

Introduction:

هذا هو الدرس الأول من الجزء الثاني من سلسلة تعلم الهندسة العكسية...الجزء الثاني سيكون مخصصا للـ Manual Unpacking أو اختصارا (MUN) . سنبدأ اليوم مع برنامج الضغط الأسهل ألا وهو UPX ...هذه الأحرف هي اختصار لـ Ultimate Packer for eXecutable

لمحة عن الـ packers :

يشير هذا المصطلح الى تلك البرامج التي تضغط الملفات التنفيذية وتقوم بإنتاج ما يسمى بـ decompression stub وهو الكود الذي يضاف الى البرنامج الأصلي حيث يقوم بفك الضغط عن الملف التنفيذي ويبدؤه (to initialize it) بحيث لا يكون هناك فرق - بالنسبة للمستخدم - بين الملف قبل وبعد الضغط . تختلف نسبة الضغط باختلاف الـ packer (أو: compressor) .

بالمقابل ، فان زمن تحميل البرنامج (loading) سيصبح أطول...لأنك لن تقوم بتشغيل البرنامج - الملف التنفيذي - فقط ، بل ستقوم بداية بتشغيل الكود المسئول عن فك الضغط عن الملف التنفيذي (أي الـ stub) الذي بدوره يفك ضغط الملف التنفيذي ويقوم بإبدائه (تشغيله) .

وكما هو واضح ، فالفائدة الأخرى للـ packers هي جعل عملية الـ reverse engineering للبرنامج أكثر صعوبة (بالطبع لن تكون مستحيلة :) . مما يجدر ذكره ، أن جميع الـ packers الموجودين اليوم - على حد علمي - قد تم التغلب عليهم لأنه - وكفكرة عامة - فإن كل packer يجب أن يصل في وقت من الأوقات الى اللحظة التي يفك ضغط البرنامج فيها ، عندها نقوم بعمل dump للبرنامج وانتهى الأمر ! طبعا العملية أعقد قليلا من ذلك لكن أنا اتكلم بشكل عام.

أما الـ protectors :

فهي كالـ packers تقريبا لكن هناك القليل من الفروقات ، فالـ protector يحوي بعض الأساليب والـ "حيل" التي تجعل عملية الهندسة العكسية للملف التنفيذي أكثر صعوبة...لكن الـ protector قد لا تكون من مهامه أن يضغط الملف..لكن ويكل تأكيد فمن مهامه أن يصعب كسر حماية الملف...وهنا تأتي سلبية الـ protectors...إنها - وبسبب تلك الأكواد الزائدة - تجعل حجم البرنامج أكبر من ذي قبل ! . معلومة أخرى...العديد من صانعي الفيروسات (هل سمعت عن 29a :)) يقومون بضغط فيروساتهم...أولا لتصغير حجمها...وثانيا والأهم لمحاولة جعلها غير مكشوفة من قبل الـ antivirus . لكن بالمقابل أصبحت الأخيرة تتسبب في عدة إنذارات كاذبة عن فيروسات بسبب هذا الموضوع...

Get The Party Started :**: UPX x.xx**

بداية أشير الى أنني أرفقت UPX packer مع الدرس ، حتى يمكنك استخدامه لضغط البرامج وبالتالي التدرب على الدرس بنفسك... الآن قم بفتح UPX 1.25 مستخدما PEiD لترى التالي :

UPX 0.89.6 - 1.02 / 1.05 - 1.24 -> Markus & Laszlo

حسننا الآن افتح الملف مستخدما LordPE ثم اختر directories ثم اضغط زر  بجانب Import Table والآن اختر ملف KERNEL32.dll وانظر الى الدوال المستوردة منه بالاسفل...كما ترى لا يوجد الا 3 دوال فقط...وهي LoadLibraryA و GetProcAddress و ExitProcess . ان بقية الدوال اختفت بسبب "بعثرة" جدول الموارد من قبل UPX لكن بقية هذه الدوال لان UPX يحتاجها كي يتم تشغيل الملف بصورة سليمة . ايضا لو اخترت الملف الثاني ستجد انه يستورد دالة واحدة فقط...

والآن شغل الملف مستخدما صديقنا العزيز Olly كما ترى فان اول تعليمة هي تعليمة PUSHAD والتي تقوم بعمل PUSH لجميع المسجلات. لاحظ انه في اغلب - اغلب وليس كل - اصدارات UPX تكون هذه التعليمة هي اول ما تراه عند فتح الملف وآخر تعليمتان هما POPAD يتبعها jmp الى الـ OEP .

