



INDICE

- Traccia Esercizio	pag.4
- Introduzione	pag.5-8
- VirtualBox	pag.5
- CFF Explorer	pag.6-7-8
- Librerie Importate	pag.9-10
- Sezioni Malware	pag.11-12
- Costrutti Noti	pag.13-14
- Funzionalità Implementata	pag.15
- Ringraziamenti	pag.16

TRACCIA ESERCIZIO

Con riferimento al file `Malware_U3_W2_L5` presente all'interno della cartella «Esercizio_Pratico_U3_W2_L5» sul desktop della macchina virtuale dedicata per l'analisi dei malware, rispondere ai seguenti quesiti:

Quali librerie vengono importate dal file eseguibile?

Quali sono le sezioni di cui si compone il file eseguibile del malware?

Con riferimento alla figura in slide 4, risponde ai seguenti quesiti:

Identificare i costrutti noti (creazione dello stack, eventuali cicli, costrutti)

Ipotizzare il comportamento della funzionalità implementata

TRACCIA ESERCIZIO

```
push    ebp
mov     ebp, esp
push    ecx
push    0          ; dwReserved
push    0          ; lpdwFlags
call    ds:InternetGetConnectedState
mov     [ebp+var_4], eax
cmp     [ebp+var_4], 0
jz      short loc_40102B
```

```
push    offset aSuccessInterne ; "Success: Internet Connection\n"
call    sub_40117F
add     esp, 4
mov     eax, 1
jnp     short loc_40103A
```

```
loc_40102B:
push    offset aError1_1NoInte ; "Error 1.1: No Internet\n"
call    sub_40117F
add     esp, 4
xor     eax, eax
```

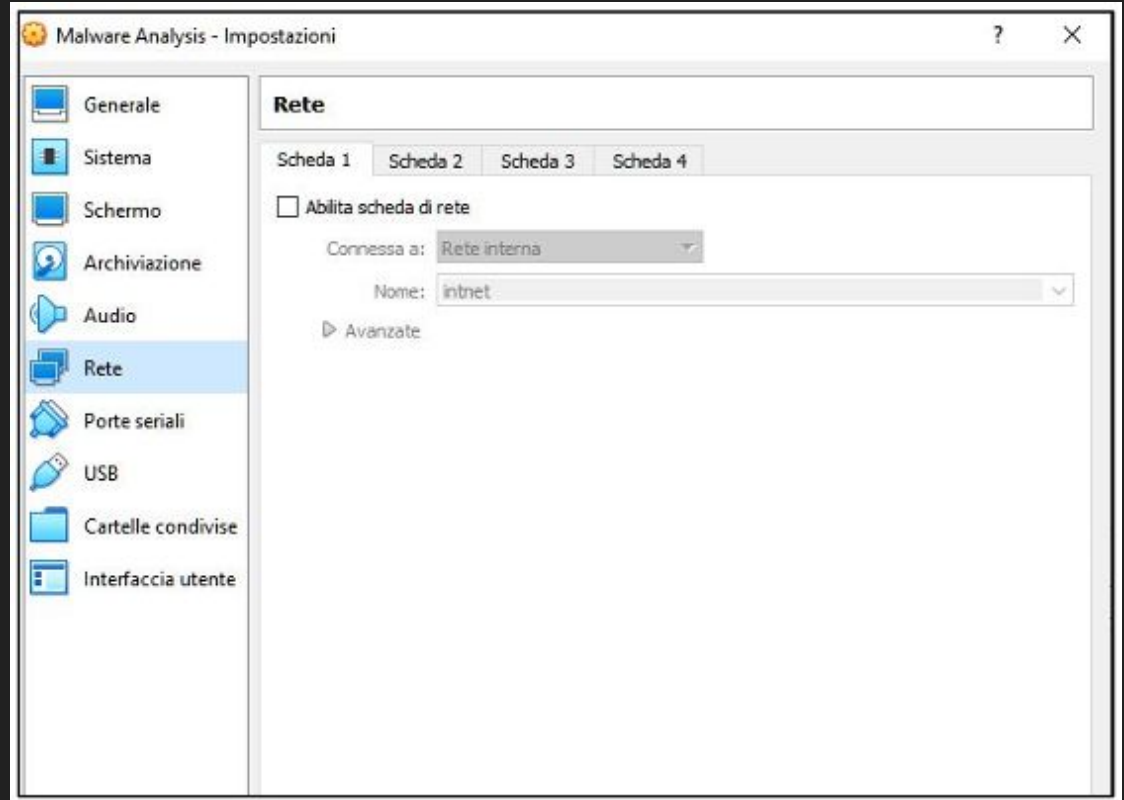
```
loc_40103A:
mov     esp, ebp
pop     ebp
retn
sub_401000 endp
```

