

S10_L5

Unit 3 - CS0424

MATTEO BELTRAMI MARZOLINI
CYBEREAGLES

Giorno 5 - Progetto

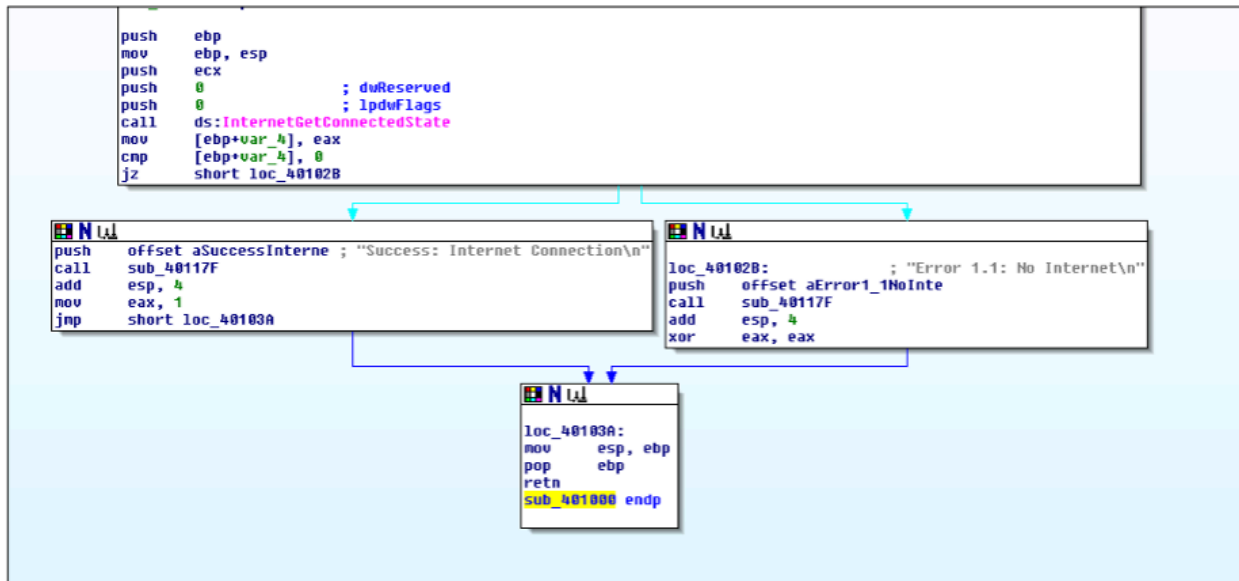
TRACCIA

Con riferimento al file **Malware_U3_W2_L5** presente all'interno della cartella **«Esercizio_Pratico_U3_W2_L5»** sul desktop della macchina virtuale dedicata per l'analisi dei malware, rispondere ai seguenti quesiti:

1. Quali librerie vengono importate dal file eseguibile? Fare anche una descrizione;
2. Quali sono le sezioni di cui si compone il file eseguibile del malware? Fare anche una descrizione;

Con riferimento alla figura in slide 3, risponde ai seguenti quesiti:

3. Identificare i costrutti noti, creazione dello stack, eventuali cicli, altri costrutti;
4. Ipotesizzare il comportamento della funzionalità implementata;
5. Fare una tabella per spiegare il significato delle singole righe di codice.



BONUS:

Un giovane dipendente neo assunto segnala al reparto tecnico la presenza di un programma sospetto.

Il suo superiore gli dice di stare tranquillo ma lui non è soddisfatto e chiede supporto al SOC. Il file "sospetto" è **ieexplore.exe** contenuto nella cartella C:\Programmi\Internet Explorer (no, non ridete ragazzi).

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

Possono essere usati gli strumenti di analisi statica basica e/o analisi dinamica basica visti a lezione. No disassembly no debug o similari.

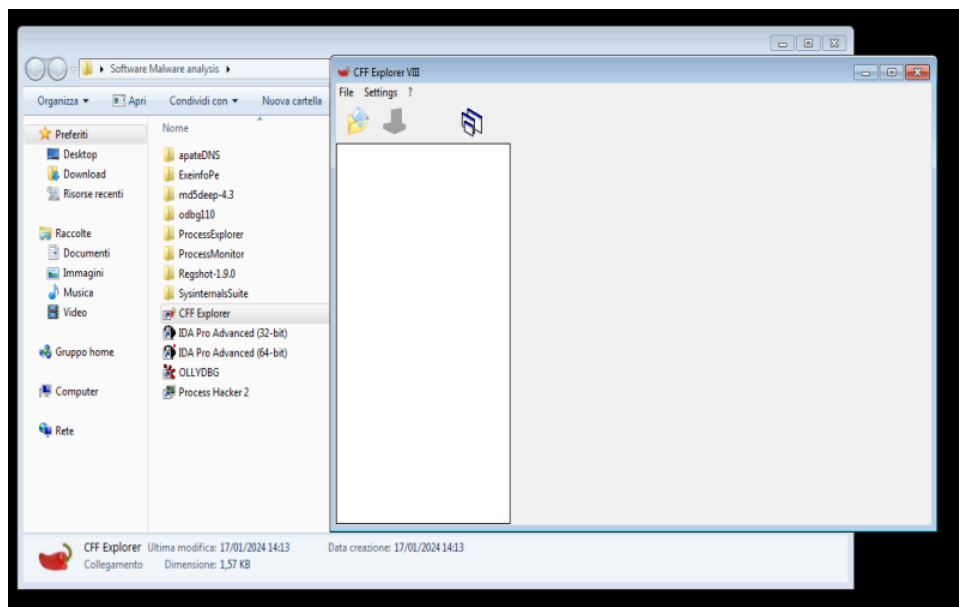
VirusTotal non basta, ovviamente.

