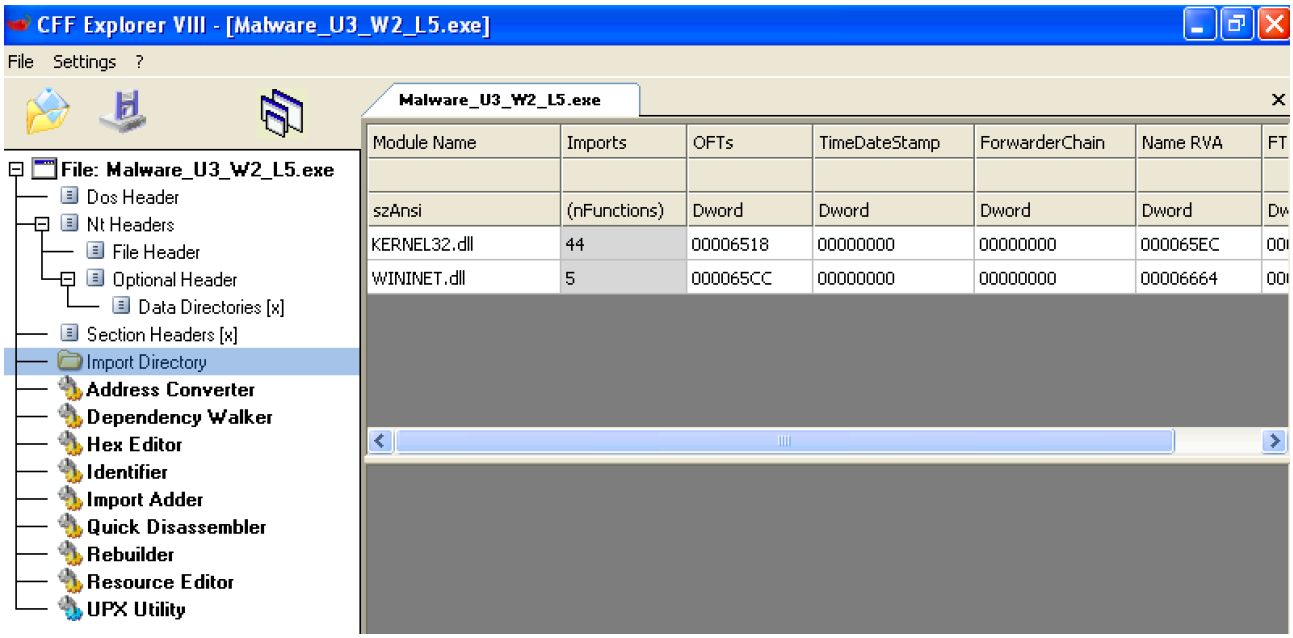
ESERCIZIO WEEK 10
ANALISI DI UN MALWARE

Riferimento malware_U3_W2_L5

- **Librerie importate**



Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FT
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dw
KERNEL32.dll	44	00006518	00000000	00000000	000065EC	00
WININET.dll	5	000065CC	00000000	00000000	00006664	00

Con un analisi attraverso CFF Explorer possiamo vedere che le librerie importate dal malware sono:

KERNEL32.DLL: libreria piuttosto comune che contiene le funzioni principali per interagire con il sistema operativo, ad esempio: manipolazione dei file e gestione della memoria
Chiamate alle funzioni:

