Manuel Di Gangi

S10_L4

Costrutti C -Assembly x86

28 marzo 2024

INDICE

Traccia.....2
1. Costrutti noti.....3
2. Ipotesi di funzionamento.....3
3. Studio del codice riga per riga.....4

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
```

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

1. Costrutti noti

00401000	1° costrutto	push	ebp
00401001		mov	ebp, esp
00401003	2° costrutto	push	ecx
00401004		push	0
00401006		push	0
00401008		call	ds: InternetGetConnectedState
0040100E	3° costrutto	mov	[ebp+var_4], eax
00401011		cmp	[ebp+var_4], 0
00401015		jz	short loc_40102B
00401017		push	offset aSuccessInterne
0040101C	4° costrutto	call	sub_40105F

1° costrutto - Creazione di uno stack di dimensione non specificata, viene allocata dinamicamente;

2° costrutto - Chiamata di funzione, i parametri sono passati tramite lo stack

3° costrutto - Condizione if, se la variabile ebp+var_4 è == 0 salta.

4° costrutto - Chiamata di funzione

2. Ipotesi di funzionamento

Il codice in esame chiama la funzione InternetGetConnectedState e ne verifica il valore di ritorno medianto un costrutto IF, qualora il valore ritornato sia diverso da zero la macchina è connessa

ad internet e si suppone che il programma stampi un messaggio del tipo “Success: Internet Connection”.

3. Studio del codice riga per riga

1. push ebp -> Inserisce il contenuto di ebp all’interno dello stack
2. mov ebp, esp -> sposto il contenuto di esp in ebp
3. push ecx -> Inserisco il contenuto di ecx nello stack

   push 0 ;dwReserved -> Inserisco 0 nello stack, il ‘ ; ‘ serve ad indicare un commento. Nella funzione “InternetGetConnectedState”, il parametro dwReserved viene riservato per l'uso futuro e non è attualmente utilizzato. In pratica, viene lasciato vuoto o impostato su zero e non ha alcuna influenza sul comportamento della funzione stessa.

4. push 0 ;lpdwFlags -> Inserisco 0 nello stack.

   E’ un parametro di tipo puntatore utilizzato per passare informazioni aggiuntive o per ricevere dati da una funzione. Il prefisso "lp" sta per "long pointer", indicando che si tratta di un puntatore a un valore DWORD. Nella funzione InternetGetConnectedState menzionata in precedenza, il parametro lpdwFlags è utilizzato per restituire lo stato della connessione Internet. La funzione scrive i dati relativi allo stato della connessione all'indirizzo di memoria a cui punta lpdwFlags, consentendo al chiamante di ottenere tali informazioni dopo aver chiamato la funzione.

5. call ds:InternetGetConnectedState -> Chiama la funzione InternetGetConnectedState. Questa funzione viene utilizzata per verificare lo stato della connessione Internet su un sistema operativo Windows.Quando la funzione restituisce TRUE, la connessione Internet è attiva, se al contrario restituisce FALSE, non c'è alcuna connessione.

```cpp
#include <windows.h>
#include <wininet.h>
#include <iostream>

int main() {
    // Variabile per memorizzare lo stato della connessione
    DWORD flags;

    // Verifica lo stato della connessione Internet
    if(InternetGetConnectedState(&flags, 0)) {
        // La connessione è attiva
        std::cout << "Connessione Internet attiva." << std::endl;
    } else {
        // Nessuna connessione Internet
        std::cout << "Nessuna connessione Internet attiva." << std::endl;
    }

    return 0;
}
```

6. mov [ebp+var_4], eax -> Sposta il contenuto di eax nella variabile locale. In particolare essendo eax il registro accumulatore al suo interno è stato salvato il risultato ritornato dalla funzione chiamata in precedenza.
7. cmp [ebp+var_4],0 -> compara la variabile locale con 0
8. jz short loc_40102B -> Salta se il risultato dell'istruzione precedente è Ttrue
9. push offset aSuccesInterne -> Inserisce la variabile nello stack
10. call sub_40105F -> Chiama una sotto funzione
11. add esp, 4 -> aumenta il valore del registro esp di 4 liberando lo stack dalla push effettuata a riga 9
12. mov eax, 1 -> Sposta il valore 1 all'interno del registro eax
13. jump short loc_40103A -> effettua un salto incondizionato, probabilmente per saltare il codice eseguito dopo il salto di riga 8