

MALWARE ANALYSIS

TASK

1. Identificare le librerie importate dal file eseguibile
2. Identificare le sezioni di cui si compone il file eseguibile
3. Identificare i costrutti noti in linguaggio Assembly
4. Ipotesizzare il comportamento delle funzionalità implementate
5. Spiegare qualche istruzione assembly complessa
6. Convincere il dipendente neo assunto che il file IEXPLORER.EXE non è un file maligno

TASK 1: IDENTIFICARE LE LIBRERIE IMPORTATE DAL FILE ESEGUIBILE

Dato il file eseguibile oggetto d'interesse è Malware_U3_W2_L5.exe, iniziamo un processo di Malware Analysis sullo stesso per indagarne e studiarne il comportamento.

MALWARE ANALYSIS è una procedura complessa che coinvolge l'insieme di competenze e tecniche utilizzate dagli esperti di sicurezza informatica **per indagare accuratamente un malware al fine di studiare e capire esattamente il suo comportamento per poi rimuoverlo dal sistema**. Queste competenze sono di vitale importanza per i membri tecnici del **CSIRT** (Computer Security Incident Response Team) durante la gestione degli incidenti di sicurezza

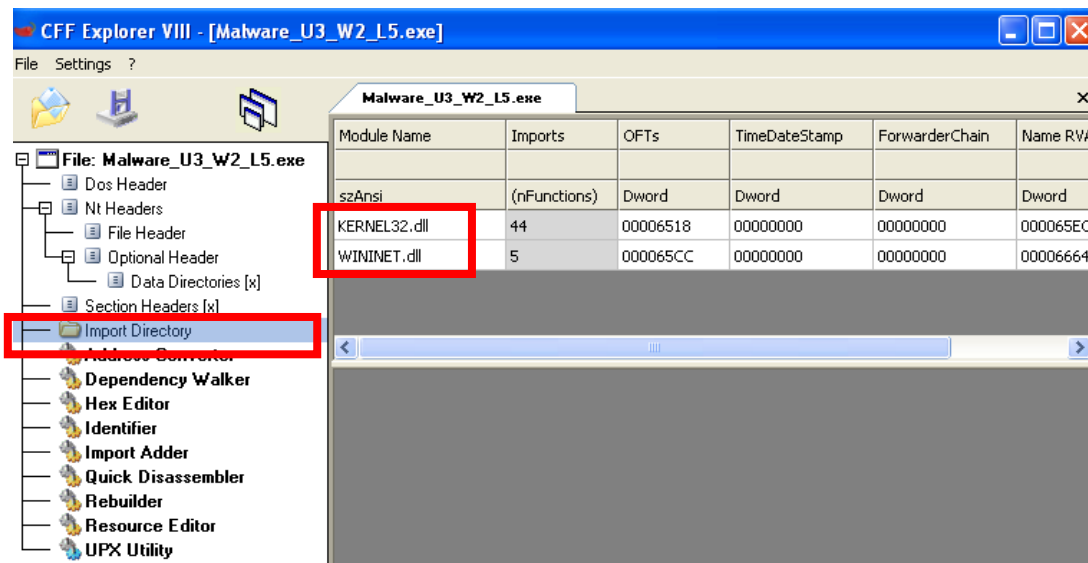
Per queste prime task ci serviremo dell'**ANALISI STATICA BASICA**, che consiste **nell'esaminare un file eseguibile senza tener conto delle istruzioni che lo compongono**, al fine di:

- **confermare la natura malevola o meno** del file oggetto di interesse
- **fornire informazioni generiche circa le sue funzionalità**

A tal proposito, ci serviremo del tool **CFF EXPLORER VIII**, il quale ci consente di caricare un file eseguibile (nel nostro caso **Malware_U3_W2_L5.exe**) per analizzarne l'**HEADER**.

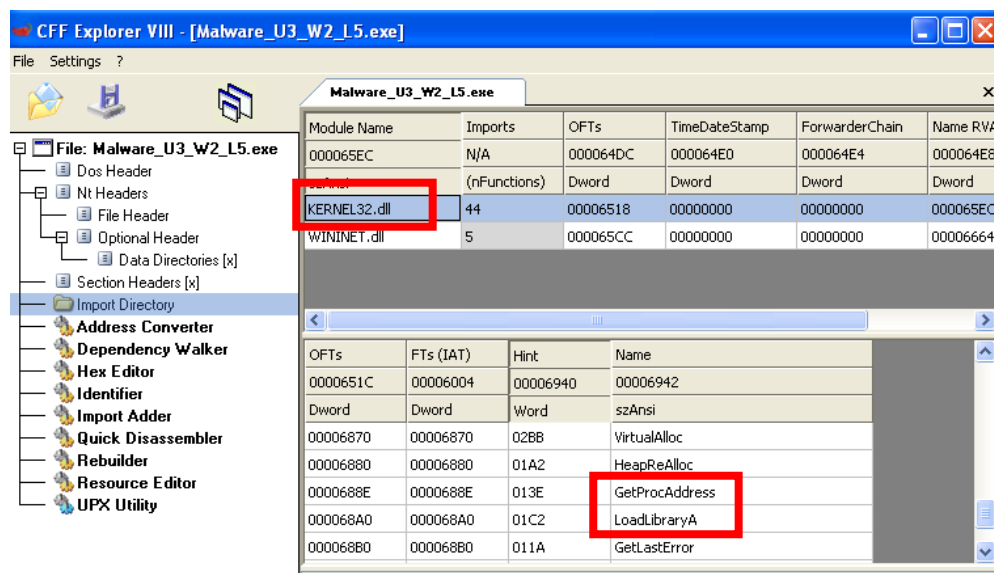
Selezionando la tab **IMPORT DIRECTORY**, il tool ci restituisce librerie importate nel malware:

- **kernel32.dll**: libreria usata per le funzioni principali per interagire con il sistema operativo. Un malware potrebbe sfruttare tale libreria per manipolare i file e per accedere la gestione della memoria
- **wininet.dll**: libreria che contiene le funzioni per l'implementazione di alcuni protocolli di rete come http, FTP, NTP. Un malware potrebbe utilizzare le funzioni presenti nella libreria per comunicare con server remoti, trasferire file o inviare dati sensibili senza il consenso dell'utente

