CONSEGNA SETTIMANA 11 LEZIONE 2

Analisi statica avanzata con IDA

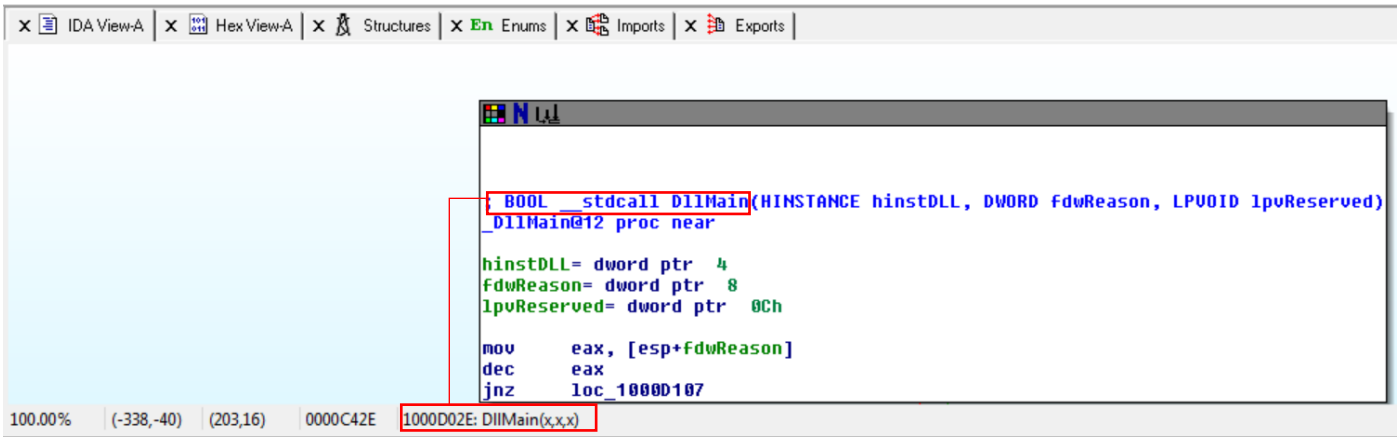
Lo scopo dell'esercizio di oggi è di acquisire esperienza con IDA, un tool fondamentale per l'analisi statica.

Ci viene chiesto di rispondere ai seguenti quesiti, utilizzando IDA Pro.

- **1.** Individuare l'indirizzodella funzione DLLMain (così com'è, in esadecimale)
- **2.** Dalla scheda «imports» individuare la funzione «gethostbyname». Qual è **l'indirizzo** dell'import? Cosa fa la funzione?
- **3.** Quante sono le **variabili locali** della funzione alla locazione di memoria 0x10001656?
- **4.** Quanti sono, invece, i **parametri** della funzione sopra?
- **5.** Inserire altre considerazioni macro livello sul malware (comportamento)