00408160	\$ 60	PUSHAD
00408161	. BE 00604000	MOV ESI,UPX1_25.00406000
00408166	. 8DBE 00B0FFFF	LEA EDI,DWORD PTR DS:[ESI+FFFFB000]
0040816C	. 57	PUSH EDI
00408170	. 33D0 FF	OR EBP,FFFFFFFF
00408176	. 5E 10	JMP SHORT UPX1_25.00408182
00408172	90	NOP

ها نحن عند أول تعليمة...الآن لنرى ماذا يوجد في نهاية هذه التعليمات..انزل للأسفل حتى تصل الى آخر تعليمة كما في الصورة التالية :

0040829D	09C0	OR EAX,EAX
0040829F	74 07	JE SHORT UPX1_25.004082A8
004082A1	8903	MOV DWORD PTR DS:[EBX],EAX
004082A3	83C3 04	ADD EBX,4
004082A6	EB E1	JMP SHORT UPX1_25.004082B9
004082A8	FF96 54850000	CALL DWORD PTR DS:[ESI+8554]
004082AF	E9 0C90FFFF	JMP UPX1_25.004012C0
004082B4	00	DB 00
004082B5	00	DB 00
004082B6	00	DB 00

كما ترى فان آخر تعليمتان هما POPAD يليها jmp . ضع bp عند القفزة(4082AF) ثم F9 . الآن اضغط F8 لتصل الى هنا :

004012C0	55	PUSH EBP
004012C1	8BEC	MOV EBP,ESP
004012C3	6A FF	PUSH -1
004012C5	68 F8404000	PUSH UPX1_25.004040F8
004012C7	68 F1D40000	PUSH UPX1_25.00401DF4
004012C9	8A 01 00000000	MOV EAX,DWORD PTR FS:[0]
004012CB	50	PUSH EAX

هذه بداية البرنامج الأصلية....لاحظ ان ال OEP تساوي 4012C0 (كما ترى في الصورة) اما ال EP فتحسب كالتالي :

$$EP = OEP - ImageBase = 4012C0 - 400000 = 12C0$$

الآن اختر Plug-in ثم OllyDump ثم اختر Dump (يفضل ازالة علامة الصح من الخيار الذي في الاسفل...على الرغم من انني لم ازله وتم كل شيء بنجاح...)