Non basta dire ieexplorer è Microsoft quindi è buono, punto.

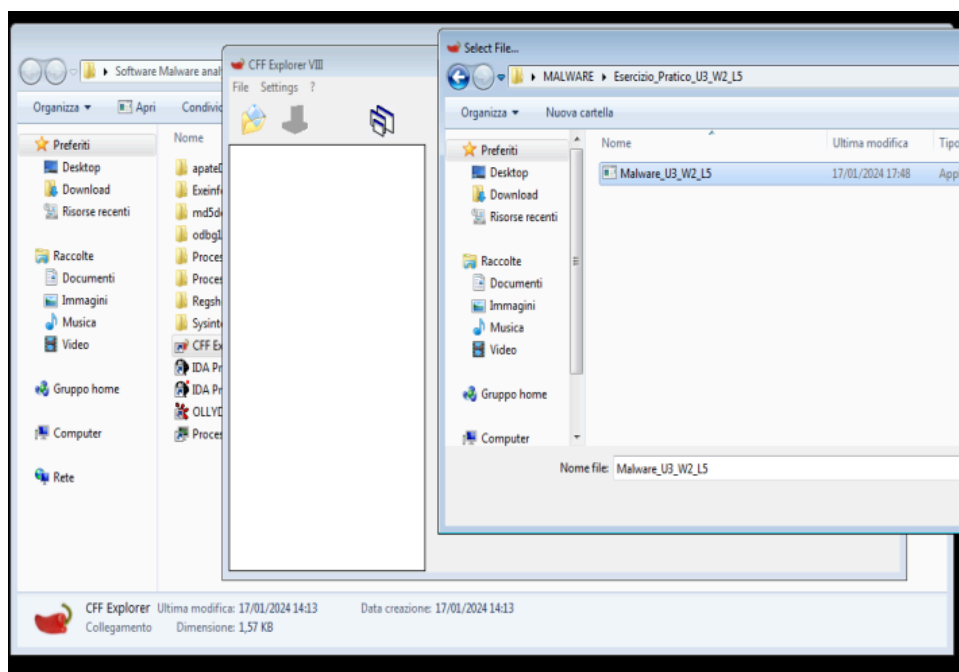
SVOLGIMENTO

Quali librerie vengono importate dal file eseguibile?

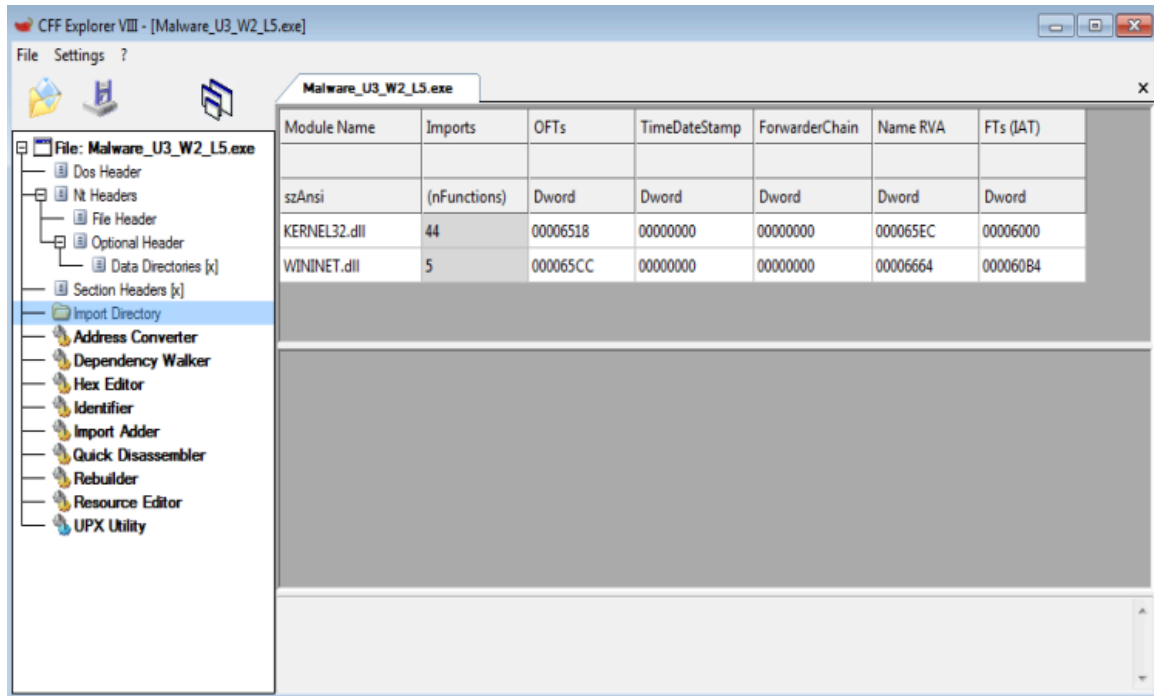
Per procedere all'esercizio, si avvia CFF Explorer.



