

BugkuCTF Crypto write up

原创

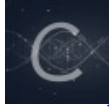
西西回 于 2018-07-18 14:39:31 发布 1106 收藏 2

分类专栏: [write up Crypto](#) 文章标签: [CTF Crypto write up](#)

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/qq_41725312/article/details/81088571

版权



[write up](#) 同时被 2 个专栏收录

4 篇文章 0 订阅

订阅专栏



[Crypto](#)

2 篇文章 0 订阅

订阅专栏

1.滴答~滴

滴答~滴

20

.....

答案格式KEY{xxxxxxxx}

Flag

Submit

https://blog.csdn.net/qq_41725312

(1)思路:很简单,一眼看出是摩斯密码,直接用工具解码得到flag

解密工具:CTFCrakTools

2.聪明的小羊

聪明的小羊

20

一只小羊翻过了2个栅栏

KYsd3js2E[a2jda]

Flag	Submit
------	--------

(1)思路:提示当中看到栅栏就想起栅栏密码,2个栅栏解密,得到flag

解密地址:<http://tool.bugku.com/jiemi/>

3.ok

ok
30

Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook! Ook? Ook! Ook! Ook. Ook? Ook. Ook.
Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook.
Ook. Ook? Ook. Ook? Ook! Ook. Ook? Ook. Ook. Ook. Ook. Ook!
Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook. Ook?
Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook! Ook? Ook! Ook! Ook. Ook? Ook!
Ook! Ook! Ook!
Ook! Ook! Ook? Ook. Ook? Ook! Ook. Ook? Ook! Ook! Ook! Ook!
Ook! Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook.
Ook? Ook.

(1)思路:直接用Ook解密,得到flag

解密地址:<https://www.splitbrain.org/services/ook>

Text to Ook!	Text to short Ook!	Ook! to Text
Text to Brainfuck	Brainfuck to Text	

4.这不是摩斯密码

这不是摩斯密码

30

下载看看吧

1.txt

Flag

Submit

https://blog.csdn.net/qq_41725312

(1)思路:下载下来查看,确实不是摩斯密码,是一种叫**Brainfuck**的语言,解密出来就是flag

```
+++++ +++++ [->++ +++++ ++<] >+.,+ +++++ .<+++ [->-- -<]>- -.+++ +++.<
+++++ [->++++ +<]>+ +++.< +++[- >---< ]>--- .---- .<+++ +++++[->--- ----<
]>--- ----< .<+++ +++++[->++++ +++++< ]>++++ ++.<+ ++++++ +[->- ----<
-<]>. <+++++ +++++[->++++ +++++< ]>+ .<+++ [->-- -<]>- ----. <+++++ +++[->
>--- ----< ]>--- ----. ++++++ +.,++ +.,+ +.,+ .<+++ [->-- -<]>- --.<+ ++++++
+[->+ ++++++ +<]>+ ++.++ +.,+++ ++++++ +.,--- -.+++ ++.<+ ++[-> +++<] >+++++
++.<
```

https://blog.csdn.net/qq_41725312

Brainfuck是一种极小化的计算机语言，按照“Turing complete（完整图灵机）”思想设计的语言，它的主要设计思路是：用最小的概念实现一种“简单”的语言，BrainF**k 语言只有八种符号，所有的操作都由这八种符号(> < + - . , [])的组合来完成。

解密地址:<https://www.splitbrain.org/services/ook>



Text to Ook! | Text to short Ook! | Ook! to Text

Text to Brainfuck | **Brainfuck to Text**

5.简单加密

简单加密

60

e6Z9i~]8R~U~QHE{RnY{QXg~QnQ{^XV1RX1p^XI5Q6Q6SKY8jUAA

Flag Submit

https://blog.csdn.net/qq_41725312

(1)思路:看到后面的字符串最后面的AA想到的是凯撒秘密和base64的混合加密.对照ASCII,的ASCII是65, =的ASCII是61, 偏移了四位(base64一般以'=='结束),再用base64解码得到flag

(2)脚本:python脚本将所有的字符都偏移四位

```
def main():
    string = 'e6Z9i~]8R~U~QHE{RnY{QXg~QnQ{^XV1RX1p^XI5Q6Q6SKY8jUAA'
    list = [chr(ord(i)-4) for i in string]
    print(''.join(list))

if __name__ == '__main__':
    main()
```

