

S11 L2 ANALISI STATICA AVANZATA CON IDA

Nel'esercitazione di oggi andremo ad approcciarci all'analisi statica avanzata utilizzando il tool IDA. Quest'ultimo è un potente disassembler che traduce le istruzioni da linguaggio macchina a linguaggio assembly. Presenta un'utilissima interfaccia grafica con svariate possibilità, oltre alla traduzione completa del linguaggio macchina di un eseguibile in linguaggio assembly, IDA identifica: Funzioni / chiamate di funzione, Analisi dello stack, Variabili locali e parametri.

1. Individuare l'indirizzo della funzione **DLLMain** (così com'è, in esadecimale)

```
.text:1000D02E
.text:1000D02E
.text:1000D02E ; BOOL __stdcall DllMain(HINSTANCE hinstDLL,DWORD fdwReason,LPVOID lpvRes
.text:1000D02E _DllMain@12      proc near                                ; CODE XREF: DllEntryPoint+4B↓p
.text:1000D02E                                           ; DATA XREF: sub_100110FF+2D↓o
.text:1000D02E
.text:1000D02E hinstDLL      = dword ptr  4
.text:1000D02E fdwReason    = dword ptr  8
.text:1000D02E lpvReserved  = dword ptr 0Ch |
.text:1000D02E
.text:1000D02E mov     eax, [esp+fdwReason]
.text:1000D032 dec     eax
.text:1000D033 jnz     loc_1000D107
.text:1000D039 mov     eax, [esp+hinstDLL]
.text:1000D03D push    ebx
.text:1000D03E mov     ds:hModule, eax
.text:1000D043 mov     eax, off_10019044
.text:1000D048 push    esi

0000C42E | 1000D02E: DllMain(x,x,x)
```

Retrieving information from the database... ok

Notiamo che la
posizione della
funzione DLLMain è
all'indirizzo
"1000D02E"

2. Dalla scheda «imports» individuare la funzione «**gethostbyname** ». Qualè l'indirizzo de ll'import? **Cosa fa la funzione?**

IDA View-AHex View-AExportsImportsNamesFunctionsStringsStructuresEnums

Address	Ordinal	Name	Library
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

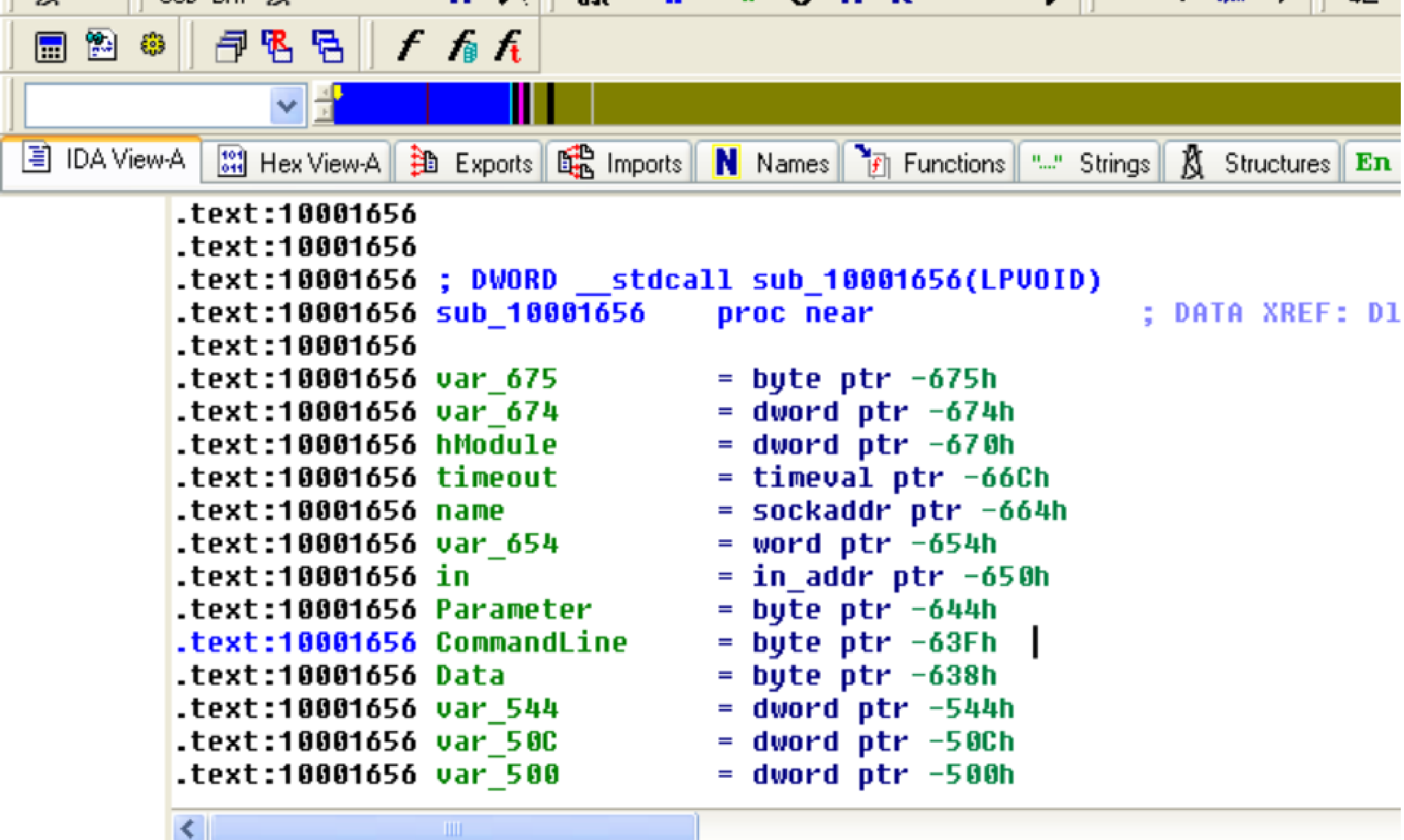
Line 235 of 253

Compiling file 'C:\Documents and Settings\Epicode_user\Desktop\ida pro\idc\ida.idc'...
Executing function 'main'...
Compiling file 'C:\Documents and Settings\Epicode_user\Desktop\ida pro\idc\onload.idc'...
Executing function 'OnLoad'...
IDA is analysing the input file...
You may start to explore the input file right now.
Using FLIRT signature: Microsoft VisualC 2-8/net runtime
Propagating type information...
Function argument information is propagated
The initial autoanalysis has been finished.

