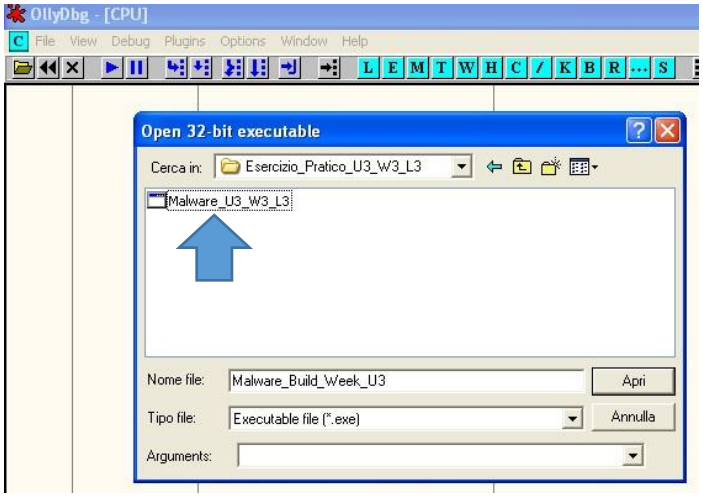
Gabriele Di giampietro **Esercitazione 18/01/2023**

Obbiettivo: Con riferimento al Malware_U3_W3_L3, rispondete ai seguenti quesiti utilizzando il tool Debuger 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

Prima Parte

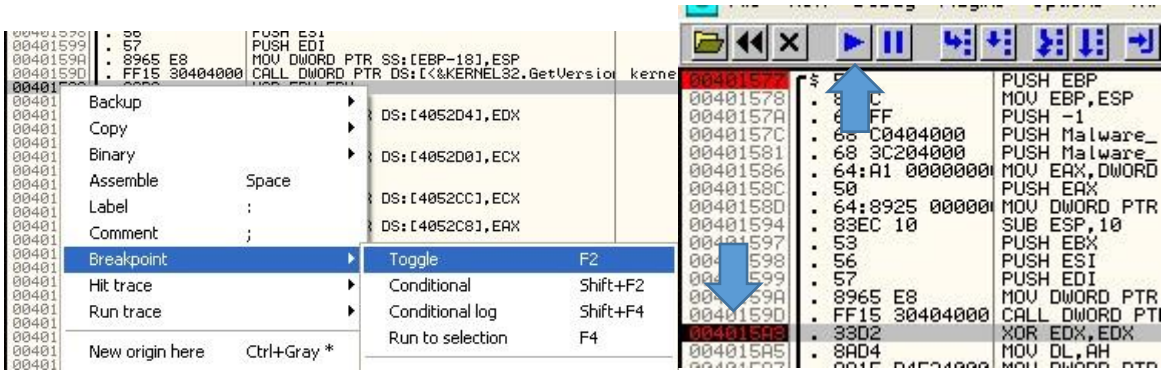
Per prima cosa andiamo a caricare il nostro Malware sul tool OllyDBG, che ci mostrerà il codice e potremo vedere attraverso i suoi comandi come il programma si sviluppa e cosa fa in alcune parte del codice che analizzeremo utilizzando i **breakpoint** che vedremo fra poco.



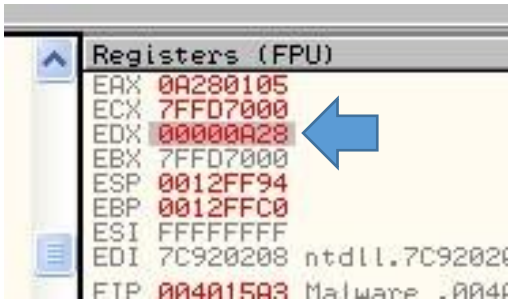
Dopo di che come prima analisi andiamo ad indentificare quale è il valore del parametro "CommandLine" che effettua una chiamata di funzione **"CreatePorecessA"** all'indirizzo di memoria 0040106E, che possiamo vedere essere **"cdm"** inserito nello stak dall'istruzione Push nella parte di codice sopra.

