Esercizio S10-L4

L’esercizio di oggi prevedeva di illustrare e fornire una descrizione ad un codice di un Malware in 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

.text:00401017 push offset aSuccessInterne ; “Success: Internet Connection\n”

.text:0040101C call sub_40105F

.text:00401021 add esp, 4

.text:00401024 inc eax, 1

.text:00401029 jmp short loc_40103°

1. Viene salvato il valore corrente di EBP nello stack.
2. Viene copiato il valore del corrente ESP nel registro EBP.
3. Viene salvato il valore di ECX nello stack.
4. Inserisce valore 0 nello stack
5. Viene chiamata la funzione “Internet…” dalla libreria di sistema.
6. Copiato il valore della chiamata precedente nella variabile locale [ebp+var_4]
7. Compara il valore di [ebp+var_4] con 0
8. Salta a loc_40102B se la comparazione è uguale a zero (zero flag impostato)
9. Inserisce l’indirizzo della stringa “Success: Internet Connection\n” nello stack.
10. Funzione responsabile alla stampa della stringa.
11. Libera spazio nello stack occupato dai parametri della chiama funzione.
12. Incrementa il valore EAX di 1
13. Salto incondizionato a loc_40103

Le prime 2 righe sono la costruzione dello stack.

La riga 8 e 9 restituiscono quella che nel linguaggio C potrebbe essere segnata come condizione IF.