Consegna S10-L4

Costrutti C - Assembly X86

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

Opzionale: Provate ad ipotizzare che funzionalità è implementata nel codice assembly.

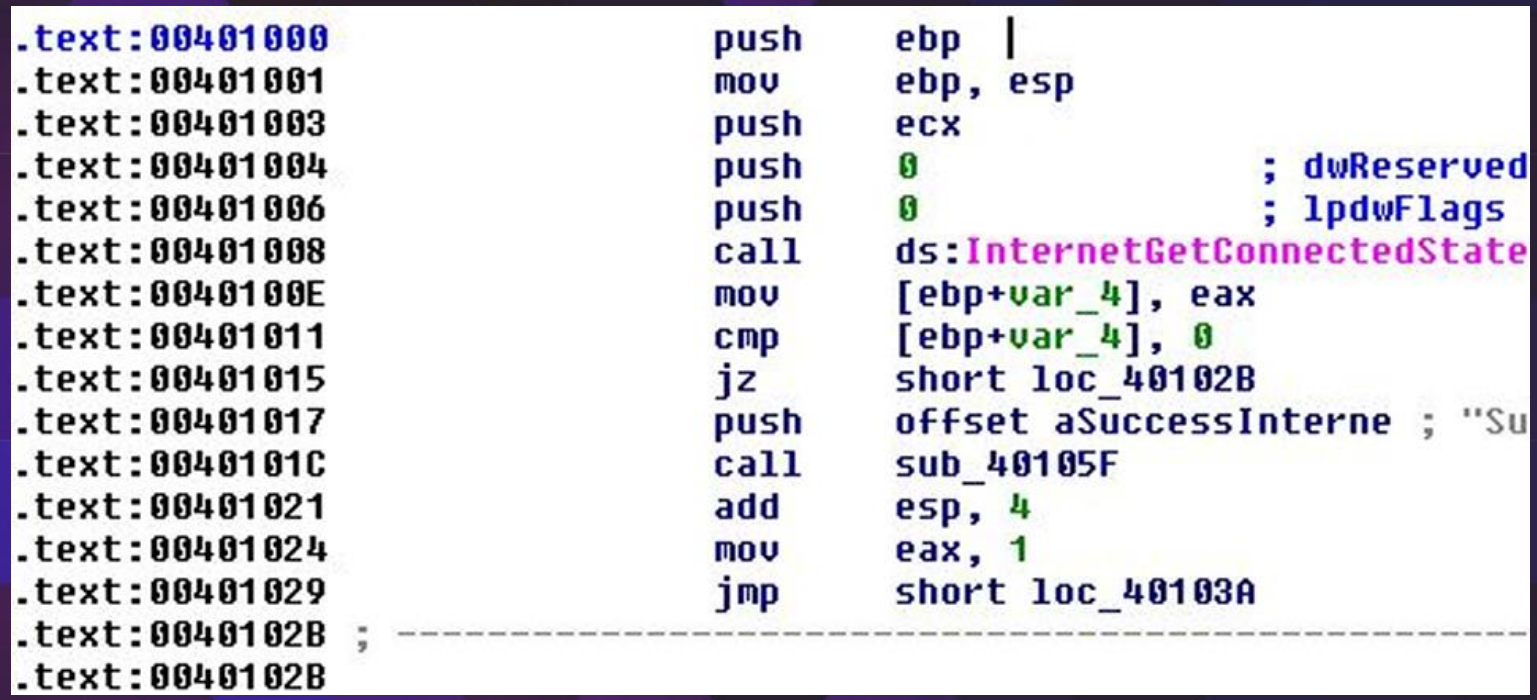
Hint: La funzione internetgetconnectedstate prende in input 3 parametri e permette di controllare se una macchina ha accesso ad internet.

Creazione dello stack

Il primo costrutto che si può notare è la creazione dello stack ed è principalmente composto da due istruzioni:

push ebp: Questa istruzione mette il valore corrente del registro base del puntatore dello stack (ebp) nello stack. Questo è un passo comune per salvare il contesto della funzione chiamante prima di creare un nuovo frame dello stack per la funzione corrente.

mov ebp, esp: Questa istruzione imposta il registro base del puntatore dello stack (ebp) uguale al puntatore corrente dello stack (esp). Questo crea un nuovo frame dello stack per la funzione corrente, consentendo l'accesso ai parametri della funzione e alle variabili locali utilizzando un offset da ebp.



