Progetto Modulo 6 - Alessio Russo

Malware Analysis

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.

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»?

Analisi dinamica: Eseguite il Malware

- 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
- Analizzare i risultati con procmon

Malware Analysis

Filtrate includendo solamente l'attività sul registro di Windows.

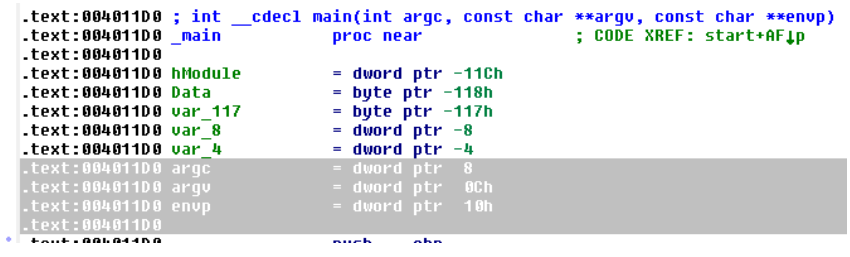
- Quale chiave di registro viene creata?
- Quale valore viene associato alla chiave di registro creata?

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.

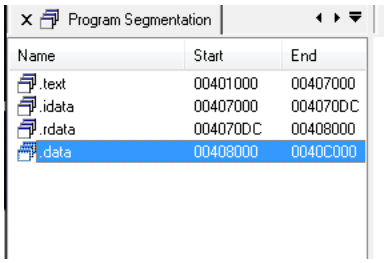
- Parametri: notiamo in modo immediato che i parametri passati alla funzione main() sono: argc, argv, anvp questi possono essere notati sia alla definizione di funzione mai in blu che rispetto alle righe di codice evidenziate in grigio, in generale per convenzione IDA Pro identifica i parametri quelli con i valori positivi come da evidenze.



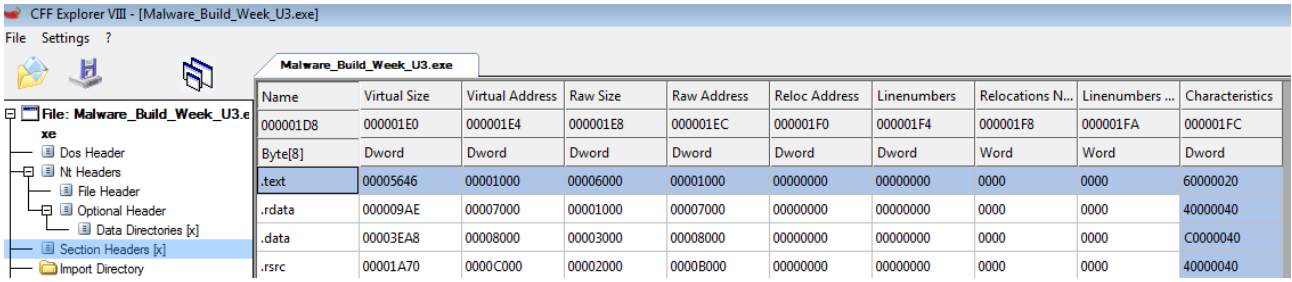
- Variabili: per quanto riguarda le variabili si nota immediatamente che queste sono 5, in genere vengono identificate da quei caratteri che hanno valore negativo come da evidenziazione in grigio

```
.text:004011D0 ; int __cdecl main(int argc, const char **argv, const char **envp)
.text:004011D0 _main           proc near               ; CODE XREF: start+AF↓p
.text:004011D0
.text:004011D0 hModule         = dword ptr -11Ch
.text:004011D0 Data            = byte ptr -118h
.text:004011D0 var_117         = byte ptr -117h
.text:004011D0 var_8           = dword ptr -8
.text:004011D0 var_4           = dword ptr -4
.text:004011D0 argc            = dword ptr  8
.text:004011D0 argv            = dword ptr  0Ch
.text:004011D0 envp            = dword ptr  10h
.text:004011D0
```

