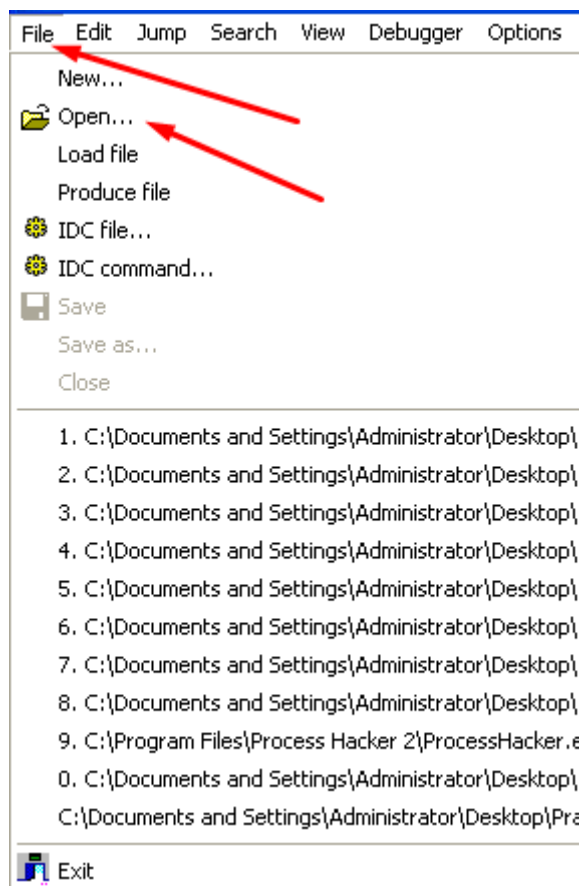
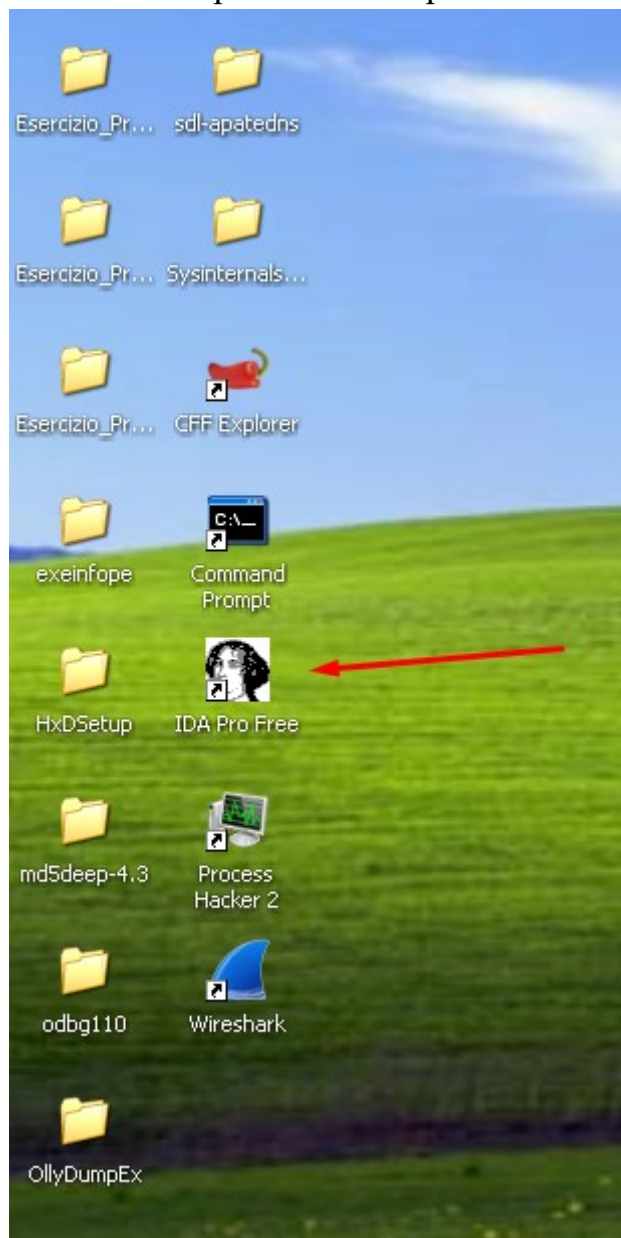
