MALWARE ANALISYS

Cybersecurity & Ethical Hacking Progetto Finale

Matteo Mattia

INDICE GENERALE

PARTE I: INTRODUIONE

Nel presente report, procedo con l'analisi statica dettagliata del campione di malware denominato `notepad-classico.exe` , fornito attraverso l'ambiente FLARE VM (FireEye Labs Automated Reverse Engineering). Tale ambiente rappresenta uno dei standard più riconosciuti nell'analisi sicura dei malware, offrendo un ecosistema isolato e controllato per l'esame di campioni potenzialmente malevoli.

L'analisi è stata condotta utilizzando CFF Explorer VIII, uno strumento professionale di analisi per file PE (Portable Executable) di Windows, che ha permesso l'estrazione di informazioni strutturali critiche senza esporre il sistema di analisi a rischi operazionali. Il f ile oggetto di studio presenta caratteristiche interessanti dal punto di vista della sicurezza, evidenziando diverse anomalie che necessitano di approfondimento investigativo.

L'obiettivo di questa analisi è fornire una caratterizzazione completa del malware attraverso l'identificazione delle librerie di sistema importate e la mappatura dettagliata delle sezioni dell'eseguibile, elementi fondamentali per comprendere le capacità operative, i vettori di attacco e le potenziali tecniche di evasion utilizzate dal codice malevolo.

PARTE II: OFFICIAL: ANALISI STATICA

- 2.1 Librerire Importate del Malware

Librerie Importate dal Malware

Attraverso l'utilizzo della funzionalità "Import Directory" di CFF Explorer, ho identificato un set articolato di librerie dinamiche (DLL) importate dal malware.

L'analisi delle dipendenze esterne rappresenta un passaggio cruciale per comprendere le capacità funzionali e i vettori di attacco disponibili al codice malevolo.

Librerie di Sistema Principali:

1. **KERNEL32.dll** (57 funzioni importate)
   - **Funzionalità:** API principali del kernel di Windows, gestione memoria, processi, file I/O
   - **Significato per il malware:** Accesso diretto alle funzionalità di sistema più sensibili
   - **Potenziale utilizzo malevolo:** Creazione processi, manipolazione memoria, accesso file di sistema, escalazione privilegi

2. **32.dll** (74 funzioni importate)
   - **Funzionalità:** API dell'interfaccia utente Windows, gestione finestre, messaggi
   - **Significato per il malware:** Capacità di manipolazione dell'interfaccia utente e controllo finestre
   - **Potenziale utilizzo malevolo:** Keylogging, screenshot, finestre di phishing, hijacking interfaccia utente

3. **DI32.dll** (24 funzioni importate)
   - **Funzionalità:** Interfaccia grafica Windows, disegno, font, rendering
   - **Significato per il malware:** Capacità di manipolazione grafica e rendering
   - **Potenziale utilizzo malevolo:** Creazione falsi interfacce, watermark dannosi, manipolazione bitmap

4. **DVAPI32.dll** (7 funzioni importate)
   - **Funzionalità:** API avanzate Windows, sicurezza, registro di sistema, servizi
   - **Significato per il malware:** Accesso a funzioni di sistema avanzate e critiche
   - **Potenziale utilizzo malevolo:** Manipolazione registro, installazione servizi, gestione sicurezza, tokens

5. **svcrt.dll** (22 funzioni importate)- **Funzionalità:** Runtime library Microsoft Visual C++
   - **Significato per il malware:** Funzioni standard di programmazione e manipolazione dati
   - **Potenziale utilizzo malevolo:** Manipolazione stringhe, operazioni matematiche, gestione memoria

Librerie di Interfaccia e Sistema:

1. **dlg32.dll** (9 funzioni importate)
   - **Funzionalità:** Dialoghi comuni Windows (Apri/Salva file, Font, Stampa, etc.)
   - **Funzioni specifiche identificate:**
   ◦ `GetOpenFileNameW` , `GetSaveFileNameW` (Gestione file)
   ◦ `ChooseFontW` , `PrintDlgExW` (Interfaccia avanzata)
   ◦ `FindTextW` , `ReplaceTextW` (Ricerca/Sostituzione)
   ◦ **Significato per il malware:** Capacità di inganno utente attraverso interfacce legittime
   ◦ **Potenziale utilizzo malevolo:** Forcing di download file malevoli, esfiltrazione dati attraverso finti dialoghi