- Le sezioni utilizzate dal malware possiamo verificarle sia con IDA pro che con CFF Explorer:
    - IDA Pro: effettuando la visualizzazione sulla segmentation possiamo notare che le sezioni utilizzate dal malware sono:
        - .text: contiene le istruzioni (le righe di codice) che la CPU eseguirà una volta che il software sarà avviato
        - .idata: contiene le informazioni sull'importazione di funzioni da altre librerie dinamiche (DLL)
        - .rdata: include generalmente le informazioni circa le librerie e le funzioni importate ed esportate dall'eseguibile
        - .data: contiene tipicamente i dati / le variabili globali del programma eseguibile, che devono essere disponibili da qualsiasi parte del programma

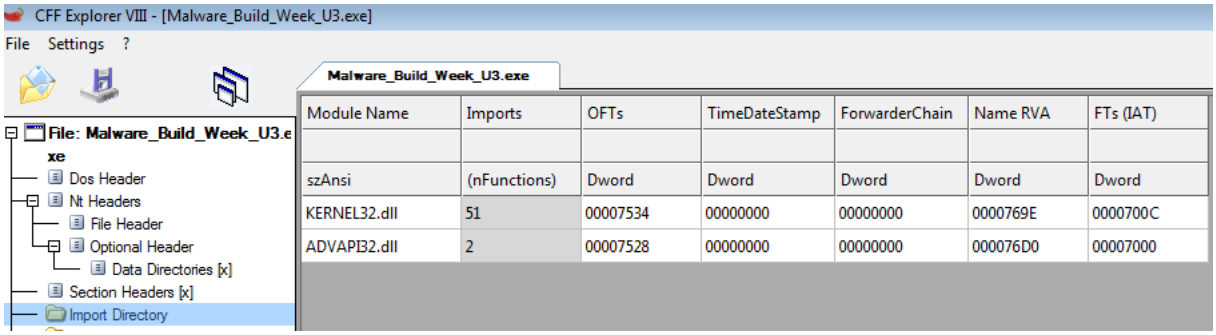


    - Stessa cosa viene evidenziata attraverso CFF Explorer, il quale a differenza di IDA Pro mette in evidenza l'utilizzo di. rsrc: include le risorse utilizzate dall'eseguibile come ad esempio icone, immagini, menu e stringhe

