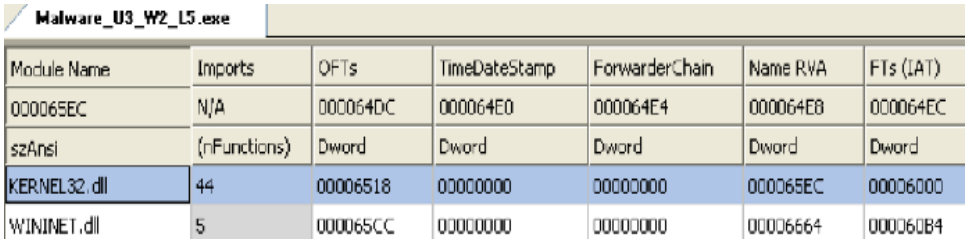
Individuazione delle librerie

Per individuare le librerie di un malware o di un semplice file eseguibile, è necessario effettuare un'analisi statica del file sospetto utilizzando strumenti specifici come CFF Explorer. Questo tipo di analisi consente di identificare le librerie importate dal file, comprese quelle potenzialmente dannose.

Dall utilizzo dello strumento compaiono 2 librerie



Malware_U3_W2_L5.exe

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
000065EC	N/A	000064DC	000064E0	000064E4	000064E8	000064EC
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.dll	44	00006518	00000000	00000000	000065EC	00006000
WININET.dll	5	000065CC	00000000	00000000	00006664	000060B4

La libreria kernel32.dll è una libreria di sistema essenziale fornita con Windows. Fornisce una serie di funzioni di basso livello che sono utilizzate da molti programmi Windows, inclusi malware. Alcune delle funzioni più comuni utilizzate dai malware includono:

- CreateFile(): Questa funzione viene utilizzata per creare un nuovo file. Un malware può utilizzare questa funzione per creare un file nascosto che contiene codice malevolo.
- ReadFile(): Questa funzione viene utilizzata per leggere i dati da un file. Un malware può utilizzare questa funzione per leggere i file di registro di Windows o altri file sensibili.
- WriteFile(): Questa funzione viene utilizzata per scrivere i dati su un file. Un malware può utilizzare questa

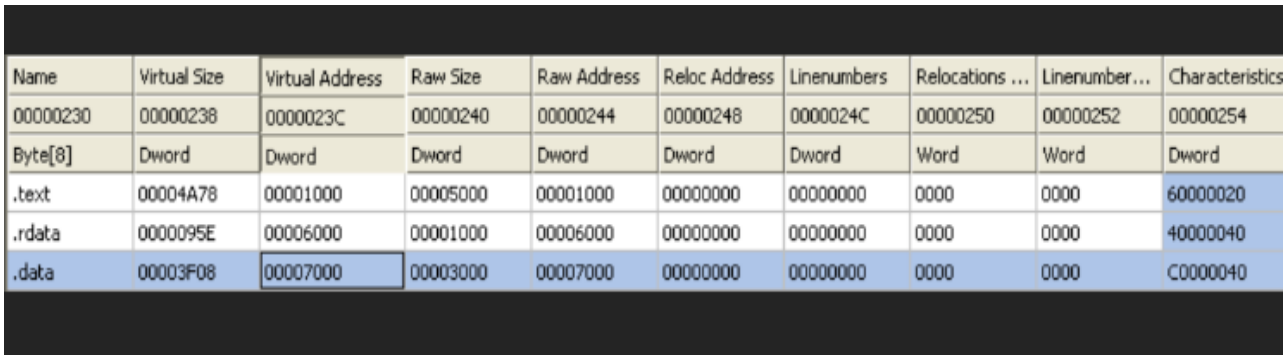
funzione per scrivere codice malevolo su un file eseguibile o per modificare i file di registro di Windows.

Libreria wininet.dll

La libreria wininet.dll è una libreria di sistema utilizzata per accedere alla rete. Fornisce una serie di funzioni che consentono ai programmi di stabilire connessioni di rete, scaricare file e inviare richieste HTTP. I malware possono utilizzare questa libreria per svolgere una serie di attività dannose, tra cui:

- Establishing unauthorized network connections: Un malware può utilizzare questa libreria per stabilire connessioni di rete non autorizzate con server remoti. Ciò consente al malware di comunicare con i suoi creatori o di scaricare altri file dannosi.
- Downloading malicious files: Un malware può utilizzare questa libreria per scaricare file dannosi da Internet. Questi file possono contenere codice malevolo che può essere eseguito sul computer compromesso.
- Stealing sensitive user information: Un malware può utilizzare questa libreria per rubare informazioni sensibili dell'utente, come credenziali di accesso, numeri di carte di credito o informazioni personali.

