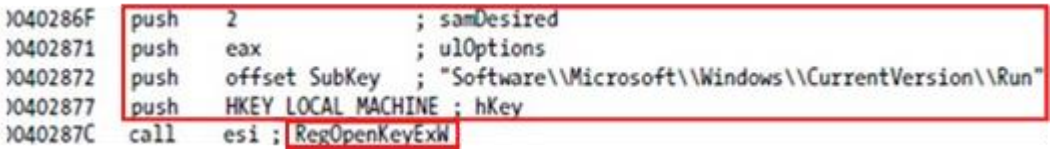
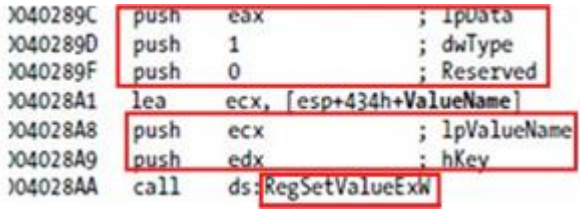
CONSEGNA U3 S11 L1

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



Qui si nota che vengono caricati sullo **stack** tramite l'istruzione **push** i vari parametri **(ulOptions, samDesired, hkey)** necessari alla funzione **RegOpenKeyExW** prima di eseguirla. **SAM** sta per **Security Account Manager**, ed è un precesso di Windows che si occupa di fornire la sicurezza necessaria agli utenti connessi alla macchina, come ad esempio la funzione di storage sicuro delle credenziali. La funzione in questione serve per accedere alle chiavi di registro di windows. In questo caso il programma vuole accedere ai registri **HKEY_LOCAL_MACHINE**, ovvero alle chiavi di registro che contengono le impostazioni di sistema della macchina windows relative a tutti gli utenti.



In questa porzione di codice si vede che il programma sta preparando l'esecuzione della funzione **RegSetValueExW** caricando i vari parametri necessari **(ldData, dwType, lpValueName, hkey)**. Vengono caricati la chiave e anche il valore da inserire. Infatti questa è la funzione che si occupa di scrivere e modificare le chiavi di registro.

Questa procedura serve molto probabilmente per guadagnare la persistenza sul sistema infetto.

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

```
push    offset szAgent  ; "Internet Explorer 8.0"
call    ds:InternetOpenA
```

Il client che viene invocato per la connessione ad internet dal programma malevolo è **Internet Explorer**.

- **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    0               ; dwFlags
push    0               ; lpszProxyBypass
push    0               ; lpszProxy
push    1               ; dwAccessType
push    offset szAgent  ; "Internet Explorer 8.0"
call    ds:InternetOpenA
```

Per prima cosa il malware tenta la connessione tramite la funzione **InternetOpenA**.

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

Per connettersi all'url malevolo "**http://www.malware12.COM"** il programma utilizza la funzione **InternetOpenUrlA**.

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

Sta per **Load Effective Address**. È stata creata per ottimizzare la compilazione di codice Pascal o C. Permette le operazioni tra più di 2 operandi e consente di mantenere inalterati i registri dopo aver eseguito l'operazione a differenza di istruzioni come **add** o **sub**.

LEA EAX, [EAX + EBX + 123]

Inoltre viene spesso usato nei cicli al posto dell'istruzione per l'incremento **inc**, dato che usando l'istruzione **lea** non si vanno a modificare gli **EFLAGS (CF, ZF)** non andando ad alterare il risultato di eventuali istruzioni **cmp**.

INC EAX *diventa* **LEA EAX, [EAX +1]**