REVERSE-PRACTICE-BUUCTF-27

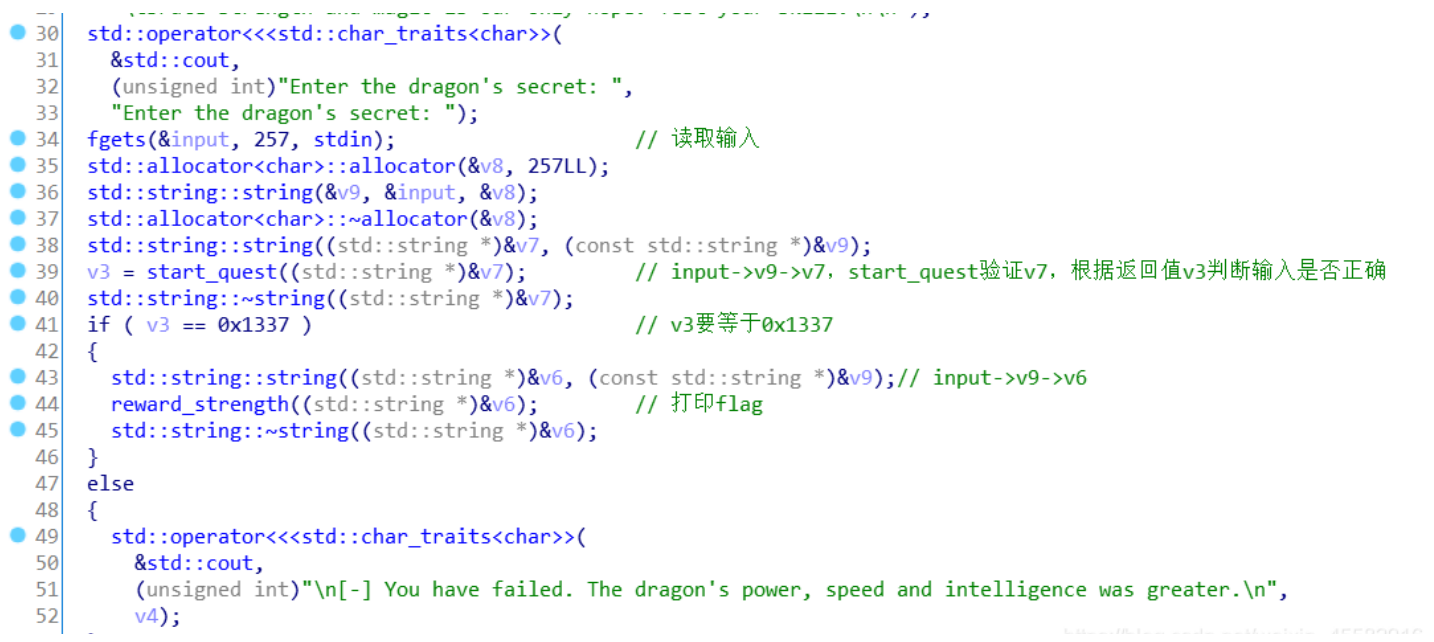
[P1umH0](https://blog.csdn.net/weixin_45582916) 于 2021-02-27 14:26:19 发布 109 收藏
分类专栏： [Reverse-BUUCTF](#)

[Reverse-BUUCTF 专栏收录该内容](#)
32 篇文章 3 订阅
订阅专栏

REVERSE-PRACTICE-BUUCTF-27

[XMAN2018排位赛]Dragon Quest

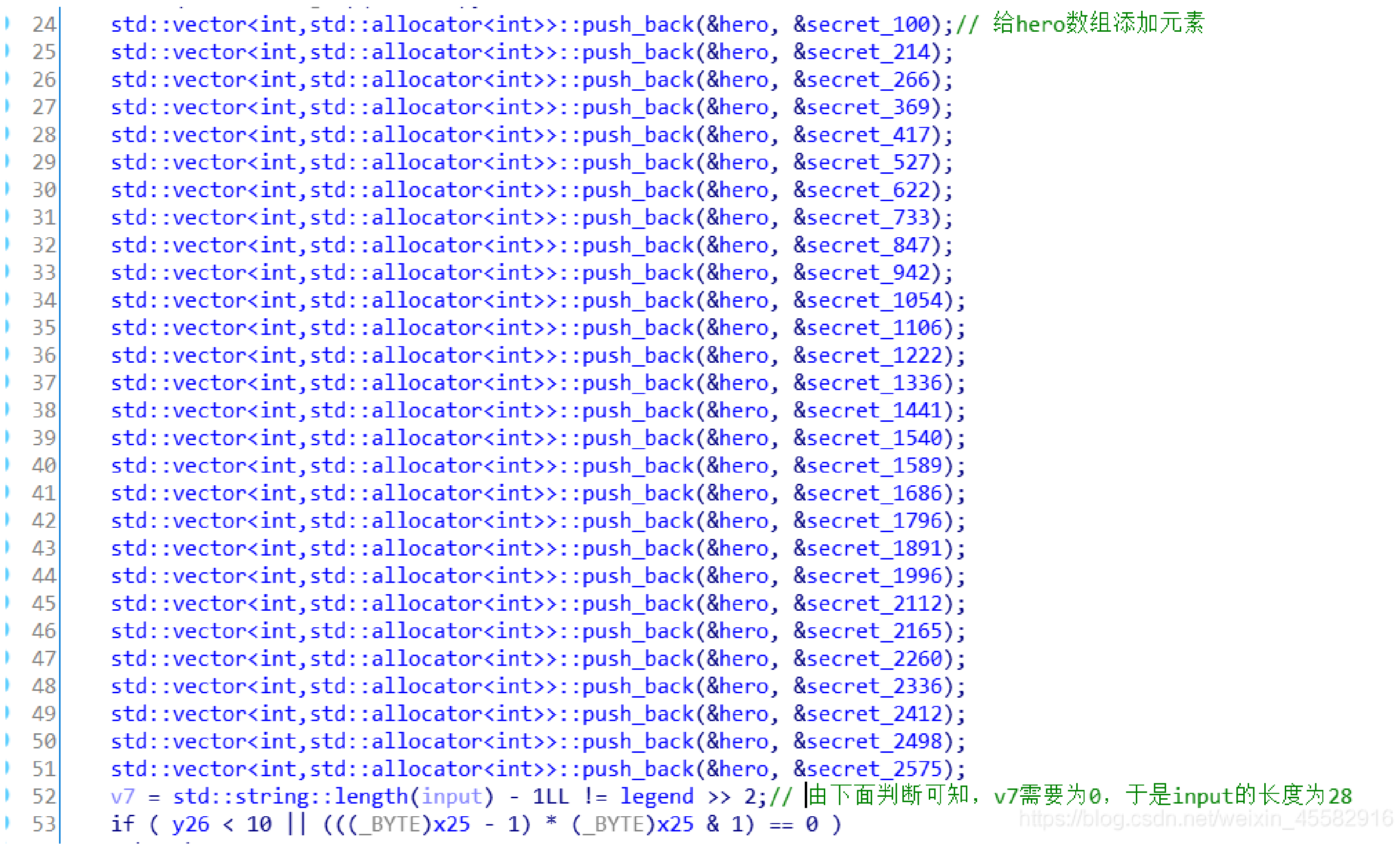
elf文件，无壳，ida分析
main函数，读取输入，start_quest函数验证输入，根据返回值判断输入是否正确



