2022-3-18

原创

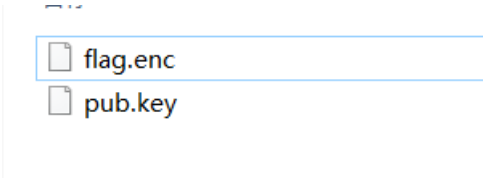
无名函数 于 2022-03-18 21:07:26 发布 205 收藏

分类专栏： Buu-re 文章标签： python 算法

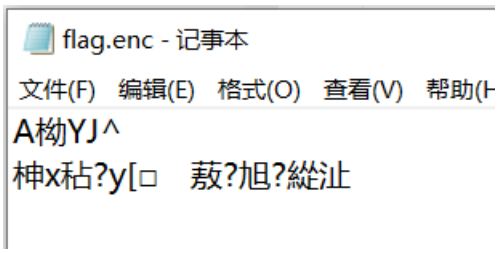
本文链接：https://blog.csdn.net/m0_57291352/article/details/123582471

rsa

ras是一个非常神秘的算法，那么它神秘在哪里 请少侠自己摸索！ 注意：得到的 flag 请包上 flag{} 提交




```
-----BEGIN PUBLIC KEY-----
MDwwDQYJKoZIhvcNAQEBBQADKwAwKAIhAMAzLFxkrkcYL2wch21CM2kQVFpY9+7+
/AvKr1rzQczdAgMBAAE=
-----END PUBLIC KEY-----
```



这是道密码题吧

公钥解析一下，然后分解个n，然后常规解就可

注意一下，flag.enc文件直接open打开的话会报错：`UnicodeDecodeError: 'gbk' codec can't decode byte 0xd7 in position 13: illegal multibyte sequence`，所以要加上encoding

但是utf-8的话同样会报错：`UnicodeDecodeError: 'utf-8' codec can't decode byte 0x96 in position 1: invalid start byte`