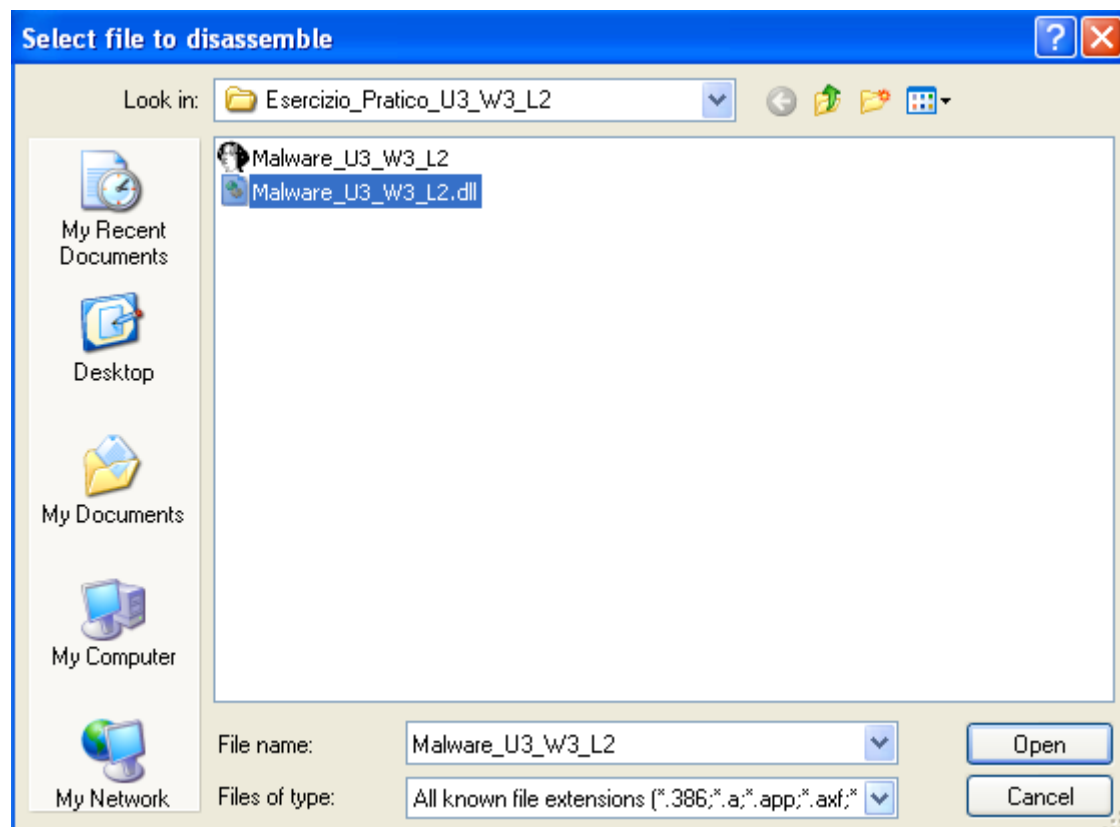


