

# 实验吧密码学WriteUp 三)

原创

再干杨超越  于 2018-11-13 17:55:12 发布  367  收藏

版权声明：本文为博主原创文章，遵循 [CC 4.0 BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) 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\\_43679627/article/details/84033070](https://blog.csdn.net/qq_43679627/article/details/84033070)

版权

分享一下我老师大神的人工智能教程！零基础，通俗易懂！<http://blog.csdn.net/jiangjunshow>

也欢迎大家转载本篇文章。分享知识，造福人民，实现我们中华民族伟大复兴！

## 1.兔子你好 (<http://www.shiyanbar.com/ctf/1815>)

兔子即为rabbit,按照提示rabbit解码即可

## 2.凯撒和某某加密 (<http://www.shiyanbar.com/ctf/1822>)

凯撒的话平时碰到的都是跟着字母表移位的，这里也没有多少字母，自然想到根据ascii来移位，可是移几位呢，要知道最后的格式中一定有 {, }, f,l,a,g等，于是就试着看看移动多少位能得到这些，或者也可以用python跑，脚本如下：

**-- coding:utf-8 --**

```
import string
```

```
def foo():
```

```
    s="aZZg/x\ZbavpZiEZp+n)o+"
```

```
    #ascii码表中可打印的字符
```

```
    a=string.maketrans(",")[33:127]
```

```
    for n in xrange(0,26):
```

```
        b=a[n:]+a[:n]
```

```
        table=string.maketrans(a,b)
```

```
        print string.translate(s,table)
```

```
    pass
```

```
if name == 'main':
```

```
    foo()
```

```
    print 'ok'
```

得到了其中最像的结果：f\_l4}a\_gf{u\_nJ\_u0s.t0

用脚指头想想都知道是栅栏加密

解密即可

## 3.一串奇怪的数 (<http://www.shiyanbar.com/ctf/1824>)

(引用pcat,侵删)

这题有提供加密程序，其实看懂后，最关键的是加密密码，如果不知道，就坑爹了，除非暴力破解匹配结果是否存在密钥的格式，鉴于这题密码为空，我就提供下面的python代码（如果觉得非要暴力破密码，自己加个循环读取密码字典之类，再判断结果是否符合密钥格式即可），程序理解不难，我就不赘述了。题目给出的密文解出来是4组密钥（附带4个\n），提交最后一组密钥即可。

**-- coding: utf8 --**

```

import hashlib

def md5(s):
    return hashlib.md5(s).hexdigest()

def evalCrossTotal(strMd5):
    r = 0
    for i in strMd5:
        r += int("0x%s" % i, 16)
    return r

def encryptString(strString, strPasswd):
    strPasswdMd5 = md5(strPasswd)
    intMd5 = evalCrossTotal(strPasswdMd5)
    r = []
    for i in range(len(strString)):
        r.append(ord(strString[i]) + int("0x%s" % strPasswdMd5[i%32], 16) - intMd5)
        intMd5 = evalCrossTotal(md5(strString[(i+1)]):16] + md5(str(intMd5)):16])
    return "".join(map(lambda x: str(x), r))

```

## 以上就是自己写的

```

def decryptString(nList,strPasswd):
    strPasswdMd5 = md5(strPasswd)
    intMd5 = evalCrossTotal(strPasswdMd5)
    r = ""
    for i in range(len(nList)):
        r+=chr(nList[i] - int("0x%s" % strPasswdMd5[i%32], 16) + intMd5)
        intMd5 = evalCrossTotal(md5(r[(i+1)]):16] + md5(str(intMd5)):16])
    return r

def foo():
    s="-149 -234 -157 -132 -187 -140 -157 -241 -158 -177 -85 -215 -180 -187 -173 -218 -161 -183 -133 -226 -136 -171 -126 -169 -155 -96 -169 -240 -163 -153 -137 -111 -123 -191 -151 -213 -151 -142 -152 -208 -118 -137 -136 -244 -157 -168 -187 -201 -170 -176 -192 -209 -205 -174 -163 -189 -126 -133 -148 -194 -145 -212 -170 -155 -148 -165 -167 -206 -171 -177 -88 -173 -125 -129 -129 -235 -121 -190 -161 -165"
    nList=eval('[%s]' %s.replace(' ',';'))
    strPasswd="" #密码为空
    print decryptString(nList,strPasswd)

if name == 'main':
    foo()
    print "ok"
    pass

```

### 4.Play (<http://www.shiyanbar.com/ctf/1825>)

play的加密方式，百度可得playfair，依照百科中playfair的定义，先构造5\*5密码表，

s n f m t  
h b g o u  
i c j p v  
y d k q w  
a e l r x

如何构造?

