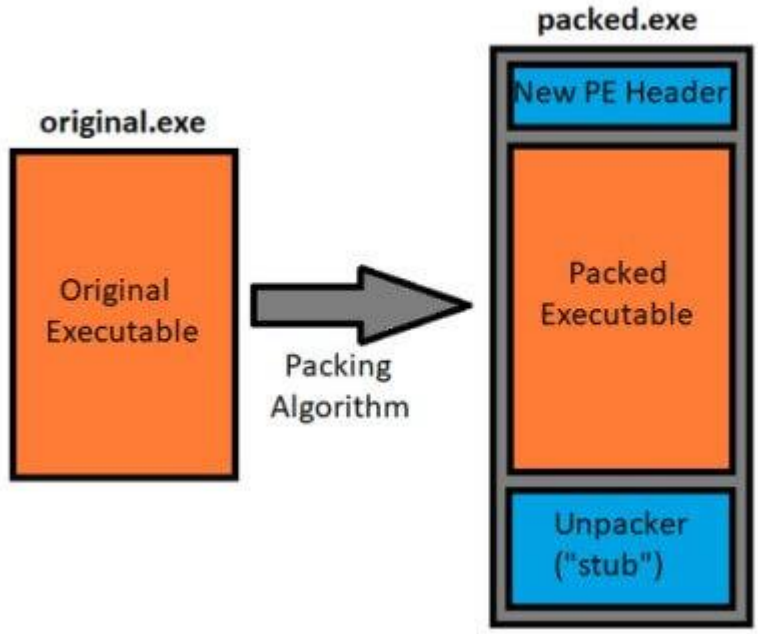
What is a malware packer

و ليش نستخدمه packer وش هو

packer وش يعني

خلني اشرح بشكل بسيط مره وش هو packing



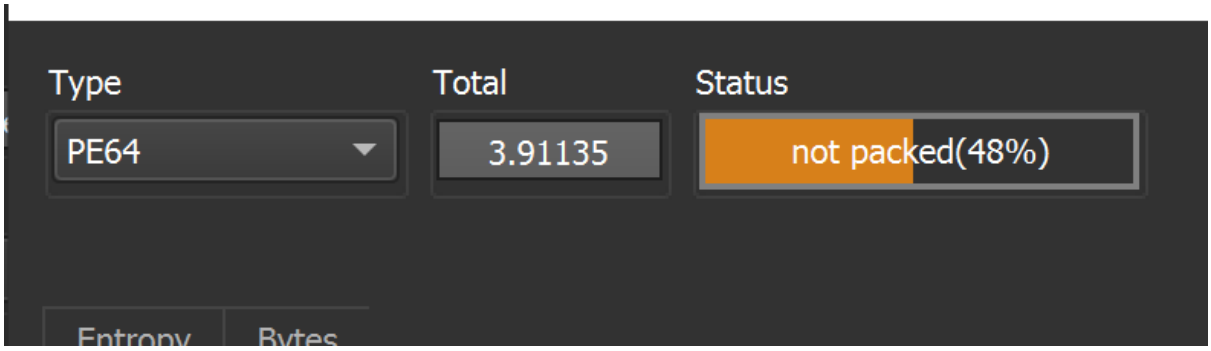
باختصار شديد هي عمليه ضغط لملف بمعنى عندي ملف exe سواء كان malware او غيره اقدر استعمل packer عشان اضغط الملف و اذا ضغطته و حاولت اسوي له RE او هندسة عكسية ماراح يطلع لي الاسمبلي حق exe الأصلي نفس لما يكون عندك txt و تضغطة باستخدام winrar بالطريقة ذي لما تفتح الملف بعد الضغط ماتقدر تشوف محتويات ملف txt حقك, و في مرحلة التشغيل راح ينفك packer و يشتغل البرنامج طبيعي

packer وش هدف ال

له اهداف كثير بس الهدف الرئيسي هو اني امنعك من انك تقرا الاسمبلي حق الملف حقي بصورة مباشرة و من استخداماتة ايضا هو ضغط الملفات لأن exe بعد packing يكون حجمة اصغر

malware ان نعرف كيف packed حقنا او لا

عندنا اشياء كثير ممكن نسويها من اسهلها هو Entropy و كيف يشتغل؟ اذا شاف الملف عشوائي يزيد Entropy طيب بناخذ الحين calc.exe و بنفحصها قبل ما نسوي لها pack و بعد pack و نشوف الفرق



