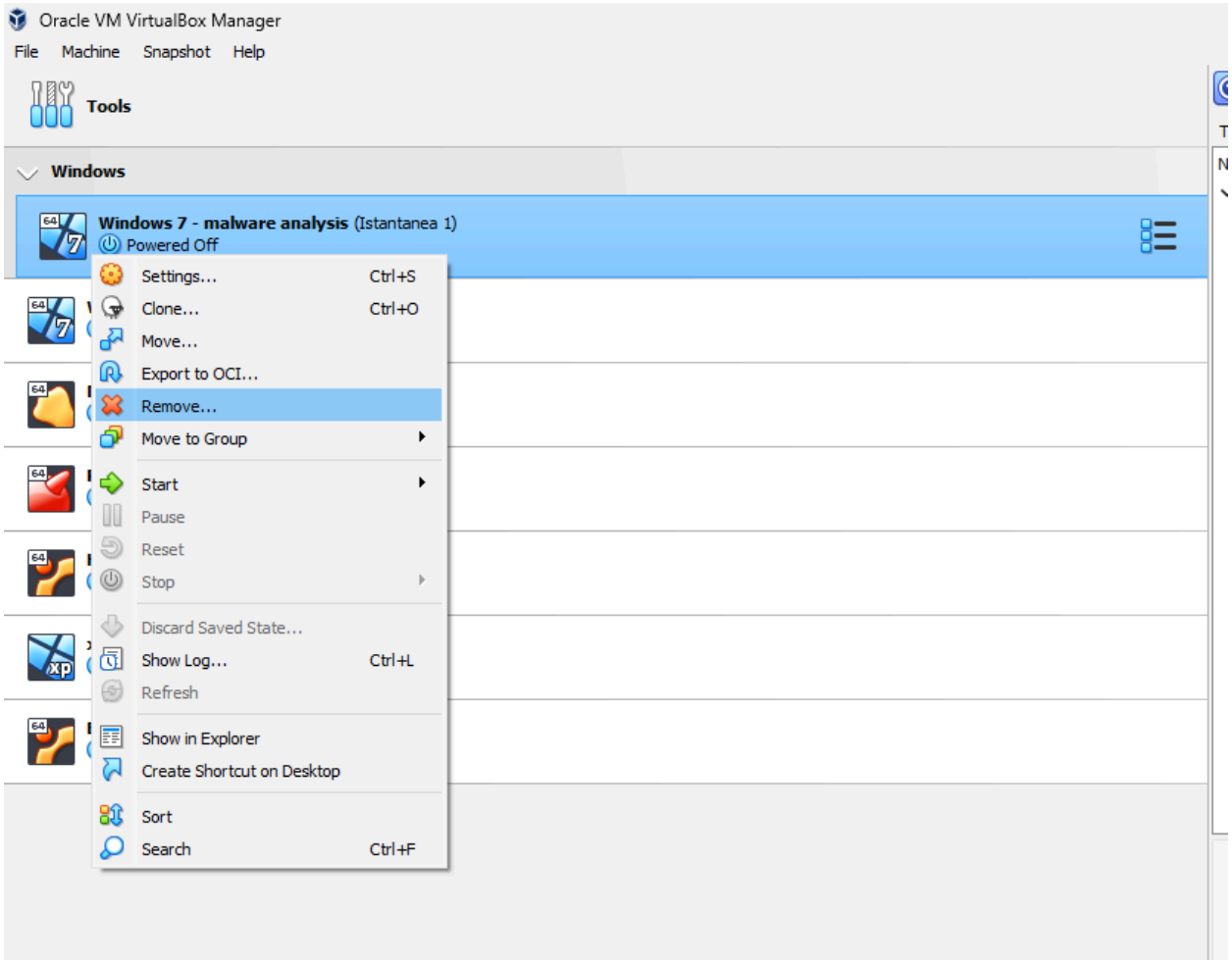
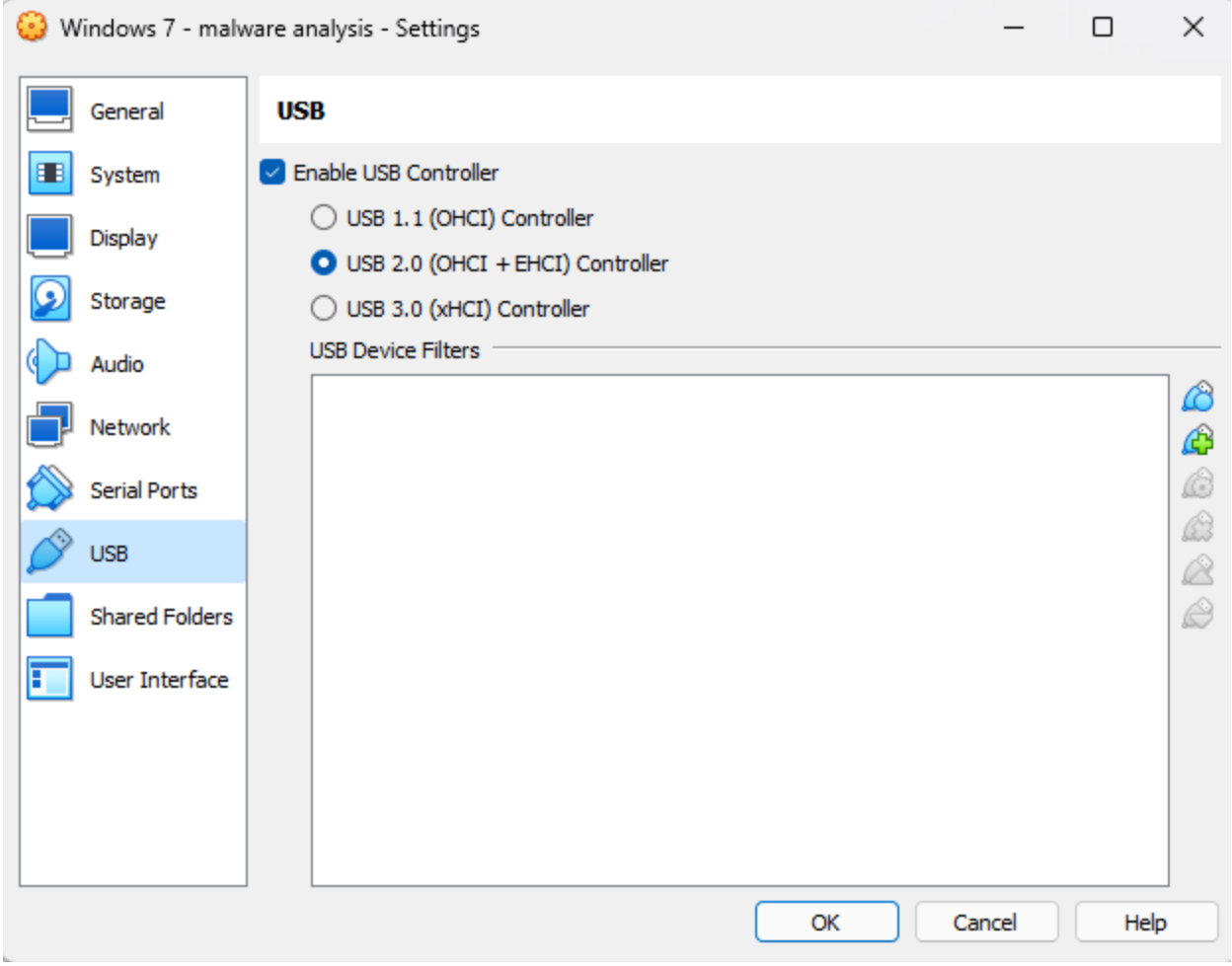
Progetto di fine modulo 6 - Analisi malware

