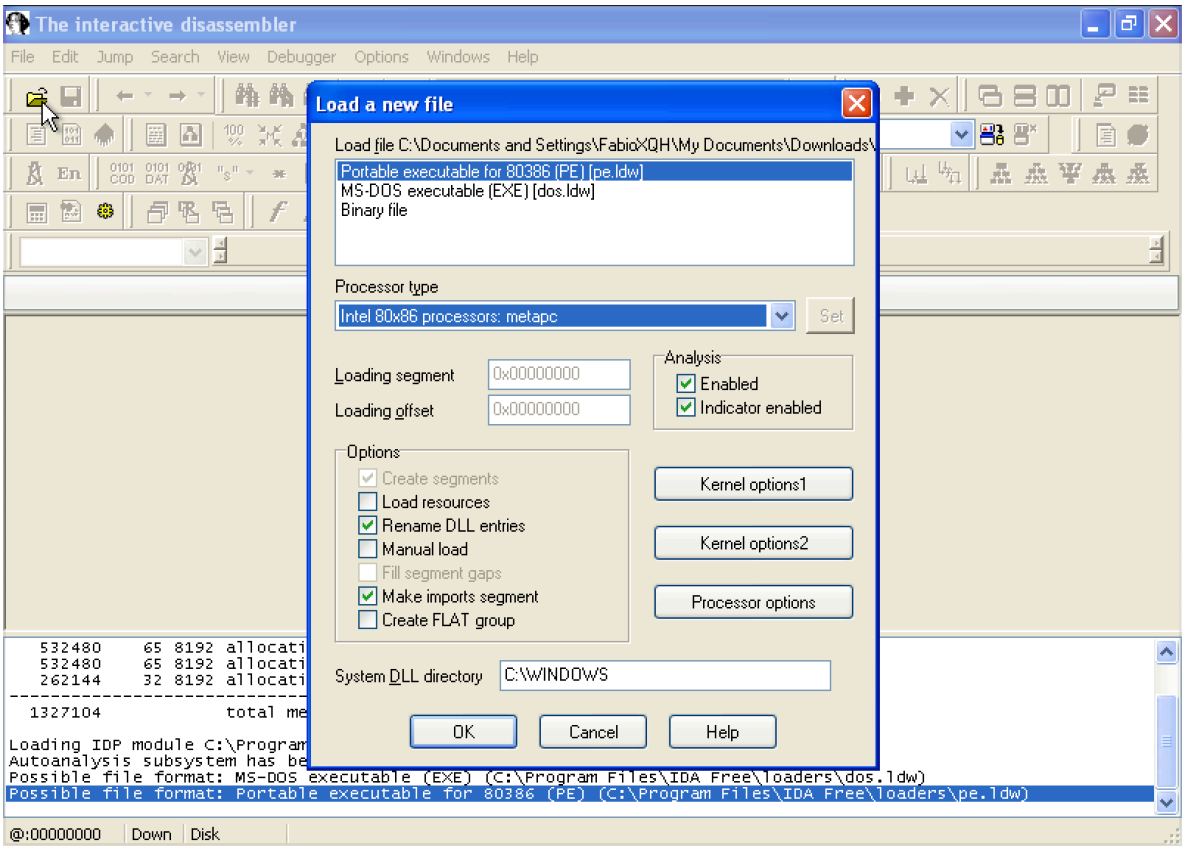
ANALISI STATICA AVANZATA CON IDA

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 0×10001656?
4. Quanti sono, invece, i parametri della funzione sopra?

