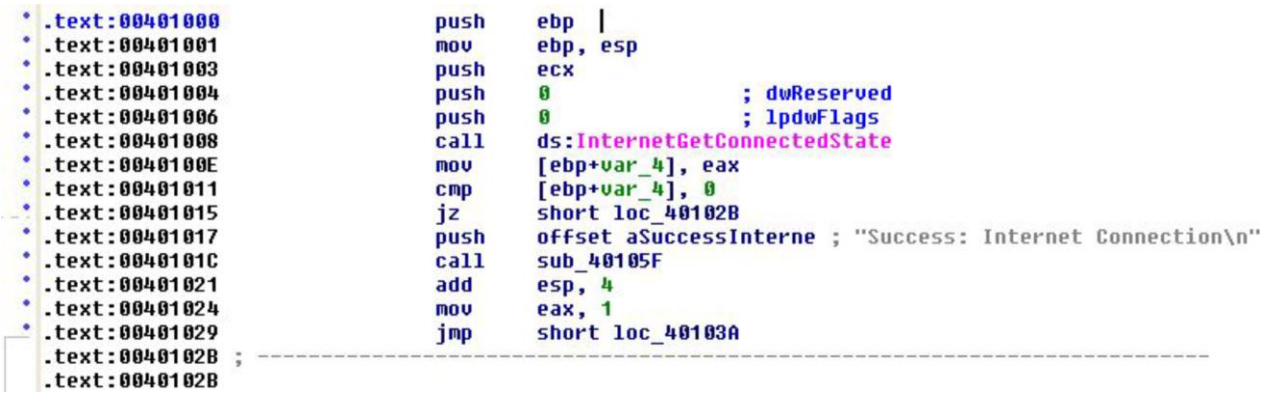
Report UNIT3 WEEK 10.4

ASSEMBLY



L'estratto di oggi fa parte della sezione .text, dove vengono indicate le istruzioni del codice del programma.

Viene definita la subroutine (**sub_401000**), una funzione con codice sorgente scritto in C++.

In assembly queste vengono implementate tramite etichette per indicare inizio e fine (call, ret).

push ebp

Inserisce il valore di EBP (base point register) nello stack. Crea un frame pointer per accedere ai parametri e alle variabili locali.

mov ebp, esp

Collega il frame pointer allo stack, copiando il valore di ESP (stack point register) in EBP.

push ecx

Inserisce il valore di ECX (extended counter register) nello stack.

push 0 ; dwReserved

Inserisce il valore 0 nello stack come argomento preparatore per la funzione "**InternetGetConnectedState**".

Questo valore corrisponde a dwReserved.

push 0 ; lpdwFlags

Inserisce il valore 0 nello stack come argomento preparatore per la funzione "**InternetGetConnectedState**".

Questo valore corrisponde a lpdwFlags.

call ds:InternetGetConnectedState

Chiama la funzione **InternetGetConnectedState** dal segmento DS (Data Segment) per verificare lo stato della connessione Internet.

mov [ebp+var_4], eax

Copia il risultato della chiamata alla funzione nella variabile locale [ebp+var_4].

cmp [ebp+var_4], 0

Confronta il valore della variabile locale [ebp+var_4] con 0.

jz short loc_40102B

Salta all’indirizzo loc_40102B se il risultato del confronto precedente è uguale a zero.

push offset aSuccessInterne

Push dell'offset della stringa "**Success: Internet Connection\n**" nello stack.

call sub_40105F

Chiama la funzione sub_40105F per eseguire l’output se la condizione precedente risulta falsa ([ebp+var_4] != 0).

Add esp, 4

Ripristina l’indicatore dello stack dai valori pushati per continuare con le istruzioni successive.

mov eax, 1

Imposta EAX a 1.

jmp short loc_40103A

Salta all’indirizzo loc_40103A.

Opzionale

```
.text:00401000 ; ||||||||||||||| S U B R O U T I N E |||||||||||||||||||||||||||||||||||||||
.text:00401000
.text:00401000 ; Attributes: bp-based frame
.text:00401000
.text:00401000 sub_401000      proc near               ; CODE XREF: _main+6↓p
.text:00401000
.text:00401000 var_4           = dword ptr -4
.text:00401000
.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_40117F
.text:00401021                 add     esp, 4
.text:00401024                 mov     eax, 1
.text:00401029                 jmp     short loc_40103A
.text:0040102B ; ---------------------------------------------------------------------------
.text:0040102B
.text:0040102B loc_40102B:                             ; CODE XREF: sub_401000+15↑j
.text:0040102B                 push    offset aError1_1NoInte ; "Error 1.1: No Internet\n"
.text:00401030                 call    sub_40117F
.text:00401035                 add     esp, 4
.text:00401038                 xor     eax, eax
.text:0040103A
.text:0040103A loc_40103A:                             ; CODE XREF: sub_401000+29↑j
.text:0040103A                 mov     esp, ebp
.text:0040103C                 pop     ebp
.text:0040103D                 retn
.text:0040103D sub_401000      endp
.text:0040103D
.text:0040103D ; ---------------------------------------------------------------------------
.text:0040103E                 align 10h
.text:00401040
.text:00401040 ; ||||||||||||||| S U B R O U T I N E |||||||||||||||||||||||||||||||||||||||
.text:00401040
```

Viene definita la funzione sub_401000 con offset +6 byte dalla main.

Viene dichiarata la variabile var_4 all'indirizzo ESP -4 (posizionata 4 byte sopra ESP), questa è un double word a 32bit.

Viene inizializzato lo stack per la funzione, InternetGetConnectedState è una funzione della libreria WININET.DLL in ambienti Windows. Viene usata per determinare se l'utente ha una connessione internet attiva.

Questa funzione confronta i parametri, se il risultato è uguale a zero salta alla funzione loc_40102B (print "Error 1.1: No Internet) che chiama la sub_40117F. In seguito, vengono puliti gli argomenti dallo stack e con xor viene impostato EAX a 0.

Se il risultato è diverso da zero salta alla funzione loc_40102B (print "Success: Internet Connection"), vengono puliti gli argomenti dallo stack e viene impostato EAX a 1. Successivamente fa un jump a loc_40103 che ripristina lo stato dello stack ed esegue la funzione di retn (return) che termina la funzione sub_401000(endp).

Codice in C

```c
#include <stdio.h>
#include <wininet.h>

void sub_401000() {
    int var_4;

    int dwReserved = 0;
    int lpdwFlags = 0;

    InternetGetConnectedState(&lpdwFlags, dwReserved);

    var_4 = lpdwFlags;

    if (var_4 == 0) {
        printf("Success: Internet Connection\n");
    } else {
        printf("Error 1.1: No Internet\n");
    }
}

int main() {
    sub_401000();

    return 0;
}
```