2. **HELL32.dll** (4 funzioni importate)
   - **Funzionalità:** Gestione shell di Windows e file system
   - **Significato per il malware:** Capacità di manipolazione del sistema operativo
   - **Potenziale utilizzo malevolo:** Esecuzione comandi shell, accesso file sistema, escalazione privilegi

3. **MCTL32.dll** (1 funzione importata)
   - **Funzionalità:** Controlli comuni interfaccia utente
   - **Significato per il malware:** Capacità di interfacciamento con componenti UI avanzati
   - **Potenziale utilizzo malevolo:** Creazione interfacce sofisticate per l'inganno

4. **INSPOOL.DRV** (3 funzioni importate)
   - **Funzionalità:** Gestione spooler di stampa
   - **Significato per il malware:** Accesso ai servizi di stampa di sistema
   - **Potenziale utilizzo malevolo:** Spooling documenti sensibili, attacchi al sistema di stampa

Analisi delle Dipendenze:

La quantità significativa di funzioni importate da KERNEL32.dll e USER32.dll suggerisce un malware con capacità extensive di manipolazione di sistema e interfaccia utente.

La presenza di ADVAPI32.dll indica potenziali funzionalità di sistema avanzate, mentre comdlg32.dll permette l'interazione sofisticata con l'utente attraverso interfacce apparently innocent.

- 2.2 SEZIONE DELL'ESEGUIBILE

L'analisi dettagliata della struttura delle sezioni del file PE ha rivelato un'architettura complessa e tecnicamente sofisticata, con caratteristiche che suggeriscono l'utilizzo di tecniche di packing o offuscamento avanzate:

Sezioni di Codice:

1. **.text (Prima Istanza)**
   - **Virtual Size:** 0x00007748 (30,536 bytes)
   - **Virtual Address:** 0x00001000
   - **Raw Size:** 0x00007800 (30,720 bytes)
   - **Raw Address:** 0x00000400
   - **Characteristics:** 0x60000020 (EXECUTE_READ)
   - **Funzionalità:** Contiene il codice eseguibile principale
   - **Significato:** Segmento di codice primary del malware, normalmente protetto da scrittura

2. **.text (Seconda Istanza) ANOMALA**
   - **Virtual Size:** 0x0002B6AC (178,604 bytes)
   - **Virtual Address:** 0x00014000
   - **Raw Size:** 0x0002B800 (180,224 bytes)
   - **Raw Address:** 0x00011200
   - **Characteristics:** 0xE0000020 (EXECUTE_READ_WRITE)
   - **Funzionalità:** Codice eseguibile con capacità di scrittura
   - **Significato Critico:** La combinazione EXECUTE_READ_WRITE per una sezione di codice è estremamente sospetta e indica:
     ◦ **Packing/Upacking:** Possibile utilizzo di packer come UPX
     ◦ **Codice Auto-modificante:** Capacità di modificare il proprio codice in runtime
     ◦ **Iniezione di Codice:** Preparazione per injection di shellcode o payload

Sezioni di Dati:

1. **.data- Virtual Size:** 0x00001BAB (6,827 bytes)
   - **Virtual Address:** 0x00009000
   - **Raw Size:** 0x00000800 (2,048 bytes)
   - **Raw Address:** 0x00007000
   - **Characteristics:** 0xC0000040 (INITIALIZED_DATA_READ_WRITE)
   - **Funzionalità:** Variabili globali inizializzate e strutture di dati
   - **Significato:** Contiene configurazioni, chiavi, stringhe e dati di runtime del malware

2. **.idata**- **Virtual Size:** 0x0000113E (4,414 bytes)
   - **Virtual Address:** 0x00040000
   - **Raw Size:** 0x00001200 (4,608 bytes)
   - **Raw Address:** 0x0003CA00
   - **Characteristics:** 0xC2000040 (INITIALIZED_DATA_READ_WRITE)

- **Funzionalità:** Tabella di importazione (Import Address Table)
- **Significato:** Contiene metadati delle librerie e funzioni importate

Sezioni di Risorse:

1. **.rsrc (Prima Istanza)**
   - **Virtual Size:** 0x00008DB4 (36,212 bytes)
   - **Virtual Address:** 0x00008000
   - **Raw Size:** 0x00008E00 (36,352 bytes)
   - **Raw Address:** 0x00008400
   - **Characteristics:** 0x40000040 (INITIALIZED_DATA_READ)

