Unpacking Vmprotect packer

muha2xmad.github.io/unpacking/Vmprotect/

January 9, 2022



Muhammad Hasan Ali

Malware Analysis learner

2 minute read

As-salamu Alaykum

Introduction

Unpacking a file which is packed using commercial Vmprotect packer.

Download the sample: <u>Here</u>

MD5: A39B4F74B5108A2B9F1A33B2FEB22CC5

Static

DIE

This sample is written in **Delphi** language and is packed through a commercial packer called **VmProtect**. And its **Entropy** is very high in section **Vmp1**

protector	VMProtect(-)[-]			S ?	
linker	Turbo Linke		S ?		
					Options
Signatures			Deep scan	Corr	About
	100%	> Log 70)8 msec	Scan	Exit

Figure(1):

IDA

Open it in IDA. We will notice that there is so many jumps and calls which the packer uses to obfuscate the code. So if you **notice** that if you press on any jump in the function you still in the same function. But the last jump or call will go to another function. So to short efforts and time, we will go to the last call or jump of the function and keep doing that till we get to this one.

↓ · · · · · · · · · · · · · · · · · · ·				
00000003CBDF92				
00000003CBDF92 loc_3CBDF92:				
00000003CBDF92 pushf				
000000003CBDF93 pusha				
00000003CBDF94 mov [ebp+6], ax				
000000003CBDF98 push 31726E4Bh				
000000003CBDF9D push dword ptr [esp+4]				
000000003CBDFA1 jmp loc_3CBD098				
000000003CBDFA1 ; END OF FUNCTION CHUNK FOR sub_3CBC3D4				

Figure(2):

We will reache to a function which has a loop if we press on the last jump of that function it will bring us to the same function. And after checking all the calls and jumps. this call is our way.

```
000000003CBD867 shld ecx, ebp, 5

000000003CBD86B push [esp-14h+arg_10]

000000003CBD86E mov ecx, dword ptr ds:loc_3CBDFAC[eax*4]

000000003CBD875 call sub_3CBC639

000000003CBD875 ; END OF FUNCTION CHUNK FOR sub_3CBC0DB
```

```
Figure(3):
```

Till we find an interesting function. because its retn 48 it will return the last instruction which is push [esp+14h+arg_2C] and we need its address 03CBF12E beacause it will help us in debugging.



Figure(4):

Unpacking process

We set breakpoints on 03CBF12E, VirtualAlloc, and VirtualProtect. Then we hit run to see us in 03CBF12E and we Follow in disassembler of its value. We did that we might see a call to a register in this section. But we won't find any then we unset this breakpoint 03CBF12E. And see Strings references you will see a few strings. Then press run to hit VirtualProtect breakpoint and keep pressing run. Till we see this address 01287000 then we dump it.

👿 CPU 📑 Log 📑 Notes 🍨 Breakpoints 📟 Memory Map	🗊 Call Stack 🕿 SEH 🗵 Script 🔰 Symbols 🔗 Source 🥒 References ≫ Threads 👗 Handles 🤌 Trace	
Strut Log Notes Breadpants Immediation Memory Map ETP EAX EDX File Sec File	Collstox 3 SH B Sorgh A Symbols Source References Threads Handles Troce poor edit add VirtualProtect poor boy poor poor boy poor poor poor poor poor poor poor po	Hide FPU EAX 76548386 EXX 4507240 CEX 60000000 EXX 6107240 CEX 6107240 EXX 61072
7664387 7664397 766439	Jap Ref 132.70.500.4 nop nop nop nop move di.eddi push edp push edp push edp push doord ptr ss:[ebp+10] ref est push doord ptr ss:[ebp+2] inc est push nord ptr ss:[ebp+2] jush nord ptr ss:[ebp+2] push ref is 1; educer dir inc est push edit is 1; educer dir jush nord ptr ss:[ebp+2] push ref is 1; educer dir jush nord ptr ss:[ebp+2] push ref is 1; educer dir jush nord ptr ss:[ebp+3] push ref is 1; educer dir jush ref is 1; educer	EFLAGS 00000244 Zr 1 pF 1 Ar 0 0 OF 0 SF 0 DF 0 0 LastError 00000000 (ERBOR_SUCCESS) LastError 00000004 (STATUS_OBJECT_NAWE_NOT_FOL CS 0028 56 0028 CS 0028 50028 CS 0028 50028 CS 0028 50028 CS 0028 5 Default (addal) ~ 5 12: (ESPH2) 00003148 00003148
edi-0018F6A0 . (3) 8C443BE kernel32.dll:\$143BE #143BE «VirtualProtect» Dump 1 100 Dump 2 100 Dump 3 100 Dump 4 100 Dump 5	Wetch 1 Locals 2 Struct	4: [esp+10] 0018FF60 5: [esp+14] 3ECA8F9A
And Sec. Hest Accurate 112.870.000 100.000 000.0000 000.000 000.000<	018 F64 000018 F64 0018 F64 000018 F64 0018 F64 000018 F64 0018 F64 000018 F64 0018 F64 00000000 0018 F64 000000000 0018 F64 0000018 F64 00000000 0018 F64 000000000 0018 F64 000000000 0018 F64 000000000 0018 F64 00000000000000000000000000000000000	
command: Commands are comma separated (like assembly instructio		Default 👻

