

S11-L1

by Andreoli Michael & Hmich Otman

Task

With reference to the excerpts from a real malware present in the next slides, answer the following questions:

- Describe how the malware gains persistence, highlighting the assembly code where the relevant instructions and function calls are executed.
- Identify the client software used by the malware to connect to the Internet.
- Identify the URL the malware attempts to connect to and highlight the function call that allows the malware to connect to a URL.
- BONUS: What is the meaning and function of the assembly command "lea"

Traccia:

```
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: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
```


Traccia:

```

.text:00401150 ; ##### SUBROUTINE #####
.text:00401150
.text:00401150
.text:00401150 ; DWORD __stdcall StartAddress(LPVOID)
.text:00401150 StartAddress proc near ; DATA XREF: sub_401040+ECF0
.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

```

Il browser utilizzato dal malware è Internet Explorer - versione 8.0.

URL MALWARE

Identifying the URL the malware attempts to connect to and highlighting the function call for connecting to an URL

```

.text:00401160 loc_401160: ; CODE XREF: StartAddress+30↓j
.text:00401160 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

```

Tries to push the URL `http://www.malware12.com` with the next function call **InternetOpenUrlA**

BONUS

Meaning and Functionality of the Assembly Command "lea"

The assembly command "lea" (Load Effective Address) is used to calculate the effective address of a variable or a memory area and load it into a register without accessing or modifying the content of the memory itself. The syntax of the "lea" command varies depending on the specific processor architecture. Here's an example of using the "lea" command in x86 assembly:

```
lea eax, [ebx + 8]
```

This calculates the effective address by taking the base address in **ebx**, and then adding an offset of 8 bytes, the resulting address is loaded into the **eax** register.