2. **.rsrc (Seconda Istanza)**
   - **Virtual Size:** 0x00008DB0 (36,208 bytes)
   - **Virtual Address:** 0x00042000- Raw Size: 0x00008E00 (36,352 bytes)
   - **Raw Address:** 0x0003DC00
   - **Characteristics:** 0x40000040 (INITIALIZED_DATA_READ)
   - **Funzionalità:** Risorse applicazione (icone, stringhe, dialoghi, bitmap)
   - **Significato:** Contiene elementi di interfaccia utente e risorse multimediali

Anomalie Strutturali Critiche:

La presenza di sezioni duplicate (.text e .rsrc) rappresenta un'anomalia significativa nella struttura PE standard.

Questa caratteristica è tipica di:
- **Eseguibili Packed:** Utilizzo di packer per compressione e offuscamento
- **Multi-stage Malware:** Presenza di loader e payload separati
- **Bypass di Antivirus:** Tecniche per eludere il rilevamento statico

Implicazioni di Sicurezza:

La combinazione di una sezione .text scrivibile ed eseguibile, insieme alle sezioni duplicate, configura un profilo di rischio elevato che richiede analisi dinamica approfondita per comprendere le capacità reali del malware.

PARTE III: FACOLTATIVO - ANALISI AGGIUNTA

Considerazioni Finali sul Malware `notepad-classico.exe`

Basandomi sui dati raccolti attraverso l'analisi statica approfondita, posso fornire considerazioni avanzate sul campione in esame:

Caratteristiche Tecniche Critiche:

Il malware `notepad-classico.exe` presenta un'architettura sofisticata che combina tecniche di evasion avanzate con capacità di manipolazione di sistema extensive.

L'analisi ha rivelato segnali preoccupanti che vanno oltre il semplice mascheramento del nome file:

1. **Tecniche di Packing Identificate:**
   - La presenza di UPX Utility nelle funzionalità di CFF Explorer suggerisce l'utilizzo del packer UPX
   - Sezione .text con caratteristiche EXECUTE_READ_WRITE (0xE0000020) indica codice auto-modificante
   - Doppia presenza di sezioni .text e .rsrc suggerisce multi-stage payload

2. **Capacità di Sistema Avanzate:**
   - 57 funzioni importate da KERNEL32.dll indicano controllo system-level
   - 74 funzioni da USER32.dll suggeriscono capacità di UI manipulation
   - ADVAPI32.dll con 7 funzioni permette manipolazione registro e sicurezza

3. **Vettori di Attacco Diversificati:**
   - **File System Manipulation:** Accesso completo tramite SHELL32.dll e WINSPOOL.DRV
   - **User Interface Deception:** comdlg32.dll per dialoghi falsi e download forzosi
   - **Memory Manipulation:** Capacità di gestione memoria avanzata per code injection

Analisi del Comportamento Potenziale:

Dal punto di vista comportamentale, il malware dimostra pattern tipici di:

1. **Trojanization:** Mascheramento come applicazione legittima (notepad) per evitare detection Multi-stage

2. **Execution:** Struttura packed suggerisce payload principale nascosto System

3. **Persistence:** Dipendenze da ADVAPI32.dll indicano potenziale persistenza nel sistema

4. **User Interface Hijacking:** USER32.dll e comdlg32.dll permettono controllo dell'interfaccia utente

Capacità di Evasion:

L'utilizzo di packer UPX rappresenta una tecnica di primary evasion che:
- Comprime il codice malevolo per ridurre la signature footprint
- Rende l'analisi statica più complessa senza decodifica
- Permette polymorphic behavior durante decompressione
- Elude signature-based detection antivirus

Implicazioni per la Sicurezza Organizzativa:

Dal punto di vista difensivo, questo malware rappresenta una minaccia di livello intermediario-avanzato con le seguenti caratteristiche:

1. **Sophistication Level:** Medio-Alto, utilizzando packer e tecniche di offuscamento
2. **Attack Vectors:** Multipli (file system, UI, network se configurato)
3. **Persistence Capabilities:** Elevate, con accesso alle API di sistema avanzate
4. **Detection Evasion:** Buone, grazie al packing e al mascheramento

Raccomandazioni Tecniche Specifiche:

1. **Network Segmentation:** Isolamento dei sistemi potenzialmente infetti
2. **Behavioral Analysis:** Monitoraggio runtime delle funzioni KERNEL32.dll e USER32.dll
3. **Memory Forensics:** Dump analysis della sezione .text durante execution
4. **Registry Monitoring:** Controllo modifiche alle chiavi di registro di sistema Process
5. **Monitoring:** Osservazione dei processi child generati

Conclusione Tecnica:

`notepad-classico.exe` rappresenta un example representative di malware moderno che combina social engineering (nome falso), technical evasion (packing) e system-level access (extensive API imports). La sua analisi rivela un threat actor con competenze tecniche intermediate-advanced, capace di sviluppare codice che supera i controlli di sicurezza di base.

