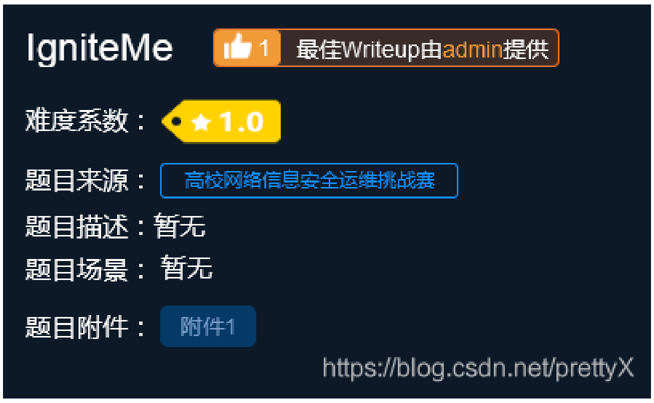
XCTF RE:IgniteMe

原创

prettyX 于 2020-12-04 16:46:04 发布 56 收藏
分类专栏： reverse

本文链接：https://blog.csdn.net/prettyX/article/details/110641384
版权

reverse 专栏收录该内容
12 篇文章 0 订阅
订阅专栏



IgniteMe 1 最佳Writeup由admin提供

难度系数：★1.0

题目来源：高校网络信息安全运维挑战赛

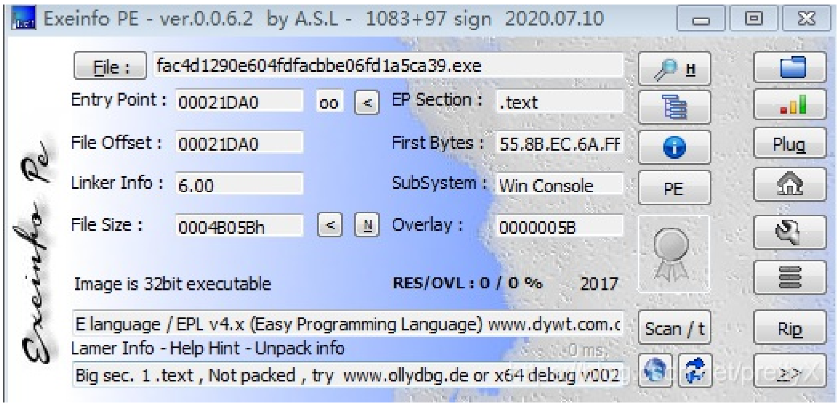
题目描述：暂无

题目场景： 暂无

题目附件： 附件1

https://blog.csdn.net/prettyX

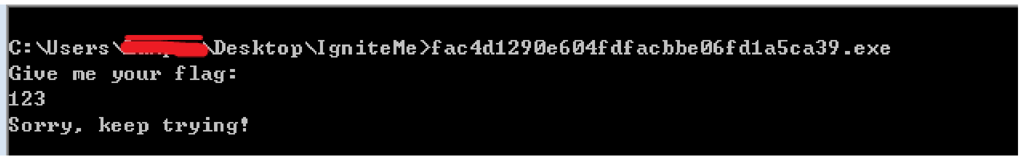
查看基本信息，无壳



Exeinfo PE - ver.0.0.6.2 by A.S.L - 1083+97 sign 2020.07.10

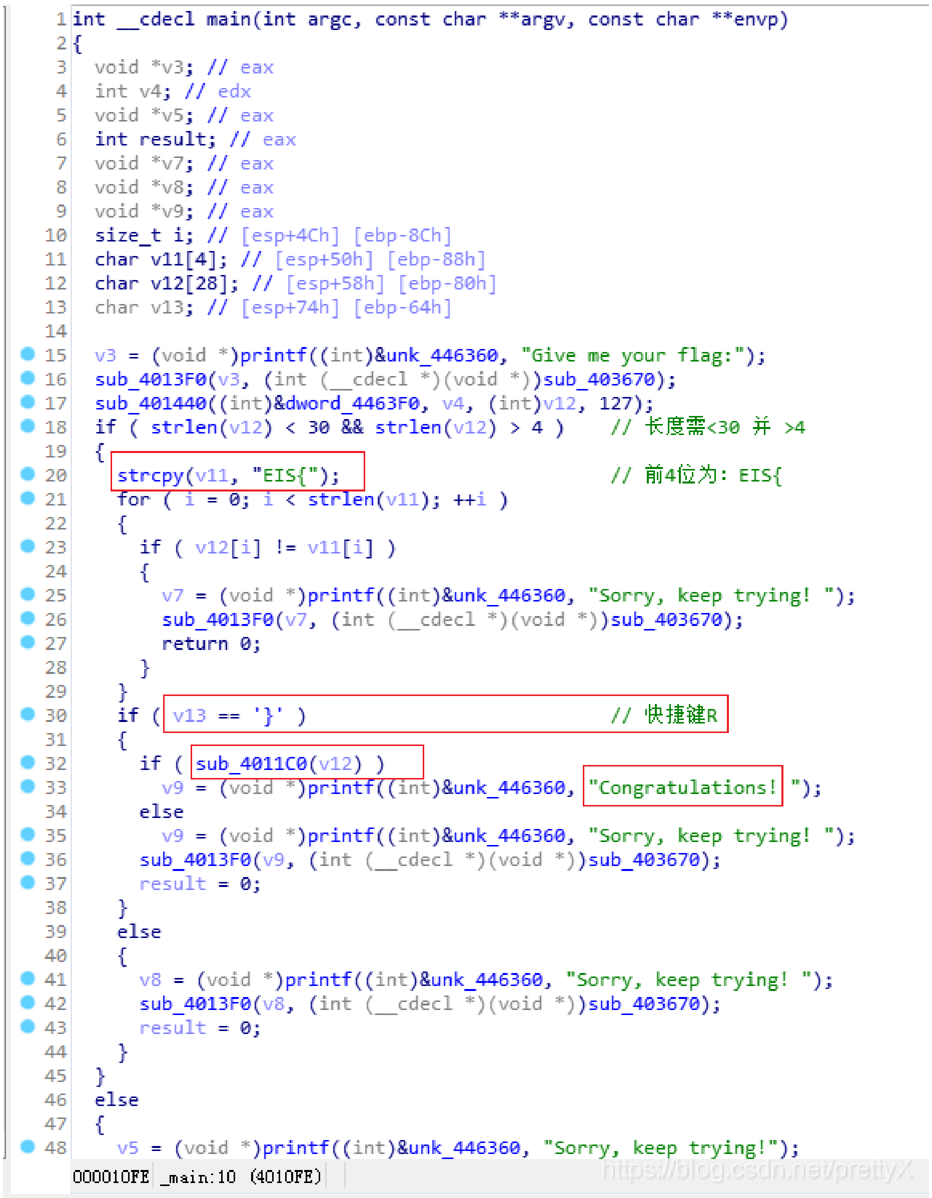
File : fac4d1290e604fdfacbbe06fd1a5ca39.exe
Entry Point : 00021DA0 oo < EP Section : .text
File Offset : 00021DA0 First Bytes : 55.8B.EC.6A.FF
Linker Info : 6.00 SubSystem : Win Console
File Size : 0004B05Bh < N Overlay : 0000005B
Image is 32bit executable RES/OVL : 0 / 0 % 2017
E language / EPL v4.x (Easy Programming Language) www.dywt.com.c
Lamer Info - Help Hint - Unpack info 0 ms
Big sec. 1 .text , Not packed , try www.ollydbg.de or x64 debug v002

尝试运行



```
C:\Users\[redacted]\Desktop\IgniteMe>fac4d1290e604fdfacbbe06fd1a5ca39.exe
Give me your flag:
123
Sorry, keep trying!
```