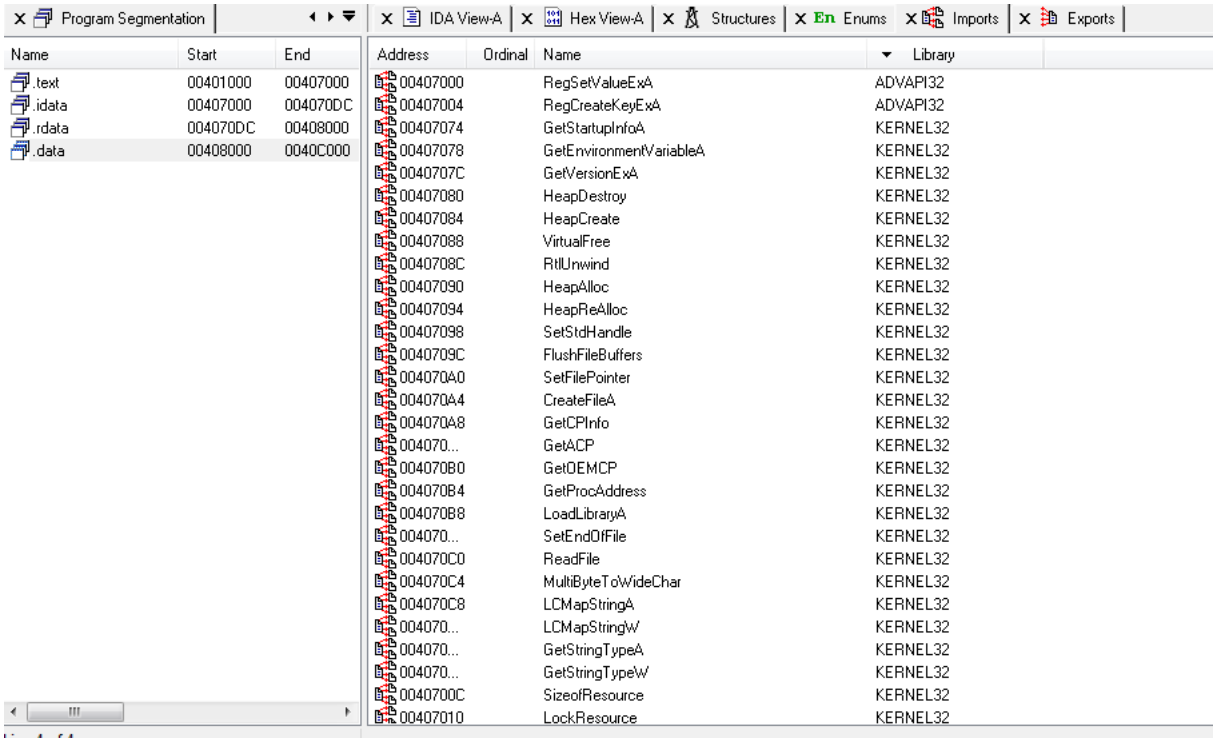


- Import Librerie: come sopra possiamo vedere il risultato sia attraverso IDA Pro che attraverso CFF Explorer: Entrambi i software mettono in evidenza l'utilizzo di due librerie quali:
    - advapi32.dll: fornisce una serie di funzioni per l'accesso ai servizi di sicurezza avanzati e per la gestione delle autorizzazioni di sistema
    - kernel32.dll: fornisce una serie di funzionalità di basso livello necessarie per il corretto funzionamento del sistema operativo e delle applicazioni

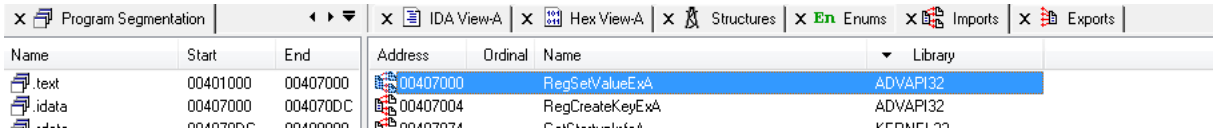
- Librerie importate con CFF Explorer



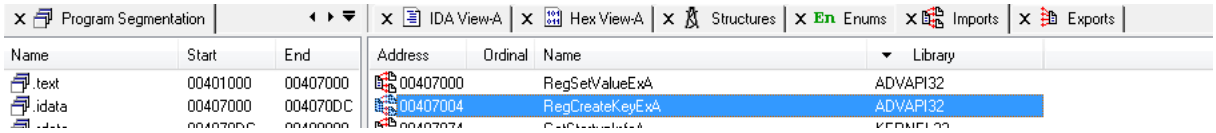
- Librerie importate con IDA Pro: Da queste possiamo notare anche che il malware in questione è un dropper rispetto alle operazioni quali:
    - CreateFileA: utilizzata per creare o aprire un file
    - SizeofResource: utilizzata per recuperare la dimensione, in byte, di una risorsa all'interno di un file eseguibile o di una DLL
    - LockResources: utilizzata per ottenere un puntatore al contenuto di una risorsa all'interno di un file eseguibile o di una DLL



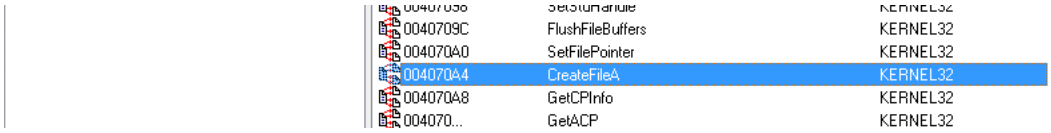
Possiamo notare l'utilizzo di alcuni registri specifici come RegSetValueExA, RegCreateKeyExA entrambi registri della libreria advapi32.dll
RegSetValueExA: viene utilizzata per impostare il valore di un dato all'interno del registro di sistema - advapi32.dll



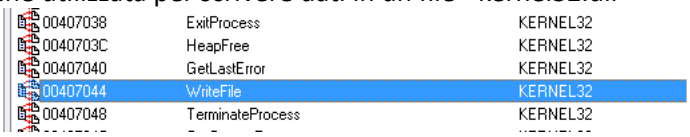
RegCreateKeyExA: funzione utilizzata per creare una nuova chiave o aprire una chiave esistente del registro di sistema di Windows - advapi32.dll



