Primi passi

Bene, dopo questa pallossisima introduzione, sarete (spero) impazienti di iniziare a fare qualcosa, ma ovviamente non posso farvi qui su due piedi un corso completo di reversing, non tanto per il tempo e lo spazio (servirebbe un intero sito solo per quello!), quanto per il fatto che non potrebbe esistere un "corso", ma soltanto un po' di teoria e molti suggerimenti pratici, perché la maggior parte delle cose si apprendono molto meglio sperimentando e facendo esperienza sul campo.

Quindi, se non ce l'avete già, il mio consiglio è di iniziare innanzitutto a farsi un idea di come un calcolatore moderno funziona al suo interno a livello software (sistema operativo, gestione dei processi, memoria, files, cartelle, ecc...) e per questo va bene un qualsiasi testo introduttivo sull'argomento; ce ne sono talmente tanti che non saprei nemmeno indicarvi quale e fate sicuramente prima a cercare direttamente su Google, oppure smanettate notte e giorno con il PC che è sempre il modo migliore per capirci qualcosa!

Passata la prima fase, se non li sapete già, secondo me è bene che cominciate a mettere le mani anche sui primi rudimenti di programmazione; magari, quando reversate, questo alla fine non vi servirà nemmeno, ma vi assicuro che conoscendo le strutture ad alto livello dei programmi, sarà più facile capire cosa sta succedendo al piano di sotto...

Se ne avete la possibilità, risfoderate tutti i vecchi libri su C, C++, Pascal, COBOL, ecc... , quello che vi pare; l'importante è fare mente locale sulle concetti logici fondamentali alla base della programmazione, a prescindere dal linguaggio con cui avete a che fare (tra parentesi le istruzioni tipiche usate per ciascun elemento):

-

Controllo del Flusso

- Cicli (FOR e WHILE)

- Decisioni (IF)

- Scelte multiple (CASE)

- Rientro dalle funzioni (RETURN)

-

Funzioni e Procedure

-

Parametri in ingresso ed in uscita

-

Chiamate ricorsive

-

Passaggio per valore o per riferimento

-

Variabili

-

Tipi di dati e dimensioni in memoria

-

Allocazione e deallocazione

-

Visibilità locale e globale

- Compilatori e Sistema Operativo

- Files sorgente e file header

- Modalità di creazione di un eseguibile (compilazione + linking)

- Direttive ed opzioni di compilazione

Questo elenco potrebbe essere (molto!) più lungo, ma di sicuro questo non è il momento adatto per scrivere un corso di programmazione, quindi vi ho indicato soltanto i punti più importanti da tenere presente quando state cercando di ricostruire il comportamento interno di un software; anche se improntato al C++, il famoso "Thinking in C++" di Bruce Eckel rimane tuttora un'ottimo libro sull'argomento che vi consiglio di leggere se avete tempo, dato che la maggior parte dei programmi che troverete in giro sarà scritta in C++.

Una volta acquisita una certa padronanza con la programmazione (per il reversing tenete conto che la cosa più importante è capire cosa fa un certo programma a partire dal suo codice macchina), potete cominciare a studiare proprio la base di tutto, le solide fondamenta sulle quali ogni programma si basa per funzionare: l'Assembler e l'architettura x86

Prima tappa, manco a dirlo, sempre sul sito della UIC: nella sezione Assembly potrete trovare un sacco di tutorial in italiano sulla programmazione a basso livello; vi consiglio caldamente di fare anche un salto sul sito di Giobe, che contiene un ottimo corso sull'Assembly, con tanti esempi, esercizi e reference di tutti i comandi. Degni di nota sono anche il sito di Iczelion, ricco di tutorials per imparare a programmare "duro" ed ovviamente il bel sito di Randall Hyde, un'eminenza in questo campo (è di sua invenzione la sintassi HLA), mentre per un buon forum sul Win32Assembler, andate qui.

Mi raccomando, prima di andare avanti, è molto importante che siano chiari almeno i seguenti concetti chiave:

-

Numerazione binaria ed esadecimale e relative operazioni (AND, OR, XOR, ecc...)

Registri del processore (EAX, EDX, ECX, EBX, ESI, ecc...)

Segmenti di memoria (Data, Stack, ecc...)

Struttura di base del Portable Executable Format (PE)

Puntatori con riferimenti assoluti o relativi

Gestione dello Stack e SEH

Istruzioni fondamentali assembly (MOV, ADD, PUSH, POP, CALL, JMP, JNZ, ecc...)

Se un giorno vi troverete improvvisamente a parlare in esadecimale con il vostro mouse oppure riuscite a vedere strani simboli verdi che scorrono sui muri, allora o siete completamente impazziti oppure state diventando un vero reverser!

E' il momento di cominciare ad aprire queste maledette scatolette e non c'è posto migliore della UIC per esercitarsi: iniziate dai Tutorials per Newbies per poi passare alle Lezioni vere e proprie, anch'esse divisi in Newbies e Studenti... dateci sotto!

Quando avete fatto tutte le lezioni o comunque vi sentite ad un buon livello, seguite i numerosi tutorials che trovate nello Store, così vedrete come quello che avete imparato finora può essere applicato a target reali; se a questo punto non riuscite proprio a placare la vostra sete di reversing, vi consiglio di farvi un giro su Crackmes, ottimo come palestra gratis!

Per quanto riguarda il mio contributo alla vostra "formazione", vi posso indicare alcuni tra i miei Strumenti software preferiti che possono arricchire la vostra cassetta degli attrezzi, e non meno importanti alcuni Tutorials in italiano che ho scritto per il sito della UIC.