




# 【原创】实验吧此处无声WP

原创

Z员外  于 2019-08-14 18:38:25 发布  45  收藏

分类专栏: [随笔](#) 文章标签: [Writeup](#) [实验吧](#) [此处无声](#) [逆向](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/zoleo8088/article/details/99560466>

版权



[随笔](#) 专栏收录该内容

3 篇文章 0 订阅

订阅专栏

## 实验吧此处无声WP

[step1 Get软件](#)

[step2 PEID初步分析](#)

[step3 OD逆向](#)

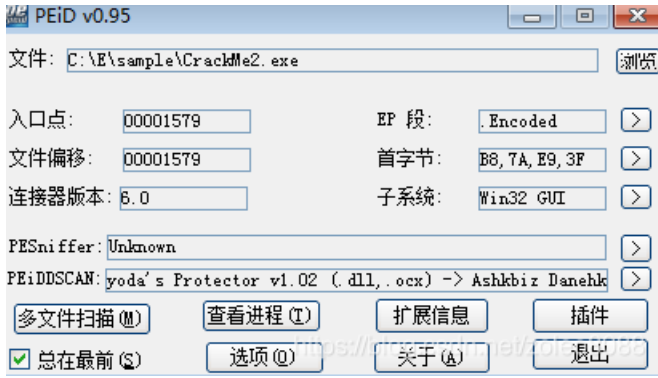
## step1 Get软件

实验吧下载软件, <http://ctf5.shiyanbar.com/crack/5/>

Get基本要求: 找出nsfocus的正确注册码。

## step2 PEID初步分析

先PEID，发现有壳，尝试脱壳失败，大致的流程是启动时先调用LoadLibrary加载kernel32.dll，然后调用GetProcAddress把函数加载到内存中。

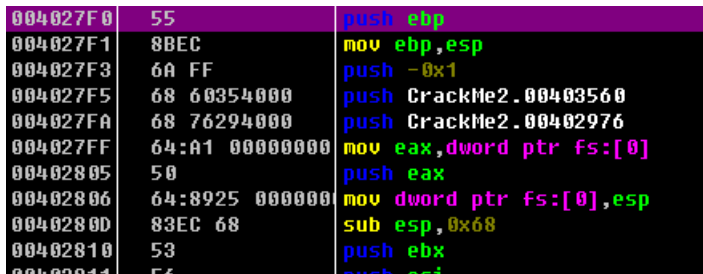


使用插件查找到OEP地址4027F0

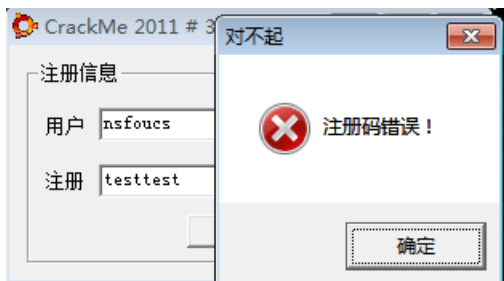


### step3 OD逆向

加载OD，下硬件断点he 4027f0，然后F9执行，程序直接停在了EOP，比较欣慰。



用户名输入nsfocus，注册信息testtest，提示失败



OD下端口 bp MessageBoxA，并初步分析如下：

### 1. 先获取用户名并计算长度

EDI0018F648 ASCII "nsfocus"

00401762	E8 3D100000	call CrackMe2.004027A4	jmp 到 mfc42.#CWnd::GetWind
00401767	8D4C24 38	lea ecx,dword ptr ss:[esp+0x38]	
0040176B	E8 F0010000	call CrackMe2.00401960	
00401770	8D8C24 D4000000	lea edi,dword ptr ss:[esp+0xD4]	用户名
00401777	83C9 FF	or ecx,-0x1	
0040177A	33C0	xor eax,eax	
0040177C	C78424 E0010000	mov dword ptr ss:[esp+0x1E0],0x0	
00401787	F2:AE	repne scas byte ptr es:[edi]	
00401789	F7D1	not ecx	
0040178B	49	dec ecx	用户名长度
0040178C	51	push ecx	
0040178D	8D8C24 D8000000	lea ecx,dword ptr ss:[esp+0xD8]	
00401794	51	push ecx	https://blog.csdn.net/zoleo8088

### 2. 取注册信息的输入，并判断是否是32字节0~9A-F之间的数据，判断为MD5。

于是计算nsfocus的MD5填入注册文本框，通过分析把数据内容保存到下面的地址

```
0018F648 42 39 42 37 44 44 31 43 34 32 31 45 30 30 35 42 B9B7DD1C421E005B
0018F658 43 39 41 37 46 37 30 42 38 34 38 45 33 44 30 45 C9A7F70B848E3D0E
```