Si seleziona il ***Malware_U3_W2_L5***, e lo si carica nel programma.



Selezionando la Import Directory, troviamo le librerie che vengono importate dal file eseguibile.



Osservando cosa ci mostra CFF Explorer possiamo notare la presenza di due librerie:

- **Kernel32.dll**
- **Wininet.dll**

1. Kernel32.dll

Questa libreria ha lo scopo di fornire diverse funzioni essenziali per la gestione delle risorse di sistema e l'esecuzione delle applicazioni. E' considerata come una delle librerie più importanti per il funzionamento del sistema operativo Windows, in quanto gestisce la memoria, i file, i thread, i processi e la temporizzazione.

Lo sfruttamento di questa libreria in modo malevolo può garantire:

- l'accesso non autorizzato alle risorse del sistema;
- l'interruzione del servizio;
- e l'escalation dei privilegi.

2. Wininet.dll

Questa libreria fornisce funzioni per connettersi ad internet e quindi gestire le comunicazioni attraverso la rete. Tra le varie funzioni principali abbiamo l'HTTP/FTP, l'autenticazione degli utenti sui siti web e funzioni per la gestione della cache web.

Come si può intendere, tra le varie funzioni abbiamo meccanismi che garantiscono l'accesso sicuro e protetto ai contenuti.

Anche in questo caso non è da sottovalutare l'impatto nel quale venga sfruttata questa libreria in modo malevolo. Infatti essa può garantire, per esempio:

- il bypass della sicurezza della cache;
- il furto delle credenziali;
- e l'intercettazione e manipolazione del traffico internet.

Quindi, **Kernel32.dll** e **Wininet.dll** sono componenti essenziali del sistema operativo Windows, offrendo una vaste funzionalità per la gestione delle risorse di sistema e delle connessioni internet.

Lo sfruttamento di queste due librerie insieme, può variare dal furto di dati personali, alla compromissione completa del sistema, con potenziali danni finanziari e reputazionali per le vittime.

Librerie presenti in **Kernel32.dll**

Module Name	Imports	OFTs	TimeStamp	ForwarderChain	Name RVA	FTs (IAT)
000065EC	N/A	000064DC	000064E0	000064E4	000064E8	000064EC
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.dll	44	00006518	00000000	00000000	000065EC	00006000
OFTs	FTs (IAT)	Hint	Name			
Dword	Dword	Word	szAnsi			
000065E4	000065E4	0296	Sleep			
00006940	00006940	027C	SetStdHandle			
0000692E	0000692E	0156	GetStringTypeW			
0000691C	0000691C	0153	GetStringTypeA			
0000690C	0000690C	01C0	LCMapStringW			
000068FC	000068FC	01BF	LCMapStringA			
000068E6	000068E6	01E4	MultiByteToWideChar			
00006670	00006670	00CA	GetCommandLineA			
00006682	00006682	0174	GetVersion			
00006690	00006690	007D	ExitProcess			
0000669E	0000669E	029E	TerminateProcess			
000066B2	000066B2	00F7	GetCurrentProcess			
000066C6	000066C6	02AD	UnhandledExceptionFilter			
000066E2	000066E2	0124	GetModuleFileNameA			
000066F8	000066F8	00B2	FreeEnvironmentStringsA			
00006712	00006712	00B3	FreeEnvironmentStringsW			
0000672C	0000672C	02D2	WideCharToMultiByte			
00006742	00006742	0106	GetEnvironmentStrings			
0000675A	0000675A	0108	GetEnvironmentStringsW			
00006774	00006774	026D	SetHandleCount			
00006786	00006786	0152	GetStdHandle			
00006796	00006796	0115	GetFileType			