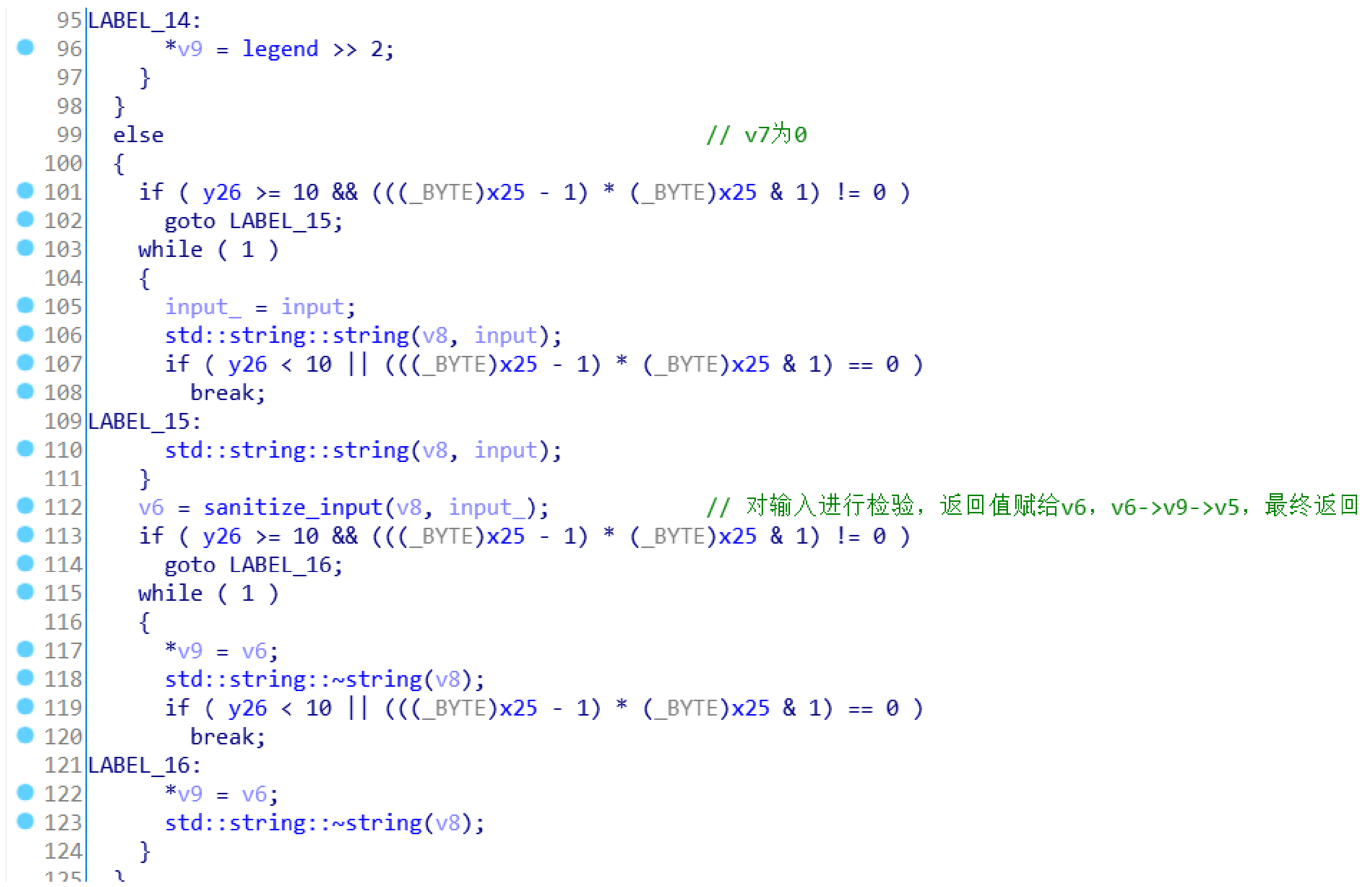
```
std::operator<<<std::char_traits<char>>(
  &std::cout,
  (unsigned int)"Enter the dragon's secret: ",
  "Enter the dragon's secret: ");
fgets(&input, 257, stdin);                    // 读取输入
std::allocator<char>::allocator(&v8, 257LL);
std::string::string(&v9, &input, &v8);
std::allocator<char>::~allocator(&v8);
std::string::string((std::string *)&v7, (const std::string *)&v9);
v3 = start_quest((std::string *)&v7);         // input->v9->v7，start_quest验证v7，根据返回值v3判断输入是否正确
std::string::~string((std::string *)&v7);
if ( v3 == 0x1337 )                           // v3要等于0x1337
{
  std::string::string((std::string *)&v6, (const std::string *)&v9);// input->v9->v6
  reward_strength((std::string *)&v6);        // 打印flag
  std::string::~string((std::string *)&v6);
}
else
{
  std::operator<<<std::char_traits<char>>(
    &std::cout,
    (unsigned int)"\n[-] You have failed. The dragon's power, speed and intelligence was greater.\n",
    v4);
```

