

## OLLY

### Traccia:

Fate riferimento al malware: **Malware\_U3\_W3\_L3**, presente all'interno della cartella **Esercizio\_Pratico\_U3\_W3\_L3** sul desktop della macchina virtuale dedicata all'analisi dei malware. Rispondete ai seguenti quesiti utilizzando OllyDBG.

- All'indirizzo 0040106E il Malware effettua una chiamata di funzione alla funzione «CreateProcess». Qual è il valore del parametro «CommandLine» che viene passato sullo **stack**? **(1)**
- Inserite un breakpoint software all'indirizzo 004015A3. Qual è il valore del registro EDX? **(2)** Eseguite a questo punto uno «step-into». Indicate qual è ora il valore del registro EDX **(3)** motivando la risposta **(4)**. Che istruzione è stata eseguita? **(5)**
- Inserite un secondo breakpoint all'indirizzo di memoria 004015AF. Qual è il valore del registro ECX? **(6)** Eseguite un step-into. Qual è ora il valore di ECX? **(7)** Spiegate quale istruzione è stata eseguita **(8)**.
- BONUS: spiegare a grandi linee il funzionamento del malware

Per prima cosa effettuo una scansione dell'hash del **Malware\_U3\_W3\_L3** trovato con il software **CFF Explorer**

43 / 71

43 security vendors and no sandboxes flagged this file as malicious

f153dfacec09dd69809c3bbf68270a38ee3701f44220c7bf181c14a68c138133

Size: 24.00 KB | Last Analysis Date: 13 days ago

Lab09-02.exe

peexe, idle, armadillo, checks-user-input

Community Score

DETECTION | DETAILS | RELATIONS | BEHAVIOR | COMMUNITY 9

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label: trojan.genericcrlt/meanvzc | Threat categories: trojan | Family labels: genericcrlt, meanvzc, r002c0pik20

1) A questo indirizzo 0040106E il malware effettua una chiamata di funzione alla funzione «CreateProcess»

```
PUSH EDX
LEA EAX, DWORD PTR SS:[EBP-58]
PUSH EAX
PUSH 0
PUSH 0
PUSH 0
PUSH 1
PUSH 0
PUSH 0
PUSH 0
PUSH Malware_.00405030
PUSH 0
CALL DWORD PTR DS:[&KERNEL32.CreateProcessA]
```

```
pProcessInfo
pStartupInfo
CurrentDir = NULL
Environment = NULL
CreationFlags = 0
InheritHandles = TRUE
pThreadSecurity = NULL
pProcessSecurity = NULL
CommandLine = "cmd"
ModuleFileName = NULL
CreateProcessA
```

- Il valore di CommandLine è «CMD» che viene passato sullo stack utilizzando l'indirizzo 0040106E

2) Inserendo un BreakPoint software all'indirizzo 004015A3, qual'è il valore del registro EAX ?

- Una volta inserito il BreakPoint sull'indirizzo 004015A3 eseguo uno Step Into e avvio

00401571	> 8BC7	MOV EAX,EDI	
00401573	> FC	CLD	
00401574	. 5F	POP EDI	
00401575	. C9	LEAVE	
00401576	. C3	RETN	
00401577	. 55	PUSH EBP	
00401578	. 8BEC	MOV EBP,ESP	
0040157A	. 6A FF	PUSH -1	
0040157C	. 68 C0404000	PUSH Malware_.004040C0	
00401581	. 68 3C204000	PUSH Malware_.0040203C	
00401586	. 64:A1 00000000	MOV EAX,DWORD PTR FS:[0]	SE handler installation
0040158C	. 50	PUSH EAX	
0040158D	. 64:8925 00000000	MOV DWORD PTR FS:[0],ESP	
00401594	. 83EC 10	SUB ESP,10	
00401597	. 53	PUSH EBX	
00401598	. 56	PUSH ESI	
00401599	. 57	PUSH EDI	
0040159A	. 8965 E8	MOV DWORD PTR SS:[EBP-18],ESP	
0040159D	. FF15 30404000	CALL DWORD PTR DS:[<&KERNEL32.GetVersion	kernel32.GetVersion
004015A3	. 33D2	XOR EDX,EDX	

  