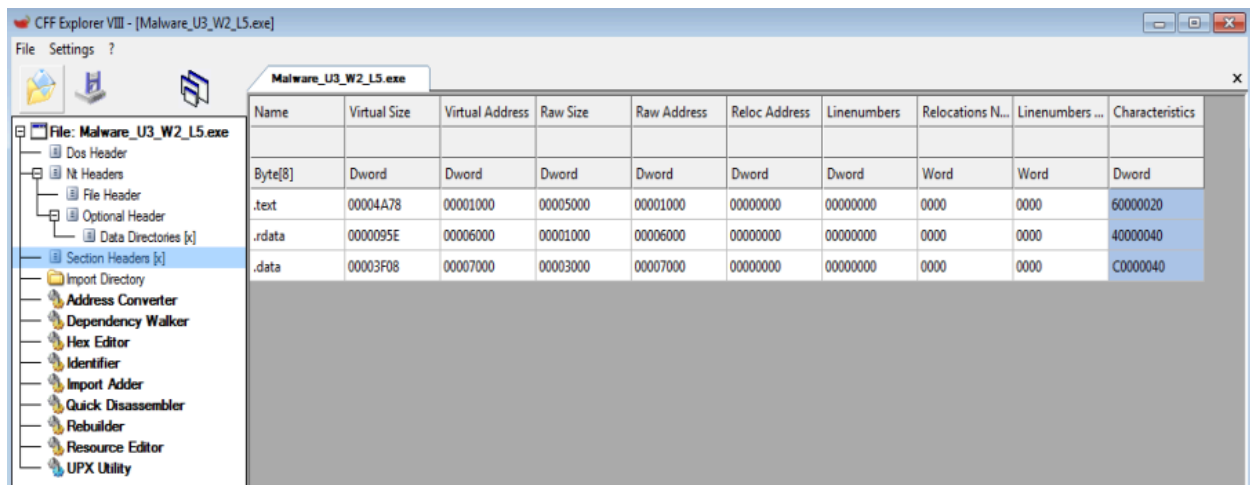
Module Name	Imports	OFTs	TimeStamp	ForwarderChain	Name RVA	FTs (IAT)
000065EC	N/A	000064DC	000064E0	000064E4	000064E8	000064EC
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.dll	44	00006518	00000000	00000000	000065EC	00006000
OFTs	FTs (IAT)	Hint	Name			
Dword	Dword	Word	szAnsi			
000067A4	000067A4	0150	GetStartupInfoA			
000067B6	000067B6	0126	GetModuleHandleA			
000067CA	000067CA	0109	GetEnvironmentVariableA			
000067E4	000067E4	0175	GetVersionExA			
000067F4	000067F4	019D	HeapDestroy			
00006802	00006802	019B	HeapCreate			
00006810	00006810	02BF	VirtualFree			
0000681E	0000681E	019F	HeapFree			
0000682A	0000682A	022F	RtlUnwind			
00006836	00006836	02DF	WriteFile			
00006842	00006842	0199	HeapAlloc			
0000684E	0000684E	00BF	GetCPInfo			
0000685A	0000685A	00B9	GetACP			
00006864	00006864	0131	GetOEMCP			
00006870	00006870	02BB	VirtualAlloc			
00006880	00006880	01A2	HeapReAlloc			
0000688E	0000688E	013E	GetProcAddress			
000068A0	000068A0	01C2	LoadLibraryA			
000068B0	000068B0	011A	GetLastError			
000068C0	000068C0	00AA	FlushFileBuffers			
000068D4	000068D4	026A	SetFilePointer			
00006950	00006950	001B	CloseHandle			

Librerie presenti in **Wininet.dll**

Module Name	Imports	OFTs	TimeStamp	ForwarderChain	Name RVA	FTs (IAT)
00006664	N/A	000064F0	000064F4	000064F8	000064FC	00006500
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.dll	44	00006518	00000000	00000000	000065EC	00006000
WININET.dll	5	000065CC	00000000	00000000	00006664	000060B4
OFTs	FTs (IAT)	Hint	Name			
Dword	Dword	Word	szAnsi			
00006640	00006640	0071	InternetOpenUrlA			
0000662A	0000662A	0056	InternetCloseHandle			
00006616	00006616	0077	InternetReadFile			
000065FA	000065FA	0066	InternetGetConnectedState			
00006654	00006654	006F	InternetOpenA			

Quali sono le sezioni di cui si compone il file eseguibile del malware?

Ritornando su CFF Explorer, andiamo su Section Headers, per cercare le sezioni di cui si compone il file eseguibile del malware.



The screenshot shows the CFF Explorer interface for the file 'Malware_U3_W2_L5.exe'. The 'Section Headers' tab is selected in the left sidebar. The main window displays a table with the following data:

Name	Virtual Size	Virtual Address	Raw Size	Raw Address	Reloc Address	Linenumbers	Relocations N...	Linenumbers ...	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

Da quello che si può osservare, CFF Explorer ci mostra la presenza di 3 sezioni non ofuscate, è quindi leggibili.

Le 3 sezioni presenti sono:

- **.text**
- **.rdata**
- **.data**

Queste 3 sezioni sono parti fondamentali di un file eseguibile di una libreria dinamica sui sistemi operativi Windows.

1. .text

Questa sezione contiene il codice eseguibile del programma, ovvero le istruzioni macchina che vengono eseguite dal processore. Generalmente è una sezione di

sola lettura e non modificabile durante l'esecuzione del programma, per garantire l'integrità del codice.

Lo sfruttamento in modo malevolo di questa sezione può garantire:

- l'alterazione del codice eseguibile, cambiando quindi il comportamento del programma e permettendo attacchi come il bypass di autenticazioni e la disabilitazione delle funzionalità di sicurezza;
- e l'esecuzione di codice arbitrario, quindi dando, ad esempio, la possibilità di iniettare ed eseguire codice arbitrario portando al possibile controllo del sistema da parte dell'attaccante.

2. .rdata

Read-only data, questa sezione contiene i dati di sola lettura che durante l'esecuzione del programma utilizza. Essendo che contiene dati costanti, anch'essa non può essere modificata durante l'esecuzione. Tra i vari contenuti presenti troviamo, stringhe costanti, tabelle di import/export, puntatori a funzione e altri dati.

Lo sfruttamento in modo malevolo di questa sezione può garantire:

- l'accesso a informazioni sensibili, perchè contiene dati come chiavi crittografiche, stringhe importanti e tabelle di funzioni.
- e la manipolazione delle tabelle di import/export, dando la possibilità da un attaccante si reindirizzare le chiamate di funzione a codice dannoso.


