



Marco Bagnaschi

m.bagnaschi@icloud.com

## Malware Analysis

Il Malware da analizzare è nella cartella Build\_Week\_Unit\_3 presente sul desktop della macchina virtuale dedicata.

### Analisi statica

Con riferimento al file eseguibile Malware\_Build\_Week\_U3, rispondere ai seguenti quesiti utilizzando i tool e le tecniche apprese nelle lezioni teoriche:

- Quanti parametri sono passati alla funzione Main()?
- Quante variabili sono dichiarate all'interno della funzione Main()?
- Quali sezioni sono presenti all'interno del file eseguibile? Descrivete brevemente almeno 2 di quelle identificate
- Quali librerie importa il Malware? Per ognuna delle librerie importate, fate delle ipotesi sulla base della sola analisi statica delle funzionalità che il Malware potrebbe implementare. Utilizzate le funzioni che sono richiamate all'interno delle librerie per supportare le vostre ipotesi.

### Svolgimento :

Risposta alla prima domanda : 3 parametri

Risposta alla seconda domanda : 5 variabili

Spiegazione + Screenshot

Eseguiamo il software IDAPro in modalità amministratore. Una volta avviato il software con la funzione apri importiamo il malware in questione, che grazie alle peculiarità del software di analisi ci proporrà a prima schermata la funzione main del codice.

Possiamo oltre ad individuare i parametri all'interno delle parentesi, fare riferimento agli offset e notare come quelli delle variabili siano preceduti da segno negativo mentre quelli dei parametri abbiano segno positivo.

The screenshot shows the IDA Pro interface with the title bar "Windows 7 - malware analysis (Istantanea 1 - Mal Aperta) [Running]". The menu bar includes File, Edit, Jump, Search, View, Debugger, Options, Windows, Help. The toolbar has various icons for file operations, search, and analysis. The status bar at the bottom shows the path "IDA - C:\Users\user\Desktop\MALWARE\Build\_Week\_Unit\_3\Malware\_Build\_Week\_U3.exe" and the message "No debugger".

The main window displays the assembly view. On the left is the "Functions window" showing function names and their segments:

- sub\_401000 .text
- sub\_401080 .text
- \_main .text
- sub\_401299 .text
- \_fclose .text
- \_fwrite .text
- \_fopen .text
- \_fopen .text
- \_strchr .text

The assembly code for the main function is shown in the central pane:

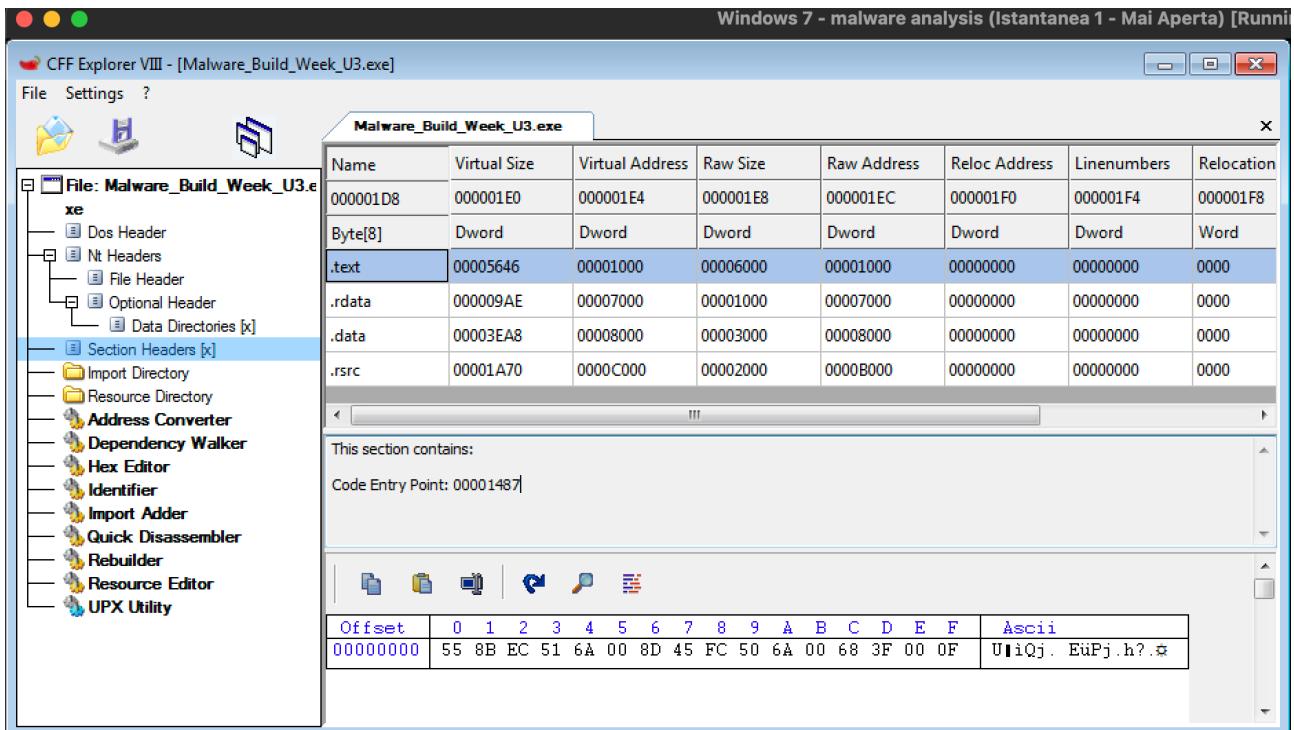
```
; Attributes: bp-based frame
; int __cdecl main(int argc, const char **argv, const char **envp)
_main proc near

hModule= dword ptr -11Ch
Data= byte ptr -118h
var_117= byte ptr -117h
var_8= dword ptr -8
var_4= dword ptr -4
argc= dword ptr 8
argv= dword ptr 0Ch
envp= dword ptr 10h
```

