

## ANALISI Malware\_U3\_W2\_L5

Quando si analizza un malware, la prima cosa da fare è capire se è un malware. Ogni file ha una firma, quindi per controllare se stiamo analizzando un malware, posso controllare se nei database la firma del malware è nota.

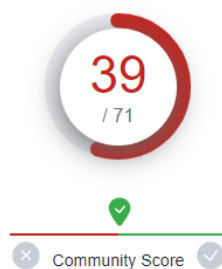
Prima di fare ciò utilizzo **CFF Explorer**, software che permette di esaminare la struttura interna del file, modificare sezioni, risorse e importazioni/esportazioni. Ci restituisce diverse info tra cui l'**hash** del file, che posso riportare su **VirusTotal**, per la verifica del malware.

Questa schermata, oltre all'hash, ci restituisce altre info, come la data di creazione dell'eseguibile, grandezza del file e in quale directory si trova. In questo caso, mi serve l'hash da riportare su VirusTotal.

Property	Value
File Name	C:\Documents and Settings\Administrator\Desktop\Malanalisy\Esercizio_Pr...
File Type	Portable Executable 32
File Info	No match found.
File Size	40.00 KB (40960 bytes)
PE Size	40.00 KB (40960 bytes)
Created	Wednesday 02 February 2011, 16.29.06
Modified	Monday 03 July 2023, 14.17.26
Accessed	Friday 07 July 2023, 10.56.39
MD5	C0B54534E188E1392F28D17FAFF3D454
SHA-1	BB6F01B1FEF74A9CFC83EC2303D14F492A671F3C



Procedo con la verifica dell'hash su VirusTotal, quest'ultimo mi riporta 39 vendors su 71, lo etichettano come file maligno e l'etichetta di minaccia popolare riporta che è un trojan.



39 security vendors and no sandboxes flagged this file as malicious

b71777edbf21167c96d20ff803cbcb25d24b94b3652db2f286dcd6efd3d8416a

Lab01-02.exe.exe

peexe checks-network-adapters runtime-modules armadillo direct-cpu-clock-access

Community Score

DETECTION

DETAILS

RELATIONS

BEHAVIOR

COMMUNITY 6

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label trojan.r002c0pdm21

Threat categories trojan



Nei dettagli VirusTotal mi riporta anche le **librerie** importate e le **Sections**, che analizzerò a breve con l'utilizzo di CFF Explorer.

#### Sections

Name	Virtual Address	Virtual Size	Raw Size	Entropy	MD5	Chi2
.text	4096	19064	20480	6.37	4b8aaeb128744c00b1f9b29dd120616e	196535.5
.rdata	24576	2398	4096	3.66	e5e39acc53e64c50fa5a35693a911478	304856
.data	28672	16136	12288	0.7	305514f6ece00473b7ff8bc023f57e15	2765274

#### Imports

+ KERNEL32.dll  
+ WININET.dll

Avvio CFF Explorer, inserisco il file eseguibile da analizzare e mi sposto sulle librerie importate. Trovo due librerie **KERNEL32.dll** e **WININET.dll**, librerie importate dinamicamente, cioè vengono caricate dal sistema operativo quando l'eseguibile è avviato. Al loro interno hanno delle funzioni importate, 44 per la libreria Kernel32 e 5 per la libreria WININET.

Malware_U3_W2_L5.exe						
Module Name	Imports	OFTs	TimeStamp	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

**KERNEL32.DLL** è una libreria di sistema essenziale in ambienti Windows che fornisce funzioni di basso livello per interagire con il sistema operativo. Contiene un'ampia gamma di funzioni che coprono diverse aree, inclusa la manipolazione dei file, la gestione di memoria, l'accesso al registro di sistema, la sincronizzazione dei processi, la gestione degli errori, ecc. In questo caso vengono importate 44 funzioni, quelle che più risaltano all'occhio e che potrebbero essere più pericolose sono:

0000669E	0000669E	029E	TerminateProcess	0000669E	0000669E	029E	TerminateProcess
0000669E	0000669E	00F7	GetCurrentProcess	000066B2	000066B2	00F7	GetCurrentProcess
0000688E	0000688E	013E	GetProcAddress				
000068A0	000068A0	01C2	LoadLibraryA				

**TerminateProcess**: questa funzione consente di terminare in modo forzato un processo in esecuzione;

**GetCommandLineA**: restituisce la riga di comando utilizzata per avviare l'applicazione consentendo di ottenere informazioni sulle opzioni di avvio e gli argomenti passati al programma;

**GetCurrentProcess**: consente di ottenere informazioni sul processo stesso o di eseguire operazioni specifiche su di esso, come modifica delle impostazioni;

**LoadLibrary:** consente di caricare in runtime una libreria nel processo corrente, incrementando funzionalità esterne o aggiuntive;

**GetProcAddress:** permette di ottenere l'indirizzo di una funzione specifica all'interno di una libreria caricata, anche questa viene caricata in runtime.

