

UIC packer Project Killing Neolite 2

Data	by "-Death_Reaver-"	
07/09/2005	<u>UIC's Home Page</u>	Published by Quequero
.....	<i>Dannato Neolite, grazie Death!</i>
....	E-mail: death.reaver@virgilio.it Nick, UIN, canale IRC/EFnet frequentato
Difficolt	()NewBies (*)Intermedio ()Avanzato ()Master	-

Introduzione

Il titolo dice tutto: buona lettura!

Tools usati

[OllyDbg](#)
[Plugins per Olly](#)
[ImportRec](#)
Un qualsiasi PeEditor e HexEditor

URL o FTP del programma

Il programma packato in allegato

[ALLEGATO](#)

Essay

Prima di cominciare vi faccio notare una cosa:

Se andate (sotto XP) in C:\Documents and settings\\SendTo (una cartella nascosta) e ci mettete dentro un collegamento a Olly e/o a IDA etc. potete, facendo click con il destro su un file exe e scegliendo invia a e successivamente OllyDBG, caricare direttamente il file exe dentro olly senza la solita Olly->File->Open Fico no!?

Ora cominciamo:

Prima di tutto caricate il programma in allegato con Olly (potete farlo anche con il metodo sopra citato) e vi ritroverete dinanzi a questo codice:

```
0043809A neoLite.<ModuleEnt> E9 A6000000 JMP 00438145
0043809F AA STOS BYTE PTR ES:[EDI]
004380A0 99 CDQ
004380A1 48 INC EBX
004380A2 0048 80 ADD BYTE PTR DS:[EAX-80],CL
004380A5 43 INC EBX
004380A6 004480 43 ADD BYTE PTR DS:[EAX+EAX*4+43],AL
004380AA 0000 ADD BYTE PTR DS:[EAX],AL
004380AC 0000 ADD BYTE PTR DS:[EAX],AL
004380AE 000A 7900005C ADD BYTE PTR DS:[EDX+80000079],CH
004380B4 8143 00 4E656F44 ADD DWORD PTR DS:[EBX],4C6F654E
004380BB 697465 20 457869 INUL ESI, DWORD PTR SS:[EBP+20],63657845
004380C3 75 74 JNZ SHORT 00438139
004380C5 61 POP EDI
004380C6 626C65 20 BOUND EBP, QWORD PTR SS:[EBP+20]
004380CA 46 INC ESI
004380CB 696C65 20 436F65 INUL EBP, QWORD PTR SS:[EBP+20],706D6F43
```

Come potete vedere la prima istruzione un JMP. Eseguitelo e vi ritroverete qui:

```
00438117 . 31 39 39 37 21 ASCII "1997-1999 Lee Ha"
00438127 . 73 69 75 68 01 ASCII "siuk/ALL Rights"
00438137 . 20 52 65 73 64 ASCII " Reserved.™",0
00438144 . 00 DB 00
00438145 > 8B4424 04 MOV EAX, DWORD PTR SS:[ESP+4]
00438149 . 2305 AB804300 AND EAX, DWORD PTR DS:[4380AB]
0043814F . E8 71030000 CALL 004384C5
00438154 . FE05 44814300 INC BYTE PTR DS:[438144]
0043815A . FF00 JMP EBX
0043815C . 803D 44814300 CMP BYTE PTR DS:[438144],0
00438163 . 75 13 JNZ SHORT 00438178
00438165 . 90 NOP
00438166 . 90 NOP
00438167 . 90 NOP
00438168 . 90 NOP
00438169 . 50 PUSH EAX
0043816A . 2800 SUB EBX, EBX
0043816C . E8 54030000 CALL 004384C5
00438171 . 58 POP EAX
00438172 . FE05 44814300 INC BYTE PTR DS:[438144]
00438178 . C3 RETN
kernel32.77E5EB69
```

La prima cosa che salta all'occhio quella call: non prende parametri a quanto pare, ma prima di essere eseguita il programma fa dei lavoretti con eax. E a quanto si vede dopo, la call restituisce l'indirizzo a cui JMP a 0043815A ci spedisce.