Risposta alla terza domanda : all'interno dell'eseguibile sono presenti 4 sezioni .text .rdata .data .rsrc

La sezione .text è quella che contiene tutte le istruzioni che saranno effettivamente eseguite dalla CPU, mentre la sezione .data contiene le variabili globali del programma ( quelle dichiarate non all'interno di funzioni )

Per questa estrazione di informazioni il tool utilizzato è stato CFF explorer eseguito in modalità di amministratore.



Risposta alla quarta domanda : il malware importa due librerie che sono KERNEL32.dll e ADVAPI32.dll

Direttamente dal materiale fornитoci da Epicode spieghiamo la funzione principale di queste librerie :

Kernel32.dll: libreria piuttosto comune che contiene le funzioni principali per interagire con il sistema operativo, ad esempio: manipolazione dei file, la gestione della memoria.

Advapi32.dll: libreria che contiene le funzioni per interagire con i servizi ed i registri del sistema operativo Microsoft

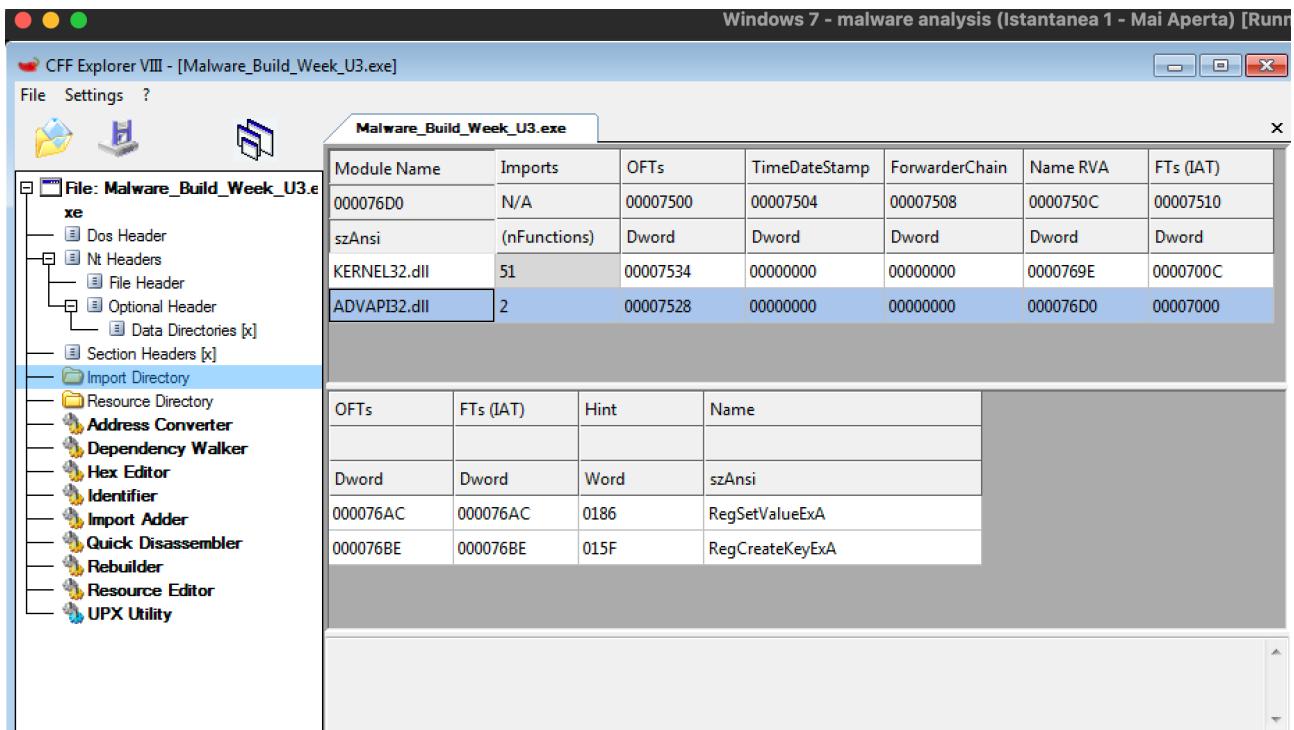
Con riferimento alla seconda libreria ci accorgiamo che richiama le funzioni RegCreateKeyExA ( Crea la chiave del Registro di sistema specificata. Se la chiave esiste già, la funzione lo apre. ) e RegSetValueExA ( Imposta i dati e il tipo di un valore specificato in una chiave del registro di sistema )

Quanto scritto sopra ci fa ipotizzare che il malware andrà a scrivere o modificare il registro di Windows. Una delle possibilità è che stia cercando di garantirsi la persistenza.