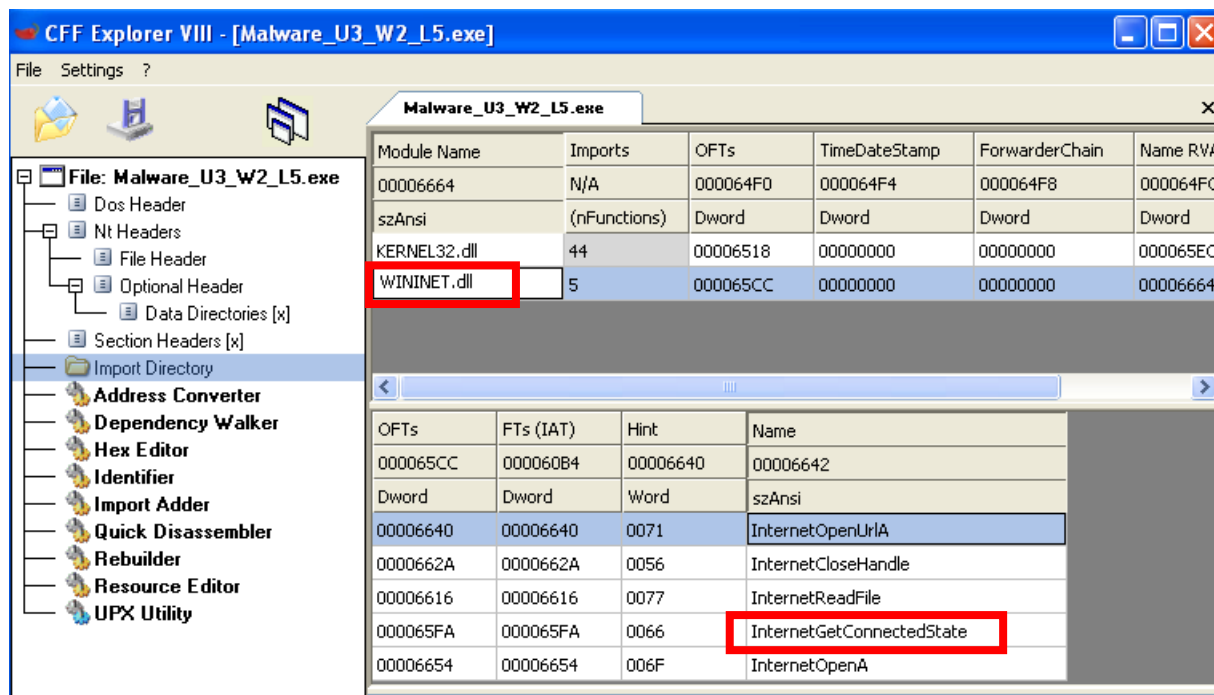


Indagando nello specifico, selezionando la libreria **KERNEL 32.DLL**, è possibile notare che al suo interno vi sono le funzioni **LoadLibraryA** e **GetProcAddress**, che permettono di importare le funzioni della libreria a tempo di esecuzione (runtime). Ciò significa che l'eseguibile richiama la libreria solo quando ha bisogno di utilizzare una specifica funzione.

Questo è un comportamento tipico dei malware, i quali, attraverso questo meccanismo, cercano di risultare meno invasivi e rilevabili.

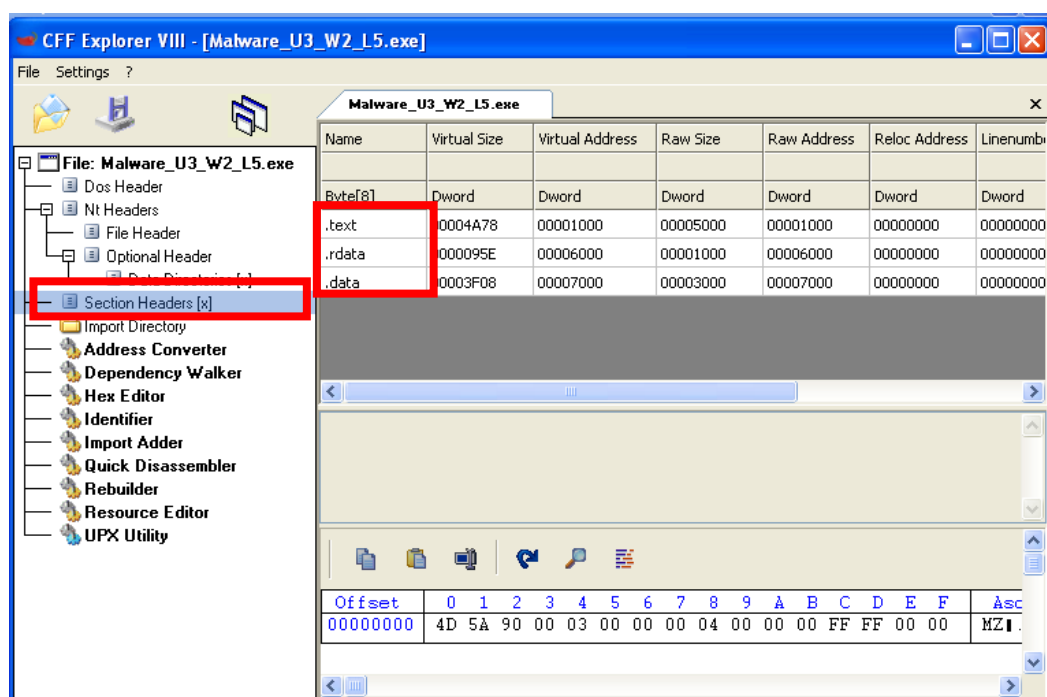


Equal procedimento per la libreria **wininet.dll**, dove troviamo ad esempio la funzione **InternetGetConnectedState**, la quale verifica se la macchina infetta ha accesso o meno ad Internet.



TASK 2: IDENTIFICARE LE SEZIONI DI CUI SI COMPONE IL FILE ESEGUIBILE

Restando all'interno del tool **CFF EXPLORER**, spostandoci nella tab **SECTION HEADER**, abbiamo la possibilità di identificare le sezioni di cui si compone il file oggetto d'interesse.



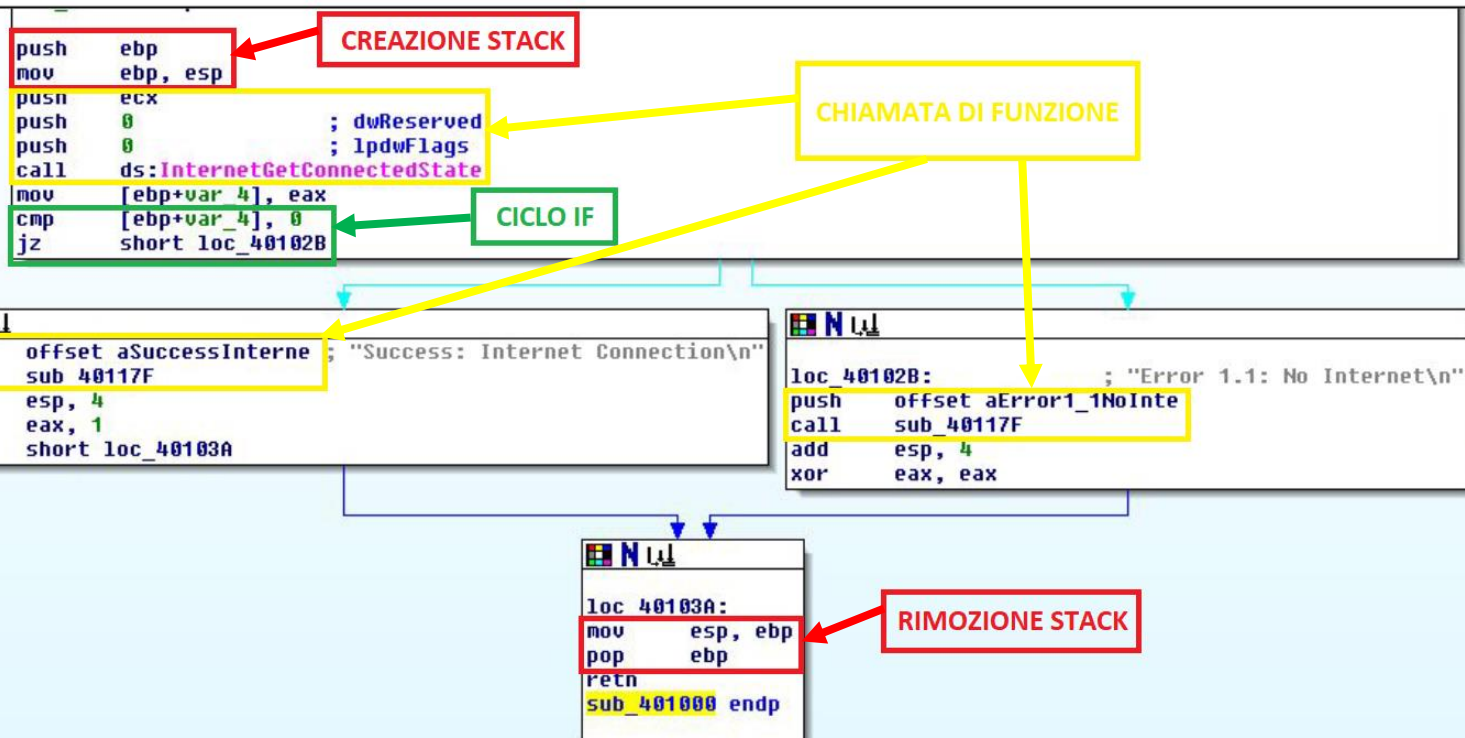
Il file eseguibile oggetto d'interesse è composto da tre sezioni, non compresse da UPX:

- **.text**: sezione che contiene le istruzioni, ovvero le righe di codice che la CPU eseguirà quando il software viene avviato. È la sezione principale di un file eseguibile, poiché contiene il codice effettivo che viene eseguito per far funzionare il programma. Tutte le altre sezioni contengono dati o informazioni di supporto per questa sezione.
- **.rdata**: sezione che contiene informazioni sulle librerie e le funzioni importate ed esportate dall'eseguibile. Qui vengono memorizzate le informazioni sui moduli esterni che l'eseguibile utilizza, come librerie di sistema o librerie condivise, e le funzioni che vengono importate o esportate per l'utilizzo all'interno del programma.
- **.data**: sezione che contiene dati e variabili globali del programma eseguibile. Le variabili definite in questa sezione sono accessibili da qualsiasi parte del programma, poiché sono globalmente dichiarate.

