S10/L4

EPICODE

Esercizio
Linguaggio Assembly

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
```

Creazione Stack

Push ebp
Mov ebp, esp

Le prime righe del codice si utilizzano per creare uno stack per le variabili locali, si nota dei puntatori EBP (extended base pointer) ed ESP (extended stack pointer) che puntano alla base ed alla cima dello stack.
Non viene definito lo spazio dedicato alle variabili.

STACK: è una memoria che viene usata per le variabili locali e I parametri delle fuzioni. E' molto utile per l'analisi dei malware, immaginiamo di avere una pila di piatti dove si possono aggiungere o rimuovere un piatto dalla cima secondo la struttura LIFO (last in first out).
Quando si aggiunge un piatto si chiama PUSH.
Quando si toglie un piatto si chiama POP.

Funzione internetGetconnected State

- Push ecx
- Push 0 ; dwReversed
- Push 0 ; lpdwFlags
- Call ds: InternetGetConnectedState

Con i primi tre push vengono inserite in cima allo stack tre parametri : ecx, 0 e 0 di variaili , che saranno trasmessi alla funzione InternetGetConnectedState, Verifica che la macchina abbia accesso a internet.

If Statement

- Cmp [ebp+var_4], 0
- Jz short loc_40102B
- Certamente, ecco una versione modificata per evitare il plagio:
- Certamente, ecco una versione modificata per evitare il plagio:

- Nell'analisi del codice sorgente, si osserva innanzitutto l'utilizzo dell'istruzione cmp (compare), impiegata per confrontare la variabile archiviata nel registro EBP+var_4 con il valore zero. Se il risultato di tale confronto è pari a zero, il flag ZF (Zero Flag) viene impostato a 1 e si verifica una condizione di salto (jz) alla locazione di memoria 40102B. In caso contrario, se il risultato fosse diverso da zero, la ZF assume il valore zero ed il programma eseguirebbe le righe di codici successivi , fino ad arrivare al salto non condizionale all'indirizzo di memoria 40103A.

Continuzione...

- Push ebp : L'istruzione viene spinto l'extender base pointer sulla cima dello stack.
- Mov ebp, esp : Viene assegnato il valore dell'extended stack pointer al registro dell'extended base pointer.
- Push ecx: viene posto il valore inserito nel registro ecx in cima allo stack.
- Push 0 ;dwReserved o ipdwFlags: si crea un buffer vuoto di 4 byte sullo stack. Il commento (dwreserved) viene utilizzato alla chiamata successiva, riservato e con valore 0.
- Call ds:InternetgetConnectedState: la funzione viene chiamata e recupera lo stato di connessione del sistema locale.
- mov [ebp+var_4], eax: Viene copiato il valore contenuto nel registro eax nel registro ebp+var_4, a distanza di 4 byte verso la cima dello stack.
- cmp [ebp+var_4], 0 : Viene effettuata una sottrazione tra il parametro contenuto nel registro "ebp+4_var e 0 andando a modificare lo zero flag del registro.
- jz short loc_40102B : con l'istruzione di jump vinee controllata la zero flag ottenuta dalla precedente istruzione cmp, se risulterà uguale a 1 verrà effettuato uno short jump alla memoria 40102b.
- push offset aSuccessInterne ; “Success: Internet Connection\n" Con questa istruzione viene inserita la stringa "aSuccessInterne" in un registro in cima allo stack.
- call sub_40105F Viene chiamata una funzione inserita nell 'indirizzo di memoria "40105F".
- add esp, 4 Con questa istruzione viene effettuata una somma tra il valore inserito all'interno del registro "esp".
- mov eax, 1 Viene sostituito il valore contenuto all'interno del registro "eax" con il valore 1.
- jmp short loc_40103A Viene effettuato un salto all'indirizzo di memoria "40103A".