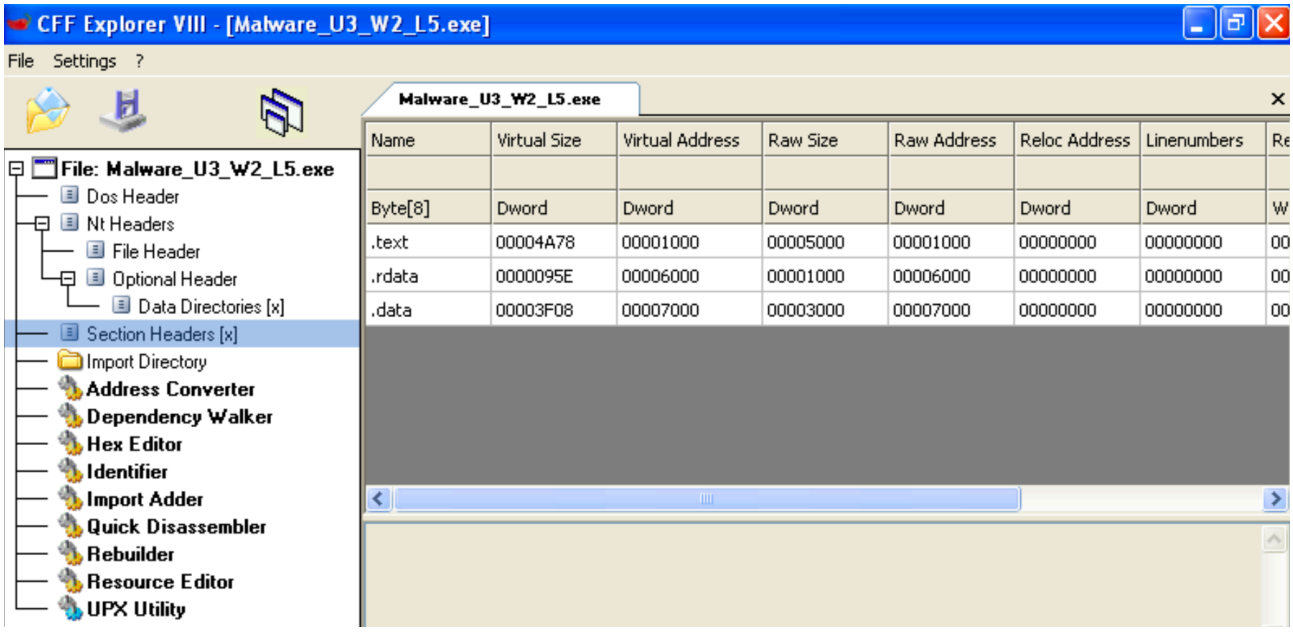
OFTs	FTs (IAT)	Hint	Name
00006568	00006050	00006786	00006788
Dword	Dword	Word	szAnsi
000065E4	000065E4	0296	Sleep
00006940	00006940	027C	SetStdHandle
0000692E	0000692E	0156	GetStringTypeW
0000691C	0000691C	0153	GetStringTypeA
0000690C	0000690C	01C0	LCMapStringW
000068FC	000068FC	01BF	LCMapStringA
000068E6	000068E6	01E4	MultiByteToWideChar
00006670	00006670	00CA	GetCommandLineA
00006682	00006682	0174	GetVersion
00006690	00006690	007D	ExitProcess
0000669E	0000669E	029E	TerminateProcess
000066B2	000066B2	00F7	GetCurrentProcess
000066C6	000066C6	02AD	UnhandledExceptionFilter
000066E2	000066E2	0124	GetModuleFileNameA
000066F8	000066F8	00B2	FreeEnvironmentStringsA
00006712	00006712	00B3	FreeEnvironmentStringsW
0000672C	0000672C	02D2	WideCharToMultiByte
00006742	00006742	0106	GetEnvironmentStrings
0000675A	0000675A	0108	GetEnvironmentStringsW
00006774	00006774	026D	SetHandleCount
00006786	00006786	0152	GetStdHandle

00006796	00006796	0115	GetFileType
000067A4	000067A4	0150	GetStartupInfoA
000067B6	000067B6	0126	GetModuleHandleA
000067CA	000067CA	0109	GetEnvironmentVariableA
000067E4	000067E4	0175	GetVersionExA
000067F4	000067F4	019D	HeapDestroy
00006802	00006802	019B	HeapCreate
00006810	00006810	02BF	VirtualFree
0000681E	0000681E	019F	HeapFree
0000682A	0000682A	022F	RtlUnwind
00006836	00006836	02DF	WriteFile
00006842	00006842	0199	HeapAlloc
0000684E	0000684E	00BF	GetCPInfo
0000685A	0000685A	00B9	GetACP
00006864	00006864	0131	GetOEMCP
00006870	00006870	02BB	VirtualAlloc
00006880	00006880	01A2	HeapReAlloc
0000688E	0000688E	013E	GetProcAddress
000068A0	000068A0	01C2	LoadLibraryA
000068B0	000068B0	011A	GetLastError
000068C0	000068C0	00AA	FlushFileBuffers
000068D4	000068D4	026A	SetFilePointer
00006950	00006950	001B	CloseHandle

