

CS0424 S11-L2
ANALISI STATICA CON IDA Pro

Mattia Fossati

20/08/2024

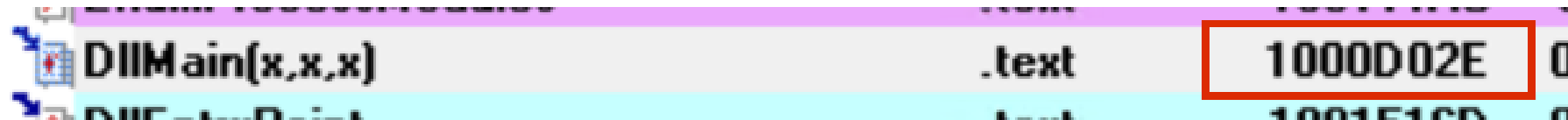
Traccia



Con riferimento al malware chiamato **"Malware_U3_W3_L2"** 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. Individuare l'indirizzo della funzione DLLMain (così com'è, in esadecimale).
2. Dalla scheda "imports" individuare la funzione "gethostbyname". Qual è l'indirizzo dell'import? Cosa fa la funzione?
3. Quante sono le variabili locali della funzione alla locazione di memoria 0x10001656?
4. Quanti sono, invece, i parametri della funzione sopra?
5. Inserire altre considerazioni macro livello sul malware (comportamento)

1. Indirizzo DLLMain

Individuando la funzione DLLMain nella finestra delle funzioni è possibile vedere anche il suo indirizzo in esadecimale (**1000D02E**), a fianco nella sezione "start"

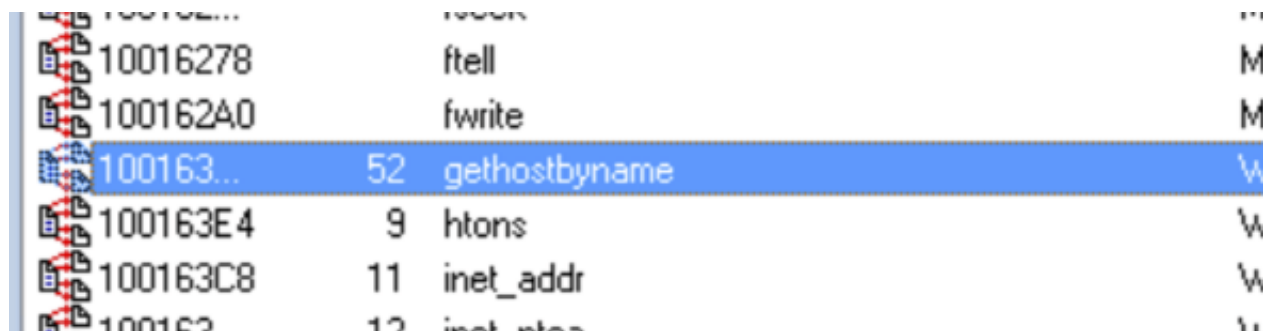


	DLLMain(x,x,x)	.text	1000D02E	0
	DLLMain(x,x,x)	.text	10015100	0

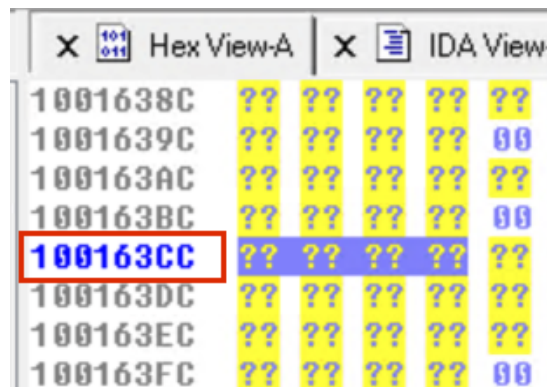
2. Funzione "gethostbyname"

La funzione "gethostbyname" recupera le informazioni host corrispondenti a un nome host da un database host. È una funzione tipica delle librerie di rete in molti linguaggi di programmazione, e viene usata principalmente per ottenere l'indirizzo IP associato a un nome di dominio.

L'indirizzo della funzione gethostbyname del malware si trova all'indirizzo **100163CC**



10016278	ftell	M
100162A0	fwrite	M
100163CC	52 gethostbyname	W
100163E4	9 htons	W
100163C8	11 inet_addr	W
100163...	12 inet_ntoa	W



Hex View-A	IDA View
1001638C	?? ?? ?? ?? ??
1001639C	?? ?? ?? ?? ??
100163AC	?? ?? ?? ?? ??
100163BC	?? ?? ?? ?? ??
100163CC	?? ?? ?? ?? ??
100163DC	?? ?? ?? ?? ??
100163EC	?? ?? ?? ?? ??
100163FC	?? ?? ?? ?? ??

