2023

Progetto M6

AUTORI IN ORDINE ALFABETICO

MARCO CERFOGLI
DAVIDE COPPOLA
SIMONA DI MAGGIO
MIKE LO PRESTI
SALVATORE PRINCIOTTA

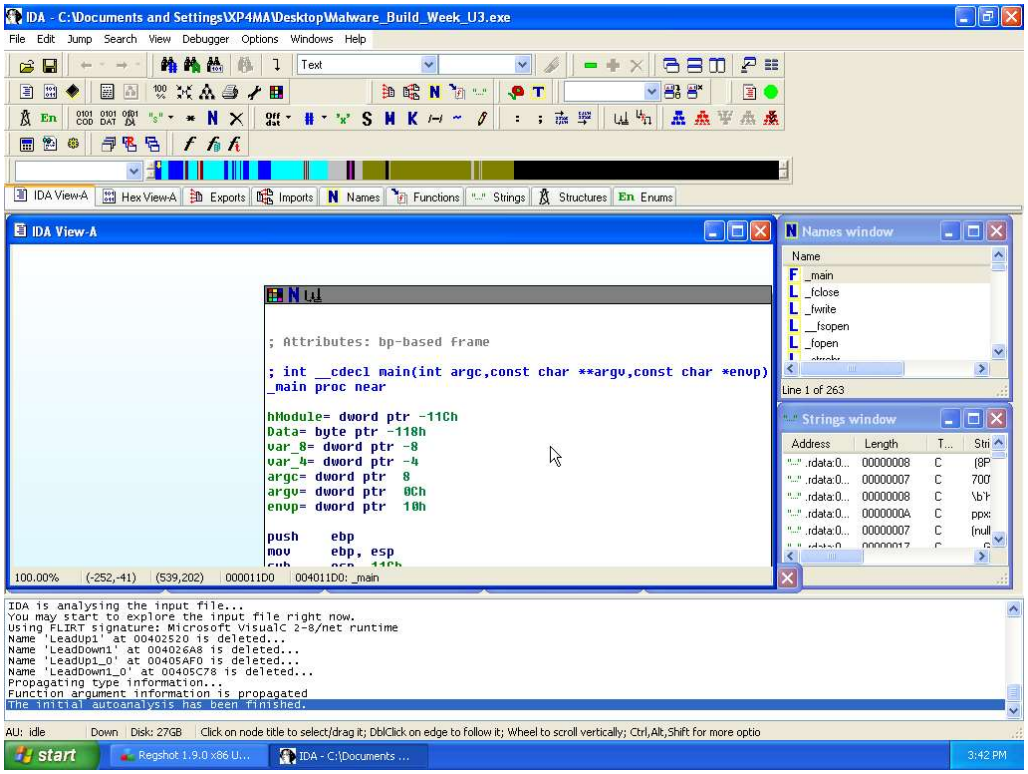
Malware Analysis

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.

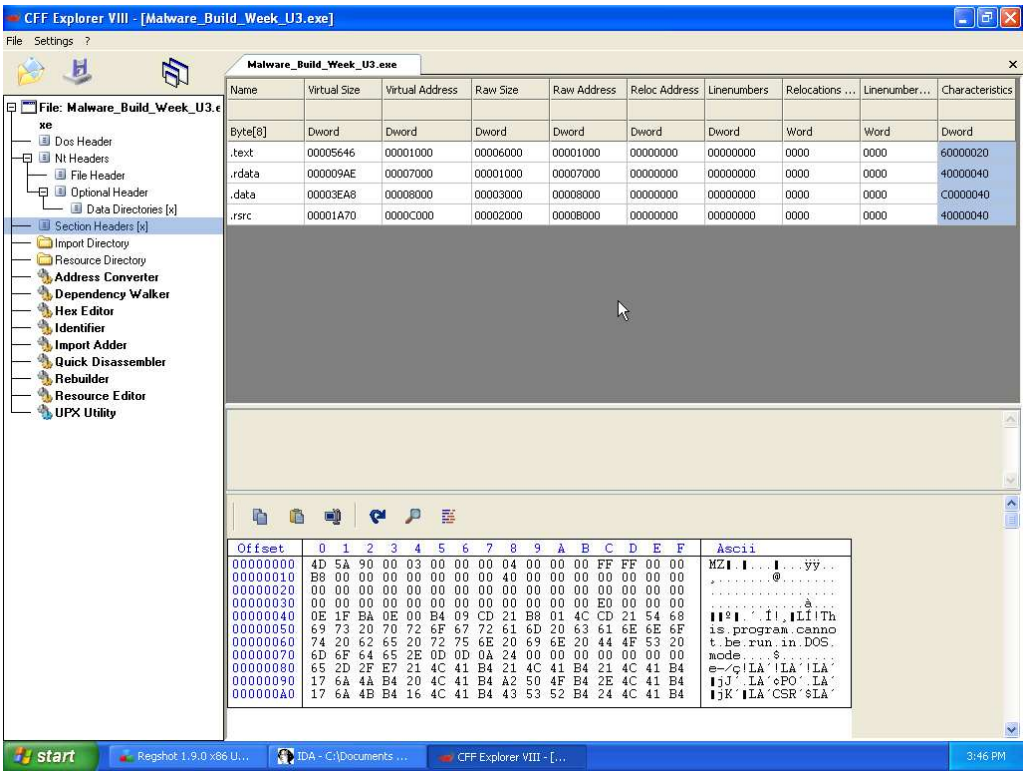
Si procede all'analisi del malware utilizzando due tool: RegShot ed Exeinfo PE.