只有unicode_escape不会报错，但是这样出来的c是str型的不是int

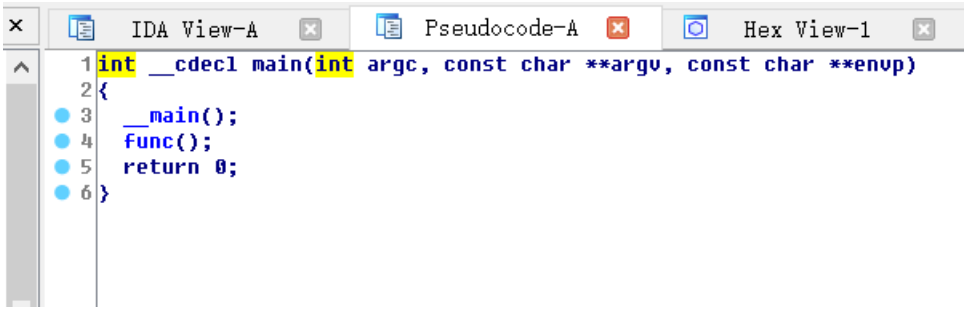
所以太麻烦了，还是

发现我之前做过这道题，在刷密码题的时候，重温一下上次解题过程

好，下一题吧

[ACTF新生赛2020]rome

IDA打开
找到main函数



点近func()

```
int func()
{
  int result; // eax@1
  int v1; // [sp+14h] [bp-44h]@8
  int v2; // [sp+18h] [bp-40h]@8
  int v3; // [sp+1Ch] [bp-3Ch]@8
  int v4; // [sp+20h] [bp-38h]@8
  unsigned __int8 v5; // [sp+24h] [bp-34h]@1
  unsigned __int8 v6; // [sp+25h] [bp-33h]@2
  unsigned __int8 v7; // [sp+26h] [bp-32h]@3
  unsigned __int8 v8; // [sp+27h] [bp-31h]@4
  unsigned __int8 v9; // [sp+28h] [bp-30h]@5
  int v10; // [sp+29h] [bp-2Fh]@8
  int v11; // [sp+2Dh] [bp-2Bh]@8
  int v12; // [sp+31h] [bp-27h]@8
  int v13; // [sp+35h] [bp-23h]@8
  unsigned __int8 v14; // [sp+39h] [bp-1Fh]@7
  char v15; // [sp+3Bh] [bp-1Dh]@1
  char v16; // [sp+3Ch] [bp-1Ch]@1
  char v17; // [sp+3Dh] [bp-1Bh]@1
  char v18; // [sp+3Eh] [bp-1Ah]@1
  char v19; // [sp+3Fh] [bp-19h]@1
  char v20; // [sp+40h] [bp-18h]@1
  char v21; // [sp+41h] [bp-17h]@1
  char v22; // [sp+42h] [bp-16h]@1
  char v23; // [sp+43h] [bp-15h]@1
  char v24; // [sp+44h] [bp-14h]@1
  char v25; // [sp+45h] [bp-13h]@1
  char v26; // [sp+46h] [bp-12h]@1
  char v27; // [sp+47h] [bp-11h]@1
  char v28; // [sp+48h] [bp-10h]@1
  char v29; // [sp+49h] [bp-Fh]@1
  char v30; // [sp+4Ah] [bp-Eh]@1
  char v31; // [sp+4Bh] [bp-Dh]@1
  int i; // [sp+4Ch] [bp-Ch]@8

  v15 = 81;
  v16 = 115;
  v17 = 119;
  v18 = 51;
```

```
  v19 = 115;
  v20 = 106;
  v21 = 95;
  v22 = 108;
  v23 = 122;
  v24 = 52;
  v25 = 95;
  v26 = 85;
  v27 = 106;
  v28 = 119;
  v29 = 64;
  v30 = 108;
  v31 = 0;
  printf("Please input:");
  scanf("%s", &v5);
  result = v5;
  if ( v5 == 65 )
  {
    result = v6;
    if ( v6 == 67 )
    {
      result = v7;
      if ( v7 == 84 )
      {
        result = v8;
        if ( v8 == 70 )
        {
          result = v9;
          if ( v9 == 123 )
          {
            result = v14;
            if ( v14 == 125 )
            {
              v1 = v10;
              v2 = v11;
              v3 = v12;
              v4 = v13;
              for ( i = 0; i <= 15; ++i )
              {
                if ( *((_BYTE *)&v1 + i) > 64 && *((_BYTE *)&v1 + i) <= 90 )
                  *((_BYTE *)&v1 + i) = (*((_BYTE *)&v1 + i) - 51) % 26 + 65;
                if ( *((_BYTE *)&v1 + i) > 96 && *((_BYTE *)&v1 + i) <= 122 )
                  *((_BYTE *)&v1 + i) = (*((_BYTE *)&v1 + i) - 79) % 26 + 97;
              }
              for ( i = 0; i <= 15; ++i )
              {
                result = (unsigned __int8)*(&v15 + i);
                if ( *((_BYTE *)&v1 + i) != (_BYTE)result )
                  return result;
              }
              result = printf("You are correct!");
            }
          }
        }
      }
    }
  }
  return result;
}
```

重点就在

```
for ( i = 0; i <= 15; ++i ){
 if ( *((_BYTE *)&v1 + i) > '@' && *((_BYTE *)&v1 + i) <= 'Z' )// 大写字母
  *((_BYTE *)&v1 + i) = (*((_BYTE *)&v1 + i) - '3') % 26 + 'A';
 if ( *((_BYTE *)&v1 + i) > '`' && *((_BYTE *)&v1 + i) <= 'z' )// 小写字母
     *((_BYTE *)&v1 + i) = (*((_BYTE *)&v1 + i) - 79) % 26 + 'a';
}
for ( i = 0; i <= 15; ++i )
{
 result = (unsigned __int8)*(&v15 + i);
 if ( *((_BYTE *)&v1 + i) != (_BYTE)result )
  return result;
}
result = printf("You are correct!");
```

然后还是有点懵

爆破一下吧

```
v15= [ 'Q','s','w','3','s','j', '_','l','z','4','_','U','j','w','@','l' ]
flag=""

for i in range(16):#v15长度16
    for j in range(128):#ASCII字符个数
        x=j
        if chr(x).isupper():
            x=(x-51)%26+65
        if chr(x).islower():
            x=(x-79)%26+97
        if chr(x)==v15[i]:
            flag+=chr(j)

print (flag)
```

运行得到：`Cae3ar_th4_Gre@t`