3. Variabili della funzione

La funzione presente alla locazione di memoria 0x10001656 utilizza **23** variabili locali.

```
; DWORD __stdcall sub_10001656(LPVOID)
sub_10001656 proc near

var_675= byte ptr -675h
var_674= dword ptr -674h
hLibModule= dword ptr -670h
timeout= timeval ptr -66Ch
name= sockaddr ptr -664h
var_654= word ptr -654h
Dst= dword ptr -650h
Parameter= byte ptr -644h
var_640= byte ptr -640h
CommandLine= byte ptr -63Fh
Source= byte ptr -63Dh
Data= byte ptr -638h
var_637= byte ptr -637h
var_544= dword ptr -544h
var_50C= dword ptr -50Ch
var_500= dword ptr -500h
Buf2= byte ptr -4FCh
readfds= fd_set ptr -4BCh
phkResult= byte ptr -3B8h
var_3B0= dword ptr -3B0h
var_1A4= dword ptr -1A4h
var_194= dword ptr -194h
WSAData= WSAData ptr -190h
arg_0= dword ptr 4

sub     esp, 678h
push    ebx
push    ebp
push    esi
push    edi
call    sub_10001000
test    eax, eax
jnz     short loc_100016BC

var_675= byte ptr -675h
var_674= dword ptr -674h
hLibModule= dword ptr -670h
timeout= timeval ptr -66Ch
name= sockaddr ptr -664h
var_654= word ptr -654h
Dst= dword ptr -650h
Parameter= byte ptr -644h
var_640= byte ptr -640h
CommandLine= byte ptr -63Fh
Source= byte ptr -63Dh
Data= byte ptr -638h
var_637= byte ptr -637h
var_544= dword ptr -544h
var_50C= dword ptr -50Ch
var_500= dword ptr -500h
Buf2= byte ptr -4FCh
readfds= fd_set ptr -4BCh
phkResult= byte ptr -3B8h
var_3B0= dword ptr -3B0h
var_1A4= dword ptr -1A4h
var_194= dword ptr -194h
WSAData= WSAData ptr -190h
arg_0= dword ptr 4
```

4. Parametri della funzione

La funzione presente alla locazione di memoria 0x10001656 utilizza **1** solo parametro.

```
var_1A4= dword ptr -1A4h  
var_194= dword ptr -194h  
WSAData= WSAData ptr -190h  
arg_0= dword ptr 4  
  
sub     esp, 678h  
push    ebx  
push    ebp  
push    esi  
push    edi  
call    sub_10001000
```

5. Considerazioni sul malware

Andando a vedere tra le librerie importante ce ne sono alcune che consentono la modifica/creazione di chiavi nel registro e di file.

Questo potrebbe essere utilizzato dal malware ad esempio per eseguire una privilege escalation, per ottenere persistenza sul sistema o eseguire codice

Address	Disassembly	Comment
10016000	RegOpenKeyA	ADVAPI32
10016008	RegCloseKey	ADVAPI32
10016038	RegCreateKeyA	ADVAPI32
1001603C	RegDeleteKeyA	ADVAPI32
1001601C	RegDeleteValueA	ADVAPI32
10016020	RegEnumKeyA	ADVAPI32
10016030	RegEnumValueA	ADVAPI32
10016024	RegOpenKeyA	ADVAPI32
10016010	RegOpenKeyExA	ADVAPI32
1001600C	RegQueryValueExA	ADVAPI32
10016018	RegSetValueExA	ADVAPI32
10016040	RegSetKeyValueA	ADVAPI32
10016198	CreateDirectoryA	
1001619C	SetFileAttributesA	
100161A0	GetFileAttributesA	
100161A4	RemoveDirectoryA	
100161A8	MoveFileA	
100161...	GetFileTime	
100161B0	CreateFileA	
100161B4	SetFileTime	
100161B8	TerminateThread	
10016198	CreateDirectoryA	
100160...	WriteFile	
10016160	DeleteFileA	

Analizzando il malware con virustotal viene riconosciuto come un trojan che essenzialmente crea una backdoor e si connette a due IP da cui potrebbe ricevere comandi o a cui potrebbe inviare file locali.

The screenshot shows the VirusTotal analysis interface for a file named 'X-doorc'. The file's SHA-256 hash is 'eb1079bdd96bc9cc19c38b76342113a09666aad47518ff1a7536eebff8aadb4a'. The file size is 130.94 KB, and it was last analyzed 8 days ago. A red circular badge indicates a 'Community Score' of 64 out of 75. A warning message states '64/75 security vendors flagged this file as malicious'. Below the file name, several tags are displayed: 'pedll', 'overlay', 'armadillo', 'corrupt', 'checks-user-input', and 'idle'. The 'RELATIONS' tab is selected, showing a section for 'Contacted IP addresses (2)' with a table of IP addresses, detection counts, autonomous systems, and countries. Below this, there is a section for 'Execution Parents (30)'. A banner at the top of the relations section encourages joining the community for more insights and an API key.

64 / 75
Community Score

64/75 security vendors flagged this file as malicious

Reanalyze Similar More

eb1079bdd96bc9cc19c38b76342113a09666aad47518ff1a7536eebff8aadb4a

X-doorc

Size: 130.94 KB | Last Analysis Date: 8 days ago

pedll overlay armadillo corrupt checks-user-input idle

DETECTION DETAILS RELATIONS BEHAVIOR COMMUNITY 19+

[Join our Community](#) and enjoy additional community insights and crowdsourced detections, plus an API key to [automate checks](#).

Contacted IP addresses (2)

IP	Detections	Autonomous System	Country
172.217.214.94	0 / 93	15169	US
204.79.197.203	0 / 93	8068	US

Execution Parents (30)

Per stabilire le connessioni e comunicare con i server malevoli, il malware utilizza principalmente le funzioni importate: **"gethostbyname"** e **"socket"**

	100163...	52	gethostbyname
---	-----------	----	---------------

	100163F8	23	socket
--	----------	----	--------