Prima di effettuare una qualunque analisi è buona pratica creare un ambiente sandbox , ossia ambiente “sterile” isolato dalla macchina host così da poter effettuare analisi in sicurezza

Per fare ciò apriamo VirtualBox e clicchiamo tasto destro sulla macchina virtuale che utilizzeremo,

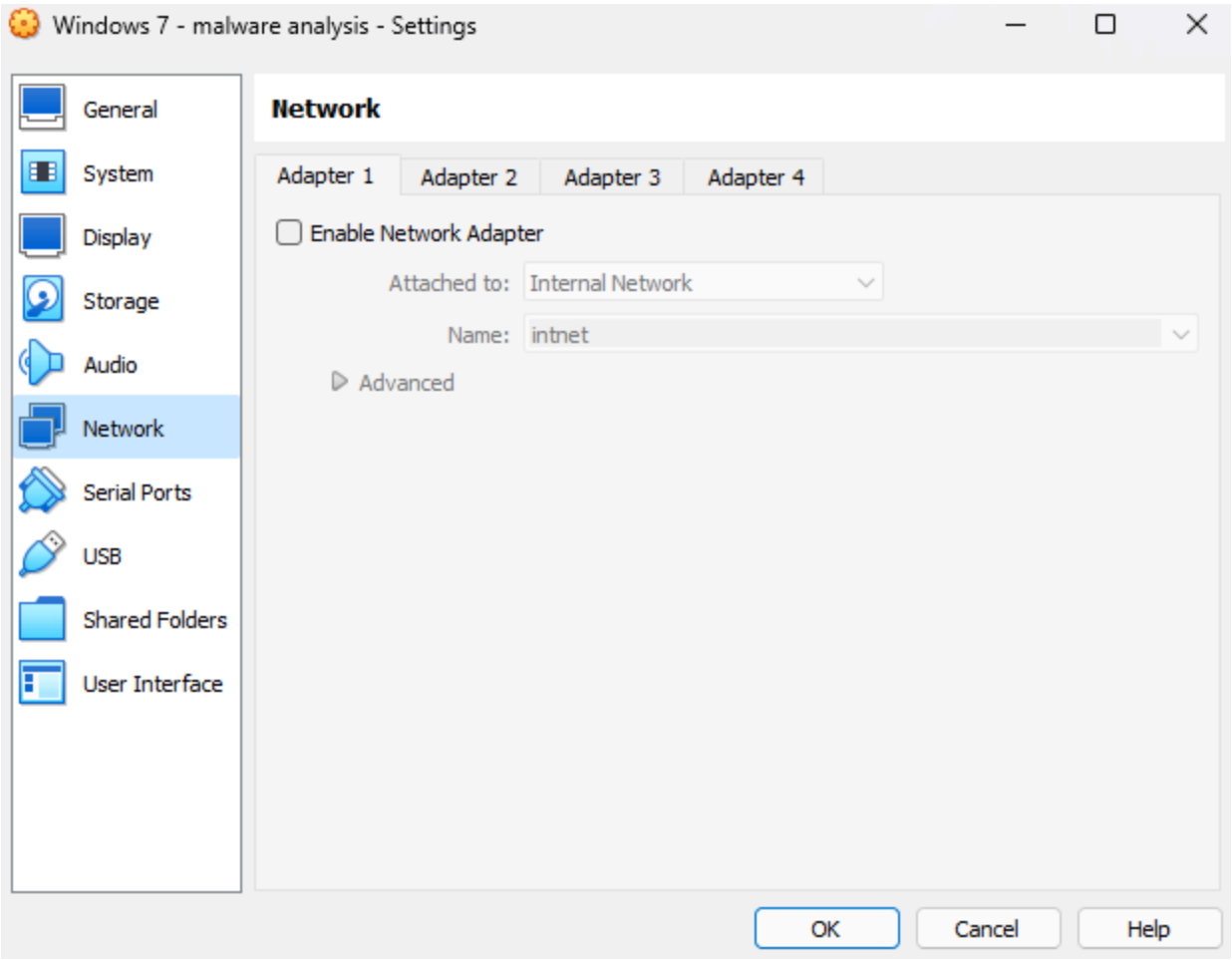


successivamente andiamo in impostazioni e poi usb

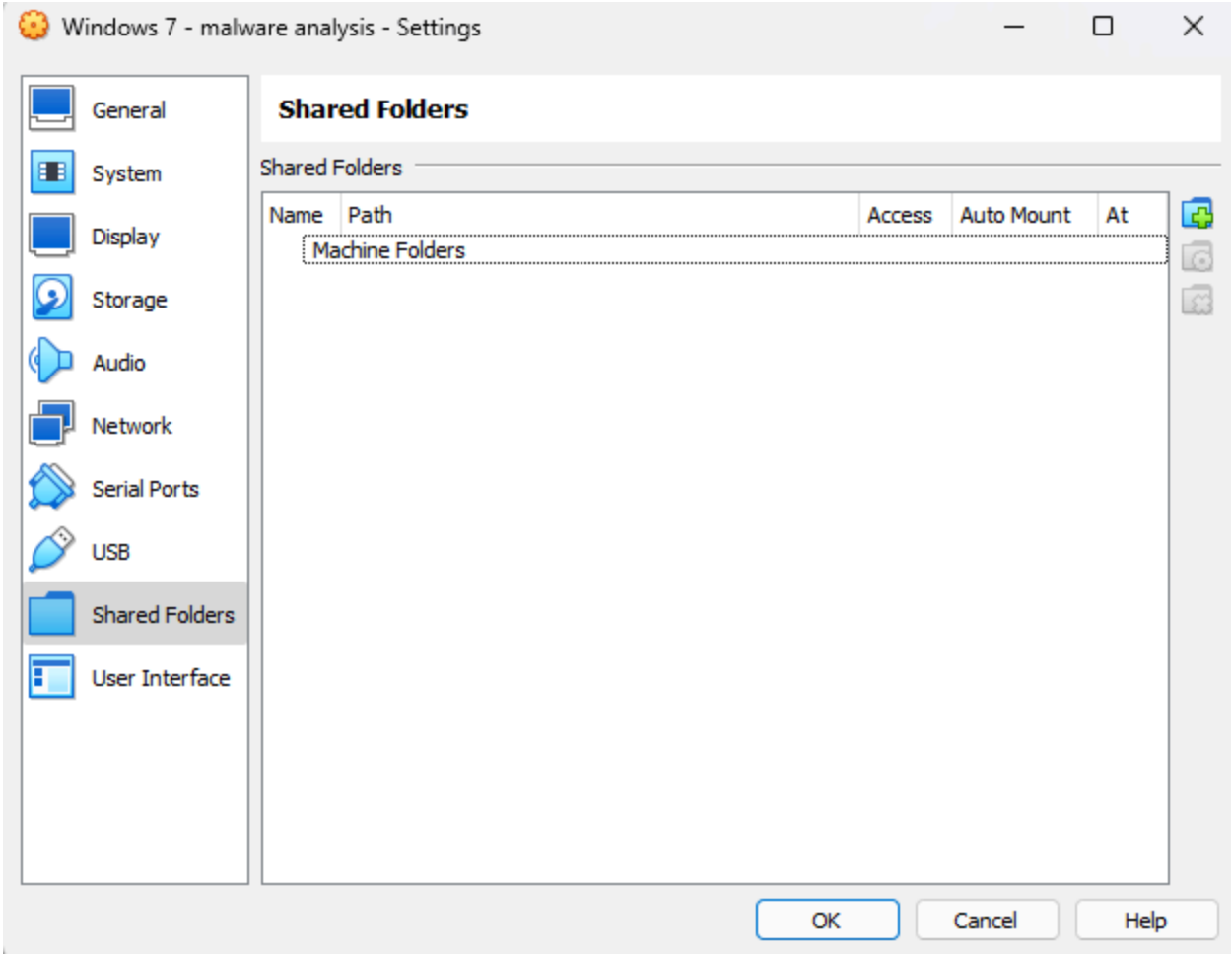


togliamo dunque la spunta da “enable usb controller”

