Analisi Statica Campione "Agent Tesla"

1. Introduzione

L'obiettivo di questa sessione è condurre un'analisi statica preliminare su un campione di malware, identificato come una variante di **Agent Tesla**. L'analisi è stata condotta in un ambiente virtualizzato sicuro (FlareVM) privo di connessione di rete verso l'esterno (Host-Only/Internal Network) per prevenire qualsiasi rischio di propagazione. L'approccio statico mira a identificare gli Indicatori di Compromissione (IOC) di base, la struttura del file e l'eventuale presenza di offuscamento (packing).

2. Strumenti Utilizzati

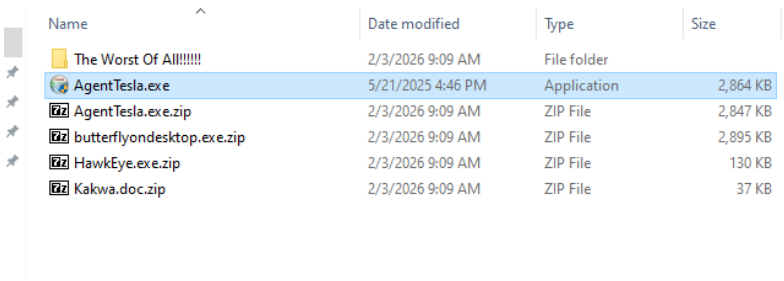
Per l'analisi sono stati usati i seguenti strumenti standard della distribuzione FlareVM:

- **HashMyFiles:** Per il calcolo dell'impronta digitale (fingerprinting).
- **CFF Explorer:** Per l'analisi dell'header PE e dei timestamp.
- **Detect It Easy :** Per l'identificazione di packer e compilatori.
- **Strings / BinText:** Per l'estrazione di stringhe ASCII e Unicode.

3. Esecuzione dell'Analisi

Preparazione Ambiente

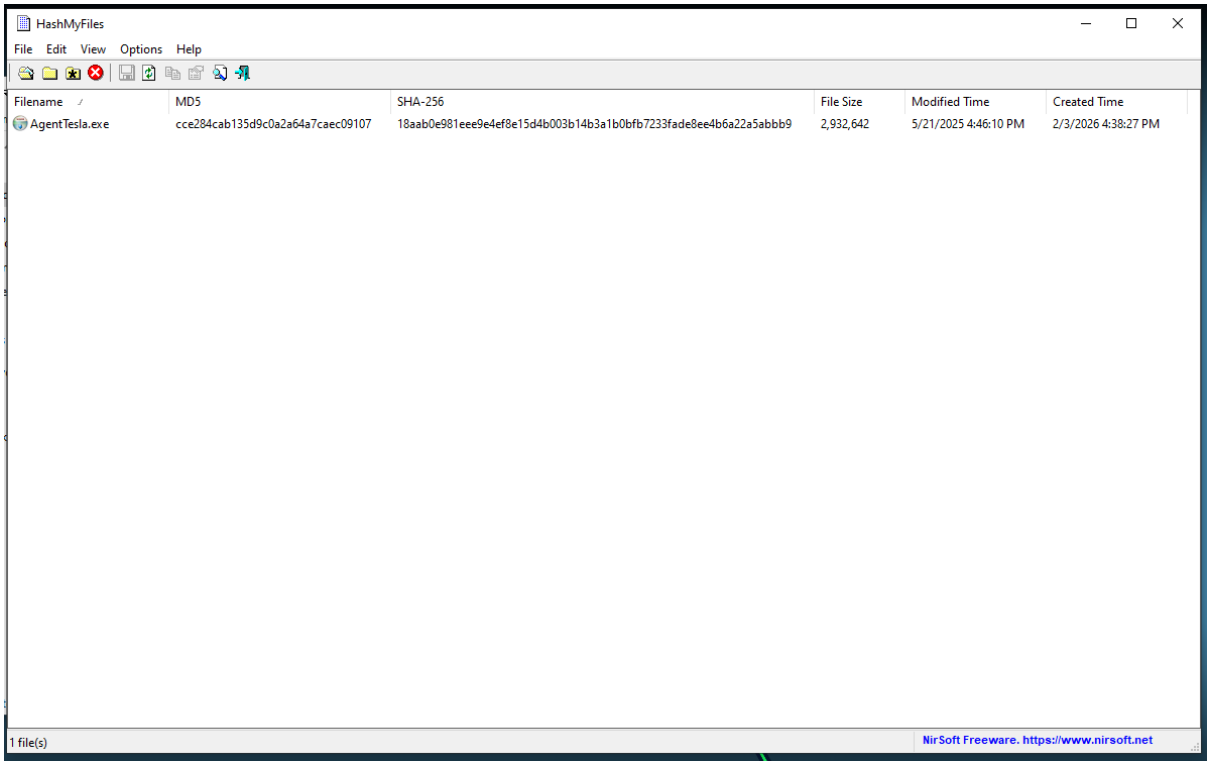
- Ho avviato la macchina virtuale FlareVM e localizzato il file compresso Agent Tesla.zip.
- Ho estratto il file
- **Risultato:** Ora ho il file eseguibile (.exe) sul desktop.