استخدمنا برنامج Detect it easy عشان نقدر نعرف Entropy حق الملف و زي ما نشوف يقول not packed

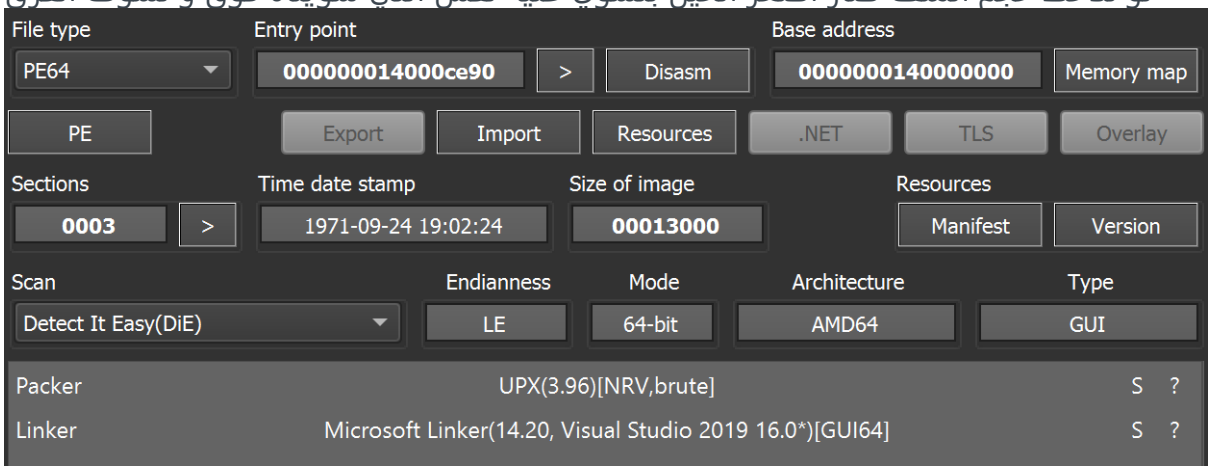
name (33)	blacklist (5)	group (7)	ordinal (0)	library (7)
EventRegister	-	diagnostic	-	advapi32.dll
EventSetInformation	-	diagnostic	-	advapi32.dll
EventWriteTransfer	-	diagnostic	-	advapi32.dll
GetCurrentProcess	-	execution	-	kernel32.dll
GetCurrentProcessId	x	execution	-	kernel32.dll
GetCurrentThreadId	x	execution	-	kernel32.dll
GetModuleHandleW	-	dynamic-library	-	api-ms-win-co...
GetStartupInfoW	-	reckoning	-	api-ms-win-co...
GetSystemTimeAsFileTime	-	file	-	kernel32.dll
GetTickCount	-	reckoning	-	kernel32.dll
QueryPerformanceCounter	-	reckoning	-	kernel32.dll
RtlCaptureContext	-	exception	-	kernel32.dll
RtlLookupFunctionEntry	x	diagnostic	-	kernel32.dll
RtlVirtualUnwind	-	memory	-	kernel32.dll
SetUnhandledExceptionFilter	-	exception	-	kernel32.dll
ShellExecuteW	x	execution	-	shell32.dll
Sleep	-	execution	-	api-ms-win-co...
TerminateProcess	x	execution	-	kernel32.dll
UnhandledExceptionFilter	-	exception	-	kernel32.dll
_XcptFilter	-	-	-	msvcrt.dll
__C_specific_handler	-	-	-	msvcrt.dll
__set_app_type	-	-	-	msvcrt.dll
__setusermatherr	-	-	-	msvcrt.dll
__wgetmainargs	-	-	-	msvcrt.dll
_amsg_exit	-	-	-	msvcrt.dll
_cexit	-	-	-	msvcrt.dll
_commode	-	-	-	msvcrt.dll
_exit	-	-	-	msvcrt.dll
_fmode	-	-	-	msvcrt.dll
initterm	-	-	-	msvcrt.dll