Spostiamoci poi su network e eliminiamo l’interfaccia di rete per l’analisi statica (per l’analisi dinamica abiliteremo un'interfaccia di rete)


Windows 7 - malware analysis - Settings
General
System
Display
Storage
Audio
Network
Serial Ports
USB
Shared Folders
User Interface
Network
Adapter 1
Adapter 2
Adapter 3
Adapter 4
Enable Network Adapter
Attached to:
Internal Network
Name:
intnet
Advanced
OK
Cancel
Help

Assicuriamoci che in “shared folders” non ci siano cartelle condivise con la macchina host , in quanto il malware potrebbero essere sfruttate da un malware per propagarsi all’esterno della macchina virtuale.



infine disabilitiamo il drag n drop per la da e verso la vm , per fare ciò lanciamo la vm è successivamente clicchiamo su device -> drag and drop e disabilitiamo l'impostazione

Adesso per procedere con l'analisi avviamo IDApro e via drag n drop iniziamo l'analisi del file richiesto.

Quanti parametri sono passati alla funzione Main()?

- i parametri passati alla funzione main sono 3 argc, argv e envp

```
main(int argc, const char **argv, const char **envp)
```

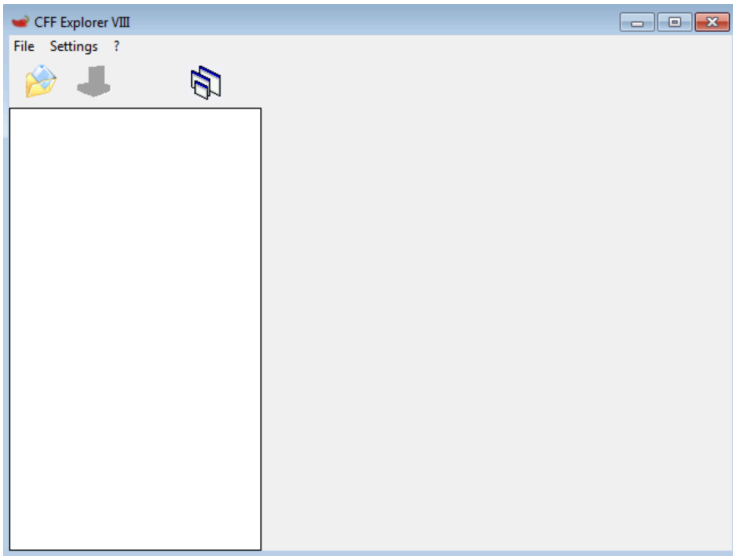
Quante variabili sono dichiarate all'interno della funzione Main()?

- Le variabili dichiarate all'interno della funzione Main sono 5 e sono tutte quelle con offset negativo : hModule- Data - var_117

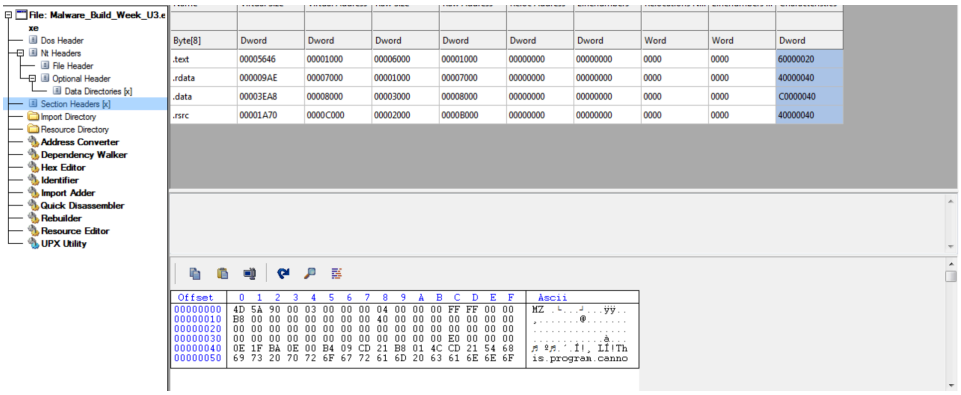
```
hModule= dword ptr -11Ch
Data= byte ptr -118h
var_117= byte ptr -117h
var_8= dword ptr -8
var_4= dword ptr -4
argc= dword ptr  8
argv= dword ptr  0Ch
envp= dword ptr  10h
```

Quali sezioni sono presenti all'interno del file eseguibile? Descrivete brevemente almeno 2 di quelle identificate

- Per verificare quali sezioni sono presenti all'interno dell'eseguibile utilizzerò CFF Explorer. Dunque una volta aperto il programma basterà cliccare sull'icona della cartella e selezionare exe in analisi



adesso bisognerà cliccare in section headers



le sezioni sono 4 e sono quelle già viste durante le lezioni
:
.text che contiene le stringhe di codice che la cpu eseguirà a exe avviato

.rdata che include le info circa le librerie e le funzioni importate ed esportate

.data contiene dati e variabili globali dell'eseguibile.