Fingerprinting

Il primo passo è identificare univocamente il file. Se il nome del file cambia, l'hash rimane identico. Questo è fondamentale per cercare report online (es. su VirusTotal) relativi allo stesso sample.

- Clicco con il tasto destro sul file scompattato e seleziono **"HashMyFiles"**

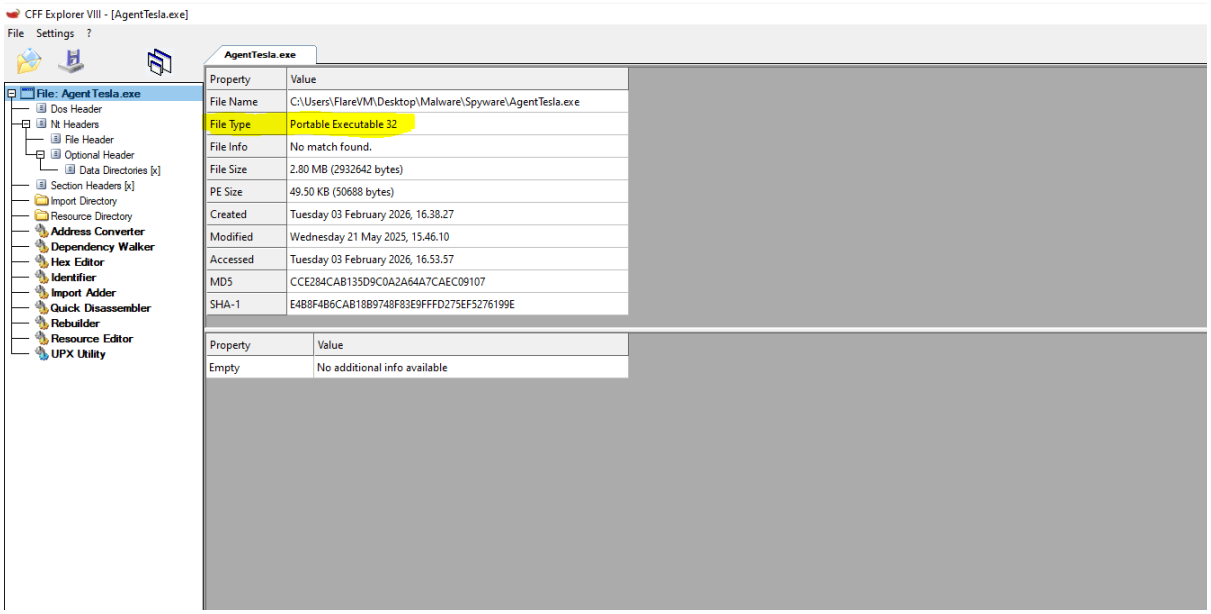


Analisi Struttura PE

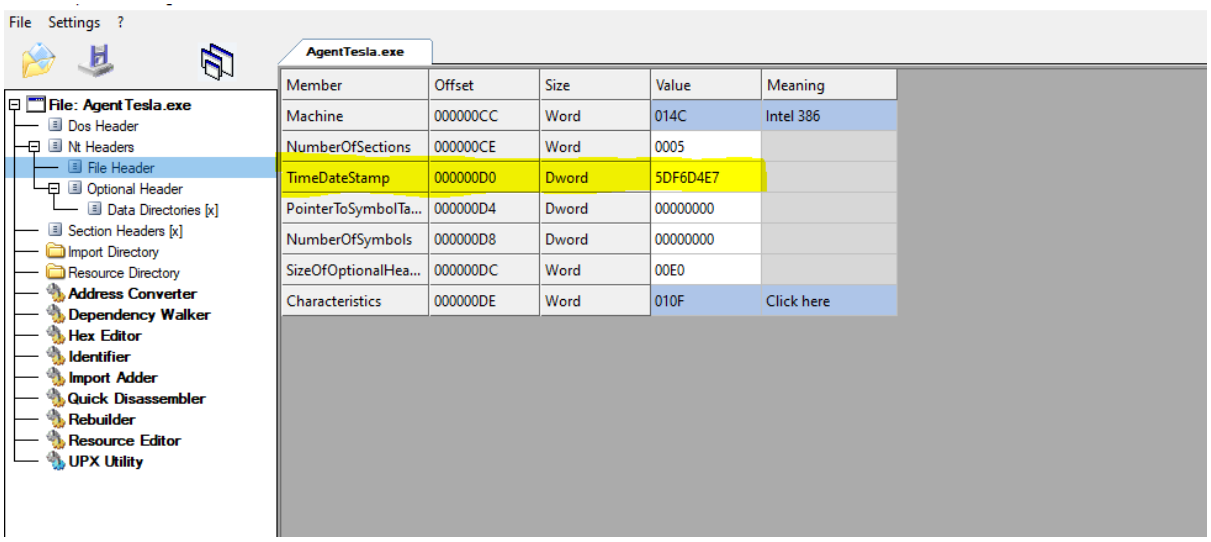
Ora analizzo l'intestazione del file (Portable Executable Header) per capire come è stato costruito.

