

REPORT 06.07.2023

ASSEMBLY

```
.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  IF
.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
```

Viene effettuata una chiamata alla funzione InternetGetConnectedState() per controllare lo stato della connessione a Internet. I risultati della chiamata vengono assegnati a var_4. Viene eseguito un confronto per verificare se var_4 è uguale a zero. Se la condizione è verificata, viene stampato il messaggio "Success: Internet Connection".

push ebp

Salva il valore di ebp (base point register) nello stack;

mov ebp, esp

Copia il valore esp (stack point register) nello stack di ebp;

push ecx

Inserisce il valore di ecx (extended counter register) nello stack;

push 0 ; dwReserved

Inserisce il valore 0 nello stack;

push 0 ; lpdwFlags

Inserisce il valore 0 nello stack;

call ds:InternetGetConnectedState

Chiama la funzione "InternetGetConnectedState" situata nella sezione di dati (data segment);

mov [ebp+var_4], eax

Copia il valore contenuto nel registro eax nella variabile locale [ebp+var_4];

cmp [ebp+var_4], 0

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

jz short loc_40102B

Effettua un salto condizionale, jz (jump if zero) e il suffisso "short" indica che si tratta di un salto a breve raggio, dunque effettua il salto su loc_40102B se l'operazione precedente ha prodotto un risultato zero;

```
push    offset aSuccessInterne ; "Success: Internet Connection\n"
```

"Pusha"/Esegue un push dell'offset dell'etichetta "aSuccessInterne" nello stack.

```
call    sub_40105F
```

Effettua una chiamata alla subroutine sub_40105F, se la condizione precedente risulta vera

```
add     esp, 4
```

Questo sposta l'indirizzo di cima dello stack verso l'alto, "liberando" uno spazio di stack di 4 byte. Questa istruzione è spesso utilizzata per ripristinare lo stack dopo aver utilizzato dati temporanei o parametri di una funzione;

```
mov     eax, 1
```

Imposta il valore 1 al nuovo registro eax;

```
jmp     short loc_40103A
```

Effettua un salto incondizionato all'indirizzo loc_40103A;