Link alla pagina Microsoft delle funzioni sopra elencate

<https://learn.microsoft.com/it-it/windows/win32/api/winreg/nf-winreg-regcreatekeyex>

<https://learn.microsoft.com/it-it/windows/win32/api/winreg/nf-winreg-regsetvalueex>



## Malware Analysis

Con riferimento al Malware in analisi, spiegare:

Lo scopo della funzione chiamata alla locazione di memoria 00401021

Come vengono passati i parametri alla funzione alla locazione 00401021;

Che oggetto rappresenta il parametro alla locazione 00401017

Il significato delle istruzioni comprese tra gli indirizzi 00401027 e 00401029.

Con riferimento all'ultimo quesito, tradurre il codice Assembly nel corrispondente costrutto C.

Valutate ora la chiamata alla locazione 00401047, qual è il valore del parametro «ValueName»?

Per le domande sopra citate viene utilizzato per la maggiore il software IDA pro

Domanda 1:

Alla locazione di memoria 00401021 viene chiamata la funzione "RegCreateKeyExA"

Come da link Microsoft in precedenza questa funzione crea oppure apre ( se già esistente ) una determinata chiave di registro

```
00401017: push    offset SubKey      ; "SOFTV
0040101C: push    80000002h        ; hKey
00401021: call    ds:RegCreateKeyExA
00401027: test    eax, eax
00401029: jz     short loc_401032
0040102B: mov     eax, 1
00401030: jmp     short loc_40107B
```