.rsrc tutte le risorse utilizzabili dall'eseguibile come icone immagini ecc..

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.

- Le librerie importate dal Malware sono due e sono **kernel32** e **advapi32**:

  La libreria **kernel32** è fondamentale per lo sviluppo di software su piattaforma Windows. Offre una vasta gamma di funzionalità di basso livello e servizi di sistema che sono essenziali per creare applicazioni.

Tra le sue principali funzionalità, **kernel32** consente agli sviluppatori di gestire i processi e i thread del sistema. Questo significa che è possibile creare nuovi processi, generare nuovi thread di esecuzione e gestire il loro ciclo di vita. Inoltre, offre un'ampia gamma di strumenti per manipolare file e directory, consentendo agli sviluppatori di leggere, scrivere e manipolare dati su disco.

Un'altra importante area di funzionalità è la gestione della memoria. **kernel32** fornisce strumenti per allocare e gestire la memoria del sistema.

La libreria **advapi32** offre un'ampia gamma di funzionalità avanzate per la programmazione su windows. Tra le funzioni più usate troviamo la possibilità di gestire i servizi di windows , l'accesso ai servizi di sistema, controllo di accessi e permessi alle risorse di sistema, crittografia e decrittografia. Le funzioni che vengono utilizzate dal malware sono RegsetVaue e RegCreateKey , per modificare e creare chiavi di registro

Malware Analysis

scopo della funzione chiamata alla locazione di memoria 00401021

- questo codice prepara i parametri necessari.

```
.text:00401004                push    0               ; lpdwDisposition
.text:00401006                lea     eax, [ebp+hObject]
.text:00401009                push    eax             ; phkResult
.text:0040100A                push    0               ; lpSecurityAttributes
.text:0040100C                push    0F003Fh         ; samDesired
.text:00401011                push    0               ; dwOptions
.text:00401013                push    0               ; lpClass
.text:00401015                push    0               ; Reserved
.text:00401017                push    offset SubKey   ; "SOFTWARE\\Microsoft\\Windows NT\\CurrentVe"...
.text:0040101C                push    80000002h       ; hKey
```

e poi chiama la funzione per creare o modificare un registro di sistema con i parametri specificati

```
.text:00401021                call    ds:RegCreateKeyExA
```

è una funzione usata per creare o aprire un registro di sistema

Come vengono passati i parametri alla funzione alla locazione 00401021

- i parametri vengono aggiunti ad uno stack e passati alla funzione mediante l'istruzione call. I parametri verranno letti dalla funzione in ordine inverso a come sono stati inseriti

Che oggetto rappresenta il parametro alla locazione 00401017

- alla locazione indicata è presente una subkey

```
; char SubKey[]
SubKey          db 'SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon',0
                                        ; DATA XREF: sub_401000+17↑o
```

  il processo winlogon è estremamente importante in windows perchè si occupa dell'autenticazione degli utenti durante il processo di accesso al sistema

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

- Questa operazione esegue AND tra il registro eax e se stesso, se eax è 0 lo ZF viene impostato a 1 se invece non è zero il flag viene impostato a 0.
  se il ZF è impostato a 1 viene effettuato un jump all'indirizzo 00401032 sennò il flusso procede all'istruzione successiva

```
.text:00401027                  test    eax, eax
.text:00401029                  jz      short loc_401032
```

  sembrerebbe essere un controllo per verificare se l'operazione precedente è stata effettuata correttamente.

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

- potremmo convertire in modo banale il codice in C scrivendo analogamente :

```
if (eax == 0 ){
        goto loc_401032
}
```

  per semplificare ho voluto utilizzare le medesime variabili presenti nel codice disassemblato

