xctf 攻防世界 key

原创

pipixia233333 于 2019-04-24 20:16:36 发布 1134 收藏 2
分类专栏： 逆向之旅
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/qq_41071646/article/details/89501881
版权


逆向之旅 专栏收录该内容
128 篇文章 2 订阅
订阅专栏

好久没有去攻防世界玩了 主要是这段时间太忙了 有很多 杂事 缠身吧 然后没事看看堆 就。。 看自闭了

然后这道题其实还是很简单的 有很多函数 还有指令其实感觉 用od 或者ida 直接静态 就能查出来什么意思 就没有必要 去深究 里面的算法

点进去 主要的函数

```
  v47 = 0;
  v37 = 15;
  v36 = 0;
  LOBYTE(v35) = 0;
  LOBYTE(v47) = 1;
  v0 = 0;
  v42 = 'dime';
  LOWORD(v43) = 'a';
  *Memory = xmmword_40528C;                     // htadimehtadimeht
  v45 = '.<';
  v46 = 0;
  v44 = xmmword_4052A4;                         // <<<....++++---->
  do
  {
    sub_4021E0(&v35, 1u, (*(Memory + v0) ^ *(&v44 + v0)) + 22);
    ++v0;
  }
  while ( v0 < 18 );
  v1 = 0;
  v43 = 15;
  v42 = 0;
  LOBYTE(Memory[0]) = 0;
  LOBYTE(v47) = 2;
  v2 = v37;
  v3 = v35;
  do
  {
    v4 = &v35;
    if ( v2 >= 0x10 )
      v4 = v3;
    sub_4021E0(Memory, 1u, *(v4 + v1++) + 9);
  }
  while ( v1 < 18 );
  memset(&Dst, 0, 0xB8u);
  sub_401620(&Dst, v5, v6, v7, v8);
```

https://blog.csdn.net/qq_41071646

发现了这个东西 注意这个 Memory 这个东西 我们往下看

```
  v32 = 0;
  v29 = 0;
  std::basic_streambuf<char,std::char_traits<char>>::_Init(&v27);
  v30 = dword_408590;
  File = 0;
  v31 = dword_408594;
  v28 = 0;
  if ( !v10 )
    std::basic_ios<char,std::char_traits<char>>::setstate(&Dst + *(Dst + 4), 2, 0);
  v12 = Memory;
  if ( v43 >= 0x10 )
    v12 = Memory[0];
  v13 = sub_4020C0(&v38, v11, v39, v12, v42);
  v14 = std::cout;
  if ( v13 )
  {
    v22 = "=W=r=o=n=g=K=e=y=";
  }
  else
  {
    v15 = putss(std::cout, "|------------------------------|");
    std::basic_ostream<char,std::char_traits<char>>::operator<<(v15, sub_402C50);
    v16 = putss(std::cout, "|==============================|");
    std::basic_ostream<char,std::char_traits<char>>::operator<<(v16, sub_402C50);
```

