Analisi statica e dinamica di un malware

Approccio Pratico

Analisi Statica

Consiste nell'esaminare un eseguibile vedere le istruzioni che lo compongono. L'obbiettivo della animi statica è quella di confermare se un file è malevolo e fornire circa le sue funzionalità.

Analisi Dinamica

Presuppone l'esecuzione del malware in modo da poter osservare il suo comportamento sul sistema infetto col fine di rimuovere l'infezione. I malware devono essere eseguiti in un ambiente sicuro e controllato in modo tale da non mettere a rischio il sistema o la rete.

Malware

Un malware (Malicius software) serve per descrivere un programma o codice malevolo/dannoso per un sistema.

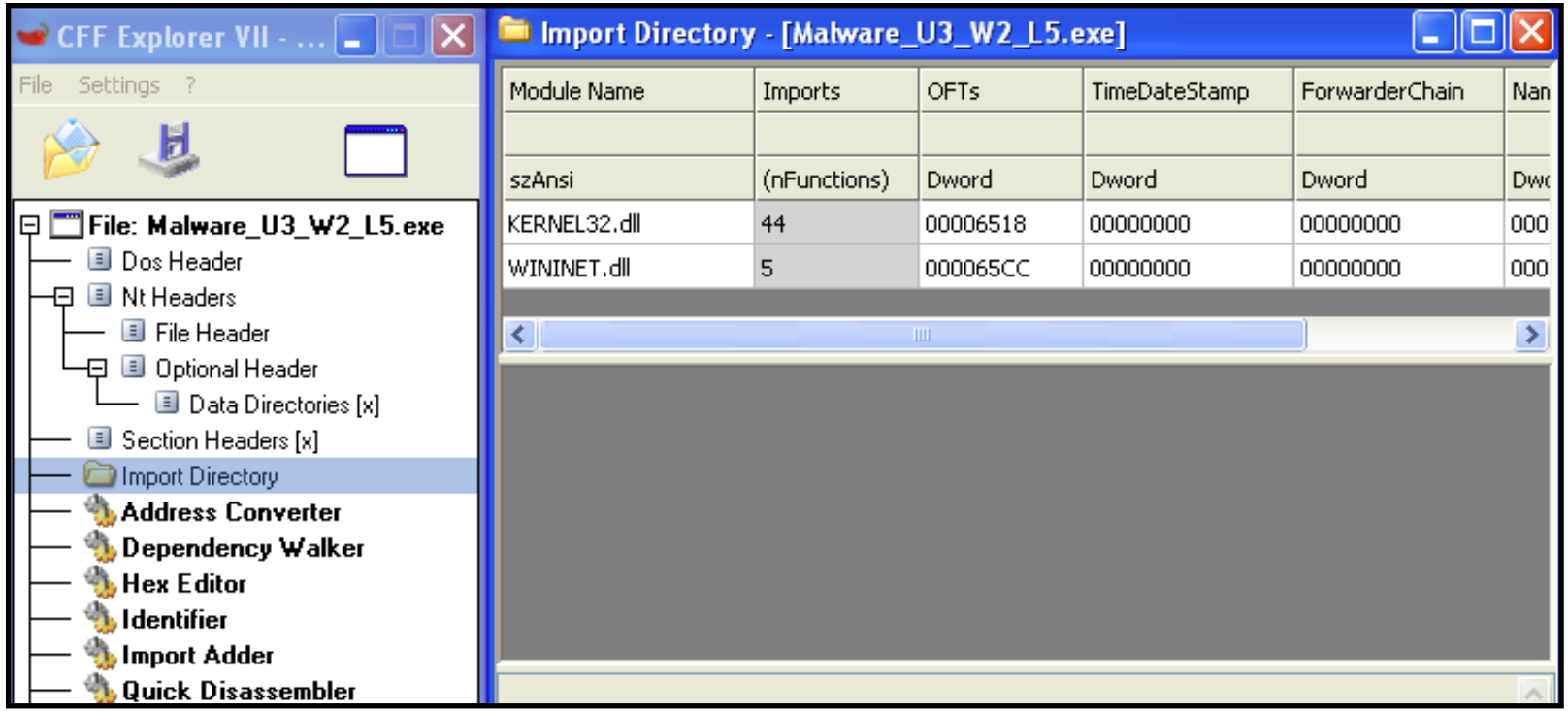
Tipo d'analisi utilizzato: Analisi Statica

Librerie del malware

Le librerie che vengono importate dall'eseguibile sono due: Kernel32.dll e Winnet.dll.
Come si può dall'immagine Kernel32.dll importa 44 funzioni e Winnet.dll 5 funzioni.