**Winnet.DLL:** fornisce funzioni e servizi di rete per il sistema operativo Windows, come la gestione della connessione di rete, l'accesso ai protocolli di rete e la gestione dei socket. In questo caso vengono caricate 5 funzioni:

00006640	00006640	0071	InternetOpenUrlA
0000662A	0000662A	0056	InternetCloseHandle
00006616	00006616	0077	InternetReadFile
000065FA	000065FA	0066	InternetGetConnectedState
00006654	00006654	006F	InternetOpenA

**InternetOpenUrlA:** è una funzione che permette di aprire una connessione verso una risorsa specificata da un URL;

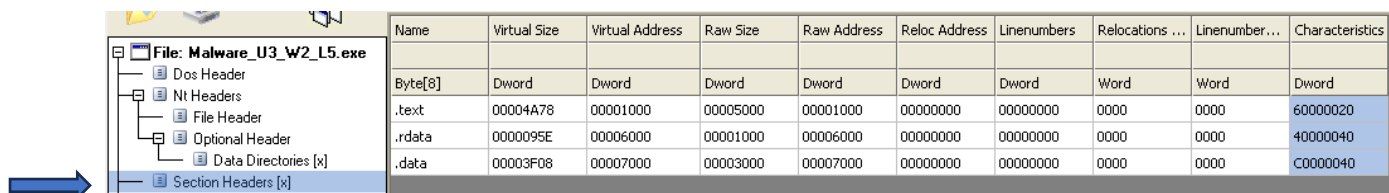
**InternetCloseHandle:** viene utilizzata per chiudere l'handle di una connessione Internet precedentemente aperta;

**InternetReadFile:** viene utilizzata per leggere i dati di una connessione Internet aperte;

**InternetGetConnectedState:** è una funzione che determina se il sistema operativo è connesso ad Internet;

**InternetOpenA:** funzione che crea un handle per l'accesso a Internet.

Successivamente mi sposto in **Section Headers**, per vedere le sezioni dell'eseguibile. Ci sono 3 sezioni: .text, .rdata, .data.



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

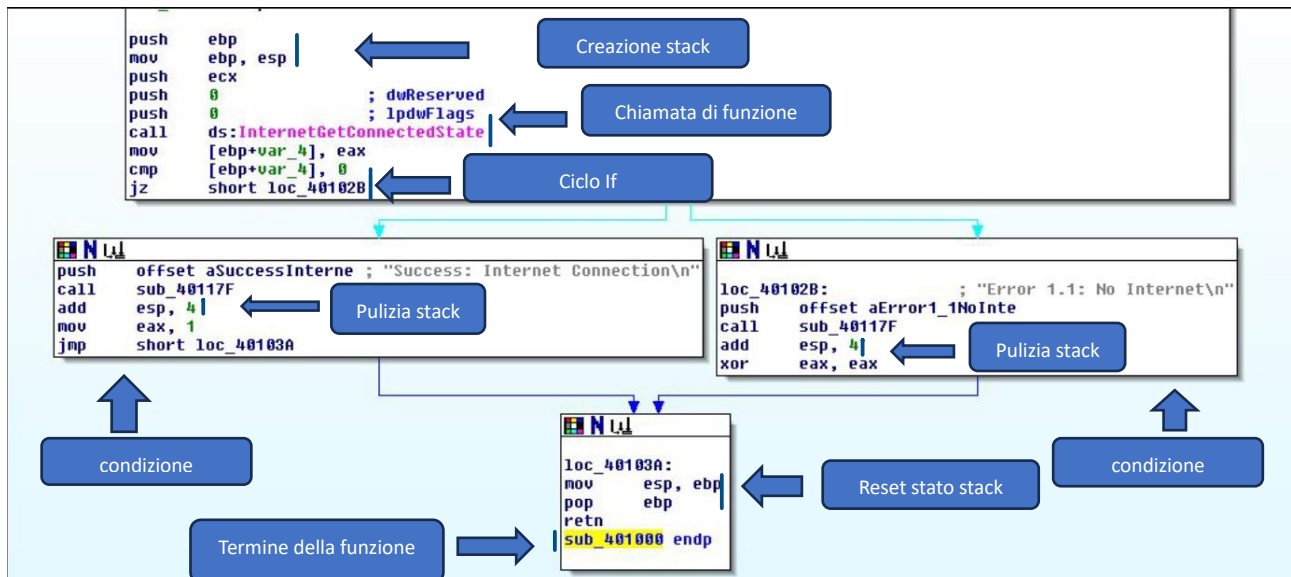
**.text:** contiene le istruzioni (righe di codice) che la CPU eseguirà una volta che il software sarà avviato.