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

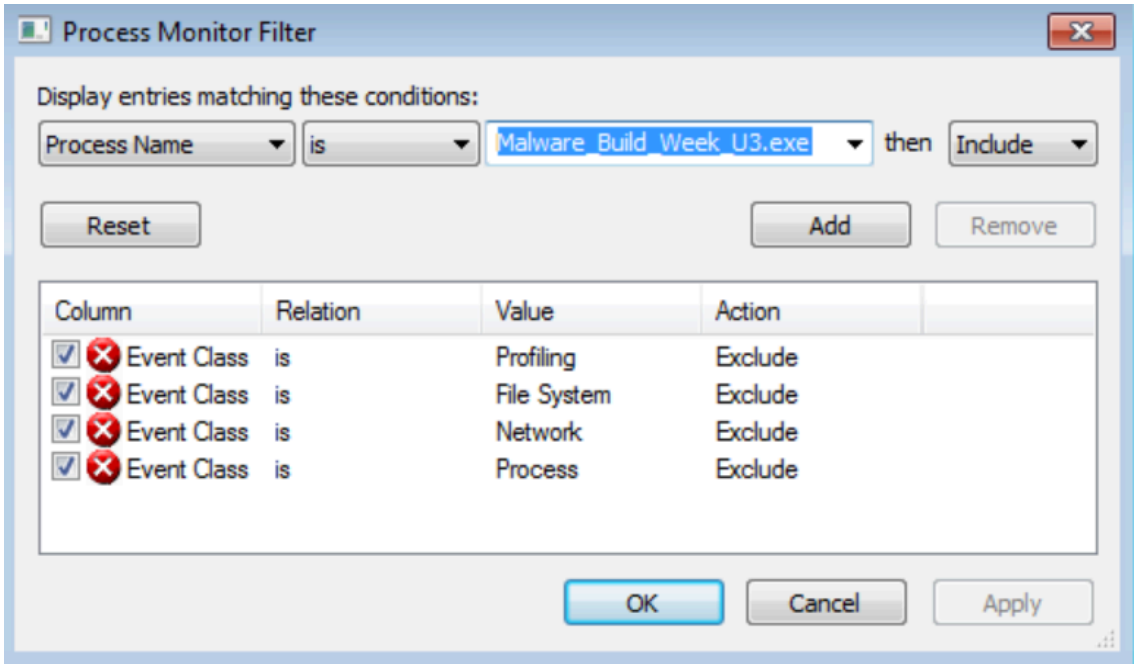
- 

  Il valore del parametro ValueName è l'indirizzo GinaDLL

Analisi Dinamica

adesso studiamo il comportamento del malware semplicemente avviando , ma prima di fare ciò avviamo process monitor
Dopo aver avviato il programma , per facilitare la lettura chiediamo al programma di mostrarci solamente le attività che vengono svolte sui registri di sistema , visto che, da una analisi preventiva abbiamo visto che uno dei compiti del malware è quello di far variare alcuni registri di sistema

Aggiungiamo anche il processo in analisi e clicchiamo add

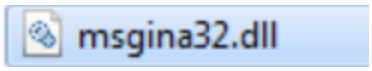


Time ...	Process Name	PID	Operation	Path	Result	Detail
19:34:...	Malware_Build_...	2644	RegOpenKey	HKLM\System\CurrentControlSet\Control\SafeBoot\Option	NAME NOT FOUND	Desired Access: Q...
19:34:...	Malware_Build_...	2644	RegOpenKey	HKLM\System\CurrentControlSet\Control\Srp\GP\DLL	REPARSE	Desired Access: R...
19:34:...	Malware_Build_...	2644	RegOpenKey	HKLM\System\CurrentControlSet\Control\Srp\GP\DLL	NAME NOT FOUND	Desired Access: R...
19:34:...	Malware_Build_...	2644	RegOpenKey	HKLM\Software\Wow6432Node\Policies\Microsoft\Windows\Safer\CodeIdentifiers	REPARSE	Desired Access: Q...
19:34:...	Malware_Build_...	2644	RegOpenKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\Safer\CodeIdentifiers	SUCCESS	Desired Access: Q...
19:34:...	Malware_Build_...	2644	RegSetInfoKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers	SUCCESS	KeySetInformation...
19:34:...	Malware_Build_...	2644	RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers\TransparentEnabled	NAME NOT FOUND	Length: 80
19:34:...	C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3\Malware_Build_Week_U3.exe			Windows\safer\codeidentifiers	SUCCESS	
19:34:...	Malware_Build_...	2644	RegOpenKey	HKCU\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers	NAME NOT FOUND	Desired Access: Q...
19:34:...	Malware_Build_...	2644	RegOpenKey	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions	REPARSE	Desired Access: R...
19:34:...	Malware_Build_...	2644	RegOpenKey	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions	SUCCESS	Desired Access: R...
19:34:...	Malware_Build_...	2644	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions	SUCCESS	KeySetInformation...
19:34:...	Malware_Build_...	2644	RegQueryValue	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions\(Default)	SUCCESS	Type: REG_SZ, Le...
19:34:...	Malware_Build_...	2644	RegOpenKey	HKLM\System\CurrentControlSet\Control\Terminal Server	REPARSE	Desired Access: R...
19:34:...	Malware_Build_...	2644	RegOpenKey	HKLM\System\CurrentControlSet\Control\Terminal Server	SUCCESS	Desired Access: R...
19:34:...	Malware_Build_...	2644	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Terminal Server	SUCCESS	KeySetInformation...
19:34:...	Malware_Build_...	2644	RegQueryValue	HKLM\System\CurrentControlSet\Control\Terminal Server\TSAppCompat	NAME NOT FOUND	Length: 548
19:34:...	Malware_Build_...	2644	RegQueryValue	HKLM\System\CurrentControlSet\Control\Terminal Server\TSUserEnabled	SUCCESS	Type: REG_DWO...
19:34:...	Malware_Build_...	2644	RegCloseKey	HKLM\System\CurrentControlSet\Control\Terminal Server	SUCCESS	
19:34:...	Malware_Build_...	2644	RegOpenKey	HKLM	SUCCESS	Desired Access: M...
19:34:...	Malware_Build_...	2644	RegQueryKey	HKLM	SUCCESS	Query: HandleTag...
19:34:...	Malware_Build_...	2644	RegOpenKey	HKLM\Software\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Diagnostics	NAME NOT FOUND	Desired Access: R...
19:34:...	Malware_Build_...	2644	RegQueryKey	HKLM	SUCCESS	Query: HandleTag...
19:34:...	Malware_Build_...	2644	RegCreateKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon	SUCCESS	Desired Access: All...
19:34:...	Malware_Build_...	2644	RegSetInfoKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon	SUCCESS	KeySetInformation...
19:34:...	Malware_Build_...	2644	RegQueryKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon	SUCCESS	Query: HandleTag...
19:34:...	Malware_Build_...	2644	RegSetValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon\GinaDLL	ACCESS DENIED	Type: REG_SZ, Le...
19:34:...	Malware_Build_...	2644	RegCloseKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon	SUCCESS	
19:34:...	Malware_Build_...	2644	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options	SUCCESS	
19:34:...	Malware_Build_...	2644	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options	SUCCESS	
19:34:...	Malware_Build_...	2644	RegCloseKey	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions	SUCCESS	
19:34:...	Malware_Build_...	2644	RegCloseKey	HKLM	SUCCESS	