这里 决定了我们的走向他就是参数之一 然后我们我们 发现 如果直接 运行程序 会失败 原因就是 他这个是 读取的 txt

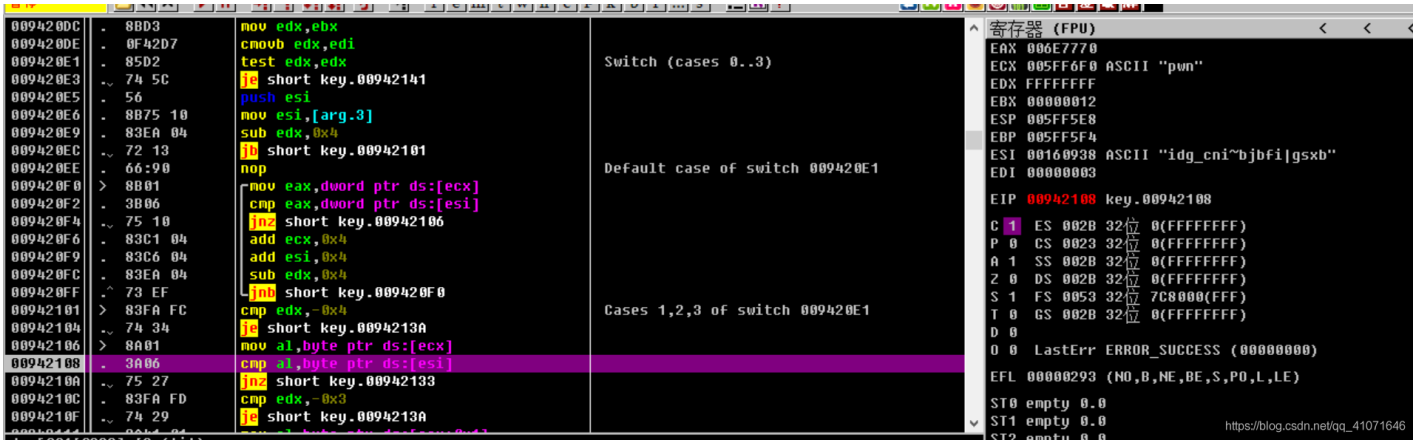
```
  std::codecvt_base *v7; // eax
  void (__thiscall ***v8)(_DWORD, signed int); // eax
  char v10; // [esp+Ch] [ebp-14h]
  int v11; // [esp+10h] [ebp-10h]
  int v12; // [esp+1Ch] [ebp-4h]

  v4 = this;
  if ( this[19] )
    return 0;
  v5 = std::_Fiopen("C:\\Users\\CSAW2016\\haha\\flag_dir\\flag.txt", 1, 64);
  if ( !v5 )
    return 0;
  sub_402430(v4, v5, 1);
  v6 = std::basic_streambuf<char,std::char_traits<char>>::getloc(v4, &v10);
  v12 = 0;
  v7 = sub_402C80(v6);
  if ( std::codecvt_base::always_noconv(v7) )
  {
    v4[14] = 0;
  }
  else
  {
    v4[14] = v7;
    std::basic_streambuf<char,std::char_traits<char>>::_Init(v4);
  }
  v12 = 1;
  if ( v11 )
  {
    v8 = (*(*v11 + 8))();
    if ( v8 )
      (**v8)(v8, 1);
  }
```

我们在路径下 直接建立一个文件就行 然后 写上pwn 然后我们分析一下sub_4020C0 这个函数

```
{
  v7 = a4;
  v9 = v6 < 4;
  v8 = v6 - 4;
  if ( v9 )
  {
LABEL_11:
    if ( v8 == -4 )
      goto LABEL_20;
  }
  else
  {
    while ( *this == *v7 )
    {
      ++this;
      v7 += 4;
      v9 = v8 < 4;
      v8 -= 4;
      if ( v9 )
        goto LABEL_11;
    }
  }
  v9 = *this < *v7;
  if ( *this != *v7
    || v8 != -3
    && ((v10 = *(this + 1), v9 = v10 < *(v7 + 1), v10 != *(v7 + 1))
     || v8 != -2
     && ((v11 = *(this + 2), v9 = v11 < *(v7 + 2), v11 != *(v7 + 2))
      || v8 != -1 && (v12 = *(this + 3), v9 = v12 < *(v7 + 3), v12 != *(v7 + 3)))) )
  {
    result = -v9 | 1;
    goto LABEL_21;
  }
LABEL_20:
```

https://blog.csdn.net/qq_41071646

这里的v7 应该就是我们的 memory 但和他比较的 并不是很确定是不是我们的 file文件 我们用od 来调试一下



emmmmm 直接出来flag 了 如果想 用程序解密的话 下面是脚本

idg_cni~bjbfi|gsxb

```
str1 = "themidathemidathemida"
str2 = ">----++++....<<<<."

key =""
flag=""
for i in range(18):
 key += chr((ord(str1[i]) ^ ord(str2[i]))+22)
for i in key:
 flag+=chr(ord(i)+9)


print(flag)
```

这个题 还是很简单的啊