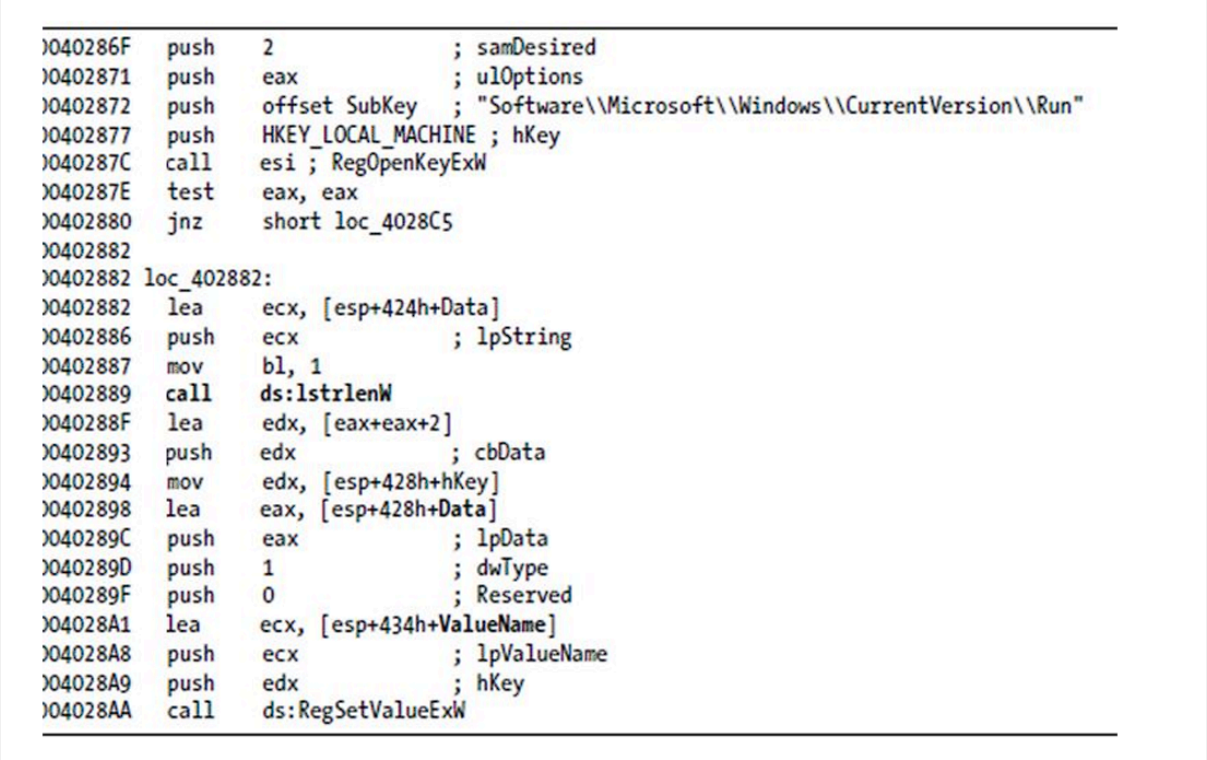
S11L1

Rosario Z.

Traccia:

Con riferimento agli estratti di un malware reale presenti nelle prossime slide, rispondere alle seguenti domande: Descrivere come il malware ottiene la persistenza, evidenziando il codice assembly dove le relative istruzioni e chiamate di funzioni vengono eseguite Identificare il client software utilizzato dal malware per la connessione ad Internet Identificare l'URL al quale il malware tenta di connettersi ed evidenziare la chiamata di funzione che permette al malware di connettersi ad un URL



```
)040286F   push    2               ; samDesired
)0402871   push    eax             ; ulOptions
)0402872   push    offset SubKey   ; "Software\\Microsoft\\Windows\\CurrentVersion\\Run"
)0402877   push    HKEY_LOCAL_MACHINE ; hKey
)040287C   call    esi ; RegOpenKeyExW
)040287E   test    eax, eax
)0402880   jnz     short loc_4028C5
)0402882
)0402882 loc_402882:
)0402882   lea     ecx, [esp+424h+Data]
)0402886   push    ecx             ; lpString
)0402887   mov     bl, 1
)0402889   call    ds:lstrlenW
)040288F   lea     edx, [eax+eax+2]
)0402893   push    edx             ; cbData
)0402894   mov     edx, [esp+428h+hKey]
)0402898   lea     eax, [esp+428h+Data]
)040289C   push    eax             ; lpData
)040289D   push    1               ; dwType
)040289F   push    0               ; Reserved
)04028A1   lea     ecx, [esp+434h+ValueName]
)04028A8   push    ecx             ; lpValueName
)04028A9   push    edx             ; hKey
)04028AA   call    ds:RegSetValueExW
```

L'immagine inviata mostra un codice assembly di un malware. Il codice è scritto in linguaggio assembly x86 e utilizza la convenzione di chiamata standard di Windows.

Analisi del codice

Il codice inizia con la funzione sub_401000. Questa funzione esegue le seguenti operazioni:

Push ebp: Salva il valore del registro ebp sullo stack.
Mov ebp, esp: Imposta il registro ebp su esp.
Push ecx: Salva il valore del registro ecx sullo stack.

Push 1pdwFlags: Salva il valore della variabile 1pdwFlags sullo stack.
Call ds:InternetGetConnectedState: Chiama la funzione InternetGetConnectedState per controllare se il computer è connesso a Internet.
Mov [ebp+var_4], eax: Memorizza il valore restituito dalla funzione InternetGetConnectedState nella variabile [ebp+var_4].
Cmp [ebp+var_4], 8: Confronta il valore della variabile [ebp+var_4] con 8.
Jz short loc_401028: Se il valore della variabile [ebp+var_4] è uguale a 8, salta alla posizione loc_401028.
Push offset aSuccessInterne: Salva l'indirizzo della stringa "Success: Internet Connection\n" sullo stack.
Call sub_40117F: Chiama la funzione sub_40117F per stampare la stringa "Success: Internet Connection\n".
Jmp short loc_40103A: Salta alla posizione loc_40103A.
Loc_401028: Push offset aError1: Salva l'indirizzo della stringa "Error 1.1: No Internet\n" sullo stack.
Add esp, 4: Incrementa il valore del registro esp di 4.
Push 1NoInte: Salva il valore 1 sullo stack.
Call sub_40117F: Chiama la funzione sub_40117F per stampare la stringa "Error 1.1: No Internet\n".
Add esp, 4: Incrementa il valore del registro esp di 4.
Jmp short loc_40103A: Salta alla posizione loc_40103A.
Loc_40103A:
Mov esp, ebp: Ripristina il valore del registro esp da ebp.
Pop ebp: Ripristina il valore del registro ebp dallo stack.
Retn: Restituisce il controllo al chiamante.
Azioni dannose

InternetGetConnectedState controlla se il computer è connesso a Internet