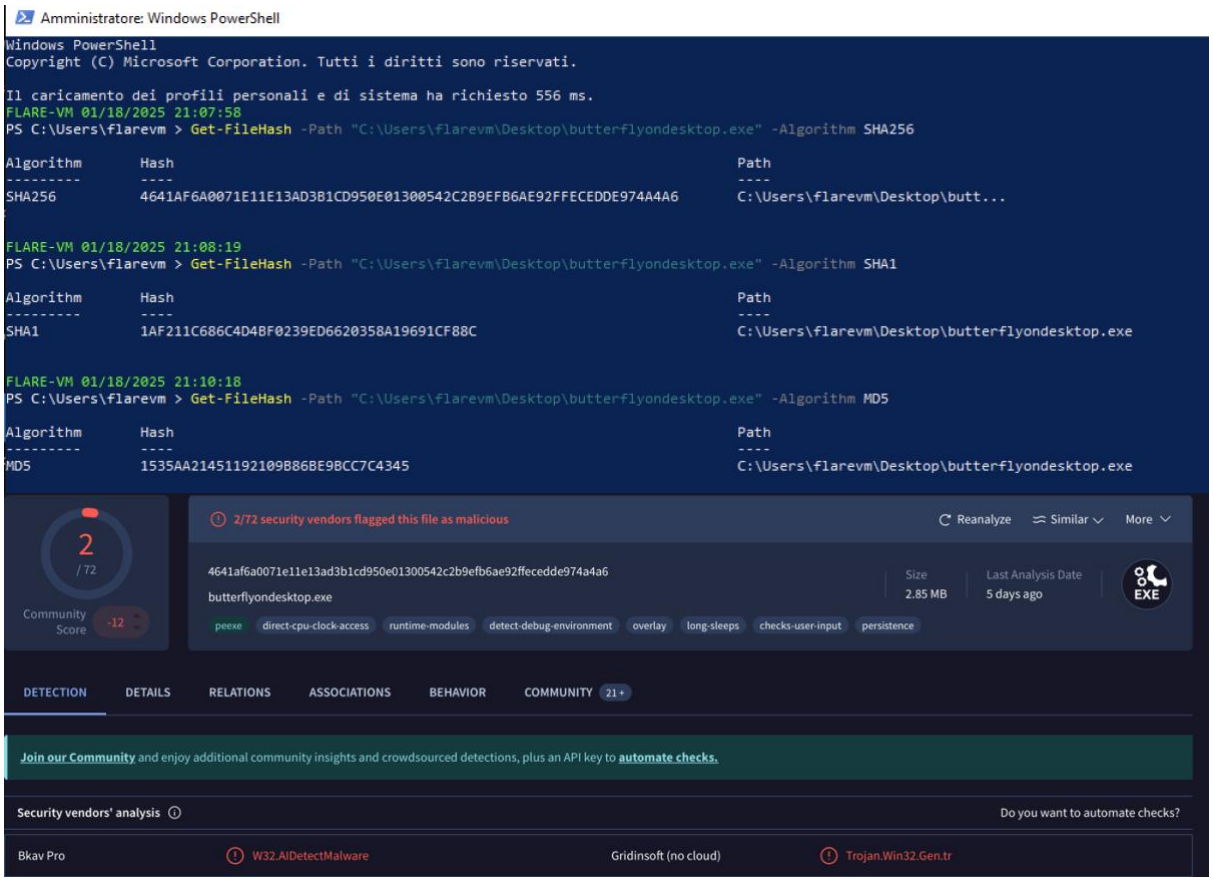
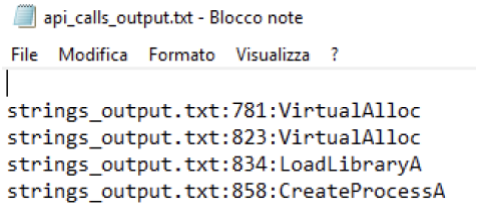
RAPPORTO

Static Analysis

Per prima cosa ho generato diversi hash del malware e poi ho cercato SHA256 in Virus Total. Mi ha mostrato una possibilità di Trojan:



Poi ho estratto le stringhe leggibili in un file di testo. Ho usato il file per filtrare le API call, gli URL e le chiavi di registro:



Risultati principali dall'output di stringhe:

1. API call

• VirtualAlloc (righe 781, 823): comunemente utilizzato nei malware per allocare memoria durante l'esecuzione.

• LoadLibraryA (riga 834): suggerisce che il malware carica dinamicamente le librerie (possibilmente DLL).

• CreateProcessA (riga 858): indica la creazione di processi, potenzialmente per l'avvio di altri processi dannosi.

• GetCommandLineA (righe 790, 852): utilizzato per recuperare argomenti della riga di comando; il malware potrebbe basarsi su argomenti specifici per l'esecuzione.

2. Metadati incorporati

• Riga 1039: "Drive Software Company": potrebbe essere il nome originale o contraffatto dello sviluppatore.

• Righe 1053–1089: i metadati XML incorporati fanno riferimento a:

o Microsoft.Windows.Common-Controls (riga 1064): indica la dipendenza dai controlli Windows comuni. o tag dpiAware (riga 1081): suggerisce un adattamento per display ad alta risoluzione, indicando forse un tentativo di apparire legittimo.

