BUUCTF reverse_3

原创

doudoudedi 于 2019-06-28 14:59:28 发布 1045 收藏

分类专栏： 题目 逆向 文章标签： BUUCTF

本文链接：https://blog.csdn.net/qq_37433000/article/details/94002835

题目 同时被 2 个专栏收录

83 篇文章 2 订阅

订阅专栏

逆向

2 篇文章 0 订阅

订阅专栏

今天考计算机组成原理被老师一顿讲，诶还是自己太菜了

我感觉都要挂科了~~~~~

中午做了一个逆向分享一下

main函数:

```
__int64 main_0()
{
  int v0; // eax
  const char *v1; // eax
  size_t v2; // eax
  int v3; // edx
  __int64 v4; // ST08_8
  signed int j; // [esp+DCh] [ebp-ACh]
  signed int i; // [esp+E8h] [ebp-A0h]
  signed int v8; // [esp+E8h] [ebp-A0h]
  char Dest[108]; // [esp+F4h] [ebp-94h]
  char Str; // [esp+160h] [ebp-28h]
  char v11; // [esp+17Ch] [ebp-Ch]

  for ( i = 0; i < 100; ++i )
  {
    if ( (unsigned int)i >= 0x64 )
      j____report_rangecheckfailure();
    Dest[i] = 0;
  }
  sub_41132F((int)"please enter the flag:");
  sub_411375("%20s", &Str);
  v0 = j_strlen(&Str);
  v1 = (const char *)sub_4110BE((int)&Str, v0, (int)&v11);
  strncpy(Dest, v1, '(');
  v8 = j_strlen(Dest);
  for ( j = 0; j < v8; ++j )
    Dest[j] += j;
  v2 = j_strlen(Dest);
  if ( !strncmp(Dest, Str2, v2) )
    sub_41132F((int)"rigth flag!\n");
  else
    sub_41132F((int)"wrong flag!\n");
  HIDWORD(v4) = v3;
  LODWORD(v4) = 0;
  return v4;
}
```

我们就可以看出只要你输入的字符串经过一个函数编码，然后，在根据长度循环每一个加上其长度数（从**0**开始）,然后和**str2**比配相同就是**flag**
我们再看看那个函数我是没看懂啥但我从字符串中看见了**base64**，然后还有编码表所以应该就是**base64**
所以上脚本

```
import base64
import binascii
str1='e3nifIH9b_C@n@dH'
flag=''
for i in range(0,len(str1)):
 print binascii.b2a_hex(str1[i])
 #print int(binascii.b2a_hex(str1[i]),16)
 flag+=chr(int(binascii.b2a_hex(str1[i]),16)-i)
print base64.b64decode(flag)
```

这里再讲讲binascii.b2a_hex()这个函数
作用是先把字符串转换成二进制数据然后在用十六进制表示,int(‘字符串’,‘字符串的进制’)
flag到手~~~