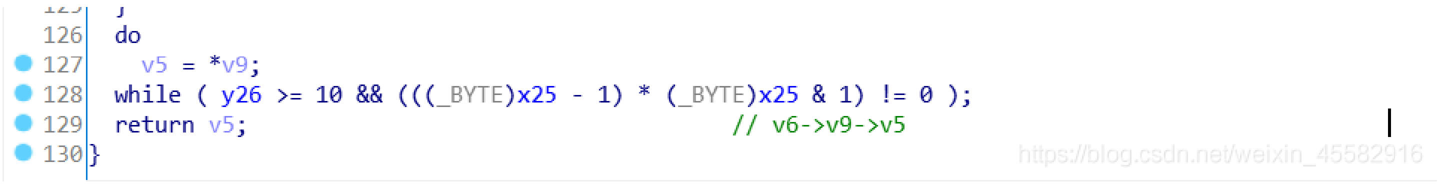
```
}
```

进入start_quest函数，首先是给hero数组添加元素，检验输入的长度是否为28，输入长度等于28则v7为0，否则v7为1


```
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_100);// 给hero数组添加元素
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_214);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_266);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_369);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_417);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_527);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_622);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_733);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_847);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_942);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_1054);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_1106);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_1222);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_1336);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_1441);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_1540);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_1589);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_1686);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_1796);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_1891);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_1996);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_2112);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_2165);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_2260);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_2336);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_2412);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_2498);
  std::vector<int,std::allocator<int>>::push_back(&hero, &secret_2575);
  v7 = std::string::length(input) - 1LL != legend >> 2;// 由下面判断可知，v7需要为0，于是input的长度为28
  if ( y26 < 10 || (((_BYTE)x25 - 1) * (_BYTE)x25 & 1) == 0 )
```

往下走，由于需要start_quest返回0x1337，则需v7为0，即输入的长度等于28
sanitize_input函数对输入进行检验，由变量值传递可知，sanitize_input函数也要返回0x1337


```
LABEL_14:
      *v9 = legend >> 2;
    }
  }
  else                                        // v7为0
  {
    if ( y26 >= 10 && (((_BYTE)x25 - 1) * (_BYTE)x25 & 1) != 0 )
      goto LABEL_15;
    while ( 1 )
    {
      input_ = input;
      std::string::string(v8, input);
      if ( y26 < 10 || (((_BYTE)x25 - 1) * (_BYTE)x25 & 1) == 0 )
        break;
LABEL_15:
      std::string::string(v8, input);
    }
    v6 = sanitize_input(v8, input_);          // 对输入进行检验，返回值赋给v6，v6->v9->v5，最终返回
    if ( y26 >= 10 && (((_BYTE)x25 - 1) * (_BYTE)x25 & 1) != 0 )
      goto LABEL_16;
    while ( 1 )
    {
      *v9 = v6;
      std::string::~string(v8);
      if ( y26 < 10 || (((_BYTE)x25 - 1) * (_BYTE)x25 & 1) == 0 )
        break;
LABEL_16:
      *v9 = v6;
      std::string::~string(v8);
    }
  }
```



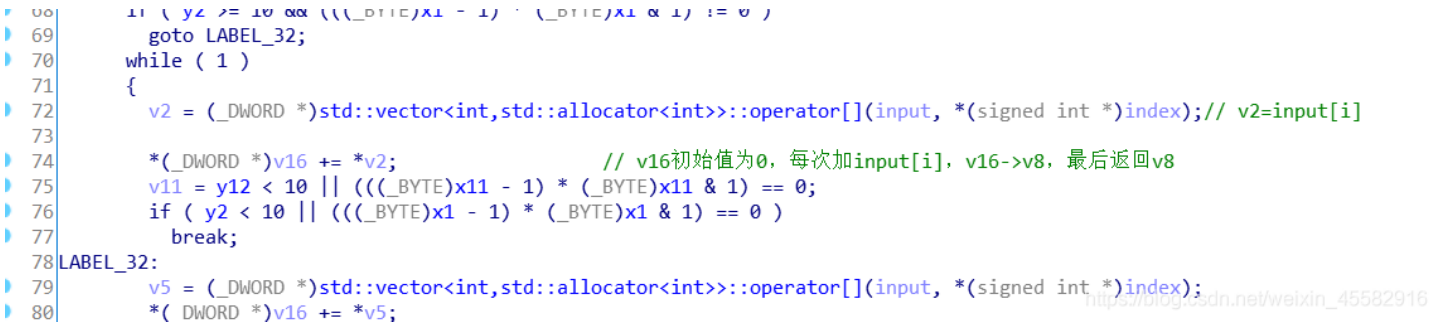
```
}
do
  v5 = *v9;
while ( y26 >= 10 && (((_BYTE)x25 - 1) * (_BYTE)x25 & 1) != 0 );
return v5;                                  // v6->v9->v5
}
```