- Apro il file con CFF Explorer

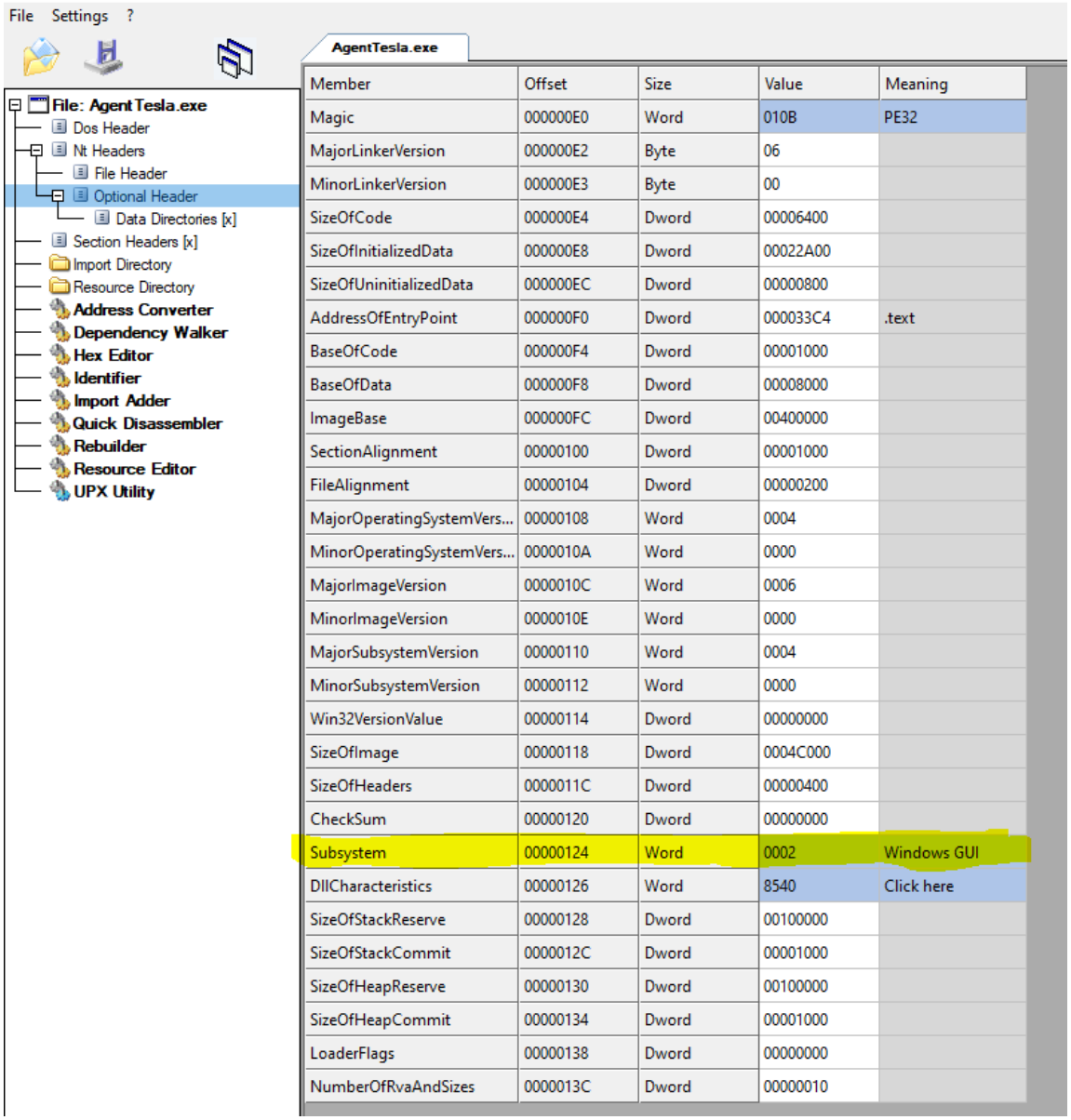
- **Analisi dei campi:**
    - **Architettura:** Cerco se è a 32-bit (x86) o 64-bit (x64). Agent Tesla è spesso a 32-bit per compatibilità.



    - **Timestamp:** Guardo la data di compilazione.
        - Analisi critica: Se la data è nel futuro (es. 2099) o molto nel passato (es. 1992), siamo di fronte a **"Time Stomping"** (una tecnica per falsificare la data di creazione e nascondersi). Se è una data recente e plausibile la annoto.



- **Subsystem:** GUI (Graphical User Interface) o Console. I malware spesso usano GUI anche se non mostrano finestre, per evitare che appaia la schermata nera del terminale.