.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

This section contains:

Data: 00006000

Import Directory: 0000640C

Import Address Table Directory: 00006000



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Ascii
00000120	78	78	78	08	07	08	00	00	07	00	08	08	08	00	00	08	xxx00000000000
00000130	00	08	00	07	08	00	00	00	28	00	6E	00	75	00	6C	00	0000000000000000
00000140	6C	00	29	00	00	00	00	00	28	6E	75	6C	6C	29	00	00	0000000000000000
00000150	5F	5F	47	4C	4F	42	41	4C	5F	48	45	41	50	5F	53	45	GLOBAL_HEAP_SELECT
00000160	4C	45	43	54	45	44	00	00	5F	5F	4D	53	56	43	52	54	LECTED...MSVCRT
00000170	5F	48	45	41	50	5F	53	45	4C	45	43	54	00	00	00	00	HEAP_SELECT...
00000180	72	75	6E	74	69	6D	65	20	65	72	72	6F	72	20	00	00	runtime.error...
00000190	0D	0A	00	00	54	4C	4F	53	53	20	65	72	72	6F	72	0D	...TLOSS.error...
000001A0	0A	00	00	00	53	49	4E	47	20	65	72	72	6F	72	0D	0A	...SING.error...
000001B0	00	00	00	00	44	4F	4D	41	49	4E	20	65	72	72	6F	72	...DOMAIN.error...
000001C0	0D	0A	00	00	52	36	30	32	38	0D	0A	2D	20	75	6E	61	...R6028...unable
000001D0	62	6C	65	20	74	6F	20	69	6E	69	74	69	61	6C	69	7A	ble.to.initializ
000001E0	65	20	68	65	61	70	0D	0A	00	00	00	00	52	36	30	32	e.heap.....R602
000001F0	37	0D	0A	2D	20	6E	6F	74	20	65	6E	6F	75	67	68	20	7...not.enough.
00000200	73	70	61	63	65	20	66	6F	72	20	6C	6F	77	69	6F	20	space.for.lowio.
00000210	69	6E	69	74	69	61	6C	69	7A	61	74	69	6F	6E	0D	0A	initialization..
00000220	00	00	00	00	52	36	30	32	36	0D	0A	2D	20	6E	6F	74	...R6026...not
00000230	20	65	6E	6F	75	67	68	20	73	70	61	63	65	20	66	6F	enough.space.fo
00000240	72	20	73	74	64	69	6F	20	69	6E	69	74	69	61	6C	69	r.stdio.initiali
00000250	7A	61	74	69	6F	6E	0D	0A	00	00	00	00	52	36	30	32	zation.....R602
00000260	35	0D	0A	2D	20	70	75	72	65	20	76	69	72	74	75	61	5...pure.virtua
00000270	6C	20	66	75	6E	63	74	69	6F	6E	20	63	61	6C	6C	0D	l.function.call.
00000280	0A	00	00	00	52	36	30	32	34	0D	0A	2D	20	6E	6F	74	...R6024...not
00000290	20	65	6E	6F	75	67	68	20	73	70	61	63	65	20	66	6F	enough.space.fo
000002A0	72	20	5F	6F	6E	65	78	69	74	2F	61	74	65	78	69	74	r._onexit/_atexit
000002B0	20	74	61	62	6C	65	0D	0A	00	00	00	00	52	36	30	31	.table.....R601
000002C0	39	0D	0A	2D	20	75	6E	61	62	6C	65	20	74	6F	20	6F	9...unable.to.o
000002D0	70	65	6E	20	63	6F	6E	73	6F	6C	65	20	64	65	76	69	pen.console.devi
000002E0	63	65	0D	0A	00	00	00	00	52	36	30	31	38	0D	0A	2D	ce.....R6018...-
000002F0	20	75	6E	65	78	70	65	63	74	65	64	20	68	65	61	70	unexpected.heap

3. .data

Questa sezione contiene dati variabili del programma, però, in questo caso, possono essere modificati durante l'esecuzione. Quindi questa sezione è leggibile e riscrivibile. I contenuti presenti in questa sezione sono variabili globali e statiche, strutture di dati che vengono inizializzati e usati durante l'esecuzione del programma.

Lo sfruttamento in modo malevolo i di questa sezione può garantire:

- la modifica dei dati, un attaccante potrebbe alterare questi dati per cambiare il comportamento dell'applicazione ed eseguire operazioni non autorizzate.

