

OllyDBG

Traccia

Facendo riferimento al malware: Malware_U3_W3_L3 rispondere 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
4. Motivate la risposta.
5. Che istruzione è stata eseguita?
6. Inserite un secondo breakpoint all'indirizzo di memoria 004015AF. Qual è il valore del registro ECX?
7. Eseguite un step-into. Qual è ora il valore di ECX?
8. Spiegate quale istruzione è stata eseguita.
9. BONUS: spiegare a grandi linee il funzionamento del malware.

Traccia 1

Il valore passato è "cmd" ovvero il comando prompt di Windows.

| | | | |
|----------|---------------|--|-------------------------|
| 00401056 | 52 | PUSH EDX | pProcessInfo |
| 00401057 | 8D45 A8 | LEA EAX,DWORD PTR SS:[EBP-58] | pStartupInfo |
| 0040105A | 50 | PUSH EAX | CurrentDir = NULL |
| 0040105B | 6A 00 | PUSH 0 | pEnvironment = NULL |
| 0040105D | 6A 00 | PUSH 0 | CreationFlags = 0 |
| 0040105F | 6A 00 | PUSH 0 | InheritHandles = TRUE |
| 00401061 | 6A 01 | PUSH 1 | pThreadSecurity = NULL |
| 00401063 | 6A 00 | PUSH 0 | pProcessSecurity = NULL |
| 00401065 | 6A 00 | PUSH 0 | CommandLine = "cmd" |
| 00401067 | 68 30504000 | PUSH Malware_.00405030 | ModuleFileName = NULL |
| 00401069 | 6A 00 | PUSH 0 | |
| 0040106E | FF15 04404000 | CALL DWORD PTR DS:[<&KERNEL32.CreateProcessA>] | CreateProcessA |
| 00401070 | 8B45 EC | MOV DWORD PTR SS:[EBP-14],EDX | |

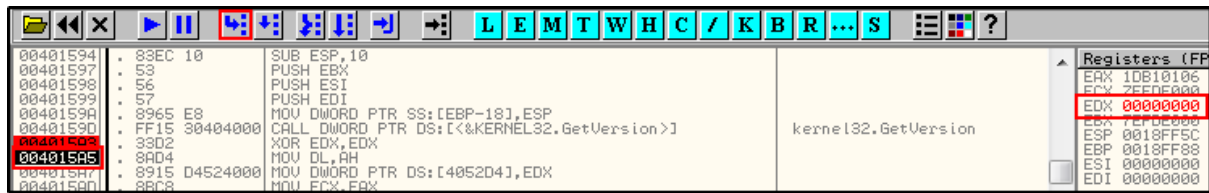
Traccia 2

Inseriamo il breakpoint nell'indirizzo richiesto e premiamo play, quindi vediamo che il valore di EDX è 00001DB1.

| | | | | |
|----------|------------------|--|-------------------------|--------------------------------|
| 0040157C | 68 C0404000 | PUSH Malware_.004040C0 | SE handler installation | EAX 1DB10106 |
| 00401581 | 68 3C204000 | PUSH Malware_.0040203C | | ECX 7EFD0000 |
| 00401586 | 64:A1 00000000 | MOV EAX,DWORD PTR FS:[0] | | EDX 00001DB1 |
| 0040158C | 50 | PUSH EAX | | EBX 7EFD0000 |
| 0040158D | 64:8925 00000000 | MOV DWORD PTR FS:[0],ESP | | ESP 0018FFC0 |
| 00401594 | 83EC 10 | SUB ESP,10 | | EBP 0018FF88 |
| 00401597 | 53 | PUSH EBX | | ESI 00000000 |
| 00401598 | 56 | PUSH ESI | | EDI 00000000 |
| 00401599 | 57 | PUSH EDI | | EIP 004015A3 Malware_.004015A3 |
| 0040159A | 8965 E8 | MOV DWORD PTR SS:[EBP-18],ESP | | C 0 ES 002B 32bit 0(FFFFFFFF) |
| 0040159D | FF15 30404000 | CALL DWORD PTR DS:[<&KERNEL32.GetVersion>] | kernel32.GetVersion | P 1 CS 0023 32bit 0(FFFFFFFF) |
| 004015A3 | 33D2 | XOR EDX,EDX | | Q 0 SS 002B 32bit 0(FFFFFFFF) |
| 004015A5 | 8AD4 | MOV DL,AH | | Z 0 DS 002B 32bit 0(FFFFFFFF) |
| 004015A7 | 8915 04524000 | MOV DWORD PTR DS:[405204],EDX | | |

Traccia 3

Cliccando su Step Into il programma entra nella funzione e il valore di EDX cambia in 00000000.

