PROGETTO S10_L5

Poggiali Michael

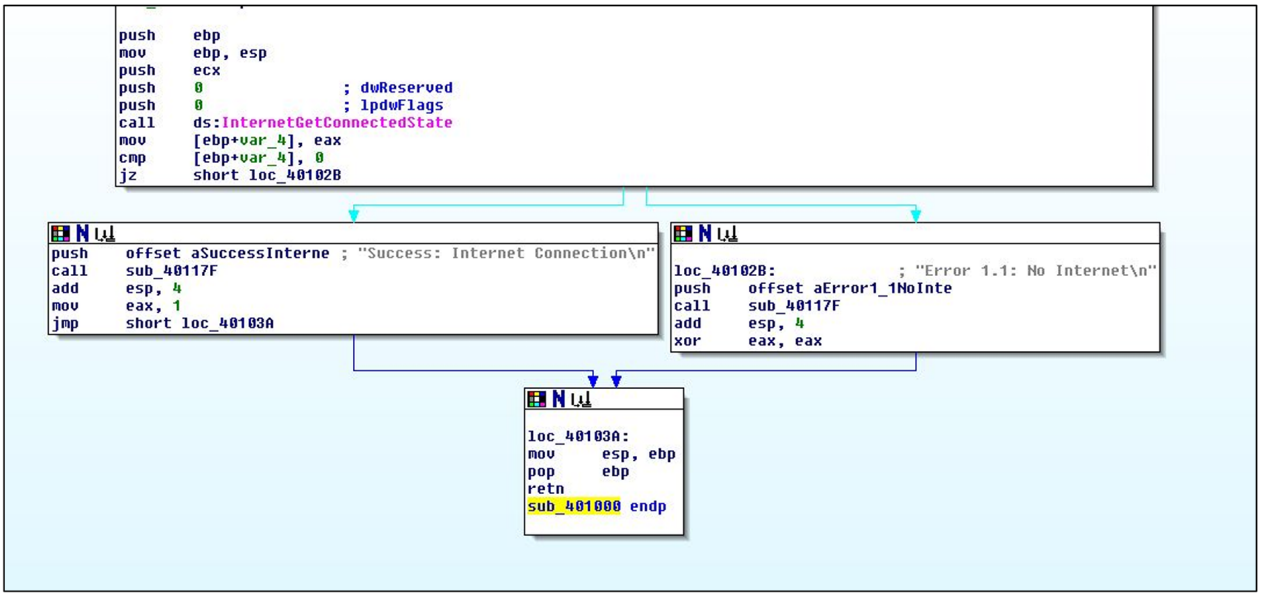
Malware Analysis

L’esercizio odierno chiede di rispondere a dei quesiti relativi ad un malware nominato “Malware_U3_W2_L5” presente su una macchina virtuale Windows XP creata come laboratorio di test, le domande sono:

- Quali librerie vengono importate dal file eseguibile?
- Quali sono le sezioni di cui si compone il file eseguibile del malware?

Inoltre in riferimento alla figura 1 seguente:

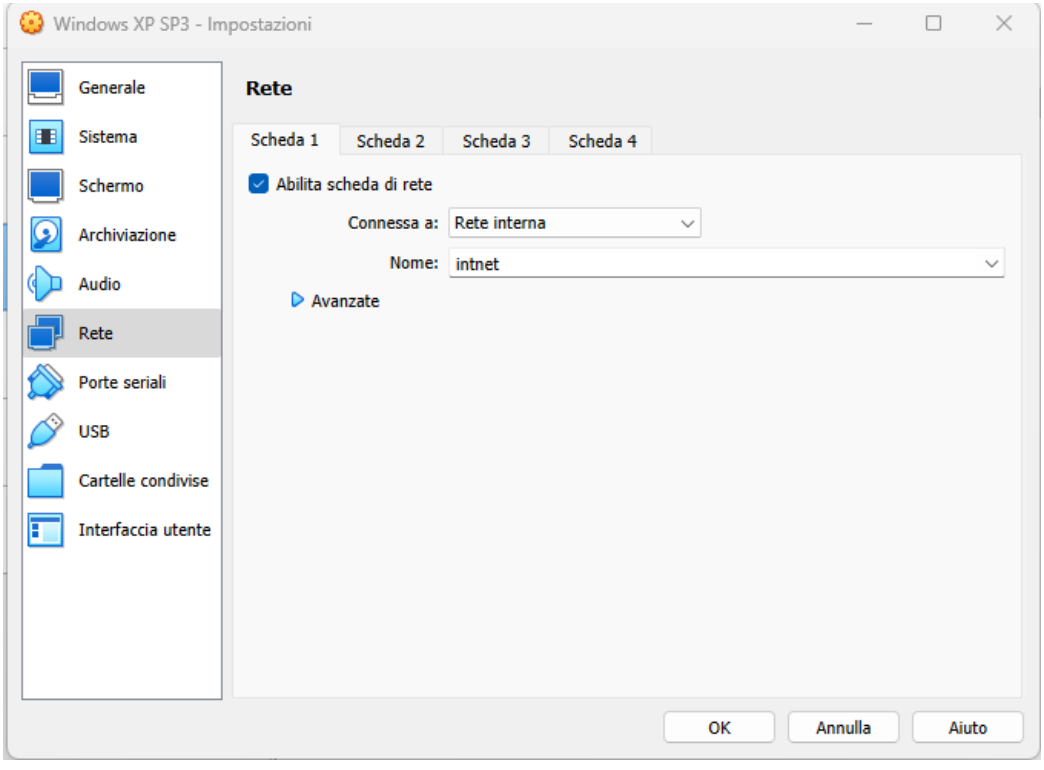
Figura 1

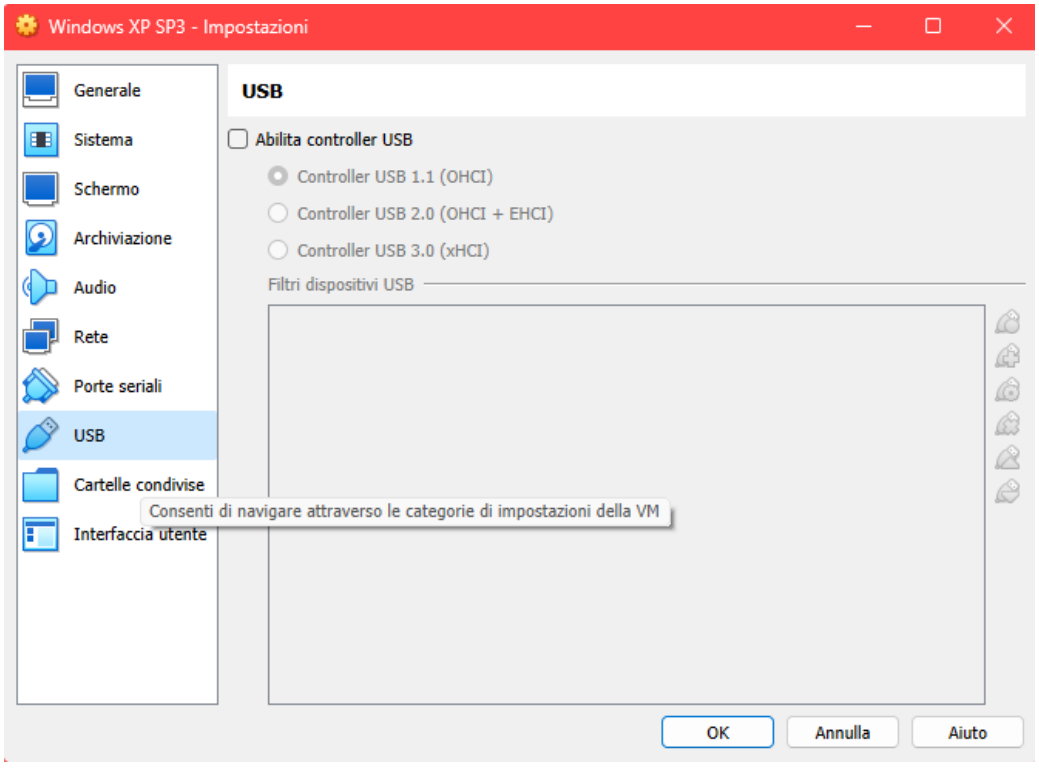


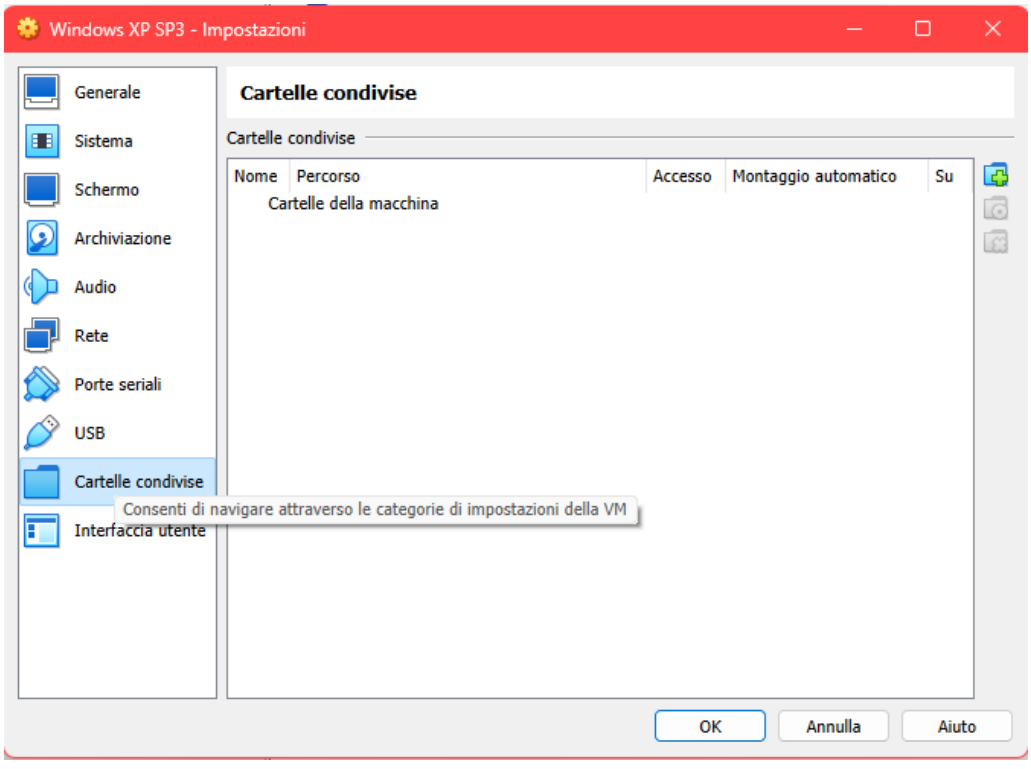
- Identificare i costrutti noti;
- Ipotizzare il comportamento della funzionalità implementata.

Inizio impostando la macchina virtuale su rete interna, per avere un scheda di rete attiva ma non connessa, così da poter visionare eventuali tentativi di connessione ad internet da parte del malware, e disattivando i controller USB e le cartelle condivise, in modo da

