Funzionalità dei malware

.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

Il codice sembra appartenere a un tipo di malware che cerca di ottenere la persistenza sul sistema e potenzialmente eseguire altre azioni malevole, come il keylogging o il furto di informazioni. Questo tipo di malware è spesso chiamato "Trojan" o "Rootkit", specialmente quando imposta dei hook sui processi di sistema e tenta di replicarsi nella cartella di avvio del sistema operativo.

2. Chiamate di Funzione Principali

Ecco le chiamate di funzione principali e la loro descrizione:

- **SetWindowsHookEx**: Questa funzione imposta un hook su eventi di sistema. Nel codice, viene specificato un hook del mouse (`WH_Mouse`), il che potrebbe indicare che il malware sta cercando di monitorare o manipolare gli eventi del mouse, possibilmente per intercettare input dell'utente.

- **CopyFile**: Questa funzione viene utilizzata per copiare un file da una sorgente a una destinazione. Nel contesto di questo codice, sembra che il malware stia copiando se stesso (o un altro file malevolo) nella cartella di avvio del sistema operativo, garantendo così che venga eseguito ogni volta che il sistema si avvia.

3. Metodo di Persistenza

Il malware ottiene la persistenza copiando se stesso nella cartella di avvio del sistema operativo. Questo assicura che il malware venga eseguito automaticamente ad ogni avvio del sistema, rendendo difficile la sua rimozione.

4. Analisi a Basso Livello delle Istruzioni

Ecco un'analisi a basso livello delle singole istruzioni:

- **push eax**
  - Salva il contenuto del registro `eax` sullo stack.
- **push ebx**
  - Salva il contenuto del registro `ebx` sullo stack.
- **push ecx**
  - Salva il contenuto del registro `ecx` sullo stack.
- **push WH_Mouse**
  - Salva la costante `WH_Mouse` sullo stack, che indica che si sta impostando un hook del mouse.
- **call SetWindowsHook()**
  - Chiama la funzione `SetWindowsHookEx` per impostare un hook del mouse.
- **XOR ECX, ECX**
  - Imposta il registro `ecx` a zero. Questo è un modo comune e veloce di azzerare un registro.

- **mov ecx, [EDI]**
  - Carica il valore contenuto nell'indirizzo puntato da `EDI` nel registro `ecx`. `EDI` sembra puntare alla destinazione dove il malware vuole copiare se stesso.
- **mov edx, [ESI]**
  - Carica il valore contenuto nell'indirizzo puntato da `ESI` nel registro `edx`. `ESI` sembra puntare al file del malware che deve essere copiato.
- **push ecx**
  - Salva il contenuto di `ecx` sullo stack, che contiene il percorso di destinazione.
- **push edx**
  - Salva il contenuto di `edx` sullo stack, che contiene il percorso del file da copiare.
- **call CopyFile()**
  - Chiama la funzione `CopyFile` per copiare il file del malware dalla sorgente alla destinazione.

Conclusione

Questo codice mostra un esempio di malware che tenta di ottenere persistenza nel sistema attraverso la copia di un file malevolo nella cartella di avvio del sistema e potenzialmente monitorare gli eventi del mouse. Le chiamate a `SetWindowsHookEx` e `CopyFile` sono cruciali per le sue operazioni.