**.rdata:** include le informazioni circa le librerie e le funzioni importate ed esportate dall'eseguibile, informazioni che abbiamo appena visto.

**.data:** contiene tipicamente i dati /le variabili globali del programma eseguibile, che devono essere disponibili da qualsiasi parte del programma.

Con riferimento al codice Assembly che segue, rispondo ai seguenti quesiti:

- 1) Identificare i costrutti noti (creazione dello stack, eventuali cicli, altri costrutti)
- 2) Ipotizzare il comportamento della funzionalità implementata
- 3) (BONUS) spiegare qualche istruzione assembly complessa



(1) L'identificazione dei costrutti noti è stata implementata nello screenshot. Maggiori dettagli, come ad esempio le chiamate alla subroutine o i vari jump, sono stati dettagliati nel punto 2-3.

(2-3) Inizialmente, viene inizializzato lo **stack** e viene salvato il valore di **EBP** nello stack. Successivamente, i parametri vengono pushati nello stack per essere passati alla funzione **InternetGetConnectedState**, che viene chiamata con **call**. Il risultato della chiamata viene quindi memorizzato nella variabile **[ebp+var\_4]**.

Successivamente, viene eseguito un confronto (**cmp**) tra il valore della variabile **[ebp+var\_4]** e **0**. Quando il confronto restituisce **0**, il salto viene eseguito. In questo caso, si salta all'etichetta **loc\_40102B**, che chiama la subroutine **sub\_40117F** per stampare "Error 1.1: No Internet". Dopodiché, vengono eseguite ulteriori istruzioni per pulire lo stack e impostare **eax** a **0**.

Se il salto non viene eseguito (quindi se il risultato è diverso da **0**), viene pushata una stringa nello stack e viene chiamata la subroutine **sub\_40105F** per eseguire l'output del messaggio di successo. Successivamente, viene eseguito **add esp, 4** per liberare lo spazio dallo stack e **mov eax, 1** per impostare il registro **eax** a **1**.

Infine, viene eseguito un salto (**jmp**) all'indirizzo **loc\_40103A**, che resetta lo stato dello stack ed esegue la funzione di return (**retn**), che termina la funzione **sub\_401000** con **endp**.

In sintesi, il codice verifica lo stato di connessione a Internet utilizzando la funzione **InternetGetConnectedState**. Dopo l'inizializzazione dello stack e la chiamata alla funzione, viene eseguito un confronto per determinare se la connessione è attiva o meno. In base al risultato, vengono eseguite istruzioni diverse. Se la connessione è attiva, viene visualizzato un messaggio di successo; altrimenti, viene visualizzato un messaggio di errore. Alla fine, il programma termina.

## Bonus

Un giovane dipendente neo assunto segnala al reparto tecnico la presenza di un programma sospetto IEXPLORE.EXE, il suo superiore gli dice di stare tranquillo ma lui non è soddisfatto e chiede supporto al SOC.

Come membro senior del SOC mi è richiesto di convincere il dipendente che il file non è maligno.

### ANALISI STATICA BASE

Inizio l'analisi statica base sull'eseguibile **IEXPLORE.EXE**, utilizzando **CFF Explorer**, così da esaminare la struttura interna del file e ricavarne l'hash per poter verificare la firma dell'eseguibile su **VirusTotal**.

Dalla schermata principale di CFF Explorer, dopo aver inserito l'eseguibile, mi viene mostrata già un primo feedback molto importante. Oltre alla grandezza del file, agli hash e alla data di creazione, mi viene mostrata la compagnia proprietaria del file, la descrizione, la versione, il copyright e il nome del prodotto. Questa è già una prova chiara che il file è sicuro, in quanto la compagnia produttrice del file è Microsoft Corporation e c'è un copyright sull'eseguibile.