و هنا استعدملت pestudio عشان اطلع معلومات عن functions المستخدمة و نلاحظ ان functions كثير هنا ,و الحين نجرب upx و هو نوع من انواع packers و مشهور جدا

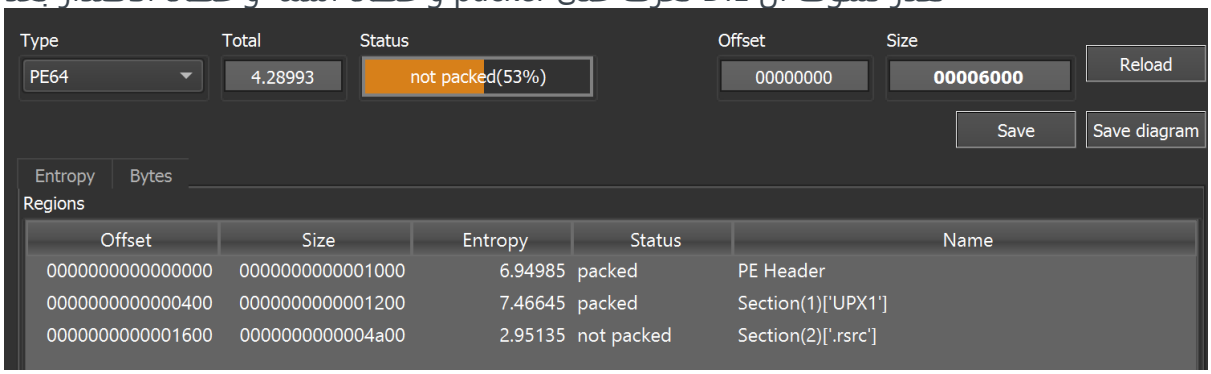
```
C:\Users\kira2\Desktop\upx-3.96-win64\upx-3.96-win64>upx.exe -9 calc.exe
                       Ultimate Packer for eXecutables
                          Copyright (C) 1996 - 2020
UPX 3.96w       Markus Oberhumer, Laszlo Molnar & John Reiser   Jan 23rd 2020

        File size         Ratio      Format      Name
   --------------------   ------   -----------   -----------
     27648 ->     24576   88.89%    win64/pe     calc.exe
Packed 1 file.
```