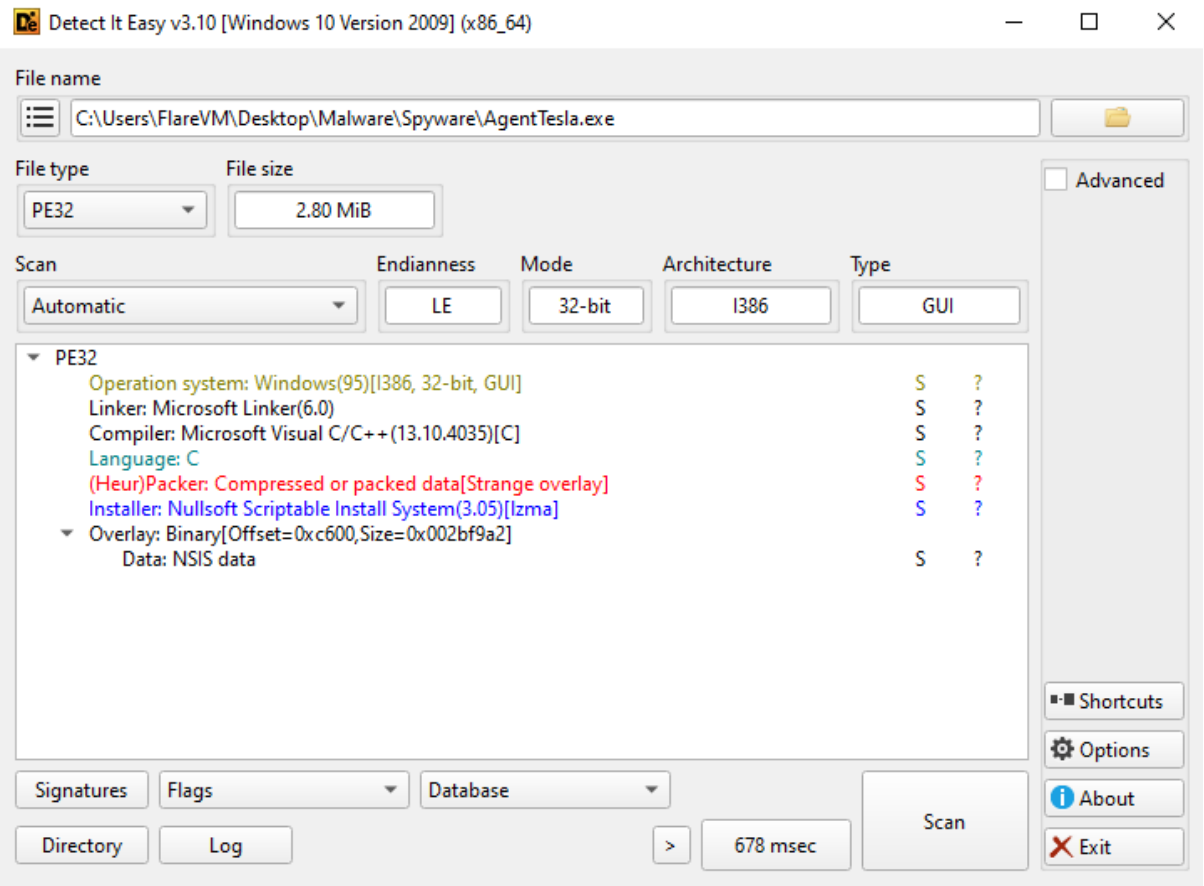


Member	Offset	Size	Value	Meaning
Magic	000000E0	Word	010B	PE32
MajorLinkerVersion	000000E2	Byte	06	
MinorLinkerVersion	000000E3	Byte	00	
SizeOfCode	000000E4	Dword	00006400	
SizeOfInitializedData	000000E8	Dword	00022A00	
SizeOfUninitializedData	000000EC	Dword	00000800	
AddressOfEntryPoint	000000F0	Dword	000033C4	.text
BaseOfCode	000000F4	Dword	00001000	
BaseOfData	000000F8	Dword	00008000	
ImageBase	000000FC	Dword	00400000	
SectionAlignment	00000100	Dword	00001000	
FileAlignment	00000104	Dword	00000200	
MajorOperatingSystemVers...	00000108	Word	0004	
MinorOperatingSystemVers...	0000010A	Word	0000	
MajorImageVersion	0000010C	Word	0006	
MinorImageVersion	0000010E	Word	0000	
MajorSubsystemVersion	00000110	Word	0004	
MinorSubsystemVersion	00000112	Word	0000	
Win32VersionValue	00000114	Dword	00000000	
SizeOfImage	00000118	Dword	0004C000	
SizeOfHeaders	0000011C	Dword	00000400	
CheckSum	00000120	Dword	00000000	
Subsystem	00000124	Word	0002	Windows GUI
DllCharacteristics	00000126	Word	8540	Click here
SizeOfStackReserve	00000128	Dword	00100000	
SizeOfStackCommit	0000012C	Dword	00001000	
SizeOfHeapReserve	00000130	Dword	00100000	
SizeOfHeapCommit	00000134	Dword	00001000	
LoaderFlags	00000138	Dword	00000000	
NumberOfRvaAndSizes	0000013C	Dword	00000010	

Rilevamento Packer

Molti malware sono "impacchettati" (compressi o criptati) per nascondere il vero codice agli antivirus. Dobbiamo capire se questo sample è "Packed".