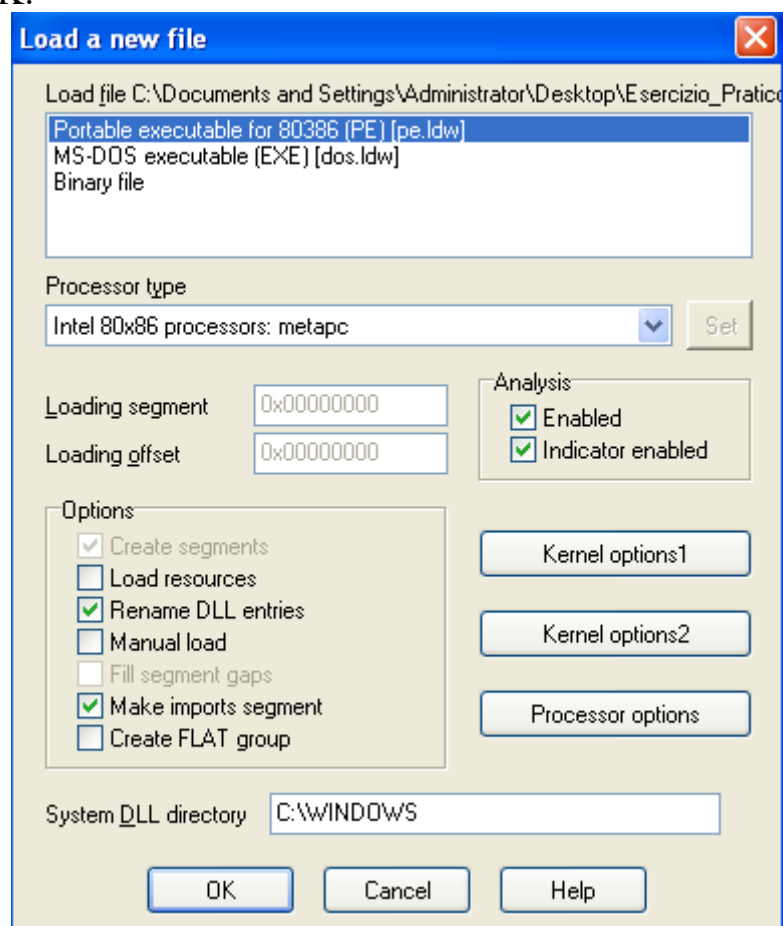
Report Analisi Statica con IDA

Come richiesto dall'esercizio, andiamo ad avviare IDA dal nostro desktop ed andiamo ad aprire il nostro potenziale malware:

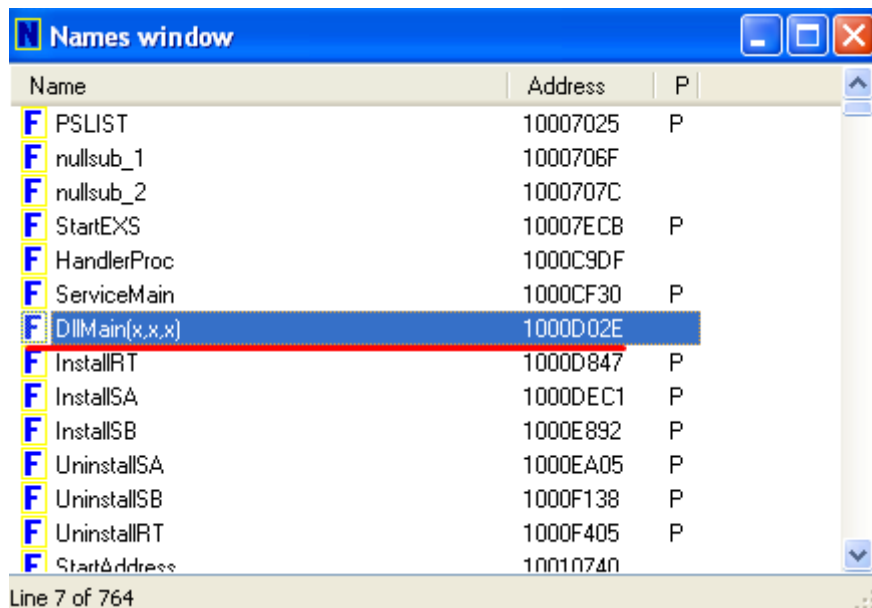




Una volta cliccato su open, lasciamo le configurazioni così per come sono e clicchiamo su OK:



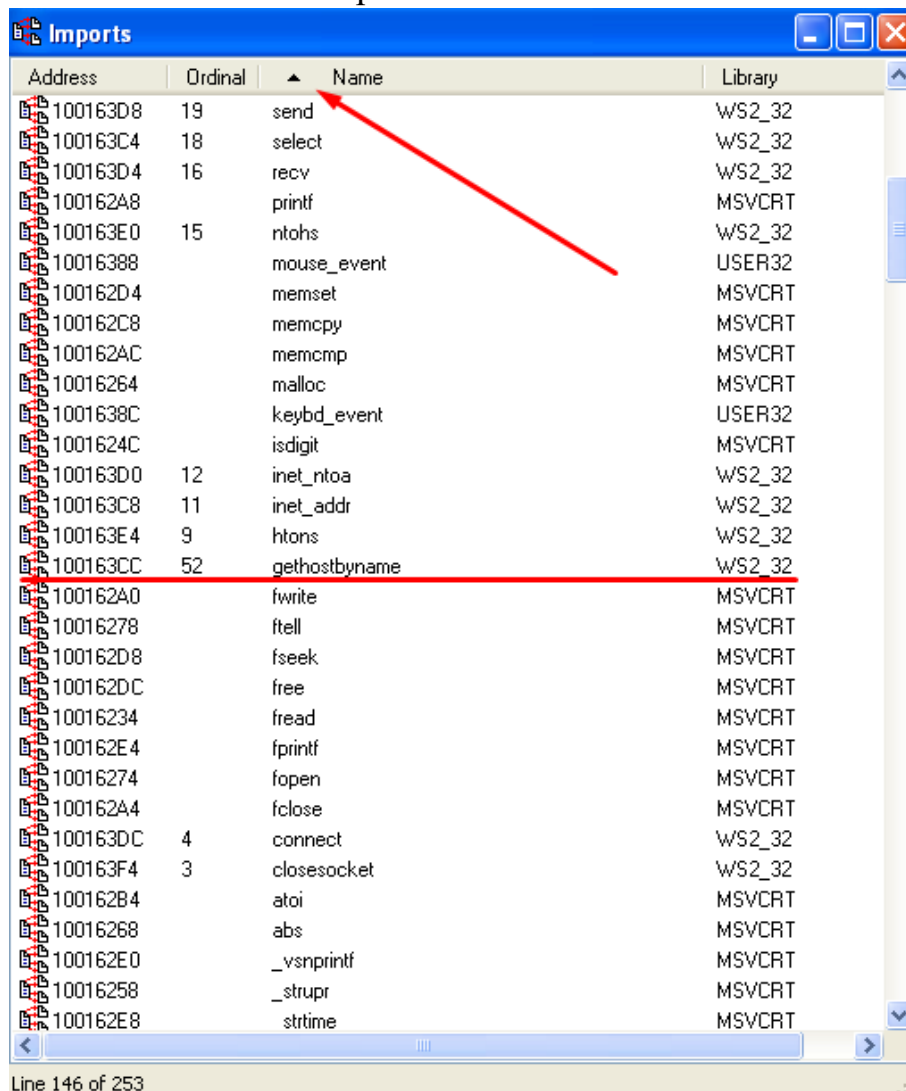
Adesso andiamo a cercare in che indirizzo di memoria si trova la funzione **DllMain** tramite il pannello **Names window**:

A screenshot of the 'Names window' in a debugger. It shows a list of symbols with their names, addresses, and types. The entry 'DllMain(x,x,x)' is highlighted with a red line. A red arrow points from the text 'gethostbyname' in the next section to the 'Name' column header in this window.

