S10-LS5

MALWARE

CONTENT

COS'È UN MALWARE?

Un malware è un termine generico che si riferisce a software dannoso progettato per infiltrarsi o danneggiare un computer o un sistema informatico senza il consenso dell'utente. Il termine "malware" deriva dalla combinazione delle parole "malicious" (malizioso) e "software". Esistono vari tipi di malware, tra cui virus, worm, trojan, ransomware, adware, spyware e rootkit, ognuno con scopi e comportamenti diversi, ma tutti mirano a danneggiare, rubare dati o interferire con il normale funzionamento del sistema informatico o della rete.

TOOL

01

CFF EXPLORER

CFF Explorer è uno strumento per analizzare e modificare file eseguibili. Utile per sviluppatori e ricercatori per comprendere il comportamento dei programmi.

02

PROCESS MONITORING

Il monitoraggio dei processi è il controllo in tempo reale delle attività e delle risorse utilizzate dai processi software su un sistema informatico, utile per il debug, la sicurezza e l'ottimizzazione delle prestazioni.

03

PROCESS EXPLORER

Process Explorer è un software di utilità per Windows che visualizza dettagli sui processi in esecuzione, inclusi handle e DLL, utile per il debugging e la gestione delle risorse di sistema.

04

REGSHOT

Regshot è uno strumento per Windows che confronta lo stato del registro di sistema prima e dopo un'azione, utile per individuare modifiche e problemi di installazione o configurazione del software.

TRACCIA 1 - ANALISI

TRACCIA:

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

Malware_U3_W2_L5.exe

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.dll	44	00006518	00000000	00000000	000065EC	00006000
WININET.dll	5	000065CC	00000000	00000000	00006664	000060B4

per controllare quali librerie il malware stia utilizzando possiamo usare il tool “cff explorer”,

La libreria Kernel32.dll è una componente fondamentale del sistema operativo Microsoft Windows. Essa fornisce una vasta gamma di funzioni di basso livello che gestiscono il sistema operativo, tra cui gestione dei processi, della memoria, dei file, della sincronizzazione e delle interruzioni hardware.

La libreria Wininet.dll è una componente di Windows che fornisce funzionalità per la comunicazione via Internet. Essa offre supporto per protocolli come HTTP, HTTPS, FTP e altri, consentendo alle applicazioni di accedere e interagire con risorse online come pagine web, file e servizi remoti.

Malware_U3_W2_L5.exe

Name	Virtual Size	Virtual Address	Raw Size	Raw Address	Reloc Address	Linenumbers	Relocations ...	Linenumber...	Characteristics
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

Possiamo controllare le sezioni di cui si compone il file utilizzando ancora CFF Explorer, spostandoci nella sezione «section headers».