Analisi delle sezioni

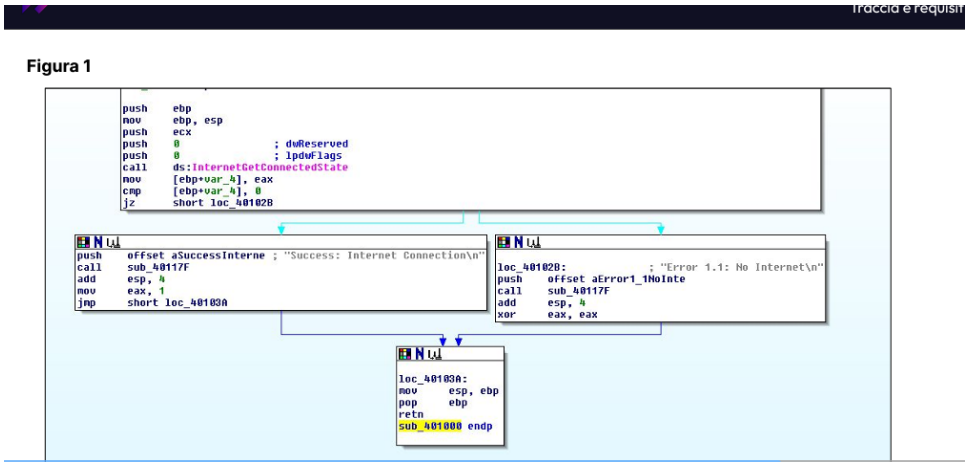


Name	Virtual Size	Virtual Address	Raw Size	Raw Address	Reloc Address	Linenumbers	Relocations ...	Linenumber...	Characteristics
00000230	00000238	0000023C	00000240	00000244	00000248	0000024C	00000250	00000252	00000254
Byte[8]	Dword	Dword	Dword	Dword	Dword	Dword	Word	Word	Dword
.text	00004A78	00001000	00005000	00001000	00000000	00000000	0000	0000	60000020
.rdata	0000095E	00006000	00001000	00006000	00000000	00000000	0000	0000	40000040
.data	00003F08	00007000	00003000	00007000	00000000	00000000	0000	0000	C0000040

- **.text (o TEXT):** Questa sezione contiene il codice eseguibile. In linguaggi come C, C++, o linguaggi assembly, questa è la parte del programma che viene tradotta direttamente in istruzioni macchina. Queste istruzioni sono ciò che il processore eseguirà durante l'esecuzione del programma.
- **.rdata (o RDATA):** I dati di sola lettura sono contenuti in questa sezione. Questi dati non possono essere modificati durante l'esecuzione del programma. Possono includere stringhe di testo, costanti o tabelle di lookup. Nel contesto della sicurezza informatica, i malware possono accedere a queste informazioni per eseguire determinate azioni, ma non possono modificarle.
- **.data (o DATA):** Questa sezione contiene dati modificabili, che possono essere letti e modificati durante l'esecuzione del programma. Questi dati possono includere variabili, strutture dati o altre informazioni che vengono manipolate o aggiornate dal programma o, nel contesto della tua descrizione,

Analisi dell assembly

Dato il seguente codice assembly eseguiremo uno sua analisi



I costrutti noti che possiamo visualizare sono i seguenti :

- `push` e `pop`: per manipolare lo stack
- `cmp`: per confrontare due valori
- `jz`: per saltare a un'etichetta se il valore di confronto è zero
- `push`: per memorizzare un valore sullo stack
- `call`: per chiamare una funzione
- `retn`: per tornare dalla funzione

Come funzionano nel dettaglio

push e pop

Le istruzioni `push` e `pop` vengono utilizzate per manipolare lo stack. Lo stack è una struttura dati LIFO (Last In First Out), ovvero l'elemento inserito per ultimo viene estratto per primo.

L'istruzione `push` viene utilizzata per inserire un valore sullo stack. Il valore viene memorizzato nell'indirizzo di memoria specificato.

L'istruzione pop viene utilizzata per estrarre un valore dallo stack. Il valore viene memorizzato nella variabile specificata.

Ad esempio, l'istruzione `push eax` inserisce il valore del registro `eax` sullo stack. L'istruzione `pop ebx` estrae il valore dallo stack e lo memorizza nel registro `ebx`.

cmp

L'istruzione `cmp` viene utilizzata per confrontare due valori. I valori possono essere registri, variabili, costanti o espressioni.

L'istruzione `cmp` restituisce un valore intero che indica se i due valori sono uguali, se il primo valore è maggiore del secondo valore o se il primo valore è minore del secondo valore.

Ad esempio, l'istruzione `cmp eax, ebx` confronta i valori dei registri `eax` e `ebx`. Se i valori sono uguali, l'istruzione restituisce 0. Se il valore di `eax` è maggiore del valore di `ebx`, l'istruzione restituisce un valore maggiore di 0. Se il valore di `eax` è minore del valore di `ebx`, l'istruzione restituisce un valore minore di 0.

jz

L'istruzione `jz` viene utilizzata per saltare a un'etichetta se il valore di confronto è zero.