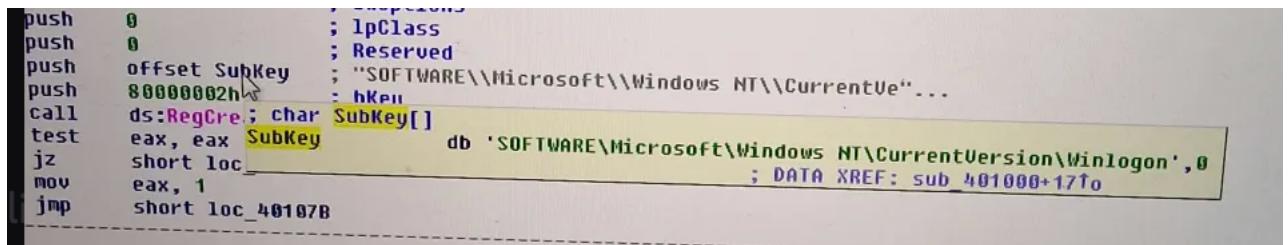
## Domanda 2:

Viene creato uno stack per la funzione chiamata e i parametri vengono passati sullo stack con delle push

```
.text:00401000      push    ebp
..text:00401000      mov     ebp, esp
.text:00401001      push    ecx
.text:00401003      push    0
.text:00401004      push    0          ; lpdwDisposition
.text:00401006      lea    eax, [ebp+hObject]
.text:00401009      push    eax
.text:0040100A      push    0          ; phkResult
.text:0040100C      push    0F003Fh
.text:00401011      push    0          ; lpSecurityAttributes
.text:00401013      push    0          ; samDesired
.text:00401015      push    0          ; dwOptions
.text:00401017      push    offset SubKey
.text:0040101C      push    80000002h
.text:00401021      call    ds:RegCreateKeyExA
```

## Domanda 3:

Sull'analisi della singola riga viene creato l'offset per poter contenere il valore dell'oggetto SubKey o sottochiave o sottocartella del registro di Windows che verrà creata dal Malware



```
push    0          ; lpClass
push    0          ; Reserved
push    offset SubKey
push    80000002h
call    ds:RegCreateKeyExA
test   eax, eax
jz     short loc_
mov    eax, 1
jmp    short loc_40107B
```

## Domanda 4:

Le istruzioni sono una test ed un jz

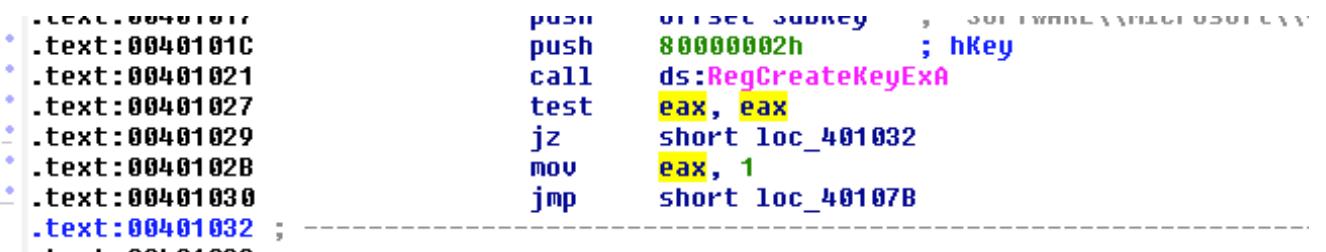
La test, come da documentazione Epicode, è molto simile a un'istruzione AND ma rispetto ad essa non modifica il valore contenuto negli operandi

Nel nostro caso la test è sullo stesso registro ... test EAX, EAX

In questo possiamo affermare che la test viene utilizzata per controllare se un valore è zero

In caso di riscontro lo ZF ( Zero Flag ) viene settato ad 1 e quindi viene poi eseguita l'istruzione successiva ... jz ShortLoc...

Questo salto condizionale viene effettuato se lo Zero Flag è a 1 ( Jump Zero )



```
.text:00401010      push    0
.text:00401010      push    0          ; lpClass
.text:00401010      push    0          ; Reserved
.text:00401010      push    offset SubKey
.text:00401010      push    80000002h
.text:00401010      call    ds:RegCreateKeyExA
.text:00401010      test   eax, eax
.text:00401010      jz     short loc_401032
.text:00401010      mov    eax, 1
.text:00401010      jmp    short loc_40107B
```

Domanda 5:

Traduzione delle istruzioni alla domanda precedente in linguaggio C

Test eax, eax → if ( var == 0 )

Jz short loc\_401032 → esegui codice contenuto tra le parentesi graffe

```
If ( var == 0 ){
    Istruzioni
}
```