- Quanti parametri sono passati alla funzione Main()?
- Quante variabili sono dichiarate all'interno della funzione Main()?



Da una prima analisi possiamo distinguere un parametro da una variabile. Nel primo caso si avrà un valore positivo mentre nel secondo uno negativo. Pertanto, i parametri assegnati alla funzione Main() sono**: argc, argv e envp.** Mentre le variabili dichiarate sono: **hmodule, data, var_8, var_4.**

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



La *sezione* di un file eseguibile portabile (PE) è un blocco di memoria contiguo denominato che contiene codice o dati dichiarati che verranno poi usati direttamente dal programma.

.RSRC → La sezione .rsrc (abbreviazione di *resource*) è una sezione presente nei file eseguibili di Windows, che contiene le risorse utilizzate da un programma. Le risorse possono includere immagini, icone, suoni, file di configurazione, stringhe di testo e altri tipi di dati utilizzati durante l'esecuzione dello stesso. La sezione .rsrc viene solitamente letta dal sistema operativo per rendere disponibili le risorse all'applicazione in fase di esecuzione. Inoltre può contenere altre informazioni importanti, tra cui la versione del software, informazioni sul copyright e metadati relativi al file eseguibile stesso. In sintesi, la sezione .rsrc è una parte dei file eseguibili che contiene le risorse utilizzate da un programma, inclusi file multimediali, stringhe di testo e altre informazioni necessarie per il corretto funzionamento dell'applicazione.

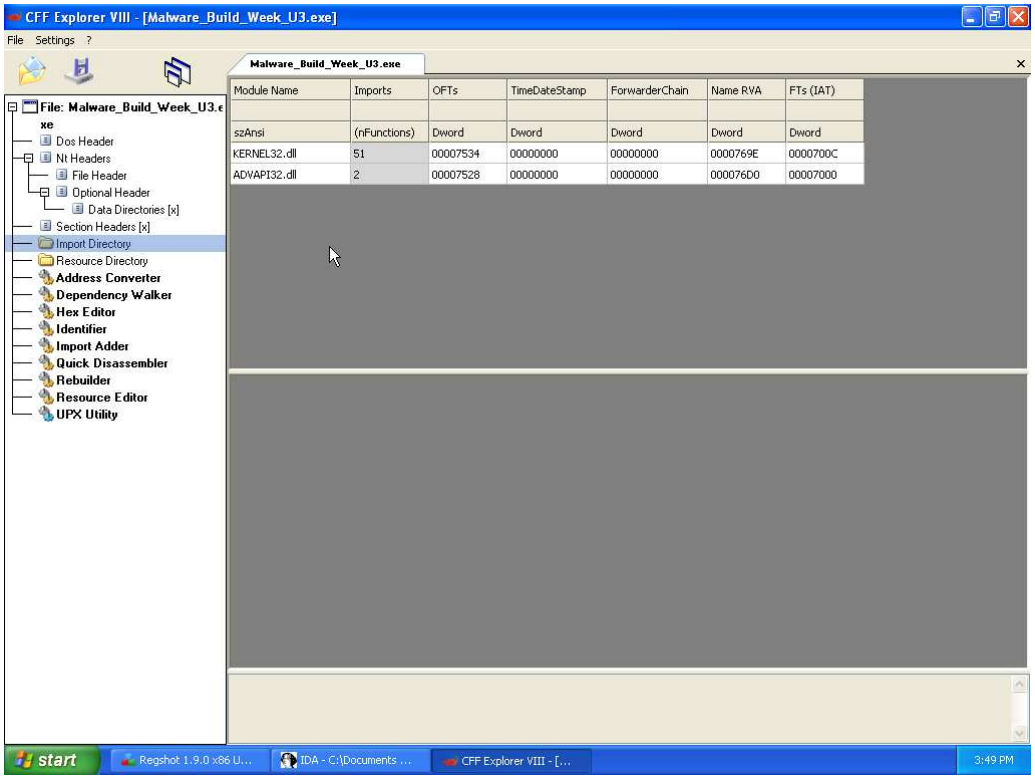
.RDATA → La sezione .rdata è una sezione di un programma o di un file eseguibile che contiene dati di sola lettura. Questi dati possono includere stringhe, costanti o tabelle di dati predefinite che vengono utilizzate dal programma durante l'esecuzione. La sezione .rdata può essere trovata principalmente nei file eseguibili dei programmi Windows.