IEXPLORE.EXE	
Property	Value
File Name	C:\Program Files\Internet Explorer\IEXPLORE.EXE
File Type	Portable Executable 32
File Info	No match found.
File Size	623.84 KB (638816 bytes)
PE Size	618.00 KB (632832 bytes)
Created	Wednesday 14 June 2023, 15.26.41
Modified	Wednesday 29 January 2020, 05.40.36
Accessed	Friday 07 July 2023, 14.15.42
MD5	B60DDDD2D63CE41CB8C487FCFB6419E
SHA-1	EADCE51C88C8261852C1903399DDE742FBA2061B

Property	Value
CompanyName	Microsoft Corporation
FileDescription	Internet Explorer
FileVersion	8.00.6001.18702 (longhorn_ie8_rtm(wmbla).090308-0339)
InternalName	ieexplore
LegalCopyright	© Microsoft Corporation. All rights reserved.
OriginalFilename	IEXPLORE.EXE
ProductName	Windows® Internet Explorer

L'hash ricavato poco fa, lo inserisco su **VirusTotal**, così da verificare la firma, in quanto quest'ultimo offre un servizio di aggregazione dei risultati di molti motori di scansione antivirus.

VirusTotal mi riporta il file non maligno, infatti il risultato è **0/70**. Anche qui, mi escono le informazioni ricavate da CFF explorer, come produttore, versione e copyright.

0

/ 70

Community Score

No security vendors and no sandboxes flagged this file as malicious

Reanalyze Similar More

b18a0d4beba606bf30f5010ba3c72abafac80d5f303a8bffb24d7f7b78b786e6

Size623.84 KB

Last Analysis Date15 days ago

EXE

peexe

via-tor

overlay

runtime-modules

signed

detect-debug-environment

idle

direct-cpu-clock-access

checks-user-input

File Version Information

Copyright© Microsoft Corporation. All rights reserved.

ProductWindows® Internet Explorer

DescriptionInternet Explorer

Original NameIEXPLORE.EXE

Internal Nameiexplore

File Version8.00.6001.18702 (longhorn\_ie8\_rtm(wmbla).090308-0339)

Date signed2009-03-08 20:09:00 UTC

Inoltre mi riporta le librerie e le sezioni che servono a questo eseguibile.

Sections

Name	Virtual Address	Virtual Size	Raw Size	Entropy	MD5	Chi2
.text	4096	40549	40960	5.82	6f4b4058b7d8a274f013151130b1fe63	847691.12
.data	45056	1632	2048	0.22	419d60dc6b239eaff0e14f76eacce1a5	502476
.rsrc	49152	585240	585728	6.78	57661f31be42fec90a9312f710873968	7560525
.reloc	634880	2820	3072	6.23	a92eabd61b4ac11fdf667c8da4e34bf1	19489.07

