04/04/2024

Funzionalità dei Malware

Prepared by:
Manuel Buonanno

Organized by:
EPICODE

Indice

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();
```

Tipologia malware

Il codice presente nella tabella sotto ci fa pensare ad un Malware di tipo Keylogger, infatti vediamo l'utilizzo della funzione «SetWindowsHook», per l'installazione di un «hook» per controllare un device. Quello che notiamo, tuttavia è che a differenza del codice della lezione teorica, l'ultimo parametro passato sullo stack è «WH_MOUSE». Questo ci fa pensare che il Malware non registra la digitazione dei tasti della tastiera dell'utente, ma bensì la digitazione dei tasti del mouse!

.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 di funzione

In questo codice, ci sono due chiamate di funzioni:

1. call SetWindowsHook(): Questa istruzione chiama la funzione SetWindowsHook(), presumibilmente per impostare un hook di Windows, come suggerito dal nome della funzione. Questo tipo di hook è comunemente utilizzato per intercettare e monitorare eventi del sistema, come eventi di input (nel caso specifico, potrebbe essere un hook del mouse).
2. call CopyFile(): Questa istruzione chiama la funzione CopyFile(), che probabilmente copia un file da una posizione all'altra nel filesystem. Presumibilmente, sta cercando di copiare un file dal percorso del malware alla cartella di avvio del sistema.

Persistenza

Il Malware ottiene la persistenza copiando il suo eseguibile nella cartella di «startup del sistema operativo». Il codice presente nella tabella a partire dall'istruzione 00401040, dapprima setta a zero il registro ECX, successivamente inserisce rispettivamente il path della cartella «startup_folder_system» e l'eseguibile del Malware nei registri ECX ed EDX. In seguito, passa entrambi i registri alla funzione CopyFile() con le due istruzioni push ECX e push EDX. La funzione CopyFile() quindi copierà il contenuto di EDX (ovvero l'eseguibile del malware) nella cartella di startup del sistema operativo.

.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();	

Bonus

1. push eax: Inserisce il valore del registro eax nello stack.
2. push ebx: Inserisce il valore del registro ebx nello stack.
3. push ecx: Inserisce il valore del registro ecx nello stack.
4. push WH_Mouse: Riferimento a una costante o a un indirizzo specifico associato all'hook del mouse. Viene inserito nello stack.
5. call SetWindowsHook(): Chiama una funzione denominata SetWindowsHook(), che probabilmente imposta un hook di Windows, probabilmente per intercettare eventi del mouse.
6. XOR EXC,ECX: Esegue un'operazione di XOR tra i registri EXC e ECX.
7. mov ecx, [EDI]: Muove il valore memorizzato all'indirizzo puntato da EDI nel registro ECX. EDI sembra contenere il percorso della cartella di avvio del sistema.
8. mov edx, [ESI]: Muove il valore memorizzato all'indirizzo puntato da ESI nel registro edx. ESI sembra contenere il percorso del malware.
9. push ecx: Inserisce il percorso della cartella di avvio del sistema nello stack.
10. push edx: inserisce il percorso del malware nello stack.
11. call CopyFile(): Chiama una funzione denominata CopyFile(), probabilmente per copiare un file dal percorso del malware alla cartella di avvio del sistema.

In breve, sembra che il codice stia cercando di impostare un hook del mouse, quindi copiare un file (il malware) dalla sua posizione al percorso della cartella di avvio del sistema.