Name	Address	P
PSLIST	10007025	P
nullsub_1	1000706F	
nullsub_2	1000707C	
StartEXS	10007ECB	P
HandlerProc	1000C9DF	
ServiceMain	1000CF30	P
DllMain(x,x,x)	1000D02E	
InstallIRT	1000D847	P
InstallSA	1000DEC1	P
InstallSB	1000E892	P
UninstallSA	1000EA05	P
UninstallSB	1000F138	P
UninstallIRT	1000F405	P
StartAddress	10010740	

Line 7 of 764

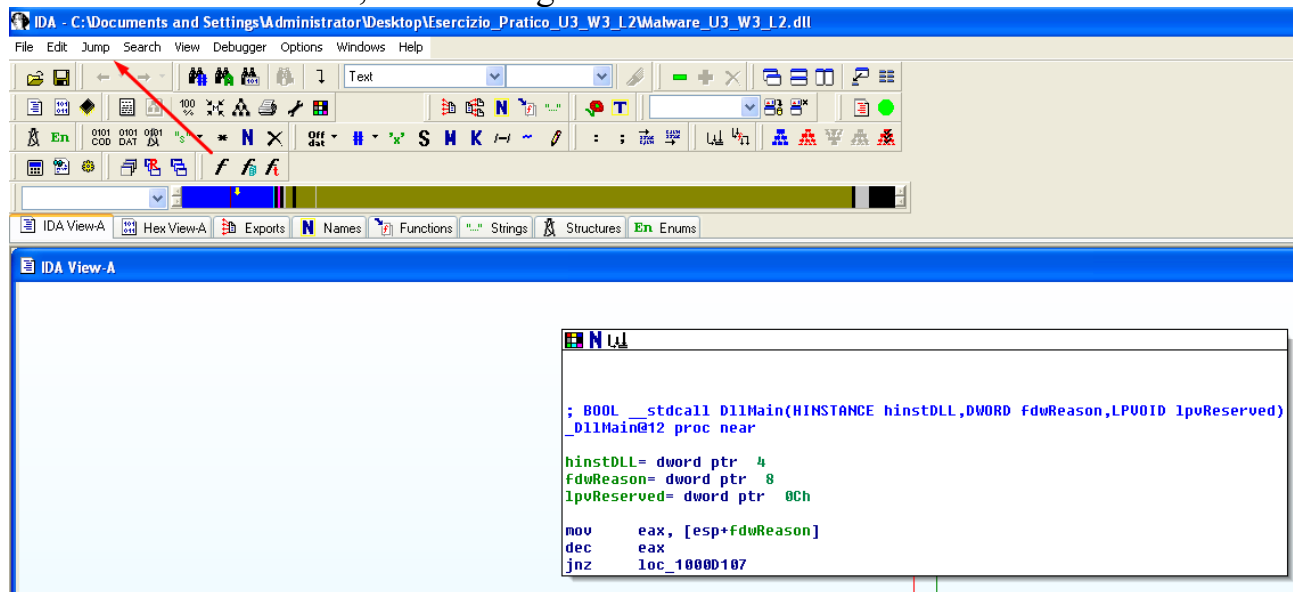
Adesso cercando nella finestra **imports** l'indirizzo di memoria della funzione **gethostbyname** andando ad ordinare per nome:

A screenshot of the 'Imports' window in a debugger. It shows a list of imported functions with their addresses, ordinals, names, and libraries. The entry 'gethostbyname' is highlighted with a red line. A red arrow points from the text 'gethostbyname' in the previous section to this entry.