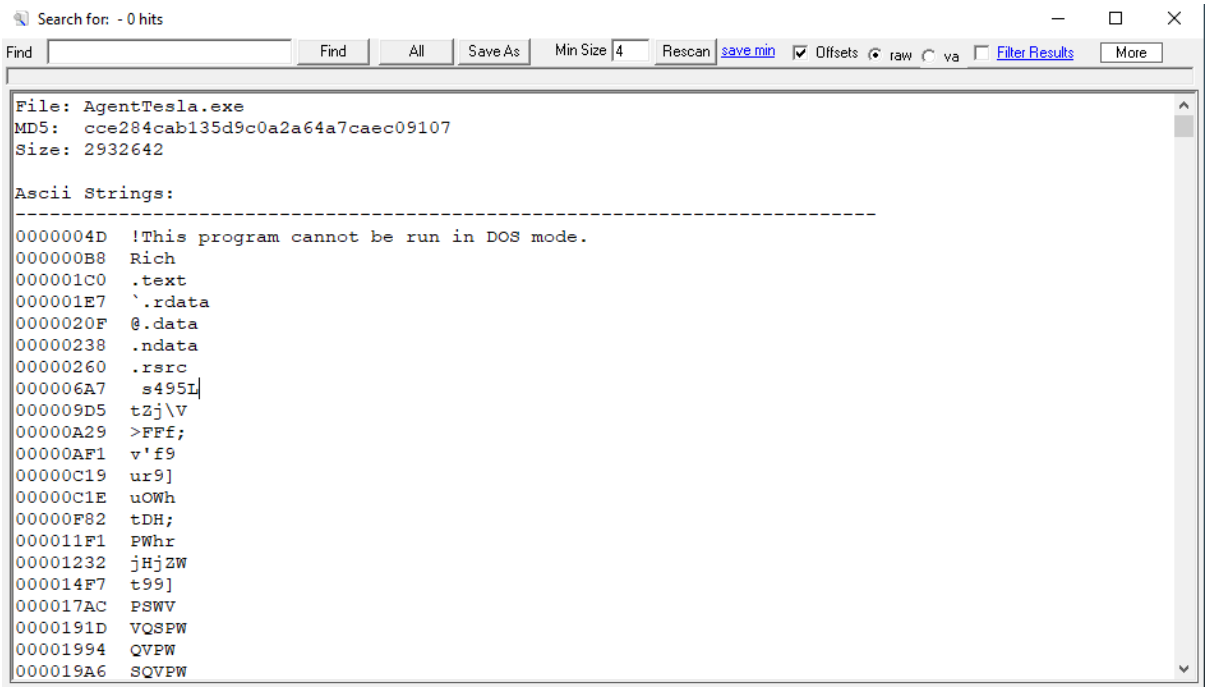
- Apro il file con **Detect It Easy**
- **Cosa cercare:**
    - Guardo la scritta in basso o nella sezione "Compiler". Se leggi "Microsoft Visual Studio .NET", il malware è scritto in C# o VB.NET (tipico di Agent Tesla).
    - Guardo l'**Entropia**: Se è molto alta (sopra 7.0), il file è quasi sicuramente "Packed" o criptato.



Analisi Stringhe

Le stringhe sono sequenze di caratteri leggibili nel codice. Se il file non è impacchettato bene, possiamo trovare indirizzi IP, password o nomi di file. Se è impacchettato, vedremo per lo più caratteri senza senso.

- In PEStudio, clicca sulla tab **"Strings"**.
- **Cosa cercare (Hunting):** Scorri velocemente e cerca parole chiave sospette



4. Conclusione

Dall'analisi statica effettuata sul campione, si evince che il file è un eseguibile per ambiente Windows (PE32), scritto in linguaggio C. Gli alti valori di entropia e la scarsità di stringhe funzionali in chiaro indicano che il sample è protetto da un **Packer**. Tuttavia, alcune stringhe residue suggeriscono intenti malevoli di persistenza ed esfiltrazione dati tipici degli Infostealer come Agent Tesla.