Start Address: 400000 Size: A000 Dump

Entry Point: 8160 -> Modify: 82AF Get EIP as OEP Cancel

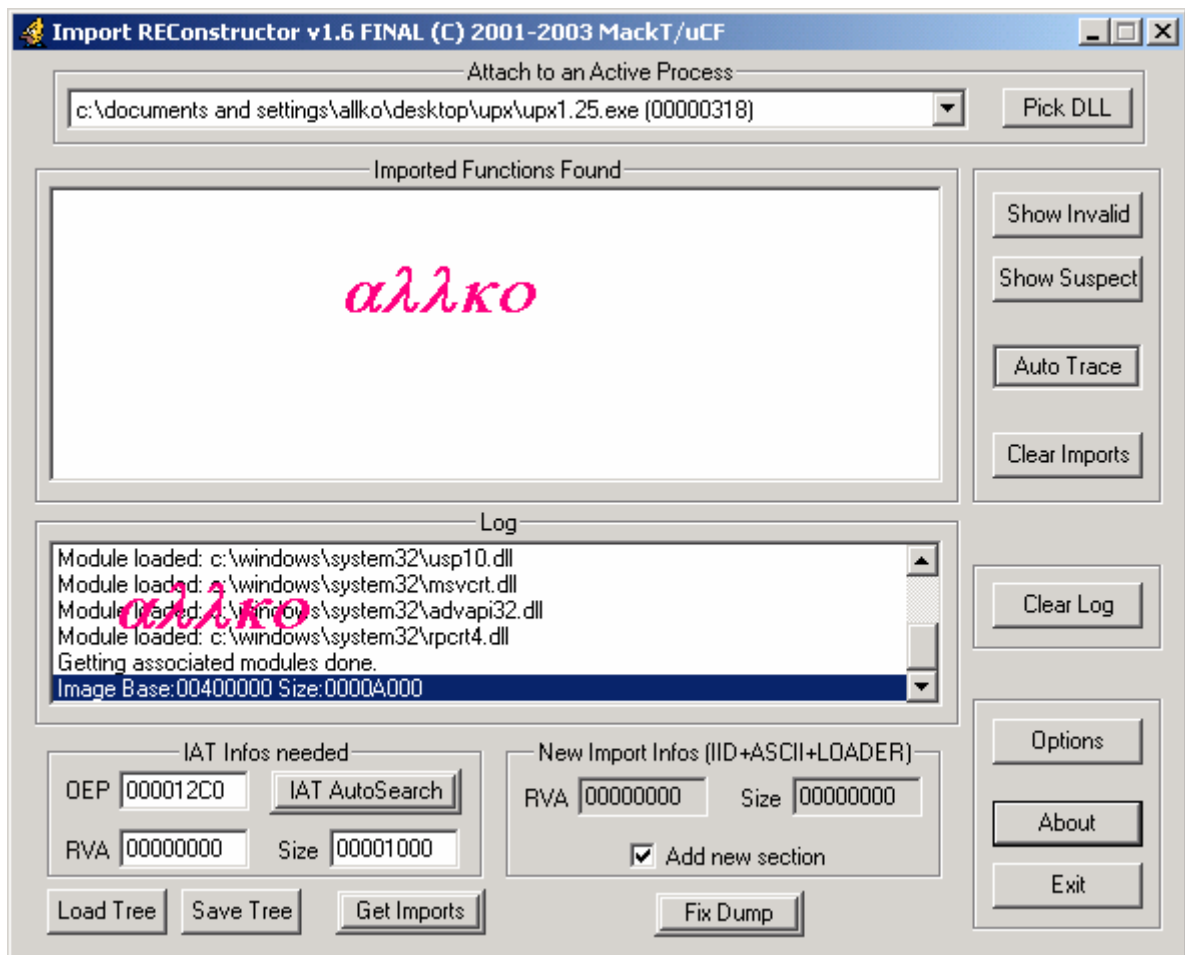
Base of Code: 6000 Base of Data: 9000

☒ Fix Raw Size & Offset of Dump Image

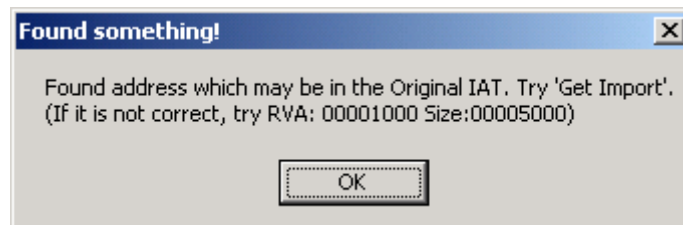
Section	Virtual Size	Virtual Offset	Raw Size	Raw Offset	Characteristics
UPX0	00005000	00001000	00005000	00001000	E0000080
UPX1	00003000	00006000	00003000	00006000	E0000040
.rsrc	00001000	00009000	00001000	00009000	C0000040

☒ Rebuild IAT using "reblt.dll" by y0da

والان احفظ الملف الناتج بأي اسم.
الان قم بتشغيل ImpREC (ابق oly مفتوحا ولا تغلقه) . من القائمة بالا على اختر ملفنا...اي 1.25 upx ثم في خانة OEP اكتب ال EP . (ال EP وليس ال OEP) كما بالصورة



والان اختر IAT AutoSearch لتخرج لك الشاشة التالية :



حسنا ، الآن اضغط على GetImports لترى في المستطيل الابيض الفارغ ، مجموعة من الدوال المستوردة وبجانبها كلمة YES دلالة على نجاح العملية . في النهاية اختر FixDump ثم اختر الملف الذي كنا قد خزناه من Olly .
ها قد انتهينا...افحص الملف الناتج بـ PEid

Microsoft Visual C++ 6.0

ها قد انتهينا...بالمرفقات مع هذا الدرس تجد ملفا باسم UPX1.91 وآخر باسم UPX 2.00 وثالث باسم UPX-Scrambler RC1.x حاول التدرب عليهم فهم مطابقون تماما لما قمنا بعمله للتو.

UPX \$hit 0.0.1

الآن دعنا نتفحص UPX \$hit 0.0.1 . بداية افحصه بـ PEid لترى التالي :

UPX\$HiT 0.0.1 -> dj-siba

كما ترى فهو من برمجة dj-siba أحد أعضاء منتدانا (منتدى البرمجة والحماية) .
حملة الى olly لترى التالي.