Domanda 6:

lpValueName di tipo LPCSTR ( Stringa ) è “GinaDLL”

The screenshot shows the IDA Pro interface with assembly code and memory dump panes. The assembly pane displays the following code:

```
.text:00401035      push    ecx          ; cbData
.text:00401036      mov     edx, [ebp+lpData] ; lpData
.text:00401039      push    edx          ; dwType
.text:0040103A      push    1             ; Reserved
.text:0040103C      push    0             ; Reserved
.text:0040103E      push    offset ValueName ; "GinaDLL"
.text:00401043      mov     eax, [ebp+hObject]
.text:00401046      push    eax          ; hKey
.text:00401047      call    ds:RegSetValueExA
.text:0040104D      test   eax, eax
```

The memory dump pane shows the string "GinaDLL" at address 0040103E.

Imports from ADVAPI32.dll:

```
LSTATUS __stdcall RegSetValueExA(HKEY hKey, LPCSTR lpValueName, DWORD Reserved, DWORD dwType, const BYTE *lpData, DWORD cbData)
extrn RegSetValueExA:dword ; CODE XREF: sub_401000+47↑p
```

Segment type: Externs

idata

Loading IDP module C:\Program Files (x86)\IDA\procs\pc64.w64 for processor metapc...OK

Analysis subsystem has been initialized.

Database for file 'Malware\_Build\_Week\_U3.exe' is loaded.

Compiling file 'C:\Program Files (x86)\IDA\idc\ida.idc'...

Executing function 'main'...

Can not set debug privilege: Non tutti i privilegi o i gruppi menzionati sono assegnati al chiamante.

IDC

AU: idle Down Disk: 24GB

## Malware Analysis

-Cosa notate all'interno della cartella dove è situato l'eseguibile del Malware? Spiegate cosa è avvenuto, unendo le evidenze che avete raccolto finora per rispondere alla domanda

Filtrate includendo solamente l'attività sul registro di Windows.

-Quale chiave di registro viene creata?

-Quale valore viene associato alla chiave di registro creata?

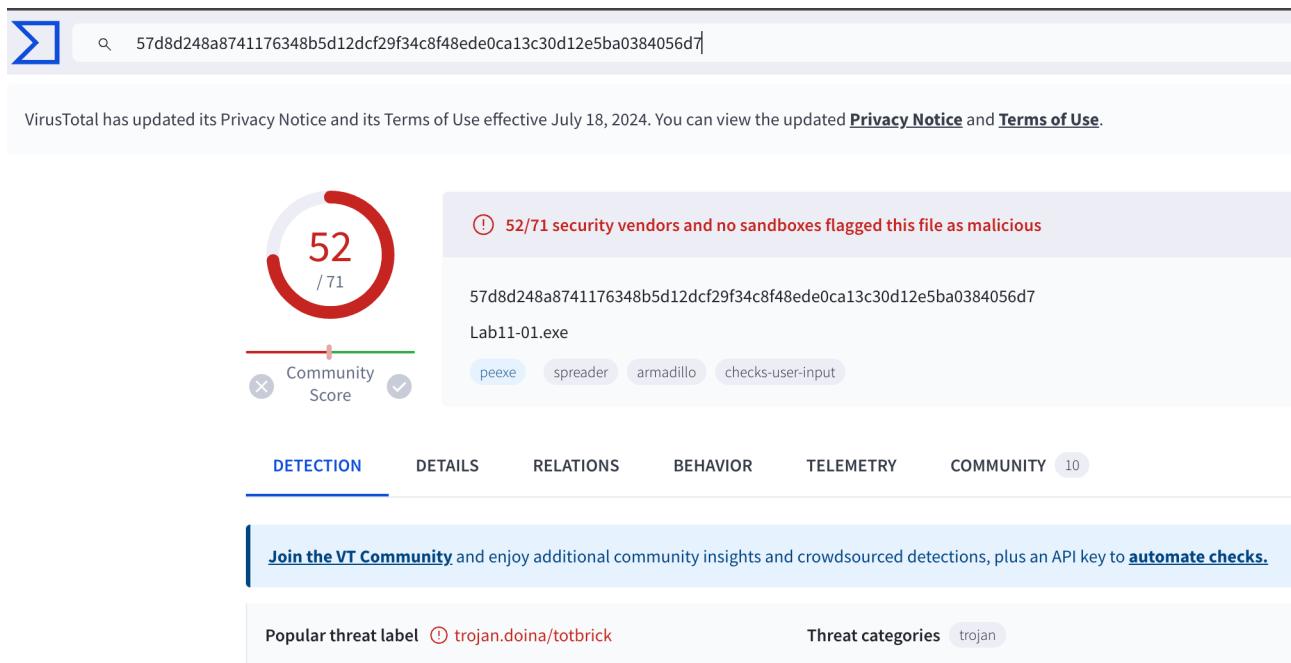
Passate ora alla visualizzazione dell'attività sul file system.