.TEXT → è una sezione di un file eseguibile o di un oggetto compilato, come ad esempio un programma o una libreria. Questa sezione contiene il codice eseguibile del programma, detto anche codice macchina. Il codice presente nella sezione .text viene caricato in memoria quando le istruzioni del programma vengono eseguite dal processore. In breve, la sezione .text contiene il codice che determina il comportamento del programma durante l'esecuzione.

.DATA → presente nei programmi in linguaggio assembly o in linguaggi di programmazione basso livello, è una sezione di memoria riservata alla memorizzazione di dati statici o inizializzati. Questa sezione viene utilizzata per dichiarare variabili che mantengono il loro valore durante l'esecuzione del programma.

I dati presenti nella sezione .data sono generalmente allocati in modo statico e vengono assegnati dei valori predefiniti o inizializzati durante la compilazione. I dati in questa sezione possono essere letti o scritti durante l'esecuzione del programma.

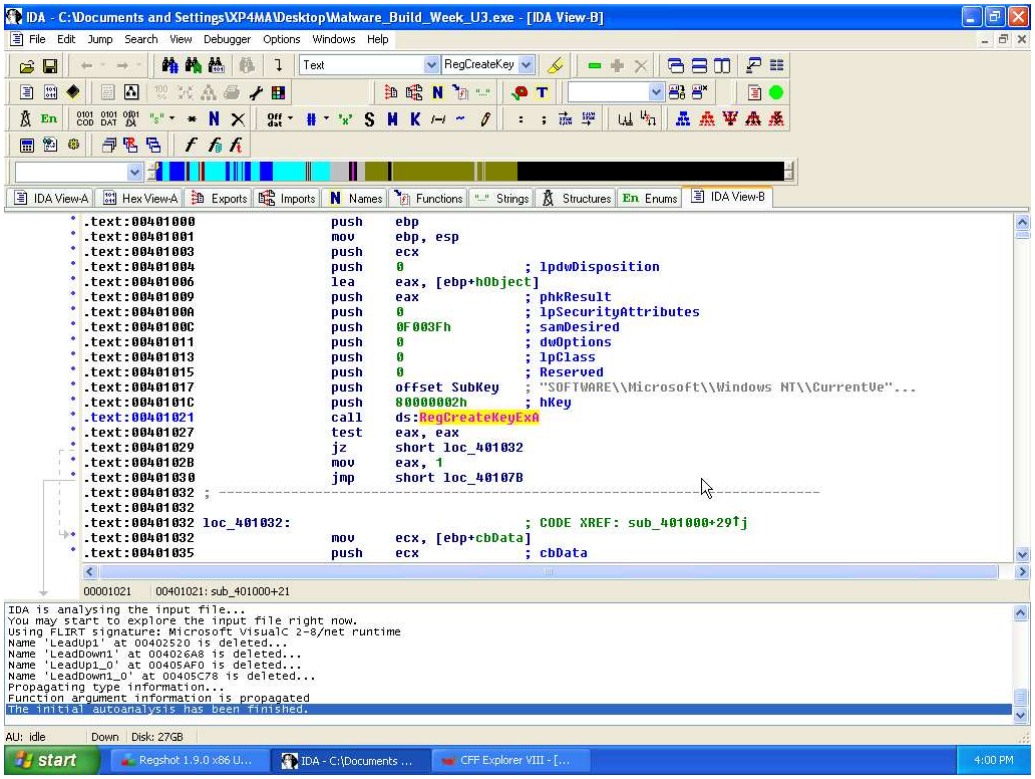
- Quali librerie importa il Malware? Per ognuna delle librerie importate, fate delle ipotesi sulla base della sola analisi statica delle funzionalità che il Malware potrebbe implementare. Utilizzate le funzioni che sono richiamate all'interno delle librerie per supportare le vostre ipotesi.



