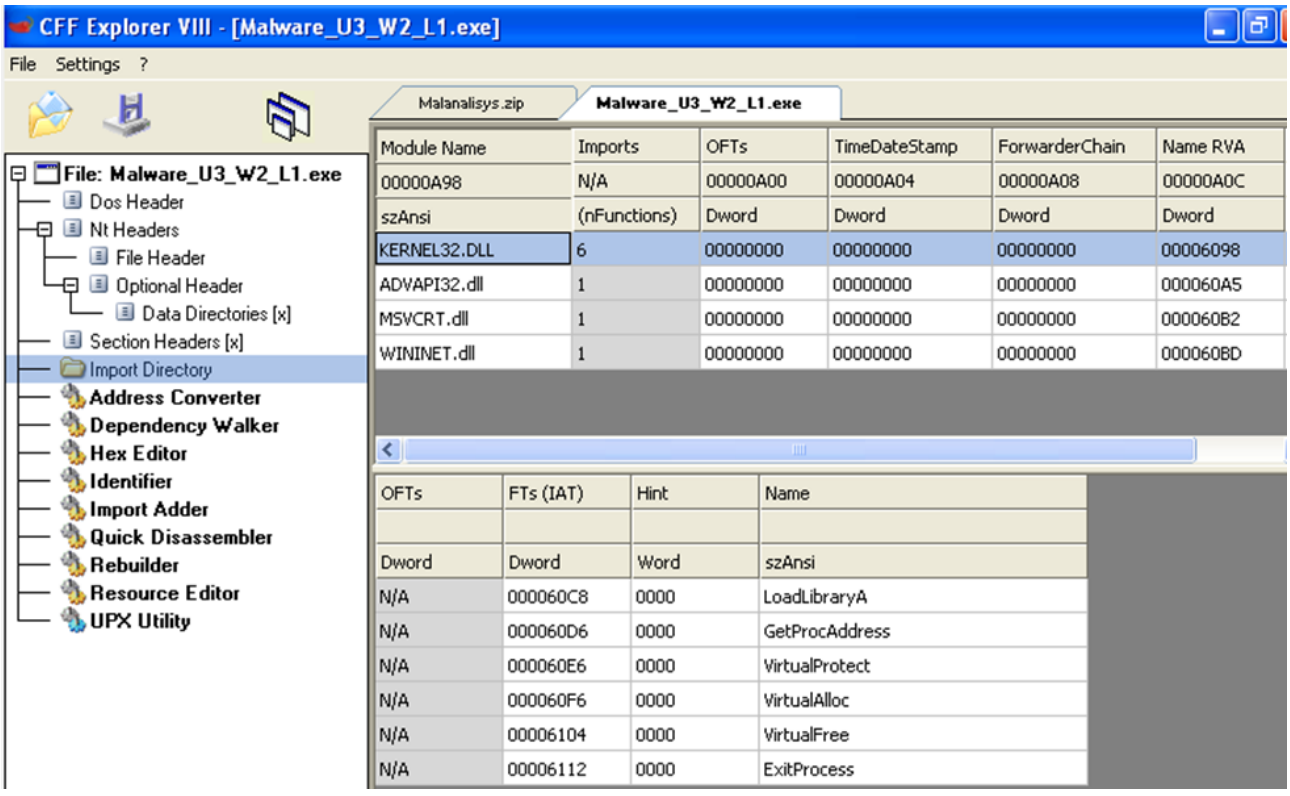
Epicode Unit 3 Week 2

Static Malware Analysis

Nell'esercitazione odierna è richiesta l'analisi statica basica di un potenziale Malware per Windows XP. Come richiesto, utilizzando il tool "CFF Explorer", si procede dapprima con lo studio delle librerie importate dal Malware tramite la sezione "Import Directory" del tool.