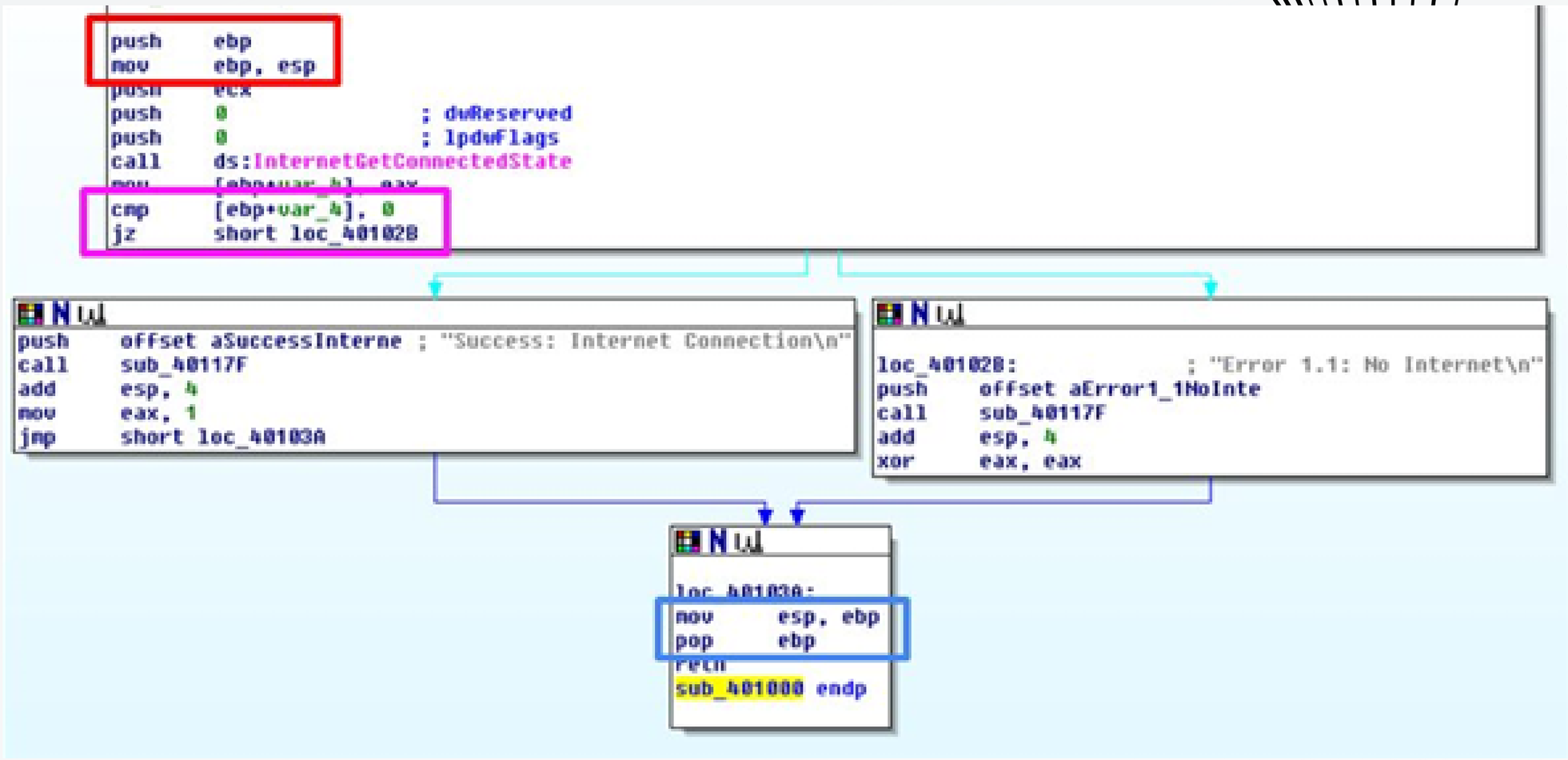
La sezione ".text" è una parte dei file eseguibili che contiene il codice macchina, le istruzioni e le funzioni del programma, fondamentali per l'esecuzione delle operazioni specificate nel software.

La sezione ".rdata" contiene dati di sola lettura, come stringhe e costanti, utilizzati dal programma durante l'esecuzione. È separata dalla sezione ".data" che contiene dati modificabili.

ASSEMBLY

Il linguaggio assembly è un linguaggio di programmazione a basso livello che rappresenta istruzioni di macchina in formato mnemonico, comprensibili agli esseri umani e traducibili direttamente in codice macchina dai processori. Ogni istruzione assembly corrisponde a un'operazione specifica del processore, come movimenti di dati, operazioni aritmetiche o controllo del flusso. L'assembly è molto vicino all'hardware e offre un controllo preciso sulle risorse del sistema. È spesso utilizzato per scrivere codice altamente ottimizzato, dispositivi embedded, driver di dispositivo e in contesti di reverse engineering. Tuttavia, a causa della sua complessità e della necessità di familiarità con l'architettura del processore target, l'assembly è considerato più difficile da scrivere e da comprendere rispetto ai linguaggi di programmazione ad alto livello.

ASSEMBLY - IDENFICAZIONE COSTRUTTORI



PUSH EBP: *mette il valore corrente del registro di base (EBP) nello stack*

MOV EBP, ESP: *copia il valore dello stack pointer (ESP) nel registro di base (EBP)*

CMP [EBP+VAR_4], 0: *confronta il valore memorizzato nella posizione di memoria [ebp+var_4] con zero. Questo può essere parte di una condizione, controllando se il valore in [ebp+var_4] è uguale a zero o meno, o se è diverso da zero*

JZ SHORT LOC_40102B: *è una istruzione di salto condizionato nell'assembly x86. "JZ" sta per "Jump if Zero", e "short" indica che il salto è limitato a una distanza corta rispetto all'indirizzo di istruzione corrente*

MOV ESP, EBP *copia il valore del registro di base (EBP) nello stack pointer (ESP)*

POP EBP: *recupera il valore del registro di base (EBP) dallo stack, che è stato precedentemente salvato all'inizio della funzione*

IPOTESI FUNZIONAMENTO

La funzione "getinternetconnectstate" controlla la presenza di una connessione Internet. Se il suo valore restituito è 0, il programma mostra "no internet"; altrimenti, mostra "Success: Internet Connection". Il costrutto "IF" valuta questo valore e gestisce l'output di conseguenza. Questa funzionalità è utile per verificare lo stato della connettività Internet prima di eseguire operazioni dipendenti dalla rete, consentendo al programma di adattarsi dinamicamente in base alla disponibilità della connessione.