3. Messaggi di errore

• Righe 556–597: messaggi di errore come ECompressError e lzmadecompsmall: i dati compressi sono danneggiati (%d) puntano a routine di compressione/decompressione. Ciò potrebbe significare che il malware utilizza payload o dati compressi.

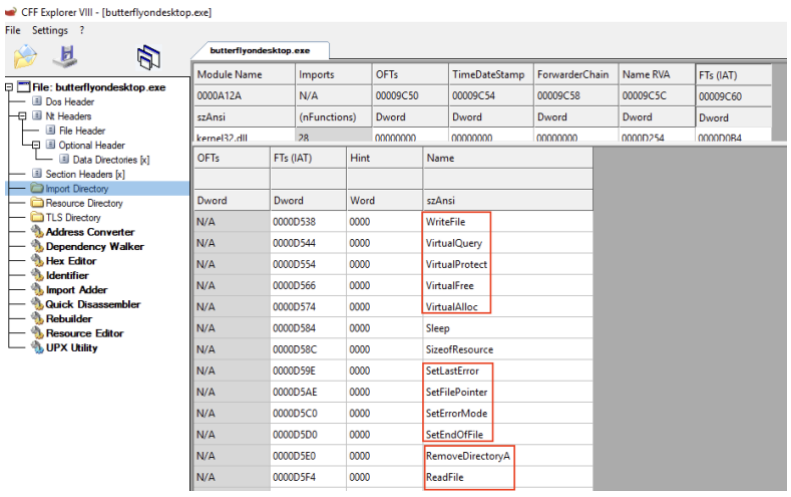
4. Stringhe sospette

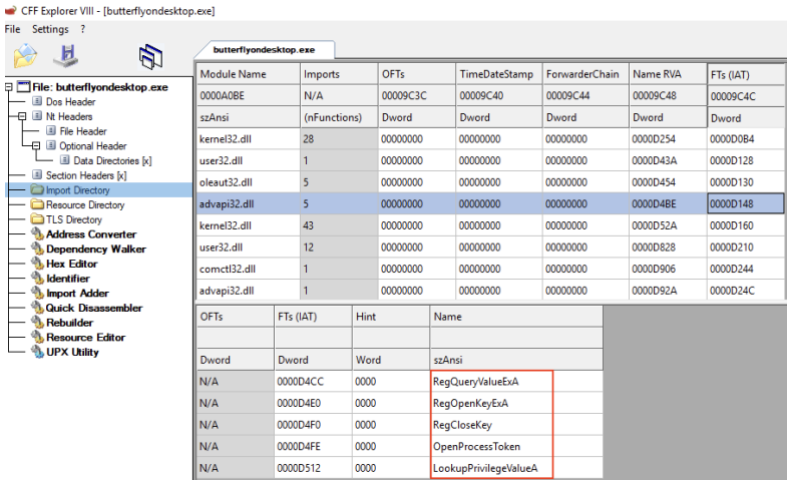
• Righe 72441, 85207: stringhe dall'aspetto offuscato come 5NqORG e oZ3oRG. Potrebbero essere dati codificati (ad esempio, Base64 o codifica personalizzata).

5. URL

• http://schemas.microsoft.com/SMI/2005/WindowsSettings (riga 1081): sembra benigno ma richiede una convalida per confermare che non sia stato abusato.

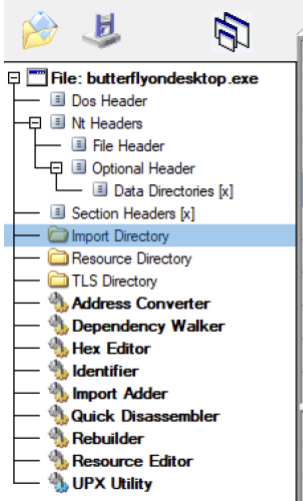
Poi ho analizzato le strutture dei file utilizzando CFF Explorer:





CFF Explorer VIII - [butterflyondesktop.exe]

File Settings ?



- File: butterflyondesktop.exe
  - Dos Header
  - Nt Headers
    - File Header
    - Optional Header
      - Data Directories [x]
  - Section Headers [x]
  - Import Directory
  - Resource Directory
  - TLS Directory
  - Address Converter
  - Dependency Walker
  - Hex Editor
  - Identifier
  - Import Adder
  - Quick Disassembler
  - Rebuilder
  - Resource Editor
  - UPX Utility

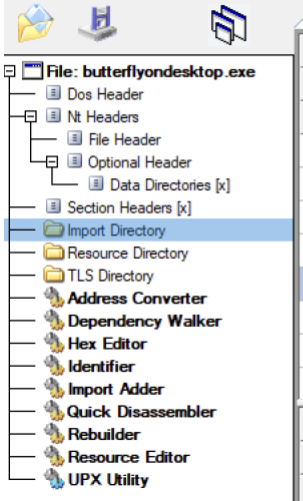
butterflyondesktop.exe

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
0000A03A	N/A	00009C14	00009C18	00009C1C	00009C20	00009C24
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
kernel32.dll	28	00000000	00000000	00000000	0000D254	0000D0B4
user32.dll	1	00000000	00000000	00000000	0000D43A	0000D128
oleaut32.dll	5	00000000	00000000	00000000	0000D454	0000D130
advapi32.dll	5	00000000	00000000	00000000	0000D4BE	0000D148
kernel32.dll	43	00000000	00000000	00000000	0000D52A	0000D160
user32.dll	12	00000000	00000000	00000000	0000D828	0000D210
comctl32.dll	1	00000000	00000000	00000000	0000D906	0000D244
advapi32.dll	1	00000000	00000000	00000000	0000D92A	0000D24C

OFTs	FTs (IAT)	Hint	Name
Dword	Dword	Word	szAnsi
N/A	0000D446	0000	MessageBoxA

CFF Explorer VIII - [butterflyondesktop.exe]

File Settings ?