004095E8	94	XCHG EAX,ESP
004095E9	BC D1954000	MOV ESP,UPX\$hit-.004095D1
004095EE	B9 17000000	MOV ECX,17
004095F3	80340C DB	XOR BYTE PTR SS:[ESP+ECX],0DB
004095F7	E2 FA	LOOPD SHORT UPX\$hit-.004095F3
004095F9	94	XCHG EAX,ESP
004095FA	90	NOP
004095FB	90	NOP
004095FC	90	NOP
004095FD	FFE0	JMP EAX
004095FF	FF00	INC DWORD PTR DS:[EAX]
00409601	0000	ADD BYTE PTR DS:[EAX],AL
00409603	0000	ADD BYTE PTR DS:[EAX],AL

والآن لنضع نقطة توقف عند تلك القفزة ثم اضغط F9 . ها قد تغير الكود...يجب ان ترى التالي :

004095CE	0000	ADD BYTE PTR DS:[EAX],AL
004095D0	00B8 50814000	ADD BYTE PTR DS:[EAX+408150],BH
004095D6	B9 54010000	MOV ECX,154
004095DB	803408 08	XOR BYTE PTR DS:[EAX+ECX],8
004095DF	E2 FA	LOOPD SHORT UPX\$hit-.004095DB
004095E1	FFE0	JMP EAX
004095E3	80 08 00F83	MOV ECX,838F508B
004095E9	BC D1954000	DEC EDI
004095EE	B9 17000000	MOV ESP,UPX\$hit-.004095D1
004095F3	80340C DB	MOV ECX,17
004095F7	E2 FA	XOR BYTE PTR SS:[ESP+ECX],0DB
004095F9	94	LOOPD SHORT UPX\$hit-.004095F3
004095FA	90	XCHG EAX,ESP
004095FB	90	NOP
004095FC	90	NOP
004095FD	FFE0	NOP
004095FF	FF00	JMP EAX
		INC DWORD PTR DS:[EAX]

ضع نقطة توقف عند القفزة العلوية . ثم run (F9) الآن اذا تتبعنا باستخدام step over أي F8 فسوف يشتغل البرنامج...لذا تتبع مستخدما step into أي F7 لتصل الى التالي :

00408150	60	DB 60	CHAR 'A'
00408151	BE	DB BE	
00408152	00	DB 00	
00408153	60	DB 60	CHAR 'A'
00408154	40	DB 40	CHAR '@'
00408155	00	DB 00	

ان ما نراه هو اشبه بال junk code ..لا يوجد أي تعليمات هنا...لذا اضغط باليمين ثم اختر analysis ثم Analyze code (ستظهر لك رسالة..اضغط ok) . أو ببساطة يمكنك استخدام الاختصار وهو ctrl+A . والآن لنرى مالذي حصلنا عليه :

00408150	60	PUSHAD
00408151	BE 00604000	MOV ESI,UPX\$hit-.00406000
00408156	80BE 00B0FFFF	LEA EDI,DWORD PTR DS:[ESI+FFFFB000]
0040815C	57	PUSH EDI
00408160	83CD FF	OR EBP,FFFFFFFF
00408162	EB 10	JMP SHORT UPX\$hit-.00408172
00408163	90	NOP

رائع...ها هي تعليمة PUSHAD المألوفة...إذن انزل للأسفل (آخر البرنامج) حتى ترى POPAD يليها jmp . ضع bp عند تلك ال jmp ثم F9 ومن ثم F8 . نحن الآن عند ال OEP . اعتقد أنك تعرف البقية...

Do you know ?

معركة عين جالوت

وقعت في 3 سبتمبر 1260 م، في منطقة عين جالوت قرب مدينة بيسان في فلسطين . تعد من أهم المعارك الفاصلة في تاريخ العالم الإسلامي. انتصر فيها المسلمون بقيادة المماليك انتصارا ساحقا على المغول وكانت هذه هي المرة الأولى التي يهزم فيها المغول في معركة حاسمة منذ عهد جنكيز خان. أدت المعركة لانحسار نفوذ المغول في بلاد الشام وخروجهم منها نهائيا وإيقاف المد المغولي المكتسح الذي أسقط الخلافة العباسية سنة 1258 م. كما وأدت المعركة لتعزيز موقع دولة المماليك كأقوى دولة إسلامية في ذلك الوقت لمدة قرنين من الزمان أي إلى أن قامت الدولة العثمانية.

Outroduction :

Nothing but a lot of thxz to dj-siba...

Thx:

DiDi

Dahaka

Dj-siba

And all www.alalame.net/vb members...

There are many crackers, but only one allko
allko34@hotmail.com