Byte[8]	Dword	Dword	Dword	Dword	Dword	Dword
.text	00004A78	00001000	00005000	00001000	00000000	00000000
.rdata	0000095E	00006000	00001000	00006000	00000000	00000000
.data	00003F08	00007000	00003000	00007000	00000000	00000000

TASK 3: IDENTIFICARE I COSTRUTTI NOTI IN LINGUAGGIO ASSEMBLY E TASK

- 1) Push ebp
- 2) Mov ebp, esp
- 3) Push ecx
- 4) Push 0 ; dwReserved
- 5) Push 0 ; lpdwFlags
- 6) Call ds: InternetGetConnectedState
- 7) Mov [ebp+var_4], eax
- 8) Cmp [ebp+var_4], 0
- 9) Jz short loc_40102B
- 10) Push offset aSuccessInterne ; "Success: Internet Connection\n"
- 11) Call sub_40117F
- 12) Add esp, 4
- 13) Mov eax, 1
- 14) Jmp short loc_40103A
- 15) Loc_40102B:
- 16) Push offset aError1_1NoInte
- 17) Call sub_40117F
- 18) Add esp, 4
- 19) Xor eax, eax
- 20) Loc 40103°:
- 21) Mov esp, ebp
- 22) Pop ebp
- 23) Retn
- 24) Sub_401000 endp



TASK 5: SPIEGARE QUALCHE ISTRUZIONE ASSEMBLY COMPLESSA

N.B. Ho spiegato i costrutti noti, in quanto credo siano queste le istruzioni un po' più complesso rispetto a tutte le altre righe del codice

PRIMO COSTRUTTO NOTO: CREAZIONE DELLO STACK

- 1) **Push ebp**
- 2) **Mov ebp, esp**

- Le prime due righe di codice servono a creare uno stack per le variabili locali: notiamo la presenza dei due puntatori EBP (Extended Base Pointer) ed ESP (Extended Stack Pointer) che puntano rispettivamente alla base ed alla cima dello stack

SECONDO COSTRUTTO NOTO: FUNZIONE INTERNETGETCONNECTEDSTATE

- 3) **Push ecx**
- 4) **Push 0 ; dwReversed**
- 5) **Push 0 ; lpdwFlags**
- 6) **Call ds: InternetGetConnectedState**

- Le prime tre istruzioni "**push**" vengono utilizzate per mettere in cima allo stack tre parametri (**ecx, 0 e 0**) che saranno successivamente passati alla funzione **InternetGetConnectedState**. Questa funzione ha il compito di verificare se il computer ha accesso a Internet

TERZO COSTRUTTO NOTO: IF STATEMENT

- 8) **Cmp [ebp+var_4], 0**
- 9) **Jz short loc_40102B**

- Notiamo per prima cosa la presenza di un'istruzione **cmp**, che confronta (facendo la differenza) la variabile scritta nel registro **EBP+var_4** e **0**.
 - 1) Se il risultato di questa operazione dà come valore **0**, la ZF (Zero Flag) assume valore **1**, e la condizione **jz (Jump Zero)** viene confermata e pertanto viene fatto un salto alla locazione di memoria con indirizzo **40102B**.
 - 2) Se il risultato dell'operazione è diverso da **0**, la ZF assume valore **0** ed il programma continuerebbe ad eseguire le righe di codice successive, fino ad arrivare all'indirizzo di memoria **40103A**, la quale ci rimanderà alla rimozione dello stack. In questo caso, data l'istruzione **Jump short loc_40103A**, siamo in presenza di un salto non condizionale, che verrà dunque sempre eseguito.

QUARTO E QUINTO COSTRUTTO NOTO: CHIAMATA DI FUNZIONE

```
10) Push offset aSuccessInterne ; "Success: Internet Connection\n"  
11) Call sub_40105F
```

```
16) Push offset aError1_1NoInte  
17) Call sub_40117F
```

- Come detto in precedenza, se la Zero Flag assume valore 0, verrà stampato a schermo "Success: Internet Connection (riga 10 e riga 11); viceversa, se la ZF assume valore 1, verrà stampato a schermo Error: No Internet

SESTO COSTRUTTO NOTO: RIMOZIONE DELLO STACK

```
21) Mov esp, ebp  
22) Pop ebp
```

- Viene copiato il contenuto del registro EBP nel registro ESP, e in seguito si passa all'eliminazione del registro ebp dallo stack.

TASK 4: IPOTIZZARE IL COMPORTAMENTO DELLE FUNZIONALITÀ IMPLEMENTATE

A seguito dell'analisi del codice assembly oggetto d'interesse, abbiamo notato il programma, tramite la funzione **InternetGetConnectedState**, verifica se la macchina target ha o meno una connessione ad Internet.

Dopo tale verifica, il **programma**, tramite la stampa di messaggi a schermo, ci fornirà in output un feedback positivo in caso di connessione avvenuta, o un feedback negativo in caso di mancata connessione.

Si specifica che, in caso di mancata connessione, il programma re-inizializza il valore del registro eax a zero, operazione questa operazione che è chiaro sintomo che il probabile malware potrebbe non essere in grado di sfruttare completamente le sue funzionalità in caso di mancata connessione internet.

Premesso ciò, il malware potrebbe utilizzare la connessione ad Internet per eseguire operazioni specifiche di cui tuttavia non abbiamo certezza, in quanto non specificate nella porzione di codice in nostro possesso.

In base a queste informazioni, possiamo soltanto ipotizzare che il malware sia stato progettato per sfruttare una connessione a Internet al fine di eseguire varie operazioni, come **l'invio di file e dati sensibili a server controllati dall'attaccante**, connessione a **domini infetti con conseguenti download di ulteriori malware**, o la **creazione di una backdoor** per consentire una comunicazione persistente tra la macchina vittima e l'attaccante in caso la vulnerabilità sfruttata da quest'ultimo venisse "patchata".