- File: butterflyondesktop.exe
  - Dos Header
  - Nt Headers
    - File Header
    - Optional Header
      - Data Directories [x]
  - Section Headers [x]
  - Import Directory
  - Resource Directory
  - TLS Directory
  - Address Converter
  - Dependency Walker
  - Hex Editor
  - Identifier
  - Import Adder
  - Quick Disassembler
  - Rebuilder
  - Resource Editor
  - UPX Utility

butterflyondesktop.exe

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
0000A12A	N/A	00009C50	00009C54	00009C58	00009C5C	00009C60
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
kernel32.dll	28	00000000	00000000	00000000	0000D254	0000D0B4
user32.dll	1	00000000	00000000	00000000	0000D43A	0000D128
oleaut32.dll	5	00000000	00000000	00000000	0000D454	0000D130
advapi32.dll	5	00000000	00000000	00000000	0000D4BE	0000D148
kernel32.dll	43	00000000	00000000	00000000	0000D52A	0000D160
user32.dll	12	00000000	00000000	00000000	0000D828	0000D210
comctl32.dll	1	00000000	00000000	00000000	0000D906	0000D244
advapi32.dll	1	00000000	00000000	00000000	0000D92A	0000D24C

OFTs	FTs (IAT)	Hint	Name
Dword	Dword	Word	szAnsi
N/A	0000D6F8	0000	GetLastError
N/A	0000D708	0000	GetFullPathNameA
N/A	0000D71C	0000	GetFileSize
N/A	0000D72A	0000	GetFileAttributesA
N/A	0000D740	0000	GetExitCodeProcess
N/A	0000D756	0000	GetEnvironmentVariableA
N/A	0000D770	0000	GetCurrentProcess
N/A	0000D784	0000	GetCommandLineA
N/A	0000D796	0000	GetACP
N/A	0000D7A0	0000	InterlockedExchange
N/A	0000D7B6	0000	FormatMessageA
N/A	0000D7C8	0000	FindResourceA
N/A	0000D7D8	0000	DeleteFileA
N/A	0000D7E6	0000	CreateProcessA
N/A	0000D7F8	0000	CreateFileA
N/A	0000D806	0000	CreateDirectoryA
N/A	0000D81A	0000	CloseHandle

Principali risultati di CFF Explorer:

1. Funzioni correlate alla gestione della memoria:

• VirtualAlloc, VirtualFree, VirtualProtect: queste sono funzioni comuni utilizzate per l'allocazione e la protezione dinamica della memoria, spesso utilizzate dal malware per allocare memoria per shellcode o altri payload. Se il malware utilizza queste funzioni, potrebbe allocare ed eseguire codice in memoria.

• VirtualQuery: può essere utilizzata per interrogare le informazioni sulla memoria. Ciò potrebbe indicare un tentativo di raccogliere informazioni sulla memoria o evitare il rilevamento.

2. Funzioni correlate alle operazioni sui file:

• WriteFile, ReadFile, SetFilePointer, SetEndOfFile: queste sono funzioni standard per interagire con i file. Il malware potrebbe utilizzarle per scrivere payload su disco, leggere file o modificare i puntatori dei file per nascondere le proprie tracce.

• RemoveDirectoryA: questa funzione elimina le directory. Se utilizzata dal malware, potrebbe eliminare prove o file dannosi. • SizeofResource, LoadResource, LockResource, LoadLibraryA: queste funzioni sono correlate alla gestione delle risorse e potrebbero indicare che il malware sta caricando risorse (probabilmente il suo payload) dall'interno dell'eseguibile o di altre risorse.

3. Funzioni correlate all'interfaccia utente di Windows e alla gestione dei messaggi:

• MessageBoxA, MessageBoxW: queste funzioni mostrano le caselle dei messaggi, che potrebbero essere utilizzate dal malware per l'interazione con l'utente o per visualizzare avvisi o messaggi falsi all'utente.

• CreateWindowExA, DestroyWindow, SetWindowLongA, DispatchMessageA: queste sono funzioni per creare, gestire e gestire le finestre. Il malware potrebbe utilizzarle per creare finestre nascoste o false per scopi dannosi.

4. Funzioni del registro:

• RegOpenKeyExA, RegQueryValueExA, RegCloseKey: queste funzioni vengono utilizzate per interagire con il registro di Windows. Il malware le utilizza comunemente per verificare determinate chiavi del registro, creare nuove chiavi del registro (per la persistenza) o modificare le impostazioni per mantenere la propria presenza nel sistema. 5. Funzioni di escalation dei privilegi e di sicurezza:

• OpenProcessToken, LookupPrivilegeValueA, AdjustTokenPrivileges: queste funzioni sono correlate alla modifica dei privilegi utente. Il malware può utilizzare queste funzioni per elevare i privilegi e ottenere l'accesso amministrativo.