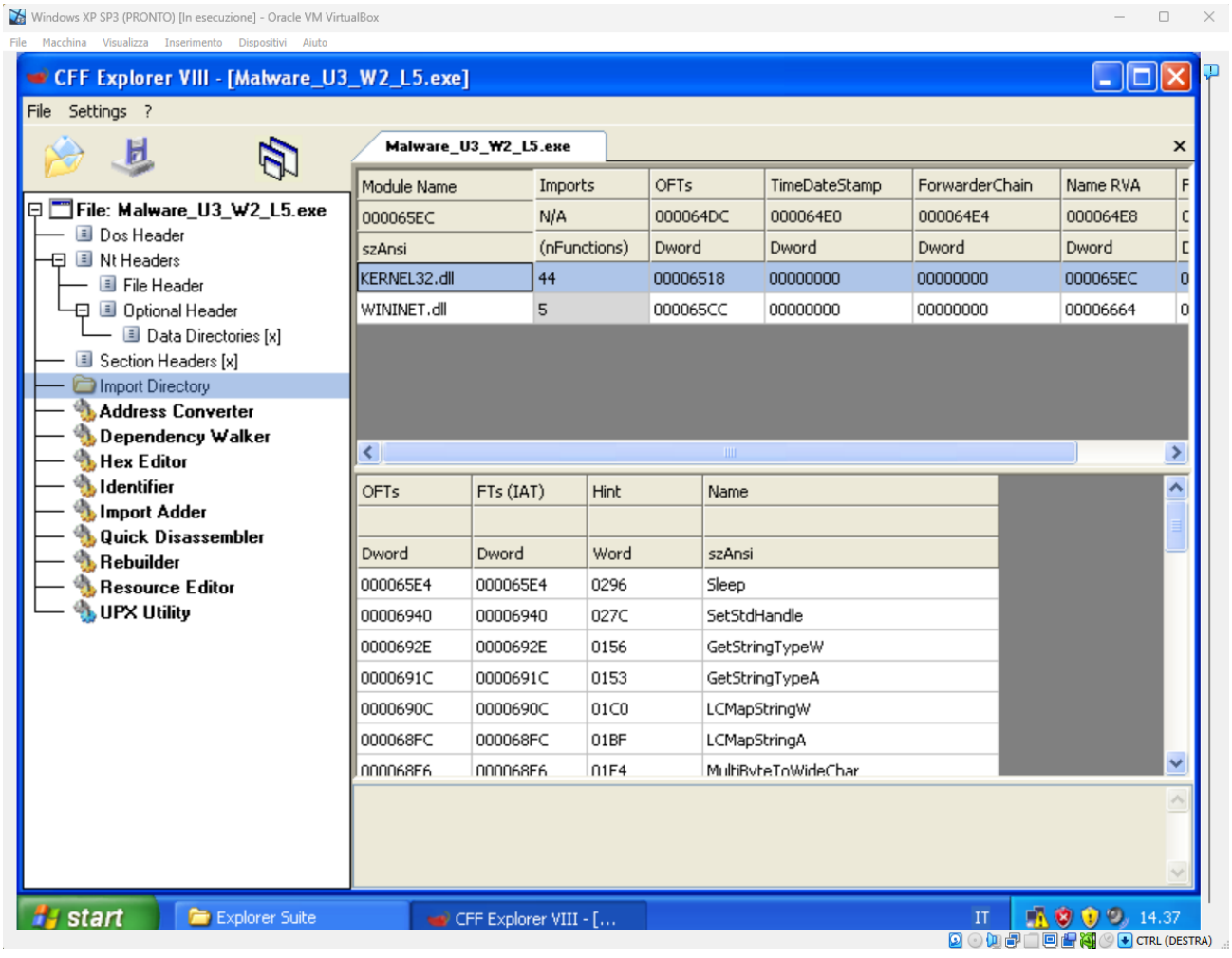
isolare totalmente il laboratorio onde evitare la propagazione involontaria del software malevolo sul mio dispositivo fisico.