-Quale chiamata di sistema ha modificato il contenuto della cartella dove è presente l'eseguibile del Malware?

Unite tutte le informazioni raccolte fin qui sia dall'analisi statica che dall'analisi dinamica per delineare il funzionamento del Malware.

## Analisi del Malware

Come prima azione, quando nella cartella c'era solo il malware ( Malware\_Build\_Week\_U3 ) si è fatta un' analisi con CFF Explorer in modo da poter accedere all'Hash del file e confrontarlo con un tool come Virus Total



Da una cattura della parte iniziale della schermata si capisce che abbiamo a che fare con un Trojan, mentre alcune analisi dei vari tool inseriti nella pagina web ci suggeriscono in particolare della presenza di un trojan-dropper

Dopo aver eseguito il malware ci accorgiamo che viene creata all'interno della cartella msgina32.dll

MALWARE > Build_Week_Unit_3			
a	Includi nella raccolta	Condividi con	Nuova cartella
Nome	Ultima modifica	Tipo	
sktop	Malware_Build_Week_U3	17/01/2024 17:48	Applicazione
wnload	Malware_Build_Week_U3	19/04/2024 21:35	File I64
orse recenti	Malware_Build_Week_U3.id0	21/04/2024 16:56	File ID0
olte	Malware_Build_Week_U3.id1	21/04/2024 16:56	File ID1
cumenti	Malware_Build_Week_U3.nam	21/04/2024 16:56	File NAM
agini	Malware_Build_Week_U3.til	21/04/2024 16:56	File TIL
rsica	msgina32.dll	21/04/2024 18:58	Estensione dell'ap...
eo			

Anche la dll creata è stata passata a VirusTotal per confronto e otteniamo lo stesso risultato

51 /71

Community Score

① 51/71 security vendors and no sandboxes flagged this file as malicious

f8a4f61bcccd5bab1cad0ab9e57f6f3092a8bd4dd0adfc4d853e89ba96afc93f9  
msgina32.dll

Size 6.50 KB | Last Modification Date 20 days ago | DLL

Detection Details Relations Behavior Telemetry Community 4

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

Popular threat label	trojan.fragtor/tiggre	Threat categories	trojan	Family labels	fragtor tiggre
Security vendors' analysis	①				Do you want to automate checks?
Alibaba	① Trojan:Win32/Tiggre.387d5a16	AliCloud	①	Trojan.Win.Generic.f3be1728	
ALYac	① Gen:Variant.Fragtor.510142	Antiy-AVL	①	Trojan/Win32.FakeGina	
Arcabit	① Trojan.Fragtor.D7C8BE	Avast	①	Win32:Trojan-gen	
AVG	① Win32:Trojan-gen	Avira (no cloud)	①	HEUR/AGEN.1326250	
BitDefender	① Gen:Variant.Fragtor.510142	BitDefenderTheta	①	Gen:NN.ZedlaF.36802.aq4@a0clrOb	

Attraverso l'utilizzo di Proces Monitor e con l'ausilio di un filtro ad hoc come mostrato nelle slide procediamo alla cattura della parte di registro

Guardando le operazioni ritroviamo la RegCreateKey e la RegSetValue che avevamo incontrato prima nel codice assembly analizzato con il software IDA pro

Qui di seguito le evidenze delle operazioni sul registro come la creazione ed il set value e nella parte Data il valore contenuto nella chiave che punta alla dll malevola

