COMPITO L11 S3malware

**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



1)

```
. 52              PUSH EDX                                 pProcessInfo
. 8D45 A8         LEA EAX,DWORD PTR SS:[EBP-58]
. 50              PUSH EAX                                 pStartupInfo
. 6A 00           PUSH 0                                   CurrentDir = NULL
. 6A 00           PUSH 0                                   pEnvironment = NULL
. 6A 00           PUSH 0                                   CreationFlags = 0
. 6A 01           PUSH 1                                   InheritHandles = TRUE
. 6A 00           PUSH 0                                   pThreadSecurity = NULL
. 6A 00           PUSH 0                                   pProcessSecurity = NULL
. 68 30504000     PUSH Malware_.00405030                   CommandLine = "cmd"
. 6A 00           PUSH 0                                   ModuleFileName = NULL
. FF15 04404000   CALL DWORD PTR DS:[<&KERNEL32.CreateProcessA>]  CreateProcessA
8945 EC           MOV DWORD PTR SS:[EBP-14],EAX
```

2)

```
004015A3  . 33D2          XOR EDX,EDX
```

```
Registers (FPU)
EAX 0A280105
ECX 7FFD4000
EDX 00000A28
EBX 7FFD4000
ESP 0012FF94
EBP 0012FFC0
ESI FFFFFFFF
EDI 7C910208 ntdll.7C910208
EIP 004015A3 Malware_.004015A3
```

3)

```
Registers (FPU)
EAX 0A280105
ECX 7FFD4000
EDX 00000000
EBX 7FFD4000
ESP 0012FF94
EBP 0012FFC0
ESI FFFFFFFF
EDI 7C910208 ntdll.7C910208
EIP 004015A5 Malware_.004015A5
```

4/5) XOR fa resettare il valore delle variabili essendo la stessa, XOR = NOT OR, quindi EDX essendo uguale EDX, ritorna 0.

6)

```
004015AF   . 81E1 FF000000      AND ECX,0FF
```

```
EAX 0A280105
ECX 0A280105
EDX 00000001
EBX 7FFDF000
ESP 0012FF94
EBP 0012FFC0
ESI FFFFFFFF
EDI 7C910208 ntdll.7C910208
EIP 004015AF Malware_.004015AF
```

7)

```
Registers (FPU)
EAX 0A280105
ECX 00000005
EDX 00000001
EBX 7FFDF000
ESP 0012FF94
EBP 0012FFC0
ESI FFFFFFFF
EDI 7C910208 ntdll.7C910208
EIP 004015B5 Malware_.004015B5
```

8) AND fa passare entrambi i valori se son veri, compiono con la funzione e riporta 1.