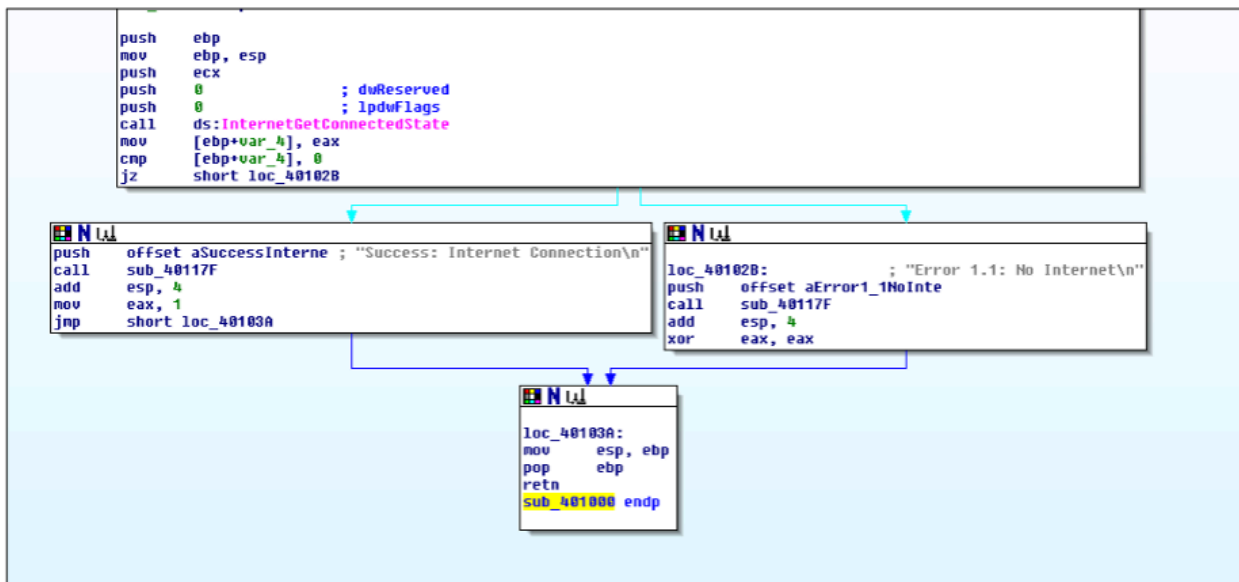
- e cambiando i dati che gestiscono i livelli di accesso o i privilegi, permette ad un attaccante di ottenere permessi maggiori di quelli concessi inizialmente, dando la possibilità di operare con azioni dannose più ampie.

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

This section contains:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Ascii
00000000	00	00	00	00	00	00	00	00	00	00	00	00	09	1C	40	00@.
00000010	64	35	40	00	00	00	00	00	00	00	00	00	AE	1C	40	00	d5@.....@.
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000030	45	72	72	6F	72	20	31	2E	31	3A	20	4E	6F	20	49	6E	Error.1.1:No.In
00000040	74	65	72	6E	65	74	0A	00	53	75	63	63	65	73	73	3A	ternet..Success:
00000050	20	49	6E	74	65	72	6E	65	74	20	43	6F	6E	6E	65	63	.Internet.Connec
00000060	74	69	6F	6E	0A	00	00	00	45	72	72	6F	72	20	32	2E	tion....Error.2.
00000070	33	3A	20	46	61	69	6C	20	74	6F	20	67	65	74	20	63	3:..Fail.to.get.c
00000080	6F	6D	6D	61	6E	64	0A	00	45	72	72	6F	72	20	32	2E	ommand..Error.2.
00000090	32	3A	20	46	61	69	6C	20	74	6F	20	52	65	61	64	46	2:..Fail.to.ReadF
000000A0	69	6C	65	0A	00	00	00	00	45	72	72	6F	72	20	32	2E	ile....Error.2.
000000B0	31	3A	20	46	61	69	6C	20	74	6F	20	4F	70	65	6E	55	1:..Fail.to.OpenU
000000C0	72	6C	0A	00	68	74	74	70	3A	2F	2F	77	77	77	2E	70	rl..http://www.p
000000D0	72	61	63	74	69	63	61	6C	6D	61	6C	77	61	72	65	61	racticalmalvarea
000000E0	6E	61	6C	79	73	69	73	2E	63	6F	6D	2F	63	63	2E	68	alysis.com/cc.h
000000F0	74	6D	00	00	49	6E	74	65	72	6E	65	74	20	45	78	70	ta..Internet.Exp
00000100	6C	6F	72	65	72	20	37	2E	35	2F	70	6D	61	00	00	00	lorer.7.5/pma...
00000110	53	75	63	63	65	73	73	3A	20	50	61	72	73	65	64	20	Success:Parsed.
00000120	63	6F	6D	6D	61	6E	64	20	69	73	20	25	63	0A	00	00	command.is.%c...
00000130	00	1D	40	00	01	00	00	00	48	61	40	00	38	61	40	00	.@...Ha@.8a@.
00000140	00	9F	40	00	00	00	00	00	00	9F	40	00	01	01	00	00	.!@.....!@.
00000150	00	00	00	00	00	00	00	00	00	10	00	00	00	00	00	00+.....
00000160	00	00	00	00	00	00	00	00	00	00	00	00	00	02	00	007.....
00000170	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	007.....
00000180	00	00	00	00	00	00	00	00	00	00	00	00	00	02	00	007.....
00000190	02	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	7.....
000001A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000001B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000001C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000001D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Lo sfruttamento di queste 3 sezioni in modo malevolo può compromettere la sicurezza, l'integrità e l'affidabilità del software. Può portare a furti di dati, interruzioni del servizio e altri tipi di attacchi informatici.



Identificare i costrutti noti, creazione dello stack, eventuali cicli, altri costrutti

Creazione dello Stack

```
push    ebp
mov     ebp, esp
```

Chiamata di funzione dove i parametri sono passati sullo stack tramite le istruzioni di push.

```
push    ecx
push    0           ; dwReserved
push    0           ; lpdwFlags
call    ds:InternetGetConnectedState
```

Costrutto IF

```
cmp     [ebp+var_4], 0
jz      short loc_401028
```

Condizione di successo
connessione

```

push    offset aSuccessInterne ; "Success: Internet Connection\n"
call    sub_40117F
add     esp, 4
mov     eax, 1
jmp     short loc_40103A

