攻防世界 Reverse高手进阶区 2分题 Windows_Reverse1

原创

思源湖的鱼 于 2020-12-31 14:40:56 发布 199 收藏 1

分类专栏： ctf 文章标签： ctf 攻防世界 reverse

本文链接：https://blog.csdn.net/weixin_44604541/article/details/112008503

版权

ctf 专栏收录该内容

200 篇文章 23 订阅

订阅专栏

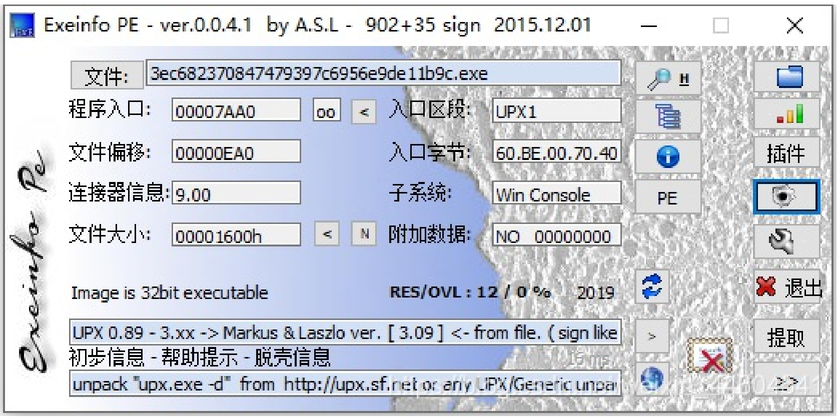
前言

继续ctf的旅程
攻防世界Reverse高手进阶区的2分题
本篇是Windows_Reverse1的writeup

发现攻防世界的题目分数是动态的
就仅以做题时的分数为准了

解题过程

PE查壳



Exeinfo PE - ver.0.0.4.1 by A.S.L - 902+35 sign 2015.12.01

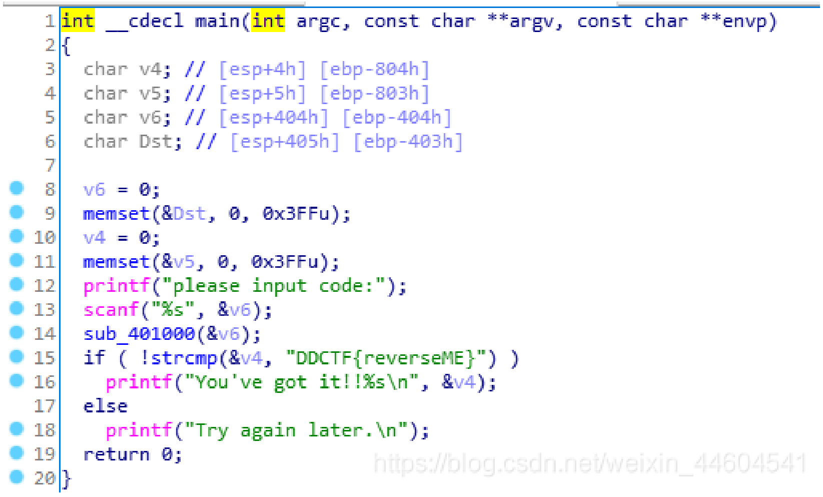
文件: 3ec682370847479397c6956e9de11b9c.exe
程序入口: 00007AA0
入口区段: UPX1
文件偏移: 00000EA0
入口字节: 60.BE.00.70.40
连接器信息: 9.00
子系统: Win Console
文件大小: 00001600h
附加数据: NO 00000000

Image is 32bit executable
RES/OVL : 12 / 0 % 2019

UPX 0.89 - 3.xx -> Markus & Laszlo ver. [3.09] <- from file. (sign like
初步信息 - 帮助提示 - 脱壳信息
unpack "upx.exe -d" from http://upx.sf.net or any UPX/Generic unpa

