

## Pratica S10/L4 - Costrutti C - Assembly X8

L'esercizio di oggi consiste nell'identificare i costrutti del malware sotto in figura.

```
.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
```

Come prima cosa possiamo notare la creazione dello stack nelle prime 2 righe

```
.text:00401000      push    ebp |
.text:00401001      mov     ebp, esp
```

Come seconda cosa vediamo come i parametri vengano passati allo stack tramite push e il call prende in input i 3 parametri per controllare se la macchina ha accesso ad internet.

```
.text:00401003      push    ecx
.text:00401004      push    0           ; dwReserved
.text:00401006      push    0           ; lpdwFlags
.text:00401008      call    ds:InternetGetConnectedState
```

E sotto inizia un ciclo IF-ELSE

```
.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
```

Probabilmente questa parte di codice del malware controlla se c'è la connessione ad internet e se esse c'è stampa un messaggio di connessione avvenuta.