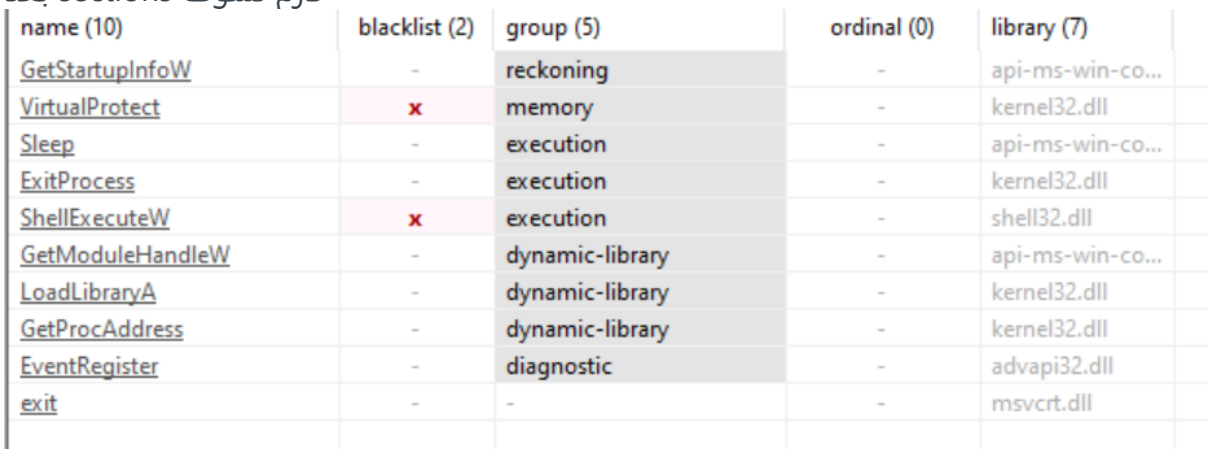
لو نلاحظ حجم الملف صار اصغر الحين بنسوي عليه نفس اللي سويناه فوق و نشوف الفرق



نقدر نشوف ان DIE تعرف على packer و عطانا اسمه و عطانا الاصدار بعد



لو نشوف Entropy بنلاحظ انه not packed بس لما ننزل تحت نشوف ان فيه اشياء packed و هي sections و الحين عرفنا معلومه ان مو شرط نعتمد على Entropy حق الملف بشكل كامل لازم نشوف sections بعد



الحين لو نشوف الfunctions المستعملة ما نلاقي الا شي قليل مقارنة بقبل عملية pack بتلاقي في packed exe بعض ال functions اللي تعطيك تلميح انه packed مثل

```
VirtualProtect VirtualAlloc WriteProcessMemory
```

هذي كلها ممكن تلمح لك ان البرنامج packed

property	value	value	value
name	UPX0	UPX1	.rsrc
md5	n/a	9B777472B2A32F4195DEA42...	D559A27E343B4EA6EDD9D4...
entropy	n/a	7.466	2.951
file-ratio (95.83%)	n/a	18.75 %	77.08 %
raw-address	0x00000400	0x00000400	0x00001600
raw-size (23552 bytes)	0x00000000 (0 bytes)	0x00001200 (4608 bytes)	0x00004A00 (18944 bytes)
virtual-address	0x0000000040001000	0x000000004000C000	0x000000004000E000
virtual-size (73728 bytes)	0x0000B000 (45056 bytes)	0x00002000 (8192 bytes)	0x00005000 (20480 bytes)
entry-point	-	0x0000CE90	-
characteristics	0xE0000080	0xE0000040	0xC0000040
writable	x	x	x
executable	x	x	-
shareable	-	-	-
discardable	-	-	-
initialized-data	-	x	x
uninitialized-data	x	-	-
unreadable	-	-	-
self-modifying	x	x	-
virtualized	x	-	-
file	n/a	n/a	n/a

و لو نروح ل sections بنلاقي اسامي غريبه مثل UPX0 و نلاقي بعد ان فيها صلاحيات writeable و executable و هذي تخلينا نتيقن اكثر انه packed

الملخص

قدرنا نعرف وش pack و ليش يستخدم و كيف نعرف ان هذا packed او لا السؤال هل بس الmalware يسوي pack؟ لا حتى البرامج اللي ماتبيك تشوف السورس السورس حقها و تعرف كيف تشتغل تحطه عشان تصعب عليك هل نقدر نفكه؟ ايه نقدر نفكه طبيعي جدا و حتى اذا كان فكه صعب لازم تعرف ان packer في الاخير لازم يشغل الملف الاصلي ف لما نستخدم debugger بنقدر نشوف اكواد الكود الاصلي تشتغل بعد مايخلص من عملية unpacking

مصادر

https://bidouillesecurity.com/tutorial-writing-a-pe-packer-intro/

https://upx.github.io/ https://www.varonis.com/blog/x64dbg-unpack-malware