




CTF-CRYPTO-2020新基建初赛-ezCrypto

原创

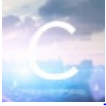
大熊何在  于 2021-01-12 09:07:05 发布  407  收藏

分类专栏: [CRYPTO](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 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/zippo1234/article/details/112478708>

版权



[CRYPTO 专栏收录该内容](#)

27 篇文章 1 订阅

订阅专栏

CTF-CRYPTO-2020新基建初赛-ezCrypto

ezCrypto

题目分析

开始

1.题目

2.逆序

3.词频分析

4.爆破偏移

5.base64换表

5.get flag

结语

试着坚持, 欢迎鞭策

ezCrypto

题目分析

1.大小写字母rot不同

2.base64换表

开始

1.题目

```
!=4IJkynJl1TaX8g7Kv1aK :mokzwof svh tc vqfo bo fsrbi ubwg fsjcz o rfosv T fsjwf ubwaawfp svh mp bkcr rbL .hosvk h  
gsjfov tc grzswt sfsk hbsasjod svh bcDi grkcfq svh ,hssfH zchgwFM bkcr ubwyoH .ubwsjs sbc hic rsyok T gL
```

2.逆序

```
!=4IJkynJl1TaX8g7Kv1aK
```

从这段明显可以看出是逆的。所以应该先逆序。得到

```
Lg T kozysr cih cbs sjsbwbu. Hozywbu rckb Mfwghcz Dhfssh, hvs qfckrg idcb hvs dojsasbh ksfs twszrg ct vofjsgh kv  
soh. Lbr rckb pm hvs pfwaawbu fwjsf T vsofr o zcjsf gwbu ibrsf ob ofqv ct hvs fowzkom: KalvK7g8XaTlJnykJI4=!
```

3.词频分析

使用Decrypto进行词频分析。当然quip也是可以的。

The screenshot shows the Decrypto 8.5 interface. At the top, the title bar reads "Decrypto 8.5 build 237, eolson@mit.edu". Below the title bar is a menu bar with "File", "Edit", "Dictionary", "Advanced", and "Help". The main window is titled "Cipher Text" and contains the following text:

```
Lg I kozysr cih cbs sjsbwbu. Hozywbu rckb Mfwghcz Dhfssh, hvs qfckrg idcb hvs dojsasbh ksfs twszrg ct vofjsgh kv  
fowzkom: KalvK7g8XaTlJnykJI4=!
```

Below the cipher text is a "Clues" field, which is highlighted with a red box and contains the text: "kozysr=walked, cih=out, cbs=one, sjsbwbu=evening, Kalv=Zxmh".

Below the clues is a "Solutions:" section, which is also highlighted with a red box. It contains a table with the following data:

Rank	Score	Solution
1	-2.9437	Ms F zalked out one evening. Talking dozn Yristol Pstreet, the crozds upon the pavexent sing under an arch of the railzay: ZxmhZ7s8QxFmVjkzVU4=!
2	-2.9956	Ms F zalked out one evening. Talking dozn Bristol Pstreet, the crozds upon the pavexent sing under an arch of the railzab: ZxmhZ7s8YxFmVjkzVU4=!
3	-3.0476	Mp F zalked out one evening. Talking dozn Yriptol Street, the crozdp uson the savexent ping under an arch of the railzay: ZxmhZ7p8QxFmVjkzVU4=!

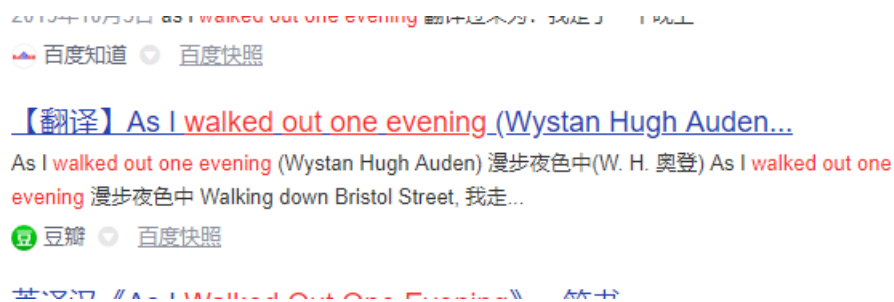
At the bottom of the interface, it says "Finished (139 solutions)". In the bottom right corner, there is a URL: <https://blog.csdn.net/zippo1234>.

这里是一步步的。比如flag经过base64后是Zxmh，然后逐步爆出几个已知的字母。进而发现小写、大写字母的偏移不同。

4.爆破偏移

爆破的关键就是找到正确的单词。。。这里就体现百度的作用了。

walked out one evening。百度之。。



果然后。。。所以正确的单词应该是As I walked out one evening

那么脚本就来了，抄袭某大神。

```
#!/python2
# -*- coding: utf-8 -*-
# @Time : 2021/1/11 16:27
# @Author : A.James
# @FileName: test.py
string = "!4IJKynJlTaX8g7Kv1aK :mokzwof svh tc vqfo bo fsrbi ubwg fsjcz o rfosv T fsjwf ubwaawfp svh mp bkcr rb
L .hosvk hgsjfov tc grzswt sfsk hbsasjod svh bcdi grkcfcq svh ,hssfhd zchgwfM bkcr ubwyoH .ubwbsjs sbc hic rsyzo
k T gL"
string = string[::-1]
print (string)
def change(c, a, b):
    dic1 = "abcdefghijklmnopqrstuvwxyz"
    dic2 = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
    if c in dic1:
        return dic1[(dic1.index(c) + a)%26]
    elif c in dic2:
        return dic2[(dic2.index(c) + b)%26]
    else:
        return c
for i in range(26):
    for j in range(26):
        m = ""
        for k in string:
            m += change(k, i, j)
        if m.find("As I walked out one evening")>=0:
            print (i, j, m)
```

得到

```
12 15 As I walked out one evening. Walking down Bristol Street, the crowds upon the pavement were fields of harv
est wheat. And down by the brimming river I heard a lover sing under an arch of the railway: ZmxhZ7s8MmIXzkwYX4
=!
```

5.base64换表

眼前一亮的情况下，直接把ZmxhZ7s8MmIxYzkwYX4=进行base64，发现数字和花括号不对。

```
输出(转换值):
flag?<2b1c90a~
```

就是base64换表无疑了。而且优先怀疑是数字和符号部分不对。爆破吧，前后也就十几次。

```
#!/python3
# -*- coding: utf-8 -*-
# @Time : 2021/1/11 16:43
# @Author : A.James
# @FileName: tt1.py
import base64

str1 = "ZmxhZ7s8MmIxYzkwYX4="
sss = ['123456789+/0', '23456789+/01', '3456789+/012', '456789+/0123', '56789+/01234', '6789+/012345', '789+/0123456',
'9+/012345678', '+/0123456789', '/0123456789+']
string = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz'
string2 = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/"
for i in range(len(sss)):
    string1 = string + sss[i]
    print (base64.b64decode(str1.translate(str.maketrans(string1,string2))))
```

得到结果

```
b'flag\xab;2b1c90a}'
b'flag\x9b:2b1c90a}'
b'flag\x8b92b1c90a}'
b'flag{82b1c90a}'
b'flagk72b1c90a\x7f'
b'flag[62b1c90a\x7f'
b'flagK52b1c90a\x7f'
b'flag\xeb?2b1c90a~'
b'flag\xdb>2b1c90a~'
b'flag\xcb=2b1c90a~'
```

5.get flag

```
flag{82b1c90a}
```

结语

什么鬼。。