ANALISI STATICA AVANZATA

1. Descrivere come il malware ottiene la persistenza ed evidenziare il pezzo di codice Assembly

Per capire come il malware si insinua nei processi, dobbiamo analizzare 2 parti del codice; la prima passa i parametri in cima allo stack per poi chiamare la funzione RegOpenKeyEx:

```
push    2                   ; samDesired
push    eax                 ; ulOptions
push    offset SubKey       ; "Software\\Microsoft\\Windows\\CurrentVersion\\Run"
push    HKEY_LOCAL_MACHINE ; hKey
call    esi ; RegOpenKeyExW
```

La seconda invece è la chiamata di funzione RegSetValueEx che, grazie al push dei parametri, consente al malware di cambiare i valori ed ottenere la persistenza:

```
)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
```

2. Identificare il client software per la connessione ad Internet

```
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:InternetOpenUrlA
mov     esi, eax
```

szAgent è un puntatore che punta ad una stringa che specifica il nome dell'applicazione o dell'entità che chiama le funzioni WinInet che, come sappiamo, consentono alle applicazioni di interagire con i protocolli FTP e HTTP per accedere alle risorse internet; detto ciò possiamo constatare con certezza che il client software usato dall' applicazione è `Internet Explorer 8.0`.

3. Identificare l’URL al quale il malware tenta di connettersi

Ancora più facile del punto precedente, qui abbiamo già chiaro quale sia l’URL con cui il malware vuole connettersi; è specificato nell’istruzione push offset szUrl ed è `http://www.malware12COM`.

4. Funzionamento dell’istruzione LEA

L’istruzione LEA (Load Effective Address) è usata per mettere un indirizzo di memoria all’interno della destinazione; è simile all’istruzione MOV, con la differenza che quest’ultimo carica il valore ad un indirizzo, mentre LEA è un puntatore all’indirizzo; quest’istruzione torna molto utile per leggere i caratteri di una stringa o i valori di una tabella.