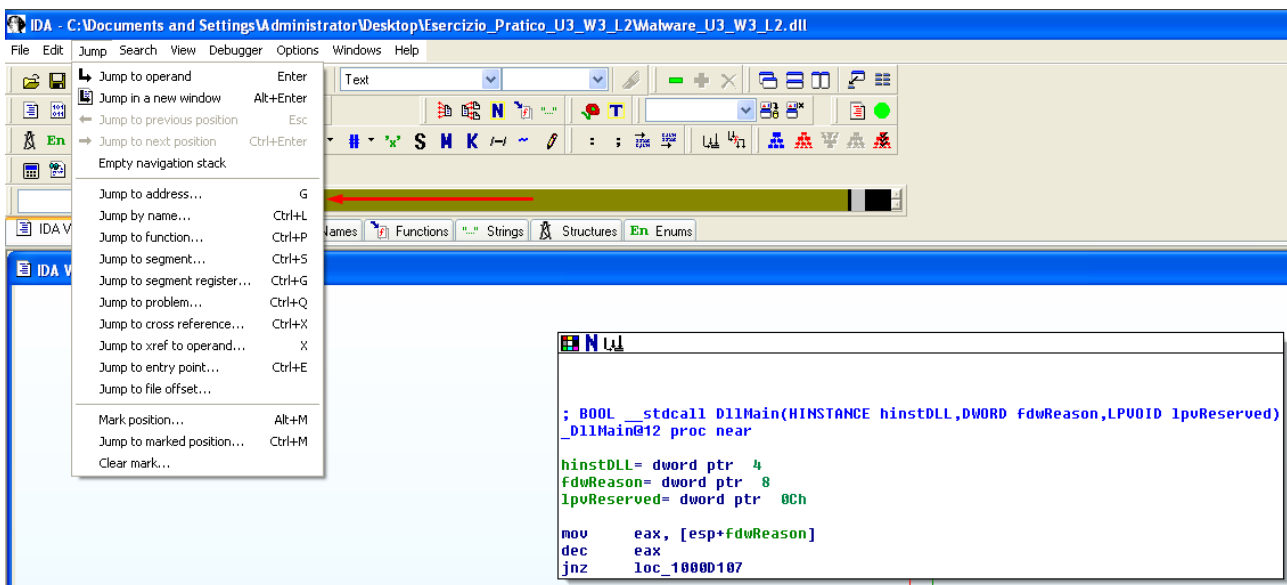
Address	Ordinal	Name	Library
100163D8	19	send	WS2_32
100163C4	18	select	WS2_32
100163D4	16	recv	WS2_32
100162A8		printf	MSVCRT
100163E0	15	ntohs	WS2_32
10016388		mouse_event	USER32
100162D4		memset	MSVCRT
100162C8		memcpy	MSVCRT
100162AC		memcmp	MSVCRT
10016264		malloc	MSVCRT
1001638C		keybd_event	USER32
1001624C		isdigit	MSVCRT
100163D0	12	inet_ntoa	WS2_32
100163C8	11	inet_addr	WS2_32
100163E4	9	htons	WS2_32
100163CC	52	gethostbyname	WS2_32
100162A0		fwrite	MSVCRT
10016278		ftell	MSVCRT
100162D8		fseek	MSVCRT
100162DC		free	MSVCRT
10016234		fread	MSVCRT
100162E4		fprintf	MSVCRT
10016274		fopen	MSVCRT
100162A4		fclose	MSVCRT
100163DC	4	connect	WS2_32
100163F4	3	closesocket	WS2_32
100162B4		atoi	MSVCRT
10016268		abs	MSVCRT
100162E0		_vsnprintf	MSVCRT
10016258		_strupr	MSVCRT
100162E8		strtime	MSVCRT

Line 146 of 253

Adesso andiamo a cercare le variabili locali ed i parametri della funzione all'indirizzo di memoria 0x1001656, come di seguito:



Adesso cerchiamo l'indirizzo tramite *Jump address* scrivendo per l'appunto 0x1001656



Fatto ciò, clicchiamo Invio e ci ritroveremo all'indirizzo di memoria richiesto trovando la funzione richiesta. Fatto ciò distinguiamo quali sono le variabili e qual è il parametro, come di seguito

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