进入sanitize_input函数，主要的逻辑为，输入进入transform_input函数处理，返回值与hero数组比较

```
    v31 = (char *)std::string::operator[](input, index);// 从input中取一个字节
    if ( y18 >= 10 && (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) != 0 )
    {
LABEL_71:
      if ( y4 >= 10 && (((_BYTE)x3 - 1) * (_BYTE)x3 & 1) != 0 )
        goto LABEL_114;
      while ( 1 )
      {
        *(_DWORD *)v40 = *v31;
        if ( y4 < 10 || (((_BYTE)x3 - 1) * (_BYTE)x3 & 1) == 0 )
          break;
LABEL_114:
        *(_DWORD *)v40 = *v31;
      }
    }
    *(_DWORD *)v40 = *v31;                      // v31->v40
    if ( y18 >= 10 && (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) != 0 )
      goto LABEL_71;
    while ( y4 >= 10 && (((_BYTE)x3 - 1) * (_BYTE)x3 & 1) != 0 )
      ;
    std::vector<int,std::allocator<int>>::push_back(v42, v40);// v40被添加到v42数组
    do
      v30 = y18 < 10 || (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) == 0;
    while ( y4 >= 10 && (((_BYTE)x3 - 1) * (_BYTE)x3 & 1) != 0 );
    if ( !v30 )
LABEL_74:
      *v37 = *v41;
    if ( y4 >= 10 && (((_BYTE)x3 - 1) * (_BYTE)x3 & 1) != 0 )
LABEL_99:
      *v37 = *v41;
    v1 = v37;
    *v37 = *v41;
    v29 = *v1;
    v28 = y18 < 10 || (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) == 0;
    if ( y4 >= 10 && (((_BYTE)x3 - 1) * (_BYTE)x3 & 1) != 0 )
      goto LABEL_99;
    if ( !v28 )
      goto LABEL_74;
    v27 = std::string::length(input);
    do
      v26 = y18 < 10 || (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) == 0;
    while ( y4 >= 10 && (((_BYTE)x3 - 1) * (_BYTE)x3 & 1) != 0 );
    if ( !v26 )
LABEL_75:
      *v37 = (v27 >> 40) & v29 | 0x1C;
    v2 = v37;
    *v37 = (v27 >> 40) & v29 | 0x1C;
    v25 = *v2 != 0;
    if ( y18 >= 10 && (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) != 0 )
      goto LABEL_75;
    while ( y4 >= 10 && (((_BYTE)x3 - 1) * (_BYTE)x3 & 1) != 0 )
```

```
      ;
    if ( v25 )
    {
      do
        index_ = *v41;
      while ( y18 >= 10 && (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) != 0 );
      v23 = (int *)std::vector<int,std::allocator<int>>::operator[]((unsigned int)&hero, index_);// 从hero中取一个字节
      do
        v22 = y18 < 10 || (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) == 0;
      while ( y4 >= 10 && (((_BYTE)x3 - 1) * (_BYTE)x3 & 1) != 0 );
      do
        v21 = *v23;                           // v23->v21
      while ( y18 >= 10 && (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) != 0 );
      std::vector<int,std::allocator<int>>::vector(v36, v42);// v42赋给v36
      do
        v20 = y18 < 10 || (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) == 0;
      while ( y4 >= 10 && (((_BYTE)x3 - 1) * (_BYTE)x3 & 1) != 0 );
      while ( y18 >= 10 && (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) != 0 )
        ;
      while ( y4 >= 10 && (((_BYTE)x3 - 1) * (_BYTE)x3 & 1) != 0 )
        ;
      v19 = transform_input(v36);             // 对v36处理，返回到v19
      if ( y18 >= 10 && (((_BYTE)x17 - 1) * (_BYTE)x17 & 1) != 0 )
        goto LABEL_79;
      while ( 1 )
      {
        v18 = v21 == v19;                     // v19与v21比较
```

进入transform_input函数，主要的逻辑为，取出输入的一个字节input[i]，v16初始值为0，v16每次加上input[i]，然后返回v16，与hero数组的元素比较