004017B8	50	push eax	
004017B9	8D4B 64	lea ecx,dword ptr ds:[ebx+0x64]	
004017BC	E8 E30F0000	call CrackMe2.004027A4	jmp 到 mfc42.#CWnd::GetWindowText
004017C1	8D8C24 D4000000	lea ecx,dword ptr ss:[esp+0xD4]	
004017C8	51	push ecx	
004017C9	8BCB	mov ecx,ebx	
004017CB	E8 A0000000	call CrackMe2.00401870	判断是否是MD5,且全部大写
004017D0	85C0	test eax,eax	
004017D2	74 54	je short CrackMe2.00401828	
004017D4	8D5424 18	lea edx,dword ptr ss:[esp+0x18]	

### 3. sub4018C0的目的是把输入的MD5的数据由字符串转换为hex，保存到下面的地址

```
0018F58C B9 B7 DD 1C 42 1E 00 5B C9 A7 F7 0B 84 8E 3D 0E 狗?B.[骚?副=
0018F59C 05 72 87 D9 ED 85 6A DA B2 97 CF 63 56 C3 61 8A r嚙齏j诨横cV胸?
```

然后比较上面两行是否相同，如果相同则成功。

那么问题来了18F59C的数据怎么来的？

004017C8	51	push ecx	
004017C9	8BCB	mov ecx,ebx	
004017CB	E8 A0000000	call CrackMe2.00401870	判断是否是MD5，且全部大写
004017D0	85C0	test eax,eax	
004017D2	74 54	je short CrackMe2.00401828	
004017D4	8D5424 18	lea edx,dword ptr ss:[esp+0x18]	
004017D8	56	push esi	
004017D9	8D8424 D8000000	lea eax,dword ptr ss:[esp+0xD8]	
004017E0	52	push edx	
004017E1	50	push eax	
004017E2	8BCB	mov ecx,ebx	
004017E4	E8 D7000000	call CrackMe2.004018C0	MD5 string to hex
004017E9	8D4C24 0C	lea ecx,dword ptr ss:[esp+0xC]	
004017ED	6A 10	push 0x10	
004017EF	51	push ecx	
004017F0	E8 FBF9FFFF	call CrackMe2.004011F0	
004017F5	8D5424 24	lea edx,dword ptr ss:[esp+0x24]	
004017F9	8D4424 24	lea eax,dword ptr ss:[esp+0x24]	
004017FD	52	push edx	
004017FE	50	push eax	
004017FF	E8 ECF9FFFF	call CrackMe2.004012F0	
00401804	83C4 10	add esp,0x10	
00401807	B9 04000000	mov ecx,0x4	
0040180C	8D7C24 1C	lea edi,dword ptr ss:[esp+0x1C]	
00401810	8D7424 2C	lea esi,dword ptr ss:[esp+0x2C]	
00401814	33D2	xor edx,edx	
00401816	F3:A7	repe cmps dword ptr es:[edi],dword ptr ds:[esi]	
00401818	5E	pop esi	0018F58C
00401819	75 0D	jnz short CrackMe2.00401828	
0040181B	52	push edx	
0040181C	68 64404000	push CrackMe2.00404064	ASCII "恭喜"
00401821	68 58404000	push CrackMe2.00404058	ASCII "破解成功"
00401826	EB 0C	jmp short CrackMe2.00401834	https://www.cnblogs.com/zoleo8088

4. 重新跟踪分析发现，sub401AA0把输入的“nsfoucs”进行了MD5，并保存在0018F59C

0040179E	8D5424 28	lea edx,dword ptr ss:[esp+0x28]	
004017A2	8D4C24 38	lea ecx,dword ptr ss:[esp+0x38]	
004017A6	52	push edx	
004017A7	E8 F4020000	call CrackMe2.00401AA0	MD5(nsfocus)
004017AC	8D8424 D4000000	lea eax,dword ptr ss:[esp+0xD4]	
004017B3	68 04010000	push 0x104	
004017B8	50	push eax	
004017B9	8D4B 64	lea ecx,dword ptr ds:[ebx+0x64]	

5. 然后才到步骤3，此时18F58C==18F59C

004017E0	52	push edx	
004017E1	50	push eax	
004017E2	8BCB	mov ecx,ebx	
004017E4	E8 D7000000	call CrackMe2.004018C0	MD5 string to hex
004017E9	8D4C24 0C	lea ecx,dword ptr ss:[esp+0xC]	
004017ED	6A 10	push 0x10	
004017EF	51	push ecx	
004017F0	E8 FBF9FFFF	call CrackMe2.004011F0	
004017F5	8D5424 24	lea edx,dword ptr ss:[esp+0x24]	
004017F9	8D4424 24	lea eax,dword ptr ss:[esp+0x24]	
004017FD	52	push edx	
004017FE	50	push eax	
004017FF	E8 ECF9FFFF	call CrackMe2.004012F0	