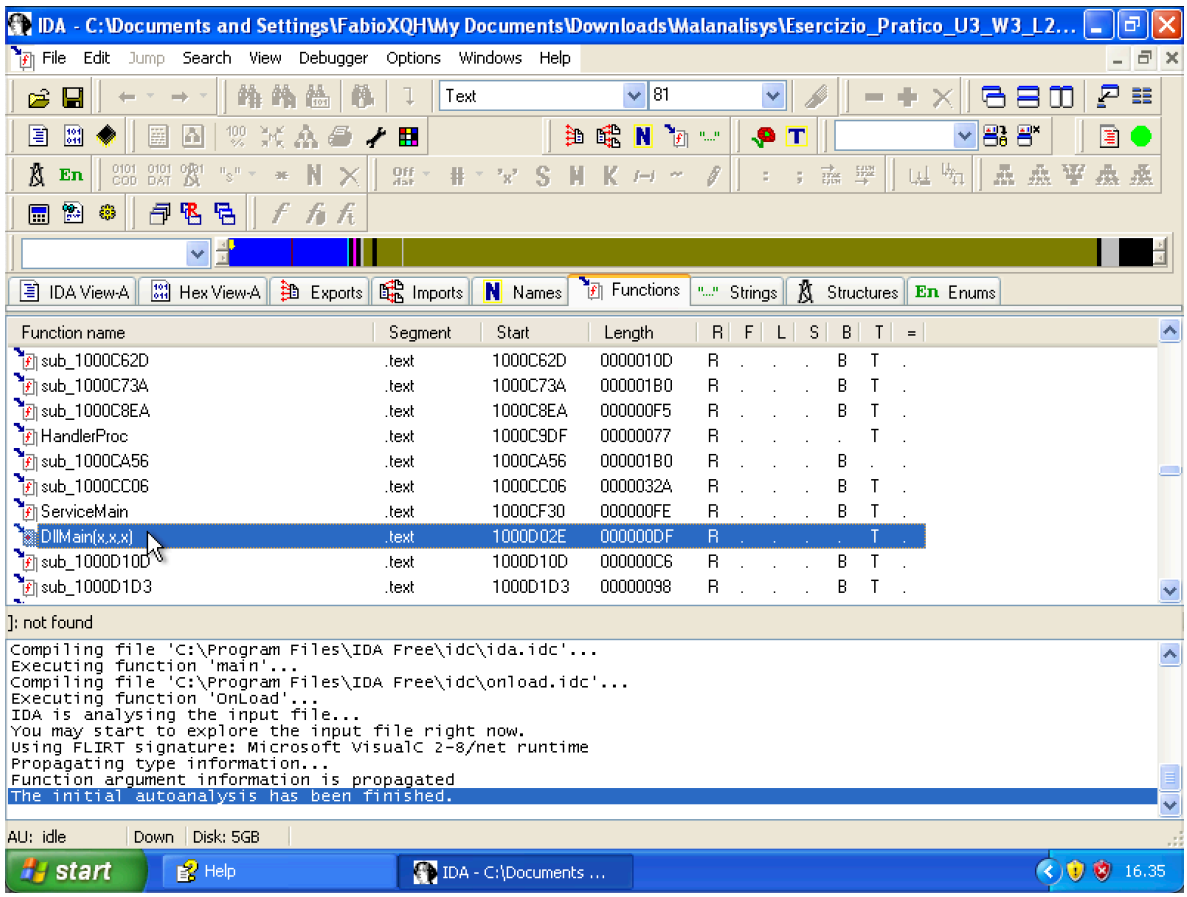
Per prima cosa andremo ad avviare la il programma nella nostra macchina virtuale. Una volta all'interno andremo ad inserire il file che abbiamo intenzione di analizzare, nel nostro caso è: "malware_U3_W3_L2".