IDA F5

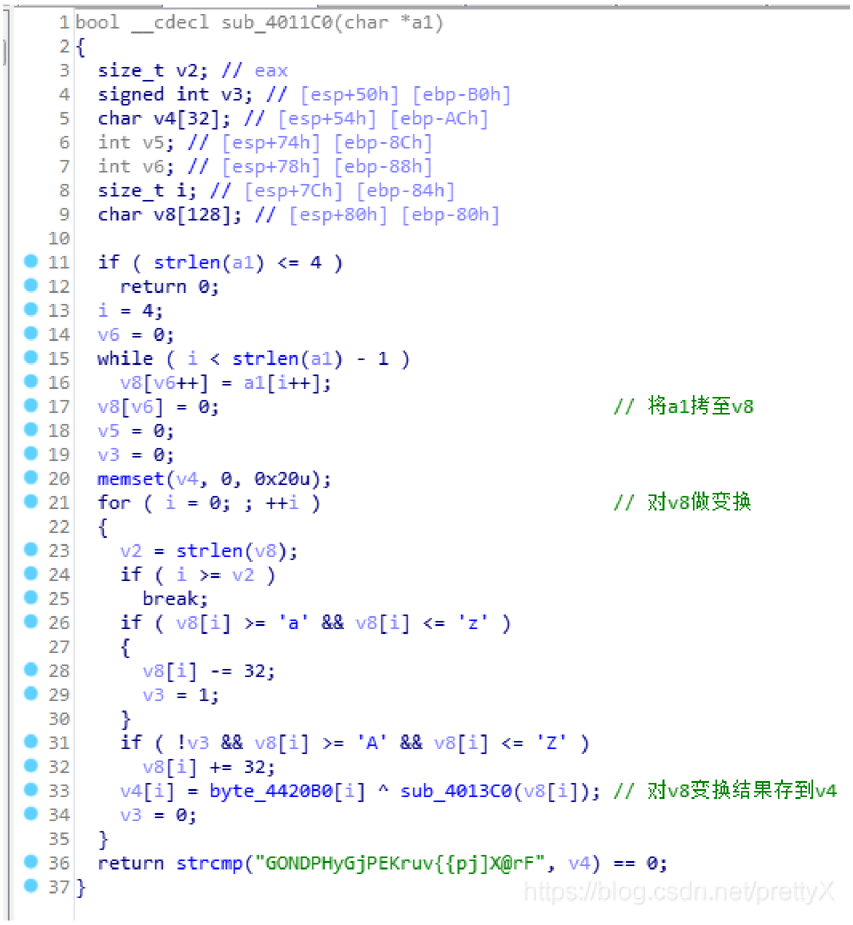


```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  void *v3; // eax
  int v4; // edx
  void *v5; // eax
  int result; // eax
  void *v7; // eax
  void *v8; // eax
  void *v9; // eax
  size_t i; // [esp+4Ch] [ebp-8Ch]
  char v11[4]; // [esp+50h] [ebp-88h]
  char v12[28]; // [esp+58h] [ebp-80h]
  char v13; // [esp+74h] [ebp-64h]

  v3 = (void *)printf((int)&unk_446360, "Give me your flag:");
  sub_4013F0(v3, (int (__cdecl *)(void *))sub_403670);
  sub_401440((int)&dword_4463F0, v4, (int)v12, 127);
  if ( strlen(v12) < 30 && strlen(v12) > 4 )    // 长度需<30 并 >4
  {
    strcpy(v11, "EIS{");                        // 前4位为: EIS{
    for ( i = 0; i < strlen(v11); ++i )
    {
      if ( v12[i] != v11[i] )
      {
        v7 = (void *)printf((int)&unk_446360, "Sorry, keep trying! ");
        sub_4013F0(v7, (int (__cdecl *)(void *))sub_403670);
        return 0;
      }
    }
    if ( v13 == '}' )                           // 快捷键R
    {
      if ( sub_4011C0(v12) )
        v9 = (void *)printf((int)&unk_446360, "Congratulations! ");
      else
        v9 = (void *)printf((int)&unk_446360, "Sorry, keep trying! ");
      sub_4013F0(v9, (int (__cdecl *)(void *))sub_403670);
      result = 0;
    }
    else
    {
      v8 = (void *)printf((int)&unk_446360, "Sorry, keep trying! ");
      sub_4013F0(v8, (int (__cdecl *)(void *))sub_403670);
      result = 0;
    }
  }
  else
  {
    v5 = (void *)printf((int)&unk_446360, "Sorry, keep trying!");
```

```
000010FE _main:10 (4010FE)
```

通过字符串“Congratulations”，我们发现重要函数sub_4011C0()，查看

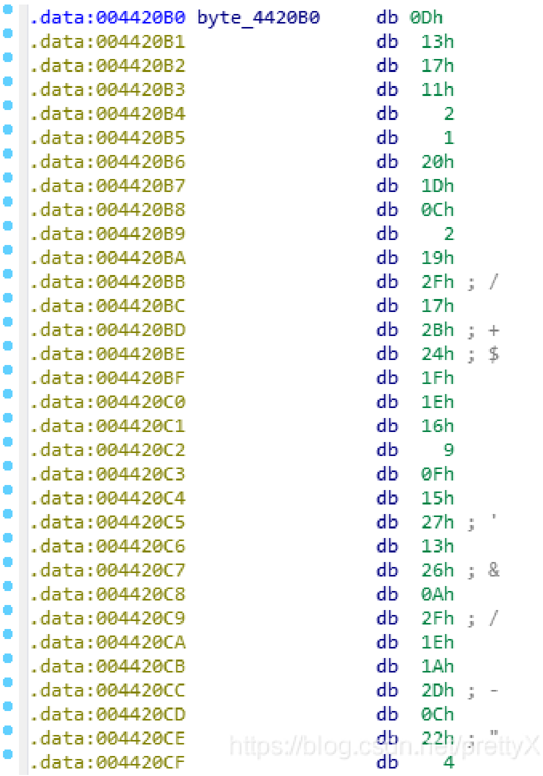


```
bool __cdecl sub_4011C0(char *a1)
{
  size_t v2; // eax
  signed int v3; // [esp+50h] [ebp-B0h]
  char v4[32]; // [esp+54h] [ebp-ACh]
  int v5; // [esp+74h] [ebp-8Ch]
  int v6; // [esp+78h] [ebp-88h]
  size_t i; // [esp+7Ch] [ebp-84h]
  char v8[128]; // [esp+80h] [ebp-80h]

  if ( strlen(a1) <= 4 )
    return 0;
  i = 4;
  v6 = 0;
  while ( i < strlen(a1) - 1 )
    v8[v6++] = a1[i++];
  v8[v6] = 0;                                   // 将a1拷至v8
  v5 = 0;
  v3 = 0;
  memset(v4, 0, 0x20u);
  for ( i = 0; ; ++i )                          // 对v8做变换
  {
    v2 = strlen(v8);
    if ( i >= v2 )
      break;
    if ( v8[i] >= 'a' && v8[i] <= 'z' )
    {
      v8[i] -= 32;
      v3 = 1;
    }
    if ( !v3 && v8[i] >= 'A' && v8[i] <= 'Z' )
      v8[i] += 32;
    v4[i] = byte_4420B0[i] ^ sub_4013C0(v8[i]); // 对v8变换结果存到v4
    v3 = 0;
  }
  return strcmp("GONDPHyGjPEKruv{{pj]X@rF", v4) == 0;
}
```

