# CONSEGNA SETTIMANA 11 LEZIONE 3

## Malware analysis – OllyDBG

*Ci viene richiesto di rispondere ai seguenti quesiti utilizzando OllyDBG*

- **PUNTO 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?

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

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

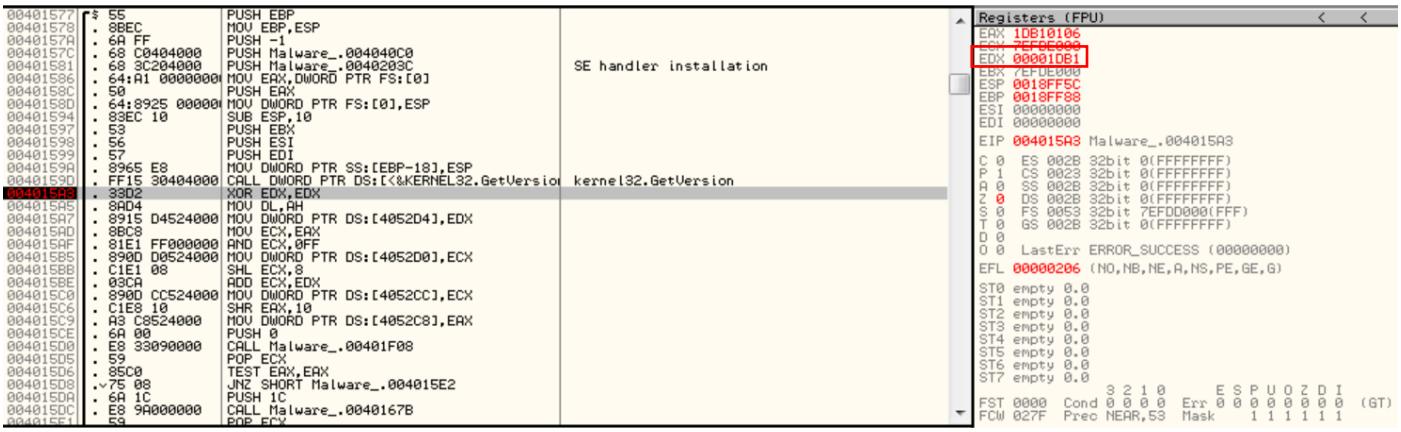
- **BONUS:** spiegare a grandi linee il funzionamento del malware.

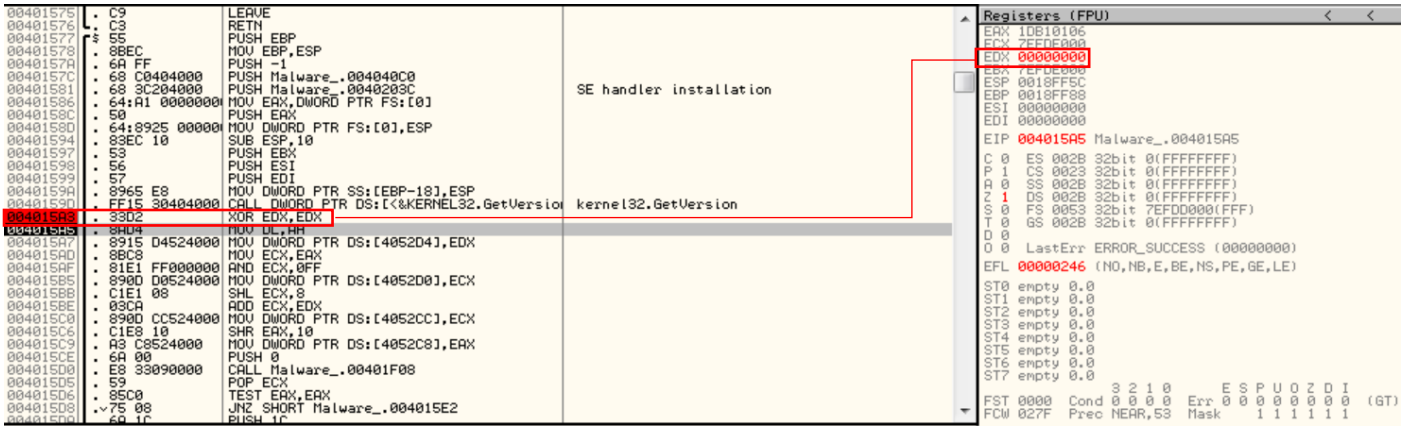
**PUNTO 1.**

Tramite il software **OllyDBG** ci è possibile analizzare dinamicamente il malware, soffermandoci sulle fasi dell'esecuzione che più ci interessano. Prendiamo in esame la chiamata alla funzione ***CreateProcess*** all'indirizzo 0040106E. Il valore del parametro ***CommandLine*** passato alla funzione in questo caso risulta essere "**cmd**".


**PUNTO 2.**

Dopo aver inserito un breakpoint all'indirizzo **004015A3** come da istruzioni, possiamo notare il valore di **EDX**, ovvero **00001DB1**. Eseguendo poi il comando *step-into*, possiamo notare che il valore diventa 0. Questo avviene a causa dell'istruzione presente proprio all'indirizzo **0004015A3**, cioè **XOR EDX,EDX**. Lo XOR è una funzione logica che risulta in valore uguale a 1 se i parametri a confronto sono diversi tra loro, ma essendo EDX sempre uguale a EDX il risultato sarà 0.






**PUNTO 3.**

Analogamente, inserendo un breakpoint all'indirizzo **004015AF**, notiamo che il valore di **ECX** sia **1DB10106**. Eseguendo uno step-into possiamo constatare che il valore cambia e diventa **00000006**. Questo avviene a causa proprio dell'operazione all'indirizzo 004015AF, ovvero **AND EXC, 0FF**. L'operatore logico AND in questo caso confronta tra di loro i singoli bit dei due valori, ovvero ECX e 0FF. Per confermare ciò convertiamo prima i valori di ECX e il valore 0FF in binario.

ECX 1DB10106 (exa) -> 0001 1101 1011 0001 0000 0001 0000 0110

0FF (exa) -> 0000 0000 0000 0000 0000 0000 1111 1111

A questo punto poniamo l'AND logico bit per bit e otteniamo il valore 0000 0000 0000 0000 0000 0000 0000 0110, ovvero 00000006.