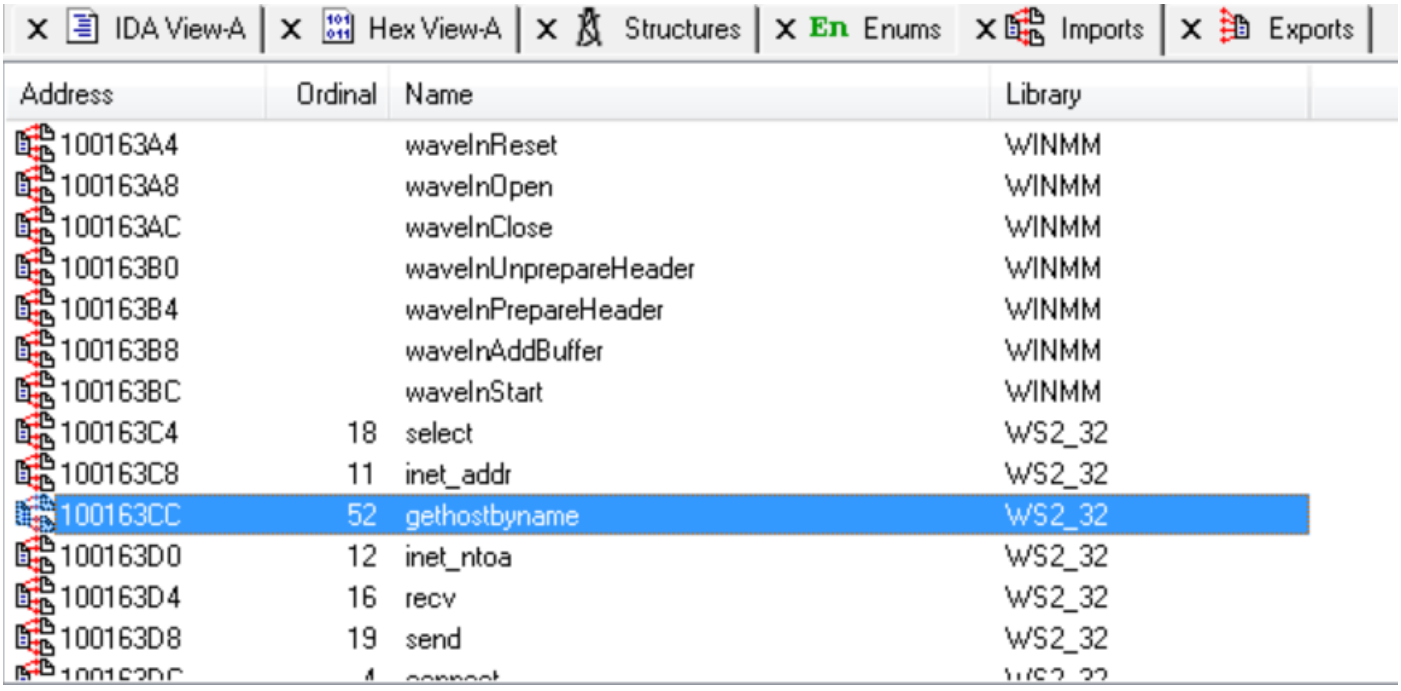
PUNTO 1.

Utilizzando il software IDA pro possiamo analizzare il malware staticamente, senza avviarlo. Tramite il grafico mostrato qui sotto possiamo notare che l'indirizzo della funzione DDLMain risulta 1000D02E.



PUNTO 2.

Dirigendoci sulla finestra "imports" e facendo una ricerca per nome possiamo individuare la funzione "gethostbyname" il cui indirizzo risulta 1000163CC.



Address	Ordinal	Name	Library
100163A4		waveInReset	WINMM
100163A8		waveInOpen	WINMM
100163AC		waveInClose	WINMM
100163B0		waveInUnprepareHeader	WINMM
100163B4		waveInPrepareHeader	WINMM
100163B8		waveInAddBuffer	WINMM
100163BC		waveInStart	WINMM
100163C4	18	select	WS2_32
100163C8	11	inet_addr	WS2_32
100163CC	52	gethostbyname	WS2_32
100163D0	12	inet_ntoa	WS2_32
100163D4	16	recv	WS2_32
100163D8	19	send	WS2_32

```
.idata:100163CC ; struct hostent *__stdcall gethostbyname(const char *name)
.idata:100163CC                 extrn gethostbyname:dword
.idata:100163CC                                         ; CODE XREF: sub_10001074:loc_100011AF↑p
.idata:100163CC                                         ; sub_10001074+1D3↑p ...
```