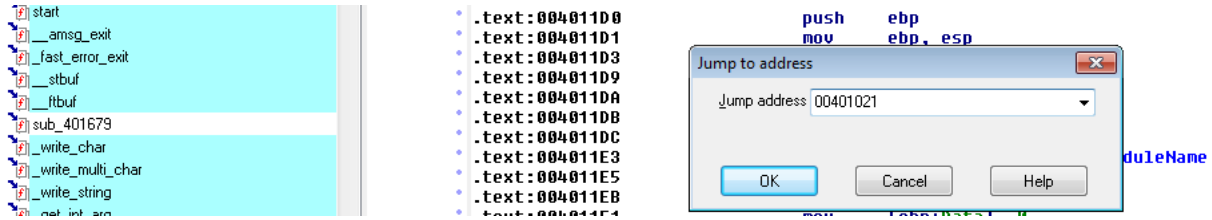
Per quanto rigurarda la libreria kernel32.dll è immediato notare le operazioni quali CreateFileA, WriteFile.
CreateFileA: utilizzata per creare o aprire file - kernel32.dll

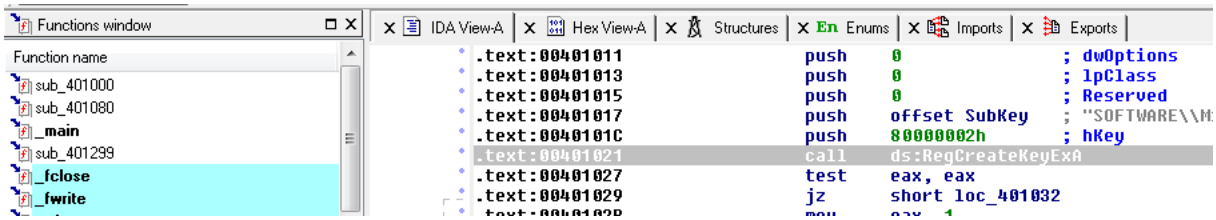


WriteFile: Questa funzione viene utilizzata per scrivere dati in un file - kernel32.dll



- Lo scopo della funzione chiamata alla locazione di memoria 00401021:

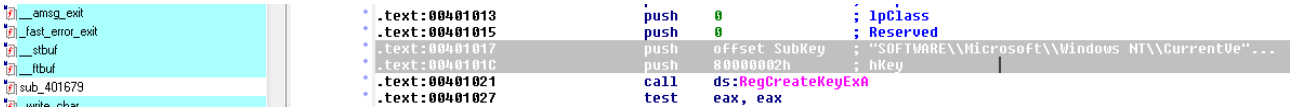




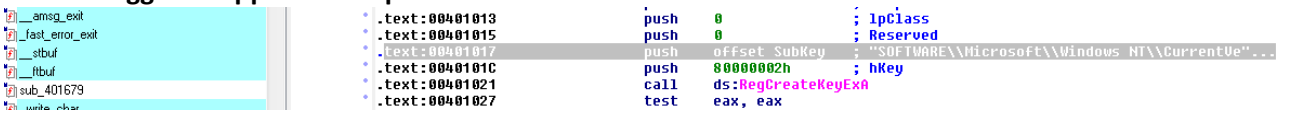
La funzione RegCreateKey è una funzione utilizzata per creare una nuova chiave o aprire una chiave esistente del registro di sistema di Windows. Questa funzione viene comunemente utilizzata per interagire con il registro di sistema al fine di memorizzare e recuperare informazioni di configurazione, impostazioni dell'applicazione, per l'installazione di software

- **Come vengono passati i parametri alla funzione alla locazione 00401021**
  I parametri vengono passati tramite le push evidenziate: offset Subkey e 80000002h (hKey):
  - hKey: handle della chiave di registro principale con cui si desidera interagire, ricordando che un handle è un oggetto che punta a processi, file o risorse che sono state aperte o create all' interno del sistema operativo

  Notiamo anche che in base al percorso del file il malware ha acquistato la persistenza all' interno del sistema operativo. La persistenza consiste nell'aggiungere il malware alle voci di avvio del sistema in modo che venga eseguito automaticamente ogni volta che il sistema operativo viene avviato.