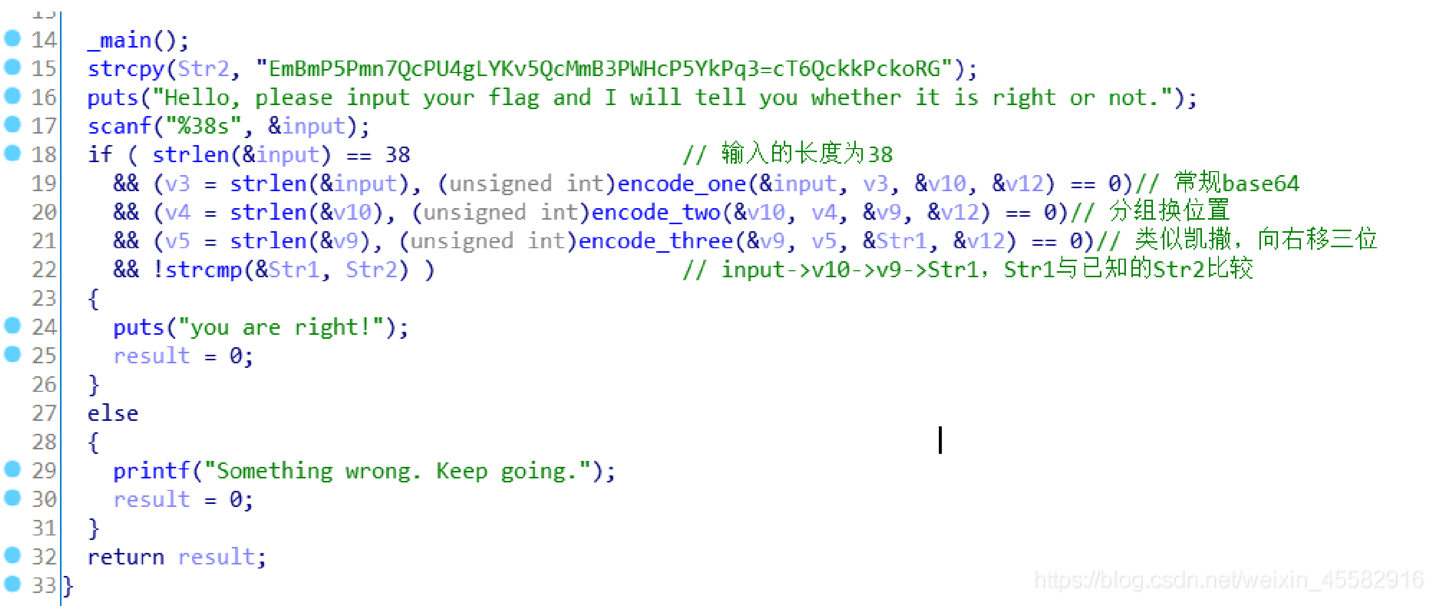
```
    if ( y2 >= 10 && (((_BYTE)x1 - 1) * (_BYTE)x1 & 1) != 0 )
      goto LABEL_32;
    while ( 1 )
    {
      v2 = (_DWORD *)std::vector<int,std::allocator<int>>::operator[](input, *(signed int *)index);// v2=input[i]

      *(_DWORD *)v16 += *v2;                  // v16初始值为0，每次加input[i]，v16->v8，最后返回v8
      v11 = y12 < 10 || (((_BYTE)x11 - 1) * (_BYTE)x11 & 1) == 0;
      if ( y2 < 10 || (((_BYTE)x1 - 1) * (_BYTE)x1 & 1) == 0 )
        break;
LABEL_32:
      v5 = (_DWORD *)std::vector<int,std::allocator<int>>::operator[](input, *(signed int *)index);
      *( DWORD *)v16 += *v5;
```

已知hero数组，写逆运算脚本即可得到flag

```python
hero=[0x64,0xd6,0xa,0x71,0xa1,0xf,0x6e,0xdd,0x4f,0xae,
      0x1e,0x52,0xc6,0x38,0xa1,0x4,0x35,0x96,0x4,0x63,
      0xcc,0x40,0x75,0xd4,0x20,0x6c,0xc2,0xf]
n=0
flag=""
for i in range(len(hero)):
    tmp=hero[i]-n
    n+=tmp
    flag+=chr(tmp%128)
print(flag)
#dr4g0n_or_p4tric1an_it5_LLVM
```

[羊城杯 2020]easyre

exe程序，运行后输入，无壳，ida分析
main函数，读取输入，检验输入的长度是否为38，对输入进行三次变换，最后与Str2比较



```
  _main();
  strcpy(Str2, "EmBmP5Pmn7QcPU4gLYKv5QcMmB3PWHcP5YkPq3=cT6QckkPckoRG");
  puts("Hello, please input your flag and I will tell you whether it is right or not.");
  scanf("%38s", &input);
  if ( strlen(&input) == 38                       // 输入的长度为38
    && (v3 = strlen(&input), (unsigned int)encode_one(&input, v3, &v10, &v12) == 0)// 常规base64
    && (v4 = strlen(&v10), (unsigned int)encode_two(&v10, v4, &v9, &v12) == 0)// 分组换位置
    && (v5 = strlen(&v9), (unsigned int)encode_three(&v9, v5, &Str1, &v12) == 0)// 类似凯撒，向右移三位
    && !strcmp(&Str1, Str2) )                     // input->v10->v9->Str1，Str1与已知的Str2比较
  {
    puts("you are right!");
    result = 0;
  }
  else
  {
    printf("Something wrong. Keep going.");
    result = 0;
  }
  return result;
}
```

三次变换都很容易理解，分别是常规base64，分组换位置，以及类似凯撒的右移三位
写逆运算脚本即可得到flag

```python
import base64
str2="EmBmP5Pmn7QcPU4gLYKv5QcMmB3PWHcP5YkPq3=cT6QckkPckoRG"
data=[]
for c in str2:
    if c.isdigit():
        data.append((ord(c)-48-3)%10+48)
    elif c.isupper():
        data.append((ord(c)-65 - 3) % 26 + 65)
    elif c.islower():
        data.append((ord(c)-97 - 3) % 26 + 97)
    else:
        data.append(ord(c))
flag=[0]*len(data)
flag[0:13]=data[13:26]
flag[13:26]=data[39:len(data)]
flag[26:39]=data[0:13]
flag[39:len(flag)]=data[26:39]
print(base64.b64decode(''.join(chr(i) for i in flag)))
# GWHT{672cc4778a38e80cb362987341133ea2}
```