Seconda Parte

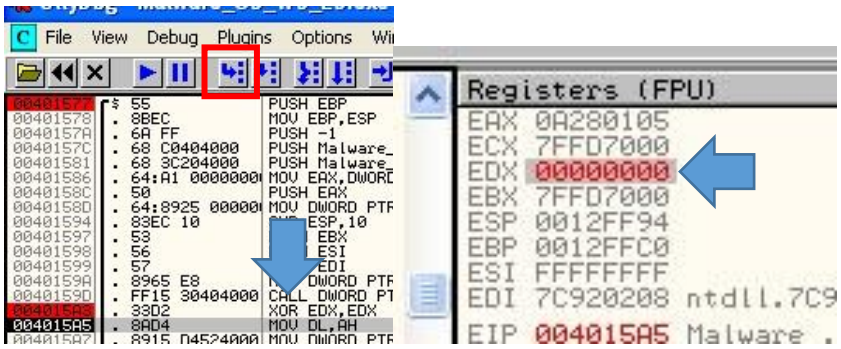
Nella seconda parte dell'esercitazione ci spostiamo nell'allocazione di memoria 004015A3, qui andiamo ad effettuare un breakpoint, che possiamo vedere evidenziato in nero con scritte rosse nell'immagine seguente, e cioè fermiamo il programma su una data istruzione eseguita.



Poi andiamo a vedere quale è il valore del registro EDX, lanciamo il programma cliccando sul tasto play sulla barra degli strumenti e possiamo constatare dalla finestra Registers FPU che il valore nel registro EDX è 00000A28 (esadecimale) o 2600 in decimale.

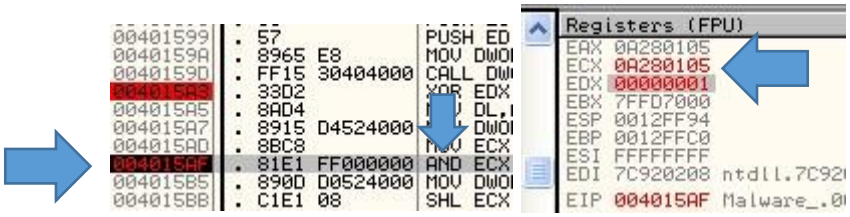


Utilizziamo la funzione **Step-into** nella barra degli strumenti, evidenziato in rosso nell'immagine seguente, che ci permette di entrare nel codice della funzione e vediamo che il valore del registro EDX cambia in 00000000 (esadecimale) o 0 in decimale. Questo perché come possiamo vedere dal codice, in questa parte il programma utilizza un operatore logico XOR che quando ha due valori uguali dà come risultato sempre 0.

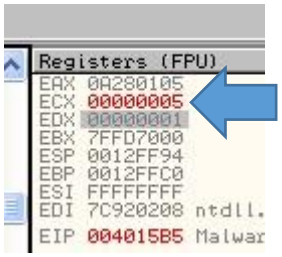


Terza Parte

Nel terzo punto andiamo ad aggiungere al breakpoint che abbiamo posizionato nell'allocazione di memoria 004015A3 nella seconda parte dell'esercitazione, un secondo breakpoint all'indirizzo 004015AF e dopo aver lanciato il programma sempre attraverso il tasto play andiamo a vedere in questo caso il valore nel registro ECX, che risulta essere 0A280105 (esadecimale) o 170393861in decimale.



Utilizziamo la funzione Step-into e vediamo che il valore di ECX è diventato 00000005 (esadecimale) o 5 in decimale. In questo caso abbiamo un operato logico AND il quale ricevendo in ingresso almeno due valori restituisce 1 solo se tutti i valori di ingresso hanno valore 1.



Bonus

Dopo avere analizzato il codice del Malware abbiamo provato ad ipotizzare quale sia la funzione, in quanto come vediamo importa librerie KERNEL32.dll e WS2_32.dll. Le quali vanno ad importare funzioni che accedono al file system e funzioni che utilizzano protocolli http e tcp, quindi possiamo dedurre che fornisca una connessione ad un controllo remoto sulla macchina attaccata creando un soket, quindi potremmo avere una reversa shell.