

Malware analysis OllyDbg

Richiesta: 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:

- 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?
- 2. Inserite un breakpoint software all’indirizzo 004015A3. Qual è il valore del registro EDX?
- 3. Eseguite a questo punto uno «step-into». Indicate qual è ora il valore del registro EDX, motiva 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

BONUS: spiegare a grandi linee il funzionamento del malware

Valore del parametro CommandLine

Il valore del parametro CommandLine che viene passato sullo stack è “CMD”, rappresenta il command prompt di Windows, osservabile all’indirizzo 0040167.

00401067	68 30504000	PUSH Malware_.00405030	ASCII "cmd"
0040106C	6A 00	PUSH 0	
0040106E	FF15 04404000	CALL DWORD PTR DS:[<&KERNEL32.CreatePro	kernel32.CreateProcessA
00401074	8945 EC	MOV DWORD PTR SS:[EBP-14],EAX	
00401077	6A FF	PUSH -1	
00401079	8B4D F0	MOV ECX,DWORD PTR SS:[EBP-10]	
0040107C	51	PUSH ECX	

Breakpoint indirizzo 004015A3

Configuriamo il breakpoint all’indirizzo in questione, avviamo il programma e osserviamo il valore del registro EDX una volta fermatosi

00401599	• 57	PUSH EDI		EAX	1DB10106
0040159A	• 8965 E8	MOV DWORD PTR SS:[EBP-18],ESP		ECX	7EFDE000
0040159D	• FF15 30404000	CALL DWORD PTR DS:[<&KERNEL32.GetVersion	kernel32.GetVersion	EDX	00001DB1
004015A3	• 33D2	XOR EDX,EDX		EBX	7EFDE000
004015A5	• 8AD4	MOV DL,AH		ESP	0018FF5C
004015A7	• 8915 04524000	MOV DWORD PTR DS:[4052D4],EDX		EBP	0018FF88
004015AD	• 8BC8	MOV ECX,EAX		ESI	00000000
004015AF	• 81E1 FF000000	AND ECX,0FF		EDI	00000000
004015B5	• 8B4D F0	MOV ECX,DWORD PTR SS:[EBP-10]			

Il valore del registro EDX è 00001DB1

Step-into

Adesso eseguiamo lo step-into, che eseguirà l’istruzione XOR EDX, EDX, che serve ad azzerare il contenuto del registro EDX.

00401599	. 57	PUSH EDI		EAX 1DB10106
0040159A	. 8965 E8	MOV DWORD PTR SS:[EBP-18],ESP		ECX 7EFDE000
0040159D	. FF15 30404000	CALL DWORD PTR DS:[<&KERNEL32.GetVersion	kernel32.GetVersion	EDX 00000000
004015A3	. 33D2	XOR EDX,EDX		EBX 7EFDE000
004015A5	. 8AD4	MOV DL,AH		ESP 0018FF5C
004015A7	. 8915 D4524000	MOV DWORD PTR DS:[4052D4],EDX		EBP 0018FF88
004015AD	. 8BC8	MOV ECX,EAX		ESI 00000000
004015AF	. 81E1 FF000000	AND ECX,0FF		EDI 00000000
004015B5	. 8900 D0524000	MOV DWORD PTR DS:[4052D0],ECX		

Il valore del registro EDX è 00000000

Breakpoint indirizzo 004015AF

Configuriamo un secondo breakpoint all’indirizzo selezionato, il valore del registro ECX è 1DB1016

004015A0	. 8BC8	MOV ECX,EAX		EAX 1DB10106
004015AF	. 81E1 FF000000	AND ECX,0FF		ECX 1DB10106
004015B5	. 8900 D0524000	MOV DWORD PTR DS:[4052D0],ECX		EDX 00000001
004015B8	. C1E1 08	SHL ECX,8		EBX 7EFDE000
004015BE	. 03CA	ADD ECX,EDX		ESP 0018FF5C
004015C0	. 8900 CC524000	MOV DWORD PTR DS:[4052CC],ECX		EBP 0018FF88
004015C6	. C1E8 10	SHR EAX,10		ESI 00000000
004015C9	. A3 C8524000	MOV DWORD PTR DS:[4052C8],EAX		EDI 00000000
004015CF	. 6A 00	PUSH 0		

Adesso eseguiamo lo step-into, notiamo che il valore del registro ECX è 00000006 dopo aver eseguito l’istruzione AND ECX, 0FF

004015A7	. 8915 D4524000	MOV DWORD PTR DS:[4052D4],EDX		Registers (FPU)
004015AD	. 8BC8	MOV ECX,EAX		EAX 1DB10106
004015AF	. 81E1 FF000000	AND ECX,0FF		ECX 00000006
004015B5	. 8900 D0524000	MOV DWORD PTR DS:[4052D0],ECX		EDX 00000001
004015B8	. C1E1 08	SHL ECX,8		EBX 7EFDE000
004015BE	. 03CA	ADD ECX,EDX		ESP 0018FF5C
004015C0	. 8900 CC524000	MOV DWORD PTR DS:[4052CC],ECX		EBP 0018FF88
004015C6	. C1E8 10	SHR EAX,10		ESI 00000000
004015C9	. A3 C8524000	MOV DWORD PTR DS:[4052C8],EAX		EDI 00000000
004015CF	. 6A 00	PUSH 0		

Qui l’istruzione esegue l’AND sui bit di ECX e del valore di FF (esadecimale), dando come risultato il valore di ECX ottenuto dopo lo step-into.