INTRODUZIONE

Prima di svolgere le richieste dell'esercizio è **NECESSARIO** lavorare in sicurezza, per questo l'esercizio andrà svolto su una macchina virtuale installata sul nostro PC ma totalmente isolata da esso, in modo che i malware testati non si propaghino e causino danni.

INTRODUZIONE

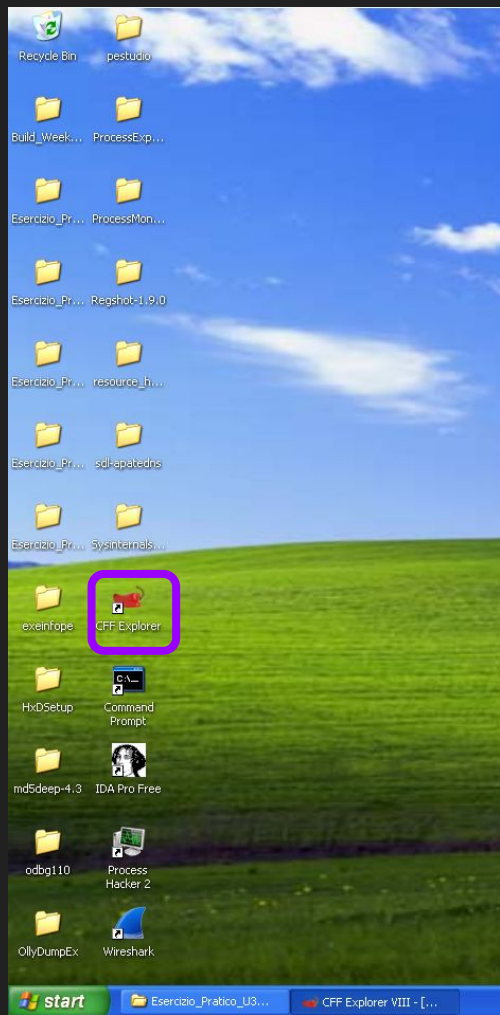
Aprendo **VirtualBox** ci assicuriamo che non ci siano **schede di rete** abilitate e che anche tutti gli altri metodi di ingresso (esempio **USB**) siano **disabilitati**



INTRODUZIONE

Ora che tutto è stato isolato possiamo avviare il software che ci consentirà di analizzare staticamente il malware ovvero **CFF Explorer**, un software che permette di controllare le funzioni importate ed esportate da un malware.

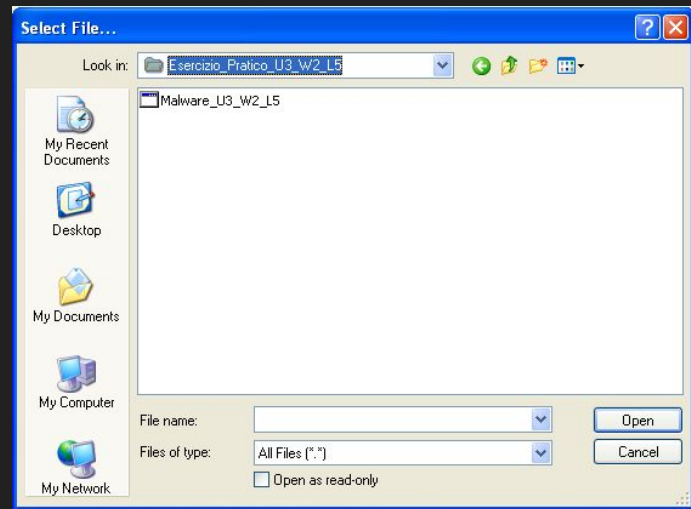
Per avviare il software è sufficiente cliccare due volte sulla sua **icona**



