

u03w11d02 - Malware Disassembling with IDA

Lo scopo dell'esercizio di oggi & di acquisire esperienza con IDA, un tool fondamentale per l'analisi statica.

Consegna: con riferimento al malware chiamato «Malware_U3_W3_L2.dll» presente all'interno della cartella «Esercizio_Pratico_U3_W3_L2» sul Desktop della macchina virtuale dedicata all'analisi dei malware, rispondere ai seguenti quesiti, utilizzando IDA Pro.

1. Qual è l'indirizzo della funzione DLLMain (in esadecimale)?
2. Dalla scheda «imports», individuare la funzione «gethostbyname». Qual è l'indirizzo dell'import?
3. Quante sono le variabili locali della funzione alla locazione di memoria 0x10001656?
4. Quanti sono i parametri della funzione sopra?
5. Inserire altre considerazioni sul comportamento macro del malware.

1. Dalla "Names Window" è necessario ordinare le risorse per nome, scrollare a "DLLMain", 2-click. La libreria viene identificata nel riquadro "View-A". Usando <SPAZIO>, la visuale cambia dal diagramma di flusso alla visualizzazione testuale del codice.

DLLMain risulta allocata all'indirizzo di memoria 1000D02E.

2. Dalla scheda "Imports" è necessario ordinare le risorse per nome, scrollare a "gethostbyname" (i nomi vengono ordinati per maiuscole e minuscole, quindi la funzione, tutta minuscola, si troverà più sotto), 2-click. Si apre la finestra "View-A" in formato testuale, e la riga risulta associata all'indirizzo di memoria 1000163C.

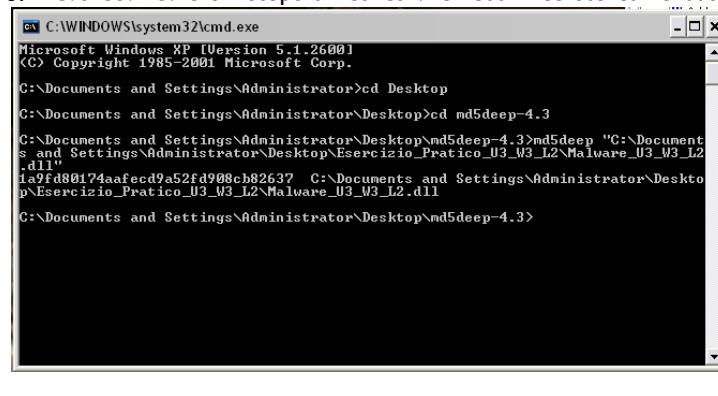
```
.text:10001651
.text:10001656
.text:10001656 ; SUBROUTINE
.text:10001656
.text:10001656 ; DWORD __stdcall sub_10001656(LPVOID)
.text:10001656 sub_10001656 proc near ; DATA XREF: DLLMain(x,x,x)+C810
.text:10001656
.text:10001656 var_675 = byte ptr -675h
.text:10001656 var_674 = dword ptr -674h
.text:10001656 hModule = dword ptr -670h
.text:10001656 timeout = timeval ptr -66Ch
.text:10001656 name = sockaddr ptr -664h
.text:10001656 var_654 = word ptr -654h
.text:10001656 in = in_addr ptr -650h
.text:10001656 Parameter = byte ptr -644h
.text:10001656 CommandLine = byte ptr -63Fh
.text:10001656 Data = byte ptr -638h
.text:10001656 var_544 = dword ptr -544h
.text:10001656 var_50C = dword ptr -50Ch
.text:10001656 var_500 = dword ptr -500h
.text:10001656 var_4FC = dword ptr -4FCh
.text:10001656 readfds = fd_set ptr -4BCh
.text:10001656 phkResult = HKEY__ ptr -3B8h
.text:10001656 var_3B0 = dword ptr -3B0h
.text:10001656 var_1A4 = dword ptr -1A4h
.text:10001656 var_194 = dword ptr -194h
.text:10001656 WSADATA = WSADATA ptr -190h
.text:10001656 arg_0 = dword ptr 4
.text:10001656
.text:10001656 sub esp, 678h
.text:10001656 push ebx
.text:1000165C
```

