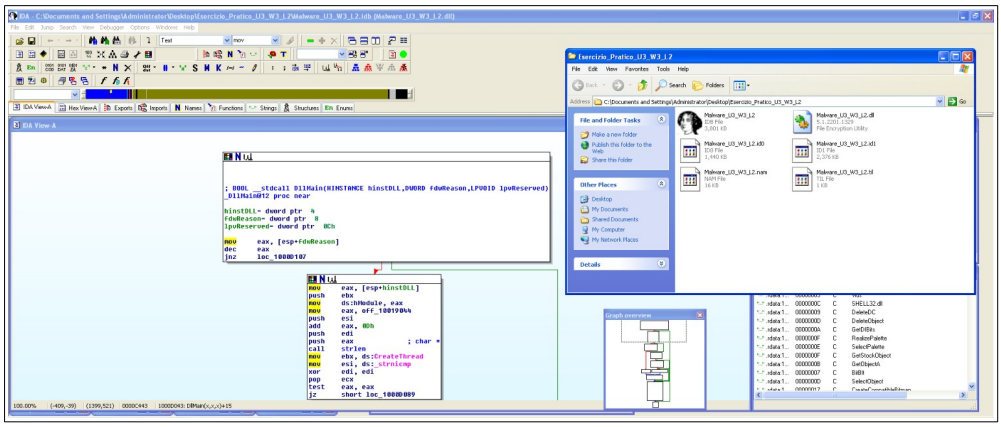
Analisi Statica Avanzata con IDA

Obiettivo: Analizzare il Malware indicato dalla traccia con il tool IDA Pro

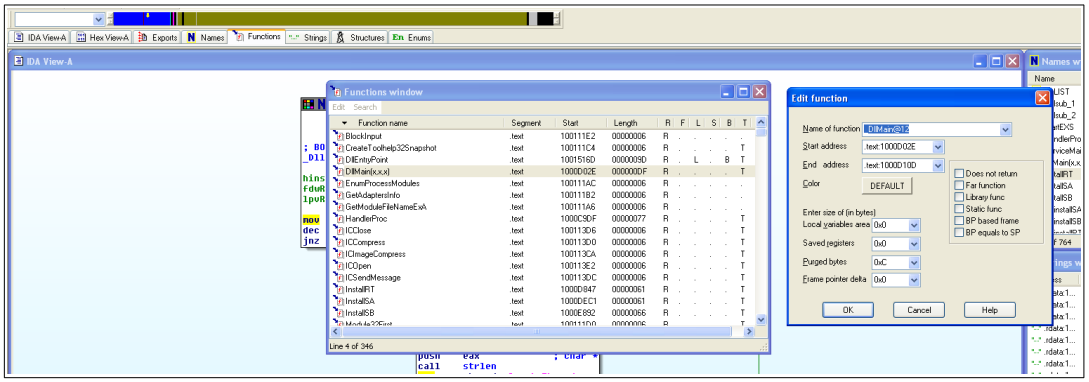
1) Apertura del File con IDA

L'esercizio chiede di utilizzare **IDA** per effettuare l'**Analisi Statica Avanzata** di un **Malware** all'interno della **Macchina MalwareAnalysis_Final**. **IDA** è un **Disassembler** e uno dei più utilizzati dagli analisti di sicurezza. Per cominciare ho avviato il *tool* e selezionato il File indicato (**Malware_U3_W3_L2**).

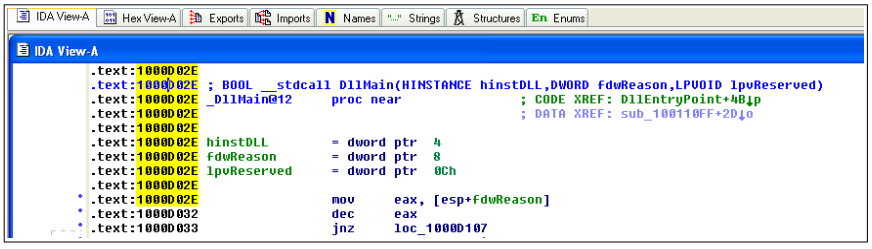


2) DLLMain

La prima richiesta della traccia è individuare l'indirizzo della Funzione **DLLMain** in esadecimale. Ho cominciato la ricerca andando sulla *Tab* "*Functions*".



