Manuel Di Gangi

S11_L4

Funzionalità dei Malware

04 aprile 2024

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();
```

1. Tipo di malware

Analizzando le chiamate di funzione possiamo affermare che il malware in questione sia un key logger, in particolare lo possiamo dedurre dalla chiamata della funzione **SetWindowsHook**. Questa funzione non fa altro che installare un metodo (una funzione) chiamato «hook» dedicato al monitoraggio degli eventi di una data periferica, come ad esempio la tastiera o il mouse. Il metodo «hook» verrà allertato ogni qualvolta l'utente digiterà un tasto sulla tastiera e salverà le informazioni su un file di log.

Tuttavia a differenza della lezione teorica questo keylogger non registra i tasti della tastiera bensì registra i comportamenti del mouse, si può capire dal parametro **WH_mouse** che viene caricato nello stack prima della chiamata della funzione.

```
.text: 0040101C    push WH_Mouse          ; hook to Mouse
.text: 0040101F    call SetWindowsHook()
```

2. Chiamate di funzione principali

SetWindowsHook: Questa funzione non fa altro che installare un metodo (una funzione) chiamato «hook» dedicato al monitoraggio degli eventi di una data periferica, come ad esempio la tastiera o il mouse. Il metodo «hook» verrà allertato ogni qualvolta l'utente digiterà un tasto sulla tastiera e salverà le informazioni su un file di log.

CopyFile: è un'API utilizzata per copiare un file da una posizione a un'altra. Accetta tre parametri principali: il percorso del file di origine da cui copiare, il percorso della destinazione in cui copiare il file e un flag che indica se sovrascrivere il file di destinazione se esiste già uno con lo stesso nome. In questo caso la funzione CopyFile() ha solo due parametri, cioè il percorso di destinazione (salvato nel registro ECX ossia la directory di avvio del sistema) ed il path per

raggiungere la directory sorgente del file, potrebbe essere perché il file di origine è implicitamente considerato nel contesto dell'applicazione o del sistema operativo.

3. Persistenza

La tecnica utilizzata dal Malware per ottenere la persistenza è quella di utilizzare la «startup folder».

La «startup folder» è una particolare cartella del sistema operativo che viene controllata all'avvio del sistema, ed i programmi che sono al suo interno vengono eseguiti. I sistemi Windows mantengono due tipi di cartelle di startup:

- Una dedicata agli utenti, e diversa per ogni utente del sistema
- Una generica del sistema operativo, comune a tutti gli utenti del sistema operativo.

Una volta che il Malware riesce correttamente a copiare il suo eseguibile all'interno di una delle cartelle sopra, verrà di conseguenza eseguito automaticamente all'avvio del sistema (se presente nella cartella generica), oppure all'avvio del sistema da parte dell'utente specifico se presente solo nella cartelle utente.

4. Analisi basso livello del codice

push	eax	Carica il contenuto del registro eax nello stack
push	ebx	Carica il contenuto del registro ebx nello stack
push	ecx	Carica il contenuto del registro ecx nello stack
push	WH_Mouse	Carica il contenuto della variabile WH_Mouse nello stack
call	SetWindowsHook()	Chiama la funzione SetWindowsHook()
xor	ecx, ecx	Azzera il registro ecx
mov	ecx, [EDI]	Sposta il contenuto dell'indirizzo puntato del registro EDI nel registro ecx
moev	edx, [ESI]	Sposta il contenuto dell'indirizzo puntato del registro ESI nel registro edx
push	ecx	Carica il contenuto del registro ecx nello stack
push	edx	Carica il contenuto del registro edx nello stack
call	CopyFile()	Chiama la funzione CopyFile()