打印出来为:

a2V5ezY4NzQzMDAwNjUwMTczMjMwZTRhNThtZTE1M2M2OGU4fQ==

解密工具:CTFCrakTools

6.一段Base64

一段Base64

80

flag格式: flag{xxxxxxxxxxxxxx}



Flag Submit

https://blog.csdn.net/qq_41725312

(1)思路:打开是一段特别长的base64编码,base64解码,是8进制转义序列,

`\134\170\65\143\134\170\67\65\134\170\63\60\1`

再转是16进制转义序列,

`\x5c\x75\x30\x30\x35\x33\x5c\x75\x30\x30\x37`

又转,出来为Unicode编码

`\u0053\u0074\u0072\u0069\u0065\u0067\u002e\u0066\u0072\u006f\u006d\u0043`

解码出ASCII码,转义出的是html编码

```
&#10;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;
2;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;
&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;
&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;&#32;
&#10;&#10;&#51;&#56;&#44;&#51;&#53;&#44;&#49;&#50;&#48;&#44;&#53;&#48;&#44;&#53;&#52;&#44;&#53;&#57;&#44;&#51;&#56;&#44;&#51;&#53;&#44;&#49;&#50;&#48;&#44;&#53;&#48;&#44;&#53;&#9;&#44;&#53;&#57;&#44;&#51;&#56;&#44;&#51;&#53;&#44;&#49;&#50;&
```

解码

```
&#102;&#108;&#97;&#103;&#37;&#55;&#66;&#99;&#116;&#102;&#95;&#116;&#102;&#99;&#50;&#48;&#49;&#55;&#49;&#55;&#113;&#119;&#101;&#37;&#55;&#68;
```

然后Unicode解码

Unicode编码	UTF-8编码	URL 编码/解码	Unix时间戳	Ascii/Native 编码互转
<pre>&#102;&#108;&#97;&#103;&#37;&#55;&#66;&#99;&#116;&#102;&#95;&#116;&#102;&#99;&#50;&#48;&#49;&#55;&#49;&#55;&#113;&#119;&#101;&#37;&#55;&#68;</pre>				转换后的结果
<p>ASCII 转 Unicode Unicode 转 ASCII Unicode 转中文 中文 转 Unicode</p>				

为url编码,最后解出flag

(2)脚本:一直解到16进制的Python3脚本

```

import base64
import re

def main ():
    cipher_1 = '*' #为base64编码,太长省略
    plain_1 = base64.b64decode(cipher_1).decode('utf-8')
    # cipher = cipher.split('\')[1:]

    cipher_2 = re.findall(r'\d+',plain_1)
    plain_2 = ''
    for i in cipher_2:
        plain_2 += chr(int(i,8))

    cipher_3 = re.findall(r'\d[0-9]|\d[a-z]',plain_2)
    plain_3 = ''
    for i in cipher_3:
        plain_3 += chr(int(i,16))

    print(plain_3)
if __name__ == '__main__':
    main()

```

(3)总结:要多熟悉常见的编码,看到就能认出来.

还有一种简单的方法,用coverter的工具

<https://blog.csdn.net/pdsu161530247/article/details/74640746>

解密网站:<http://tool.chinaz.com/tools/unicode.aspx>

解密工具:coverter

7..!?

.!?
80

```

..... !?! ?... .. ??! ?... !...
..... 1.?. .. !?! ?!!!! !!?.? 1.?! !!! .. .. 1.?.
..... !? !!?. .. ??. 1.?. .. 1.?. .. .. !?! ?!!!!
!!!! !?.? ?!.? .. ! ?!.? .. ? ?!.? .. 1.?. ..
!?! ?!!!! !!?.? 1.?! !!!!! !!!!! .. .. ! ?... ..!?.? ..
?.?. ?.. ?... .. !?! ?!!!! !!!!! ?!.? !!!!! !!!!! !!.? ..
..!?. 1.?. .. ? ?!.? .. ! !!!!! !!!!! !!!!! !!.? .. !?!
?. .. ? ?!.? .. !!!!! !!!!! 1.?. .. !?. 1.?. .. ??.?
?.. .. 1.?.

```

另类的Ook编码

直接解出flag

解密地址: <https://www.splitbrain.org/services/ook>

8. +[]-

+[]-
80

```
+++++ +++++ [->+++ +++++ +++++] >+.,+ +++++ .<+++ [->-- -<]>- -.+++
+++.<
+++++ [->+++ +<]>+ +++++< +++++ [->-- ---<] >.<+++ ++[-> +++++< ]>++++
.<+++
[->-- -<]>- ----. +++++. <++++ [->+++ <]>+. <++++ [->-- -<]> ---- -.<+++
+[->+ +<]> ++. . ---- ---.< +++ [->+++< ]>++++ . ---- .<+++ [->-- -<]>-
.<+++ +++ [->-- -<]> ---- ----. +.<+++ +++++ +[->+ +++++ +<]>
+++++
+++++ .<
```

