S11 L2

GiuliaSalani

Report Esercizio Malware Analysis

INDICE

TRACCIA	2
SVOLGIMENTO	
DEFINIZIONE: IDA PRO	
PREPARAZIONE	
QUESITO 1	5
QUESITO 2	6
QUESITO 3	7
QUESITO 4	9

TRACCIA

Lo scopo dell'esercizio di oggi è di acquisire esperienza con IDA, un tool fondamentale per l'analisi statica. A tal proposito, con riferimento al malware chiamato «**Malware_U3_W3_L2**» presente all'interno della cartella «Esercizio_Pratico_U3_W3_L2» sul desktop della macchina virtuale dedicata all'analisi dei malware, rispondere ai seguenti quesiti, utilizzando IDA Pro.

- 1. Individuare l'indirizzo della funzione DLLMain
- 2. Dalla scheda «imports» individuare la funzione «gethostbyname». Qual è l'indirizzo dell'import?
- 3. Quante sono le variabili locali della funzione alla locazione di memoria 0x10001656?
- 4. Quanti sono, invece, i parametri della funzione sopra?

SVOLGIMENTO

DEFINIZIONE: IDA PRO

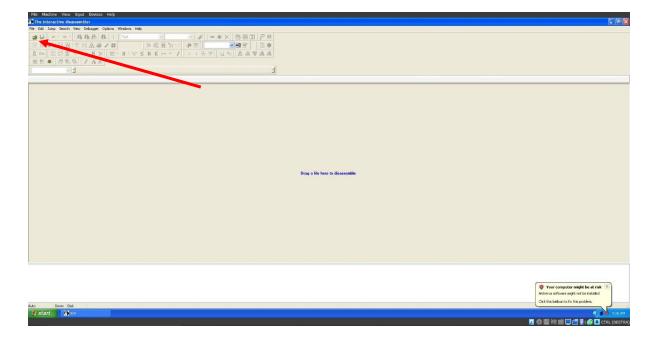
IDA Pro è uno **strumento di analisi malware e reverse engineering** che consente agli esperti di esaminare il codice binario di programmi eseguibili. Fornisce una visualizzazione dettagliata del codice assembly, facilitando l'analisi delle istruzioni e la comprensione del flusso di controllo. La **creazione automatica di grafici del flusso di controllo e la capacità di identificare automaticamente le funzioni** semplificano l'analisi. IDA Pro supporta numerosi tipi di file eseguibili e offre funzionalità avanzate come decompilazione, debugging remoto e scripting per l'automazione delle attività. È uno strumento essenziale per gli analisti di malware e gli esperti di sicurezza per comprendere il comportamento interno di software dannoso.

PREPARAZIONE

Come prima cosa, prepariamo il software e carichiamo il malware oggetto dell'analisi odierna. Apriamo la macchina virtuale e individuiamo IDA pro sul desktop:



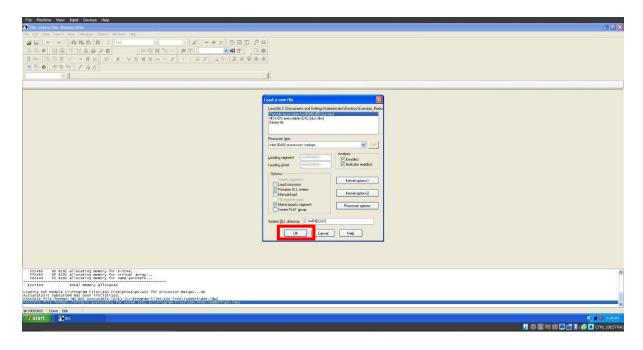
Clicchiamo l'icona a forma di cartella in alto a sinistra:



Selezioniamo il file da analizzare e clicchiamo su open:



Confermiamo a IDA che desideriamo aprire il formato PE:

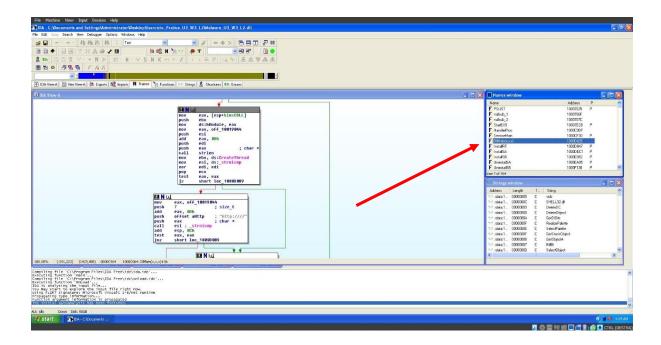


Ora possiamo affrontare le consegne.

