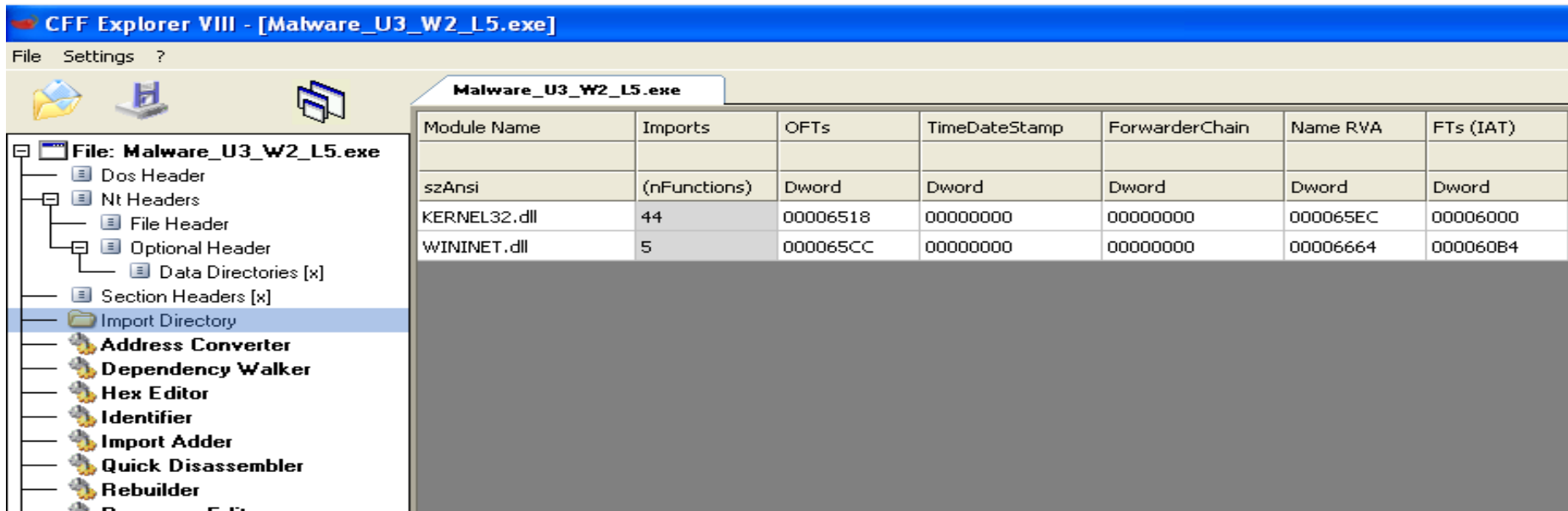


# Malware U3-W2-L5 07/07

1) Cominciamo la nostra analisi statica del malware presente sulla macchina come prima cosa lanciamo CFF controlliamo che il file non sia compresso e andiamo a controllare le librerie importate nella "Import directory".



**CFF Explorer VIII - [Malware\_U3\_W2\_L5.exe]**

File Settings ?

**Malware\_U3\_W2\_L5.exe**

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.dll	44	00006518	00000000	00000000	000065EC	00006000
WININET.dll	5	000065CC	00000000	00000000	00006664	000060B4

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

Dalla come possiamo vedere queste sono le librerie che il malware va ad importare;

KERNEL32.dll è una libreria di windows che contiene una lunga serie funzioni di basso livello che consentono di interagire con il sistema operativo.