- **Che oggetto rappresenta il parametro alla locazione 00401017**



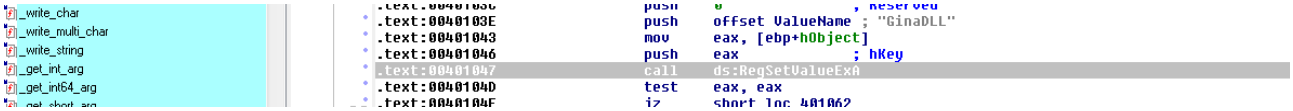
Il parametro alla locazione **00401017** rappresenta l'offset dell'etichetta SubKey.
SubKey rappresenta il percorso relativo o assoluto della chiave del registro che il malware ha creato. Questo percorso specifica la posizione della chiave all'interno del registro di sistema, quindi il valore passato come parametro alla locazione 00401017 è l'indirizzo di memoria in cui è memorizzato il percorso della chiave del registro che si desidera creare o aprire.

Queste istruzioni controllano se la chiamata alla funzione **RegCreateKeyExA** ha avuto successo o meno, controllando se il valore restituito è zero. Se la chiamata ha avuto successo il programma continua normalmente. Se la chiamata ha fallito (valore restituito zero) il programma salta a loc_401032 per gestire l'errore.

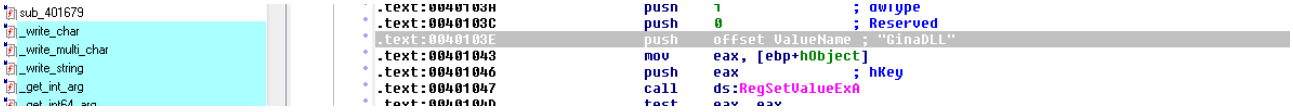
- **Con riferimento all'ultimo quesito, tradurre il codice Assembly nel corrispondente costrutto C.**

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

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



- ValueName: 0040103E = GinaDLL

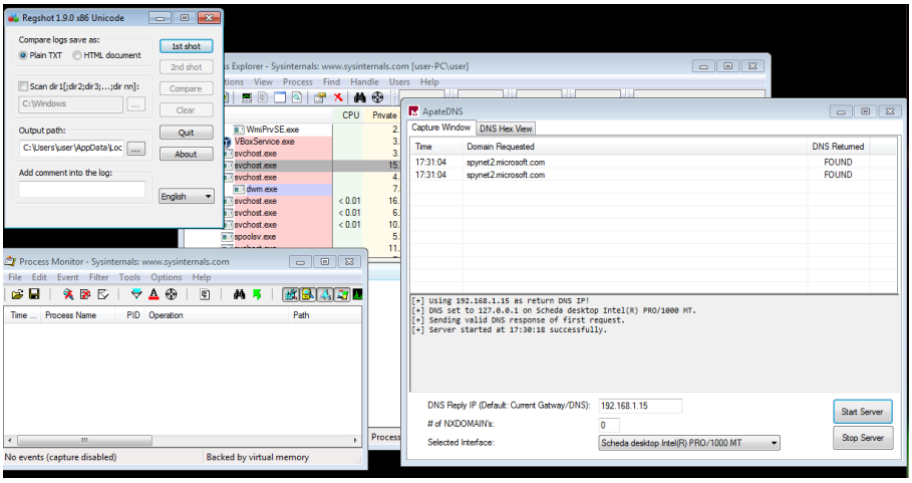


- **RegSetValueEx**: questa funzione permette invece di aggiungere un nuovo valore all'interno del registro e di settare i rispettivi dati: in questo caso possiamo notare dalle push precedenti che aggiungerà **ValueName; GinaDLL**

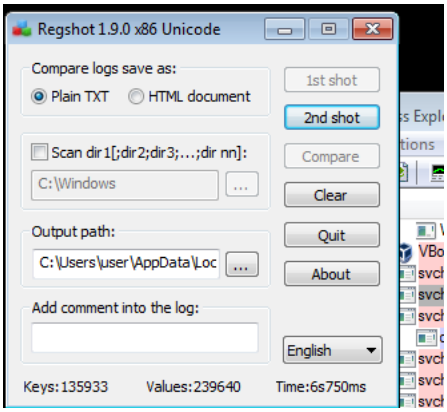
ANALISI DINAMICA

Un potenziale approccio da adottare potrebbe essere:

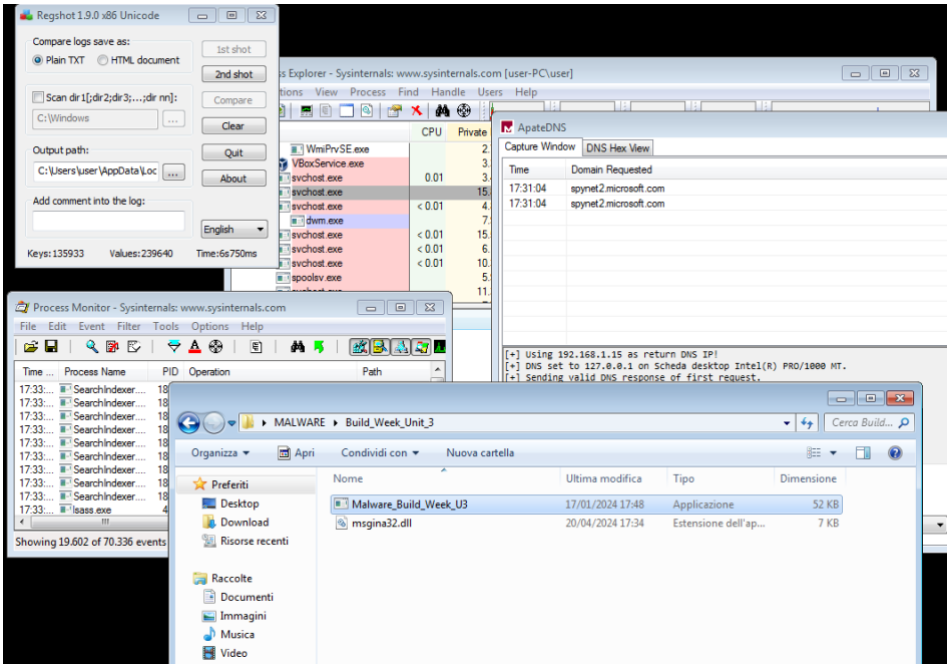
- Process Explorer (ci consente di capire quali sono i processi che la macchina sta utilizzando)
- ApateDNS (Serve per capire se qualcosa viene richiamato sulla parte DNS)
- RegShot (Serve a capire la situazione prima e post avvio malware)
- Procmon

