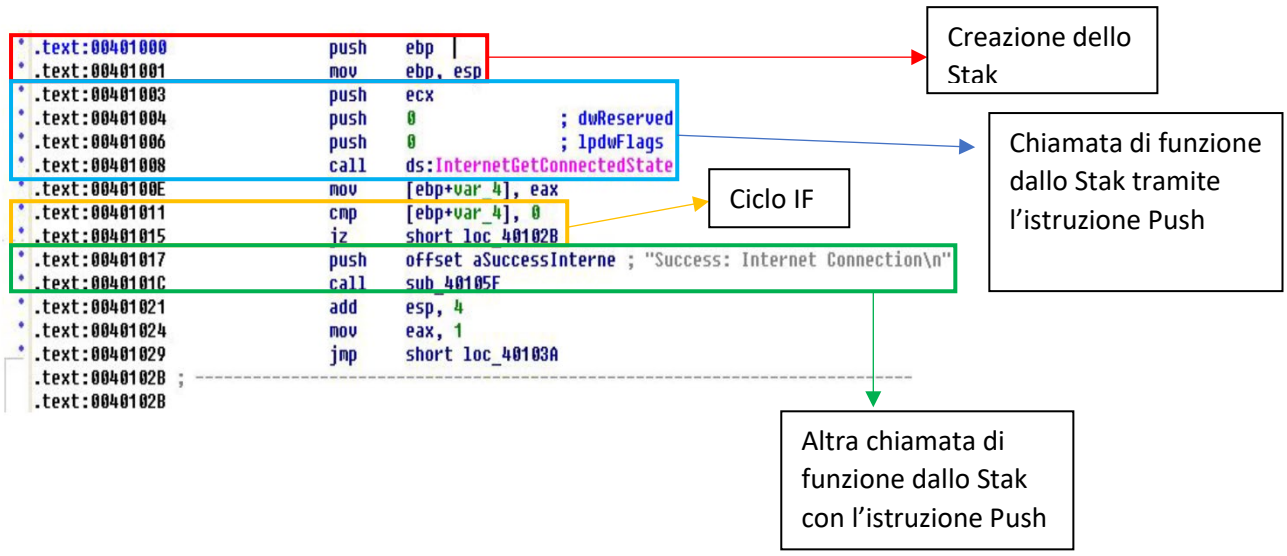
Gabriele Di giampietro **Esercitazione 12/01/2023**

Obiettivo: Nell'estratto del codice seguente in Assembly di un Malware, Identificare i costrutti noti (es. while, if, switch, ecc..) e ipotizzare la funzionalità ovvero esecuzione ad alto livello.

```
.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
```

Svolgimento:

Nell'analisi del nostro codice abbiamo individuato 4 costrutti e le relative funzionalità.



Analizzando il codice del Malware possiamo dedurre che la funzionalità implementata è la chiamata di funzione **internetgetconnectedstate**, ovvero un controllo di connessione, di fatti attraverso il **ciclo if** ne controlla il valore di ritorno. Se il valore di ritorno (return) della funzione è diverso da 0 allora significa che la connessione è andata a buon fine.

Possiamo vedere infatti che l'ultima chiamata di funzione l'istruzione Push inserisce nello Stak di memoria: "Success: Internet connection" e poi richiamato dall'istruzione Call.