Malware_U3_W2_L1.exe

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.DLL	6	00000000	00000000	00000000	00006098	00006064
ADVAPI32.dll	1	00000000	00000000	00000000	000060A5	00006080
MSVCRT.dll	1	00000000	00000000	00000000	000060B2	00006088
WININET.dll	1	00000000	00000000	00000000	000060BD	00006090



Come si può notare, il programma analizzato va ad importare quattro librerie per l'esecuzione dello stesso:

- **La libreria "KERNEL32.DLL", praticamente fondamentale per permettere al programma di interagire con le principali funzioni del Sistema Operativo,**

quali manipolazioni dei file, gestione della memoria e dei privilegi amministrativi;
- La libreria "ADVAPI32.dll", con cui il Malware interagisce con i registri ed i servizi del Sistema Operativo Microsft;
- La libreria "MSVCRT.dll", che contiene al suo interno le funzioni per la gestione e manipolazione delle stringhe; la gestione, manipolazione e allocazione della memoria; chiamate di input/output etc;
- La libreria "WININET.dll", che come si può dedurre dal nome si pccupa di servizi di rete quali HTTP, FTP ed altri.

Da questa prima analisi, seppur non avendo la sicurezza di come il programma si comporti nel dettaglio, possiamo iniziare ad ipotizzare che esso vada ad interagire con il Kernel della macchina e a sfruttare la connessione Internet della stessa per ricevere input da remoto o inviare output all'esterno.

Successivamente, andando ad indagare nella sezione "Section Headers" del tool, si può verificare in quante macrosezioni sia diviso il Malware e comprenderne più a fondo il comportamento (seppur sempre grossolanamente) in base all'output restituito da CFF EXP.

Malware_U3_W2_L1.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
UPX0	00004000	00001000	00000000	00000400	00000000	00000000	0000	0000	E0000080
UPX1	00001000	00005000	00000600	00000400	00000000	00000000	0000	0000	E0000040
UPX2	00001000	00006000	00000200	00000A00	00000000	00000000	0000	0000	C0000040

Si nota immediatamente che la formattazione del Programma è stata manomessa con UPX, un "packer" specificamente utilizzato per modificare/riorganizzare la struttura di codici.

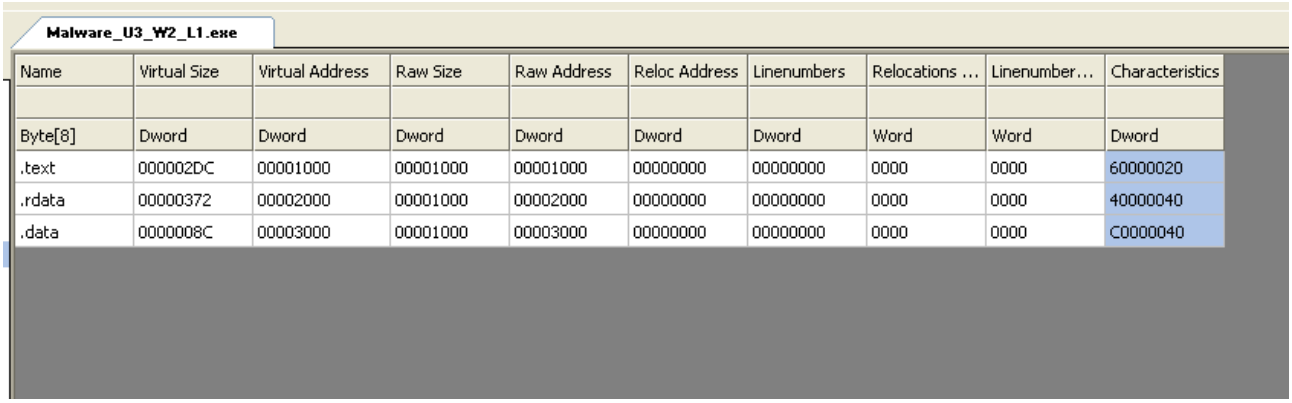
Packer come UPX sono generalmente utilizzati per criptare e mascherare la struttura del codice nel tentativo di rendere più difficile per l'analista capire cosa sta succedendo. Gli autori di malware utilizzano UPX e un programma di compressione secondario, spesso personalizzato, per evitare ulteriormente che il malware possa essere individuato da Software AV.

Alcuni packer, inoltre, sono spesso inseriti nelle blacklists dei software AV, come ad esempio Themida: questi vengono rilevati automaticamente come malware a causa della firma del packer inserito nella blacklist.

UPX, al contrario, non è quasi mai in blacklist, motivo per cui non solo è sempre più popolare, ma funziona quasi sempre comprimendo le sezioni memorizzate all'interno della tabella delle sezioni del file PE.

