

Pratica S11/L3 - Malware analysis con OllyDBG

L'esercizio di oggi consiste nell'utilizzare OllyDBG per effettuare un'analisi dinamica avanzata di un malware presente sulla nostra macchina e rispondere a questi quesiti:

1. All'indirizzo 0040106E il Malware effettua una chiamata di funzione alla funzione «CreateProcess». Qual è il valore del parametro «CommandLine» che viene passato sullo stack? (1)
2. Inserite un breakpoint software all'indirizzo 004015A3. Qual è il valore del registro EDX? (2) Eseguite a questo punto uno «step-into». Indicate qual è ora il valore del registro EDX (3) motivando la risposta (4). Che istruzione è stata eseguita? (5)
3. Inserite un secondo breakpoint all'indirizzo di memoria 004015AF. Qual è il valore del registro ECX? (6) Eseguite un step-into. Qual è ora il valore di ECX? (7) Spiegate quale istruzione è stata eseguita (8).

BONUS: spiegare a grandi linee il funzionamento del malware

1. All'indirizzo 0040106E il Malware effettua una chiamata di funzione alla funzione «CreateProcess». Qual è il valore del parametro «CommandLine» che viene passato sullo stack? (1)

```
00401063 . 8055 F0      LEA EDI, DWORD PTR SS:[EBP-10]
00401065 . 52          PUSH EDI
00401067 . 0D45 A8      LEA EAX, DWORD PTR SS:[EBP-58]
00401069 . 50          PUSH EAX
0040106B . 6A 00       PUSH 0
0040106D . 6A 00       PUSH 0
0040106F . 6A 00       PUSH 0
00401071 . 6A 01       PUSH 1
00401073 . 6A 00       PUSH 0
00401075 . 6A 00       PUSH 0
00401077 . 68 30504000 PUSH Malware_.00405030
00401079 . 6A 00       PUSH 0
0040106E . FF15 04404000 CALL DWORD PTR DS:[&KERNEL32.CreateProcessA]
00401074 . 8945 EC      MOV DWORD PTR SS:[EBP-14], EAX
```

```
pStartupInfo
CurrentDir = NULL
Environment = NULL
CreationFlags = 0
InheritHandles = TRUE
pThreadSecurity = NULL
pProcessSecurity = NULL
CommandLine = "cmd"
ModuleFileName = NULL
CreateProcessA
```

```
S 0 FS 00E
T 0 GS 00E
D 0
I 0 LastEx
EFL 0000021
ST0 empty
ST1 empty
ST2 empty
ST3 empty
ST4 empty
ST5 empty
ST6 empty
ST7 empty
```

Il parametro <commandLine> che viene passato sullo stack è "cmd"

2. Inserite un breakpoint software all'indirizzo 004015A3. Qual è il valore del registro EDX? (2) Eseguite a questo punto uno «step-into». Indicate qual è ora il valore del registro EDX (3) motivando la risposta (4). Che istruzione è stata eseguita? (5)

```
CPU - main thread, module Malware_
00401572 . 55          PUSH EBP
00401574 . 8BEC        MOV EBP, ESP
00401576 . 6A FF       PUSH -1
00401578 . 68 C0404000 PUSH Malware_.004040C0
0040157A . 68 30204000 PUSH Malware_.004020C0
0040157C . 64A1 00000000 MOV EDI, DWORD PTR FS:[0]
0040157E . 50          PUSH EDI
00401580 . 6418925 000000 MOV DWORD PTR FS:[0], ESP
00401582 . 8BEC 10     SUB ESP, 10
00401584 . 53          PUSH EBX
00401586 . 56          PUSH ESI
00401588 . 57          PUSH EDI
0040158A . 9965 E9     MOV DWORD PTR SS:[EBP-10], ESP
0040158C . FF15 30404000 CALL DWORD PTR DS:[&KERNEL32.GetVersion]
0040158E . 33D2        XOR EDX, EDX
00401590 . 9044        MOV DL, AH
00401592 . 8915 D4524000 MOV DWORD PTR DS:[4052D4], EDX
```

```
SE handler installation
kernel32.GetVersion
```

```
Registers (FPU)
EAX 00200105
ECX 77FDF000
EDX 00000A28
EBX 77FDF000
ESP 0012FF54
EBP 0012FFC0
ESI FFFFFFFF
EDI 7C910208 ntdll.7C910208
EIP 004015A3 Malware_.004015A3
C 0 ES 0023 32bit 0(FFFFFFFF)
P 1 CS 001B 32bit 0(FFFFFFFF)
D 0 SS 0023 32bit 0(FFFFFFFF)
I 0 DS 0023 32bit 0(FFFFFFFF)
F 0 FS 003B 32bit 7FDE000(FFF)
T 0 GS 0000 NULL
```

Il valore del registro EDX all'indirizzo 004015A3 è di 00000A28.

Esegui uno <step-into>.

Possiamo vedere che il valore di EDX è diventato 00000000 poichè viene fatto uno XOR tra EDX ed EDX quindi si effettua uno 'zeroing' per settare a 0 il valore di EDX, perchè il risultato di uno XOR tra due valori uguale risulta sempre 0.

3. Inserite un secondo breakpoint all'indirizzo di memoria 004015AF. Qual è il valore del registro ECX? (6) Eseguite un step-into. Qual è ora il valore di ECX? (7) Spiegate quale istruzione è stata eseguita (8).

Il valore di ECX all'indirizzo 004015AF è 0A280105.

Eseguo uno <step-into>.

Il valore di ECX diventa 00000005 perchè viene fatta l'istruzione AND tra ECX e 0, per arrivare a questo risultato possiamo trasformare i due valori in binario e eseguire l'AND tra i due.

BONUS: spiegare a grandi linee il funzionamento del malware

Sfruttando il tool CFF Explorer ho ricavato l'hash del malware e controllando su virus total risulta essere un trojan.