Cybersecurity Specialist

Cyber Security & Ethical Hacking

GIORNO 4 –MALWARE ANALYSIS

Alejandro Cristino
S11-L4

TRACCIA

La figura 1 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

Figura 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();
```

1. Il tipo di Malware in base alle chiamate di funzione utilizzate.

La funzione SetWindowsHook() viene spesso utilizzata dai malware per intercettare l’input da parte dell’utente. Se il valore passato è WH_MOUSE, possiamo dedurre che il malware sia progettato per registrare gli eventi legati all’uso del mouse. Questo tipo specifico di malware monitora e memorizza i movimenti e i click del mouse.

Questa funzionalità può essere sfruttata per raccogliere informazioni sensibili, come elementi dell’interfaccia utente, informazioni che potrebbero essere usate per accedere a dati critici o per tracciare le abitudini dell’utente. Inoltre, l’intercettazione degli eventi del mouse potrebbe essere utilizzata per eseguire attacchi più sofisticati, come il monitoraggio delle azioni all’interno di applicazioni specifiche o la cattura di screenshot in momenti mirati.

2. Evidenziate le chiamate di funzione principali aggiungendo una descrizione per ognuna di essa

1. SetWindowsHook()

La funzione SetWindowsHook() è utilizzata per installare un hook, una funzione che può monitorare diversi eventi di sistema. Nel contesto specifico del codice, viene utilizzato il parametro WH_MOUSE, che serve a catturare e monitorare gli eventi del mouse. Questa funzione permette al malware di intercettare e registrare i movimenti e i clic del mouse, consentendogli di raccogliere dati sensibili o di osservare il comportamento dell'utente.

2. CopyFile()

La funzione CopyFile() è utilizzata per duplicare un file da una posizione all'altra nel file system. In questo caso, il malware sfrutta questa funzione per copiare il proprio eseguibile in una cartella di avvio del sistema. Copiando l'eseguibile nella cartella di avvio (startup_folder_system), il malware si assicura di essere eseguito automaticamente ogni volta che il sistema viene avviato, garantendo così la sua persistenza.

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

La persistenza del malware viene garantita attraverso la chiamata della funzione CopyFile(), in cui viene passato il parametro “EDI = path to startup_folder_system”. Questo parametro specifica il percorso della cartella di avvio del sistema, permettendo al malware di essere copiato in questa directory. Di conseguenza, il malware sarà eseguito automaticamente ad ogni avvio del sistema operativo, assicurandosi di rimanere attivo senza necessità di ulteriori interventi da parte dell’attaccante.

4. BONUS: Effettuare anche un’analisi basso livello delle singole istruzioni

Istruzione Assembly	Descrizione
push eax	Effettua il push nello stack del registro eax.
push ebx	Effettua il push nello stack del registro ebx.
push ecx	Effettua il push nello stack del registro ecx.
push WH_Mouse	Effettua il push nello stack del parametro WH_Mouse.
call SetWindowsHook()	Chiamata della funzione SetWindowsHook.
XOR ECX, ECX	Operatore logico per assegnare il valore 0 al registro ecx.
mov ecx, [EDI] = path to startup_folder_system	Sposta il path della cartella startup nel registro ecx.
mov edx, [ESI] = path_to_Malware	Sposta il path della cartella malware nel registro edx.
push ecx	Effettua il push nello stack del registro ecx.
push edx	Effettua il push nello stack del registro edx.
call CopyFile()	Chiamata della funzione CopyFile.

Cybersecurity Specialist

Cyber Security & Ethical Hacking

GRAZIE.