

Analisi codice Assembly di un malware

1) Con riferimento al codice di seguito, descrivere come il malware ottiene la persistenza, evidenziando il codice assembly dove le relative istruzioni e chiamate di funzioni vengono eseguite

```

0040286F  push     2             ; samDesired
00402871  push     eax           ; ulOptions
00402872  push     offset SubKey ; "Software\\Microsoft\\Windows\\CurrentVersion\\Run"
00402877  push     HKEY_LOCAL_MACHINE ; hKey
0040287C  call     esi           ; RegOpenKeyExW
0040287E  test     eax, eax
00402880  jnz      short loc_4028C5
00402882
00402882  loc_402882:
00402882  lea      ecx, [esp+424h+Data]
00402886  push     ecx           ; lpString
00402887  mov      bl, 1
00402889  call     ds:strlenW
0040288F  lea      edx, [eax+eax+2]
00402893  push     edx           ; cbData
00402894  mov      edx, [esp+428h+hKey]
00402898  lea      eax, [esp+428h+Data]
0040289C  push     eax           ; lpData
0040289D  push     1             ; dwType
0040289F  push     0             ; Reserved
004028A1  lea      ecx, [esp+434h+ValueName]
004028A8  push     ecx           ; lpValueName
004028A9  push     edx           ; hKey
004028AA  call     ds:RegSetValueExW

```

Nel codice evidenziato otiamo 2 chiamate di funzione principali:

- La chiamata alla funzione **RegOpenKeyEx**. Il malware accede alla sottochiave di registro `Software\\Microsoft\\Windows\\CurrentVersion\\Run` prima di modificarne il valore, passando prima i parametri con l'istruzione `push`.

`Software\\Microsoft\\Windows\\CurrentVersion\\Run` è una delle chiavi di registro che viene utilizzata dai malware per ottenere persistenza su un sistema operativo Windows.

- La chiamata alla funzione **RegSetValueEx**. I parametri sono passati sullo stack tramite le istruzioni «`push ecx`» e «`push edx`». La funzione viene utilizzata dal malware per modificare il valore del registro ed aggiungere una nuova voce in modo tale da ottenere la persistenza all'avvio del sistema operativo.

2) Identificare il client software utilizzato dal malware per la connessione ad Internet

```
.text:00401150 ; SUBROUTINE
.text:00401150
.text:00401150 ; DWORD __stdcall StartAddress(LPVOID)
.text:00401150 StartAddress proc near ; DATA XREF: sub_401040+ECF0
.text:00401150 push esi
.text:00401151 push edi
.text:00401152 push 0 ; dwFlags
.text:00401154 push 0 ; lpszProxyBypass
.text:00401156 push 0 ; lpszProxy
.text:00401158 push 1 ; dwAccessType
.text:0040115A push offset szAgent ; "Internet Explorer 8.0"
.text:0040115F call ds:InternetOpenA
.text:00401165 mov edi, ds:InternetOpenUrlA
.text:00401168 mov esi, eax
.text:0040116D loc_40116D: ; CODE XREF: StartAddress+301j
.text:0040116D push 0 ; dwContext
.text:0040116F push 80000000h ; dwFlags
.text:00401174 push 0 ; dwHeadersLength
.text:00401176 push 0 ; lpszHeaders
.text:00401178 push offset szUrl ; "http://www.malware12.com"
.text:0040117D push esi ; hInternet
.text:0040117E call edi ; InternetOpenUrlA
.text:00401180 jmp short loc_40116D
.text:00401180 StartAddress endp
.text:00401180
```

Come evidenziato nell'immagine, vediamo che il malware chiama la funzione **InternetOpenA**, che permette di stabilire la connessione a Internet. Anche in questo caso i parametri richiesti dalla funzione vengono passati con una serie di istruzioni push. Tra questi, in arancione è evidenziato il parametro **szAgent**, che indica il client software utilizzato dal malware per la connessione, ovvero **Internet Explorer 8.0**.

3) 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

```
.text:00401150 ; SUBROUTINE
.text:00401150
.text:00401150 ; DWORD __stdcall StartAddress(LPVOID)
.text:00401150 StartAddress proc near ; DATA XREF: sub_401040+ECF0
.text:00401150 push esi
.text:00401151 push edi
.text:00401152 push 0 ; dwFlags
.text:00401154 push 0 ; lpszProxyBypass
.text:00401156 push 0 ; lpszProxy
.text:00401158 push 1 ; dwAccessType
.text:0040115A push offset szAgent ; "Internet Explorer 8.0"
.text:0040115F call ds:InternetOpenA
.text:00401165 mov edi, ds:InternetOpenUrlA
.text:00401168 mov esi, eax
.text:0040116D loc_40116D: ; CODE XREF: StartAddress+301j
.text:0040116D push 0 ; dwContext
.text:0040116F push 80000000h ; dwFlags
.text:00401174 push 0 ; dwHeadersLength
.text:00401176 push 0 ; lpszHeaders
.text:00401178 push offset szUrl ; "http://www.malware12.com"
.text:0040117D push esi ; hInternet
.text:0040117E call edi ; InternetOpenUrlA
.text:00401180 jmp short loc_40116D
.text:00401180 StartAddress endp
.text:00401180
```

Come evidenziato nell'immagine, vediamo che il malware chiama la funzione **InternetOpenUrlA**, che permette di stabilire la connessione ad uno specifico URL Internet. Anche in questo caso i parametri richiesti dalla funzione vengono passati con una serie di istruzioni push. Tra questi, in arancione è evidenziato il parametro **szUrl** che specifica l'URL al quale il malware tenta di connettersi, ovvero **http://www.malware12.com**.