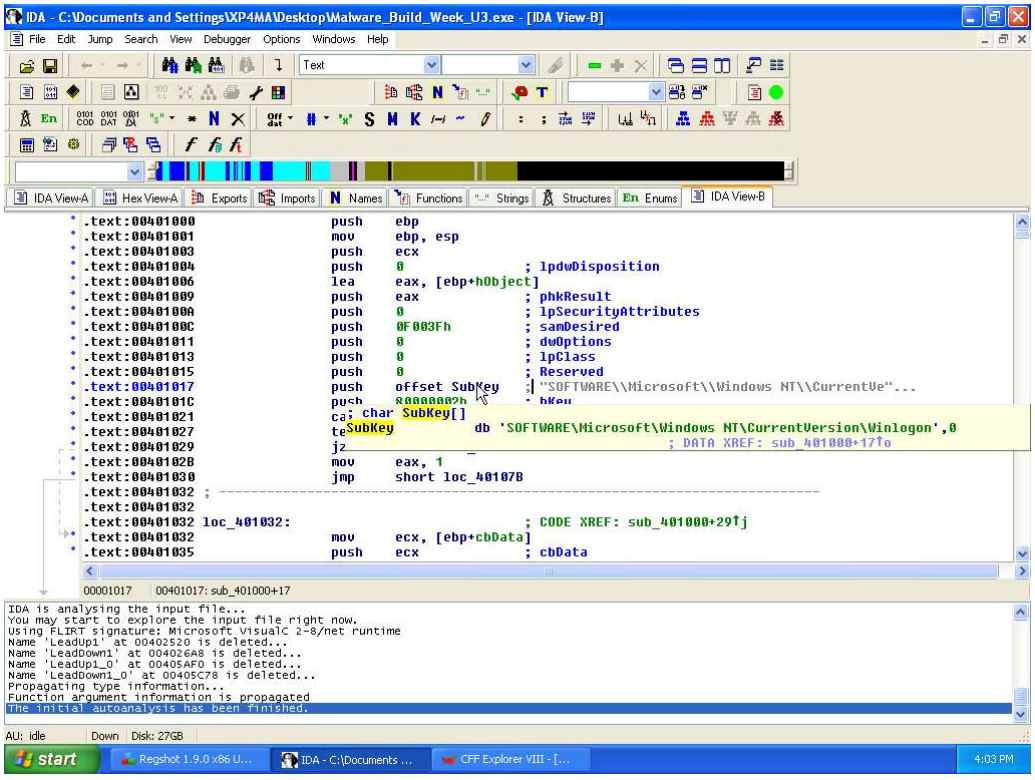
KERNEL32.dll → Il file kernel32.dll è un componente critico del sistema operativo Microsoft Windows. È responsabile della gestione dei processi e delle risorse fondamentali di sistema, tra cui la memoria, i file, la comunicazione tra processi e la gestione degli errori. Ci fornisce anche numerose funzioni di base per le applicazioni Windows, come la creazione e la gestione delle finestre, degli input e degli eventi, dei file e delle cartelle, della stampa e del suono.

ADVAPI32.dll → Questa libreria API contiene una vasta gamma di funzioni che riguardano la gestione degli account utente, la sicurezza, la crittografia, l'accesso al registro di sistema, l'amministrazione dei servizi di Windows e molte altre funzionalità di sistema. Consente ai programmatori di accedere a funzionalità avanzate del sistema operativo per sviluppare applicazioni con funzionalità di sicurezza e gestione della rete

- 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.



