PECompact v2.x Unpacking

In this paper, I will be demonstrate how to unpack PE file that packed with PECompact.

Firstly, I take a look to imported functions and sections. If a file is packed, it exhibitions some characteristics. In this sample there are least two indicators whether it is packed.

Module Name	Imports	OFTs	TimeDateSta
000766EC	N/A	00076624	00076628
szAnsi	(nFunctions)	Dword	Dword
kernel32.dll	4	00114BD0	00000000
user32.dll	1	00114BE4	00000000
advapi32.dll	1	00114BEC	00000000
oleaut32.dll	1	00114BF4	00000000
version.dll	1	00114BFC	00000000

OFTs	FTs (IAT)	Hint	Name
Dword	Dword	Word	szAnsi
00114CFC	00114CFC	0000	LoadLibraryA
00114D0C	00114D0C	0000	GetProcAddress
00114D20	00114D20	0000	VirtualAlloc
00114D30	00114D30	0000	VirtualFree

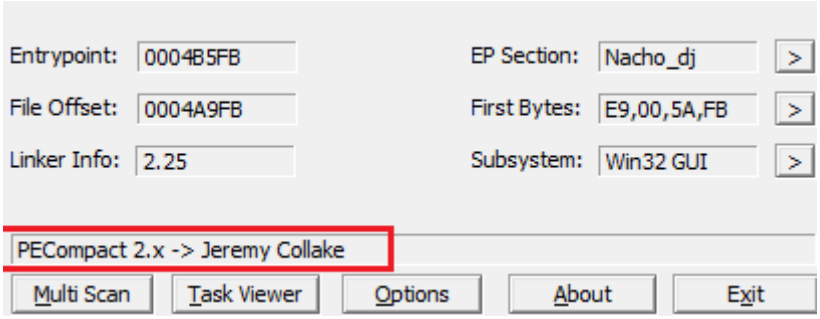
First indicator is each library has only one imported function. In additional, each packed PE file have to contains at least *GetProcAddress* and *LoadLibrary* API functions to resolve other API functions as dynamically. (But only one imported function don't need to be from each library.)

There is an another indicator in sections. Again, as packer characteristic the section names and between virtual-raw size differ can be different from usual.

Name	Virtual Size	Virtual Address	Raw Size
Byte[8]	Dword	Dword	Dword
Nacho_dj	000E8000	00001000	0004A600
Nacho_dj	0002D000	000E9000	0002CC00

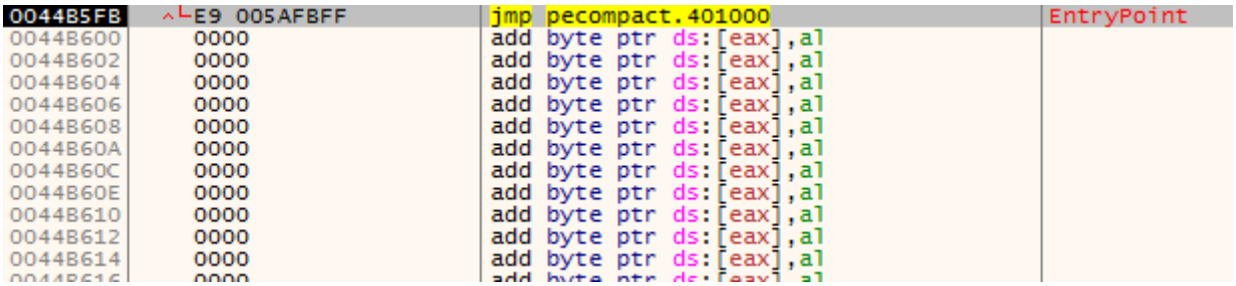
In this sample, the section names are *Nacho_dj* but section names need to be such as .text, .data. Also whether its section names different does not matter, still you can detect if packer used. To do this, you can examine between virtual size and raw size differ. The bigger the size difference, you should suspect packer.

Now we can try to detect which packer used to packing. I tried three different tools to detect packer but only PEiD has detected the packer.

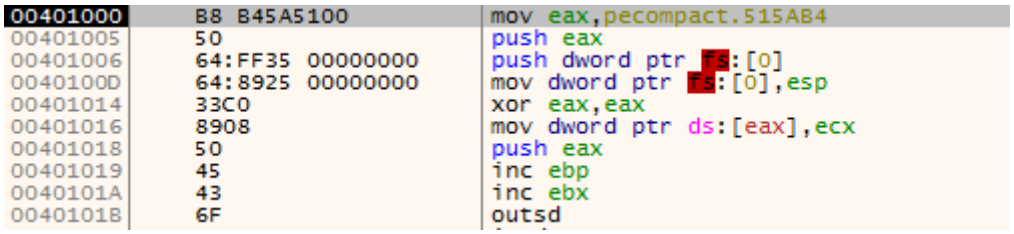


Until here OK. Now, let's learn of characteristics of PECompact packer.