INTRODUZIONE

In alto a Sx cliccando sull'icona della cartella gialla ci comparirà la finestra di selezione del file.

Ora selezioniamo il malware da analizzare



LIBRERIE IMPORTATE

CFF Explorer VIII - [Malware_U3_W2_L5.exe]

File Settings ?

Malware_U3_W2_L5.exe

File: Malware_U3_W2_L5.exe

- Dos Header
- Nt Headers
 - File Header
 - Optional Header
 - Data Directories [x]
- Section Headers [x]
- Import Directory**
- Address Converter
- Dependency Walker
- Hex Editor
- Identifier
- Import Adder
- Quick Disassembler
- Rebuilder
- Resource Editor
- UPX Utility

Module Name	Imports	OFTs	TimeStamp	ForwarderChain	Name RVA	FTs (IAT)
000065EC	N/A	000064DC	000064E0	000064E4	000064E8	000064EC
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.dll	44	00006518	00000000	00000000	000065EC	00006000
WININET.dll	5	000065CC	00000000	00000000	00006664	000060B4

LIBRERIE IMPORTATE

Come mostrato nella figura precedente cliccando a Sx su **Import Directory** verranno mostrate le librerie importate dal malware. In questo caso sono:

Kernel32.dll: libreria piuttosto comune che contiene le funzioni principali per interagire con il sistema operativo, ad esempio: manipolazione dei file, la gestione della memoria.

Wininet.dll: libreria che contiene le funzioni per l'implementazione di alcuni protocolli di rete come HTTP, FTP, NTP.

SEZIONI MALWARE

CFF Explorer VIII - [Malware_U3_W2_L5.exe]

File Settings ?



File: Malware_U3_W2_L5.exe

- File: Malware_U3_W2_L5.exe
 - Dos Header
 - Nt Headers
 - File Header
 - Optional Header
 - Data Directories [x]
 - Section Headers [x]
 - Import Directory
 - Address Converter
 - Dependency Walker
 - Hex Editor
 - Identifier
 - Import Adder
 - Quick Disassembler
 - Rebuilder
 - Resource Editor
 - UPX Utility

Malware_U3_W2_L5.exe

Name	Virtual Size	Virtual Address	Raw Size	Raw Address	Reloc Address	Linenumbers	Relocations ...	Linenumber...	Characteristics
Byte[8]	Dword	Dword	Dword	Dword	Dword	Dword	Word	Word	Dword
.text	00004A78	00001000	00005000	00001000	00000000	00000000	0000	0000	60000020
.rdata	0000095E	00006000	00001000	00006000	00000000	00000000	0000	0000	40000040
.data	00003F08	00007000	00003000	00007000	00000000	00000000	0000	0000	C0000040

SEZIONI MALWARE

Sempre nel menu di Sx troviamo la voce **Section Headers**, qui verranno mostrati invece le sezioni da cui è composto il software malevolo. In questo caso sono:

.text: la sezione «text» contiene le istruzioni (le righe di codice) che la CPU eseguirà una volta che il software sarà avviato. Generalmente questa è l'unica sezione di un file eseguibile che viene eseguita dalla CPU, in quanto tutte le altre sezioni contengono dati o informazioni a supporto.

.rdata: la sezione «rdata» include generalmente le informazioni circa le librerie e le funzioni importate ed esportate dall'eseguibile, informazione che come abbiamo visto possiamo ricavare con CFF Explorer.