Imports

+ urlmon.dll

+ iertutil.dll

+ ADVAPI32.dll

+ KERNEL32.dll

+ msvcrt.dll

+ SHELL32.dll

+ ntdll.dll

+ ole32.dll

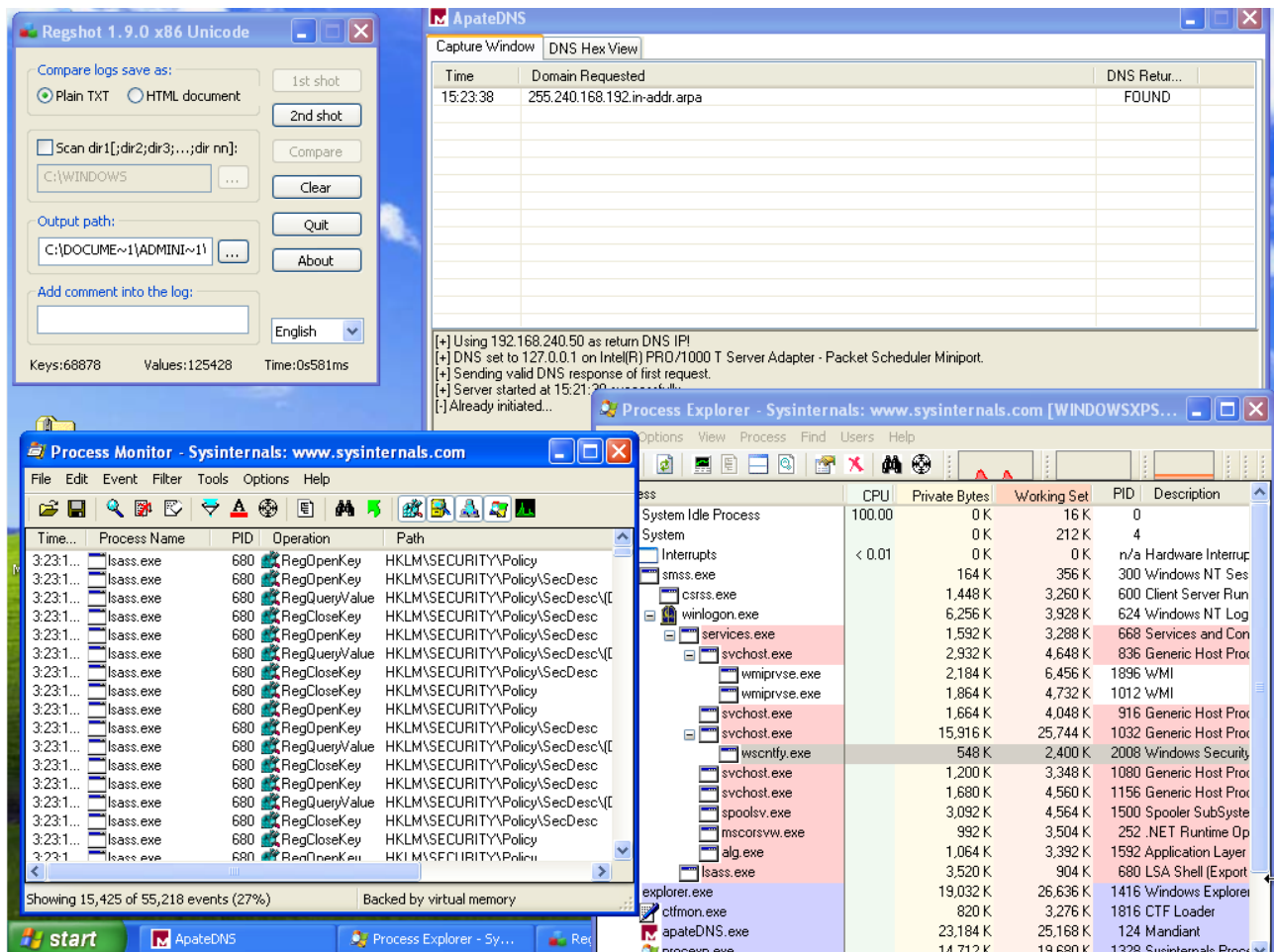
+ SHLWAPI.dll

+ USER32.dll

## ANALISI DINAMICA BASICA

Nonostante l'analisi statica basica non mi ha riportato l'eseguibile maligno, procedo ugualmente con l'analisi dinamica basica, per esser maggiormente sicuri. Dopo aver configurato l'ambiente sicuro, disattivando dispositivi USB e cartelle condivise e abilitando l'interfaccia di rete interna.

Avvio **Process Explorer**, **ApateDNS** e salvo una prima istantanea delle chiavi di registro con **Regshot** e avvio **Procmon**.



Posso ora avviare l'eseguibile



Dopo aver lasciato qualche minuto all'eseguibile di compiere le sue istruzioni, procedo con lo stoppare procmon, salvo la seconda istantanea con Regshot, fermo il server ApateDNS e fermo Process Explorer.

Ora posso iniziare l'analisi dei vari tool utilizzati.