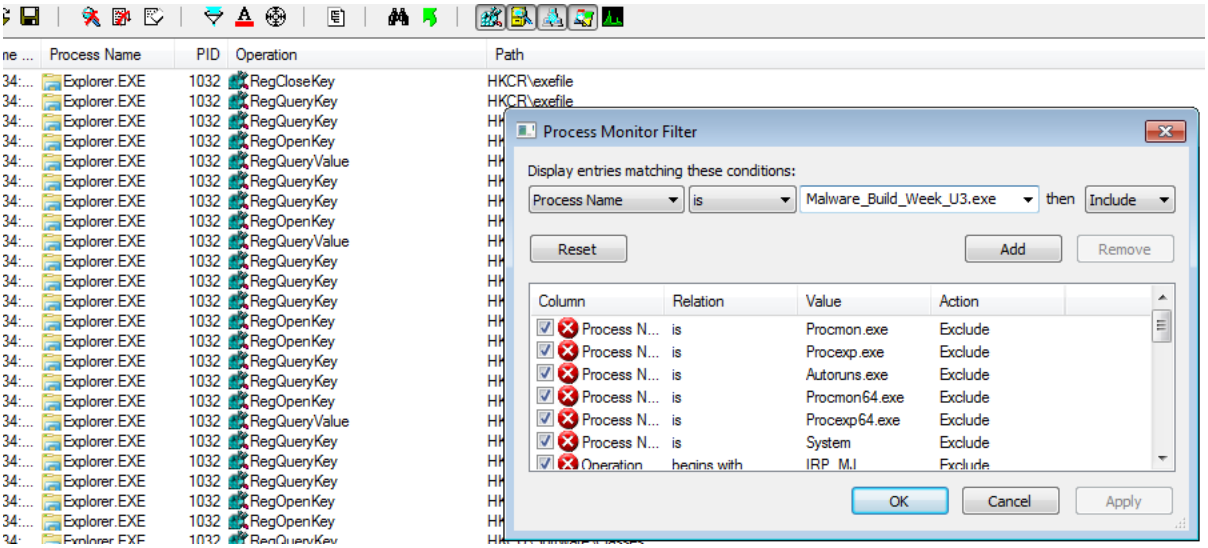


- **Regshot**: questo tool ci permette di effettuare una istantanea delle chiavi di registro per consentirci di visualizzare le chiavi di registro prima e dopo l'esecuzione del malware in modo da notare le modifiche effettuate da quest'ultimo

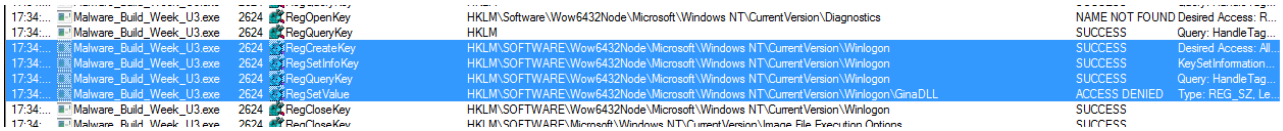


- Dopo aver effettuato un primo screen con Regshot procedo all' esecuzione del malware, prima avvio la cattura con procmon il quale ci consente monitorare in tempo reale l'attività del sistema operativo Windows, inclusi i processi in esecuzione, le operazioni sui file e le attività del registro di sistema



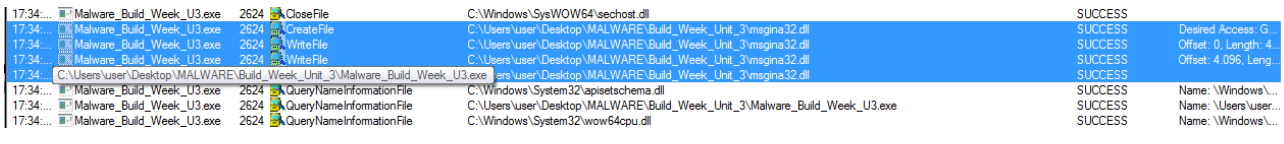


- Effettuata la cattura, dopo aver stoppato la registrazione delle attività, filtro il processo di cui ho interesse verificare le operazioni eseguite
- Filtrando per nome processo posso successivamente applicare altri filtri su operazione che ha effettuato sui registri:
  in questo caso ricade l'attenzione sulle voci selezionate in blu in basso, nello specifico per le operazioni di:
  - RegCreateKey: utilizzata per creare una nuova sottochiave o aprire una sottochiave esistente
  - RegSetInfoKey: utilizzata per impostare le informazioni di configurazione per una chiave specifica nel registro di sistema
  - RegQueryKey: utilizzata per recuperare informazioni sulle chiavi di registro, come il numero di sottochiavi e il numero di valori associati a una chiave specificata
  - RegSetValue: utilizzata per impostare il valore di una chiave del registro di sistema dove vediamo il valore del registro creato GinaDLL