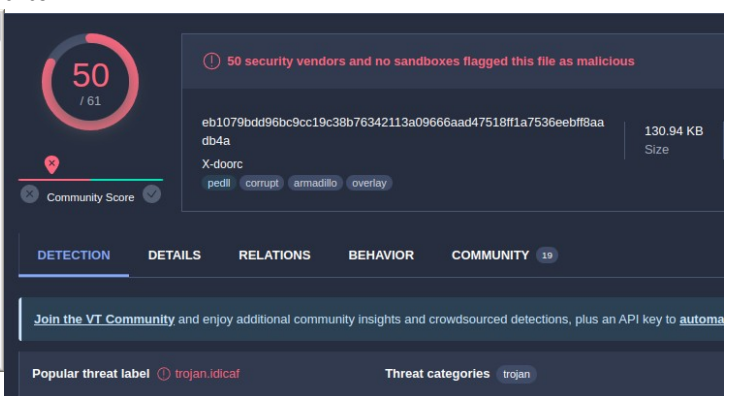
3. Le variabili sono 20:

10001656	var_675	= byte ptr -675h
10001656	var_674	= dword ptr -674h
10001656	hmodule	= dword ptr -670h
10001656	timeout	= timeval ptr -66Ch
10001656	name	= sockaddr ptr -664h
10001656	var_654	= word ptr -654h
10001656	in	= in_addr ptr -650h
10001656	Parameter	= byte ptr -644h
10001656	CommandLine	= byte ptr -63Fh
10001656	Data	= byte ptr -638h
10001656	var_544	= dword ptr -544h
10001656	var_50C	= dword ptr -50Ch
10001656	var_500	= dword ptr -500h
10001656	var_4FC	= dword ptr -4FCh
10001656	readfds	= fd_set ptr -4BCh
10001656	phkResult	= HKEY__ ptr -3B8h
10001656	var_3B0	= dword ptr -3B0h
10001656	var_1A4	= dword ptr -1A4h
10001656	var_194	= dword ptr -194h
10001656	WSADATA	= WSADATA ptr -190h

4. il paramentro è uno soltanto: "10001656 arg_0 = dword ptr 4"

5. Analisi del malware - recupero md5hash: 1a9fd80174aafe9cd9a52fd908cb82637





VirusTotal lo marca come Trojan/Backdoor.

A ricerca "instal" sul codice estratto da IDA, appare che il malware è capace di comprendere se si trova in un ambiente virtuale.

A ricerca "start" è possibile vedere che il malware lancia i seguenti servizi:

Address	Instruction
.text:100016A4	call ds:WSAStartup
.text:100016AF	push offset aWsaStartupError ; "WSAStartup() error: %d\n"
.text:100016E7	push offset ProcName ; "Plug_KeyLog_Restart"
.text:10001C40	push offset aStartxcmd ; "startxcmd"
.text:10001C52	push offset StartAddress ; lpStartAddress
.text:10001C68	push offset aStartxscreen ; "startxscreen"
.text:10001C8A	push offset aStartxfile ; "startxfile"
.text:10001CA9	push offset aStartxreg ; "startxreg"
.text:10001CC8	push offset aStartxvideo ; "startxvideo"
.text:10001CF3	push offset aStartxsound ; "startxsound"
.text:10001D1E	push offset aStartxprocess ; "startxprocess"
.text:10001D40	push offset aStartxservices ; "startxservices"
.text:10001D98	push offset sub_1000399A ; lpStartAddress
.text:100023D1	call ds:WSAStartup
.text:10003168	call ds:WSAStartup
.text:10003BF9	call ds:ExitWindowsEx ; Logoff/Restart/Shut down

È quindi possibile che apra una shell e consenta all'attaccante di registrare audio/video dal PC bersaglio, prendere screenshots, e gestire i processi in uso.

A ricerca "backdoor", il file è abbastanza esplicito:

```
IDA View-A  Hex View-A  Exports  Imports  Names  Functions  Strings  Structures  Enums  Occurences of: backdoor

xdoors_d:10093D50      db '(1) Enter Current Directory ',27h,'%s',27h,0
* xdoors_d:10093D73      align 4
* xdoors_d:10093D74 ; char aBackdoorServer[]
xdoors_d:10093D74 aBackdoorServer db 0Dh,0Ah ; DATA XREF: sub_100042DB+B5↑o
xdoors_d:10093D74      db 0Dh,0Ah
xdoors_d:10093D74      db '*****',0Dh,0Ah
xdoors_d:10093D74      db '[BackDoor Server Update Setup]',0Dh,0Ah
xdoors_d:10093D74      db '*****',0Dh,0Ah
xdoors_d:10093D74      db 0Dh,0Ah,0
* xdoors_d:10093DDB      align 4
* xdoors_d:10093DDC ; char aWarn[]
```