

# Relazione S11L2

Lo scopo dell'esercizio è prendere dimestichezza con IDA pro. Questo software è un disassembler utile per l'analisi statica avanzata, in quanto ci facilita il compito di leggere il codice assembly.

Quello che dobbiamo analizzare è il Malware\_U3\_W3\_L2 e rispondere ai seguenti quesiti:

## 1. Individuare l'indirizzo della funzione DLLMain

```
.text:1000D02E ; BOOL __stdcall DllMain(HINSTANCE hinstDLL,DWORD fdwReason,LPUUID lpvReserved)
.text:1000D02E _DllMain@12      proc near                                ; CODE XREF: DllEntryPoint+4B↓p
```

Come possiamo vedere, l'indirizzo è 1000D02E, infatti vediamo la funzione dichiarata con i suoi parametri.

## 2. Dalla scheda «imports» individuare la funzione «gethostbyname». Qual è l'indirizzo dell'import?



Individuata la funzione possiamo vedere che l'indirizzo dell'import è 100163CC e se utilizziamo il jump dando in input l'indirizzo di memoria vediamo che la funzione viene chiamata.

```
* .idata:100163CC ; struct hostent *__stdcall gethostbyname(const char *name)
.idata:100163CC      extrn gethostbyname:dword
```

## 3. Quante sono le variabili locali della funzione alla locazione di memoria 0x10001656?

```
.text:10001656 ; DWORD __stdcall sub_10001656(LPUUID)
.text:10001656 sub_10001656      proc near                                ; DATA XREF: DllMain(x,x,x)+C8↓o
.text:10001656
.text:10001656 var_675          = byte ptr -675h
.text:10001656 var_674          = dword ptr -674h
.text:10001656 hModule          = dword ptr -670h
.text:10001656 timeout          = timeval ptr -66Ch
.text:10001656 name            = sockaddr ptr -664h
.text:10001656 var_654          = word ptr -654h
.text:10001656 in              = in_addr ptr -650h
.text:10001656 Parameter        = byte ptr -644h
.text:10001656 CommandLine      = byte ptr -63Fh
.text:10001656 Data            = byte ptr -638h
.text:10001656 var_544          = dword ptr -544h
.text:10001656 var_50C          = dword ptr -50Ch
.text:10001656 var_500          = dword ptr -500h
.text:10001656 var_4FC          = dword ptr -4FCh
.text:10001656 readfds          = fd_set ptr -4BCh
.text:10001656 phkResult        = HKEY__ ptr -3B8h
.text:10001656 var_3B0          = dword ptr -3B0h
.text:10001656 var_1A4          = dword ptr -1A4h
.text:10001656 var_194          = dword ptr -194h
.text:10001656 WSADATA          = WSADATA ptr -190h
.text:10001656 arg_0            = dword ptr 4
```

All'interno della subroutine ci sono diverse variabili, per l'esattezza 20. Come abbiamo visto a lezione se l'offset è negativo allora si tratta di variabili, se è positivo invece si parla di parametri. Quindi

l'ultimo valore, avendo un offset positivo sarà un parametro.

**4. Quanti sono, invece, i parametri della funzione sopra?**

L'unico parametro è l'ultimo `arg_0`, poiché l'offset è positivo.

Davide Lecci