## ANALISI DINAMICA CON REGSHOT

Inizio col comparare le due istantanee di **RegShot** e noto che l'aggiunta delle 39 chiavi ed i valori aggiunti, riguardano i programmi in esecuzione come procmon, o l'explorer avviato. Non risulta la creazione di chiavi sospette, ne l'aggiunta di valori strani, peraltro non risultano chiavi o valori

```
Computer: WINDOWSXPSP3 , WINDOWSXPSP3
Username: Administrator , Administrator
```

```
-----
Keys added: 39
-----
```

```
HKLM\SYSTEM\ControlSet001\Control\Class\{3A1380F4-708F-49DE-B2EF-04D25EB009D5}
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_PROCMON24
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_PROCMON24\0000
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_PROCMON24\0000\Control
HKLM\SYSTEM\ControlSet001\Services\PROCMON24
HKLM\SYSTEM\ControlSet001\Services\PROCMON24\Instances
HKLM\SYSTEM\ControlSet001\Services\PROCMON24\Instances\Process Monitor 24 Instance
HKLM\SYSTEM\CurrentControlSet\Control\Class\{3A1380F4-708F-49DE-B2EF-04D25EB009D5}
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_PROCMON24
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_PROCMON24\0000
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_PROCMON24\0000\Control
HKLM\SYSTEM\CurrentControlSet\Services\PROCMON24
HKLM\SYSTEM\CurrentControlSet\Services\PROCMON24\Instances
HKLM\SYSTEM\CurrentControlSet\Services\PROCMON24\Instances\Process Monitor 24 Instance
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\{18713b81-0-
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\{18713b81-0-
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\{18713b81-0-
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\ShellNoRoam\BagMRU\0\1\2
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\ShellNoRoam\BagMRU\0\1\2\0
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\ShellNoRoam\BagMRU\6\5
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\ShellNoRoam\Bags\41
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\ShellNoRoam\Bags\41\Shell
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\ShellNoRoam\Bags\42
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\ShellNoRoam\Bags\42\Shell
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\ShellNoRoam\Bags\43
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Microsoft\Windows\ShellNoRoam\Bags\43\Shell
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\SysInternals\Process Monitor
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Classes\PML
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Classes\ProcMon.LogFile.1
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Classes\ProcMon.LogFile.1\DefaultIcon
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Classes\ProcMon.LogFile.1\Shell
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Classes\ProcMon.LogFile.1\Shell\open
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Software\Classes\ProcMon.LogFile.1\Shell\open\command
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Classes\PML
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Classes\ProcMon.LogFile.1
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Classes\ProcMon.LogFile.1\DefaultIcon
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Classes\ProcMon.LogFile.1\Shell
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Classes\ProcMon.LogFile.1\Shell\open
HKU\S-1-5-21-1004336348-842925246-1343024091-500\Classes\ProcMon.LogFile.1\Shell\open\command
```

```
-----
Values added: 154
-----
```

```
HKLM\SYSTEM\ControlSet001\Control\Class\{3A1380F4-708F-49DE-B2EF-04D25EB009D5}\Class: "PROCMON24"
HKLM\SYSTEM\ControlSet001\Control\Class\{3A1380F4-708F-49DE-B2EF-04D25EB009D5}\NoDisplayClass: "1"
```

cancellati.

## ANALISI DINAMICA CON APATEDNS

**ApateDNS** è un software che consente di creare un ambiente di test sicuro simulando la risoluzione dei nomi di dominio (DNS) falsi, fornendo così una protezione aggiuntiva contro attacchi di tipo phishing o malware.

Analizzando il tool, in questo caso i domini contattati dall'eseguibili sono domini che non destano sospetto, in quanto, l'eseguibile **IEXPLORE.EXE** prova a collegarsi con domini sicuri quali:

**go.microsoft.com** e [www.live.com](http://www.live.com).

15:23:38	255.240.168.192:in-addr.arpa	FOUND	
15:27:37	go.microsoft.com	FOUND	
15:27:38	www.live.com	FOUND	
15:27:38	www.live.com	FOUND	

```
[+] Using 192.168.240.50 as return DNS IP!
[+] DNS set to 127.0.0.1 on Intel(R) PRO/1000 T Server Adapter - Packet Scheduler Miniport.
[+] Sending valid DNS response of first request.
[+] Server started at 15:21:38 successfully.
[-] Already initiated...
[+] Stopping Server...
[+] Blank DNS detected, setting back DNS blank[+] DNS Restored.
[+] Interfaces list has been refreshed.
```

DNS Reply IP (Default: Current Gateway/DNS):

# of N:DOMAIN's:

Selected Interface:

Start Server

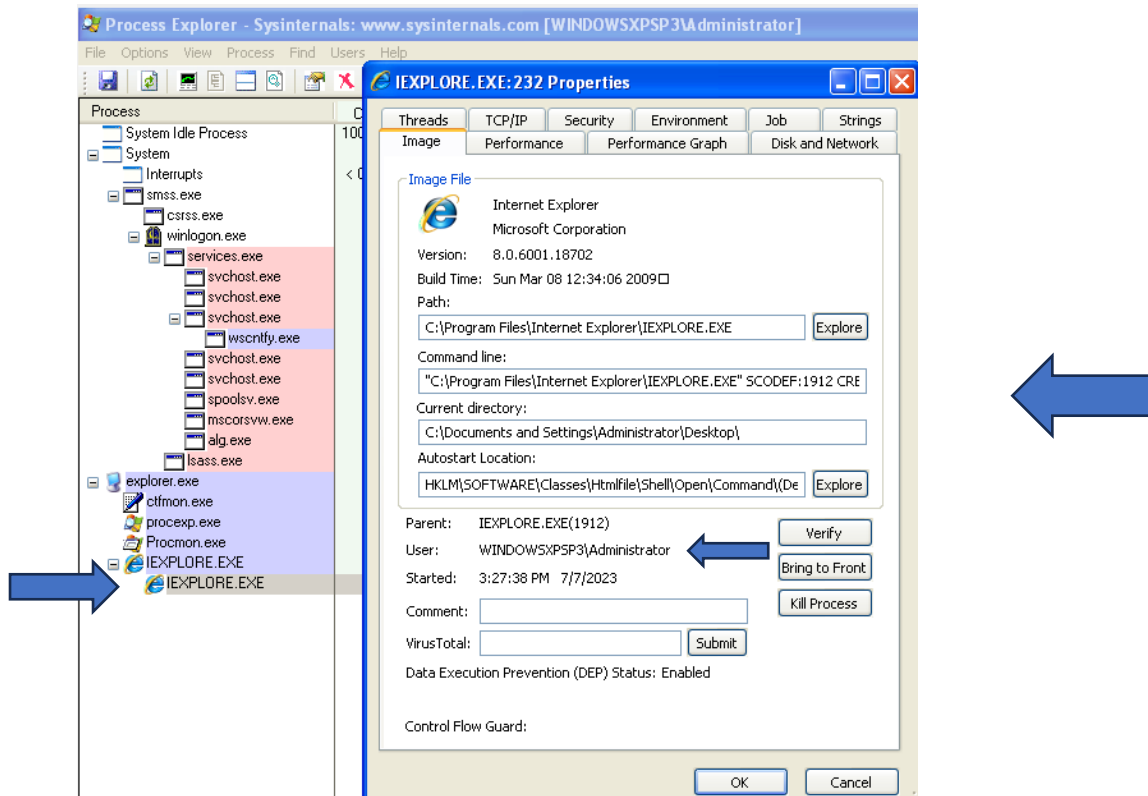
Stop Server



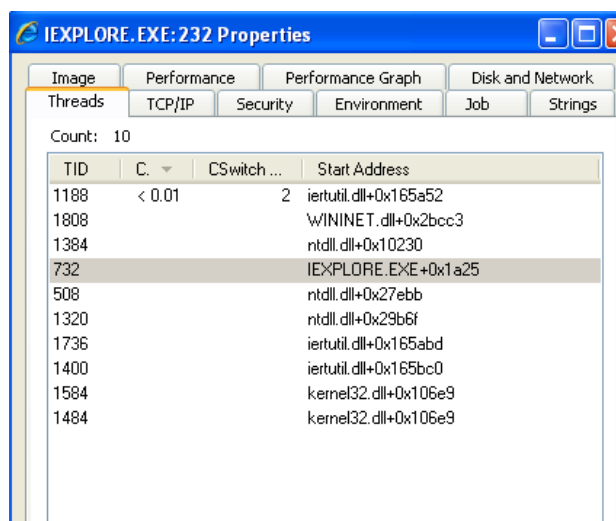
## ANALISI DINAMICA CON PROCESS EXPLORER

Passo ora ad analizzare **Process Explorer** che offre una visualizzazione dettagliata dei processi in esecuzione, inclusi i processi nascosti e i servizi di sistema.

Dopo l'avvio dell'eseguibile, è stato aggiunto un nuovo processo **IEXPLORE.EXE**, facendo dei controlli all'interno del processo, non si notano anomalie. Ad esempio il path è corretto, compreso l'utente.



Anche i **threads** attivi con le relative librerie non destano sospetti. Sono attive infatti librerie di **Kernel32**, **WININET**, **ntdll.dll** che è essenziale per il corretto funzionamento del sistema e **iertutil.dll** che fornisce funzioni e componenti utilizzati da Internet Explorer per la gestione delle operazioni di utilità.



## ANALISI DINAMICA CON PROCESS MONITOR

Infine analizzo i risultati del tool **Process Monitor**.

Dopo aver creato il processo, l'eseguibile va a caricarsi tutte le librerie che gli servono per funzionare (menzionate nell'analisi di VirusTotal). Non noto librerie strane caricate, e non c'è la creazione di nessun file particolare.

