Nell’esercizio di oggi ci viene chiesto di analizzare un malware; in particolare:

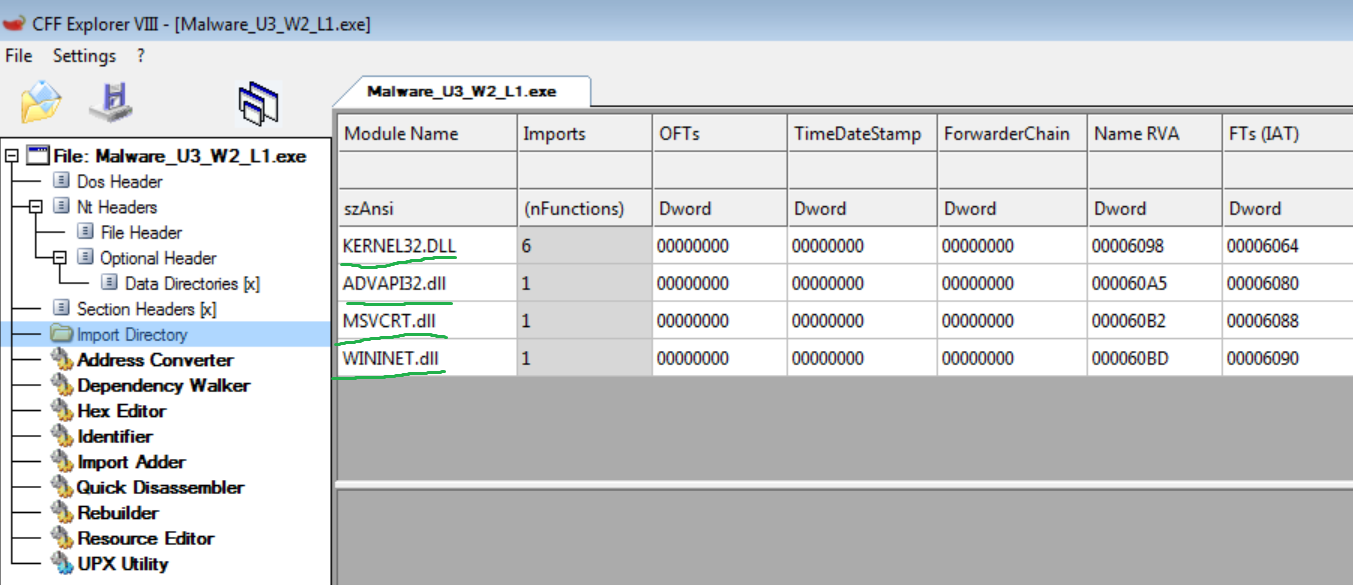
-Indicare le librerie importate dal malware, fornendo una descrizione per ognuna di esse

-Indicare le sezioni di cui si compone il malware, fornendo una descrizione per ognuna di essa

**LIBRERIE IMPORATE DAL MALWARE**

Apriamo, dunque, “CFF explorer”, particolarmente adatto per analizzare nel dettaglio la struttura interna dei file PE. Apriamo il malware e cominciamo ad analizzarlo. La maggior parte dei malware esegue **Dinamicamente** le librerie all’interno del suo codice (ovvero le librerie non vengono importate all’interno del codice ma vengono richiamate solo quando il file è in esecuzione); in questo modo sarà più facile, per i malware, aggirare controlli come antivirus.

Ci spostiamo nella sezione “import Directory per comprendere se il malware importa librerie remote.