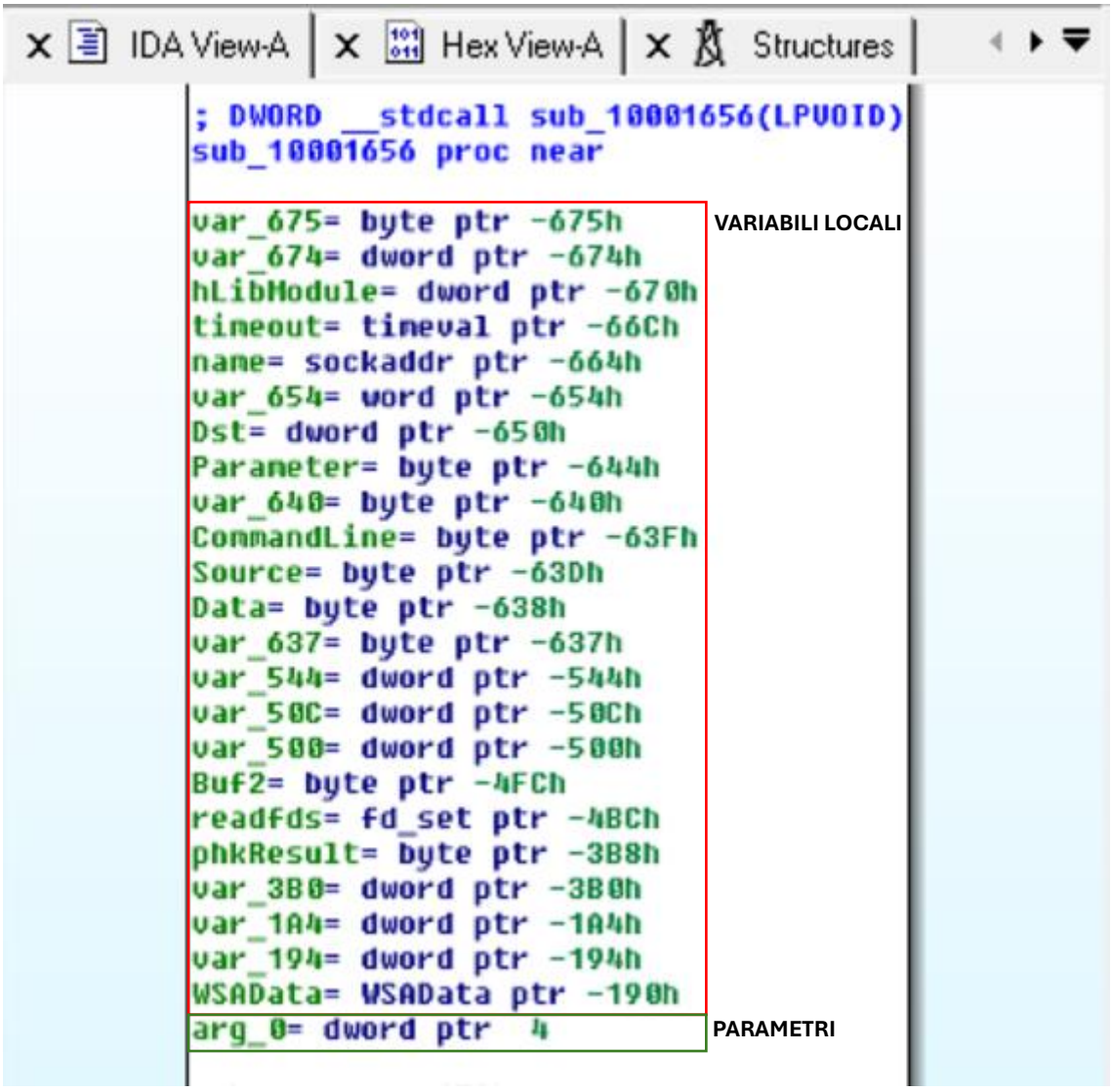
Ci è poi possibile aprire la suddetta funzione per vederne il contenuto. Facendo riferimento alle guide di microsoft learn si scopre che questa funzione ha lo scopo di recuperare informazioni riguardanti il nome dell'host.

PUNTO 3.

Facendo una ricerca nella finestra degli indirizzi possiamo risalire alla funzione correlata all'indirizzo 0x10001656. Prendendo come riferimento l'immagine sotto possiamo notare la presenza di 23 variabili locali.

PUNTO 4.

Allo stesso modo possiamo evidenziare il numero di parametri della funzione, che in questo caso risulta essere 1.



La differenza tra variabili locali e parametri, nel caso del codice assembly, si trova **nell'offset**, ovvero la distanza dal registro di partenza. Nel caso di variabili locali l'offset sarà **negativo**, al contrario dei parametri che avranno **offset positivo.**