whaleCTF-30days-密码学【第二期】-培根的愤怒-writeup

原创

sec小玖 于 2018-07-23 21:04:30 发布 717 收藏
分类专栏： CTF 密码学 文章标签： CTF 密码学 RSA 培根密码

题目：

> 我这么精辟的密码为什么不能入大流！？今天我将于RSA和AES并肩行走哈哈！
> 答案格式whaleCTF{xxxx}，xxx为解密内容

打开压缩包，发现有一个RSA.txt，一个bacon.docx，打开bacon.docx发现需要密码，考虑从RSA.txt下手：

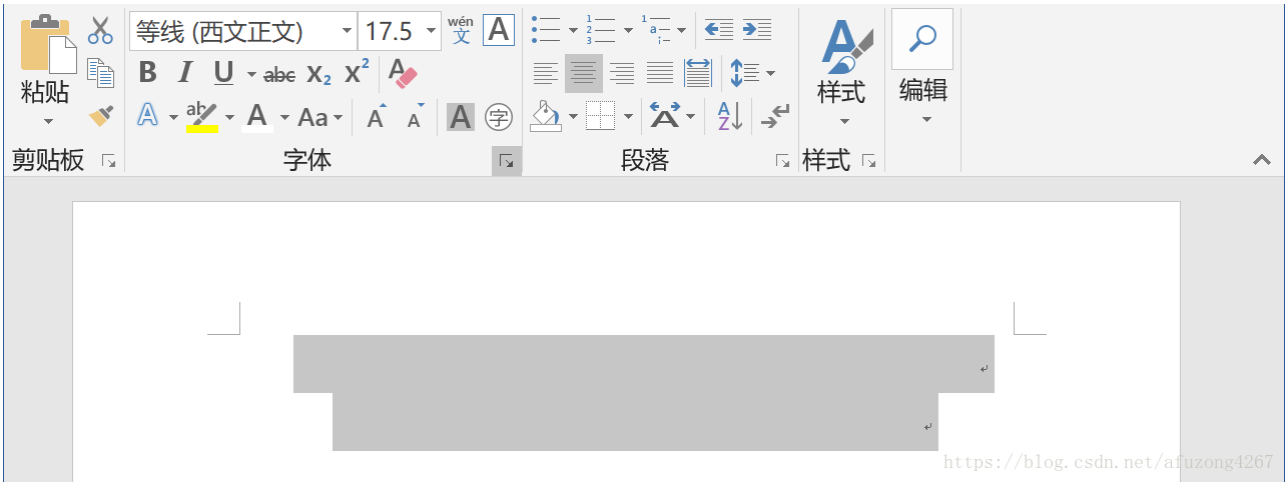
打开RSA发现熟悉的内容：

```
e=65537
d=176483621959488218784323414788013112177188929294617966775984021403530918544914390259395987050896651503958
n=201361544783408415641855912864937592840467111058171617141324248097548847644116545864032639334168068412516
c=22F4ECF53A876B17DA59DFA44235B8AB08F93D337D79427B13BC3036933FC850E0697926C625B6DCD5D21955D2BB43CB5348197FA
```