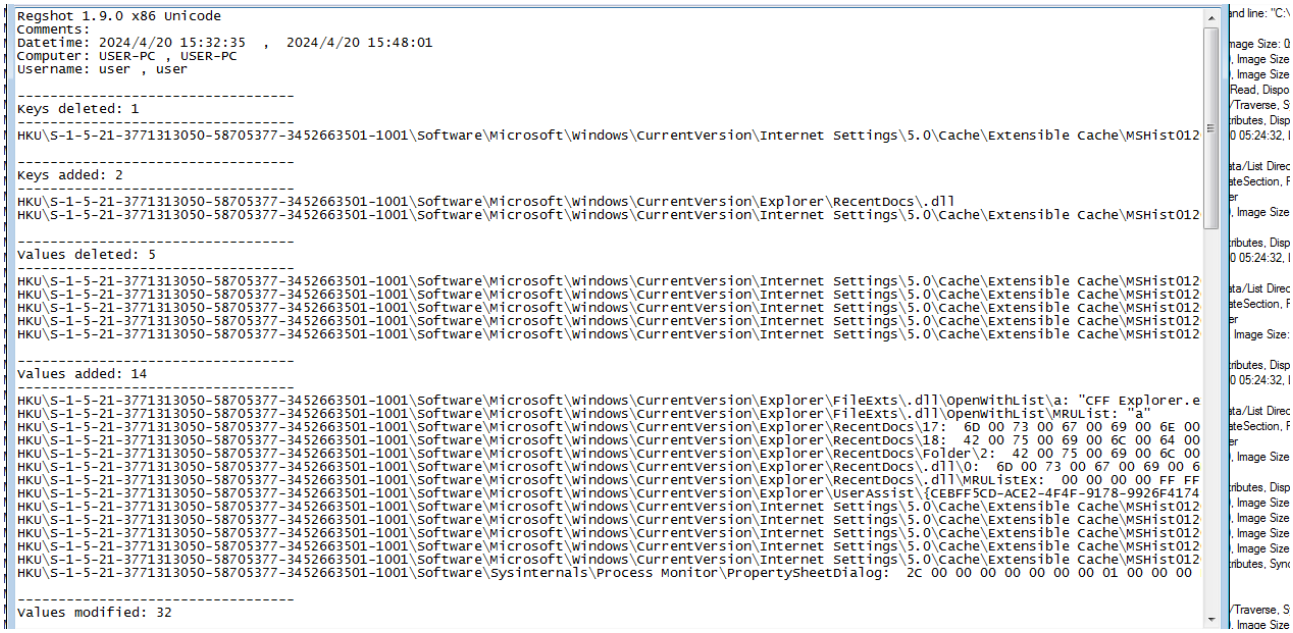


- Possiamo ancora effettuare un filtro sulle attività effettuate su FileSystem: in questo caso ricade l'attenzione sulle voci selezionate in blu in basso, nello specifico per le operazioni:
  - CreateFile: utilizzata per creare o aprire un file o un dispositivo
  - WhriteFile: utilizzata per scrivere dati in un file
  - CloseFile(coperto): chiudere un handle aperto precedentemente tramite CreateFile o altre funzioni

  Il file creato sarà msgina32.dll



- Analisi delle modifiche sulle chiavi di registro con Regshot



```
Regshot 1.9.0 x86 Unicode
Comments:
Datetime: 2024/4/20 15:32:35  ,  2024/4/20 15:48:01
Computer: USER-PC , USER-PC
Username: user , user

----------------------------------
Keys deleted: 1
----------------------------------
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012

----------------------------------
Keys added: 2
----------------------------------
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.dll
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012

----------------------------------
Values deleted: 5
----------------------------------
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012

----------------------------------
Values added: 14
----------------------------------
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\.dll\OpenWithList\a: "CFF Explorer.e
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\.dll\OpenWithList\MRUList: "a"
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\17:  6D 00 73 00 67 00 69 00 6E 00
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\18:  42 00 75 00 69 00 6C 00 64 00
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\Folder\2:  42 00 75 00 69 00 6C 00
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.dll\0:  6D 00 73 00 67 00 69 00 6
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.dll\MRUListEx:  00 00 00 00 FF FF
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{CEBFF5CD-ACE2-4F4F-9178-9926F4174
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012
HKU\S-1-5-21-3771313050-58705377-3452663501-1001\Software\Sysinternals\Process Monitor\PropertySheetDialog:  2C 00 00 00 00 00 00 00 01 00 00 00

----------------------------------
Values modified: 32
```

Concludendo possiamo dire che il malware con le operazioni sui registri e nello specifico evidenziate in precedenza e rispetto al suo comportamento nell'analisi dinamica acquista la persistenza come evidenziato alla funzione 401021. Inoltre, analizzando l'import delle librerie possiamo dire che si tratta di un dropper ossia un malware che si esegue per estrae il malware contenuto al suo interno per salvarlo sul disco (msgina32.dll).