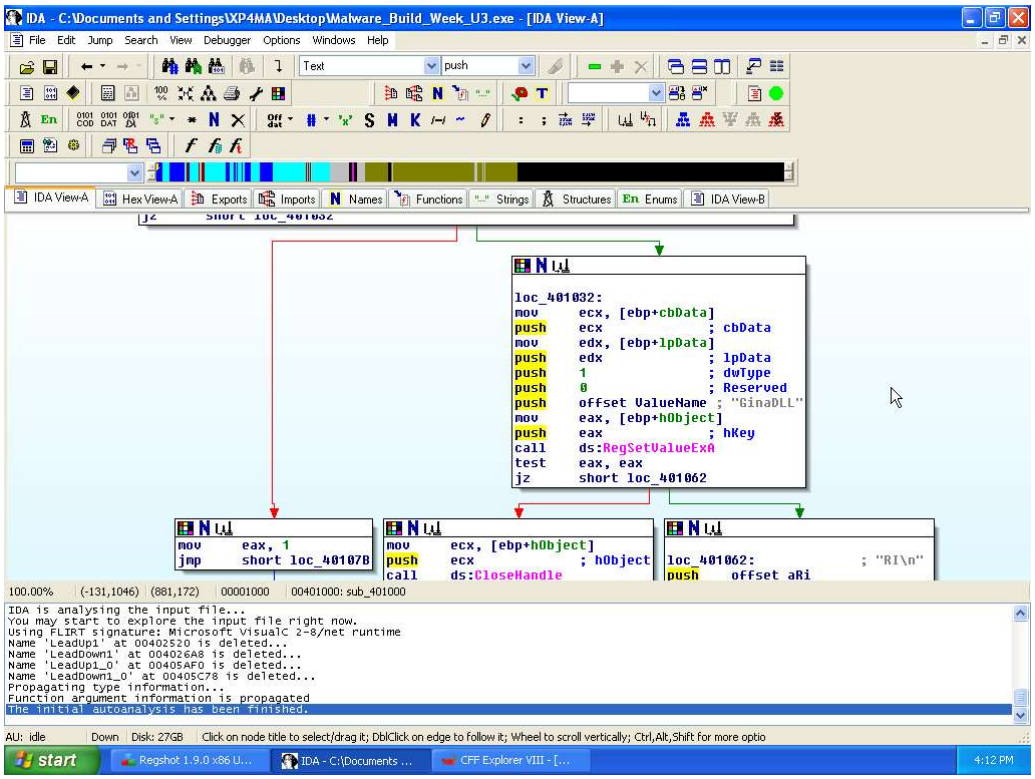
RegCreateKeyExA → funzione di Windows che serve per creare una chiave nel registro di sistema. Prima di chiamare la funzione vengono passati dei valori (parametri della funzione), attraverso diverse istruzioni push.



Offset Subkey → nel contesto delle chiavi di registro nei sistemi operativi Windows, "SubKey" si riferisce a una sottochiave all'interno del registro di sistema. Il registro di sistema di Windows è una gerarchia di database utilizzata per memorizzare configurazioni e informazioni relative al sistema operativo e alle applicazioni. Le chiavi di registro possono essere strutturate in una gerarchia di sottochiavi, che possono contenere valori o ulteriori sottochiavi.

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

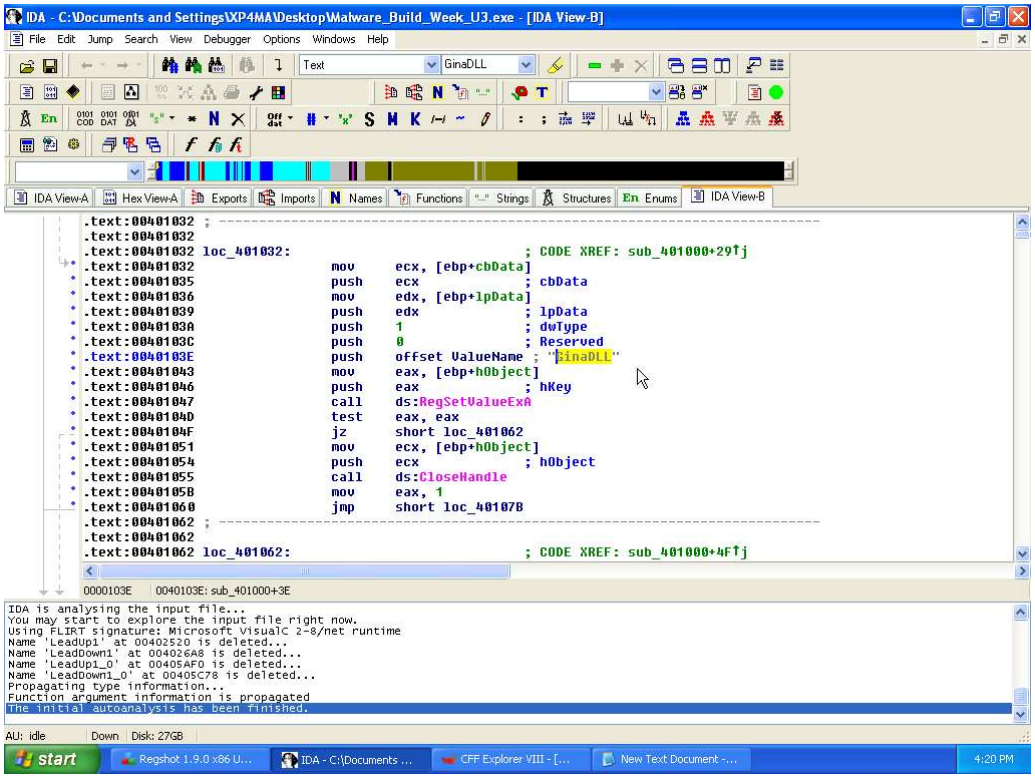
- Con riferimento all'ultimo quesito, tradurre il codice Assembly nel corrispondente costrutto C.