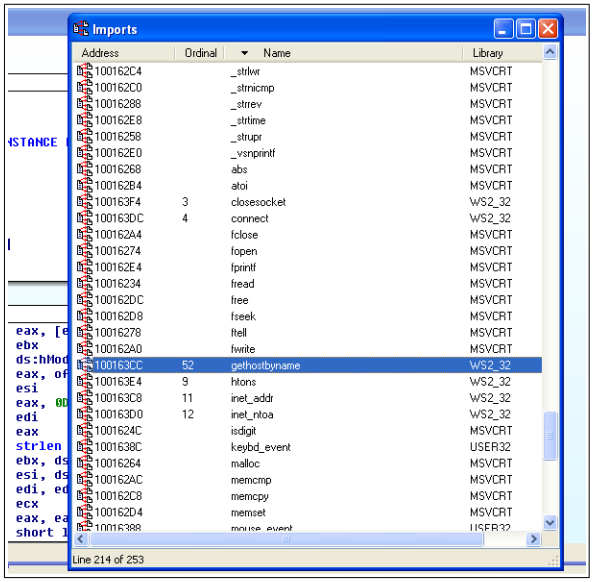
Ho messo l'elenco in modo tale da posizionare le varie funzioni in ordine alfabetico trovando subito quella richiesta. Con "*edit function*" si possono leggere gli indirizzi di **Inizio** e **Fine** della funzione selezionata ma oltre a quello ho utilizzato anche la barra spaziatrice per mettere la **Modalità Testuale** per conferma.



In conclusione l'indirizzo esadecimale è lo stesso, ovvero **1000D02E**

3) gethostbyname

Successivamente la traccia chiede di individuare la funzione **gethostbyname**.
Per trovarla sono andato nella *Tab* "*Imports*" impostando come in precedenza l'elenco in ordine alfabetico e cercando la funzione richiesta.



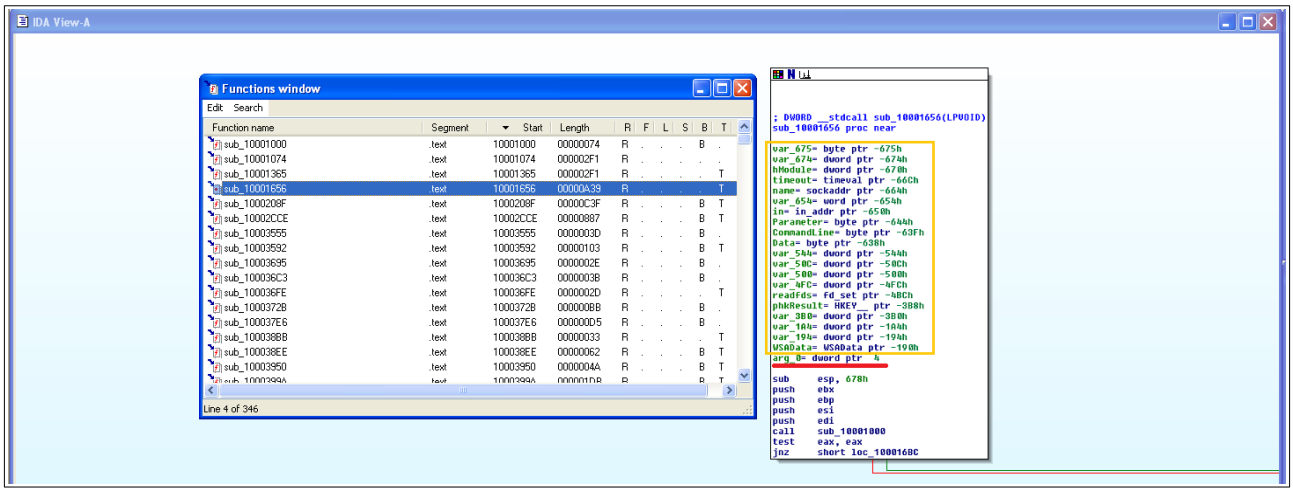
L'indirizzo è **100163CC**. Questa particolare funzione recupera le informazioni *host* corrispondenti ad un nome *host* da un database host.