Figure(5):

Why that address? Because we need to find the **OEP** which is **push ebp** which we will find it after this address **01287000** which will be later **0128C074**. After that we keep pressing **run** to hit **VirtualAlloc** breakpoint.

🖾 CPU 📴 Log 📫 Notes 🍨 Breakpoints 🚥 Memory Map	🗐 Call Stack 👒 SEH 🔟 Script 🔮	Symbols 🛇 Source 🥕 Rymonces 🐃 Threads 🝶 Handles 🦿 Trace	
EIP 76C41826 88FF 26C41828 55	mov edi,edi	VirtualAlloc	A Hide FPU
7641383 356 7641383 350 7641383 450 7641383 450 7641383 950 7641383 90 7641383 90 7641383 90 7641383 90 7641383 90 7641383 90 7641383 90 7641383 90 7641383 90 7641384 90 7641385 90 764135	puch exp pop ebp imp of bp imp of bp imp of bp imp imp imp imp imp imp imp im	JMP.&VirtualAlloc VirtuaFree JMP.&VirtualFree	EXX 00007330 EXX 00007330 EXX 0000000 EDX 00000000 EDX 0000000000 EDX 00000000 EDX 00000000 EDX 0000000 EDX 0000000 EXX 00000000 EXX 00000000 EXX 00000000 EXX 0000000 EXX 00000000 EXX 0000000 EXX 00000000 EXX 00000000 EXX 00000000 EXX 00000000 EXX 00000000 EXX 0000000000
76C41854 90	nop		1: [esp+4] 00000000 2: [esp+8] 0013FFF0
edi-7330 . 2 cc41826 kernel32.dll:\$11826 #11826 <virtualalloc></virtualalloc>			3: [esp+c] 00001000 4: [esp+10] 00000004 5: [esp+14] 00000000
🕮 Dump 1 🕮 Dump 2 🕮 Dump 3 🕮 Dump 4 🛤 Dump 5	😂 Watch 1 🖂 Locals 🤌 Struct	0018DA6C r00405A5C return to vmp.00405 0018DA70 00000000	ASC from vmp.004056A8
Index Hex Hex Autor Hex Autor 0.227000 358 86 C 31 CO 55 68 D2 70 26 01 64 F7 30 48 D1 31 <td>uhop(.dy0d. 4.ya0y.yhp 0((-, -) 0((-, -) 0((-, -) 0((-)) 0((</td> <td>00180/40003FFF0 00180/40000000 00180/400000000 00180/400000000 00180/400000000 00180/4000000000000000000000000000000000</td> <td>P2E from vmp.00405x40 mad\/besktop\/vmprotect_trojan\/vmp.bin" 226 from 777</td>	uhop(.dy0d. 4.ya0y.yhp 0((-, -) 0((-, -) 0((-, -) 0((-)) 0((00180/40003FFF0 00180/40000000 00180/400000000 00180/400000000 00180/400000000 00180/4000000000000000000000000000000000	P2E from vmp.00405x40 mad\/besktop\/vmprotect_trojan\/vmp.bin" 226 from 777
Command: Commands are comma separated (like assembly instruction			Default 👻