In questa allocazione di memoria il malware verifica se la chiave è già presente nel registro di sistema. Qualora non fosse presente verrà aggiunta, nel caso contrario il malware si chiuderà. Questo controllo è l'equivalente di un *ciclo if* che può essere rappresentato in linguaggio C con il seguente codice:

```
if (eax ==0)
        {
        funct_401032();
        }
else
        {
        eax = 1;
        funct_40107b();
        }
```

- Valutate ora la chiamata alla locazione 00401047, qual è il valore del parametro Value?



Analisi Dinamica

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.

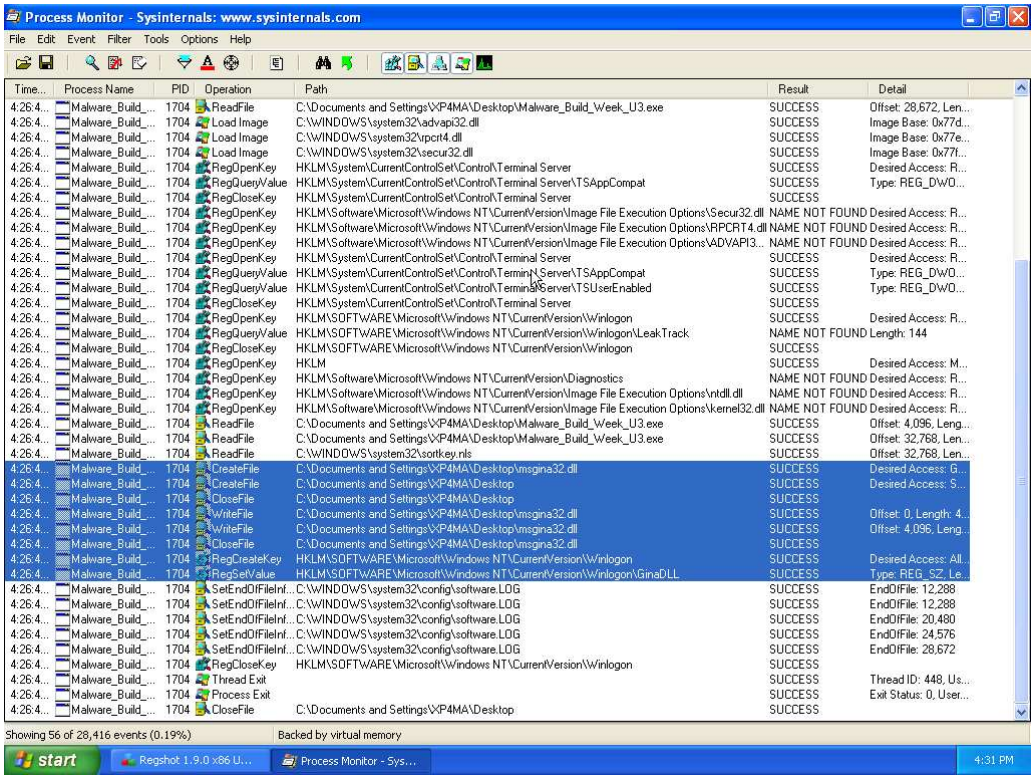
Terminata la fase di "Analisi statica", si procede con "l'analisi dinamica" avviando il malware. Si noti che il malware al primo avvio crea il file "msgina32.dll", il quale viene posizionato nella stessa directory dell'eseguibile.

