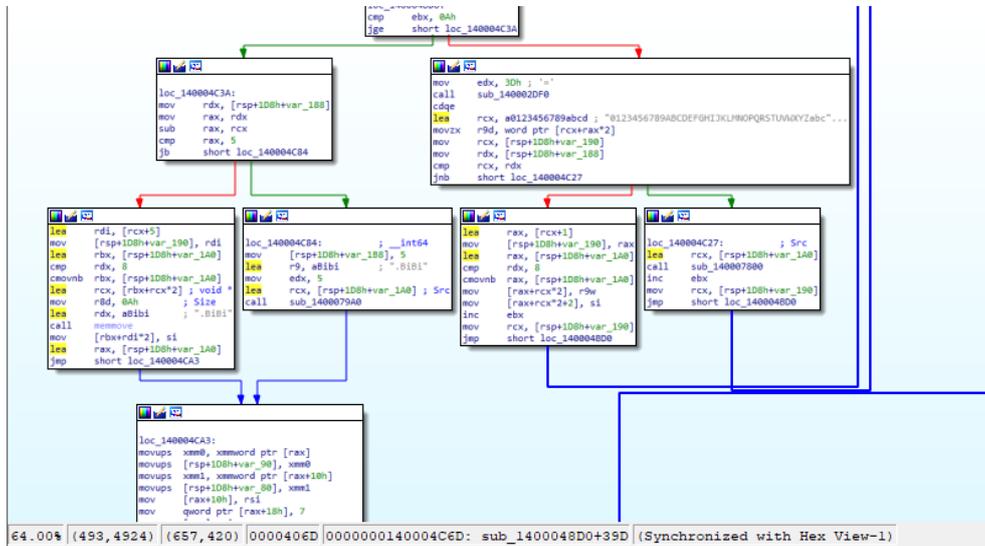
I files vengono aperti con la configurazione **r+b (read or write mode)**



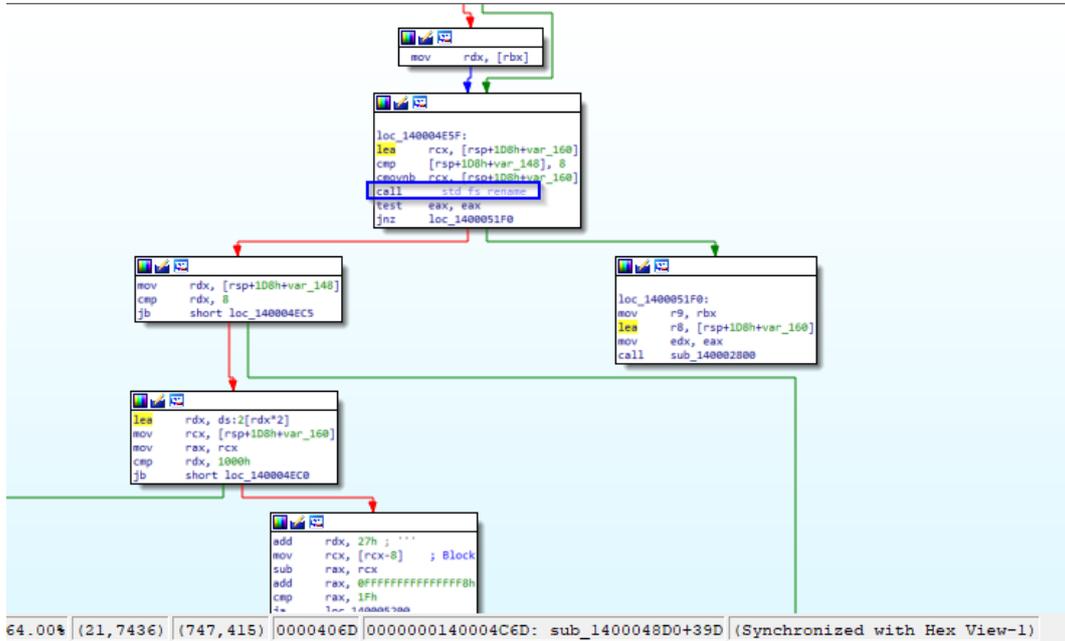
Vengono poi posti in *lock* gli oggetti *mutex* per i files in questione al fine di permettere un accesso esclusivo ad essi, senza interferenze da parte di eventuali processi esterni:



Qui la gestione del pattern randomico contestuale alla sovrascrittura dei files enumerati:



Si noti l'azione di rinomina dei files sovrascritti:



Registers: `edx=0`, `3D '=`

Memory Dump:

Address	Hex	ASCII
00007FF7DDCB110	5B 2B 5D 20 50 61 74 68 3A 20 25 73 0A 00 00 00	[+] Path: %s...
00007FF7DDCB120	5B 2B 5D 20 43 50 55 20 63 6F 72 65 73 3A 20 25	[+] CPU core: %
00007FF7DDCB130	64 0C 20 54 68 72 65 61 64 73 3A 20 25 64 0A 00	d, Threads: %d,
00007FF7DDCB140	73 65 6E 64 20 61 74 74 65 6D 70 74 20 77 68 69	send attempt whi
00007FF7DDCB150	6C 65 20 63 6C 6F 73 65 64 00 00 00 00 00 00	le closed.....
00007FF7DDCB160	69 6E 76 61 6C 69 64 20 73 74 72 69 6E 67 20 70	invalid string p
00007FF7DDCB170	6F 73 69 74 69 6F 6E 00 76 65 63 74 6F 72 20 74	osition.vector t
00007FF7DDCB180	6F 6F 20 6C 6F 6E 67 00 64 65 71 75 65 3C 54 3E	oo long deque<T>
00007FF7DDCB190	20 74 6F 6F 20 6C 6F 6E 67 00 00 00 00 00 00	too long.....
00007FF7DDCB1A0	30 00 31 00 32 00 33 00 34 00 35 00 36 00 37 00	[1.2.3.4.5.6.7.
00007FF7DDCB1B0	38 00 39 00 40 00 41 00 42 00 43 00 44 00 45 00	8.9.A.R.C.D.F.F.

IOCs:

- e26bba0304f14ef96beb60376791d32c
- 24f6785ca2e82d1d1d61f4cb01d5e753f80445cf
- 40417e937cd244b2f928150cae6fa0eff5551fdb401ea072f6ecdda67a747e17
- .BiBi
- 2e 42 69 42 69

Regola YARA

```
rule BiBiRule
{
  strings:
    $strBiBi = ".BiBi"
    $hexBiBi = { 2e 42 69 42 69 }

  condition:
    $strBiBi or $hexBiBi
}
```

CONCLUSIONI:

BiBi Wiper è una minaccia che ripercorre la falsariga dei wipers utilizzati nel contesto del conflitto Russo-Ucraino, come HermeticWiper o IsaacWiper (sviluppati e diffusi immediatamente prima dell'invasione perpetrata dalla Russia, avvenuta de facto il 24 Febbraio 2022).

Nel caso specifico, tuttavia, vi sono alcuni elementi di differenziazione rispetto alle sopra citate minacce: i dati ed i files presi in considerazione nella fase di enumeration vengono resi inaccessibili e sovrascritti mediante un pattern randomico. Tuttavia, il behaviour analizzato non è afferibile ad una classificazione di tipologia Ransomware, in quanto non viene richiesto alcun riscatto per il recupero dei files mediante una ransomnote creata a bordo delle macchine infette. L'unico obiettivo del threat è quello di perpetrare la propria azione distruttiva nei confronti delle principali infrastrutture critiche avversarie ed essa è associabile al sempre più presente concetto di guerra ibrida che abbiamo imparato a conoscere a causa della delicata situazione geopolitica attuale. Una caratteristica fondamentale di tale concetto è rappresentabile dal fatto che, anche senza un'azione di belligeranza militare, si possono comunque ottenere risultati devastanti. Vi è stata inoltre un'attenzione nel gestire anche le risorse ed i files potenzialmente in uso da altri processi esterni e a modificare le impostazioni di avvio di Windows, nonché nell'eliminare le copie shadow al fine di massimizzare l'impatto del threat.

Il crescente e costante rischio di una situazione geopolitica sempre più compromessa e deteriorata, porta ad ipotizzare che lo sviluppo e la distribuzione di tali malware siano destinati ad aumentare. Queste minacce saranno sempre più sofisticate, evasive e distruttive.

Riferimenti:

[0] (introduzione in merito a BiBi Wiper): [BiBi Wiper Used in the Israel-Hamas War Now Runs on Windows \(blackberry.com\)](#)