Figure(6):

The last part is the what you need to focus on. As we said we will find the OEP above this address 01287000 which we will be searching for this instruction push ebp. Then press execute till return after that press run to user code while doing that keep you eyes on the Memory address because the OEP is in the range of 01287000. After trying the previous and hitting VirtualAlloc 4 times, we found the push ebp our OEP.



Figure(6):

Unmaping

A friend told me to short effort and time do it automatically using scylla. After that the unpacked file is big 56.7MB

CPU CPU Log Notes Breakpoints CPU Men	tory Map 📑 Call Stack 😪 SEH 🔯 Script 🛸	Symbols Source References	Threads Handles Trace	
0128C05A A1 1C/02C01 0128C05F E8 5CE07AFF	call vmp.A3A0c0			A Hide FPU
2 1 226054 1 226054 1 226057 1 226057 1 240074 1 226072 1 226072 1 226072 1 26074 1 20074 1	mov eax_dword ptr ds:[12:701c] mov eax_dword ptr ds:[12:701c] mov eax_dword ptr ds:[eax+4] mov eax_eax mov eax mov eax_eax mov eax_eax mov eax mov eax			EXX 00000000 • • • • • • • • • • • • • •
0122007 B00014432 0122004 A144820 0122004 S000 S000 0122004 S000 S000 0122001 S45P005 0122001 S45P005 0122001 S45P005 0122005 S45P005 0122005 S45P005 0122005 S45P005 0122005 A145820 0122005 A145820	01 mov ecc.doord ptr ds:[124841] mov esc.doord ptr ds:[124841] 01 mov esc.doord ptr ds:[124841] 03 mov esc.doord ptr ds:[124841] 04 mov esc.doord ptr ds:[124710] 05 mov esc.doord ptr ds:[124710] 06 mov esc.doord ptr ds:[124841] 07 mov esc.doord ptr ds:[124841] 08 mov esc.doord ptr ds:[124841] 09 mov esc.doord ptr ds:[124841] 00 mov esc.doord ptr ds:[124844]	01265018:&"@+K" 01265938:&"@+K"		Lasterror 00000316 (EBR04_00T_ALL_ASSTONED) LastError 00000106 (STATUS_MOT_ALL_ASSTONED) CS 0028 p5 0053 ES 0028 p5 0053 ES 0028 p5 0053 ES 0028 p5 0053 ES 0028 p5 0054 Default (stdcal) - 5 _ Unicided 12: [ESPH4] 0040874C vmp.00400A74C 2: [ESPH4] 0040874C vmp.00400A74C
ebp=0018FF88				4: [esp+10] 0018FF94
.itext:0128c074 vmp.bin:\$E8c074 #0				5: [esp+14] 76C4343D kernel32.76C4343D
🕮 Dump 1 🗰 Dump 2 🕮 Dump 3 🕮 Dump 4 🕮	Dump 5 😔 Watch 1 Locals 2 Struct		0018FF78 0018FFC4 Pointer to SEH_Record[]	L]
Address Her 01280000 000	ASCII 00 Add. 10		0018FF80 0018FF80 0018FF81 0018FF81 0018FF81 0018FF81 0018FF84 0018F784 00018F784 0018F784 0008F784 0008F784 0008F784 0008F784 0008F784 0008F784 00	1343D from ??? 32 from ??? •
ommand: Commands are comma separated (like assembly instructions): mov eax, ebx Default *				
Paused ymp bin: 01280000 -> 01280000 (0x00000001 bitor)				Time Wested Debuggies, 0.00, 41, 22

Figure(7):

Article quote

ولو لم يحاصرك جيشُ الظلام لما كنت نجمًا يُرى أو قمر

REF

1- https://www.youtube.com/watch?v=aoa89Khfgr0&ab_channel=GuidedHacking