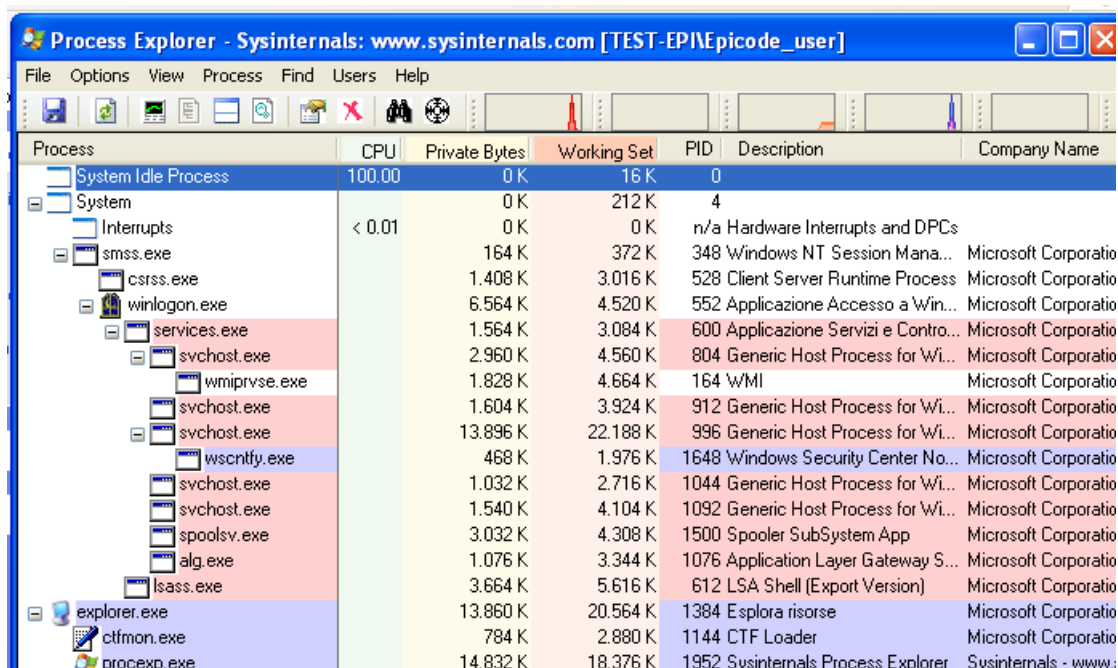
Dopo tali considerazioni, si potrebbe ipotizzare che il malware in cui è contenuta la porzione di codice in nostro possesso, potrebbe essere un **DOWNLOADER**, una **BACKDOOR** o un **TROJAN**.

TASK BONUS – CONVINCERE IL DIPENDENTE CHE IL FILE INDIVIDUATO NON È MALIGNO

Per dare conferma al dipendente che il file IEXPLORER sia innocuo, ho utilizzato tutti i tool a nostra disposizione in modo logico e strutturato per effettuare un buon processo di **ANALISI DINAMICA BASICA**, la quale presuppone l'esecuzione del malware in ambiente controllato, in modo tale da osservare il suo comportamento sul sistema infetto al fine di rimuovere l'infezione.

PASSAGGI SEGUITI:

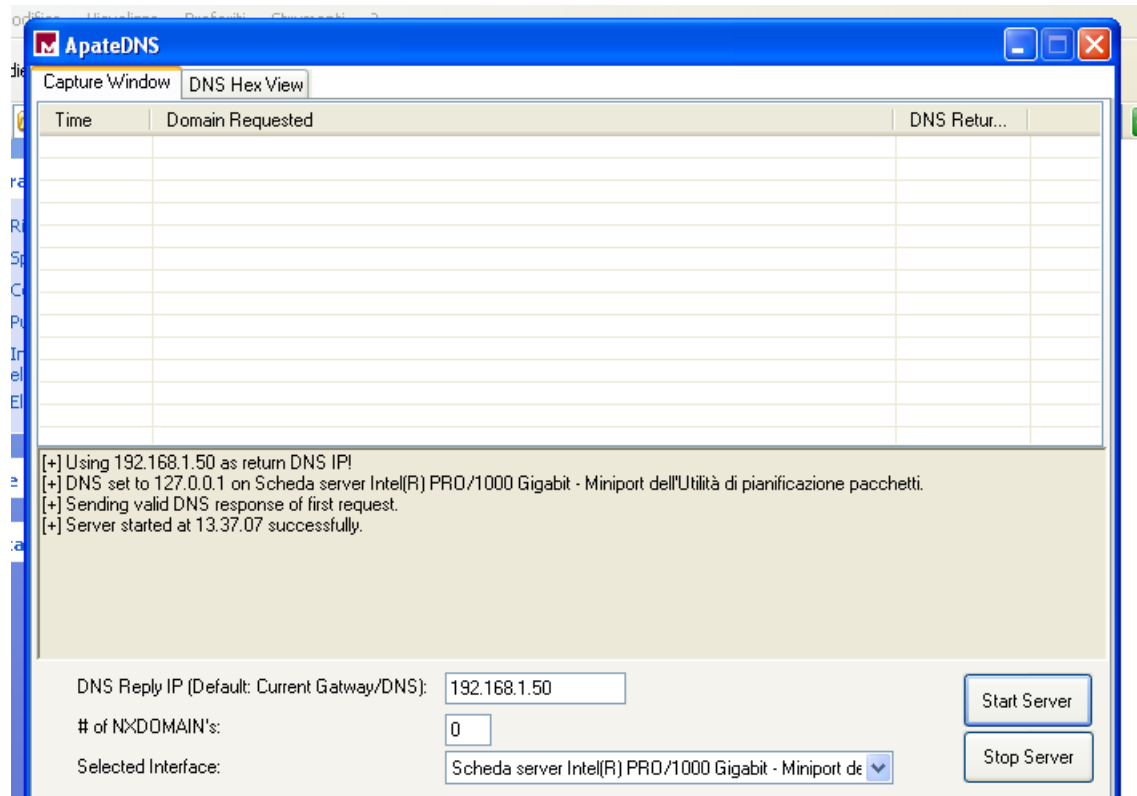
- Avviare Process Explorer
- Avviare il server DNS con ApateDNS
- Effettuare una prima istantanea con Regshot
- Avviare Process Monitor
- Avviare Wireshark
- Avviare il file eseguibile oggetto d'interesse
- Stappare le catture Wireshark e Process Monitor
- Salvare una seconda istantanea con Regshot per notare eventuali modifiche del file eseguibile alle chiavi di registro
- Fermate il server di ApateDNS
- Fermare Process Explorer
- ANALIZZARE I RISULTATI
- Ho avviato **PROCESS EXPLORER**, un tool che permette l'analisi dettagliata di tutti i processi in esecuzione su un sistema



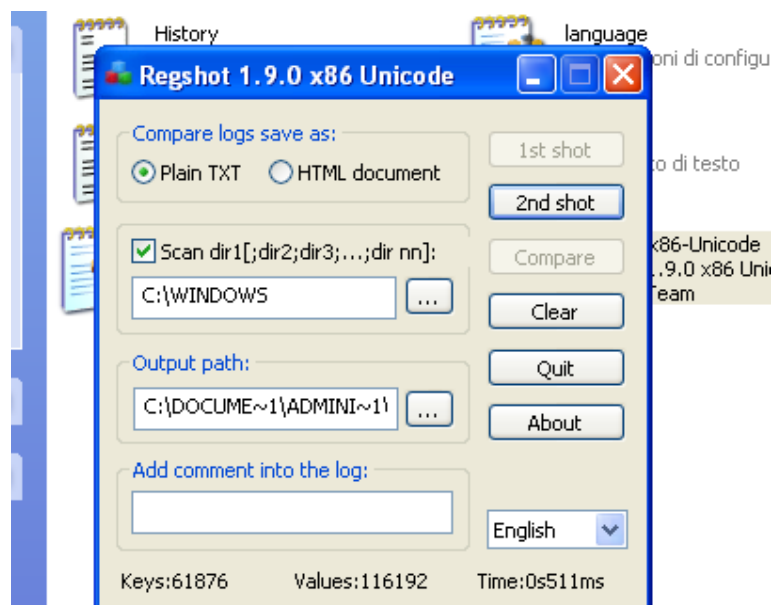
The screenshot shows the Process Explorer window from Sysinternals. The title bar reads 'Process Explorer - Sysinternals: www.sysinternals.com [TEST-EPIEpicode_user]'. The menu bar includes File, Options, View, Process, Find, Users, and Help. The toolbar contains various icons for file operations and system management. The main table lists running processes with columns for Process, CPU, Private Bytes, Working Set, PID, Description, and Company Name. The 'System' tree is expanded on the left, showing a hierarchy from System Idle Process down to explorer.exe, ctfmon.exe, and procexp.exe. The process list table shows details for these and other system processes like smss.exe, csrss.exe, winlogon.exe, services.exe, and various svchost.exe instances.