Questa funzione è stata deprecata dall'introduzione della funzione **getaddrinfo**.

Se non si verifica alcun errore, **gethostbyname** restituisce un puntatore al database *host*. In caso contrario, restituisce un puntatore **Null** e un numero di errore specifico può essere recuperato chiamando **WSAGetLastError**.

4) Variabili e Parametri

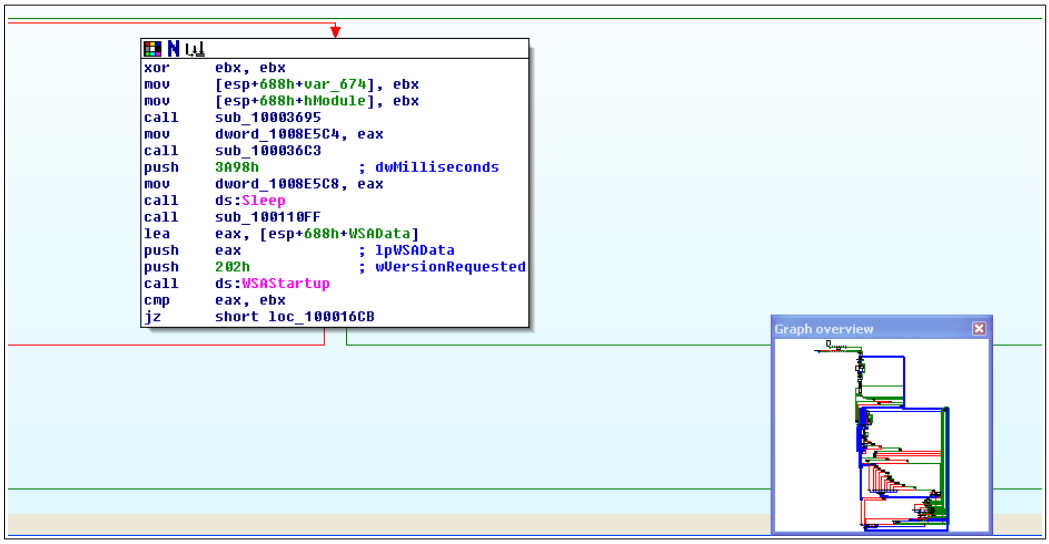
L'esercizio chiede inoltre di ricercare le **Variabili** e i **Parametri** della Funzione che si trova all'indirizzo **0 x 10001656**.



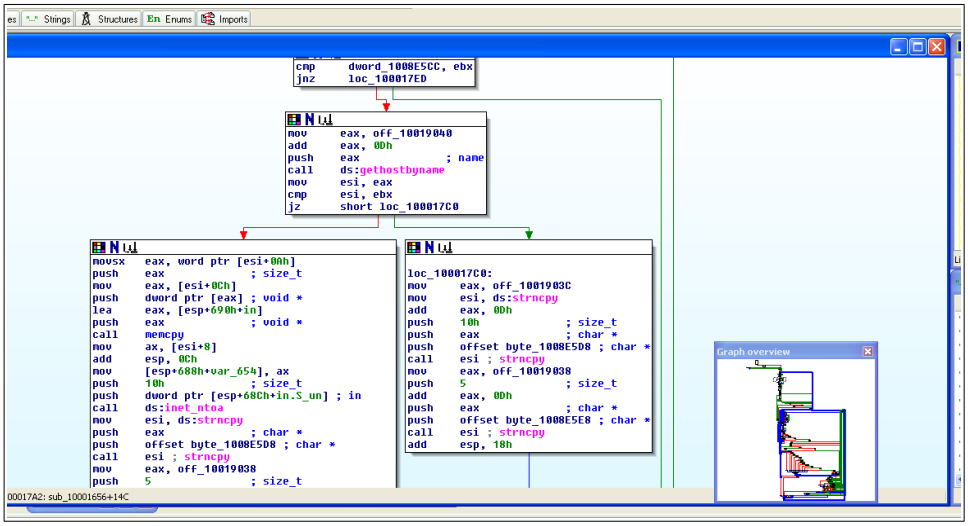
Questa è una subroutine (**sub_10001656**). Per trovarla ho utilizzato la *Tab* "*Functions*" elencando stavolta per **Indirizzo di Partenza** (Start). Una volta trovata ho fatto doppio click su di essa mostrandomi la Funzione da GUI.
Possiamo notare che le Variabili di questa funzione sono **20** (quelle del riquadro Giallo) perché hanno un offset negativo rispetto al registro EBP, mentre di Parametro ne abbiamo soltanto **uno** (sottolineato in rosso) avente un offset positivo rispetto ad EBP.