6. Funzioni di sincronizzazione dei thread:

• DeleteCriticalSection, EnterCriticalSection, LeaveCriticalSection, InitializeCriticalSection: queste sono correlate alla sincronizzazione delle sezioni critiche. Queste funzioni aiutano a gestire i thread e possono essere utilizzate dal malware per controllare o bloccare l'accesso alle risorse condivise in un ambiente multithread.

7. Gestione dei processi e delle esecuzioni:

• CreateWindowExA, ExitWindowsEx, PeekMessageA, MsgWaitForMultipleObjects: queste funzioni sono correlate alla gestione di finestre o messaggi nella GUI e possono essere utilizzate dal malware per manipolare la GUI del sistema o mantenersi in esecuzione in background senza interazione dell'utente. 8. Altre importazioni sospette varie:

• Sleep: questa funzione mette in pausa l'esecuzione del programma. Il malware potrebbe usarla per ritardare le sue azioni o per evitare il rilevamento da parte di sandbox o software antivirus.

• IsDBCSLeadByte: questa funzione è usata per l'elaborazione del set di caratteri e potrebbe essere usata dal malware per manipolare o nascondere stringhe specifiche, il che potrebbe essere parte di tattiche di evasione.

Volevo usare GHIDRA per disassemblare il binario per un'ulteriore analisi, sfortunatamente il portatile non è riuscito a procedere e ha mostrato errori fatali. Per prova, i log degli errori sono riportati di seguito:

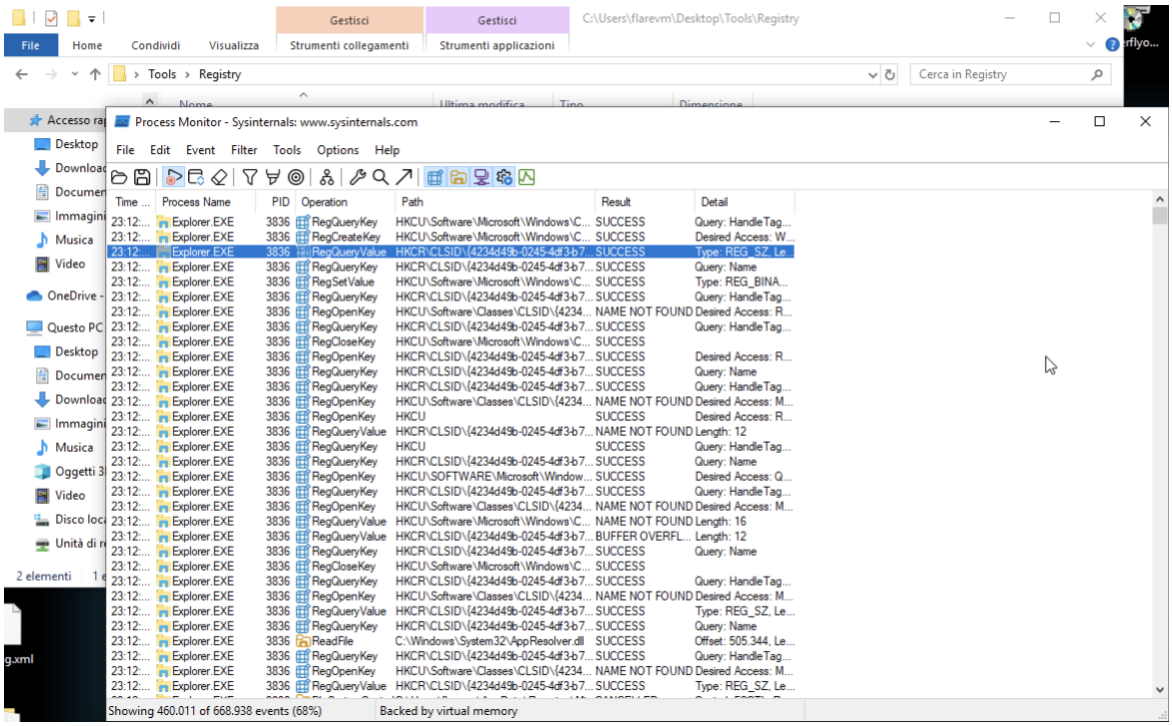

hs_err_pid3880.log - Blocco note

File Modifica Formato Visualizza ?