[watevrCTF 2019]Repyc

.pyc文件，用uncompyle6反编译得到源代码，python2会检测为非ascii码，换成python3即可

```python
佤 = 0
侰 = ~佤 * ~佤
俴 = 侰 + 侰

def 䯂(䵦):
    굴 = 佤
    굿 = 佤
    괠 = [佤] * 俴 ** (俴 * 俴)
    궓 = [佤] * 100
    괣 = []
    while 䵦[굴][佤] != '듃':
```

```
    귟 = 䵫[글][低].lower()
    亀 = 䵫[글][㑺:]
    if 귟 == '뷀':
        괠[亀[低]] = 괠[亀[㑺]] + 괠[亀[俊]]
    else:
        if 귟 == '렀':
            괠[亀[低]] = 괠[亀[㑺]] ^ 괠[亀[俊]]
        else:
            if 귟 == '렳':
                괠[亀[低]] = 괠[亀[㑺]] - 괠[亀[俊]]
            else:
                if 귟 == '냻':
                    괠[亀[低]] = 괠[亀[㑺]] * 괠[亀[俊]]
                else:
                    if 귟 == '뢜':
                        괠[亀[低]] = 괠[亀[㑺]] / 괠[亀[俊]]
                    else:
                        if 귟 == '룳':
                            괠[亀[低]] = 괠[亀[㑺]] & 괠[亀[俊]]
                        else:
                            if 귟 == '맳':
                                괠[亀[低]] = 괠[亀[㑺]] | 괠[亀[俊]]
                            else:
                                if 귟 == '괡':
                                    괠[亀[低]] = 괠[亀[低]]
                                else:
                                    if 귟 == '뫗':
                                        괠[亀[低]] = 괠[亀[㑺]]
                                    else:
                                        if 귟 == '곐':
                                            괠[亀[低]] = 亀[㑺]
                                        else:
                                            if 귟 == '뫨':
                                                귋[亀[低]] = 괠[亀[㑺]]
                                            else:
                                                if 귟 == '듷':
                                                    괠[亀[低]] = 귋[亀[㑺]]
                                                else:
                                                    if 귟 == '뎞':
                                                        괠[亀[低]] = 低
                                                    else:
                                                        if 귟 == '뭡':
                                                            귋[亀[低]] = 低
                                                        else:
                                                            if 귟 == '몯':
                                                                괠[亀[低]] = input(괠[亀[㑺]])
                                                            else:
                                                                if 귟 == '꾗':
                                                                    귋[亀[低]] = input(괠[亀[㑺]])
                                                                else:
                                                                    if 귟 == '돯':
                                                                        print(괠[亀[低]])
                                                                    else:
                                                                        if 귟 == '뭘':
                                                                            print(귋[亀[低]])
                                                                        else:
                                                                            if 귟 == '뭿':
                                                                                글 = 괠[亀[低]]
                                                                            else:
                                                                                   if 굸 == '뮓':
                                                                                       굴 = 귋[龟[佤]]
                                                                                   else:
                                                                                       if 굸 == '뭃':
                                                                                           굴 = 괿.pop()
                                                                                       else:
                                                                                           if 굸 == '뭇':
                                                                                               if 괼[龟[𠇹]] >
괼[龟[俴]]:
                                                                                                   굴 = 龟[佤]
                                                                                                   괿.append(굴
)

                                                                                                   continue
                                                                                           else:
                                                                                               if 굸 == '꿆':
                                                                                                   괼[7] = 佤
                                                                                                   for i in ran
ge(len( 괼[龟[佤]])):
                                                                                                       if 괼[龟
[佤]] != 괼[龟[𠇹]]:
                                                                                                           괼[7
] = 𠇹
                                                                                                           굴 =
 괼[龟[ 俴]]
                                                                                                           괿.a
ppend(굴)

                                                                                               else:
                                                                                                   if 굸 == '꿊
':
                                                                                                       괾 = ''
                                                                                                       for i in
 range(len(괼[龟[佤]])):
                                                                                                           괾 +
= chr(ord(괼[龟[佤]][i]) ^ 괼[龟[𠇹]])

                                                                                                       괼[龟[佤
]] = 괾
                                                                                                   else:
                                                                                                       if 굸 ==
 '귏':
                                                                                                           괾 =
 ''
                                                                                                           for
i in range(len(괼[龟[佤]])):

괾 += chr(ord(괼[龟[佤]][i]) - 괼[龟[𠇹]])

                                                                                                           괼[
龟[佤]] = 괾
                                                                                                       else:
                                                                                                           if
굸 == '떇':

if 괼[龟[𠇹]] > 괼[龟[俴]]:

    굴 = 괼[龟[佤]]

    괿.append(굴)
    continue
                                                                                                                    else
:

if 귟 == '�':

    if  괠[亀[�]] > 괠[亀[俴]]:

        굴 = 궶[亀[仾]]

        괩.append(굴)

        continue

else:

    if  귟 == '뙵':

        if 괠[亀[偖]] == 괠[亀[俴]]:

            굴 = 亀[仾]

            괩.append(굴)

            continue

    else:

        if 귟 == '뚫':

            if 괠[亀[偖]] == 괠[亀[俴]]:

                굴 = 괠[亀[仾]]

                괩.append(굴)

                continue

        else:

            if 귟 == '뛿':

                if 괠[亀[偖]] == 괠[亀[俴]]:

                    굴 = 궶[亀[仾]]

                    괩.append(굴)

                    continue
        굴 += 偖


爨([
 [
  '꼙', 仾, 'Authentication token: '],
 [
  '�', 仾, 仾],
 [
  '꼙', 6, 'á×äÓâæíäàßåÉÛãåäÉÖÓÉäàÓÉÖÓåäÉÓÚÕæïèäßÙÚÉÛÓäàÙÔÉÓâæÉàÓÚÕÓÒÙæääÉäàßåÉßåÉäàÓÉÚÓáÉ·Ôâ×ÚÕÓÔÉ³ÚÕæïèäßÙÚÉÅä
×ÚÔ ×æÔÉ×ÚïáxïåÉßÉÔÙÚäÉæÓ×ÜÜïÉà×âÓÉ×ÉÑÙÙÔÉâßÔÉÖãäÉßÉæÓ×ÜÜïÉÓÚÞÙïÉäàßåÉåÙÚÑÉßÉàÙèÓÉïÙãÉáßÜÜÉÓÚÞÙïÉßäÉ×ååÓÜÜ\x97ÉïÙããäÖÓ\x9aÕÙÛ\x99á×äÕà©â«³£ï²ÕÔÈ·±â¨ë'],
 [
  '겧', 俴, 俴 ** (3 * 俴 + 倨) - 俴 ** (俴 + 倨)],
 [
  '겧', 4, 15],
 [
  '겧', 3, 倨],
 [
  '냃', 俴, 俴, 3],
 [
  '뉂', 俴, 俴, 4],
 [
  '괡', 伵, 俴],
 [
  '댒', 3],
 [
  '꾮', 6, 3],
 [
  '겧', 伵, 'Thanks.'],
 [
  '겧', 倨, 'Authorizing access...'],
 [
  '듃', 伵],
 [
  '딇', 伵, 伵],
 [
  '꾮', 伵, 俴],
 [
  '꿚', 伵, 4],
 [
  '겧', 5, 19],
 [
  '꾯', 伵, 6, 5],
 [
  '듃', 倨],
 [
  '듡'],
 [
  '겧', 倨, 'Access denied!'],
 [
  '듃', 倨],
 [
  '듡']])
```

