



# OllyDBG

# Indice

- Traccia
- CreateProcess
- Breakpoint software 1
- Step-into 1
- Breakpoint software 2
- Step-into 2
- Bonus

---

# Traccia

Fate riferimento al malware: **Malware\_U3\_W3\_L3**, presente all'interno della cartella

**Esercizio\_Pratico\_U3\_W3\_L3** sul desktop della macchina virtuale dedicata all'analisi dei malware.

Rispondete ai seguenti quesiti utilizzando OllyDBG.

- 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)**
- 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)**
- 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



# CreateProcess

00401050	. 894D E4	MOV DWORD PTR SS:[EBP-1C],ECX	pProcessInfo pStartupInfo CurrentDir = NULL pEnvironment = NULL CreationFlags = 0 InheritHandles = TRUE pThreadSecurity = NULL pProcessSecurity = NULL CommandLine = "cmd" ModuleFileName = NULL
00401053	. 8D55 F0	LEA EDX,DWORD PTR SS:[EBP-10]	
00401056	. 52	PUSH EDX	
00401057	. 8D45 A8	LEA EAX,DWORD PTR SS:[EBP-58]	
0040105A	. 50	PUSH EAX	
0040105B	. 6A 00	PUSH 0	
0040105D	. 6A 00	PUSH 0	
0040105F	. 6A 00	PUSH 0	
00401061	. 6A 01	PUSH 1	
00401063	. 6A 00	PUSH 0	
00401065	. 6A 00	PUSH 0	
00401067	. 68 30504000	PUSH Malware_.00405030	
0040106C	. 6A 00	PUSH 0	
0040106E	. FF15 04404000	CALL DWORD PTR DS:[<KERNEL32.CreateProcessA>]	
00401074	. 8945 EC	MOV DWORD PTR SS:[EBP-14],EAX	

Come si vede dalla figura qui sopra sulla riga 0040106E il malware crea il processo "A". Sullo stack viene inserito il "CommanLine" dandogli il valore "cmd".

# Breakpoint software 1

Dopo aver impostato il breakpoint e avviato l'esecuzione con il pulsante "play", il programma si bloccherà all'istruzione XOR EDX,EDX. Prima dell'esecuzione di questa istruzione, il valore del registro è "00001DB1".

			Registers (FPU)
0040158C	. 50	PUSH EAX	EAX 10B10106
0040158D	. 64:8925 000000	MOV DWORD PTR FS:[0],ESP	ECX 7EFDE000
00401594	. 03EC 10	SUB ESP,10	EDX 000010B1
00401597	. 53	PUSH EBX	EBX 7EFDE000
00401598	. 56	PUSH ESI	ESP 0018FF5C
00401599	. 57	PUSH EDI	EBP 0018FF88
0040159A	. 8965 E8	MOV DWORD PTR SS:[EBP-18],ESP	ESI 00000000
0040159D	. FF15 30404000	CALL DWORD PTR DS:[<&KERNEL32.GetVersion>]	EDI 00000000
004015A3	. 33D2	XOR EDX,EDX	EIP 004015A3 Malware_.004015A3
004015A5	. 8AD4	MOV DL,AH	C 0 ES 002B 32bit 0(FFFFFFFF)
004015A7	. 8915 D4524000	MOV DWORD PTR DS:[4052D4],EDX	P 1 CS 0023 32bit 0(FFFFFFFF)
004015AD	. 8BC8	MOV ECX,EAX	A 0 SS 002B 32bit 0(FFFFFFFF)
004015AF	. 81E1 FF000000	AND ECX,0FF	
004015B5	. 890D 00524000	MOV DWORD PTR DS:[4052D0],ECX	
004015BB	. C1F1 0A	SHL ECX,8	

# Step-into 1

Dopo aver eseguito lo step-into, l'istruzione XOR EDX,EDX viene eseguita, il che equivale fondamentalmente a inizializzare a zero una variabile. Quindi, dopo lo step-into, il valore di EDX sarà 0.