CFF Explorer

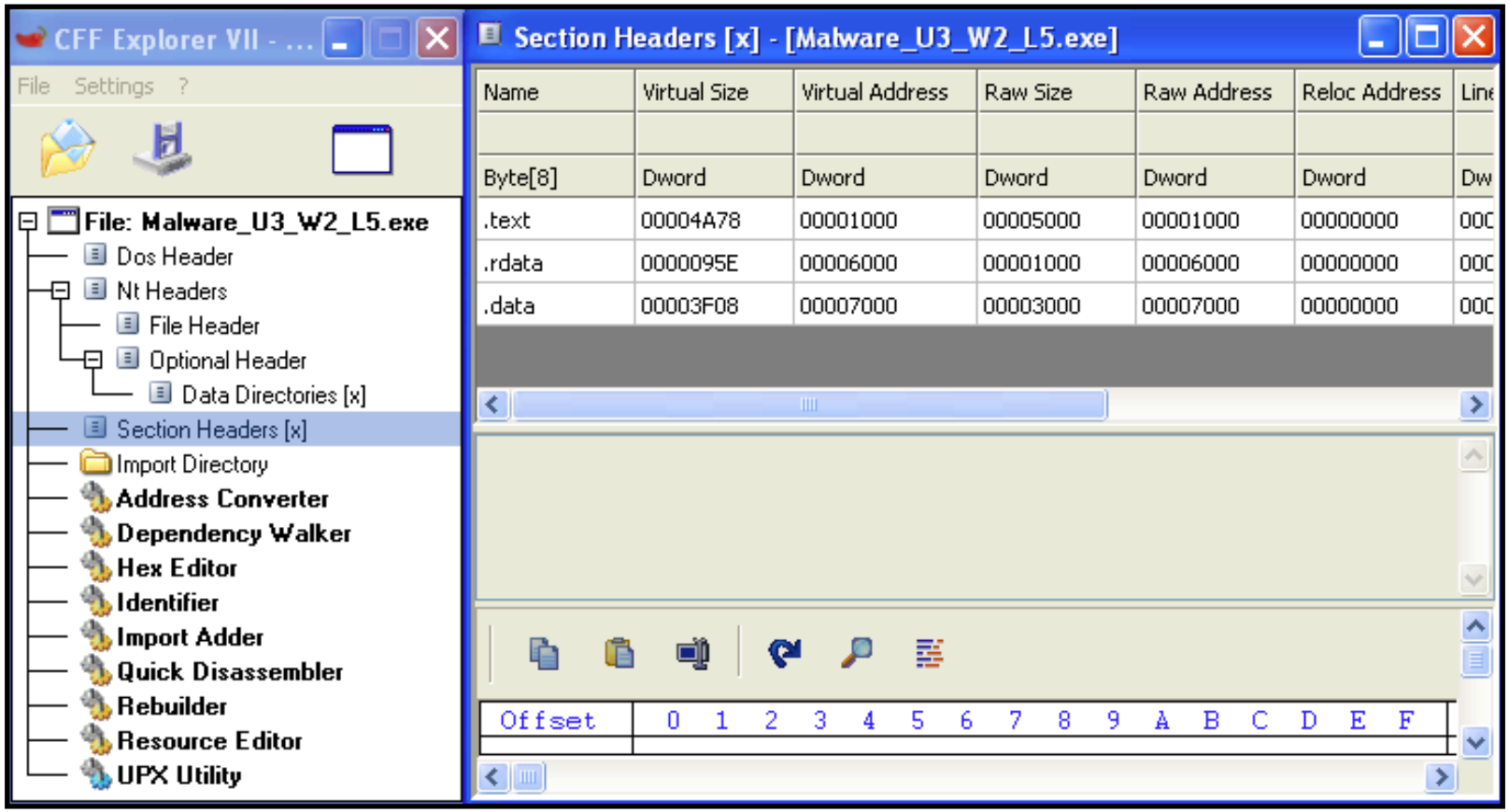
E' un tool che serve a controllare le funzioni/moduli importate ed esportate da un malware.



Kerlnel.32.dill: Contiene le funzioni principali del sistema operativo.
Wininet.dll: Contiene le funzioni per l'implementazione di alcuni protocolli di rete(HTTP, FTP, NTP).