Flag Submit

(1)思路:很眼熟,就是上面的brainfuck编码,直接解出flag

9. 奇怪密码

奇怪的密码
100

突然天上一道雷电
gndk€rlqhmtkwwpjz

Flag Submit

(1)思路:格式有点像flag的格式,对照ASCII表发现规律,gndk与flag的相差依次增多,直接写个脚本打印出类似flag的值

flagPtslei_ci_jiami

发现不对,Pts不知道怎么回事,试着改成flag{lei_ci_jiami},成功

(2)脚本:

```
def main():
    string = 'gndk€rlqhmTKwP}z'
    count = 0
    result = ''
    for i in string:
        count += 1
        result = result + chr(ord(i)- count)
    print(result)
```

10. 托马斯·杰斐逊

托马斯·杰斐逊

100

```
1: <ZWAXJGDLUBVIQHKYPNTCRMOSFE <
2: <KPBELNACZDTRXMJQOYHGVSFUWI <
3: <BDMAIZVRNSJUWFHTEQGYXPLOCK <
4: <RPLNDVHGFCUKTEBSXQYIZMJWAO <
5: <IHFRLABEUOTSGJVDKCPMNZQWXY <
6: <AMKGHIWPNYCJBFZDRUSLOQXVET <
7: <GWTHSPYBXIZULVKMRAFDCEONJQ <
8: <NOZUTWDCVRJLXKISEFAPMYGHBQ <
9: <QWATDSRFHENYVUBMCOIKZGJXPL <
10: <WABMCXPLTDSRJQZGOIKFHENYVU <
11: <XPLTDAOIKFZGHENYSRUBMCQWVJ <
12: <TDSWAYXPLVUBOIKZGJRFHENMCQ <
13: <BMCSRFHLDENQWAOXPYVUIKZGJ <
14: <XPHKZGJTDSENYVUBMLAOIRFCQW <
```

密钥: 2,5,1,3,6,4,9,7,8,14,10,13,11,12

密文: HCBTSXWCRQGLS

flag格式 flag{你解密的内容} https://blog.csdn.net/qq_41725312

(1)思路:这是个杰斐逊密码盘, 根据第一个密钥跟密文, 把第二行单独取出来, 然后从密钥与密文一样的地方开始的部分, 放到内容最前面

例如:

<KPBELNACZDTRXMJQOYHGVSFUWI < 密钥对应为:H

<HGVSFUWIKPBELNACZDTRXMJQOY <

最后得到下面的密文


```
HGVSFUWIKPBELNACZDTRXMJQOY
CPMNZQWXYIHFRLABEUOTSGJVDK
BVIQHKYPNTCRMOSFEZWXJGDLU
TEQGYXPLOCKBDMAIZVRNSJUWFH
SLOQXVETAMKGHIWPNYCJBFZDRU
XQYIZMJWAORPLNDVHGFCUKTEBS
WATDSRFHENYVUBMCOIKZGJXPLQ
CEONJQGWTHSPYBXIZULVKMRAFD
RJLXKISEFAPMYGHBQNOZUTWDCV
QWXPBKZGJTSENYVUBMLAOIRFC
GOIKFHENYVUWABMCXPLTDSRJQZ
LTDENQWAOXPYVUIKZGJBMCSRFH
ENYSRUBMCQWVJXPLTDAOIKFZGH
SWAYXPLVUBOIKZGJRFHENMCQTD
```

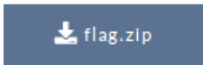
明文为是按列来读取的

然后一列列去尝试，倒数第六列是flag

最后提交的flag是小写

11.zip伪加密

zip伪加密 100



https://blog.csdn.net/qq_41725312

(1)思路:标题很明确,只要知道可以直接就解了

关于zip伪加密:https://blog.csdn.net/qq_41725312/article/details/81069184

12.告诉你个秘密

告诉你个秘密(ISCCCTF) 100

636A56355279427363446C4A49454A7154534230526D6843
56445A31614342354E326C4B4946467A5769426961453067

https://blog.csdn.net/qq_41725312

(1)思路:仔细观察发现字母都没超过F,猜测为16进制

解出来发现,应该为base54编码

cjV5RyBscDIJIEJqTSB0RmhCVDZ1aCB5N2IKIFFzWiBiaE0g

解出来几组字母与数字组合

r5yG lp9l BjM tFhBT6uh y7iJ QsZ bhM

是键盘密码,一组中对应的键盘位置回围绕一个键

比如 r5yg t就被围在他们中间

解出flag

解密网站:<http://www.ab126.com/goju/1711.html>

13.来自宇宙的信号

来自宇宙的信号

110

银河战队出击

flag格式 flag{字母小写}

 20171021180... https://blog.csdn.net/qq_41725312

(1)思路:搜索'银河战队密码',没搜到什么,再试'银河密码',发现了对应的图



直接对应解出flag