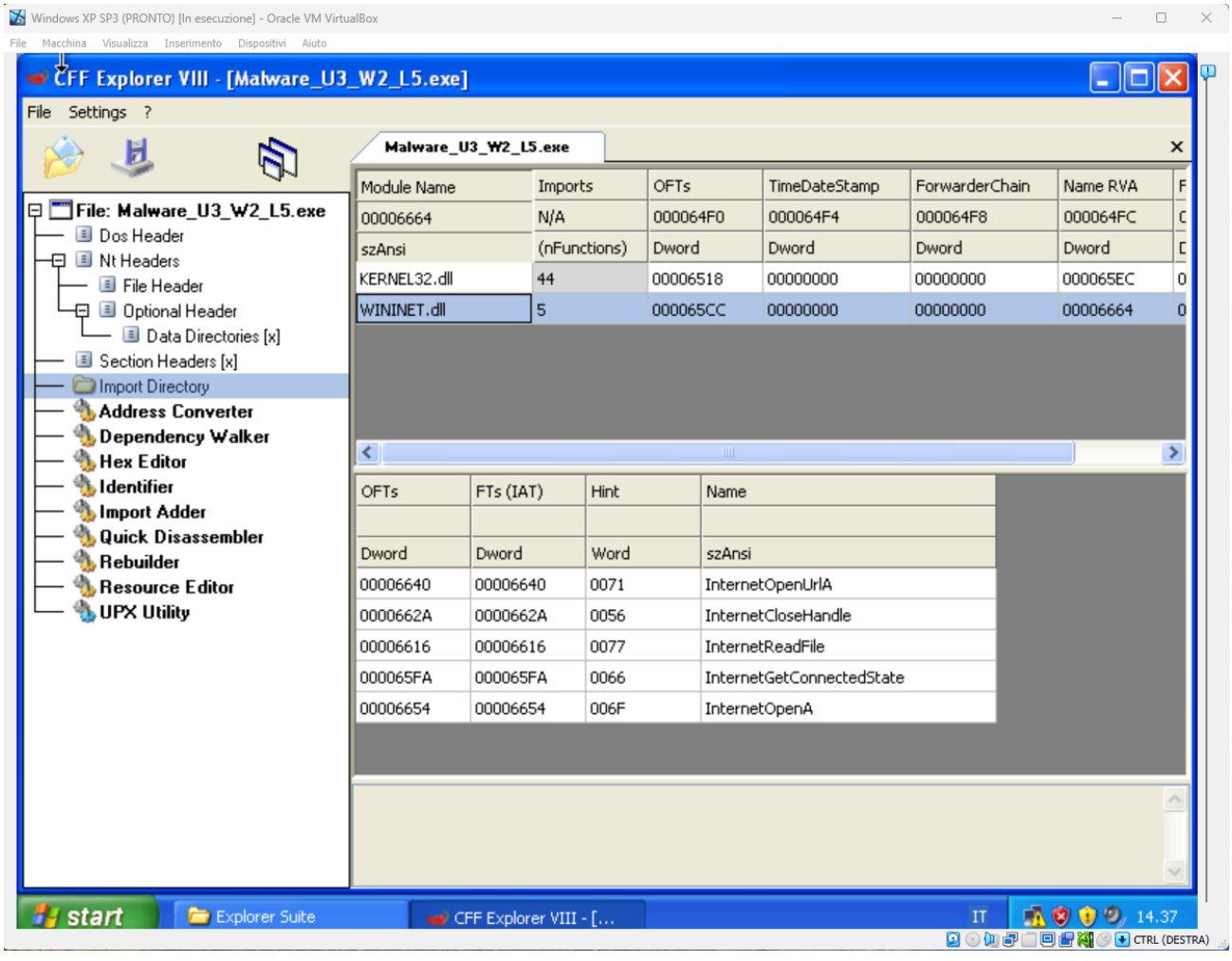




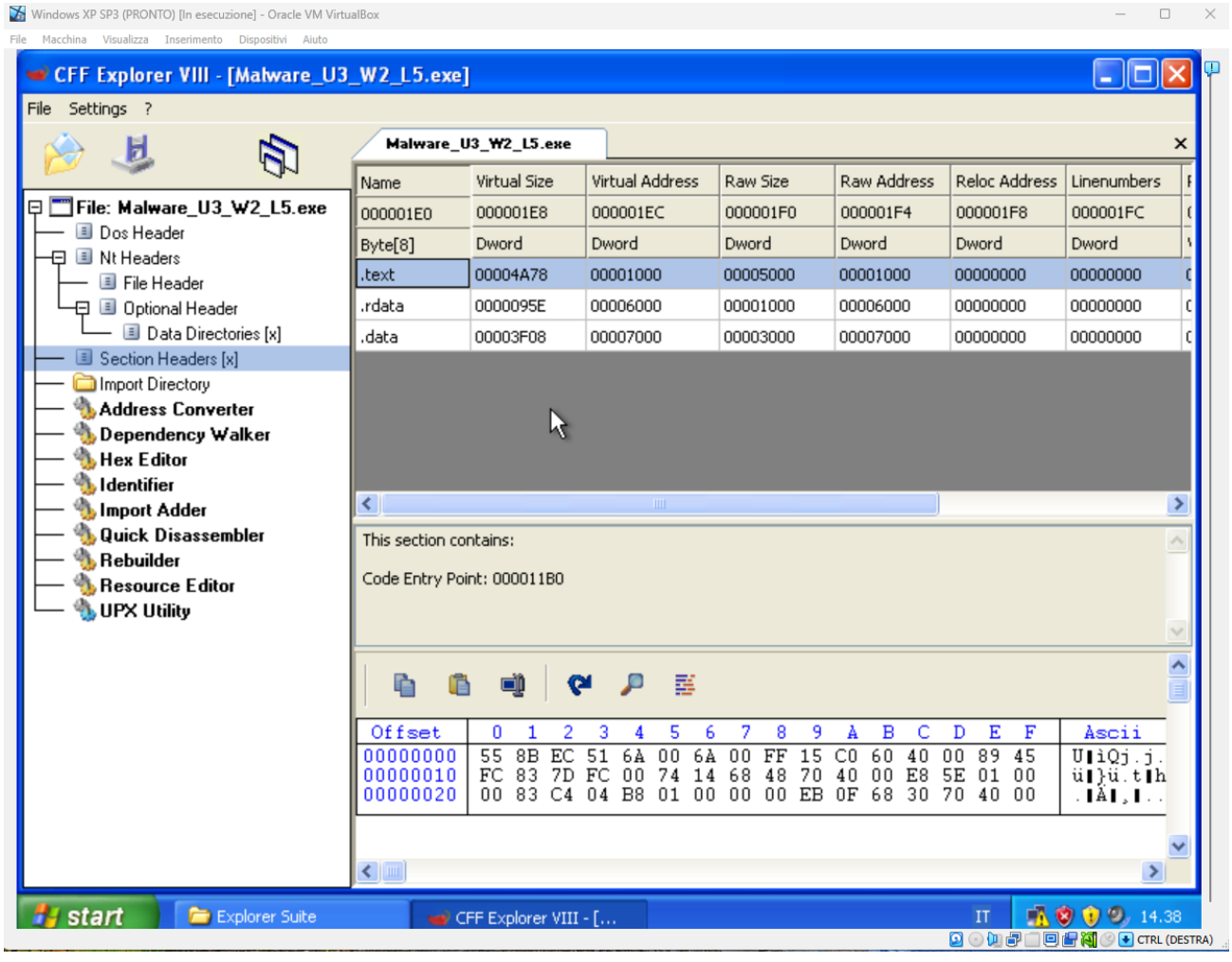
Per controllare quali sono le librerie importate dal file eseguibile malevolo (ovvero i comandi preimpostati che servono al file per eseguire le istruzioni del codice), ed anche le relative sezioni di cui si compone (ossia le parti di codice adibite ad un determinato scopo), utilizzo un tool di analisi statica basica chiamato “CFF Explorer”:



La prima libreria importata che notiamo, dal path a sinistra “Import Directory” è, nella finestra in alto a destra, “KERNEL32.dll” che contiene le principali funzioni (in basso a destra) necessarie ad interagire con il sistema operativo, come ad esempio la gestione della memoria del dispositivo e la copia/creazione/cancellazione dei file.



La seconda libreria è “WININET.dll” che contiene le funzioni per alcuni dei protocolli di rete, come ad esempio HTTP.