3:27:3...	IEEXPLORE.EXE	1912	CreateFile	C:\WINDOWS\system32\advapi32.dll	SUCCESS	Desired Access: R...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\advapi32.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	QueryStandardl...	C:\WINDOWS\system32\advapi32.dll	SUCCESS	AllocationSize: 618...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\advapi32.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	CreateFile	C:\WINDOWS\system32\rpcrt4.dll	SUCCESS	Desired Access: R...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\rpcrt4.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	QueryStandardl...	C:\WINDOWS\system32\rpcrt4.dll	SUCCESS	AllocationSize: 585...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\rpcrt4.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	CreateFile	C:\WINDOWS\system32\secur32.dll	SUCCESS	Desired Access: R...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\secur32.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	QueryStandardl...	C:\WINDOWS\system32\secur32.dll	SUCCESS	AllocationSize: 57...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\secur32.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	CreateFile	C:\WINDOWS\system32\user32.dll	SUCCESS	Desired Access: R...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\user32.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	QueryStandardl...	C:\WINDOWS\system32\user32.dll	SUCCESS	AllocationSize: 581...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\user32.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	CreateFile	C:\WINDOWS\system32\gdi32.dll	SUCCESS	Desired Access: R...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\gdi32.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	QueryStandardl...	C:\WINDOWS\system32\gdi32.dll	SUCCESS	AllocationSize: 585...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\gdi32.dll	SUCCESS	SyncType: S
3:27:3...	IEEXPLORE.EXE	1912	CreateFile	C:\WINDOWS\system32\msvcrt.dll	SUCCESS	Desired Access: R...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\msvcrt.dll	SUCCESS	SyncType: S
3:27:3...	IEEXPLORE.EXE	1912	QueryStandardl...	C:\WINDOWS\system32\msvcrt.dll	SUCCESS	AllocationSize: 585...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\msvcrt.dll	SUCCESS	SyncType: S
3:27:3...	IEEXPLORE.EXE	1912	CreateFile	C:\WINDOWS\system32\shlwapi.dll	SUCCESS	Desired Access: R...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\shlwapi.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	QueryStandardl...	C:\WINDOWS\system32\shlwapi.dll	SUCCESS	AllocationSize: 475...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\shlwapi.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	CreateFile	C:\WINDOWS\system32\shell32.dll	SUCCESS	Desired Access: R...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\shell32.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	QueryStandardl...	C:\WINDOWS\system32\shell32.dll	SUCCESS	AllocationSize: 8,4...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\shell32.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	CreateFile	C:\WINDOWS\system32\ole32.dll	SUCCESS	Desired Access: R...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\ole32.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	QueryStandardl...	C:\WINDOWS\system32\ole32.dll	SUCCESS	AllocationSize: 1,2...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\ole32.dll	SUCCESS	SyncType: SyncTy...
3:27:3...	IEEXPLORE.EXE	1912	CreateFile	C:\WINDOWS\system32\iertutil.dll	SUCCESS	Desired Access: R...
3:27:3...	IEEXPLORE.EXE	1912	CreateFileMap...	C:\WINDOWS\system32\iertutil.dll	SUCCESS	SyncType: SyncTy...

Imposto il filtro su **show network activity**, vedo che c'è una prova di connessione, con un reconnect, infatti la macchina è isolata in rete interna. Anche qui non noto anomalie.

3:27:3...	IEEXPLORE.EXE	1912	UDP Receive	localhost:1042 -> localhost:1042	SUCCESS	Length: 1
3:27:3...	IEEXPLORE.EXE	1912	UDP Send	localhost:1042 -> localhost:1042	SUCCESS	Length: 1
3:27:3...	IEEXPLORE.EXE	232	TCP Reconnect	windowsxpsp3:1041 -> windowsxpsp3:h...	SUCCESS	Length: 0
3:27:3...	IEEXPLORE.EXE	232	TCP Reconnect	windowsxpsp3:1043 -> windowsxpsp3:h...	SUCCESS	Length: 0
3:27:3...	IEEXPLORE.EXE	232	TCP Reconnect	windowsxpsp3:1041 -> windowsxpsp3:h...	SUCCESS	Length: 0
3:27:3...	IEEXPLORE.EXE	232	TCP Disconnect	windowsxpsp3:1041 -> windowsxpsp3:h...	SUCCESS	Length: 0
3:27:3...	IEEXPLORE.EXE	1912	TCP Reconnect	windowsxpsp3:1043 -> windowsxpsp3:h...	SUCCESS	Length: 0
3:27:3...	IEEXPLORE.EXE	1912	TCP Disconnect	windowsxpsp3:1043 -> windowsxpsp3:h...	SUCCESS	Length: 0

## CONCLUSIONI

Dopo aver approfondito con l'analisi statica e dinamica basica l'eseguibile IEXPLORER.EXE, c'è la certezza che **l'eseguibile non è un file malevolo**. Non ho riscontrato dati anomali nelle analisi incrociate dei tool che ho inserito nel report. Per questo, il dipendente giovane appena assunto può star tranquillo e se vuole può ricevere una copia del report sulle analisi fatte sull'eseguibile da lui indicato.