A quanto pare il malware non riesce a effettuare le modifiche necessarie a causa della mancanza di permessi di amministratore , dunque al fine di svolgere l'esercizio avvierò file exe come amministratore

19:40:...	Malware_Build_...	2240	RegOpenKey	HKLM\System\CurrentControlSet\Control\SafeBoot\Option	NAME NOT FOUND	Desired Access: Query Value, Set Value
19:40:...	Malware_Build_...	2240	RegOpenKey	HKLM\System\CurrentControlSet\Control\Srp\GP\DLL	REPARSE	Desired Access: Read
19:40:...	Malware_Build_...	2240	RegOpenKey	HKLM\System\CurrentControlSet\Control\Srp\GP\DLL	NAME NOT FOUND	Desired Access: Read
19:40:...	Malware_Build_...	2240	RegOpenKey	HKLM\Software\Wow6432Node\Policies\Microsoft\Windows\Safer\CodeIdentifiers	REPARSE	Desired Access: Query Value
19:40:...	Malware_Build_...	2240	RegOpenKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\Safer\CodeIdentifiers	SUCCESS	Desired Access: Query Value
19:40:...	Malware_Build_...	2240	RegSetInfoKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers	SUCCESS	KeySetInfomationClass: KeySetHandleTagsInformation, Length: 0
19:40:...	Malware_Build_...	2240	RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers\TransparentEnabled	NAME NOT FOUND	Length: 80
19:40:...	Malware_Build_...	2240	RegCloseKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers	SUCCESS	
19:40:...	Malware_Build_...	2240	RegOpenKey	HKCU\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers	NAME NOT FOUND	Desired Access: Query Value
19:40:...	Malware_Build_...	2240	RegOpenKey	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions	REPARSE	Desired Access: Read
19:40:...	Malware_Build_...	2240	RegOpenKey	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions	SUCCESS	Desired Access: Read
19:40:...	Malware_Build_...	2240	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions	SUCCESS	KeySetInfomationClass: KeySetHandleTagsInformation, Length: 0
19:40:...	Malware_Build_...	2240	RegQueryValue	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions\(Default)	SUCCESS	Type: REG_SZ, Length: 36, Data: 00060101.00060101
19:40:...	Malware_Build_...	2240	RegOpenKey	HKLM\System\CurrentControlSet\Control\Terminal Server	REPARSE	Desired Access: Read
19:40:...	Malware_Build_...	2240	RegOpenKey	HKLM\System\CurrentControlSet\Control\Terminal Server	SUCCESS	Desired Access: Read
19:40:...	Malware_Build_...	2240	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Terminal Server	SUCCESS	KeySetInfomationClass: KeySetHandleTagsInformation, Length: 0
19:40:...	Malware_Build_...	2240	RegQueryValue	HKLM\System\CurrentControlSet\Control\Terminal Server\TSAppCompat	NAME NOT FOUND	Length: 548
19:40:...	Malware_Build_...	2240	RegQueryValue	HKLM\System\CurrentControlSet\Control\Terminal Server\TSUserEnabled	SUCCESS	Type: REG_DWORD, Length: 4, Data: 0
19:40:...	Malware_Build_...	2240	RegCloseKey	HKLM\System\CurrentControlSet\Control\Terminal Server	SUCCESS	
19:40:...	Malware_Build_...	2240	RegOpenKey	HKLM	SUCCESS	Desired Access: Maximum Allowed, Granted Access: All Access
19:40:...	Malware_Build_...	2240	RegQueryKey	HKLM	SUCCESS	Query: HandleTags, HandleTags: 0x0
19:40:...	Malware_Build_...	2240	RegOpenKey	HKLM\Software\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Diagnostics	NAME NOT FOUND	Desired Access: Read
19:40:...	Malware_Build_...	2240	RegQueryKey	HKLM	SUCCESS	Query: HandleTags, HandleTags: 0x0
19:40:...	Malware_Build_...	2240	RegCreateKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon	SUCCESS	Desired Access: All Access, Disposition: REG_OPENED_EXISTING_KEY
19:40:...	Malware_Build_...	2240	RegSetInfoKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon	SUCCESS	KeySetInfomationClass: KeySetHandleTagsInformation, Length: 0
19:40:...	Malware_Build_...	2240	RegQueryKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon	SUCCESS	Query: HandleTags, HandleTags: 0x400
19:40:...	Malware_Build_...	2240	RegSetValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon\GinaDLL	SUCCESS	Type: REG_SZ, Length: 520, Data: C:\Users\user\Desktop\MALWAR...
19:40:...	Malware_Build_...	2240	RegCloseKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon	SUCCESS	
19:40:...	Malware_Build_...	2240	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options	SUCCESS	
19:40:...	Malware_Build_...	2240	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options	SUCCESS	
19:40:...	Malware_Build_...	2240	RegCloseKey	HKLM\System\CurrentControlSet\Control\Nls\Sorting\Versions	SUCCESS	
19:40:...	Malware_Build_...	2240	RegCloseKey	HKLM	SUCCESS	