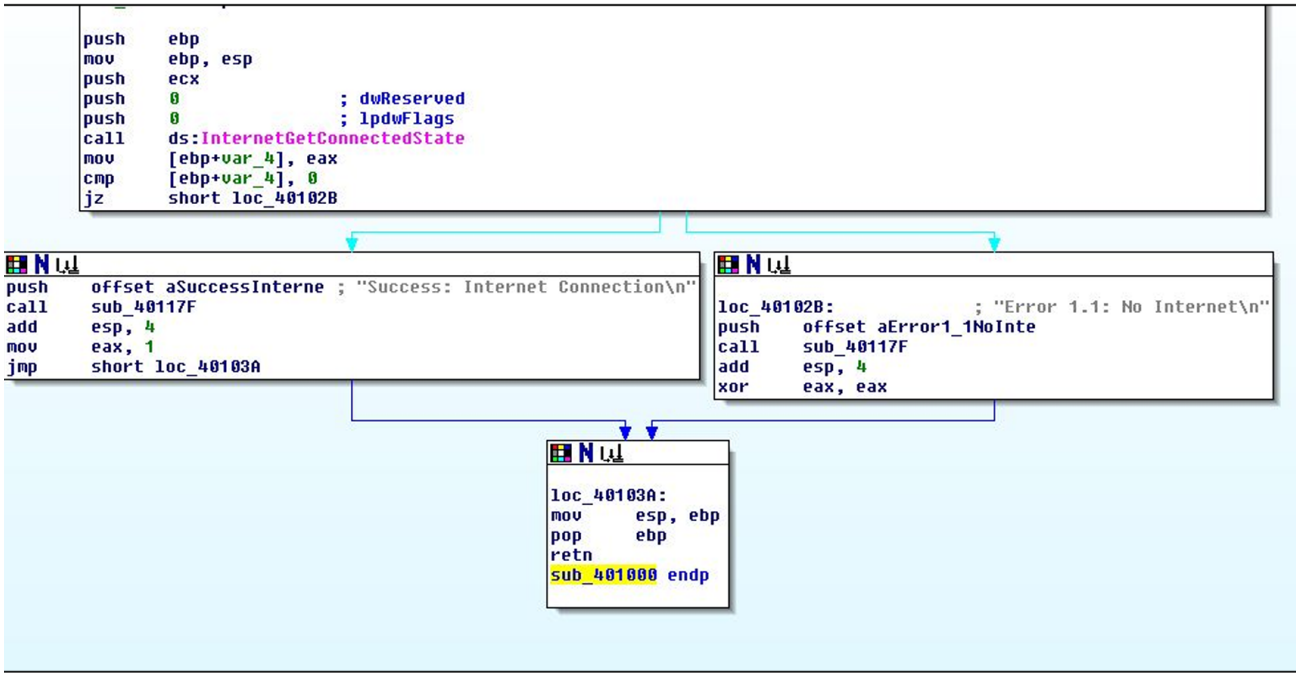


Per quanto riguarda le sezioni che compongono il file, dal path “Section Headers” a sinistra, ne notiamo 3, ovvero .text, .rdata e .data:

- .text contiene le istruzioni per la CPU, la quale le eseguirà all’avvio del software;
- .rdata contiene le informazioni di librerie e funzioni importate ed esportate dal software;
- .data contiene dati e variabili globali del software. Per globali si intende accessibili in ogni parte del programma quindi non solo per singole funzioni.

Riporto per comodità la figura 1 in basso:

1



Da un'analisi del diagramma di flusso in alto si evince che si tratta di sezioni del codice in linguaggio assembly (quindi si tratta di analisi statica avanzata) ossia un linguaggio di basso livello (vicino al codice macchina) che si utilizza per il "reverse engineering" (denominazione del processo di ricostruzione delle funzionalità di un software tramite l'osservazione del suo codice). Il programma che ci permette di "tradurre" le istruzioni binarie eseguite dalla CPU in linguaggio assembly (per permettere la comprensione da parte dell'analista) si chiama "Disassembler".

Si notano, tra i 4 costrutti, in alto la creazione di uno stack "ebp", una chiamata ad una funzione "InternetGetConnecterState" (che serve ad indicare se il sistema è connesso o meno ad internet), ed un "ciclo if" che, in caso di mancanza di connessione "salta" alla locazione di memoria "40102B" (costrutto a destra) che a causa della mancanza di connessione chiama la funzione in memoria "40117F" (costrutto a sinistra) dove si nota che in caso di connessione attiva, o nel caso in cui il "ciclo if" indichi la presenza di connessione ad un determinato ciclo di controllo, esegue un salto sulla memoria "40103A" (costrutto in basso) per eliminare lo stack e terminare la procedura.

Avendo visionato questi dati, si può resocontare che il software analizzato si avvii assieme al sistema operativo e vada a tentare la connessione ad internet, quindi si può ipotizzare che si tratti di una backdoor (che crea una connessione verso un server malevolo controllato da un blackhat consentendo l'accesso al sistema da remoto) o di un downloader (un malware di piccole dimensioni che si nasconde bene nel sistema e si tenta di connettere ad internet per scaricare altri malware più grandi e pericolosi) o anche di uno spyware (utilizzato per rubare e spiare i dati che l'utente immette nel dispositivo).