WININET.DLL: libreria che contiene le funzioni per l'implementazione di alcuni protocolli di rete come HTTP FTP NTP
Chiamate alle funzioni:

OFTs	FTs (IAT)	Hint	Name
Dword	Dword	Word	szAnsi
00006640	00006640	0071	InternetOpenUrlA
0000662A	0000662A	0056	InternetCloseHandle
00006616	00006616	0077	InternetReadFile
000065FA	000065FA	0066	InternetGetConnectedState
00006654	00006654	006F	InternetOpenA

- **Sezioni di cui si compone il malware**



Nella scheda ‘section headers’ possiamo vedere le funzioni di cui si compone il malware che in questo caso sono:

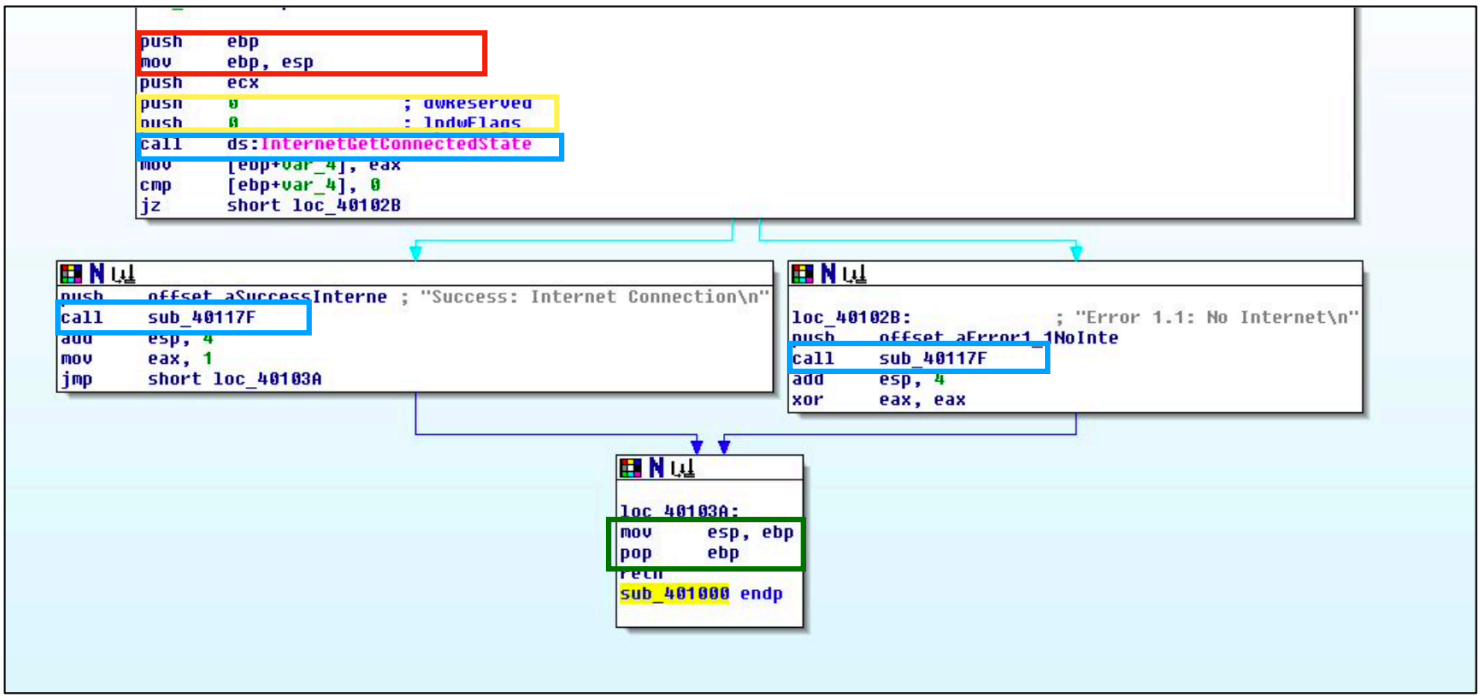
.text : la sezione «text» contiene le istruzioni che la CPU eseguirà una volta che il software sarà avviato. Generalmente questa è l’unica sezione di un file eseguibile che viene eseguita dalla CPU, in quanto tutte le altre sezioni contengono dati o informazioni a supporto.