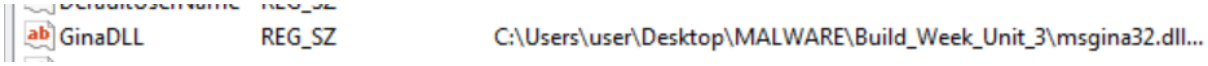
Com'è possibile notare dopo l'avvio del malware viene generata la dll msgina32 visibile nella cartella dove è stato eseguito il malware.



questa è stata creata utilizzando una funzione presente nella libreria KERNEL32, chiamata CreateFile


```
rn SetFilePointer:dword ; CODE XREF: __lseek+4F↑p
                        ; DATA XREF: __lseek+4F↑r
CreateFileA(LPCSTR lpFileName, DWORD dwDesiredAccess, DWORD dwShareMode, LPSECURITY_ATTRIBUTES lpSecurityAttributes,
rn CreateFileA:dword ; CODE XREF: __sopen+1AC↑p
                        ; DATA XREF: __sopen+1AC↑r
tCPInfo(UINT CodePage, LPCPINFO lpCPInfo)
```

Viene creato un nuovo registro di sistema,



il valore associato a questa chiave è quello presente nello screenshot questa operazione è stata effettuata utilizzando RegSetValue, funzione presente nella libreria ADVAPI32

RegSetValueExA	ADVAPI32
RegCreateKeyExA	ADVAPI32

Conclusioni
Il processo winlogin chiama la dll MSGina.dll (un componente windows che tra le varie funzioni fornisce un'interfaccia grafica per l'autenticazione. da qui il suo nome "Graphical Identification and Authentication").

Leggendo la documentazione di microsoft (https://learn.microsoft.com/en-us/windows/win32/secauthn/loading-and-running-a-gina-dll) è evidente come se si vuole usare un GINA differente è necessario posizionare la DLL ad uno specifico percorso, questo è ciò che effettua il malware.

Viene generato un nuovo registro di sistema al path indicato nella documentazione

```
\HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon
```

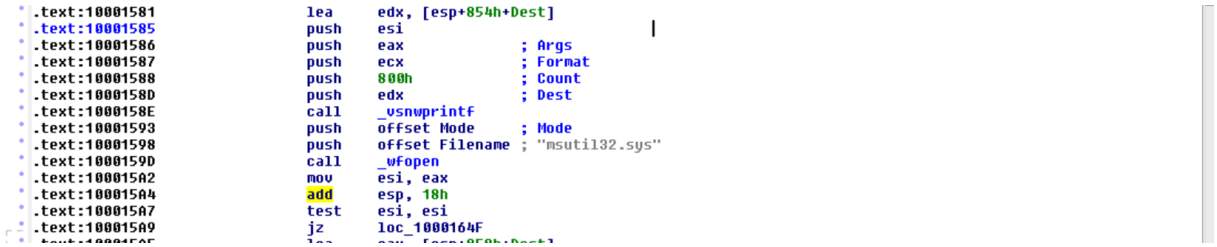
 che ha come chiave il percorso che punta alla dll modificata generata dal malware stesso e presente all'interno della cartella dell'eseguibile.

Così facendo il malware ottiene la persistenza all'interno del sistema infatti ad ogni avvio winlogon utilizzerà non più la DLL standard ma quella modificata.

Analizziamo con Idapro la dll modificata per comprendere qual'è lo scopo ultimo del malware. Tra le librerie importate, oltre alle già viste kernel32 e ADVAPI32, sono presenti anche la MSVCRT e la USER32

La libreria MSVCRT è responsabile della gestione delle operazioni runtime come la gestione della memoria e le operazioni di input output, mentre la USER32 fornisce l'interfaccia utente di base per i programmi di windows , viene però utilizzata la funzione wsprintf utilizzata per scrivere stringhe di testo in un buffer

All’inizio della funzione WlxLoggedOutSAS verrà aperto il file msutil32.sys



Dopo la chiamata della funzione su dichiarata le credenziali inserite verranno inserite all’interno del file aperto precedentemente, che poi verrà salvato.