Un forte indicatore dell'utilizzo di UPX è la ridenominazione dei nomi delle intestazioni (UPX0/UPX1 come nel nostro caso), con lo scopo principale di ridurre le dimensioni del file, aiutando a mascherare il malware come .jpg o a diffondersi tramite e-mail, per fare un paio di esempi.

Si può utilizzare a questo punto la funzione di "Unpack" di CFF Explorer per andare a verificare cosa si celi dietro alle tre sezioni "impacchettate".

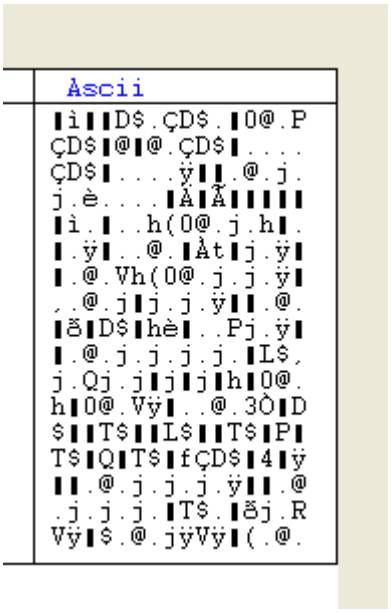


Malware_U3_W2_L1.exe

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	000002DC	00001000	00001000	00001000	00000000	00000000	0000	0000	60000020
.rdata	00000372	00002000	00001000	00002000	00000000	00000000	0000	0000	40000040
.data	0000008C	00003000	00001000	00003000	00000000	00000000	0000	0000	C0000040

Nello specifico, il programma si divide in:

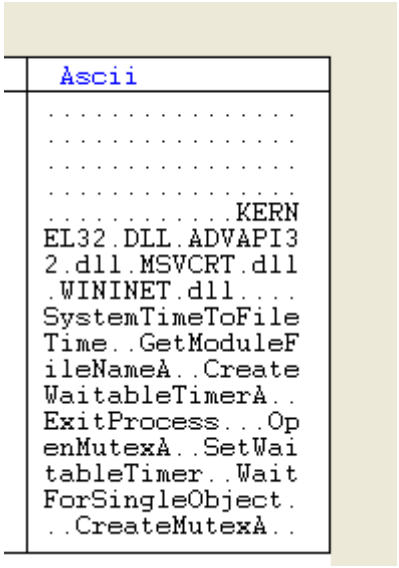
- **".text", che contiene il codice effettivo del programma e che verrà poi eseguito dalla CPU;**



Ascii

▌ì▌▌D$.ÇD$.▌0@.P
ÇD$▌@▌@.ÇD$▌....
ÇD$▌....ÿ▌▌.@.j.
j.è....▌Å▌Å▌▌▌▌
▌ì.▌..h(0@.j.h▌.
▌.ÿ▌..@.▌Åt▌j.ÿ▌
▌.@.Vh(0@.j.j.ÿ▌
..@.j▌j.j.ÿ▌▌.@.
▌ð▌D$▌hè▌..Pj.ÿ▌
▌.@.j.j.j.j.▌L$.
j.Qj.j▌j▌j▌h▌0@.
h▌0@.Vÿ▌..@.3Ò▌D
$▌▌T$▌▌L$▌▌T$▌P▌
T$▌Q▌T$▌fÇD$▌4▌ÿ
▌▌.@.j.j.j.ÿ▌▌.@
.j.j.j.▌T$.▌ðj.R
Vÿ▌$.@.jÿVÿ▌(.@.

- **".rdata", che contiene le librerie importate e che è stata analizzata dal tool;**



- **“data”, che contiene i dati e le variabili globali dell’eseguibile.**

Alla luce di quanto analizzato, si può ipotizzare che il malware in questione vada a sfruttare connessioni internet per inviare dati/attendere input da un utente remoto. In particolare le librerie e le funzioni importate di conseguenza mi fanno pensare ad un un Trojan (con conseguente backdoor) o qualche malware che collezioni dati come uno Spyware o un Keylogger. Specie per quest’ultima ipotesi, comunque, non si dispone di sufficienti informazioni.