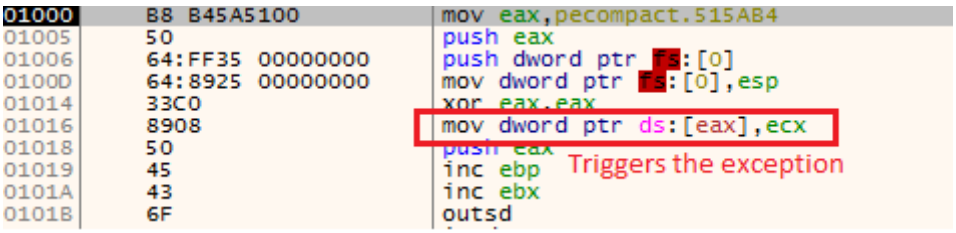
The entry point address of packer;



Take a single step,



The PECompact runs based on SEH(Structured Exceptional Header). To hide transfer controll to OEP sets a exception. Exception handler transfers the flow to a JMP instruction which located the OEP.



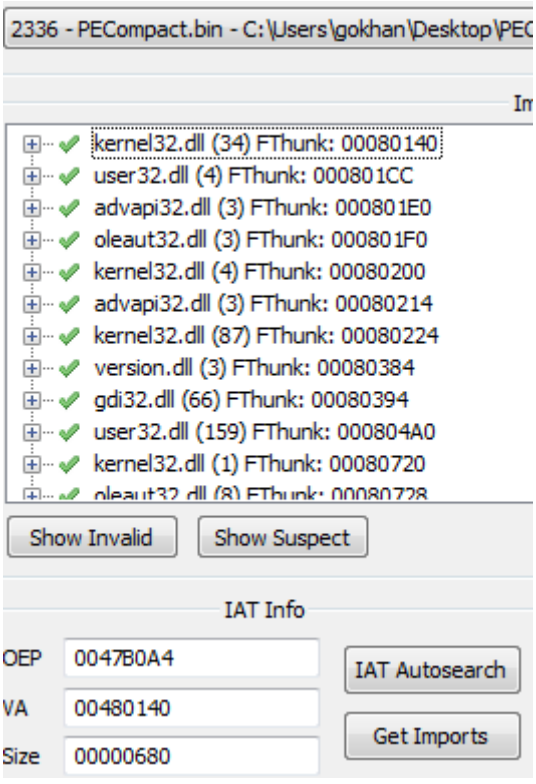
Later, press Shift+F7 when exception is handled to transferthe controll to program again and follow the instructions until '*JMP eax'* instruction.

```
00515B6E    8BC6            mov eax,esi
00515B70    5A              pop edx
00515B71    5E              pop esi
00515B72    5F              pop edi
00515B73    59              pop ecx
00515B74    5B              pop ebx
00515B75    5D              pop ebp
00515B76    FFE0            jmp eax
00515B78    0000            add byte ptr ds:[eax],al
00515B7A    0000            add byte ptr ds:[eax],al
```

After a single step you will be reach OEP.

```
0047B0A4    55              push ebp
0047B0A5    8BEC            mov ebp,esp
0047B0A7    83C4 F0         add esp,FFFFFFF0
0047B0AA    B8 74AE4700     mov eax,pecompact.47AE74
0047B0AF    E8 28AFF8FF     call pecompact.405FDC
0047B0B4    A1 D8E54700     mov eax,dword ptr ds:[47E5D8]
0047B0B9    8B00            mov eax,dword ptr ds:[eax]
0047B0BB    E8 849BFDFF     call pecompact.454C44
0047B0C0    A1 D8E54700     mov eax,dword ptr ds:[47E5D8]
0047B0C5    8B00            mov eax,dword ptr ds:[eax]
0047B0C7    BA 04B14700     mov edx,pecompact.47B104
0047B0CC    E8 6B97FDFF     call pecompact.45483C
0047B0D1    8B0D C0E64700   mov ecx,dword ptr ds:[47E6C0]
0047B0D7    A1 D8E54700     mov eax,dword ptr ds:[47E5D8]
0047B0DC    8B00            mov eax,dword ptr ds:[eax]
0047B0DE    8B15 14E34600   mov edx,dword ptr ds:[46E314]
0047B0E4    E8 739BFDFF     call pecompact.454C5C
0047B0E9    A1 D8E54700     mov eax,dword ptr ds:[47E5D8]
0047B0EE    8B00            mov eax,dword ptr ds:[eax]
```

Now you can enter OEP and rebuild IAT with Scylla.