Registers (FPU)	
EAX	0A280105
ECX	7FFD6000
EDX	00000A28
EBX	7FFD6000
ESP	0012FF94
EBP	0012FFC0
ESI	FFFFFFFF
EDI	7C910208 ntdll.7C910208
EIP	004015A3 Malware_.004015A3
C 0	ES 0023 32bit 0(FFFFFFFF)
P 1	CS 001B 32bit 0(FFFFFFFF)
A 0	SS 0023 32bit 0(FFFFFFFF)
Z 0	DS 0023 32bit 0(FFFFFFFF)
S 0	FS 003B 32bit 7FFDF000(FFF)
T 0	GS 0000 NULL
D 0	
O 0	LastErr ERROR_INVALID_HANDLE (0000)

- Il valore iniziale di EDX è A28 (esadecimale)
- Eseguendo un'altro step into il valore di EDX diventerà 0 perchè l'istruzione precedente XOR a pulito lo stack

→ **EDX 00000000 è 0 in decimale**

XOR =

1	1	=	0
0	0	=	0
1	0	=	1
0	1	=	1

00401571	> 8BC7	MOV EAX,EDI	
00401573	> FC	CLD	
00401574	. 5F	POP EDI	
00401575	. C9	LEAVE	
00401576	. C3	RETN	
00401577	. 55	PUSH EBP	
00401578	. 8BEC	MOV EBP,ESP	
0040157A	. 6A FF	PUSH -1	
0040157C	. 68 C0404000	PUSH Malware_.004040C0	
00401581	. 68 3C204000	PUSH Malware_.0040203C	
00401586	. 64:A1 00000000	MOV EAX,DWORD PTR FS:[0]	SE handler installation
0040158C	. 50	PUSH EAX	
0040158D	. 64:8925 00000000	MOV DWORD PTR FS:[0],ESP	
00401594	. 83EC 10	SUB ESP,10	
00401597	. 53	PUSH EBX	
00401598	. 56	PUSH ESI	
00401599	. 57	PUSH EDI	
0040159A	. 8965 E8	MOV DWORD PTR SS:[EBP-18],ESP	
0040159D	. FF15 30404000	CALL DWORD PTR DS:[<&KERNEL32.GetVersion	kernel32.GetVersion
004015A3	. 33D2	XOR EDX,EDX	
004015A5	. 8AD4	MOV DL,AH	

  

Registers (FPU)	
EAX	0A280105
ECX	7FFD6000
EDX	00000000
EBX	7FFD6000
ESP	0012FF94
EBP	0012FFC0
ESI	FFFFFFFF
EDI	7C910208 ntdll.7C910208
EIP	004015A5 Malware_.004015A5
C 0	ES 0023 32bit 0(FFFFFFFF)
P 1	CS 001B 32bit 0(FFFFFFFF)
A 0	SS 0023 32bit 0(FFFFFFFF)
Z 1	DS 0023 32bit 0(FFFFFFFF)
S 0	FS 003B 32bit 7FFDF000(FFF)
T 0	GS 0000 NULL
D 0	
O 0	LastErr ERROR_INVALID_HANDLE (0000)
EFL	00000246 (NO,NB,E,BE,NS,PE,GE,LE)

### 3) Motiva la risposta

Viene eseguita l'istruzione XOR tra il registro e se stesso impostando il valore a 0

Inserite un secondo breakpoint all'indirizzo di memoria 004015AF.

Qual è il valore del registro ECX?

Eseguite un step-into.

Qual è ora il valore di ECX ? Spiegate quale istruzione è stata eseguita

00401577	55	PUSH EBP			
00401578	8BEC	MOV EBP,ESP			
0040157A	6A FF	PUSH -1			
0040157C	68 C0404000	PUSH Malware_.004040C0			
00401581	68 3C204000	PUSH Malware_.0040203C			
00401586	64:A1 00000000	MOV EAX,DWORD PTR FS:[0]	SE handler installation		
0040158C	50	PUSH EAX			
0040158D	64:8925 00000000	MOV DWORD PTR FS:[0],ESP			
00401594	83EC 10	SUB ESP,10			
00401597	53	PUSH EBX			
00401598	56	PUSH ESI			
00401599	57	PUSH EDI			
0040159A	8965 E8	MOV DWORD PTR SS:[EBP-18],ESP			
0040159D	FF15 30404000	CALL DWORD PTR DS:[<&KERNEL32.GetVersion	kernel32.GetVersion		
004015A3	33D2	XOR EDX,EDX			
004015A5	8AD4	MOV DL,AH			
004015A7	8915 D4524000	MOV DWORD PTR DS:[4052D4],EDX			
004015AD	8BC8	MOV ECX,EAX			
004015AF	81E1 FF000000	AND ECX,0FF			