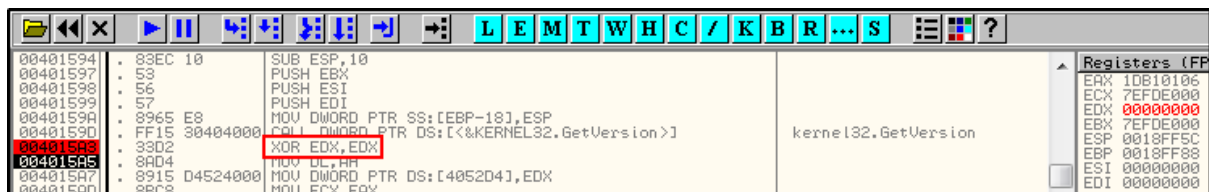


Traccia 4

Ciò avviene perchè nell'operazione XOR (exclusive OR), l'output è 1 se i bit corrispondenti degli operandi sono diversi, e 0 se sono uguali. In questo caso viene fatto XOR paragonando EDX a se stesso quindi il risultato è 0 per ciascun bit.

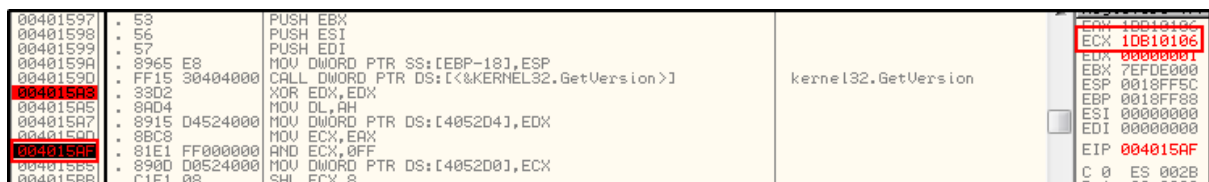
Traccia 5

XOR EDX,EDX



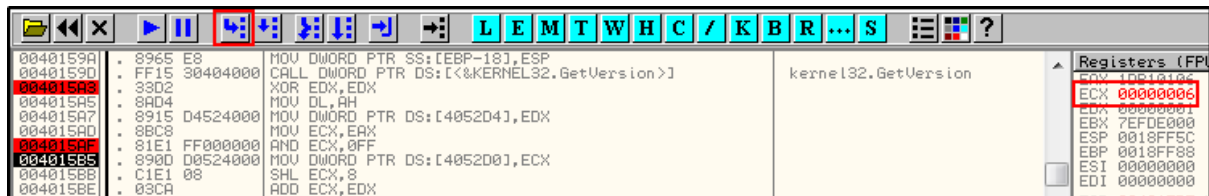
Traccia 6

Inserendo il breakpoint e premendo su play il valore di ECX diventa 1DB10106



Traccia 7

Eseguendo uno step into il valore di ECX diventa 00000006.



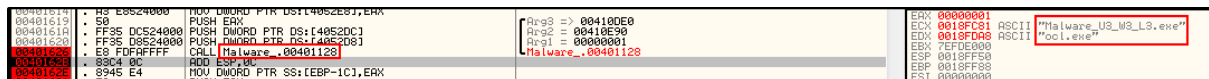
Traccia 8

E' stata eseguita AND ECX, OFF dove avviene un AND fra i bit di ECX e OFF dove viene restituito 1 solo se entrambi i bit sono uguali a 1.

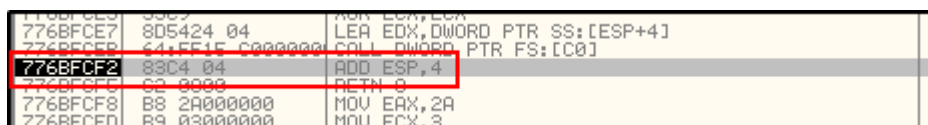
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----------|
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | = | 1DB10106 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | = | 0FF |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | = | 00000006 |

Traccia 9

Usando diversi breakpoint ci accorgiamo che la funzione Malware_.00401128 dopo essere eseguita aggiunge come valori il nome del file e ocl.exe rispettivamente a ECX e EDX.



Proseguendo senza interruzioni i due valori scompaiono e il programma ferma la sua esecuzione al seguente indirizzo.



Questo fa pensare che qualcosa impedisce la corretta esecuzione del malware.