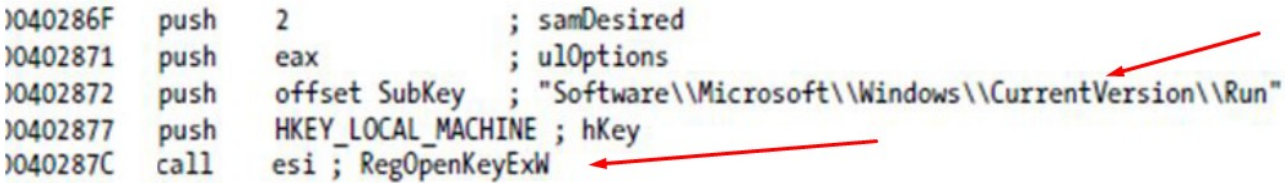
**Alaimo Alessandro**
**16/01/2023**

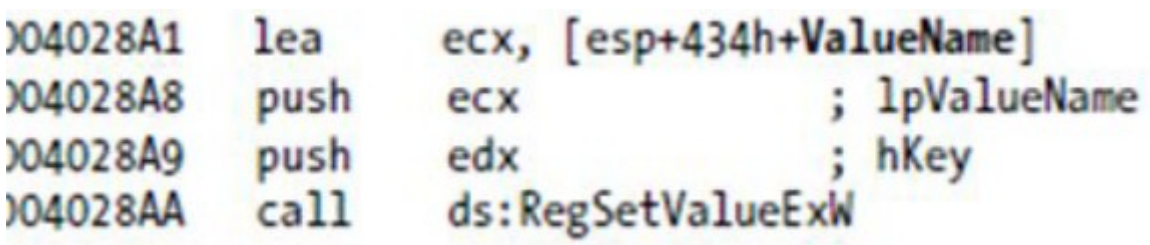
# Report Analisi Statica Avanzata

Come richiesto dall'esercizio, andiamo ad individuare nel codice Assembly come il Malware ottiene la persistenza.

Il primo punto sta nel passare i parametri alla funzione **RegOpenKeyEx** sullo stack tramite il comando push per poi chiamarla con il comando call:

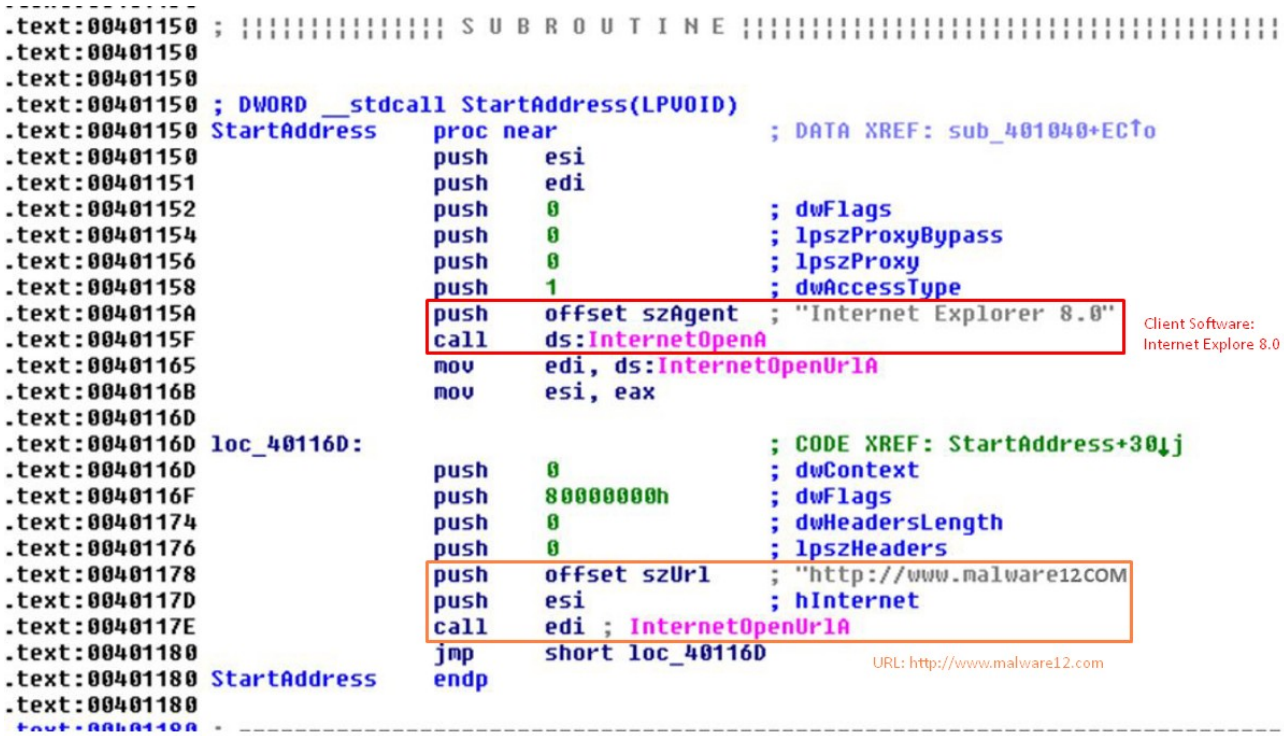



Il secondo punto sta nella chiamata alla funzione **RegSetValueEx** ed anche in questo caso, i valori vengono passati allo stack tramite il comando ***push ecx*** e ***push edx*** ed il Malware in questione modifica il valore del registro ed aggiungere una nuova entry in modo tale da avere accesso persistente all'avvio del sistema operativo:




Possiamo infine dire che nel registro Software\\Microsoft\\Windows\\ CurrentVersion\\Run il Malware si insidia per andare a modificare i valori ed ottenere, infine, accesso persistente.

Adesso andiamo ad identificare nella slide 3, nella subroutine data, qual è il client software e l'URL a cui cerca di connettersi.




Come abbiamo visto nei precedenti screen, viene usato un comando chiamato lea, ma cos'è effettivamente? LEA sta per Load Effective Address ed in breve, LEA carica un puntatore all'elemento che si sta indirizzando, a differenza di MOV che carica il valore effettivo a quell'indirizzo. Lo scopo di LEA è quello di consentire l'esecuzione di un calcolo di indirizzo non banale e la memorizzazione del risultato.