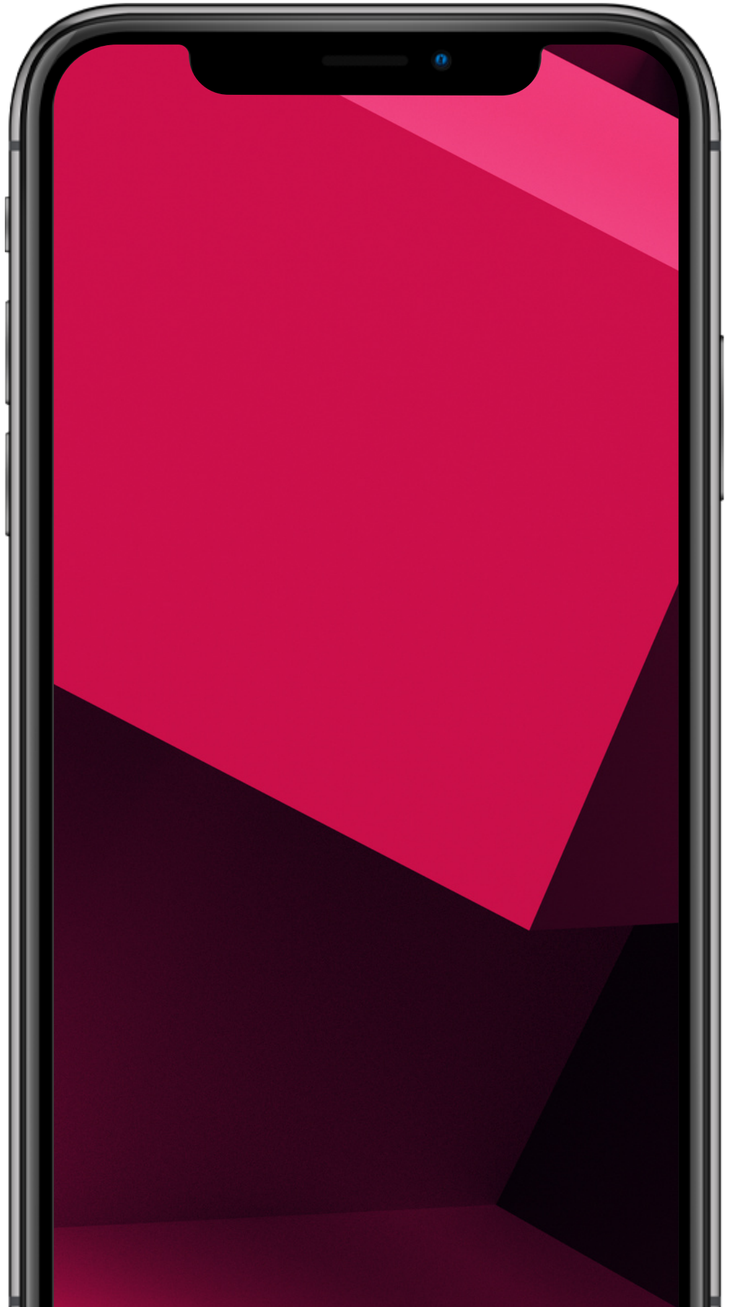
Funzionalità dei Malware

MATTIA PASTORELLI



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

Quesito 1

.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();	

L’assembly suggerisce che il seguente malware possa essere un KeyLogger.
Si può dedurre dalla presenza della funzioni “push WH_Mouse” e “call SetWindowsHook()”,ovvero, una funzioni utili per creare un Hook che andrà a monitorare ogni movimento del mouse (Inclusi spostamenti di file, cartelle aperte o programmi usati”.

Quesito 2

.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();	

Chiamate funzioni principali:

- Call SetWindowsHook() = è utilizzata per installare un hook di sistema, in particolare un hook del mouse, come indicato dal parametro WH_Mouse che viene passato alla funzione.
- Call CopyFile() = viene chiamata per copiare un file specificato (il file identificato dal registro **ESI**, che contiene il percorso del file malware) in una destinazione specificata (il percorso indicato dal registro **EDI**, che contiene il percorso della cartella di avvio del sistema).

Quesito 3

.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();	

La modalità in cui il malware persiste all'interno del sistema è la sequenza di istruzioni "push ecx e push edx" seguite dalla chiamata a "CopyFile()". Questo ci suggerisce che il malware stia copiando se stesso o parte di se stesso in una posizione specifica del sistema.
Dopodichè, attraverso la funzione mov ecx, [EDI] il malware si auto-eseguirà all'avvio del sistema operativo.
Di conseguenza, viene garantito accesso continuo al sistema, registrando le azioni del mouse e salvando ogni passo all'interno di un log specifico.

Quesito 4

.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();	

- Push Eax , push Ebx, Push ecx = spingono i registri nello stack
- Push WH_Mouse = inserisce il valore WH_Mouse nello stack, più specifico Hook per il mouse
- Call SetWindowsHook() = Chiama la funzioneSetWindowsHook() utilizzata per installare un hook di sistema, in questo caso per il mouse.
- XOR ECX, ECX = Operazione logica XOR tra il registro ECX e se stesso, ciò azzera il registro.
- Mov ecx, [EDI] = Muove il valore contenuto all'indirizzo EDI nel registro ECX, in questo caso è stato inizializzato in precedenza attraverso la Cartella Sistema
- Move edx, [ESI] = Muove il valore contenuto all'indirizzo ESI nel registro EDX. Quest'ultimo è stato azzerato con il percorso del file malware.
- Push ecx = Spinge il valore di ecx nello stack, ovvero la cartella di destinazione finale del file.
- Push edx = Spinge il valore di edx nello stack, ovvero il file da copiare
- Call CopyFile() = Chiamata della funzione CopyFile(), questa funzione viene utilizzata per copiare il file da "ESI" (il malware) a "EDI"(cartella di avvio del sistema).