堆栈地址=0018F57C  
ecx=0000000E

地址	HEX 数据	ASCII	0018F570	00000001	
0018F51C	04 01 00 00	48 F6 18 00	A6 00 3F 76	25 D4 5B 44	法.H?..??%?D
0018F52C	50 F6 18 00	01 00 00 00	D8 FD 18 00	28 F5 18 00	P?.P?.f...d?
0018F53C	C0 F5 18 00	4C F7 18 00	B6 A6 44 76	C5 21 7C 32	佚 .!..P?.f
0018F54C	FE FF FF FF	A6 00 3F 76	EF 7F 1C 6D	50 F6 18 00	d?.佚 .f...d?
0018F55C	01 00 00 00	D8 FD 18 00	E9 17 40 00	48 F6 18 00	法.H?..??%?D
0018F56C	8C F5 18 00	01 00 00 00	00 00 00 00	11 01 00 00	...f.....
0018F57C	35 47 82 5C	33 8C 85 77	9A 67 45 7A	6D 5C 16 47	5C径3真欢Ez
0018F58C	B9 B7 DD 1C	42 1E 00 5B	C9 A7 F7 0B	84 8E 3D 0E	圆?鯖?D破提
0018F59C	B9 B7 DD 1C	42 1E 00 5B	C9 A7 F7 0B	84 8E 3D 0E	狗?B.[骚?刷=
0018F5AC	00 00 00 00	01 23 45 67	89 AB CD EF	FE DC BA 98	...#Eg?器?输

6. 当执行sub4012f0时，发现18F58C的内容发生了变化，应该是对MD5(用户名)进行了处理。

004017FD	52	push edx	
004017FE	50	push eax	
004017FF	E8 ECF9FFFF	call CrackMe2.004012F0	
00401804	83C4 10	add esp,0x10	
00401807	B9 04000000	mov ecx,0x4	
0040180C	8D7C24 1C	lea edi,dword ptr ss:[esp+0x1C]	
00401810	8D7424 2C	lea esi,dword ptr ss:[esp+0x2C]	
00401814	33D2	xor edx,edx	
00401816	F3:A7	repe cmps dword ptr es:[edi],dword ptr ds:[	

004012F0=CrackMe2.004012F0

地址	HEX 数据	ASCII			
0018F51C	04 01 00 00	48 F6 18 00	A6 00 3F 76	25 D4 5B 44	法.H?..??%?D
0018F52C	50 F6 18 00	50 F6 18 00	01 00 00 00	64 F7 18 00	P?.P?.f...d?
0018F53C	D8 FD 18 00	04 00 00 00	50 F6 18 00	01 00 00 00	佚 .!..P?.f
0018F54C	64 F7 18 00	D8 FD 18 00	EC 41 40 00	14 ED 70 33	d?.佚 .f...d?
0018F55C	04 18 40 00	AC 19 30 16	8C F5 18 00	7C F5 18 00	法.H?..??%?D
0018F56C	00 00 00 00	01 00 00 00	00 00 00 00	11 01 00 00	...f.....
0018F57C	35 47 82 5C	33 8C 85 77	9A 67 45 7A	6D 5C 16 47	5C径3真欢Ez
0018F58C	87 FD 88 07	F7 49 94 3B	F5 90 44 B4	56 94 67 E5	圆?鯖?D破提
0018F59C	B9 B7 DD 1C	42 1E 00 5B	C9 A7 F7 0B	84 8E 3D 0E	狗?B.[骚?刷=

综上，程序大致的意思是把输入的注册信息进行了加密，加密的结果与MD5(用户名)相同即成功获得flag。那么相反，把MD5(用户名)进行解密的内容就是flag

直接修改汇编指令，让sub4012f0重新执行一次，发现并不能解密出想要的内容，看了一下大佬们的WP，说是用RC6的算法，（本人目前没有道行如何判断使用哪种算法...），大概调研了一下，不同于AES，RC6虽然是对称加密，但他的加密和解密的算法不同，无法直接调用sub4012f0来解密。

```
004017F0  E8 FBF9FFFF  call CrackMe2.004017F0
004017F5  8D5424 24    lea edx,dword ptr ss:[esp+0x24]
004017F9  8D4424 24    lea eax,dword ptr ss:[esp+0x24]
004017FD  52          push edx
004017FE  50          push eax
004017FF  E8 ECF9FFFF  call CrackMe2.004012F0
00401804  83C4 00     add esp,0x0
00401807  EB EC      jmp short CrackMe2.004017F5
00401809  90         nop
0040180A  90         nop
0040180B  90         nop
0040180C  8D7C24 1C    lea edi,dword ptr ss:[esp+0x1C]
00401810  8D7424 2C    lea esi,dword ptr ss:[esp+0x2C]
```

于是找工具进行解密...



收工，哎，，不是很完美。