In effetti ci sono 4 librerie sospette; queste, in particolare hanno lo scopo di:

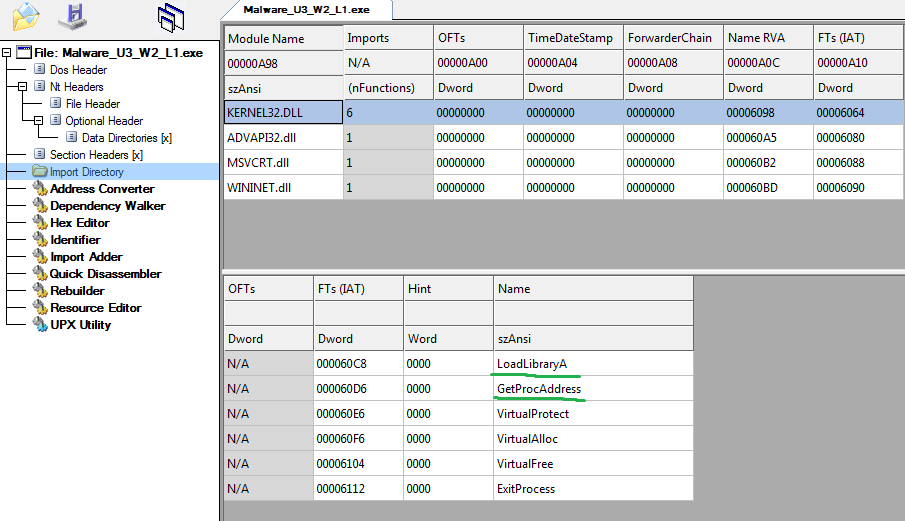
-**Kernel32.dll**: interagisce con il sistema operativo e gestisce (anche impropriamente) la memoria e le risorse del sistema operativo.

-**Advapi32.dll:** contiene informazioni per manipolare il sistema operativo

-**MSVCRT.dll:** Contiene informazioni sulla manipolazione di stringhe e allocazione di memoria

-**Wininet.dll**: permette alle applicazioni di interagire con protocolli come HTTP, HTTPS ed FTP.

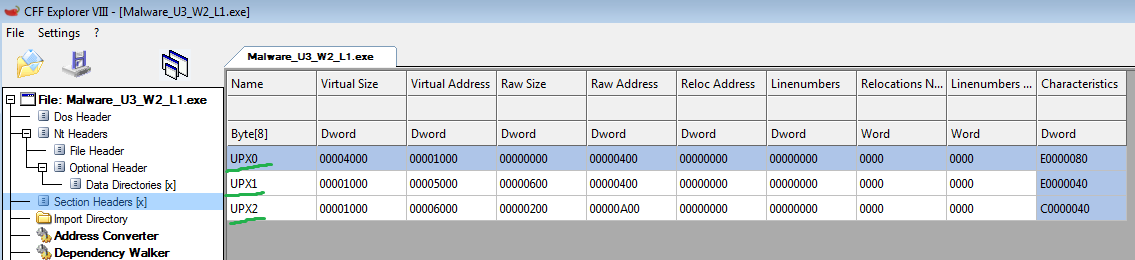
Inoltre, analizzando le librerie, ci accorgiamo che in esse sono presenti le funzioni “**LoadLibrary**” e “**GetProcAddress**”.



La presenza di questi file dovrebbe quantomeno allertarci, poiché si tratta proprio delle funzioni utilizzate per richiamare librerie da remoto (cioè dinamicamente) ed aggirare software anti-malware.

**SEZIONI DEL MALWARE**

Nella sezione “Section Headers” analizziamo da quali parti è composto il malware.



Il malware sembra essere composto da 3 sezioni: **UPX0, UPX1, UPX2.** Generalmente, si passa ad analizzare le sezioni del malwareper rilevare file di comportamento del programma; in particolare:

-“**.text**”: contiene informazioni sulla parte di memoria che verrà usata durante l’esecuzione del programma

-“**.rdata**”: contiene informazioni sulle librerie importate ed esportate del malware

-“**.data**”: contiene informazioni sulle variabili globali del sistema

-“.**rsrc**”: include informazioni sui file non eseguibili(come foto e video) utilizzati dal malware

Sfortunatamente, però non ci vengono restituite ulteriori informazioni poiché il programma riesce a camuffare le sezioni di cui si compone.

**CONCLUSIONI**

Per concludere, possiamo dire di trovarci di fronte ad un malware che proprio a causa del suo livello di sofisticatezza ci impedisce di comprendere appieno come funziona solo tramite un’analisi statica basica; bisognerà, pertanto, passare ad effettuare ulteriori analisi.