

FUNZIONALITÀ DEI MALWARE


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

.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	mov edx, [ESI]	ESI = path_to_Malware
.text: 0040104C	push ecx	; destination folder
.text: 0040104F	push edx	; file to be copied
.text: 00401054	call CopyFile();	

1. Tipo di Malware in base alle chiamate di funzione utilizzate.

.text: 00401010	push eax	
.text: 00401014	push ebx	
.text: 00401018	push ecx	
.text: 0040101C	push WH_Mouse	; hook to Mouse
.text: 0040101F	call SetWindowsHook()	

Come evidenziato dalla sezione di codice in esame, sembra che il Malware appartenga alla famiglia dei **Keylogger**, un particolare tipo di malware programmato per intercettare tutto ciò che l'utente della macchina infetta digita sulla tastiera. Siamo arrivati a questa deduzione in quanto il malware in esame utilizza la funzione **"SetWindowsHook()"**.

2. Evidenziare le chiamate di funzione principali con la loro descrizione.

La funzione “SetWindowsHook()” non fa altro che installare una funzione chiamata “hook” dedicata al monitoraggio degli eventi di una data periferica come tastiera o mouse, possiamo infatti vedere che uno dei parametri “pushati” in cima allo stack della funzione non è altro che “WH_Mouse” (hook to mouse) che consente di monitorare i messaggi del mouse, con la possibilità di monitorare l’input del mouse pubblicato in una coda di messaggi.

<https://learn.microsoft.com/it-it/windows/win32/winmsg/about-hooks>

3. Il metodo utilizzato dal Malware per ottenere persistenza sul sistema operativo.

.text: 00401044	mov ecx, [EDI]	EDI = «path to startup_folder_system»
.text: 00401048	mov edx, [ESI]	ESI = path_to_Malware
.text: 0040104C	push ecx	; destination folder
.text: 0040104F	push edx	; file to be copied
.text: 00401054	call CopyFile();	

Un metodo piuttosto comune utilizzato dai Malware per ottenere persistenza è quello di utilizzare la “startup folder”, questa è una particolare cartella del sistema operativo che viene controllata all’avvio del sistema ed i programmi al suo interno vengono eseguiti. Se un Malware riesce correttamente a copiare il suo eseguibile all’interno della “startup folder”, verrà di conseguenza eseguito automaticamente all’avvio del sistema.

Possiamo infatti vedere dal segmento di codice in esame che il malware prima inizializza a 0 il valore del registro ECX, poi copia il path della “startup folder” nel registro ECX ed il path del file eseguibile del malware nel registro EDX per poi pushare i parametri di tali registri alla funzione CopyFile() che quindi copierà il contenuto di EDX, quindi l’eseguibile del malware, nella “startup folder” del sistema operativo.

4. Effettuare un’analisi di basso livello delle singole istruzioni.

ISTRUZIONE	DESCRIZIONE
Push eax	Inserisce il valore del registro eax nello stack
Push ebx	Inserisce il valore del registro ebx nello stack
Push ecx	Inserisce il valore del registro ecx nello stack
Push WH_Mouse	Inserisce la funzione WH_Mouse nello stack
Call SetWindowsHook()	Chiama la funzione SetWindowsHook()
XOR ECX, ECX	Utilizza l’operatore XOR per inizializzare il registro ECX a 0
Mov ecx, [EDI] (EDI = path_to_startup_folder_system)	Sposta il percorso per la “startup folder” nel registro ecx. EDI (Destination Index) è un registro utilizzato per contenere un indirizzo di memoria
Mov edx, [ESI] (ESI = path_to_Malware)	Sposta il percorso del file eseguibile del malware nel registro edx. ESI (Source Index) è un registro utilizzato per contenere un indirizzo di memoria
Push ecx	Inserisce il valore del registro ecx nello stack
Push edx	Inserisce il valore del registro edx nello stack
Call CopyFile()	Chiama la funzione CopyFile()