L'istruzione `jz` prende due argomenti: l'etichetta a cui saltare e il valore di confronto. Se il valore di confronto è zero,

l'istruzione salta all'etichetta specificata. Altrimenti, l'istruzione continua l'esecuzione del codice successivo.

Ad esempio, l'istruzione `jz loc_401028` salta all'etichetta `loc_401028` se il valore di confronto è zero.

push

call

L'istruzione `call` viene utilizzata per chiamare una funzione. La funzione viene eseguita e il controllo viene restituito al codice chiamante quando la funzione termina.

L'istruzione `call` prende un solo argomento: l'indirizzo della funzione da chiamare.

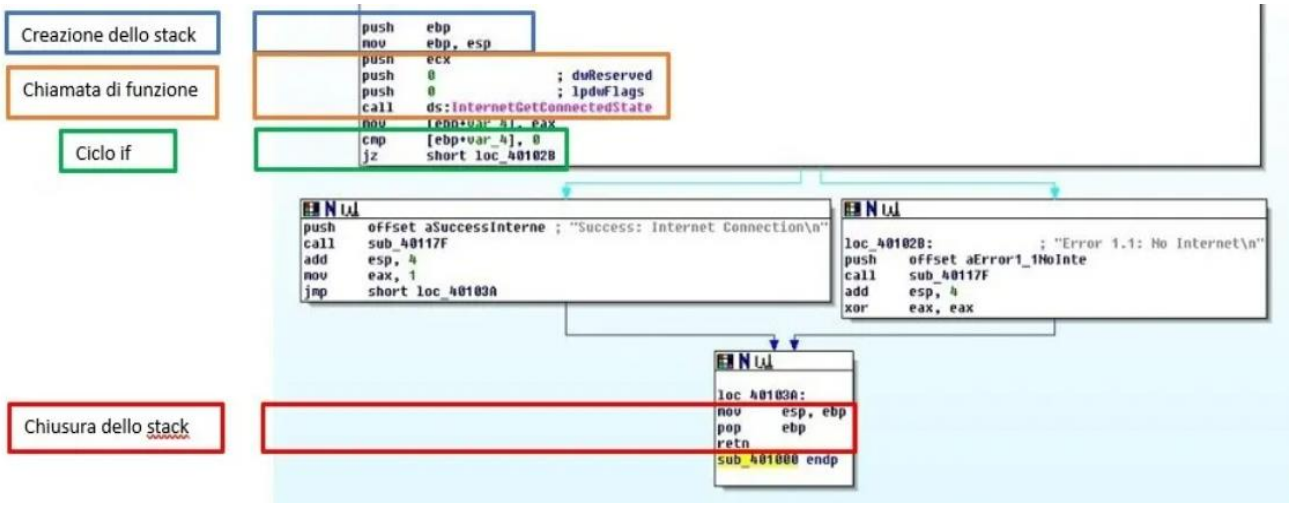
Ad esempio, l'istruzione `call sub_40117F` chiama la funzione `sub_40117F`.

retn

L'istruzione `retn` viene utilizzata per tornare dalla funzione. La funzione viene terminata e il controllo viene restituito al codice chiamante.

L'istruzione `retn` non prende argomenti.

Eccovi una presentazione grafica dei vari costrutti



Ipotesi di c

Ecco qui un piccolo esempio di come il codice posso funzionare in c: #include <stdio.h>

```
int main() {
  // Controlla lo stato di connessione a Internet
  int connected = InternetGetConnectedState();

  // Stampa un messaggio di successo se la connessione è presente
  if (connected != 0) {
    printf("Connessione a Internet presente\n");
  }
  // Altrimenti, stampa un messaggio di errore
  else {
    printf("Connessione a Internet assente\n");
```

```
  }

  return 0;
}
```

Rischio di utilizzo della funzione

1. Un utente malintenzionato potrebbe utilizzare questa funzione per verificare se un computer è connesso a Internet prima di avviare un attacco informatico.
2. Un utente malintenzionato potrebbe utilizzare questa funzione per raccogliere informazioni personali o sensibili, come ad esempio le credenziali di accesso a un sito web, una volta che il computer è connesso a Internet.
3. Un utente malintenzionato potrebbe utilizzare questa funzione per monitorare l'attività di un utente su Internet senza il loro consenso.
4. Un utente malintenzionato potrebbe utilizzare questa funzione per installare software dannoso sul computer di un utente una volta che il computer è connesso a Internet.
5. Un utente malintenzionato potrebbe utilizzare questa funzione per creare una backdoor sul computer di un utente, consentendo loro di accedere al computer in qualsiasi momento.