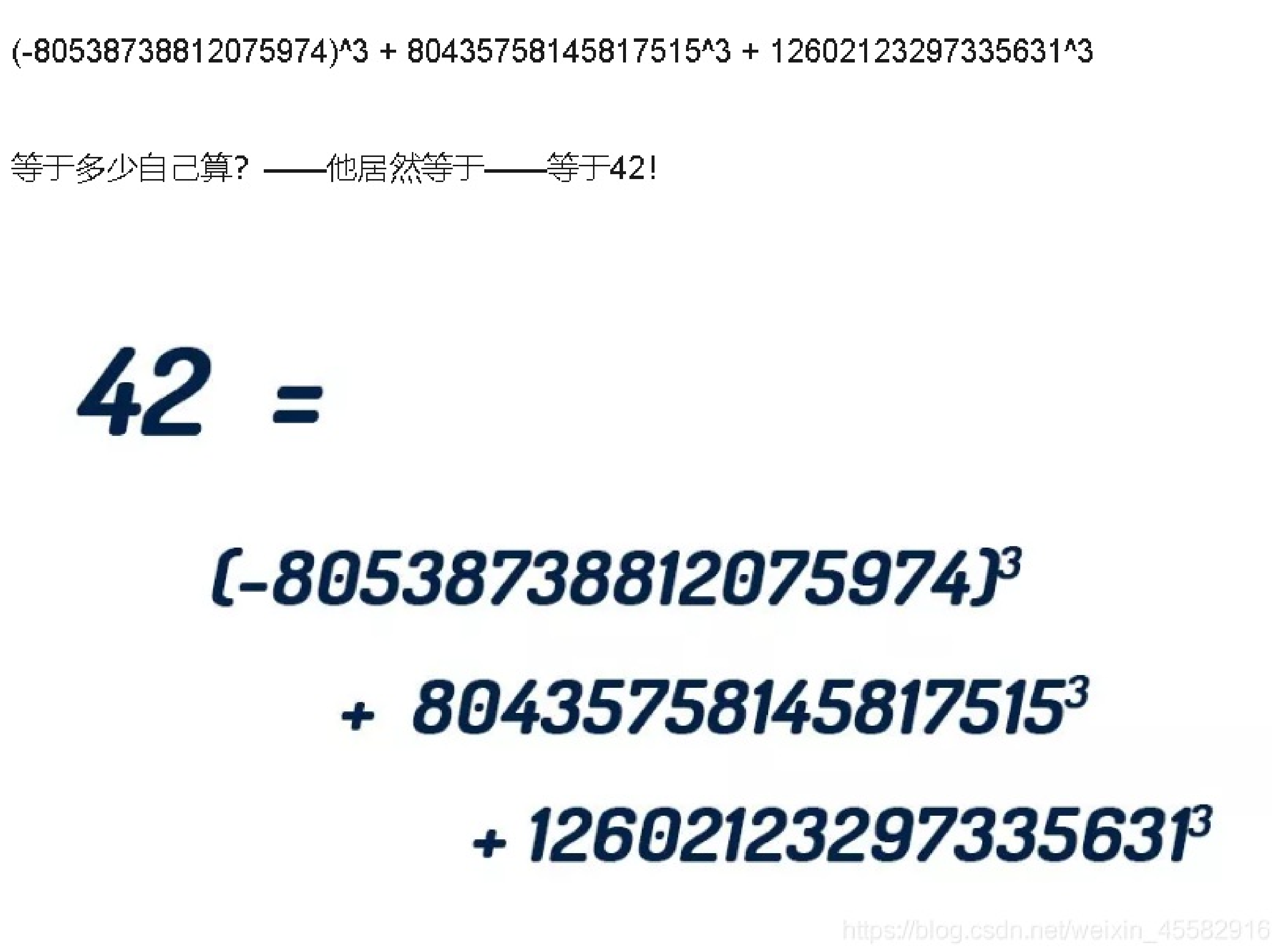
运行后输入，调试发现，对输入的处理很简单，input[i]=((input[i])^135)-15，即输入先异或135，再减去15，最后和那段长字符串比较，写脚本即可得到flag

```
res="á×äÓâæíäàßåÉÛãååäÉÖÓÉäàÓÉÖÓåäÉÓÚÕæïèäßÙÚÉÛÓäàÙÔÉÓâæÉàÓÚÕÓÒÙæäàÉäàßåÉßåÉäàÓÉÚÓáÉ·Ôâ×ÚÕÓÔÉ³ÚÕæïèäßÙÚÉÅä×ÚÔ ×æÔÉ×ÚïáxïåÉßÉÔÙÚäÉæÓ×ÜÜïÉà×âÓÉ×ÉÑÙÙÔÉâßÔÉÖãäÉßÉæÓ×ÜÜïÉÓÚÞÙïÉäàßåÉåÙÚÑÉßÉàÙèÓÉïÙãÉáßÜÜÉÓÚÞÙïÉßäÉ×ååÓÜÜ\x97ÉïÙããäÖÓ\x9aÕÙÛ\x99á×äÕà©â«³£ï²ÕÔÈ·±â¨ë"
flag=""
for c in res:
    flag+=chr((ord(c)+15)^135)
print(flag)
#watevr{this_must_be_the_best_encryption_method_evr_henceforth_this_is_the_new_Advanced_Encryption_Stand¨ard_anyways_i_dont_really_have_a_good_vid_but_i_really_enjoy_this_song_i_hope_you_will_enjoy_it_aswell!_youtube.com/watch?v=E5yFcdPAGv0}
```