Process	CPU	Private Bytes	Working Set	PID	Description	Company Name
System Idle Process	100.00	0 K	16 K	0		
System		0 K	212 K	4		
Interrupts	< 0.01	0 K	0 K	n/a	Hardware Interrupts and DPCs	
smss.exe		164 K	372 K	348	Windows NT Session Mana...	Microsoft Corporatio
csrss.exe		1.408 K	3.016 K	528	Client Server Runtime Process	Microsoft Corporatio
winlogon.exe		6.564 K	4.520 K	552	Applicazione Accesso a Win...	Microsoft Corporatio
services.exe		1.564 K	3.084 K	600	Applicazione Servizi e Contro...	Microsoft Corporatio
svchost.exe		2.960 K	4.560 K	804	Generic Host Process for Wi...	Microsoft Corporatio
wmiprvse.exe		1.828 K	4.664 K	164	WMI	Microsoft Corporatio
svchost.exe		1.604 K	3.924 K	912	Generic Host Process for Wi...	Microsoft Corporatio
svchost.exe		13.896 K	22.188 K	996	Generic Host Process for Wi...	Microsoft Corporatio
wscntfy.exe		468 K	1.976 K	1648	Windows Security Center No...	Microsoft Corporatio
svchost.exe		1.032 K	2.716 K	1044	Generic Host Process for Wi...	Microsoft Corporatio
svchost.exe		1.540 K	4.104 K	1092	Generic Host Process for Wi...	Microsoft Corporatio
spoolsv.exe		3.032 K	4.308 K	1500	Spooler SubSystem App	Microsoft Corporatio
alg.exe		1.076 K	3.344 K	1076	Application Layer Gateway S...	Microsoft Corporatio
lsass.exe		3.664 K	5.616 K	612	LSA Shell (Export Version)	Microsoft Corporatio
explorer.exe		13.860 K	20.564 K	1384	Esplora risorse	Microsoft Corporatio
ctfmon.exe		784 K	2.880 K	1144	CTF Loader	Microsoft Corporatio
procexp.exe		14.832 K	18.376 K	1952	Sysinternals Process Explorer	Sysinternals - www.s...

- Ho aperto **ApateDNS**, per intercettare, tutte le richieste effettuate dall'eseguibile IEXPLORER.exe verso i domini Internet.



- Ho aperto **REGSHOT** e ho effettuato il **1st shot**, che scatterà la prima istantanea con lo stato delle chiavi di registro del sistema Windows



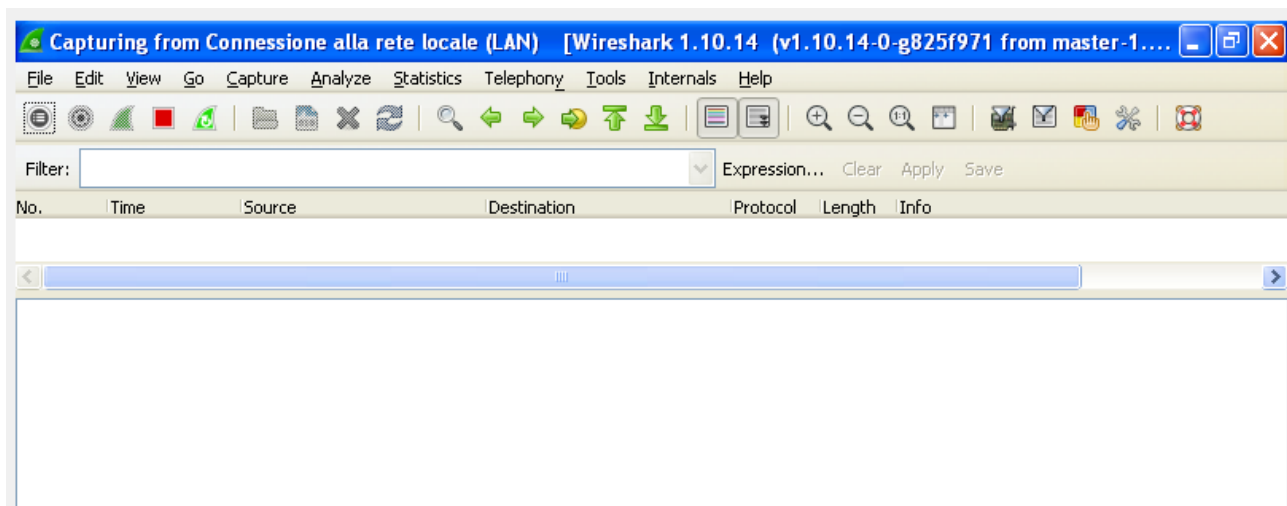
- Ho aperto **PROCESS MONITOR** per monitorare i processi ed i thread attivi, l'attività di rete, l'accesso ai file e le chiamate di sistema effettuate su un sistema operativo

Process Monitor - Sysinternals: www.sysinternals.com

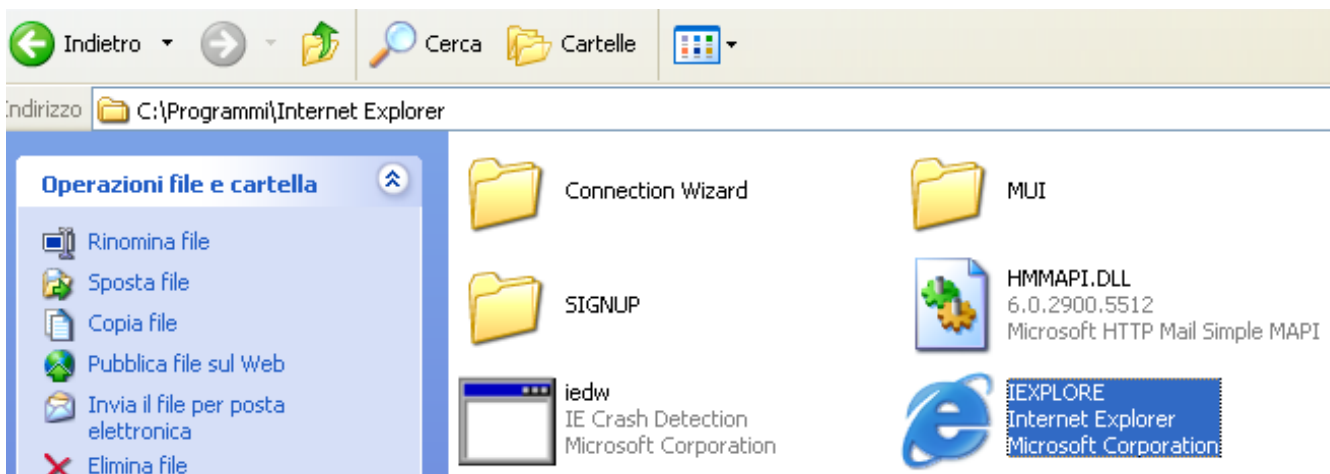
File Edit Event Filter Tools Options Help

Time...	Process Name	PID	Operation	Path	Result	Detail
13.38...	Explorer.EXE	1384	CloseFile	C:\Documents and Settings\Epicode_u...	SUCCESS	
13.38...	Explorer.EXE	1384	QueryOpen	C:\Documents and Settings\Epicode_u...	SUCCESS	CreationTime: 05/1...
13.38...	Explorer.EXE	1384	QueryOpen	C:\Documents and Settings\Epicode_u...	SUCCESS	CreationTime: 05/1...
13.38...	Explorer.EXE	1384	RegQueryKey	HKCR\exefile	SUCCESS	Query: Name
13.38...	Explorer.EXE	1384	RegOpenKey	HKCU\Software\Classes\exefile\Shelle...	NAME NOT FOUND	Desired Access: Q...
13.38...	Explorer.EXE	1384	RegOpenKey	HKCR\exefile\ShellEx\IconHandler	NAME NOT FOUND	Desired Access: Q...
13.38...	Explorer.EXE	1384	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: Name
13.38...	Explorer.EXE	1384	RegOpenKey	HKCU\Software\Classes\SystemFileAss...	NAME NOT FOUND	Desired Access: M...
13.38...	Explorer.EXE	1384	RegOpenKey	HKCR\SystemFileAssociations\exe	NAME NOT FOUND	Desired Access: M...
13.38...	Explorer.EXE	1384	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: Name
13.38...	Explorer.EXE	1384	RegOpenKey	HKCU\Software\Classes\SystemFileAss...	NAME NOT FOUND	Desired Access: M...
13.38...	Explorer.EXE	1384	RegOpenKey	HKCR\SystemFileAssociations\application	NAME NOT FOUND	Desired Access: M...

