Progetto S11L2

Alessandro Moscetti

Dovevamo rispondere ai seguenti quesiti riguardanti il file ‘Malware_U3_W2_L5’:

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)

Punto 1

L’indirizzo della funzione DLLMain è **1000D02E**.

Punto 2

L’indirizzo dell’import della funzione gethostbyname è **100163CC**.
Questa funzione converte un nome host in una struttura di tipo hostent che contiene diverse informazioni sull'host, come indirizzi IP associati, alias, e altri dettagli, in poche parole è utilizzata per ottenere informazioni su un determinato host.

Punto 3

Le variabili locali della funzione alla locazione di memoria 0x10001656 sono le seguenti mostrate in figura.
Per individuarle ci basiamo sull’offset negativo rispetto al registro epb.

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

Punto 4

Il parametro passato alla funzione alla locazione di memoria 0x10001656 è solo arg_0 mostrato in figura.
Per individuare i parametri ci basiamo, al contrario del punto 3, sull'offset positivo rispetto al registro epb.

```
arg_0           = dword ptr  4
```

Punto 5

Analizzando il codice possiamo ipotizzare che il malware sia una backdoor.

```
xdoors_d:10093D74 ; char aBackdoorServer[]
```