

ANALISI STATICA CON IDA

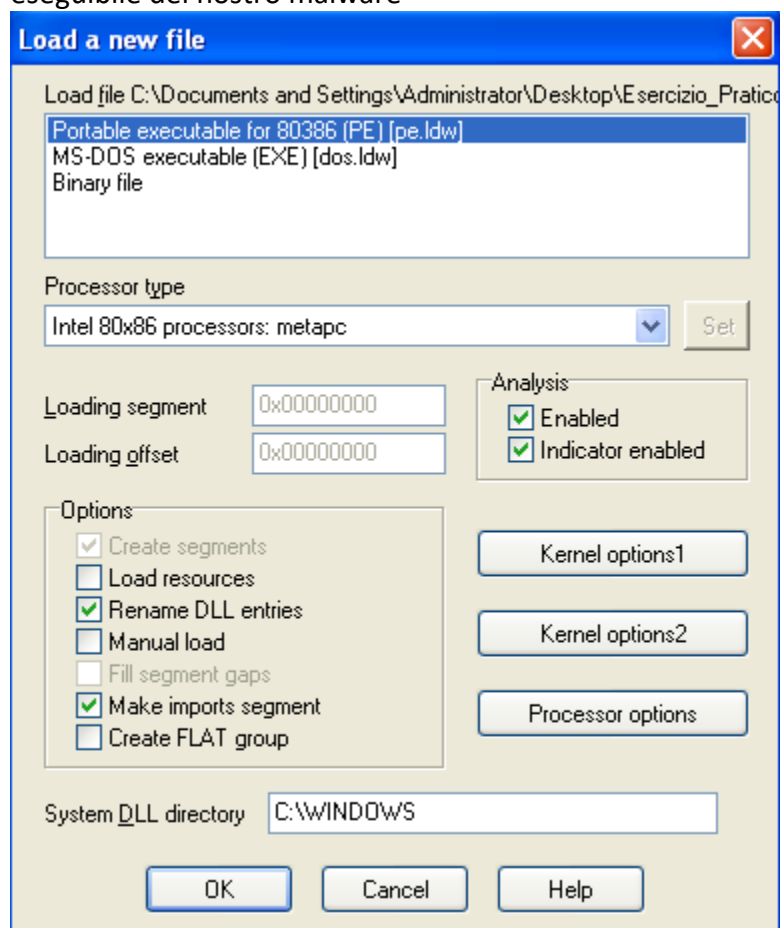
FRANCESCO PERSICHETTI

L'esercizio di oggi consiste nell'analizzare il codice malevolo con il dissambler IDA per svolgere i seguenti task:

1. Individuare l'indirizzo della funzione DLLMain
2. Dalla scheda imports individuare la funzione **"gethostbyname"** e il suo indirizzo
3. Quante sono le variabili della funzione alla locazione di memoria 0x10001656?
4. Quanti sono i parametri della stessa funzione al punto 3?
5. Considerazioni sul malware analizzato

1.

Per iniziare con l'analisi del malware utilizziamo il dissambler IDA con cui apriremo il codice eseguibile del nostro malware



Una volta aperto ci verrà restituito tutto il codice assembly del malware, con le varie funzioni importate. Se andiamo nella scheda “Names” del programma ci verranno elencate tutte le funzioni per nome, e noi andremo ad evidenziare ai fini del nostro task, l’indirizzo di memoria della funzione “DLLMain” che è uguale a **1000D02E**

IDA View-A Hex View-A Exports Imports N Names		
Name	Address	P
F nullsub_2	1000707C	
F StartEXS	10007ECB	P
F HandlerProc	1000C9DF	
F ServiceMain	1000CF30	P
F DLLMain(x,x,x)	1000D02E	
F InstallIRT	1000D847	P

2.

Sempre muovendosi tra le varie schede che ci sono nel programma, andiamo nella finestra “Imports” e andiamo ad individuare l’indirizzo di memoria della funzione “gethostbyname”.

L’indirizzo è il seguente: 100163CC

	100163CC	52	gethostbyname	WS2_32
---	----------	----	---------------	--------

3-4

Nella finestra “Functions” mettiamo nella barra di ricerca l’indirizzo di memoria **10001656** per andare a individuare la funzione corrispondente. Una volta trovata con il doppio click si aprirà tutto il codice inerente alla funzione con le relative variabili e parametri.

Variabili in rosso

Parametri in blu

```

N LAL
; 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 -48Ch
phkResult= HKEY__ ptr -388h
var_388= dword ptr -380h
var_1A4= dword ptr -1A4h
var_194= dword ptr -194h
USAData= USAData ptr -190h
arg_0= duord ptr 4

sub esp, 678h
push ebx
push ebp
push esi
push edi
call sub_10001000
test eax, eax
jnz short loc_1000160C

```

5.

Dall'analisi più approfondita del codice sembra che questo malware si occupi di creare una backdoor sulla macchina vittima, come si può anche notare dai seguenti screen.