- Ho aperto **WIRESHARK** per catturare i pacchetti di rete e analizzare il traffico



- Ora si procede all'apertura del file **IEXPLORE.exe**



- **REGHSOT 2ND SHOT E COMPARE (dopo aver eseguito il file oggetto d'interesse), dal quale non si notano anomalie, in quanto le chiavi dei registri di sistema non vengono modificate, bensì vengono aggiunte o rimosse chiavi, ma da altri programmi in esecuzione come Process Monitor.**

```

~res-x86 - Blocco note
File Modifica Formato Visualizza ?
Computer: TEST-EPI , TEST-EPI
Username: Epicode_user , Epicode_user

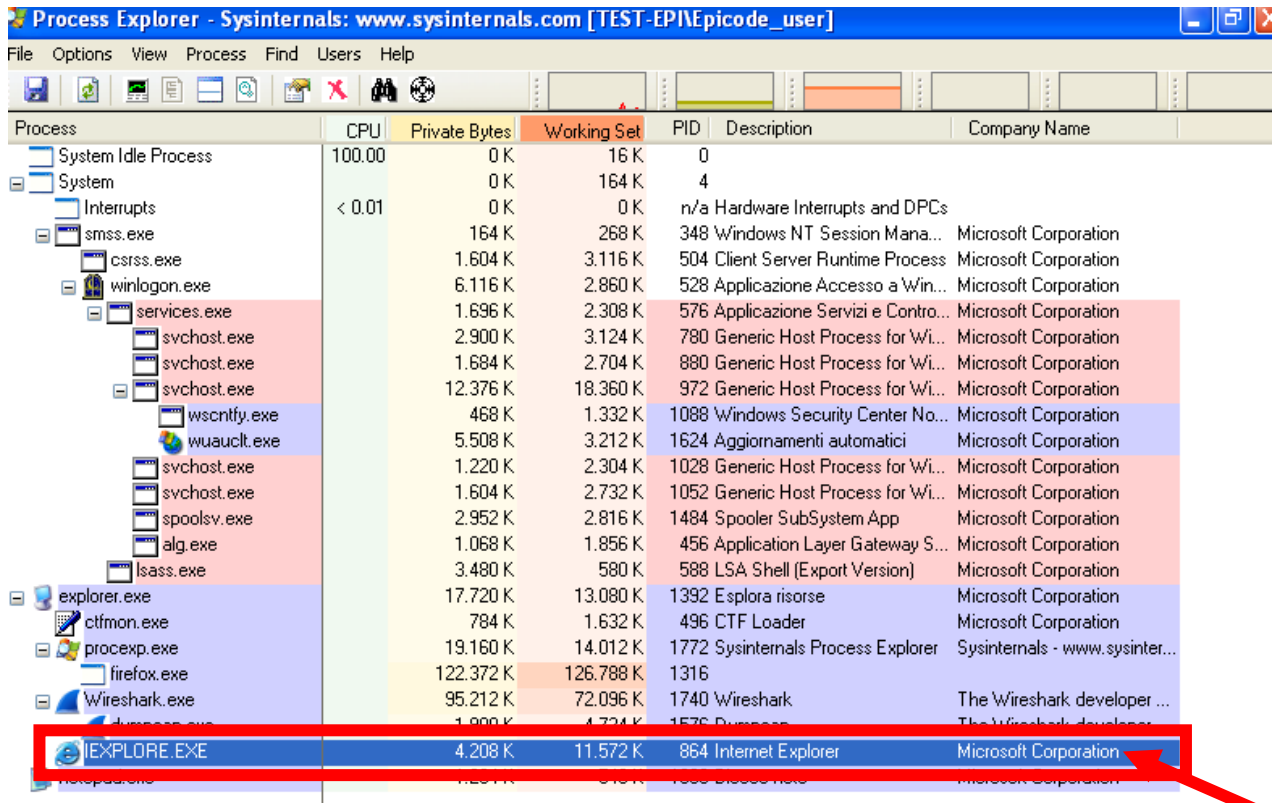
-----
Keys deleted: 2
-----
HKLM\SYSTEM\ControlSet001\Services\PROCMON24\Enum
HKLM\SYSTEM\CurrentControlSet\Services\PROCMON24\Enum
-----
Keys added: 5
-----
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_PROCMON24\0000\Control
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_PROCMON24\0000\Control
HKU\S-1-5-21-583907252-1060284298-854245398-1003\Software\Microsoft\windows\CurrentVersion\Explor
HKU\S-1-5-21-583907252-1060284298-854245398-1003\Software\Microsoft\windows\CurrentVersion\Explor
HKU\S-1-5-21-583907252-1060284298-854245398-1003\Software\Microsoft\windows\CurrentVersion\Explor
-----
Values deleted: 6
-----
HKLM\SYSTEM\ControlSet001\Services\PROCMON24\Enum\0: "Root\LEGACY_PROCMON24\0000"
HKLM\SYSTEM\ControlSet001\Services\PROCMON24\Enum\Count: 0x00000001
HKLM\SYSTEM\ControlSet001\Services\PROCMON24\Enum\NextInstance: 0x00000001
HKLM\SYSTEM\CurrentControlSet\Services\PROCMON24\Enum\0: "Root\LEGACY_PROCMON24\0000"
HKLM\SYSTEM\CurrentControlSet\Services\PROCMON24\Enum\Count: 0x00000001
HKLM\SYSTEM\CurrentControlSet\Services\PROCMON24\Enum\NextInstance: 0x00000001
-----
Values added: 8
-----
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_PROCMON24\0000\Control\ActiveService: "PROCMON24"
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_PROCMON24\0000\Control\ActiveService: "PROCMON24"
HKU\S-1-5-21-583907252-1060284298-854245398-1003\Software\Microsoft\windows\CurrentVersion\Explor
HKU\S-1-5-21-583907252-1060284298-854245398-1003\Software\Microsoft\windows\CurrentVersion\Explor
HKU\S-1-5-21-583907252-1060284298-854245398-1003\Software\Microsoft\windows\CurrentVersion\Explor
HKU\S-1-5-21-583907252-1060284298-854245398-1003\Software\Microsoft\windows\CurrentVersion\Explor
HKU\S-1-5-21-583907252-1060284298-854245398-1003\Software\Microsoft\windows\CurrentVersion\Explor