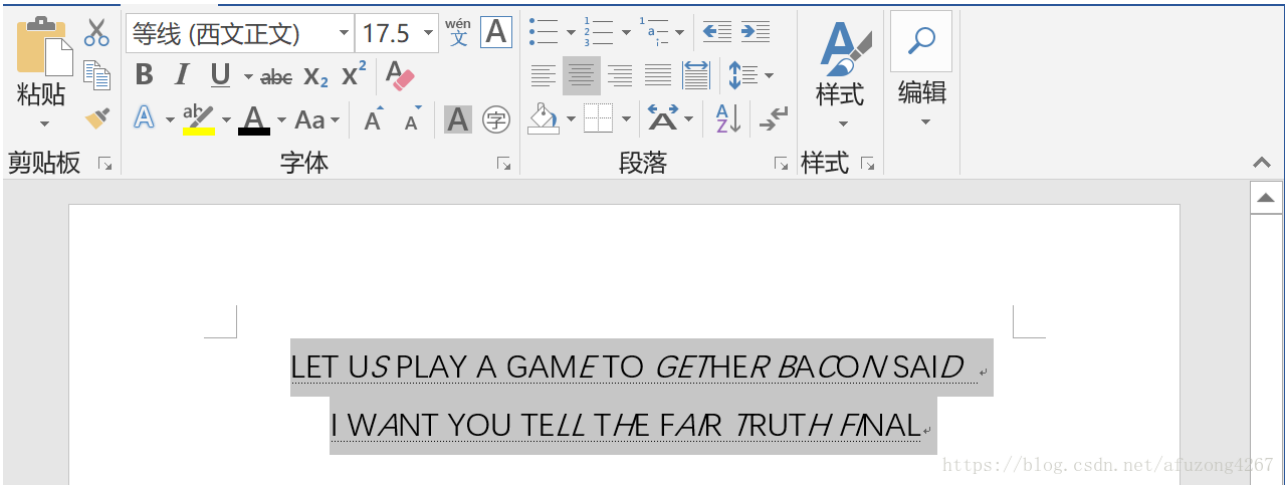
e代表公钥，d代表私钥，n代表模数，c代表密文。

使用工具直接进行解明文即可：

得到whaleCTFCR1pt0，提交flag发现不对，使用这个明文去解密bacon.docx文件，成功解密，打开文档，发现是空白，



ctrl+a 全选发现有内容，将字体更改颜色，发现内容：



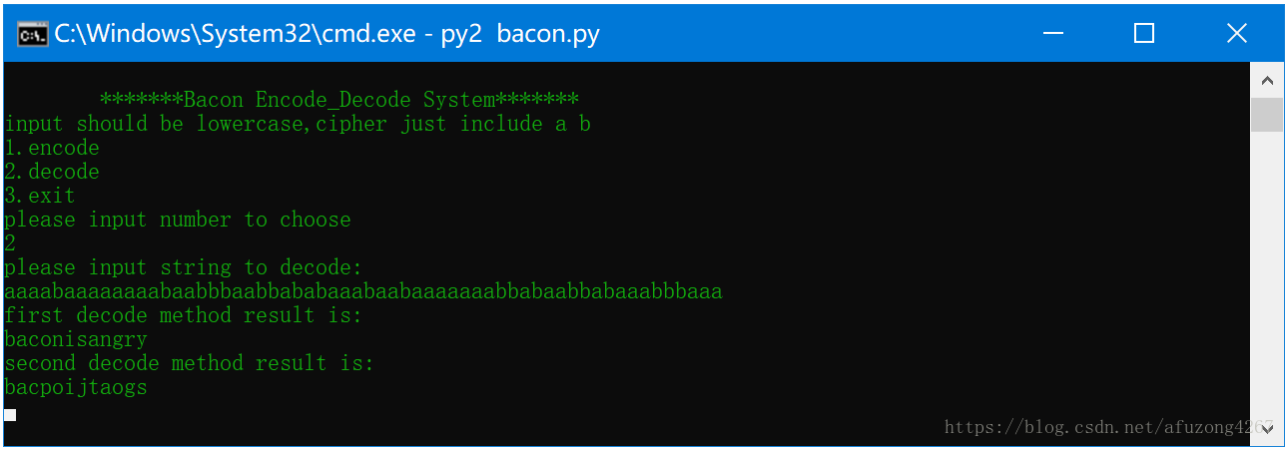
我们发现有斜体和正体两种字符，再根据文档的名称，想到培根密码，将正体用a替代，斜体用b替换，得到培根密码：

aaaab aaaaa aaaba abbba abbab abaaa baaba aaaaa abbab aabba baaab bbaaa

使用python脚本直接解密，这里提供一下培根密码解密脚本，支持两种培根密码表：

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import re

alphabet = ['a','b','c','d','e','f','g','h','i','j','k','l','m','n','o','p','q','r','s','t','u','v','w','x'

first_cipher = ["aaaaa","aaaab","aaaba","aaabb","aabaa","aabab","aabba","aabbb","abaaa","abaab","ababa","ab

second_cipher = ["aaaaa","aaaab","aaaba","aaabb","aabaa","aabab","aabba","aabbb","abaaa","abaaa","abaab","a

def encode():
    string = raw_input("please input string to encode:\n")
    e_string1 = ""
    e_string2 = ""
    for index in string:
        for i in range(0,26):
            if index == alphabet[i]:
                e_string1 += first_cipher[i]
                e_string2 += second_cipher[i]
                break
    print "first encode method result is:\n"+e_string1
    print "second encode method result is:\n"+e_string2
    return


def decode():
    e_string = raw_input("please input string to decode:\n")
    e_array = re.findall(".{5}",e_string)
    d_string1 = ""
    d_string2 = ""
    for index in e_array:
        for i in range(0,26):
            if index == first_cipher[i]:
                d_string1 += alphabet[i]
            if index == second_cipher[i]:
                d_string2 += alphabet[i]
    print "first decode method result is:\n"+d_string1
    print "second decode method result is:\n"+d_string2
    return


if __name__ == '__main__':
    while True:
        print "\t*******Bacon Encode_Decode System*******"
        print "input should be lowercase,cipher just include a b"
        print "1.encode\n2.decode\n3.exit"
        s_number = raw_input("please input number to choose\n")
        if s_number == "1":
            encode()
            raw_input()
        elif s_number == "2":
            decode()
            raw_input()
        elif s_number == "3":
            break
        else:
            continue
```

运行得到flag：


C:\Windows\System32\cmd.exe - py2 bacon.py
*******Bacon Encode_Decode System*******
input should be lowercase,cipher just include a b
1.encode
2.decode
3.exit
please input number to choose
2
please input string to decode:
aaaabaaaaaaaabaabbbaabbababaaabaabaaaaaaabbabaabbabaaabbbaaa
first decode method result is:
baconisangry
second decode method result is:
bacpoijtaogs
https://blog.csdn.net/afuzong42