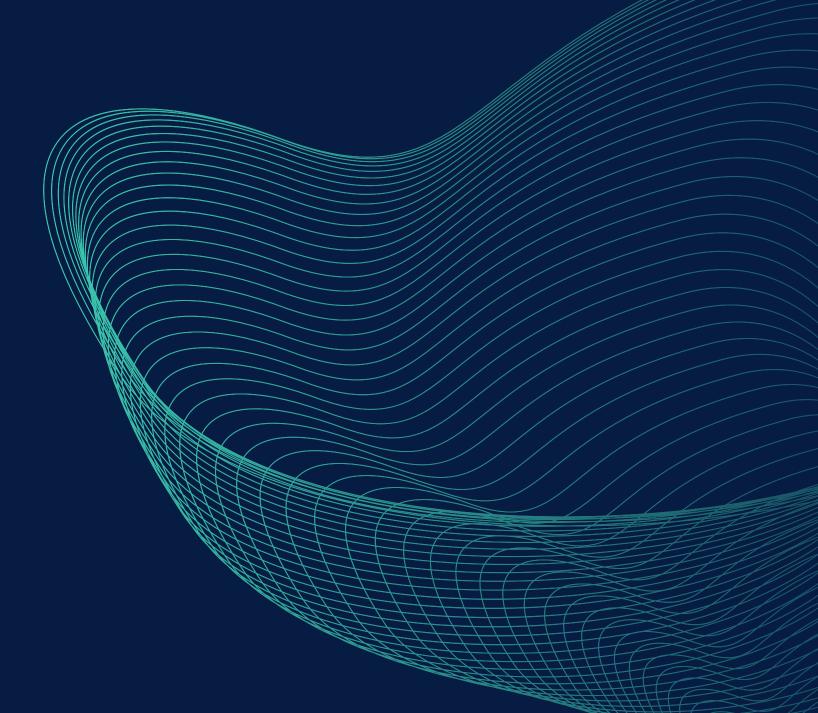
SII-L4



TRACCIA:

La figura nella slide successiva mostra un estratto del codice di un malware. Identificate:

- 1. Il tipo di Malware in base alle chiamate di funzione utilizzate.
- 2. Evidenziate le chiamate di funzione principali aggiungendo una descrizione per ognuna di essa
- 3. Il metodo utilizzato dal Malware per ottenere la persistenza sul sistema operativo
- 4. BONUS: Effettuare anche un'analisi basso livello delle singole istruzioni

TRACCIA:

.text: 00401010 push eax

.text: 00401014 push ebx

.text: 00401018 push ecx

.text: 0040101C push WH_Mouse; hook to Mouse

.text: 0040101F call SetWindowsHook()

.text: 00401040 XOR ECX,ECX

.text: 00401044 mov ecx, [EDI] EDI = «path to

startup_folder_system»

.text: 00401048 movedx, [ESI] ESI = path_to_Malware

.text: 0040104C push ecx; destinationfolder

.text: 0040104F push edx; file to be copied

.text: 00401054 call CopyFile();

Il malware sembrerebbe essere un Keylogger,in quanto viene impostato un hook che cattura gli input del mouse

.text: 0040101C push WH_Mouse ; hook to Mouse

.text: 0040101F call SetWindowsHook()



Le funzioni principali sono due:

SetWindowsHook(): una funzione utilizzata per collegare un hook che consente a un'applicazione di intercettare eventi o messaggi inviati da un altro processo o da un componente del sistema operativo.

CopyFile(): questa funzione è utilizzata per copiare un file da una posizione a un'altra.



Il codice sembra essere una combinazione di operazioni di hooking e di copia di file, ma non fornisce informazioni specifiche su come il malware ottiene persistenza. Tuttavia, è possibile ipotizzare che il malware ottenga persistenza copiando se stesso nella cartella di avvio del sistema o in un'altra posizione in cui verrà eseguito all'avvio del sistema.



.text: 00401010 push eax

.text: 00401014 push ebx

.text: 00401018 push ecx

Queste prime tre istruzioni rappresentano la creazione della stack

.text: 0040101C push WH_Mouse; hook to Mouse

.text: 0040101F call SetWindowsHook()

Questa istruzione mette il valore WH_Mouse nello stack. Probabilmente si tratta di un valore che specifica il tipo di hook da utilizzare per intercettare le operazioni del mouse.



.text: 00401040 XOR ECX,ECX

Il registro ecx viene azzerato

.text: 00401044 mov ecx, [EDI]

EDI = "path to startup_folder_system"

.text: 00401048 mov edx, [ESI]

ESI = path_to_Malware

Queste istruzioni spostano il contenuto della memoria all'indirizzo specificato da EDI e ESI (ESI è il registro di indice sorgente, EDI, d'altra parte, è il registro di indice di destinazione) nei registri ECX EDX.

.text: 0040104C push ecx; destinationfolder

.text: 0040104F push edx; file to be copied

.text: 00401054 call CopyFile();

I valori dei registri ecx e edx vengono pushati nello stack per poi essere utilizzati nella funzione di copia