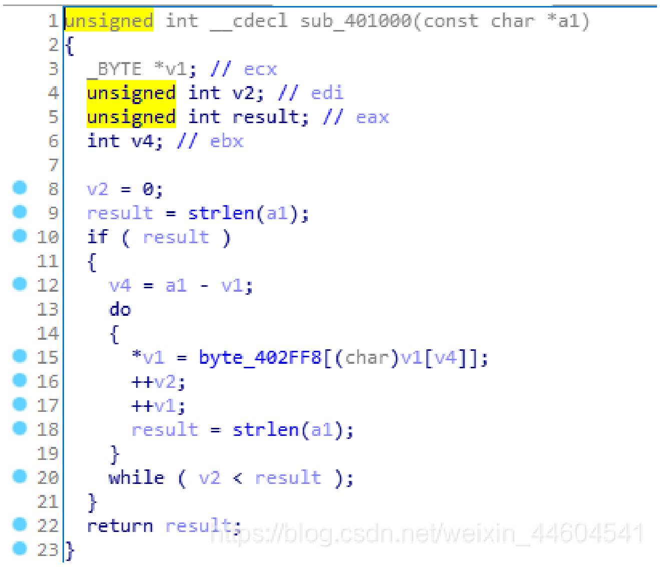
UPX的壳

UPXSHELL脱壳后
扔进IDA



```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  char v4; // [esp+4h] [ebp-804h]
  char v5; // [esp+5h] [ebp-803h]
  char v6; // [esp+404h] [ebp-404h]
  char Dst; // [esp+405h] [ebp-403h]

  v6 = 0;
  memset(&Dst, 0, 0x3FFu);
  v4 = 0;
  memset(&v5, 0, 0x3FFu);
  printf("please input code:");
  scanf("%s", &v6);
  sub_401000(&v6);
  if ( !strcmp(&v4, "DDCTF{reverseME}") )
    printf("You've got it!!%s\n", &v4);
  else
    printf("Try again later.\n");
  return 0;
}
```



```
unsigned int __cdecl sub_401000(const char *a1)
{
  _BYTE *v1; // ecx
  unsigned int v2; // edi
  unsigned int result; // eax
  int v4; // ebx

  v2 = 0;
  result = strlen(a1);
  if ( result )
  {
    v4 = a1 - v1;
    do
    {
      *v1 = byte_402FF8[(char)v1[v4]];
      ++v2;
      ++v1;
      result = strlen(a1);
    }
    while ( v2 < result );
  }
  return result;
}
```

是从`0x402ff8`开始，以我们的输入为索引，形成新字符串，这个字符串与`DDCTF{reverseME}`作比较
扔进OD取出16进制

```
00 00 00 00 00 00 00 00 64 CC 3D CC 9B 33 C2 33 FF FF FF FF FF FF FF FF FE FF FF FF 01 00 00 00
7E 7D 7C 7B 7A 79 78 77 76 75 74 73 72 71 70 6F 6E 6D 6C 6B 6A 69 68 67 66 65 64 63 62 61 60 5F
5E 5D 5C 5B 5A 59 58 57 56 55 54 53 52 51 50 4F 4E 4D 4C 4B 4A 49 48 47 46 45 44 43 42 41 40 3F
3E 3D 3C 3B 3A 39 38 37 36 35 34 33 32 31 30 2F 2E 2D 2C 2B 2A 29 28 27 26 25 24 23 22 21
```

于是有脚本

```
hexData=[
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xAE, 0x1D, 0x3E, 0x07, 0x51, 0xE2, 0xC1,
    0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFE, 0xFF, 0xFF, 0xFF, 0x01, 0x00, 0x00, 0x00,
    0x7E, 0x7D, 0x7C, 0x7B, 0x7A, 0x79, 0x78, 0x77, 0x76, 0x75, 0x74, 0x73, 0x72, 0x71, 0x70, 0x6F,
    0x6E, 0x6D, 0x6C, 0x6B, 0x6A, 0x69, 0x68, 0x67, 0x66, 0x65, 0x64, 0x63, 0x62, 0x61, 0x60, 0x5F,
    0x5E, 0x5D, 0x5C, 0x5B, 0x5A, 0x59, 0x58, 0x57, 0x56, 0x55, 0x54, 0x53, 0x52, 0x51, 0x50, 0x4F,
    0x4E, 0x4D, 0x4C, 0x4B, 0x4A, 0x49, 0x48, 0x47, 0x46, 0x45, 0x44, 0x43, 0x42, 0x41, 0x40, 0x3F,
    0x3E, 0x3D, 0x3C, 0x3B, 0x3A, 0x39, 0x38, 0x37, 0x36, 0x35, 0x34, 0x33, 0x32, 0x31, 0x30, 0x2F,
    0x2E, 0x2D, 0x2C, 0x2B, 0x2A, 0x29, 0x28, 0x27, 0x26, 0x25, 0x24, 0x23, 0x22, 0x21, 0x20, 0x00,
    0x01, 0x00, 0x00, 0x00, 0x90, 0x19, 0x9F, 0x00, 0xA8, 0x2C, 0x9F ]
s='DDCTF{reverseME}'
flag=''
for i in range(len(s)):
  flag+=chr(hexData[ord(s[i])])
print('flag{'+flag+'}')
```

得到flag：`flag{ZZ[JX#,9(9,+9QY!}`

结语

简单题