先填充给出的密钥，后面再填充其他没出现过的字符，最后丢弃频率低的（一般是z）

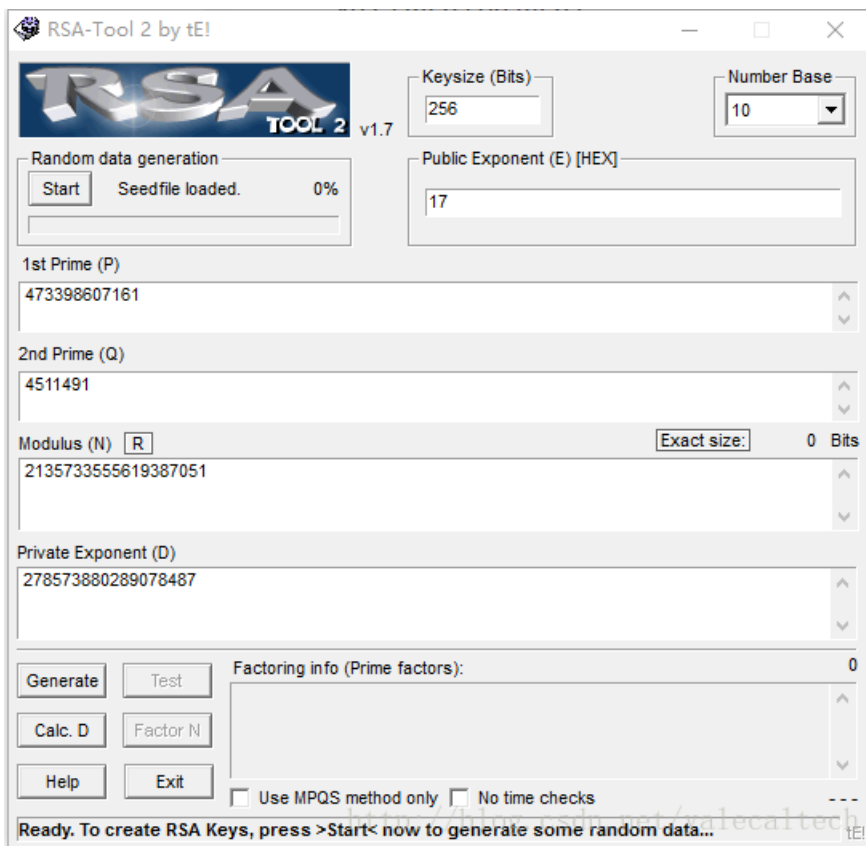
把密文KQSAMFPAOPMFPA两两分组

kq sa mf pa op mf pa

按照playfair的解密方法，得到结果

5.RSA实践 (<http://www.shiyanbar.com/ctf/1828>)

工具RSA-Tool 2 by tE!



6.Decode (<http://www.shiyanbar.com/ctf/1831>)

Decode，肯定是运用各种编码解码

0x标识表明这是十六进制编码，解码得到一串百分号打头的编码：

这是URL编码，解码，得到：

.xCjEwOQoxMDEKMTE2CjExMQoxMTUKMTA0CjEwNQoxMjEKOTcKMTEwCjk4Cjk3CjExNA==

串尾后面的==号很明显地暴露了这是Base64，再次解码一系列数字，这是ASCII码，解码得到flag  
加上flag格式后就可以提交了ctf{welc????????????????}

7.敌军情报 (<http://www.shiyanbar.com/ctf/1858>)

数字联想到ASCII码值对应成字符，得到“-.-.-.-.-”

解密摩斯电码即可

8.奇妙的音乐 (<http://www.shiyanbar.com/ctf/1862>)

海伦凯勒可以猜到图片下方是盲文，解码得到.zip的解压密码  
将音频拖入Audacity,可以看到波形是摩斯  
解密即可

9.困在栅栏里的凯撒 (<http://www.shiyanbar.com/ctf/1867>)

题目提示，那肯定涉及到栅栏密码和凯撒密码，

栅栏：首先分栏

12位只能是2\*6或6\*2，根据题意6，所以分6栏，结果为NEQ{etYdsf}

然后用凯撒解密即可

10.古典密码 (<http://www.shiyanbar.com/ctf/1870>)

将Ascii码转换为字母，得到OCU{CFTELXOUYDECTNGAHOHRNFIE}IOTA  
再尝试栅栏（准确说，是类似栅栏的思想）

$35=7*5$

OCU{CFT  
ELXOUYD  
ECTNGAH  
OHRNFIE  
NM}IOTA

根据flag的格式CTF..来调整

得到

CTF{COU  
LDYOUEX  
CHANGET  
HEINFOR  
MATION}

CTF{COULDYOUEXCHANGETHETHEINFORMATION}

给我老师的人工智能教程打call! <http://blog.csdn.net/jiangjunshow>

小编那么拼



赞一个再撤!

<http://blog.csdn.net/sunhuaqiang1>