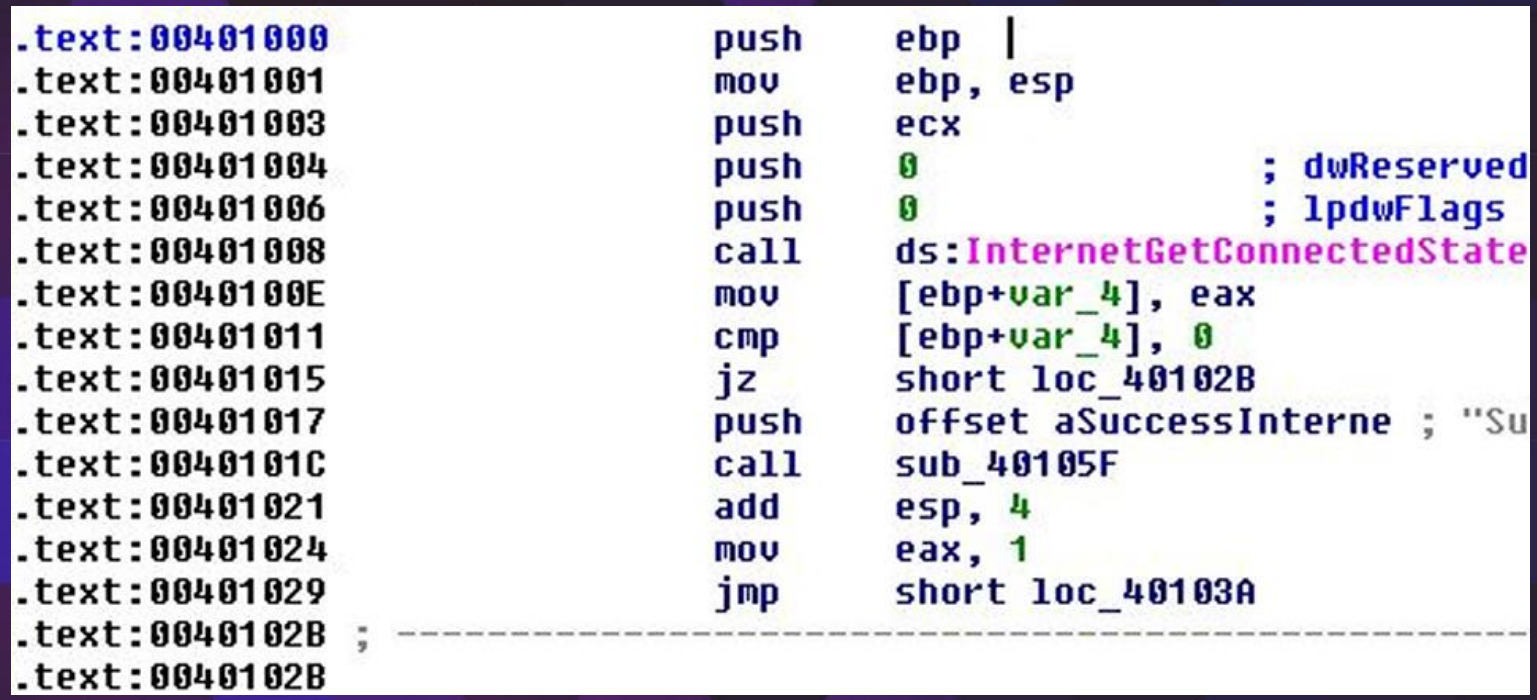
Chiamata di funzione

Questa parte del codice, riguarda la preparazione degli argomenti per la chiamata della funzione ed è principalmente composta da tre istruzioni:

push ecx: Salva il valore del registro ecx nello stack

push 0: Mette il valore 0 nello stack. In questo contesto specifico, viene utilizzata per preparare un argomento per la funzione chiamata InternetGetConnectedState. Questo valore 0 potrebbe essere utilizzato come parametro dwReserved, che potrebbe indicare un'area di memoria riservata per la funzione.

push 0: Mette il valore 0 nello stack. Nel contesto del codice, questo secondo valore 0 potrebbe essere utilizzato come parametro lpdwFlags, che potrebbe indicare dei flag per la funzione InternetGetConnectedState.



Istruzione condizionale

Questa parte del codice riguarda la creazione della condizione per l'istruzione if ed è principalmente composta da due istruzioni:

cmp [ebp+var_4], 0: Questa istruzione confronta il valore memorizzato in [ebp+var_4] con 0. Se [ebp+var_4] è uguale a 0, la condizione sarà vera.

jz short loc_40102B: Questa istruzione di salto condizionato (jz) salta a loc_40102B solo se il confronto precedente ha dato esito positivo (cioè se [ebp+var_4] è uguale a 0).

```
.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 ; "Su
.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
```

Funzionalità implementata

Questo codice assembly sembra implementare un programma dedicato a verificare lo stato della connessione Internet su un sistema operativo. La funzione chiave utilizzata, InternetGetConnectedState, accetta dei parametri e restituisce un valore che indica lo stato della connessione. La struttura condizionale if esamina questo valore restituito: se è positivo, il programma procede stampando un messaggio di successo.

Fine della presentazione

Amedeo Natalizi