La presenza di tecniche di packing e la struttura delle sezioni suggerisce che questo malware è parte di una famiglia più ampia o di un toolkit commerciale di malware development, piuttosto che un'esercitazione accademica isolata.

PARTE IV: CONCLUSIONI

L'analisi statica sul campione `notepad-classico.exe` ha rilevato un malware di complety intermedia con caratteristiche tecniche sofisticate che richiedono sttenzione specialized nella strategia di difesa.

L'approccio metodologico adottato, basato sull'utilizzo di CFF Explorer VIII in ambiente FLARE VM, ha permesso l'estrazione di informazioni critical senza esposizione a rischio operazionali.

Risultati Principali dell'Analisi:

1. **Identificazione Librerire:** Mapping completo di 9 librerie DLL con 190 funzioni totali importate, evidenziando capacità extensive di system manipulation

2. **Analisi Sttrutturale:** Identificazione di 6 ssezioni PE, con 2 istanze duplicate (.txt, rsrc) che suggeriscono tecniche di packing avanzate

3. **IOC Critici:** Rilevamento di multiole anomalie strutturali e comportamentali che confermano la natura malevola del campione

Caratteristiche Tecniche Dominanti:

Il malware dimostra un profilo tecnico caratterizzato da:
- Packing Sophistication: Utilizzo di packer (probabilmente UPX) per evasion
- System-Level Access: Dipendenze extensive da KERNEL32.dll e USER32.dll
- UI Manipolation Capabilitiers: Controlllo avanzato dell'interfaccia utente
- Multi-stage Architecture: Struttura che suggerisce payload principale nascosto

Valore dell'Analisi Statica:

L'esame ha dimostrato l'efficacia dell'analisi statica come strumento di inteligence sui malware moderni.

Attraverso l'identificazione di pattern strutturali anomali e l'analisi delle dipendenze, è stato possibile caratterizzare il threat level e le capacità operative del campione senza ricorrere all'esecuzione rischiosa.

Implicazione Strategiche:

Dal punto di vista della cybersecurity, questo malware rappresenta un example paradigmatico delle minacce moderne che combinano social engineering con technical sophistication.

Le tecniche identificate sono compatibili con campagne di cyber espionage e rappresentano un rischio significativo per organizzazioni che non implementano defense-in-depth strategies.

Riflessioni Metodologiche:

L'analisi conferma l'importanza della preparation tecnica nel campo della malware analysis.

La capacità di interpretare correttamente le anomalie strutturali PE e di correlare multiple indicators di compromise rappresenta una competenza fundamental per i professionisti della sicurezza moderna.

Il malware in questione, con le sue caratteristiche di packing e multi-stage architecture, sottolinea l'evoluzione delle tecniche di threat actor moderni e la necessità di approcci di detection multi-layered che vadano oltre il simple signature-based analysis.

Preparazione per Analisi Dinamica:

I findings statici identificano diverse aree prioritarie per l'analisi dinamica:
- Monitoraggio delle funzioni KERNEL32.dll (CreateThread, WriteFile, etc.)
- Osservazione delle modifiche sistema durante decompressione del packer
- Analysis delle interazioni con l'interfaccia utente per pattern di social engineering
- Memory forensics durante le fasi di unpacking

Raccomandazioni Finali:

1. **Environment Setup:** Continuare l'analisi in ambiente FLARE completamente isolato
2. **Tool Integration:** Combinare analisi statica con dynamic analysis tools
3. **IOC Development:** Creare detection rules basate sui findings statici
4. **Threat Intelligence:** Correlare il sample con threat intelligence databases
5. **Defense Adaptation:** Aggiornare le detection capabilities basandosi sui patterns identificati

L'analisi di `notepad-classico.exe` rappresenta un case study valuable per comprendere le tecniche di sviluppo malware moderno e per affinare le competenze di reverse engineering necessarie per la defense cybersecurity moderna.