Registers (FPU)  
EAX 0A280105  
ECX 0A280105  
EDX 00000001  
EBX 7FFDE000  
ESP 0012FF94  
EBP 0012FFC0  
ESI FFFFFFFF  
EDI 7C910208 ntdll.7C910208  
EIP 004015AF Malware\_.004015AF  
C 0 ES 0023 32bit 0(FFFFFFFF)  
P 1 CS 001B 32bit 0(FFFFFFFF)  
A 0 SS 0023 32bit 0(FFFFFFFF)  
Z 1 DS 0023 32bit 0(FFFFFFFF)  
S 0 FS 003B 32bit 7FFDD000(FFF)  
T 0 GS 0000 NULL  
D 0  
I 0 LastErr ERROR\_INVALID\_HANDLE (0000)

- Il valore del registro ECX è 0A280105

Dopo aver eseguito lo Step Into il valore di EDX è 5

00401577	55	PUSH EBP			
00401578	8BEC	MOV EBP,ESP			
0040157A	6A FF	PUSH -1			
0040157C	68 C0404000	PUSH Malware_.004040C0			
00401581	68 3C204000	PUSH Malware_.0040203C			
00401586	64:A1 00000000	MOV EAX,DWORD PTR FS:[0]	SE handler installation		
0040158C	50	PUSH EAX			
0040158D	64:8925 00000000	MOV DWORD PTR FS:[0],ESP			
00401594	83EC 10	SUB ESP,10			
00401597	53	PUSH EBX			
00401598	56	PUSH ESI			
00401599	57	PUSH EDI			
0040159A	8965 E8	MOV DWORD PTR SS:[EBP-18],ESP			
0040159D	FF15 30404000	CALL DWORD PTR DS:[<&KERNEL32.GetVersion	kernel32.GetVersion		
004015A3	33D2	XOR EDX,EDX			
004015A5	8AD4	MOV DL,AH			
004015A7	8915 D4524000	MOV DWORD PTR DS:[4052D4],EDX			
004015AD	8BC8	MOV ECX,EAX			
004015AF	81E1 FF000000	AND ECX,0FF			
004015B5	890D D0524000	MOV DWORD PTR DS:[4052D0],ECX			

Registers (FPU)  
EAX 0A280105  
ECX 00000005  
EDX 00000001  
EBX 7FFDD000  
ESP 0012FF94  
EBP 0012FFC0  
ESI FFFFFFFF  
EDI 7C910208 ntdll.7C910208  
EIP 004015B5 Malware\_.004015B5  
C 0 ES 0023 32bit 0(FFFFFFFF)  
P 1 CS 001B 32bit 0(FFFFFFFF)  
A 0 SS 0023 32bit 0(FFFFFFFF)  
Z 0 DS 0023 32bit 0(FFFFFFFF)  
S 0 FS 003B 32bit 7FFDD000(FFF)  
T 0 GS 0000 NULL  
D 0  
I 0 LastErr ERROR\_INVALID\_HANDLE (0000)

L'istruzione che è stata eseguita è AND. Confrontando i bit dei due operandi.

Se entrambi i bit corrispondenti sono impostati a 1, il risultato sarà 1. Altrimenti, il risultato sarà 0.

### Funzionamento Malware

Possiamo vedere le librerie importate in automatico da Virustotal altrimenti da CFF Explorer

Kernel32.dll

WS2\_32.dll

**Kernel32.dll** → e' una libreria di sistema di Windows che serve per interagire con il sistema operativo. Gestisce processi, thread (flussi di esecuzione), memoria, file, tempo e risorse. Essenziale per molti programmi consentendo di eseguire operazioni tipo creare processi, leggere/scrivere file, allocare memoria e gestire il tempo di sistema.

**WS2\_32.dll** per la programmazione del socket e le comunicazioni di rete.

Possiamo ipotizzare l'implementazione di una backdoor utilizzando:

**WSASocketA** per creare un socket di rete

**Connect** per connettersi al server remoto e inviare o ricevere comandi e dati tramite questa connessione.

**Il malware in questione è una backdoor**