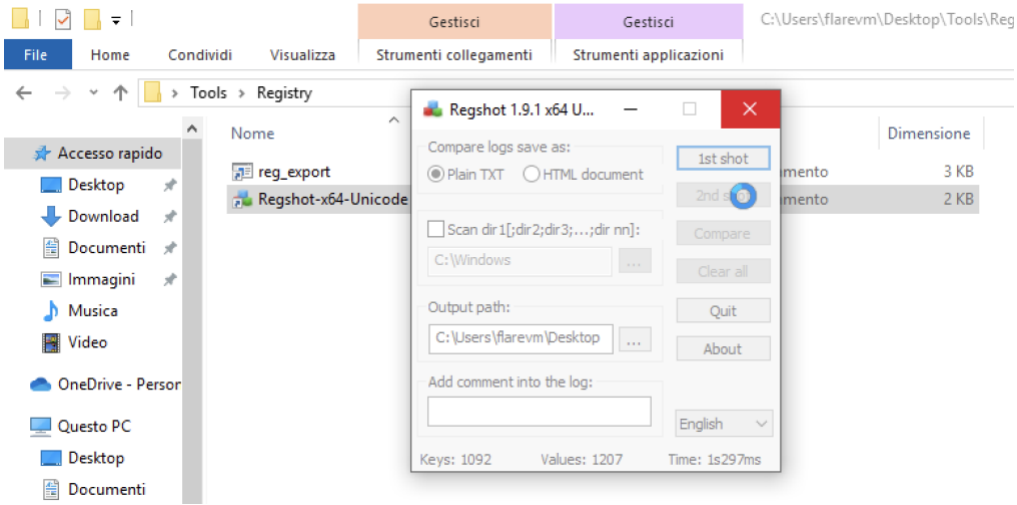
```
#
# A fatal error has been detected by the Java Runtime Environment:
#
#  EXCEPTION_ACCESS_VIOLATION (0xc0000005) at pc=0x000000000b95ba40, pid=3880, tid=1108
#
# JRE version: OpenJDK Runtime Environment (21.0.1+12) (build 21.0.1+12-29)
# Java VM: OpenJDK 64-Bit Server VM (21.0.1+12-29, mixed mode, tiered, compressed oops, compressed class ptrs, g1 gc, windows-amd64)
# Problematic frame:
# J 6648 c1 sun.security.provider.MD5.GG(IIIIIII)I java.base@21.0.1 (27 bytes) @ 0x000000000b95ba40 [0x000000000b95ba40+0x0000000000000000]
#
# No core dump will be written. Minidumps are not enabled by default on client versions of Windows
#
# If you would like to submit a bug report, please visit:
#   https://bugreport.java.com/bugreport/crash.jsp
#

---------------  S U M M A R Y ------------

Command Line: -Duser.home=C:\Users\flarevm -Djava.system.class.loader=ghidra.GhidraClassLoader -Dfile.encoding=UTF8 -Duser.country=US -Duser.language=en -Duser.variant= -Dsun.

Host: Intel Core Processor (Skylake), 5 cores, 4G,  Windows 10 , 64 bit Build 17763 (10.0.17763.1)
Time: Sat Jan 18 22:33:50 2025 ora solare Europa occidentale elapsed time: 206.952195 seconds (0d 0h 3m 26s)

---------------  T H R E A D  ---------------

Current thread (0x000000002687eeb0):  JavaThread "Task - Import" daemon [_thread_in_Java, id=1108, stack(0x000000002bd10000,0x000000002be10000) (1024K)]

Stack: [0x000000002bd10000,0x000000002be10000],  sp=0x000000002be0ead8,  free space=1018k
Native frames: (J=compiled Java code, j=interpreted, Vv=VM code, C=native code)
C  0x000000000b95ba40

The last pc belongs to nmethod (printed below).
```

Dynamic Analysis

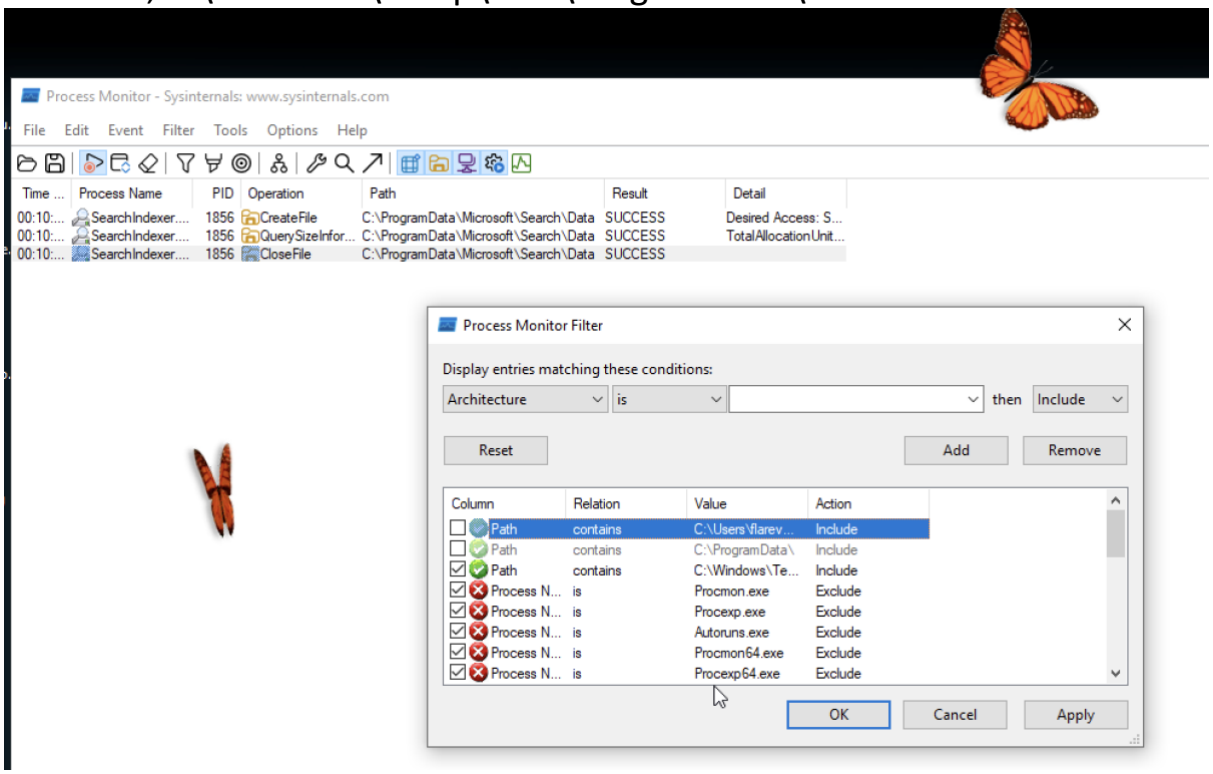
Ho iniziato accedendo allo strumento Process Monitor:



