Compito S10-L4

Traccia

La figura seguente mostra un estratto del codice di un malware.
Identificare i costrutti noti visti durante la lezione teorica.

```
.text:00401000                push    ebp
.text:00401001                mov     ebp, esp
.text:00401003                push    ecx
.text:00401004                push    0               ; dwReserved
.text:00401006                push    0               ; lpdwFlags
.text:00401008                call    ds:InternetGetConnectedState
.text:0040100E                mov     [ebp+var_4], eax
.text:00401011                cmp     [ebp+var_4], 0
.text:00401015                jz      short loc_40102B
.text:00401017                push    offset aSuccessInterne ; "Success: Internet Connection\n"
.text:0040101C                call    sub_40105F
.text:00401021                add     esp, 4
.text:00401024                mov     eax, 1
.text:00401029                jmp     short loc_40103A
.text:0040102B ; ---------------------------------------------------------------------------
.text:0040102B
```

Traccia

Provate ad ipotizzare che funzionalità è implementata nel codice assembly.

Hint :
La funzione internetgetconnectedstate prende in input 3 parametri e permette di controllare se una macchina ha accesso ad Internet.

Consegna:

1. Identificare i costrutti noti (es. while, for, if, switch, ecc.)
2. Ipotizzare la funzionalità – esecuzione ad alto livello
3. BONUS: studiare e spiegare ogni singola riga di codice

Esercizio

```
mov     [ebp+var_4], eax
cmp     [ebp+var_4], 0
jz      short loc_40102B
```

Costrutto IF se il valore di ebp+var_4 è zero, salta a loc_40102B

```
push    offset aSuccessInterne ; "Success: Internet Connection\n"
call    sub_40105F
add     esp, 4
mov     eax, 1
jmp     short loc_40103A
```

Altrimenti esegue questa parte di codice

Esercizio

Possiamo ipotizzare che il malware sta verificando lo stato della connessione a Internet sulla macchina ospite. Ecco un'ipotesi delle funzionalità implementate:

Chiamata a InternetGetConnectedState:

- Il malware chiama la funzione InternetGetConnectedState per verificare lo stato della connessione Internet sulla macchina.

Controllo dello stato di connessione:

- Dopo la chiamata a InternetGetConnectedState, il malware salva il risultato in una variabile locale (var_4).

Condizione di controllo:

- Il malware confronta il risultato con 0 utilizzando l'istruzione cmp e poi esegue una condizione jz (salta se zero) che porta a loc_40102B.

Gestione dei casi:

- Se la condizione è soddisfatta (Internet non connesso), il malware può eseguire una serie di istruzioni a loc_40102B per gestire il caso in cui la connessione Internet non è attiva. Questa parte del codice non è fornita, quindi non possiamo determinare la sua esatta funzionalità.

Output di successo:

- Se la connessione Internet è attiva, il malware stampa il messaggio "Succes Internet Connection\n" utilizzando la funzione printf (chiamando sub_40105F).

Ritorno e terminazione:

- Dopo la gestione dei casi, il malware ritorna 1 (mov eax, 1) e salta a loc_40103A. Questo potrebbe indicare che la funzionalità principale è stata eseguita con successo.

Esercizio

- **push ebp:** Salva il valore del registro base (EBP) nello stack.
- **mov ebp, esp:** Imposta il registro base (EBP) al valore corrente dello stack, creando un nuovo frame di stack**.**
- **push ecx:** Salva il valore del registro ECX nello stack.
- **push 0:** Mette il valore 0 nello stack, presumibilmente come argomento per la funzione InternetGetConnectedState.
- **push 0:** Mette il valore 0 nello stack, un altro possibile argomento per la funzione.
- **call ds:InternetGetConnectedState:** Chiama la funzione InternetGetConnectedState dalla sezione di dati (ds).
- **mov [ebp+var_4], eax:** Salva il risultato della funzione chiamata in var_4, che sembra essere una variabile locale all'interno del frame di stack.
- **cmp [ebp+var_4], 0:** Compara il valore di var_4 con 0.
- **jz short loc_40102B:** Salta a loc_40102B se la comparazione precedente restituisce zero (Internet non connesso).
- **push offset aSuccessInterne:** Mette l'indirizzo della stringa "Succes Internet Connection\n" nello stack.
- **call sub_40105F:** Chiama una funzione, probabilmente per gestire l'output della stringa.
- **add esp, 4:** Libera 4 byte dallo stack dopo la chiamata della funzione.
- **mov eax, 1:** Imposta il registro EAX a 1, indicando probabilmente una connessione Internet riuscita.
- **jmp short loc_40103A:** Salta a loc_40103A, evitando l'esecuzione della sezione a loc_40102B.