

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 (così com'è, in esadecimale)
2. Dalla scheda «imports» individuare la funzione «gethostbyname». Qual è l'indirizzo dell'import? Cosa fa la funzione?
3. Quante sono le variabili locali della funzione alla locazione di memoria 0x10001656?
4. Quanti sono, invece, i parametri della funzione sopra?
5. Inserire altre considerazioni macro livello sul malware (comportamento)

X	IDA View-A	X	Hex View-A	X	Structures	X	En	Enums	X	Imports	X	Exports
1000CFEE	06 83 7D 0C 02 76 05 E8 5C FA FF FF FF 05 98 32											.â}..v.p\..ü2
1000CFE	09 10 56 56 56 68 06 CC 00 10 56 56 FF 15 08 62											..UUUh..!..UU..b
1000D00E	01 10 6A 0A FF 15 1C 62 01 10 A1 A0 E1 08 10 83											..j..b..íáß..â
1000D01E	F8 03 74 05 83 F8 01 75 E9 5F 5E 5B C9 C2 08 00											°.t.â°.uÚ_^ [+..
1000D02E	8B 44 24 08 48 0F 85 CE 00 00 00 8B 44 24 04 53											ïD\$.H.à+...ïD\$.S
1000D03E	A3 00 30 09 10 A1 44 90 01 10 56 83 C0 0D 57 50											ú.0..íDÉ..Uâ+.WP
1000D04E	E8 F9 7E 00 00 8B 1D 08 62 01 10 8B 35 C0 62 01											p""~..ï..b..ï5+b.

Indirizzo esadecimale di DLLMain


```
; BOOL __stdcall DllMain(HINSTANCE hinstDLL,
_DllMain@12 proc near

hinstDLL = dword ptr 4
```

X	IDA View-A	X	Hex View-A	X	Structures	X	En	Enums	X	Imports	X	Exports
1000CFEE	06 83 7D 0C 02 76 05 E8	5C FA FF FF FF 05 98 32	.â}..v.p\-. .ÿ2									
1000CFFE	09 10 56 56 56 68 06 CC	00 10 56 56 FF 15 08 62	..UUUh.!..UU ..b									
1000D00E	01 10 6A 0A FF 15 1C 62	01 10 A1 A0 E1 08 10 83	..j. ..b..íáð..â									
1000D01E	F8 03 74 05 83 F8 01 75	E9 5F 5E 5B C9 C2 08 00	°.t.â°.uú_^[+-..									
1000D02E	8B 44 24 08 48 0F 85 CE	00 00 00 8B 44 24 04 53	ïD\$.H.à+...ïD\$.S									
1000D03E	A3 00 30 09 10 A1 44 90	01 10 56 83 C0 00 57 50	ú.0..íDÉ..Uâ+.WP									
1000D04E	E8 F9 7E 00 00 8B 1D 08	62 01 10 8B 35 C0 62 01	p""~..ï..b..ï5+b.									

FUNZIONE: gethostbyname

La sua funzione principale è ottenere informazioni sulle risorse di rete, come indirizzi IP, associati a un determinato nome host

 00000000100163CC	52 gethostbyname	WS2_32
INDIRIZZO	FUNZIONE	LIBRERIA

VARIABILI DELLA FUNZIONE getaddressbyhost

```
hLibModule= dword ptr -670h
timeout= timeval ptr -66Ch
name= sockaddr ptr -664h
var_654= word ptr -654h
Dst= dword ptr -650h
Parameter= byte ptr -644h
var_640= byte ptr -640h
CommandLine= byte ptr -63Fh
Source= byte ptr -63Dh
Data= byte ptr -638h
var_637= byte ptr -637h
var_544= dword ptr -544h
var_50C= dword ptr -50Ch
var_500= dword ptr -500h
Buf2= byte ptr -4FCh
readfds= fd_set ptr -4BCh
phkResult= byte ptr -3B8h
var_3B0= dword ptr -3B0h
var_1A4= dword ptr -1A4h
var_194= dword ptr -194h
WSAData= WSAData ptr -190h
```

Le variabili sono ad un offset negativo rispetto al registro EBP e sono in totale 21

PARAMETRI

E' un argomento di una funzione o una variabile di tipo "double word" (32 bit) che si trova a un offset di 4 byte rispetto a un certo punto di riferimento.

```
arg_0= dword ptr 4
```

I parametri si trovano ad un offset positivo rispetto ad EBP

Il comportamento del malware è un trojan. E tramite una backdoor può modificare file e avere informazione sugli indirizzi della macchina vittima.

I parametri potrebbero essere utilizzati per stabilire una connessione con un server remoto controllato dal malware. Questo consentirebbe al malware di ricevere istruzioni da un attaccante e trasmettere informazioni dal sistema infetto

E se un malware interviene su chiamate a funzioni di sistema come CreateThread, potrebbe cercare di eseguire codice dannoso all'interno di un nuovo thread, rendendo più difficile la rilevazione e il monitoraggio da parte degli antivirus e delle soluzioni di sicurezza