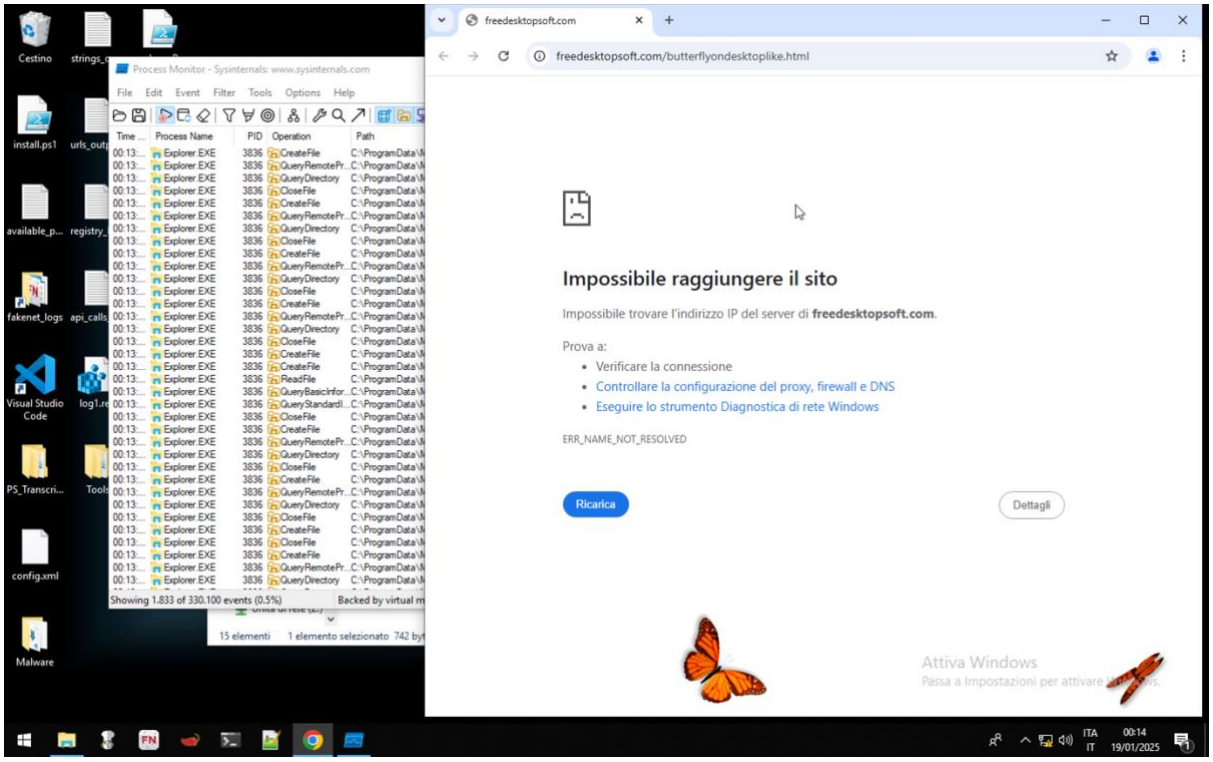
Poi ho effettuato l'accesso allo strumento Regshot, ho scattato il primo scatto e l'ho salvato sul Desktop. Volevo scattare il secondo scatto dopo aver avviato il Malware, ma sfortunatamente lo strumento si è bloccato. Se avesse funzionato, avrei confrontato il primo e il secondo scatto, per monitorare il cambiamento nel sistema:



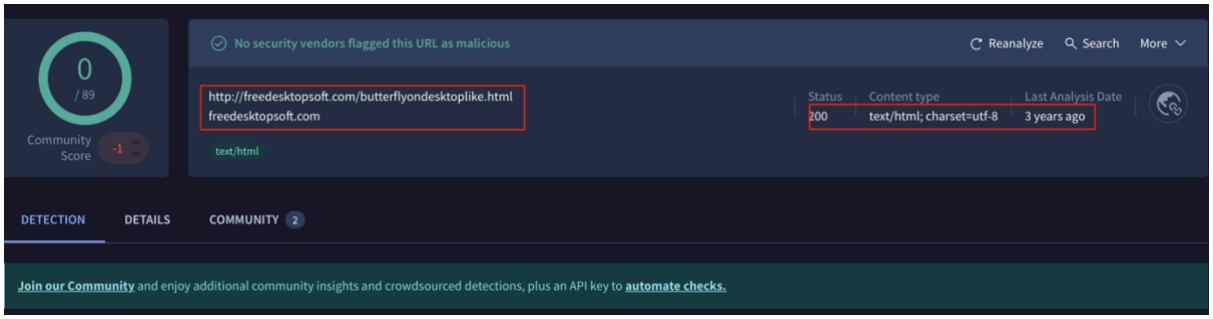
Quindi ho deciso di usare lo strumento che avevo aperto per primo, Process Monitor. L'ho filtrato per gli eventi contenenti i percorsi del malware, C:\Windows\Temp\ e C:\ProgramData\:



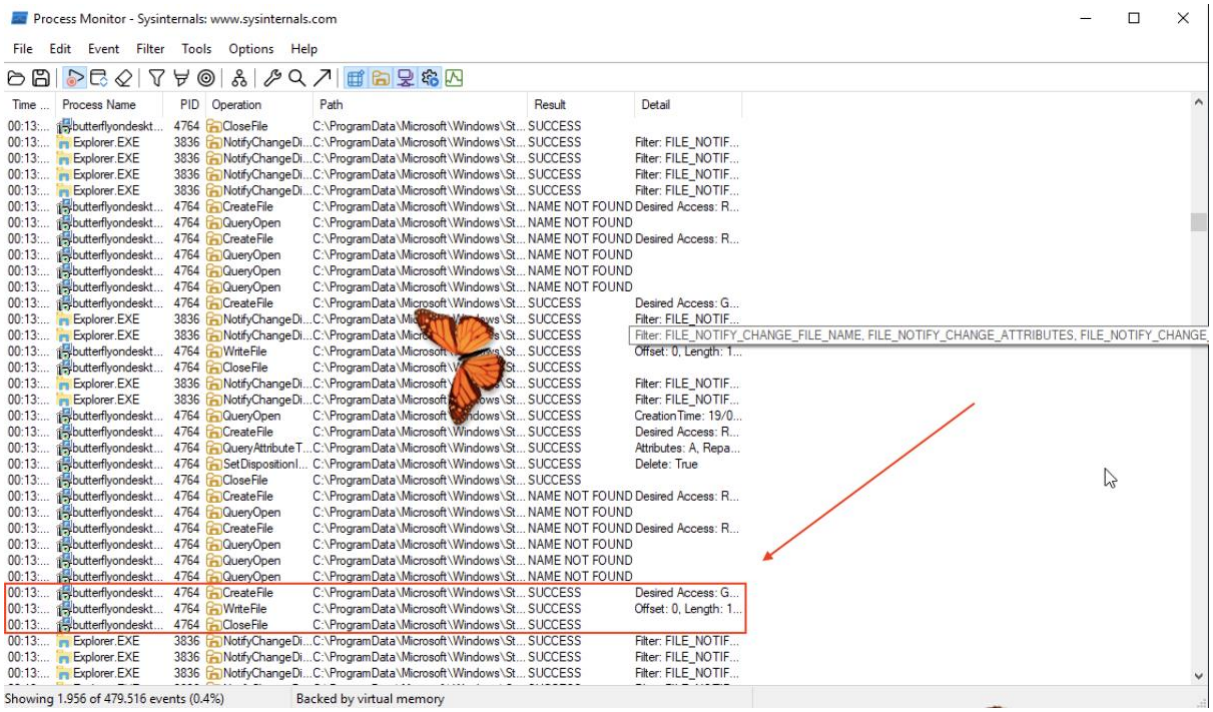
Poi stranamente è stata aperta una pagina web sul browser Chrome:



Quindi ho copiato l'URL e l'ho eseguito su Virus Total per analizzare qualsiasi traccia di malware, tuttavia tutto sembra ok. Era accessibile con il codice di stato 200. Quell'URL è stato scansionato 3 anni fa l'ultima volta..:

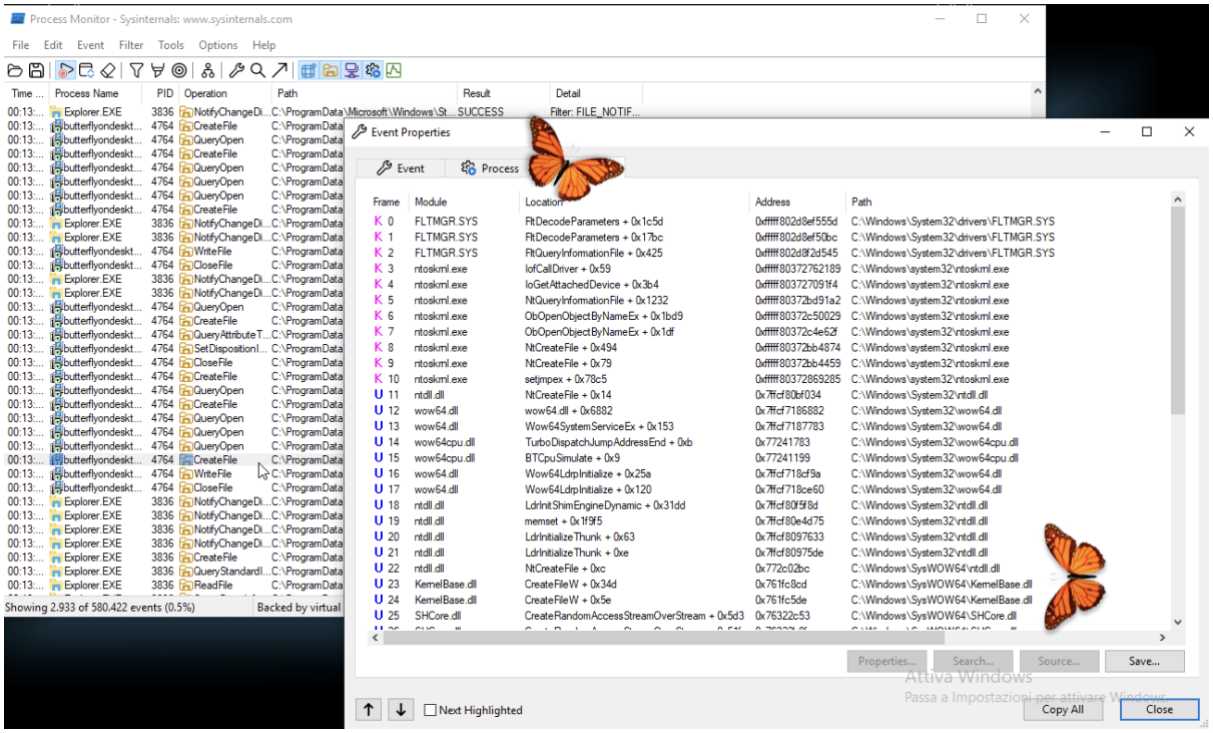


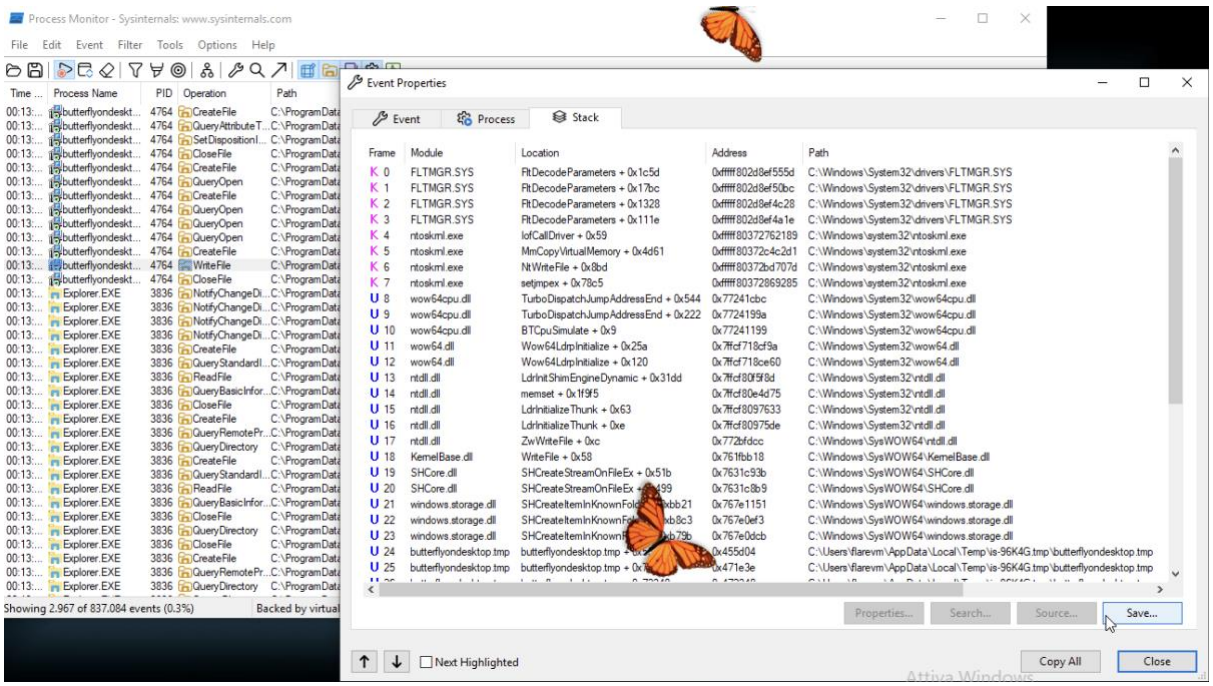
Poi sono tornato a Process Monitor e ho iniziato a cercare qualcosa di particolare e ho trovato:



1. Posizione del collegamento: la cartella C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Butterfly sul Desktop fa parte del sistema del menu Start in Windows. La directory ProgramData è comunemente utilizzata dai programmi per archiviare dati accessibili a tutti gli utenti. Poiché il malware ha creato un collegamento in questa directory, potrebbe potenzialmente essere eseguito all'avvio per tutti gli utenti.

2. Collegamento di disinstallazione: il file si chiama "Uninstall Butterfly on Desktop.lnk", che in genere indica che potrebbe essere stato progettato per apparire come un legittimo programma di disinstallazione per il malware. Il malware spesso crea questo tipo di file per mascherare la sua vera natura o per dare l'impressione di una "uscita pulita", nel caso in cui l'utente provi a rimuoverlo manualmente.





Il fatto che il malware abbia creato una scorciatoia indica che sta tentando di funzionare automaticamente dopo il riavvio del sistema. Ciò potrebbe anche suggerire che il malware potrebbe avere un meccanismo autoreplicante che posiziona i file in altre directory di sistema o modifica il registro.