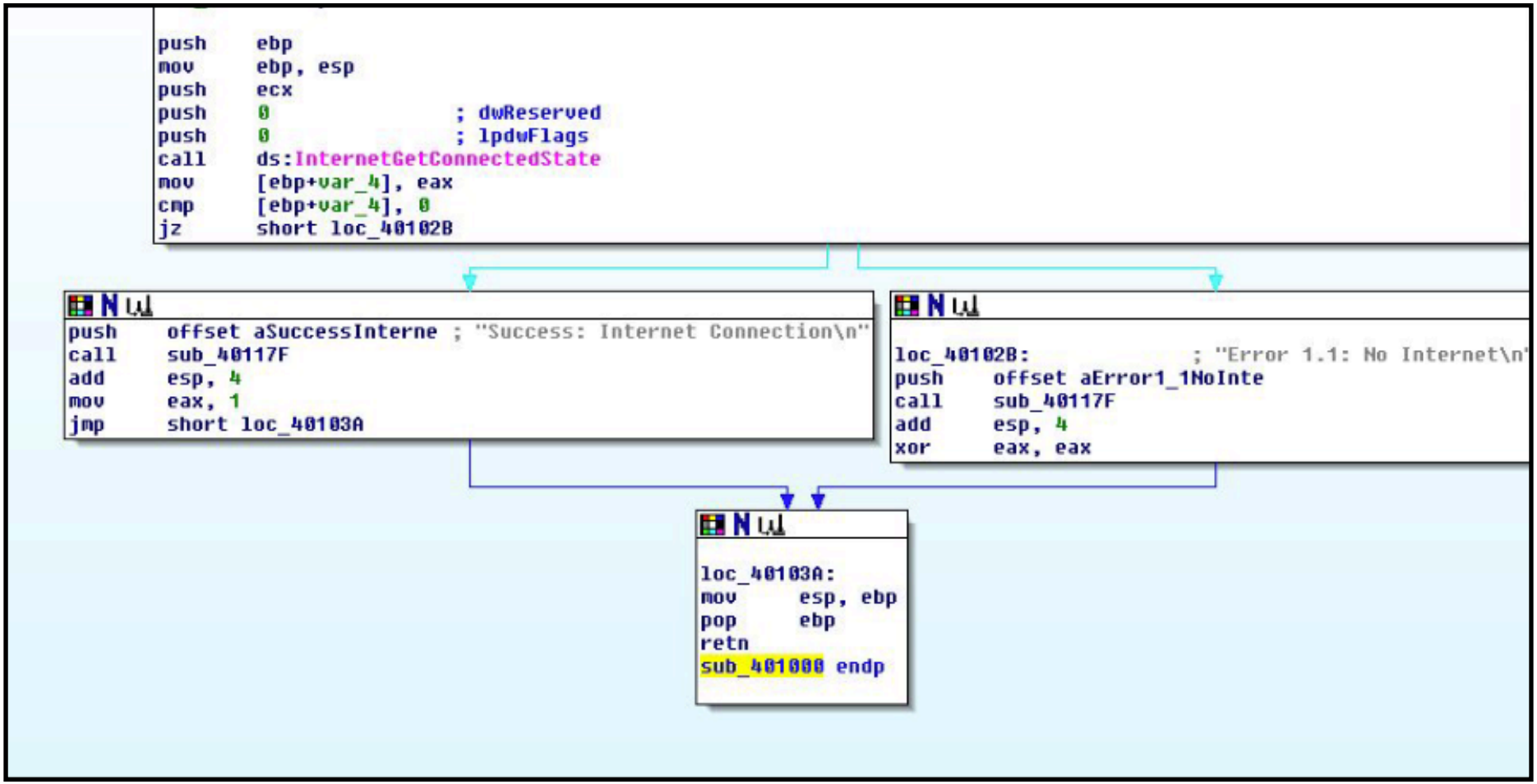
Sezioni del malware

Le sezioni che troviamo in questo eseguibile sono: .text, .rdata e .data.



.text: Contiene le istruzioni (Righe di codice) che saranno eseguite una volta il software sarà avviato.
.rdata: Include le informazioni circa le librerie e le funzioni importate ed esportate dall'eseguibile.
.data: Contiene i dati e le variabili globali del programma eseguibile, che devono essere disponibili da qualsiasi parte del programma.