Address	Disassembly	Comment
00401577	55	PUSH EBP
00401578	8BEC	MOV EBP,ESP
0040157A	6A FF	PUSH -1
0040157C	68 C0404000	PUSH Malware_.004040C0
00401581	68 3C204000	PUSH Malware_.0040203C
00401586	64:A1 00000000	MOV EAX,DWORD PTR FS:[0]
0040158C	50	PUSH EAX
0040158D	64:8925 00000000	MOV DWORD PTR FS:[0],ESP
00401594	83EC 10	SUB ESP,10
00401597	53	PUSH EBX
00401598	56	PUSH ESI
00401599	57	PUSH EDI
0040159A	8965 E8	MOV DWORD PTR SS:[EBP-18],ESP
0040159D	FF15 30404000	CALL DWORD PTR DS:[<&KERNEL32.GetVersion>]
004015A5	33D2	XOR EDX,EDX
004015A7	8AD4	MOV DL,AH
004015A7	8915 D4524000	MOV DWORD PTR DS:[4052D4],EDX
004015AD	8BC8	MOV ECX,EAX
004015AF	81E1 FF000000	AND ECX,0FF

**Registers (FPU)**  
EAX 10B10106  
ECX 7EFD0000  
EDX 00000000  
EBX 7EFD0000  
ESP 0018FF5C  
EBP 0018FF88  
ESI 00000000  
EDI 00000000  
EIP 004015A5 Malware\_.004015A5  
C 0 ES 002B 32bit 0(FFFFFFFF)  
P 1 CS 0023 32bit 0(FFFFFFFF)  
A 0 SS 002B 32bit 0(FFFFFFFF)  
Z 1 DS 002B 32bit 0(FFFFFFFF)  
S 0 FS 0053 32bit 7EFD0000(FFF)  
T 0 GS 002B 32bit 0(FFFFFFFF)  
D 0  
I 0

# Breakpoint software 2

Come si vede in figura al breakpoint il valore di ECX è 1DB10106



The screenshot shows a debugger window with the following components:

- Assembly View:** A list of instructions with their addresses. The instruction at address 004015AF is highlighted in red. The instruction is `AND ECX, 0FF`. The comment for this instruction is `kernel32.GetVersion`.
- Registers (FPU) View:** A list of registers and their values. The register ECX is highlighted in red, showing the value `1DB10106`.

Address	Disassembly	Comment
00401578	MOV EBP, ESP	
0040157A	PUSH -1	
0040157C	PUSH Malware_.004040C0	
00401581	PUSH Malware_.0040203C	
00401586	MOV EAX, DWORD PTR FS:[0]	
0040158C	PUSH EAX	
0040158D	MOV DWORD PTR FS:[0], ESP	
00401594	SUB ESP, 10	
00401597	PUSH EBX	
00401598	PUSH ESI	
00401599	PUSH EDI	
0040159A	MOV DWORD PTR SS:[EBP-18], ESP	
0040159D	CALL DWORD PTR DS:[C:\WINDOWS\system32\kernel32.GetVersion]	kernel32.GetVersion
004015A3	XOR EDX, EDX	
004015A5	MOV DL, AH	
004015A7	MOV DWORD PTR DS:[4052D4], EDX	
004015AD	MOV ECX, EAX	
004015AF	AND ECX, 0FF	kernel32.GetVersion
004015B5	MOV DWORD PTR DS:[4052D0], ECX	
004015BB	SHL ECX, 8	
004015BE	ADD ECX, EDX	
004015C0	MOV DWORD PTR DS:[4052D0], ECX	

Register	Value
EAX	1DB10106
ECX	1DB10106
EDX	00000001
EBX	7EFD0000
ESP	0018FF5C
EBP	0018FF80
ESI	00000000
EDI	00000000
EIP	004015AF Malware_.004015AF
C 0	ES 002B 32bit 0(FFFFFFFF)
P 1	CS 002B 32bit 0(FFFFFFFF)
A 0	SS 002B 32bit 0(FFFFFFFF)
Z 1	DS 002B 32bit 0(FFFFFFFF)
S 0	FS 0053 32bit 7EFD0000(FFF)
T 0	GS 002B 32bit 0(FFFFFFFF)
O 0	
0 0	LastErr ERROR_SUCCESS (00000000)
EFL	00000246 (NO, NB, E, BE, NS, PE, GE)