```

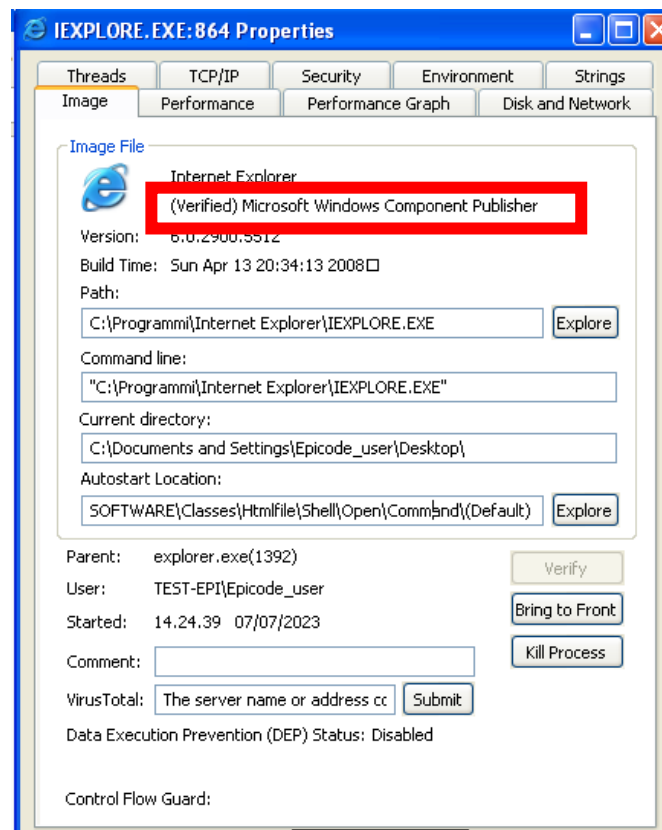
- **Su ApateDNS, dopo l'apertura del file IEXPLORER.EXE, non ho riscontrato alcuna anomalia, in quanto, le richieste effettuate riguardavano Wireshark e www.microsoft.com, ovvero la pagina iniziale aperta all'apertura dell'eseguibile IEXPLORER.EXE**

ApateDNS			
Capture Window		DNS Hex View	
Time	Domain Requested	DNS Return	
13.53.57	www.wireshark.org	FOUND	
13.54.21	255.1.168.192.in-addr.arpa	FOUND	
13.54.38	www.microsoft.com	FOUND	
(+) Using 192.168.1.50 as return DNS IP! (+) DNS set to 127.0.0.1 on Scheda server Intel(R) PRO/1000 Gigabit - Miniport dell'Utilità di pianificazione pacchetti. (+) Sending valid DNS response of first request. (+) Server started at 13.52.31 successfully.			
DNS Reply IP (Default: Current Gateway/DNS):		192.168.1.50	Start Server
# of NXDOMAIN's:		0	Stop Server
Selected Interface:		Scheda server Intel(R) PRO/1000 Gigabit - Miniport de	

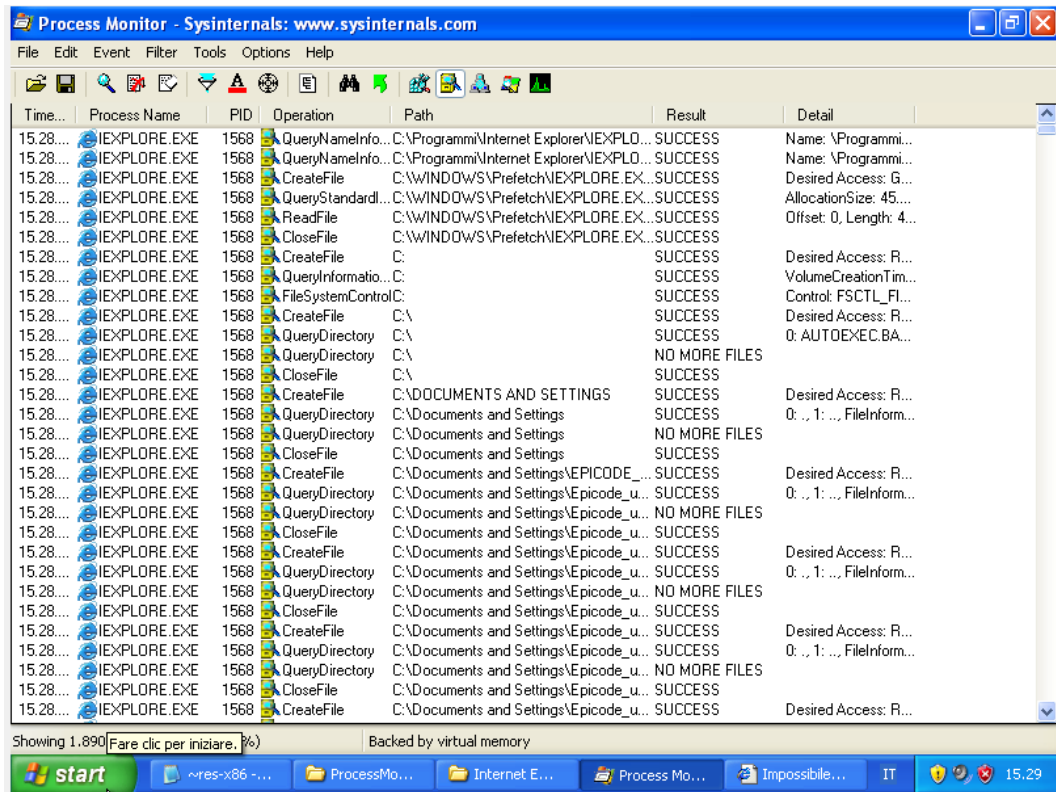
- **PROCESS EXPLORER**, post esecuzione del file, ha individuato un nuovo processo, per l'appunto **IEXPLORE.EXE**, il quale è regolarmente associata al **COMPANY NAME "MICROSOFT CORPORATION"**.



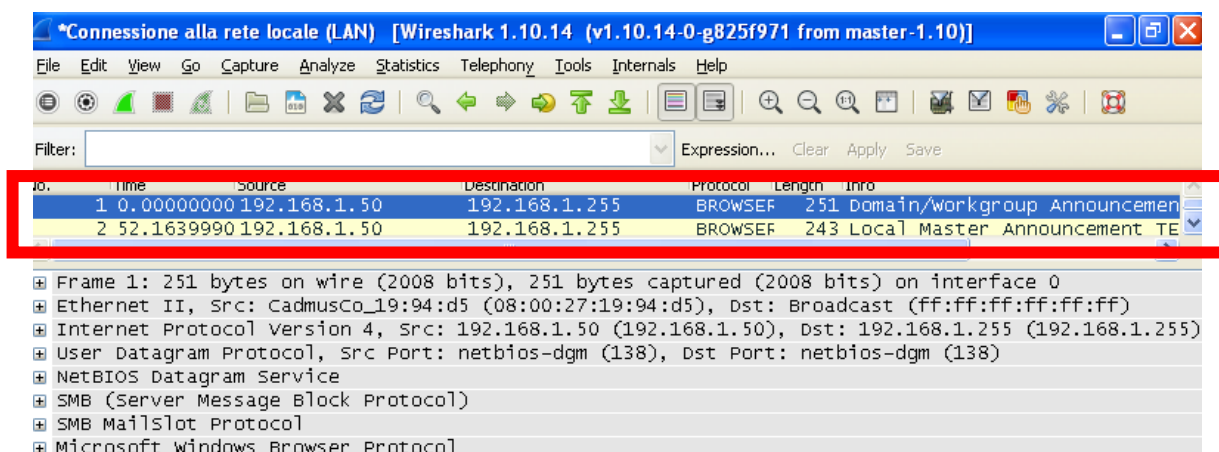
- Su **PROCESS EXPLORER** ho aperto il nuovo processo creato, e **non ho rilevato nulla di anomalo**, anzi, il programma è **"(Verified) Microsoft Windows Component Publisher"**.