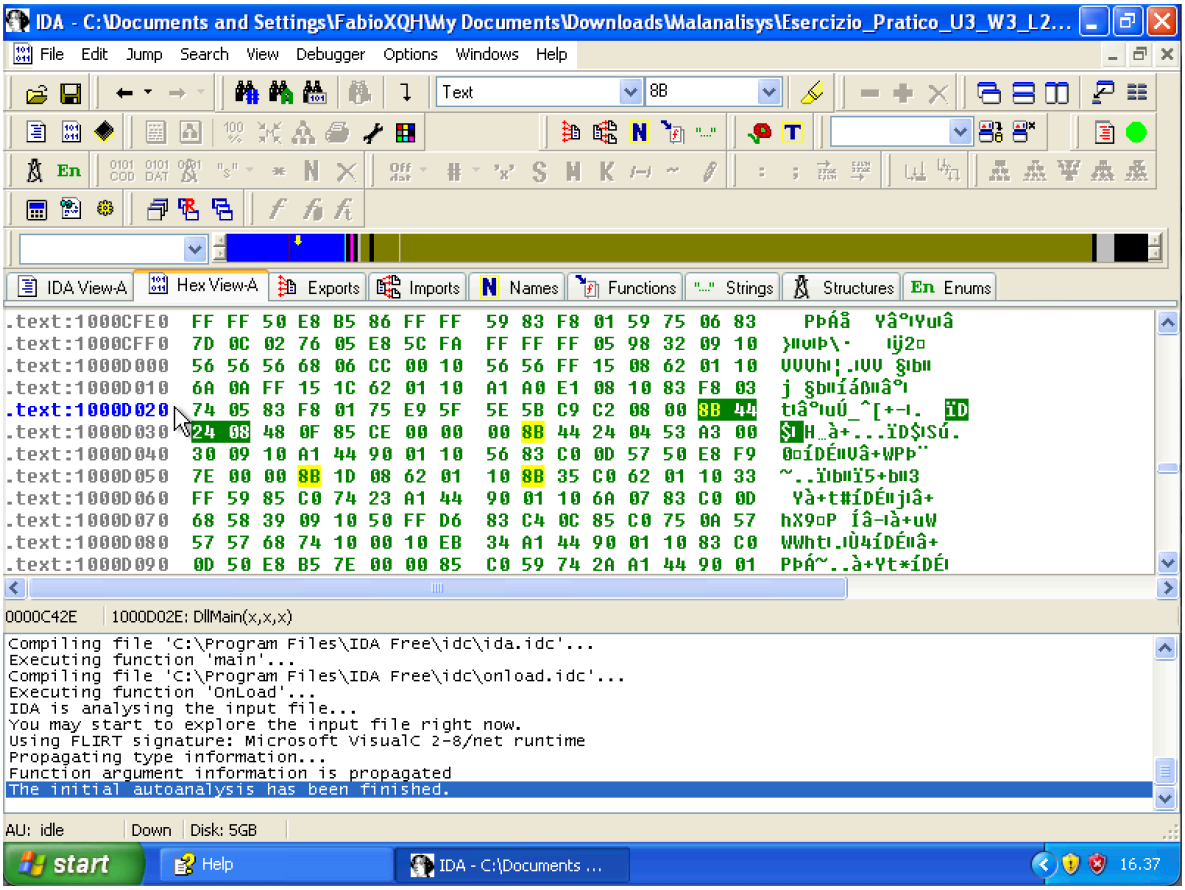


Come possiamo osservare, una volta terminata la scansione il programma ci offre diversi parametri da analizzare.

Indirizzo DLLMain:

Nel parametro funzioni andiamo ad analizzare la funzione DLLMain, la quale ci fornisce il proprio indirizzo 1000D02E.



