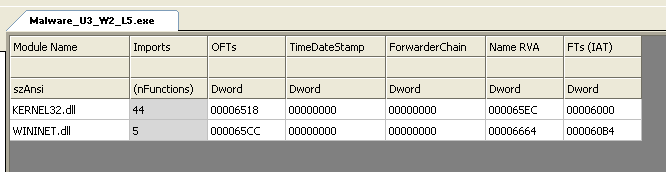
**PROGETTO SETTIMANALE – MALWARE ANALYSIS**

RISOLUZIONE TRACCIA:

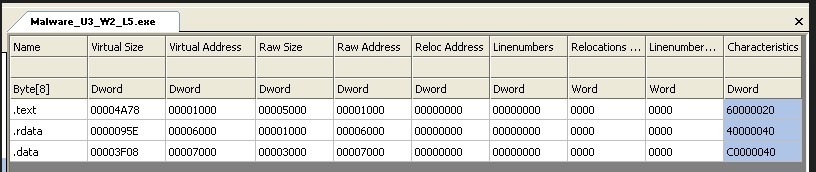
**Punto 1:**Il file eseguibile **Malware\_U3\_W2\_L5** importa le seguenti librerie:



* La libreria **KERNELL32.dll** è una delle librerie di sistema fondamentali di Microsoft Windows. Essa contiene funzioni, utilizzate dal sistema operativo e da molti programmi installati sul sistema, che includono gestione della memoria, gestione dei file, gestione della rete, gestione dei processi e molte altre funzioni di basso livello necessarie per il corretto funzionamento di windows e dei programmi installati.
* La libreria **WININET.dll** è una libreria di sistema di Microsoft Windows che fornisce una serie di funzionalità per la gestione delle connessioni Internet e delle risorse Web. Questa libreria fornisce le API (Application Programming Interface) necessarie per creare applicazioni in grado di connettersi a Internet, inviare richieste HTTP, FTP e Gopher, gestire i cookie, gestire la cache delle pagine web, e altre funzionalità relative alla connessione e alla navigazione web.

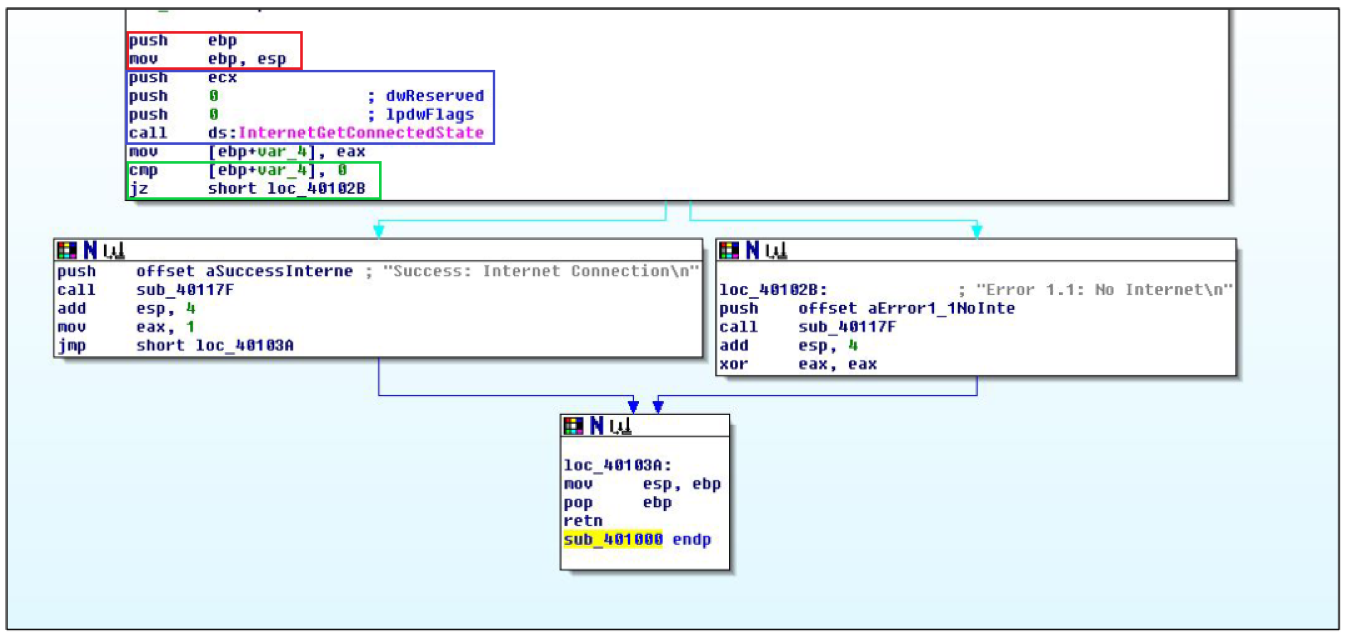
**Punto 2:**

Il file eseguibile **Malware\_U3\_W2\_L5** si compone delle seguenti funzioni:



* La funzione **.text** contiene il codice eseguibile dell'applicazione. In questa sezione vengono memorizzati tutti i blocchi di istruzioni in linguaggio macchina che vengono eseguiti dal processore quando l'applicazione viene avviata.
* La funzione **.rdata** (read-only data) contiene i dati costanti utilizzati dall'applicazione. In questa sezione vengono memorizzati i dati che l'applicazione utilizza durante l'esecuzione ma che non possono essere modificati, ad esempio le stringhe di testo, le tabelle di lookup, i valori di costanti matematiche e altro ancora.
* La funzione **.data** contiene i dati modificabili utilizzati dall'applicazione. In questa sezione vengono memorizzati i dati che possono essere modificati durante l'esecuzione dell'applicazione, ad esempio le variabili globali, le tabelle di lookup dinamiche, i buffer di input/output e altro ancora.

**Punto 3:**

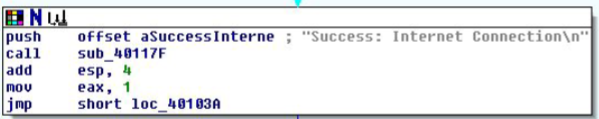


* Nel riquadro **ROSSO** si trova la creazione dello stack
* Nel riquadro **BLU** si trova la chiamata di funzione. I parametri sono passati sullo stack tramite le istruzioni push.
* Nel riquadro **VERDE** si trova il ciclo if.

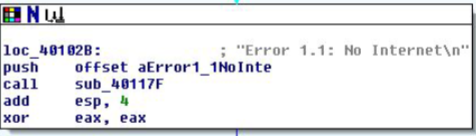
**Punto 4:**

Il comportamento della funzione implementata è quello di verificare che la macchina vittima abbia o meno la connessione ad internet e di restituire un valore che indica se tale verifica abbia avuto un riscontro positivo o negativo. Soltanto nel caso in cui non ci sia connessione ad internet viene stampato il messaggio di errore, invece nel caso in cui sia presente l’intento del codice sembrerebbe quello di poter permettere di eseguire altre azioni anche se non si può sapere con certezza.

Nello specifico nel seguente blocco viene impostato il valore di ritorno della funzione a 1 se la connessione ad internet è stabilita con successo, altrimenti viene impostato a 0.



Invece nel blocco con codice “loc\_40102B” viene chiamata la funzione “sub\_40117F” con un messaggio di errore “Error 1.1: No internet\n”



Nell’ultimo blocco invece viene svuotato lo stack ed esegue quello che sarebbe un return 0 per uscire dalla funzione