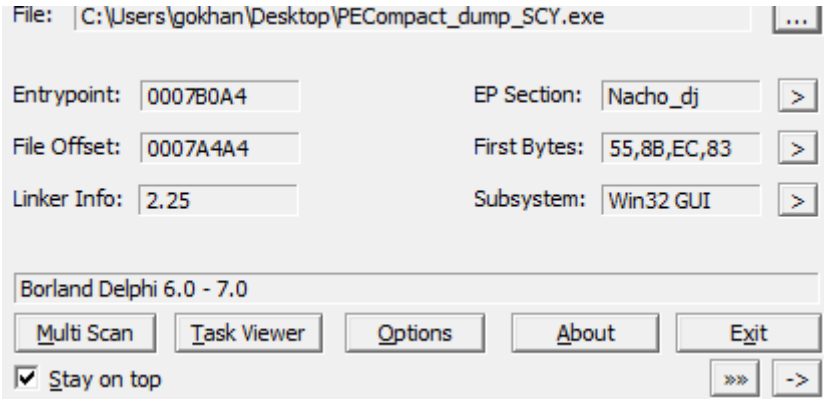


Note: Because the PE file base address is 0x00400000 you should change only 7B0A4 address part. If base address is different, you should be careful when you change OEP.

And dump it. Later you should IAT rebuild. To do this, press PE Rebuild button and select dumped PE file. Finally press Fix Dump button and save last PE file.

You can validate last PE file and continue to analysis.

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
000EAEA5	N/A	000E939C	000E93A0	000E93A4	000E93A8	000E93AC
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
kernel32.dll	34	00116000	00000000	00000000	001167C4	00080140
user32.dll	4	0011608C	00000000	00000000	00116A28	000801CC
advapi32.dll	3	001160A0	00000000	00000000	00116A6D	000801E0
oleaut32.dll	3	001160B0	00000000	00000000	00116AAB	000801F0
kernel32.dll	4	001160C0	00000000	00000000	00116AF2	00080200
advapi32.dll	3	001160D4	00000000	00000000	00116B3B	00080214
kernel32.dll	87	001160E4	00000000	00000000	00116B79	00080224
version.dll	3	00116244	00000000	00000000	0011714F	00080384
gdi32.dll	66	00116254	00000000	00000000	0011719C	00080394
user32.dll	159	00116360	00000000	00000000	001175F8	000804A0
kernel32.dll	1	001165E0	00000000	00000000	00117FCA	00080720
oleaut32.dll	8	001165E8	00000000	00000000	00117FDF	00080728
comctl32.dll	22	0011660C	00000000	00000000	0011807D	0008074C
shell32.dll	3	00116668	00000000	00000000	00118269	000807A8



Some revealed strings in IDA:

Nacho_dj:00...	00000005	C	VB5!
Nacho_dj:00...	00000005	C	VB6!
Nacho_dj:00...	00000007	C	MSVBVM
Nacho_dj:00...	00000025	C	Microsoft Visual C++ Runtime Library
Nacho_dj:00...	00000021	C	urn:schemas-microsoft-com:asm.v1
Nacho_dj:00...	00000005	C	.bss
Nacho_dj:00...	0000000E	C	Dumped file:
Nacho_dj:00...	00000009	C	Finished
Nacho_dj:00...	00000005	C	Done
Nacho_dj:00...	0000003A	C	Error when trying to delete auxiliary G'sloader.exe file.
Nacho_dj:00...	00000064	C	Not able of unpacking this target, please contact the author of the tool reporting involved...
Nacho_dj:00...	0000005F	C	Closed unexpectly, this target seems to be using antidebug tricks, please try again enabli...
Nacho_dj:00...	0000000B	C	\" checkbox
Nacho_dj:00...	00000031	C	The name of the file to be processed is missing.
Nacho_dj:00...	00000035	C	The name of the file dumped is missing or incorrect.
Nacho_dj:00...	00000033	C	The name of the file to be processed is incorrect.
Nacho_dj:00...	00000013	C	Unpacker PECompact
Nacho_dj:00...	00000039	C	========== How to use in console mode ==========\r\n\r\n
Nacho_dj:00...	00000021	C	-c <PathPECompact> <PathDumped>
Nacho_dj:00...	00000007	C	where:
Nacho_dj:00...	00000034	C	PathPECompact: (Mandatory) - Valid route to your
Nacho_dj:00...	0000000A	C	PECompact
Nacho_dj:00...	00000008	C	target
Nacho_dj:00...	0000003C	C	PathDumped: (Mandatory) - Valid route to your dumped file
Nacho_dj:00...	00000013	C	Unpacker PECompact
Nacho_dj:00...	00000029	C	Unpacker PECompact 1.2 - Nacho_dj/ARTeam