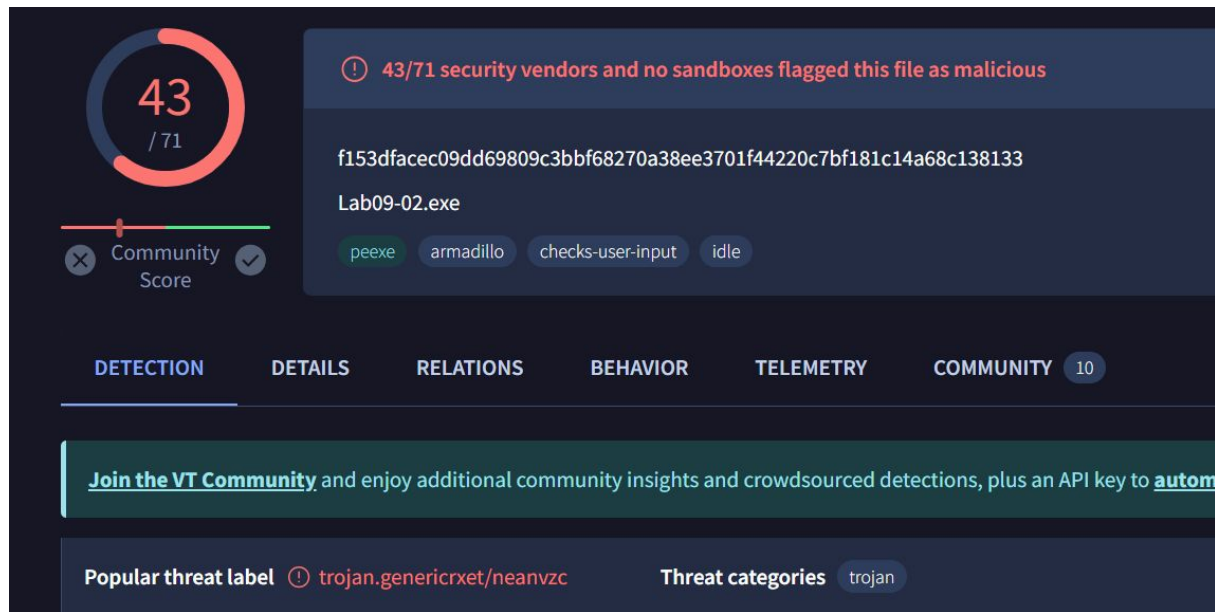
## Step-into 2

Dopo aver eseguito lo step-into il valore di ECX cambia e diventa 00000006, perché viene eseguito il comando AND ECX, 0FF che significa che il valore 0FF (in binario =0000 0000 0000 0000 1111 1111 ) viene aggiunto con l'AND logico al valore ECX (in binario 1DB10106 diventa 0001 1101 1011 0001 0000 0001 0000 0110 ) con il risultato finale di ECX = 00000006.

0040159D	. FF15 30404000	CALL DWORD PTR DS:[<&KERNEL32.GetVersion	kernel32.GetVersion	<div>Registers (FPU)</div> <div>EAX 10B10106</div> <div>ECX 00000006</div> <div>EDX 00000001</div> <div>EBX 7EFDE000</div> <div>ESP 0018FF5C</div> <div>EBP 0018FF80</div> <div>ESI 00000000</div> <div>EDI 00000000</div> <div>EIP 004015B5 Malware_.004015B5</div>
004015A3	. 33D2	XOR EDX,EDX		
004015A5	. 8AD4	MOV DL,AH		
004015A7	. 8915 D4524000	MOV DWORD PTR DS:[4052D4],EDX		
004015AD	. 8BC8	MOV ECX,EAX		
004015AF	. 81E1 FF000000	AND ECX,0FF		
004015B5	. 890D D0524000	MOV DWORD PTR DS:[4052D0],ECX		
004015BB	. C1E1 08	SHL ECX,8		
004015BE	. 03CA	ADD ECX,EDX		
004015C0	. 890D CC524000	MOV DWORD PTR DS:[4052CC],ECX		
004015C6	. C1E8 10	SHR EAX,10		



# Bonus



Dopo aver cercato l'hash del malware l'ho inserito sulla barra di ricerca di virus total e sembra essere un trojan



Grazie