```

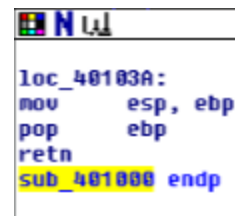
Condizione di errata connessione



```
loc_40102B:                ; "Error 1.1: No Internet\n"
push    offset aError1_1NoInte
call    sub_40117F
add     esp, 4
xor     eax, eax
```

call sub_40117F chiamata a funzione interna (gestisce la visualizzazione dei messaggi)

Ripristino allo stato iniziale dopo la fine della funzione



```
loc_40103A:
mov     esp, ebp
pop     ebp
retn
sub_401000 endp
```

Ipotizzare il comportamento della funzionalità implementata

L'ipotesi del comportamento del codice sembra che consista nel verificare se è disponibile una connessione ad Internet. Successivamente darà un messaggio di successo o di errore in base al risultato. La presenza del Costrutto IF apre la possibilità di eseguire due diverse funzioni in base al risultato. Infatti se la connessione sarà presente, verrà eseguita la funzione di successo altrimenti verrà eseguita quella di errore.

Fare una tabella per spiegare il significato delle singole righe di codice

push ebp	Salva il valore di ebp sullo stack
mov ebp, esp	Imposta il valore di esp su ebp
push ecx	Salva il valore di registro ecx sullo stack
push 0	Passa 0 come parametro per dwReserved a InternetGetConnectedState
push 0	Passa 0 come parametro per lpdwFlags a InternetGetConnectedState
call InternetGetConnectedState	Chiama la funzione per ottenere lo stato della connessione Internet
mov [ebp+var_4], eax	Copia il valore della funzione in una variabile
cmp [ebp+var_4], 0	Confronta il valore con 0 per verificare la connessione
jz short loc_401028	Salta a loc_401028 se il valore è 0, indicando assenza di connessione
push offset aSuccessInterne	Salva l'indirizzo del messaggio di successo nello stack
call sub_40117F	Chiama una funzione per stampare il messaggio del risultato
add esp, 4	Rimuove il parametro dallo stack
mov eax, 1	Imposta eax a 1
jmp short loc_40103A	Salta alla fine della funzione
loc_401028:	-
push offset aError1_1NoInte	Carica il messaggio di errore nello stack
call sub_40117F	Chiama la funzione per gestire il messaggio di errore
add esp, 4	Rimuove il parametro dallo stack

xor eax, eax	Imposta eax a 0
loc_40103A:	-
mov esp, ebp	Ripristina esp al valore di ebp per pulire lo stack
pop ebp	Ripristina il valore di ebp salvato
retn	Restituisce il controllo

BONUS

Un giovane dipendente neo assunto segnala al reparto tecnico la presenza di un programma sospetto.

Il suo superiore gli dice di stare tranquillo ma lui non è soddisfatto e chiede supporto al SOC. Il file "sospetto" è **iexplore.exe** contenuto nella cartella C:\Programmi\Internet Explorer (no, non ridete ragazzi).

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

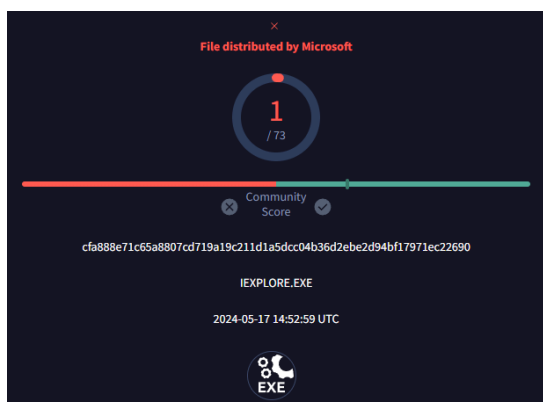
Possono essere usati gli strumenti di analisi statica basica e/o analisi dinamica basica visti a lezione. No disassembly no debug o similari.

VirusTotal non basta, ovviamente.

Non basta dire iexplorer è Microsoft quindi è buono, punto.

SVOLGIMENTO

Dalle prime analisi per controllare cosa fa il programma preso in esame (**iexplore.exe**) si può notare, attraverso un'analisi statica basica, che nel eseguibile è presente un trojan.