[2019红帽杯]calc

exe程序，运行后输入，无壳，ida分析
三次输入，对输入一顿运算，没看懂
参考网上别的师傅的wp，2019红帽杯 Writeup by X1cT34m
原来是在满足input_2<input_1<input_3的条件下，得到`input_1**3+input_2**3+input_2**3==42`，即三个整数的立方和等于42
百度一下，果然有解

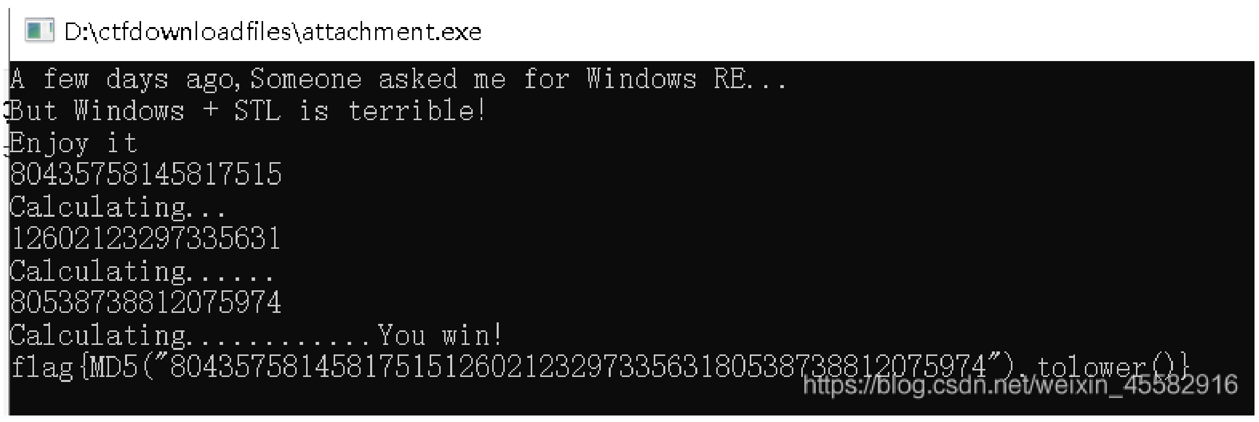
```
(-80538738812075974)**3 + 80435758145817515**3 + 12602123297335631**3==42
```


(-80538738812075974)^3 + 80435758145817515^3 + 12602123297335631^3

等于多少自己算？——他居然等于——等于42!

$$42 = (-80538738812075974)^3 + 80435758145817515^3 + 12602123297335631^3$$

将程序的三个sleep函数patch掉，按input_2<input_1<input_3的条件输入，得到flag


```
D:\ctfdownloadfiles\attachment.exe
A few days ago,Someone asked me for Windows RE...
But Windows + STL is terrible!
Enjoy it
80435758145817515
Calculating...
12602123297335631
Calculating......
80538738812075974
Calculating............You win!
flag{MD5("804357581458175151260212329733563180538738812075974").tolower()}
```

```python
import hashlib
flag="flag{"
s="804357581458175151260212329733563180538738812075974"
h=hashlib.md5()
h.update(s.encode(encoding='utf-8'))
flag+=h.hexdigest()
flag+="}"
print(flag)
# flag{951e27be2b2f10b7fa22a6dc8f4682bd}
```