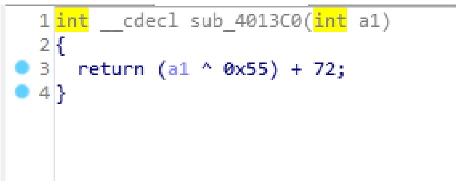
分析正如上图中注释的一样，对v8的变换在for循环中，在if判断条件中，使用快捷键R即可将数值转换为字符，可以看到，是一个大小写转换的逻辑

来查看byte_4420B0



```
.data:004420B0 byte_4420B0     db 0Dh
.data:004420B1                 db  13h
.data:004420B2                 db  17h
.data:004420B3                 db  11h
.data:004420B4                 db    2
.data:004420B5                 db    1
.data:004420B6                 db  20h
.data:004420B7                 db  1Dh
.data:004420B8                 db  0Ch
.data:004420B9                 db    2
.data:004420BA                 db  19h
.data:004420BB                 db  2Fh ; /
.data:004420BC                 db  17h
.data:004420BD                 db  2Bh ; +
.data:004420BE                 db  24h ; $
.data:004420BF                 db  1Fh
.data:004420C0                 db  1Eh
.data:004420C1                 db  16h
.data:004420C2                 db    9
.data:004420C3                 db  0Fh
.data:004420C4                 db  15h
.data:004420C5                 db  27h ; '
.data:004420C6                 db  13h
.data:004420C7                 db  26h ; &
.data:004420C8                 db  0Ah
.data:004420C9                 db  2Fh ; /
.data:004420CA                 db  1Eh
.data:004420CB                 db  1Ah
.data:004420CC                 db  2Dh ; -
.data:004420CD                 db  0Ch
.data:004420CE                 db  22h ; "
.data:004420CF                 db    4
```

查看sub_4013C0()



```
int __cdecl sub_4013C0(int a1)
{
  return (a1 ^ 0x55) + 72;
}
```

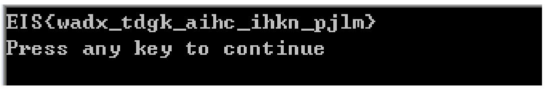
根据上述的分析，我们写如下代码

```
#include "stdafx.h"
#include <stdio.h>
#include <string.h>

int main(int argc, char* argv[])
{
 char s[]="GONDPHyGjPEKruv{{pj]X@rF";
 int v2=strlen(s);
 int byte_4420B0[]={0x0d,0x13,0x17,0x11,0x02,0x01,0x20,0x1d,
      0x0c,0x02,0x19,0x2f,0x17,0x2b,0x24,0x1f,
      0x1e,0x16,0x09,0x0f,0x15,0x27,0x13,0x26,
      0x0a,0x2f,0x1e,0x1a,0x2d,0x0c,0x22,0x04};

 for(int i=0;i<v2;i++)
 {
  s[i]=((s[i]^byte_4420B0[i])-72)^0x55;
  if(s[i]>='a' && s[i]<='z')
  {
   s[i]-=32;
   continue;
  }
  if(s[i]>='A' && s[i]<='Z')
  {
   s[i]+=32;
  }
 }
 printf("EIS{%s}\n",s);

 return 0;
}
```



```
EIS{wadx_tdgk_aihc_ihkn_pjlm}
Press any key to continue
```

EIS{wadx_tdgk_aihc_ihkn_pjlm}

加油：)