.rdata : include generalmente le informazioni circa le librerie e le funzioni importate ed esportate dall’eseguibile

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

ANALISI CODICE

- **Identificazione costrutti noti**



Con riferimento al codice in figura possiamo identificare
In rosso la creazione dello stack
In verde la chiusura dello stack
In giallo il passaggio dei parametri alla funzione
In blu la chiamata della funzione

Il costrutto identificato da questa porzione di codice sembra essere un Ciclo IF per le condizioni di avvenuta connessione ad internet

Da questa porzione di codice possiamo ipotizzare che il malware in questione stia tentando di stabilire una connessione internet, potrebbe dunque trattarsi un downloader o di un malware che cerca di creare una backdoor per un controllo remoto

BONUS
Un giovane dipendente neo assunto segnala al reparto tecnico la presenza di un programma sospetto.
Il suo superiore gli dice di stare tranquillo ma lui non è soddisfatto e chiede supporto al SOC.

Il file "sospetto" è IEXPLORE.EXE contenuto nella cartella C:\Program Files\Internet Explorer.

Come membro senior del SOC ti è richiesto di convincere il dipendente che il file non è maligno.

Possono essere usati gli strumenti di analisi statica basica e/o analisi dinamica basica visti a lezione.
No disassembly no debug o similari

Per convincere il dipendente che IEXPLORE.EXE sia un file legittimo possiamo cominciare con alcune tecniche di analisi statica basica

- possiamo analizzare il file in questione attraverso strumenti come virustotal o simili. Possiamo attuare questa procedura sia caricando il file o possiamo farlo calcolando l'hash attraverso strumenti come md5deep
Ricordiamo che l'hash di un eseguibile è una stringa alfanumerica unica per identificarlo.

- Recupero delle stringhe: con lo strumento strings da riga di comando possiamo recuperare le stringhe contenute all'interno del file controllando che non ci siano messaggi 'sospetti'

In seguito possiamo procede con l'analisi dinamica basica
Per fare questo abbiamo bisogno di creare un ambiente di lavoro isolato e controllato potremmo quindi procedere ad isolare dalla rete la macchina in questione ed eventualmente controllare l'accesso delle periferiche esterne prima di eseguire quello che potrebbe essere il file sospetto
Una volta avviato l'eseguibile possiamo usare diversi strumenti per controllare cosa accade

- Procmon : una volta avviato l'eseguibile procmon ci aiuterà a capire gli eventi in corso come le attività relative ai registri, al filesystem, ai flussi di rete, processi e thread o al tempo di utilizzo e filtrando per nome possiamo identificare quelli relativi al file di interesse

- Regshot : il tool in questione ci permette di scattare, prima e dopo l'esecuzione del file, delle istantanee relative alle chiavi di registro del sistema windows in modo da poterle successivamente confrontare e constare se ci sono state delle modifiche o delle relative aggiunte

- Apatedns: questo tool ci permette di simulare un server DNS in modo da intercettare tutte le richieste effettuate verso domini esterni ed eventualmente analizzarle

- Wiresharck: questo tool ci permette di monitorare tutto il traffico di rete che passa attraverso la nostra macchina in modo da individuare richieste sospette

Una volta concluse queste fasi di analisi possiamo analizzare i risultati ed avendo appurato che il file IEXPLORE.EXE contenuto nella cartella C:\Program Files\Internet Explorer non ha evidenziato alcun ‘comportamento’ sospetto possiamo dire al dipendente che si tratta di un browser web legittimo e che non crea alcun pericolo per la sicurezza informatica della nostra macchina.