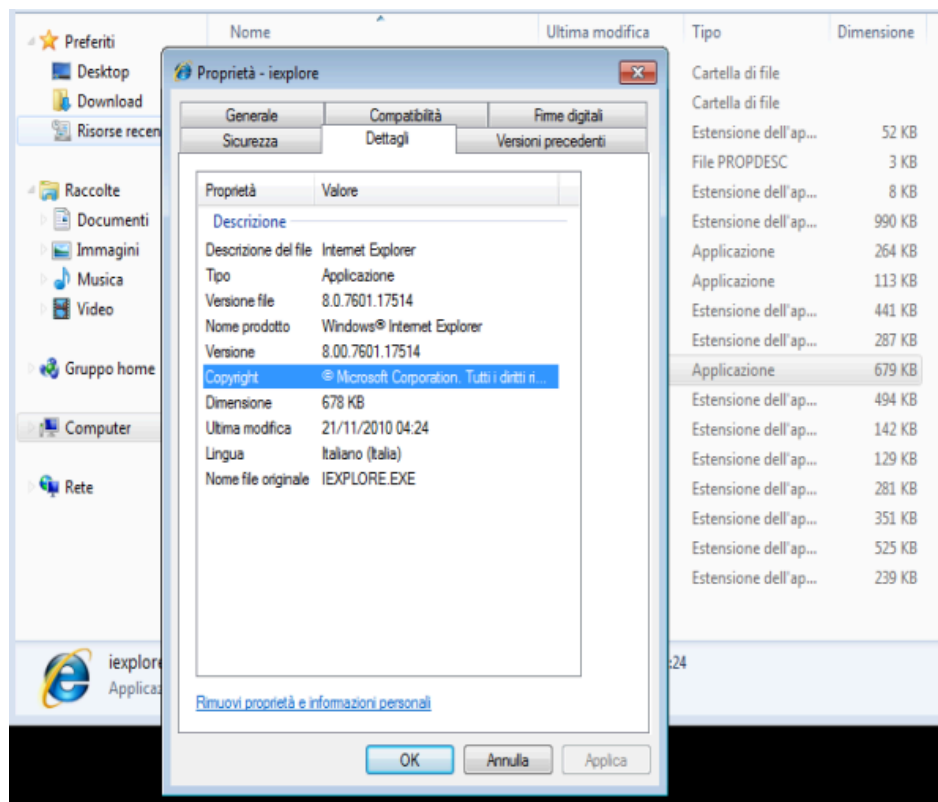


Lionic	! Trojan.Win32.Generic.4!c
Acronis (Static ML)	✓ Undetected
AhnLab-V3	✓ Undetected
Alibaba	✓ Undetected
AliCloud	✓ Undetected
ALYac	✓ Undetected
Antiy-AVL	✓ Undetected
Arcabit	✓ Undetected
Avast	✓ Undetected
AVG	✓ Undetected

Per dimostrare che il file sospetto non è un programma eseguibile maligno, si può procedere attraverso diverse azioni.

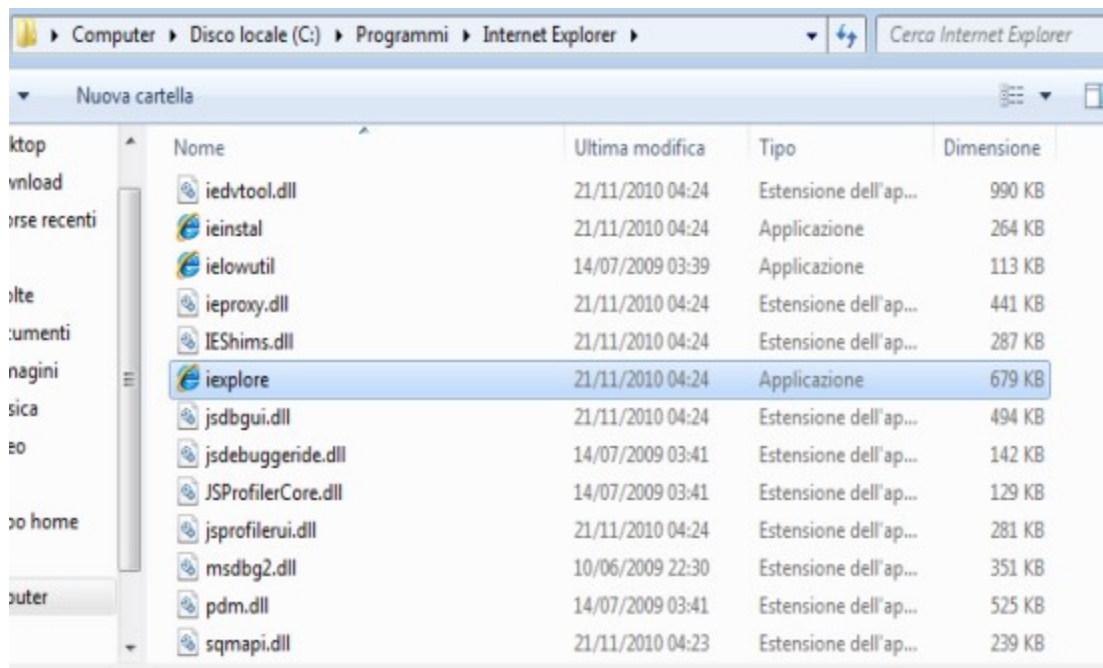
- Verifica della Firma Digitale, dovrebbe essere firmato digitalmente da Microsoft. La presenza di una firma digitale valida e riconosciuta da Microsoft è un forte indicatore dell'autenticità del file;
- Confronto con l'Hash Ufficiale, con il file preso in esame;
- Analisi delle Proprietà del File, controlla le proprietà del file;
- Controllo del Percorso del File.

Senza confrontarsi direttamente con l'hash ufficiale poiché potrebbe non essere facilmente disponibile un hash per un sistema operativo datato come Windows 7, si può dimostrare velocemente che ***ieexplore.exe*** non è un programma maligno mostrando che è un programma rilasciato da Microsoft.



Controllando dalle proprietà dell'eseguibile, si può notare che la presenza della descrizione del file con il nome del prodotto (**Internet Explorer**) (**Windows® Internet Explorer**), la versione del prodotto (**8.00.7601.17514**) ed il copyright di Microsoft.

Inoltre il file si trova nel percorso corretto, ovvero **C:\Programmi\Internet Explorer**



Quindi, basandosi sui controlli effettuati, il file iexplore.exe risulta essere legittimo e non presenta segni di alterazione o compromissione. Si tratta del browser Internet Explorer di Microsoft incluso di default nel sistema operativo Windows 7.