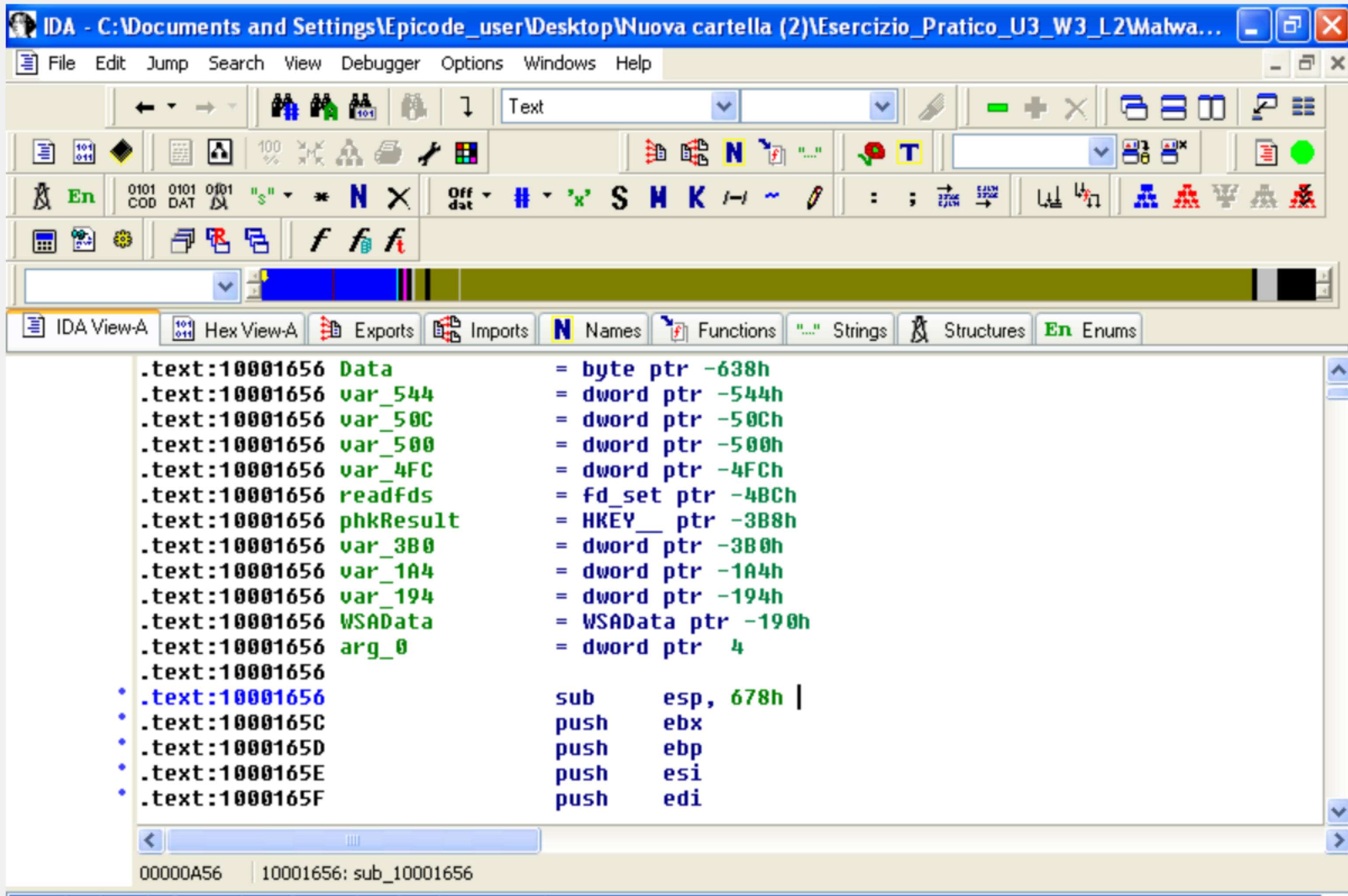
Notiamo
come
l'indirizzo
della
funzione sia
"100163CC"

La funzione gethostbyname è comunemente utilizzata per ottenere informazioni su un host tramite il suo nome.

3. Quante sono le **variabili locali** della **funzione** alla locazione di memoria 0x10001656?
4. Quanti sono, invece, i **parametri** della funzione sopra?



```
.text:10001656
.text:10001656
.text:10001656 ; DWORD __stdcall sub_10001656(LPVOID)
.text:10001656 sub_10001656 proc near ; DATA XREF: D1
.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
```



Come si evince dalle foto sopra, ci sono 20 variabili e 2 parametri all'indirizzo 1656. Come ben sappiamo si distinguono in base al offset rispetto al registro, se sono di offset negativo sono variabili se invece hanno offset positivo sono parametri

5. Inserire altre considerazioni macro livello sul malware (comportamento)

Si evince scorrendo nel codice e consultando le diverse funzioni che il malware ha importato che stiamo osservando un trojan che va a stabilire una connessione tramite i socket facciamo quindi riferimento ad una backdoor.