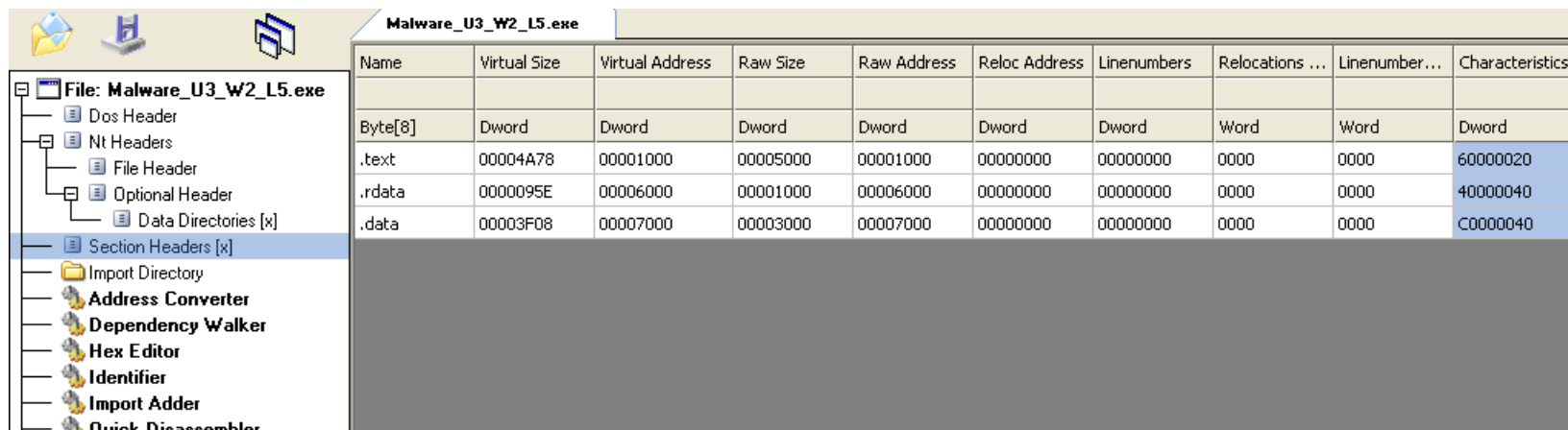
WININET.dll è una libreria che fornisce funzionalità legate all'accesso e alla gestione delle risorse di internet, permette inoltre di eseguire operazioni di gestione delle cache, gestire cookie e certificati digitali

2) Procediamo nel “section haddress” e andiamo ad analizzare le sezioni nella quale troviamo le seguenti:

.text: contiene le righe di codice che la CPU andrà ad eseguire dopo l'avvio del malware.

.rdata: contiene informazioni sulle librerie importate ed esportate.

.data generalmente contiene le variabili globali(utilizzabile da qualsiasi funzione dell'eseguibile).



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

```
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
```

3) cominciamo ad analizzare e riconoscere i costrutti di questo codice l'estratto fa parte della sezione .text

Le prime 2 righe si occupano della creazione di uno stack

Le successive 3 inseriscono tramite push i parametri della funzione e richiama internetGetConnectionState funzione della libreria wininet.

Le ultime 2 eseguono un ciclo if confronta i parametri, se il risultato è uguale a zero salta alla funzione loc\_40102B


A questo punto ci troviamo all'interno della funzione che si occupa di gestire i casi del ciclo if confrontando i parametri.

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

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

- Il primo push è la creazione dello stack
- A questo tramite “call”viene richiamata la subroutine 40117F
- a questo punto esegue la pulizia della stack
- per infine effettuare un salto condizionale alla subroutine 40103A

- Loc\_40102B è un’ancora ovvero un punto di riferimento nel nell’esecuzione del programma
- push crea uno stack
- richiama la subroutine 40117F
- esegue la pulizia dello stack
- esegue una operazione Xor impostando a 0 il valore di eax



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

Loc è ancora un punto di riferimento/ancora

-pop elimina lo stack

-retn è il ritorno della funzione

-endp fine della funzione sub\_401000

Il codice verifica lo stato della connessione a internet e mostra un messaggio di successo se la connessione è attivo altrimenti mostra un messaggio di errore.

Questo riesce a farlo probabilmente `internetGetConnectionState` funzione della libreria `wininet` responsabile della gestione delle risorse di rete del sistema.