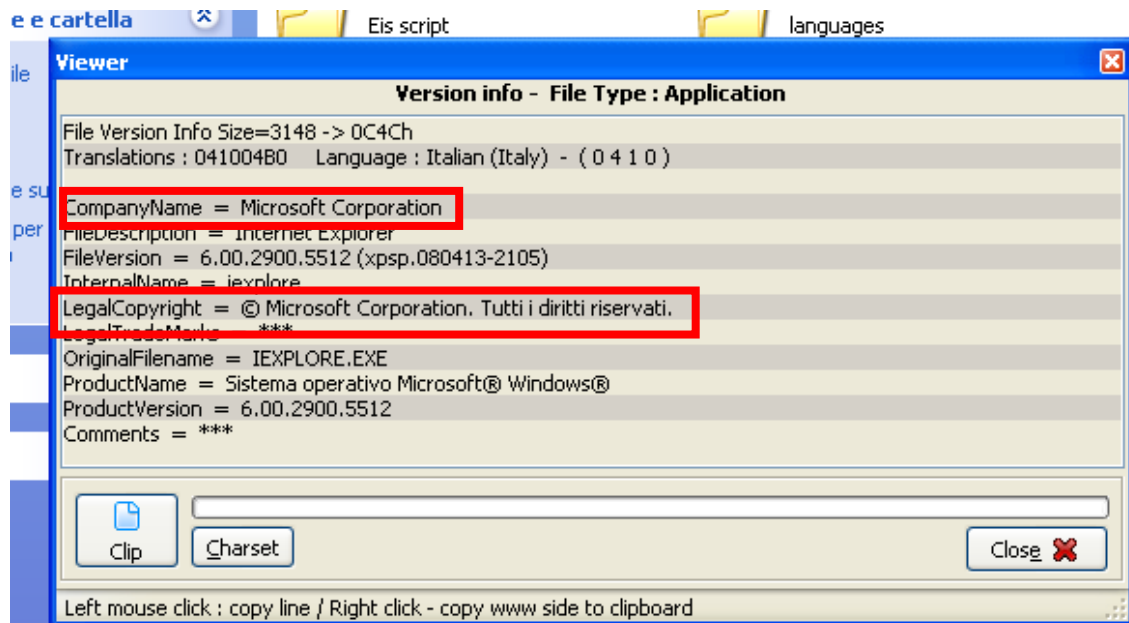
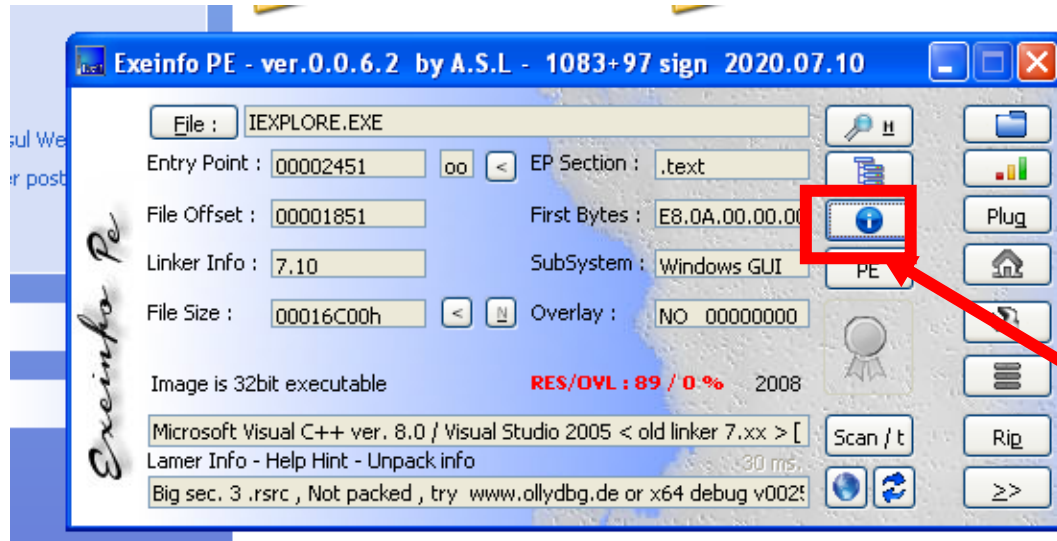
- Sono ritornato su **PROCESS MONITOR** e dopo aver impostato il filtro **PROCESS NAME** IEXPLORE.EXE, ho visto soltanto i processi interessati dal file eseguibile oggetto d'interesse. Anche in questo caso, dopo aver analizzati i risultati, non sono stati rilevati eventi anomali.



- All'apertura dell'eseguibile IEXPLORER.exe, **WIRESHARK** non ha catturato pacchetti insoliti, motivo per cui non sono state individuate anomalie.



- Infine, ho utilizzato **EXEINFO PE**, dove, grazie alla sezione VERSION INFO (in rosso), ho recuperato le informazioni sul file IEXPLORE.EXE, che, come già visto in precedenza, è **regolarmente distribuito da Microsoft Corporation**.



- Per dare un'ulteriore e ultima prova al dipendente che il file da lui stesso individuato non ha natura maligna, ho estratto l'hash del file eseguibile IEXPLORE.exe con l'utility **md5deep**

```

C:\> Prompt dei comandi
Microsoft Windows XP [Versione 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Epicode_user>cd Desktop
C:\Documents and Settings\Epicode_user\Desktop>cd md5deep-4.3
C:\Documents and Settings\Epicode_user\Desktop\md5deep-4.3>md5deep.exe "C:\Programmi\Internet Explorer\IEXPLORE.exe"
173e49aebb665c0577d751ba55f84b6c C:\Programmi\Internet Explorer\IEXPLORE.exe
C:\Documents and Settings\Epicode_user\Desktop\md5deep-4.3>

```

- Su un altro PC, ho inserito l'hash ricavato su **Virus Total**, il quale mi ha indicato che il file è **stato regolarmente distribuito da Microsoft**.

VIRUSTOTAL

Analyse suspicious files, domains, IPs and URLs to detect malware and other breaches, automatically share them with the security community.

Search for a hash, domain, IP address, URL or gain additional context and threat landscape visibility with [VT ENTERPRISE](#).

173e49aebb665c0577d751ba55f84b6c

f6ef7f7eef4f15a9b3bba4295e0bbe3acb5886a9e4d2b5aa72ca25e8f396b9cd

0 / 64

File distributed by Microsoft

f6ef7f7eef4f15a9b3bba4295e0bbe3acb5886a9e4d2b5aa72ca25e8f396b9cd
IEXPLORE.EXE

peexe trusted known-distributor

Community Score

DETECTION DETAILS BEHAVIOR COMMUNITY 2

[Join the VT Community](#) and enjoy additional community insights and crowdsourced detections, plus an API

Security vendors' analysis

Ad-Aware	✓ Undetected
AhnLab-V3	✓ Undetected

- Ho effettuato un doppio check su **Hybrid Analysis** utilizzando l'hash del file IEXPLORER.exe ricavato con md5deep. Anche in questo caso non è stata rilevata nessuna minaccia.



173e49aebb665c0577d751ba55f84b6c [Search](#)

Anti-Virus Results ✓ Up-to-date

CrowdStrike Falcon

CLEAN

Static Analysis and ML ⓘ

Last Update: 07/07/2023 12:48:58 (UTC)

View Details: [N/A](#)

Visit Vendor: [🔗](#)

[GET STARTED WITH A FREE TRIAL](#)

MetaDefender

CLEAN

Multi Scan Analysis

Last Update: 07/07/2023 12:48:58 (UTC)

View Details: [🔗](#)

Visit Vendor: [🔗](#)

VirusTotal

CLEAN

Multi Scan Analysis

Last Update: 07/07/2023 12:48:58 (UTC)

View Details: [🔗](#)

Visit Vendor: [🔗](#)

CONCLUSIONE TASK BONUS

Dopo un'analisi dinamica basica, è stato confermato che il file eseguibile IEXPLORE.EXE è stato distribuito correttamente da Microsoft Corporation, come indicato sia da Process Explorer che da Exeinfope (questo tool utilizzato in fase di analisi statica basica). Utilizzando Wireshark e Apatedns, ho osservato che il file eseguibile non si connette a domini infetti o siti di terze parti, ma raggiunge solo il sito della sua pagina iniziale, www.microsoft.com.

Inoltre, il file eseguibile non crea altri processi oltre al proprio. Infine, verifiche su siti come VirusTotal e Hybrid Analysis non hanno rilevato alcuna indicazione che il programma sia dannoso.