Analisi Malware Avanzata Statica

La traccia di oggi ci chiedeva di analizzare il codice assebly del seguente malware:

```
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:lstrlenW
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
```

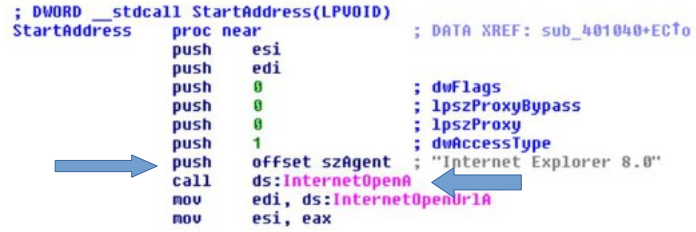
1) La prima richiesta era quella di controllare come e dove il malware ottiene la persistenza .

Come si può vedere alle prime righe di codice c'è una **call esi : RegOpenKeyExW**
Che aprira la chiave che gli ha passato sopra.
Nel secondo riquadro poi possiamo vedere come il malware crea con **call ds:RegSetValueExW** un dato.

2) Nel secondo punto ci veniva poi chiesto il client software che utilizza il malware per connettersi ad internet.

```
; DWORD __stdcall StartAddress(LPVOID)
StartAddress    proc near               ; DATA XREF: sub_401040+ECTo
                push    esi
                push    edi
                push    0               ; dwFlags
                push    0               ; lpszProxyBypass
                push    0               ; lpszProxy
                push    1               ; dwAccessType
                push    offset szAgent  ; "Internet Explorer 8.0"
                call    ds:InternetOpenA
                mov     edi, ds:InternetOpenUrlA
                mov     esi, eax

loc_40116D:                             ; CODE XREF: StartAddress+30↓j
                push    0               ; dwContext
                push    80000000h       ; dwFlags
                push    0               ; dwHeadersLength
                push    0               ; lpszHeaders
                push    offset szUrl    ; "http://www.malware12COM
                push    esi             ; hInternet
                call    edi ; InternetOpenUrlA
                jmp     short loc_40116D
StartAddress    endp
```



Come si può vedere in questo screen la funzione StartAddress e quella che permette al software poi connettersi ad internet. In particolare le righe evidenziate sono quelle che poi aprono la conessione verso internet con **Internet Explorer** e con la chiamata di funzione a **InternetOpenA.**

3) Il terzo punto ci chiedeva poi di identificare l' URL e la chiamata di funzione che permette al malware di connettersi al URL.

L' URL a qui si prova a connettere si può notare nella riga **push offset szUrl** in particolare nel commento si nota **http:/www.malware12.com** . La funzione chiamante che permette al malwere di conettersi al URL è la riga **call edi; InternetOpenUrlA**

4) BONUS: Significato del comando "**lea**".
Questa istruzione copia l'effettivo valore esadecimale a 16 bit di una etichetta, passata come operando sorgente, nel registro di Offset indicato dall'operando destinazione.