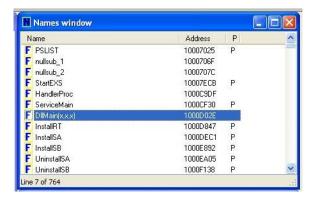
QUESITO 1

Individuare l'indirizzo della funzione DLLMain

Al caricamento del malware sul software, notiamo subito sulla destra la finestra Names Windows. A metà elenco individuiamo DLLMain e di fianco il suo indirizzo:



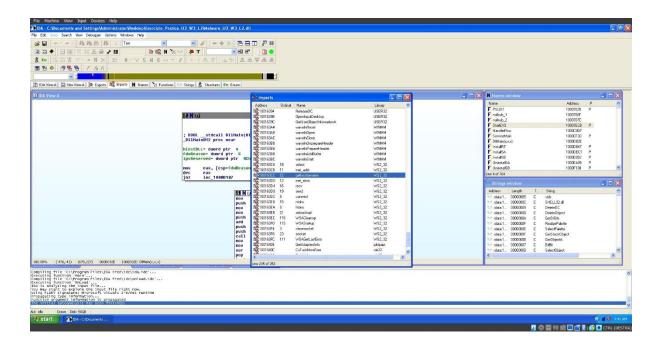
L'indirizzo è 1000D02E:



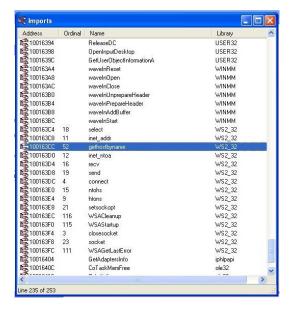
QUESITO 2

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

Selezioniamo la scheda "imports", che risulta già aperta. Scorriamo la lista fino ad individuare il nome gethostbyname:



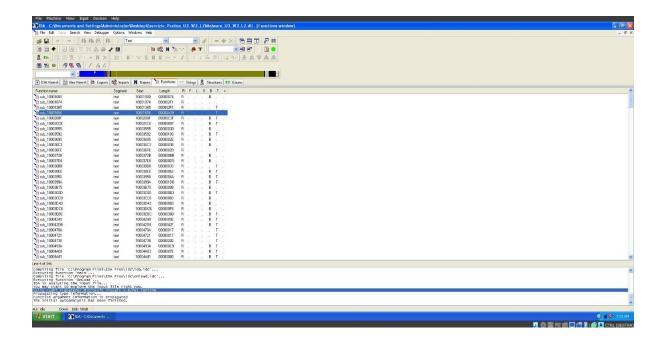
Così possiamo individuare l'indirizzo che è 100163CC:



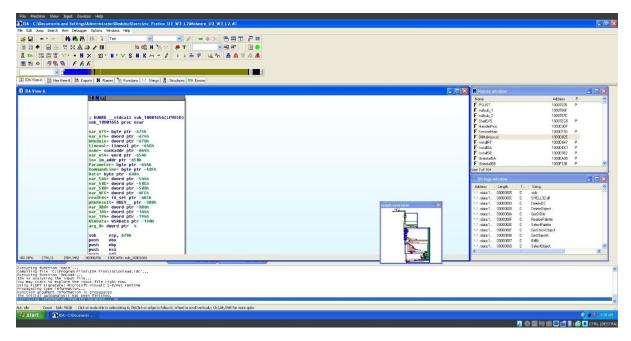
QUESITO 3

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

Apriamo la scheda "Functions" e cerchiamo l'indirizzo 0x10001656:



Con un doppio clic atterriamo sul IDA View A per quell'indirizzo:



Partendo dalla definizione teorica per cui:

- Le variabili sono ad un offset negativo rispetto al registro EBP
- I parametri si trovano ad un offset positivo rispetto ad EBP

Individuiamo 20 variabili con offset negativo:

```
; DWORD
           stdcall sub_10001656(LPVOID)
sub 10001656 proc near
var 675= byte ptr -675h
var_674= dword ptr -674h
hModule= dword ptr -670h
timeout= timeval ptr -66Ch
name= sockaddr ptr -664h
var_654= word ptr -654h
in= in_addr ptr -650h
Parameter= byte ptr -644h
CommandLine= byte ptr -63Fh
Data= byte ptr -638h
var_544= dword ptr -544h
var_50C= dword ptr -50Ch
var_500= dword ptr -500h
var_4FC= dword ptr -4FCh
readfds= fd_set ptr -4BCh
phkResult= HKEY__ ptr -3B8h
var_380= dword ptr -380h
var_1A4= dword ptr -1A4h
var_194= dword ptr -194h
WSAData= WSAData ptr -190h
arq 0= dword ptr 4
sub
        esp, 678h
push
        ebx
        ebp
push
push
        esi
         ndi
```

QUESITO 4

Quanti sono, invece, i parametri della funzione sopra?

Vi è una sola riga con offset negativo, dunque il parametro è uno:

```
stdcall sub 10001656(LPV0ID)
; DWORD
sub_10001656 proc near
var_675= byte ptr -675h
var_674= dword ptr -674h
hModule= dword ptr -670h
timeout= timeval ptr -66Ch
name= sockaddr ptr -664h
var_654= word ptr -654h
in= in_addr ptr -650h
Parameter= byte ptr -644h
CommandLine= byte ptr -63Fh
Data= byte ptr -638h
var 544= dword ptr -544h
var_50C= dword ptr -50Ch
var_500= dword ptr -500h
var_4FC= dword ptr -4FCh
readfds= fd_set ptr -4BCh
phkResult= HKEY__ ptr -3B8h
var_3B0= dword ptr -3B0h
var_1A4= dword ptr -1A4h
var 194= dword ptr -194h
WSAData = WSAData ntr -198h
arg 0= dword ptr 4
sub
         esp, 678h
push
         ebx
push
         ebp
push
         esi
nuch
         ndi
```