

Analisi statica avanzata con IDA

Ernesto Robles

Con l'uso di IDA è stata effettuata un'analisi statica avanzata come dall'immagine qui sotto raffigurata possiamo vedere come il tool abbia riconosciuto la funzione MAIN nel codice malware.

```
.text:100002E ; D00L __stdcall DllMain(HINSTANCE hinstDLL,DWORD fdwReason,LPVOID lpReserved)
.text:100002E ; DllMain@12 proc near ; CODE XREF: DllEntryPoint+40jp
.text:100002E ; DATA XREF: SUB_100110FF+2Djp
```

F	NullSub_1	10007C
F	NullSub_2	10007C
F	StateXS	10007E
F	HandleProc	1000C5

Nella scheda imports è stato individuato la funzione “gethostbyname” e nelle immagini sottostanti vediamo l'indirizzo della funzione



Dopo aver cliccato sulla funzione veniamo inviati alla sezione che raffigura il codice che la compone da cui è possibile dedurre che gethostbyname è una funzione di libreria standard utilizzata per la risoluzione dei nomi di dominio in indirizzi IP in quanto la “hostent” di solito contiene informazioni come l'indirizzo IP associato al nome host, gli alias del nome host e altri dettagli correlati.

```
.text:10001656 ; struct hostent * stdcall gethostbyname(const char *name)
.text:10001656 ; extrn gethostbyname:duword ; DATA XREF: sub_10001074+loc_1000110Ftr
.text:10001656 ; sub_10001074+1D3Tr
```

F	StateXS	10007E
F	HandleProc	1000C5
F	ServiceMain	1000CF
F	InitialSetup	100007

La funzione alla locazione di memoria 0x10001656 ha 20 variabili locali e 1 parametro come si può vedere dall'immagine



Con l'uso di VirusTotal si controlla il file del malware riuscendo ad avere ulteriori informazioni più precise che ci permettono di fare ulteriori considerazioni attraverso

l'analisi degli import oppure delle funzioni,le immagini seguenti raffigurano funzioni che verificano il tipo di hardware, versioni del software e creano backdoor.

```
.text:1000437C      push     eax                ; lpBuffer
.text:1000437D      push     edi                ; nBuffertLength
.text:1000437E      call     ds:GetCurrentDirectory
.text:10004384      mov     esi, ds:sprintf
.text:1000438A      lea     eax, [ebp+buf]
.text:10004390      push    offset aBackdoorServer ; "\r\n\r\n*****\r\n[Ba"...
.text:10004395      push    eax                ; char *
.text:10004396      call     esi                ; sprintf
.text:10004398      mov     ebx, [ebp+s]
.text:1000439B      lea     eax, [ebp+buf]
.text:100043A1      push    eax                ; buf
```

Name	Addr
PSLIST	1000
multib_1	1000
multib_2	1000
StateExS	1000
HandleProc	1000
ServiceMain	1000
main	1000

```
.text:10004033      mov     [esi+0C0h], eax
.text:10004039      lea     eax, [ebp+var_C]
.text:1000403C      push    eax                ; phidResult
.text:1000403D      push    1                ; sarDesired
.text:1000403F      push    0                ; ulOptions
.text:10004041      push    offset aHardwareDescr1 ; "HARDWARE\\DESCRIPTION\\System\\CentralProc"...
.text:10004046      push    80000002h        ; hKey
.text:10004048      mov     [ebp+var_14], 8
.text:10004052      call    edi                ; RegOpenKeyEx
.text:10004054      lea     eax, [ebp+var_14]
.text:10004057      xor     edi, edi
.text:10004059      push    eax                ; lpcbData
.text:1000405A      lea     eax, [esi+0ACh]
.text:10004060      push    eax                ; lpData
.text:10004061      push    edi                ; lpType
.text:10004062      push    edi                ; lpReserved
.text:10004063      push    offset aHhZ      ; ""HhZ""
.text:10004068      push    [ebp+var_C]        ; hKey
.text:1000406B      call    ebx                ; RegQueryValueEx
.text:1000406D      push    [ebp+var_C]        ; hKey
.text:10004070      call    ds:RegCloseKey
.text:10004076      lea     eax, [ebp+Buffer]
```

Names window

Name	Addr
PSLIST	10007
multib_1	10007
multib_2	10007
StateExS	10007
HandleProc	1000C
ServiceMain	1000C
main	1000C

Line 4 of 764

Strings window

Address	Length	T...	String
00000005	C	vids	
0000000C	C	SHELL32.dll	
00000009	C	DeleteDC	
0000000D	C	DeleteObject	
0000000A	C	GetDlgItem	
0000000F	C	RealizePalette	