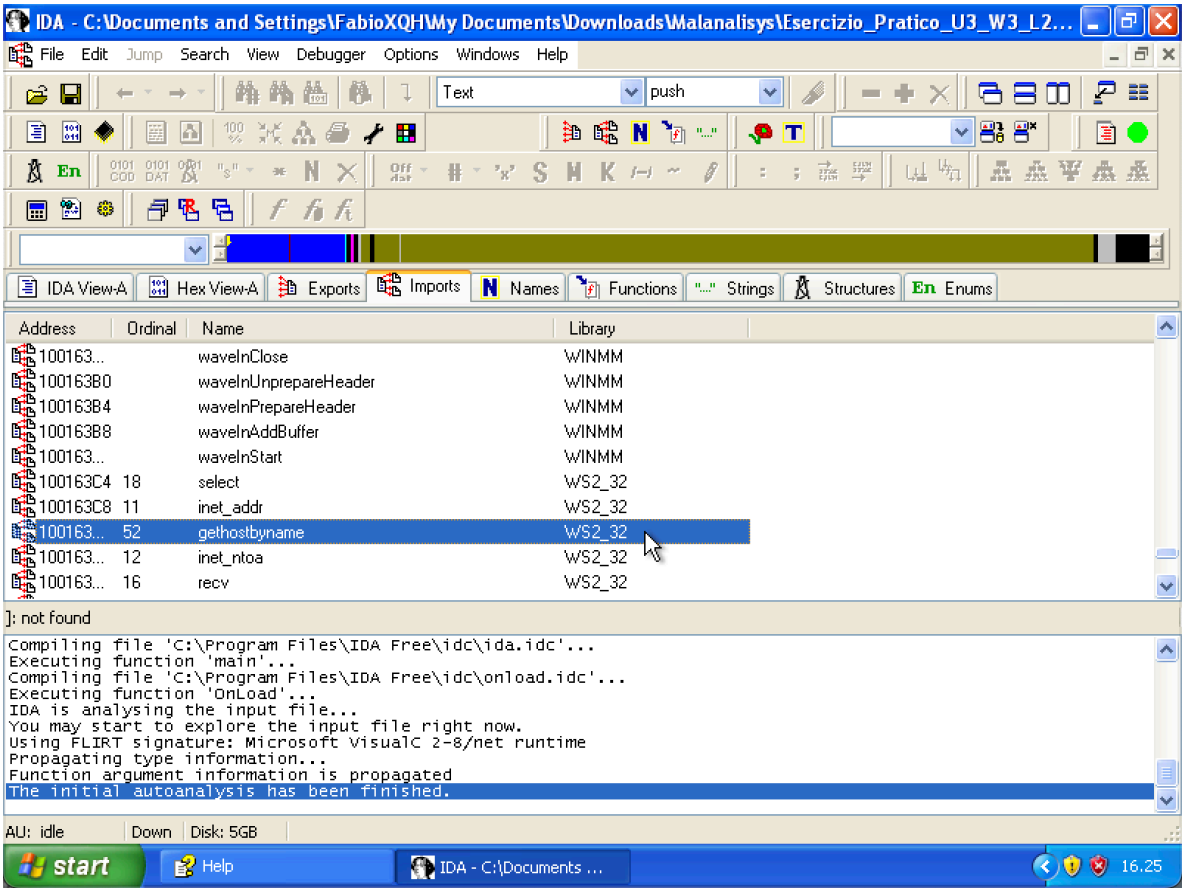


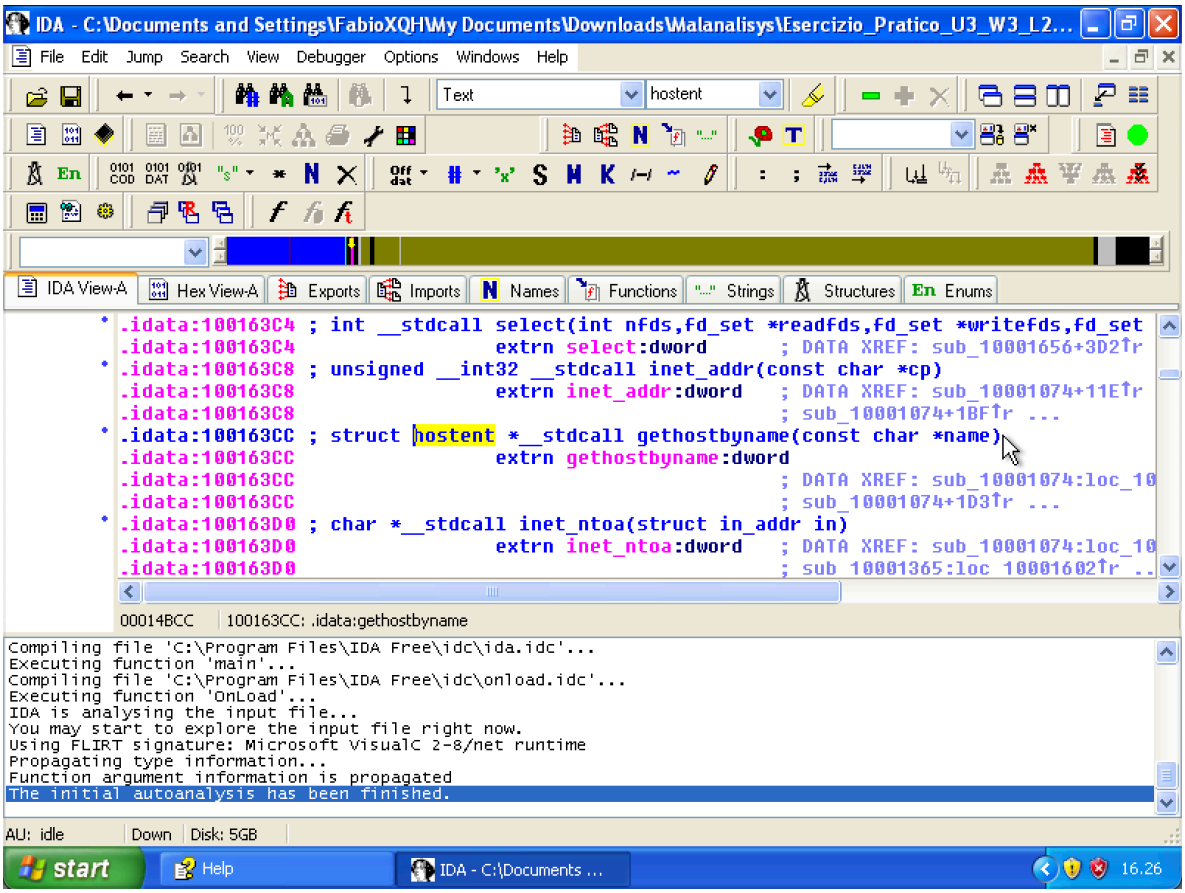

```
; int __fastcall sub_1000D3D0(int,int,char *)
sub_1000D3D0 proc near

var_410= byte ptr -410h
ExistingFileName= byte ptr -30Ch
NewFileName= byte ptr -208h
var_104= byte ptr -104h
arg_0= dword ptr  8

push    ebp
```

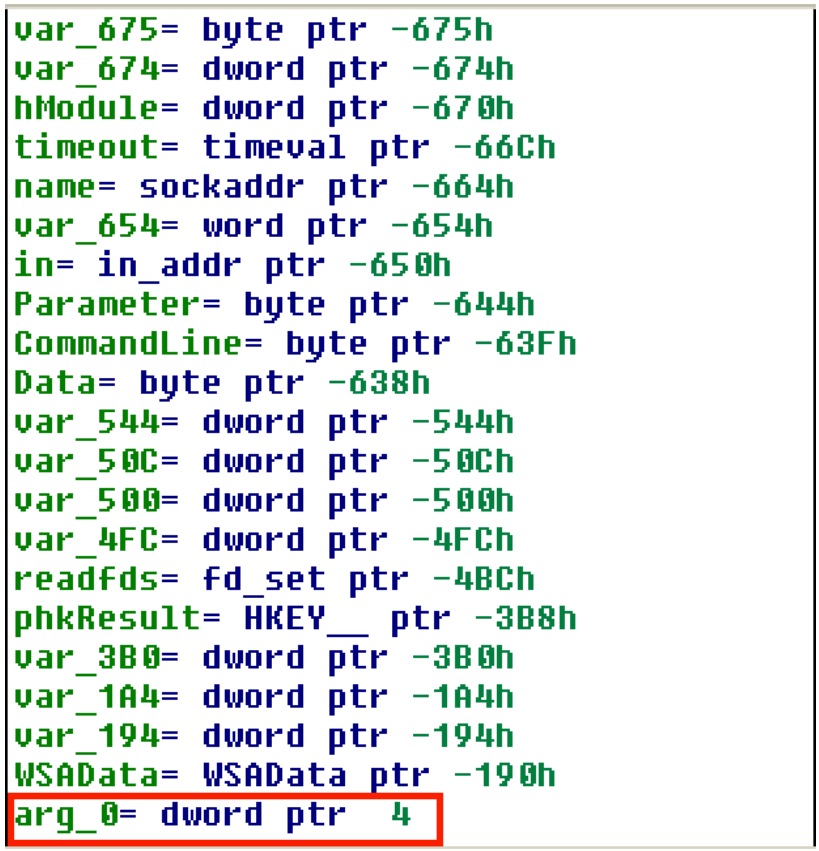
Variabili dell'indirizzo gethostbyname:
Sempre nel parametro imports andiamo ad analizzare la funzione gethostbyname, per cercare di scoprire il suo indirizzo, il quale ci indirizza al indirizzo .idata:100163CC.
Questo indirizzo ci porta alla funzione di "struct hostent", la quale ha il compito di raccogliere informazioni ci l'host della macchina bersaglio.