.data: la sezione «data» contiene tipicamente i dati / le variabili globali del programma eseguibile, che devono essere disponibili da qualsiasi parte del programma.

COSTRUTTI NOTI

Il **malware** è stato scritto in **linguaggio Assembly**, linguaggio di programmazione a basso livello che fornisce un'interfaccia simbolica per il linguaggio macchina di un computer o di un'altra architettura.

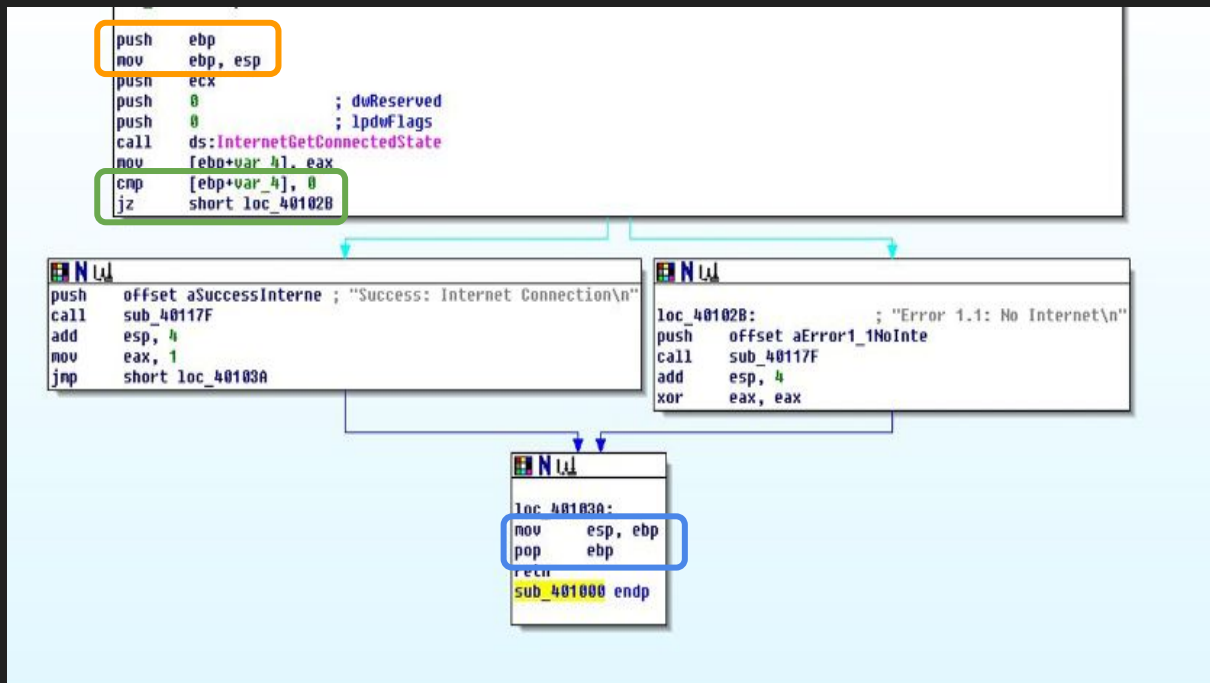
Viene considerato di basso livello perché le sue **istruzioni** sono strettamente **correlate** alle istruzioni della **CPU** e alla struttura del set di istruzioni della macchina.

COSTRUTTI NOTI

Creazione dello stack

Ciclo IF

Rimozione dello stack



FUNZIONALITA' IMPLEMENTATA

Nel codice della slide precedente si nota una scritta in rosa, ovvero **InternetGetConnectedState** questa non è altro che una funzione che ha lo scopo di controllare se sulla macchina è presente una connessione ad internet.

Tramite il ciclo IF sottostante se il risultato della funzione è 0 allora non è presente nessuna connessione internet e la funzione stamperà il messaggio “**Error 1.1: No Internet**”;

Altrimenti se il valore di ritorno della funzione è diverso da 0 vi è connessione e la funzione stampa “**Success: Internet Connection**”.