16:21...	Malware_Build_...	2588	RegOpenKey	HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Diagnos	SUCCESS	NAME NOT FOUND Desired Access: Read
16:21...	Malware_Build_...	2588	RegQueryKey	HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Diagnos	SUCCESS	Query: HandleTags, HandleTags: 0x0
gio	16:21...	Malware_Build_...	2588	RegCreateKey	SUCCESS	Desired Access: All Access, Disposition: REG_OPENED_EXISTING_KEY
di	16:21...	Malware_Build_...	2588	RegSetInfoKey	SUCCESS	KeySetInformationClass: KeySetHandleTagsInformation, Length: 0
16:21...	Malware_Build_...	2588	RegQueryKey	HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon	SUCCESS	Query: HandleTags, HandleTags: 0x400
16:21...	Malware_Build_...	2588	RegSetValue	HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon\GinaDLL	SUCCESS	Type: REG_SZ, Length: 520, Data: C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3\msgina32.dll
16:21...	Malware_Build_...	2588	RegCloseKey	HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon	SUCCESS	
16:21...	Malware_Build_...	2588	RegInseKey	HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Encryption Options	SUCCESS	
<hr/>						
HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon						
					SUCCESS	Desired Access: All Access, Disposition: REG_OPENED_EXISTING_KEY
					SUCCESS	KeySetInformationClass: KeySetHandleTagsInformation, Length: 0
					SUCCESS	Query: HandleTags, HandleTags: 0x400
					SUCCESS	Type: REG_SZ, Length: 520, Data: C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3\msgina32.dll

Viene fatta una chiamata di sistema ( CreateFile ) che crea la msgina.dll nella cartella del malware e a seguire vediamo la write file che inserisce il contenuto malevolo e poi la close file

Qui sotto l'evidenza

00:43:...	Malware_Build_...	1644	CreateFileMapping	C:\Windows\SysWOW64\sechost.dll	SUCCESS	Query: HandleTags, HandleTags: 0x0
00:43:...	Malware_Build_...	1644	CloseFile	C:\Windows\SysWOW64\sechost.dll	SUCCESS	
00:43:...	Malware_Build_...	1644	CreateFile	C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3\msgina32.dll	SUCCESS	Desired Access: All Access, Disposition: REG_OPENED_EXISTING_KEY
00:43:...	Malware_Build_...	1644	WriteFile	C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3\msgina32.dll	SUCCESS	KeySetInformationClass: KeySetHandleTagsInformation, Length: 0
00:43:...	Malware_Build_...	1644	WriteFile	C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3\msgina32.dll	SUCCESS	Query: HandleTags, HandleTags: 0x400
00:43:...	Malware_Build_...	1644	CloseFile	C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3\msgina32.dll	SUCCESS	Type: REG_SZ, Length: 520, Data: C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3\msgina32.dll

Premessa: il Virus è stato eseguito non con la modalità del doppio click ma con la modalità Esegui come Amministratore, altrimenti alcune operazione non andavano a buon fine.

Da questo possiamo dedurre che questo codice non ha una forte Privilege Escalation, in quanto eseguito come utente normale potrebbe non portare a compimento le azioni.

From chat-gpt 3.5 :

"""

La DLL msgina.dll (Graphical Identification and Authentication) è una libreria di sistema di Windows che gestisce l'interfaccia di autenticazione grafica. È responsabile dell'interazione tra l'utente e il processo di autenticazione durante l'accesso al sistema operativo Windows. Questa DLL è fondamentale per il processo di login e fornisce l'interfaccia grafica per l'inserimento delle credenziali utente come nome utente e password. Quando un utente accede al sistema, msgina.dll viene caricato per consentire il processo di autenticazione e l'accesso all'account utente.

"""

Il comportamento globale del malware è quello della creazione e/o apertura della chiave di registro Winlogon con inserimento nella stessa il valore ( PATH ) che punta alla dll malevola creata dopo la sua esecuzione.

Essendo la dll in questione fondamentale per l'autenticazione degli utenti su windows, possiamo presupporre che lo scopo del software malevolo sia quello di registrare le autenticazioni da parte degli utenti al sistema per poi impossessarsene.