5) Comportamento

Come ultima richiesta la traccia chiede di ipotizzare il comportamento di questo **Malware**. Spulciando nella **IDA View-A**, ovvero l'**Interfaccia Grafica** del tool, ho notato la presenza di **WSAStartup**, una funzione che avvia l'uso della **DLL Winsock**, una libreria che aiuta **Windows** ad utilizzare in maniera semplificata i protocolli di rete TCP/IP.

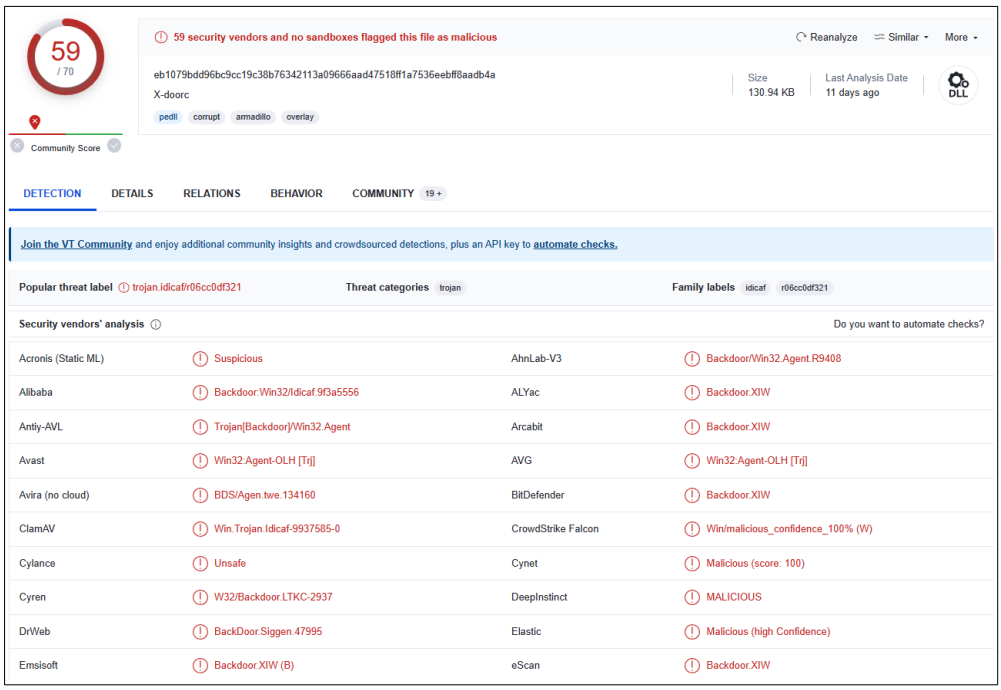


Facenti parte di questa libreria ho trovato alcune funzioni come **socket** e **gethostbyname** (visto in precedenza).



In questo caso suppongo che se abbia il nome dell'*host* tenti una connessione ad esso utilizzando **strncpy** (copia i caratteri di una stringa in un'altra), o almeno al suo indirizzo, altrimenti c'è la funzione **inet_ntoa** (che converte una stringa contenente un indirizzo con estensione IPV4 in un indirizzo dedicato alla struttura **in_addr**) che poi va a chiamare **strncpy**.

In conclusione: almeno per quanto riguarda questa prima analisi del **Malware** credo si tratti di una backdoor, confermato in parte anche dall'analisi dell'hash del **Malware** che ho effettuato alla fine su **VirusTotal**.



Dal sito si evince che il **Malware** è molto probabilmente un **Trojan** con funzione da **Backdoor**.