Per: Molti come me sanno che un JMP EAX in un programma packato MOLTO sospetto. Infatti se lo si esegue si ci ritrova a 004012A5 che come potete immaginare il nostro OEP (Naturalmente in tutti i programmi cavia di Que IOEP questo, cambia solo il codice del packer)

Ora che siamo posizionati sull'indirizzo 004012A5 siamo pronti a fare un po' di dumping:

Aprire OllyDump e deselezionare la casella Rebuild Import in basso e Dumpate il processo. ORA NON CHIUDETE OLLY!!!

Sicuramente l'import table disintegrata o come minimo contiene i dati relativi agli import del packer E NON del programma packato. Quindi ora apriamo ImportRec (anche se il lavoro di rebuilding si pu' fare benissimo con un HexEditor*) e per prima cosa scegliamo il processo chiamato in questo caso Neolite.exe. Ora premiamo il pulsante IAT autosearch che far trovare la iat a ImportRec il quale scriverà il relativo indirizzo nel box affianco. Ora che Abbiamo l'indirizzo della IAT Premiamo Get Import e ImportRec Ci mostrerà i suoi risultati. In questo caso troviamo due risultati positivi (Kernel32 e User32). Che culo!. Ok ora possiamo premere il fatidico pulsante Fix Dump che correggerà un dump con i dati trovati. Scegliamo quindi il nostro dump e avremo un bel programma funzionante!!!!

*Guardando il risultato con un PeEditor (o con l'HexEditor) si nota che ImportRec non ha fatto altro che aggiungere una sezione chiamata mackct dove ha posizionato una nuova IT con le sue varie tavole (nomi, thunk etc.).

Volendo fare a mano (che devo dire di soddisfazione ☺) basta fare:

- 1) incrementare il valore di *NumberOfSection* nel FileHeader
- 2) Modificare il valore di *SizeOfImage* nell'Optional Header secondo l'allineamento delle sezioni (se l'allineamento perfetto basta aggiungere 1000)
- 3) Aggiungere una entry nella section table
- 4) Aggiungere X byte a fine file (dove X pari in questo caso a 1000)

In questo modo abbiamo aggiunto una sezione. Ora basta preservarsi 15 Dword: 5 per Kernel32, 5 per User32 e 5 NULL per indicare la fine dell'IT. Quindi scrivere i nomi degli import da qualche parte e fategli puntare i rispettivi membri name delle entry della import table. Poi bisogna armarsi di pazienza e riempire le thunk data puntate degli OFT con gli indirizzi che puntano alle strutture Import_By_Name o azzerare il valore degli OFT nell'IT e riempire la IAT (FT) con gli indirizzi delle funzioni importare.

PS. Se volete sapere i miei gusti, io non mi piace tanto usare sia il PeEditor che ImportRec(anche se a volte necessario ☺). Preferisco un buon HexEditor -Death_Reaver-

Note finali

Ringrazio tutti quelli del forum e della mailing list.

Disclaimer

Vorrei ricordare che il software va comprato e non rubato, dovete registrare il vostro prodotto dopo il periodo di valutazione. Non mi ritengo responsabile per eventuali danni causati al vostro computer determinati dall'uso improprio di questo tutorial. Questo documento stato scritto per invogliare il consumatore a registrare legalmente i propri programmi, e non a fargli fare uso dei tantissimi file crack presenti in rete, infatti tale documento aiuta a comprendere lo sforzo che ogni sviluppatore ha dovuto portare avanti per fornire ai rispettivi consumatori i migliori prodotti possibili.

Reversiamo al solo scopo informativo e per migliorare la nostra conoscenza del linguaggio Assembly e di creare una serie di bombe da lanciare a zia Billa ogni volta che il Word/VisualC++ crasha.