- 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?



Il programma inserisce una chiave all'interno del registro di sistema di Windows. In particolare, la chiave viene settata all'interno di **HKEY_LOCAL_MACHINE** che contiene impostazioni e configurazioni del computer locale.

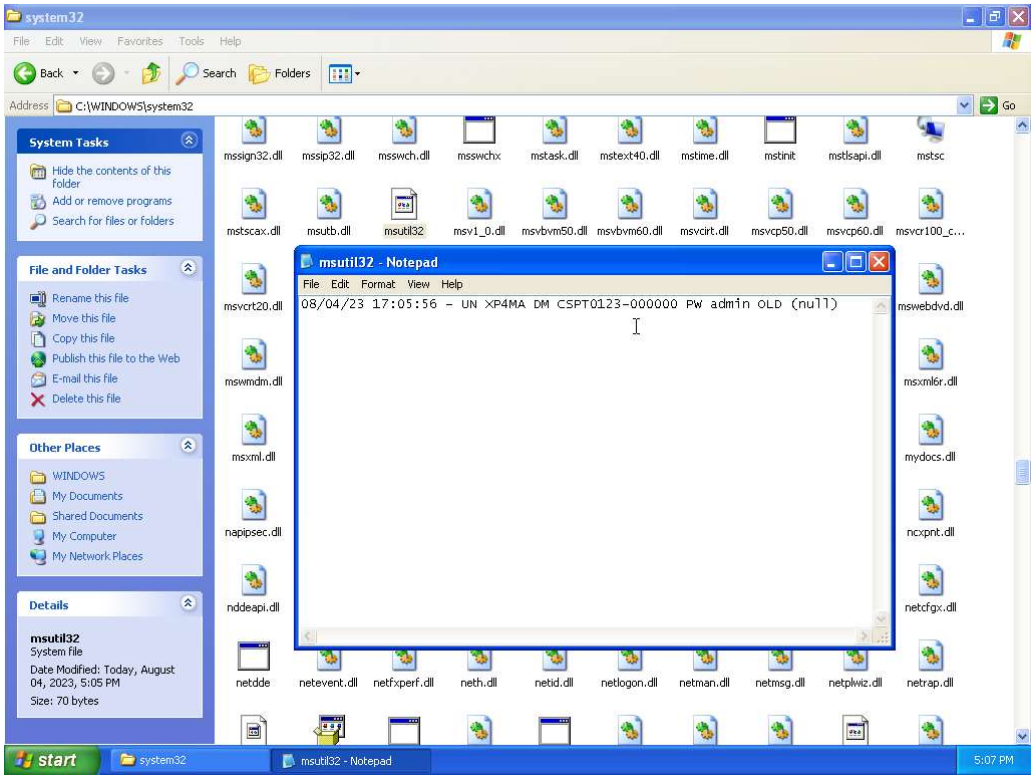
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\WinLogon

Alla chiave viene assegnato il valore "GinaDLL".