Identificazione dei costrutti



Costrutti identificati

Qui avviene la creazione dello stack. Questo viene definito da EBP che punta alla sua base ed ESP che punta alla cima

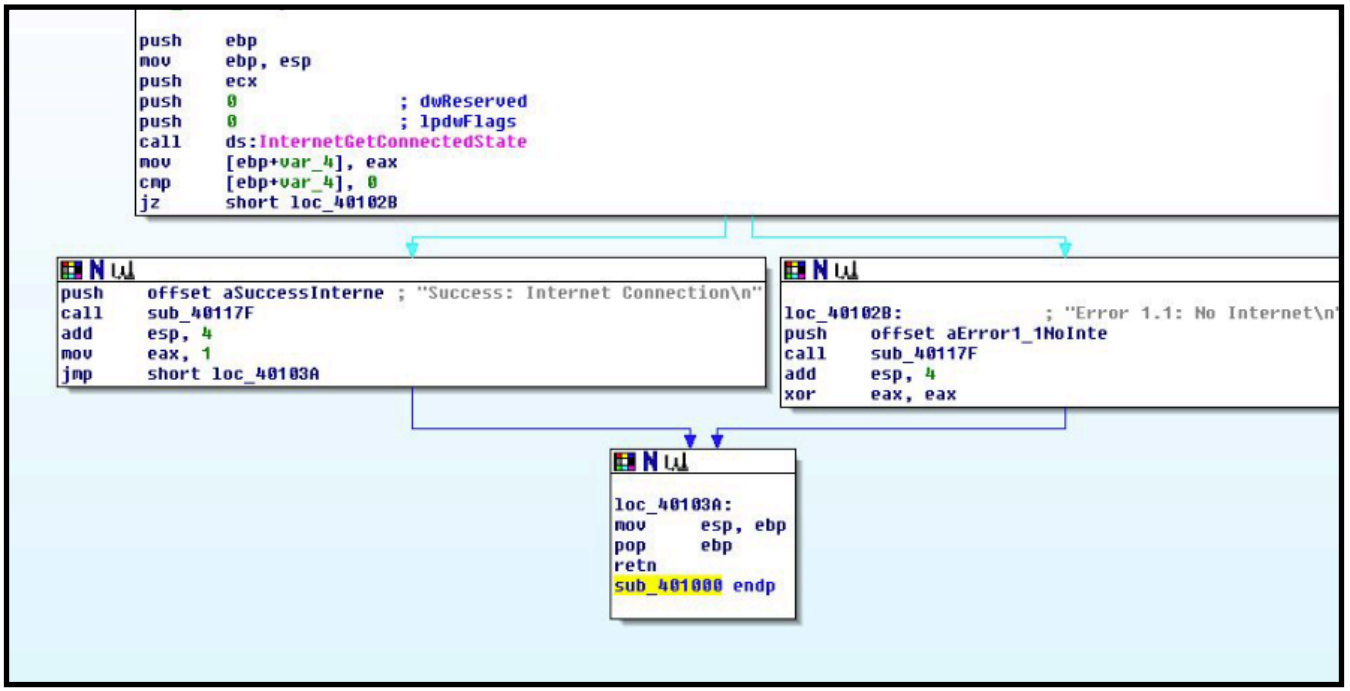
Chiamata di funzione perché i parametri sono stati passati sullo stack tramite push. Si possono identificare per la sequenza di 3° push e 1° call.

E' un ciclo IF perché l'istruzione <<cmp>> unita all'istruzione jz controlla l'uguaglianza tra le variabili. Nel caso gli operandi siano diversi tra di loro jz salterà alla locazione di memoria specificata.

Stack: E' un'area di memoria continua gestita in modalità 'Last First Out' LIFO, questo vuol dire che l'ultimo oggetto inserito è il primo che verra rimosso.

Chiamate di funzione: Sono delle funzioni che possono chiamare una seconda funzione per svolgere una determinata funzione.

Ipotesi funzionalità



Cerca lo stato della connessione internet e per farlo usa il costrutto IF. Se la connessione è andata a buon fine verrà stampato "Success: Internet Connection\n ", come si può vedere dall'immagine in basso a sinistra, invece se avviane il contrario verrà stampato "Error 1.1: No Internet\n" come si può vedere a destra nell'immagine.

Quindi il valore restituito 0 viene confrontato con 0, se non 0 è 1, viene stampato "Success: Internet Connection", se è 0, viene stampato "Error 1.1: No Internet"