Variabili all'indirizzo 0x100001656:
Per scoprire quante variabili contenga l'indirizzo 0x10001656, occorre cercarlo nella

sezione jump - jump address e digitare l'indirizzo che si vuole cercare.
Nel nostro caso ci è stato mostrato a schermo che le variabili contenuto in questo indirizzo sono 20 in totale ad offset negativo rispetto ad EBP.



Mentre per quanto riguarda i parametri ne abbiamo riscontrato solo uno in positivo rispetto ad EBPl, ed è il seguente:

Per quanto riguarda il comportamento di questo malware, si può ipotizzare dalle immagini che esso raccolga le chiavi di registro del sistema (call ds:RegOpenKeyExa), per poi andarlo a modificare attraverso (call:RegSetValueExa).

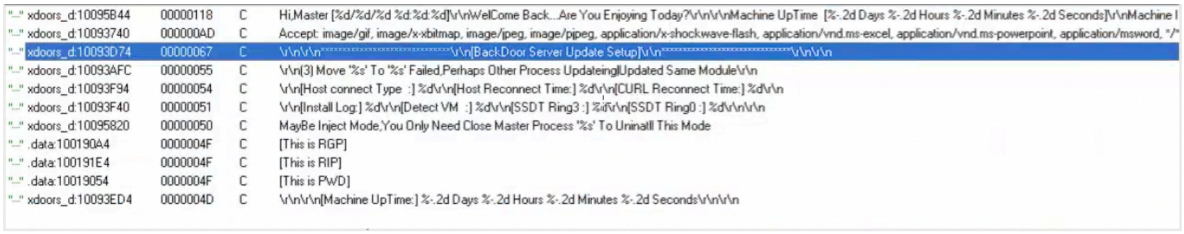
```
; int __cdecl sub_1000564E(LPCSTR lpValueName,BYTE Data)
sub_1000564E proc near

hKey= dword ptr -4
lpValueName= dword ptr  8
Data= byte ptr  0Ch

push    ebp
mov     ebp, esp
push    ecx
lea     eax, [ebp+hKey]
push    esi
push    eax             ; phkResult
xor     esi, esi
push    0F003Fh         ; samDesired
push    esi             ; ulOptions
push    offset aSoftwareMicros ; "SOFTWARE\\Microsoft\\Windows\\CurrentVersi"...
push    80000002h       ; hKey
call    ds:RegOpenKeyExA
test    eax, eax
jnz     short loc_1000568F
```

```
lea     eax, [ebp+Data]
push    4                   ; cbData
push    eax                 ; lpData
push    4                   ; dwType
push    esi                 ; Reserved
push    [ebp+lpValueName]   ; lpValueName
push    [ebp+hKey]          ; hKey
call    ds:RegSetValueExA
test    eax, eax
jnz     short loc_1000568F
```

A tal proposito, per avere maggiori dettagli siamo andati a calcolare gli hash del file del malware e caricarli sul sito di virus total il quale ci ha fornito l'identità di questo malware, ovvero di una backdoor.



https://www.virustotal.com/gui/file/eb1079bdd96bc9cc19c38b76342113a09666aad47518ff1a7536eebff8aadb4a/detection