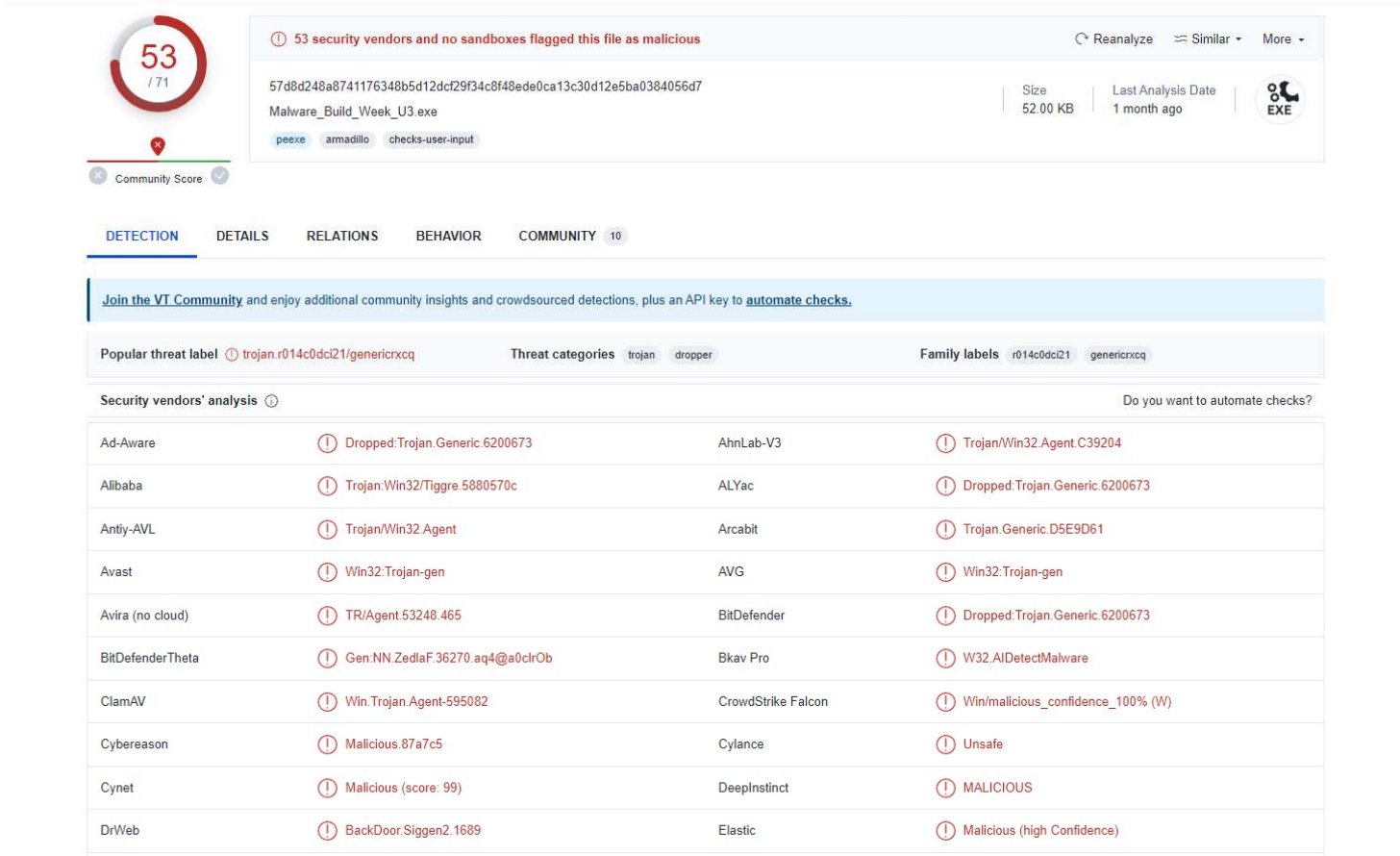
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\WinLogon\GinaDLL

Si nota inoltre, che mediante l'utilizzo della chiamata di sistema CreateFile il programma crea il file "msgina32.dll".

Riavviando il sistema il malware andrà a creare un file di log della password dell'utente corrente all'interno della directory *system32*.



Al netto delle varie fasi di analisi eseguite possiamo dunque, con una certa sicurezza, dedurre che il malware in questione è un dropper contenente un logger/trojan. Ricercando l'hash dell'eseguibile scansionato a monte dell'esercizio sul portale Virus Total otteniamo un riscontro positivo che avvalora la nostra ipotesi.



Considerazioni finali.

La libreria GINA è un componente autentico del processo di login su sistema operativo Windows che permette l'autenticazione dell'utente tramite interfaccia grafica. Il malware in analisi va a sostituire questa libreria con una copia modificata in maniera malevola. Tramite questo processo un utente malintenzionato potrebbe avere accesso alle credenziali dell'utente e ciò metterebbe in serio pericolo la sicurezza della macchina o di una rete.

Nel contesto della sicurezza informatica, questa tecnica viene chiamata *sideloading*. Un attacco *sideloading* è un tipo di attacco informatico in cui l'attaccante carica una libreria dinamica (DLL) dannosa in un'applicazione in esecuzione. La DLL dannosa può quindi essere utilizzata per eseguire codice dannoso, rubare dati o prendere il controllo del sistema. Esistono diversi modi per eseguire un attacco *sideloading*. Un dei più comuni è inserire una DLL modificata in una directory che è nel percorso di ricerca delle DLL di Windows. Quando l'applicazione viene eseguita, Windows cercherà automaticamente le DLL in questa directory e caricherà la DLL dannosa se ha lo stesso nome di quella legittima che richiede l'applicazione.