

CTF--攻防世界crypto新手训练7-11

原创

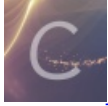
本人&ren 于 2020-05-06 16:34:42 发布 939 收藏 5

分类专栏: [CTF](#)

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_40730029/article/details/105949280

版权



[CTF 专栏收录该内容](#)

17 篇文章 1 订阅

订阅专栏

[7.不仅仅是Morse](#)

[8.easy_RSA](#)

[9.easychallenge](#)

[10.混合编码](#)

[11.转轮机加密](#)

7.不仅仅是Morse

题目: “这个题目和我们刚刚做的那个好像啊但是为什么按照刚刚的方法做出来答案却不对呢”, 你奇怪的问了问小鱼, “可能是因为还有一些奇怪的加密方式在里面吧, 我们在仔细观察观察”。两个人 安安静静的坐下来开始思考, 很耐心的把自己可以想到的加密方式一种种的过了一遍, 十多分钟后两个人 异口同声的说“我想到了! ”。一种食物,格式为cyberpeace{小写的你解出的答案}

不仅仅是Morse

👍 16 最佳Writeup由Viking • ZERO_Nu1L提供

WP 建议

难度系数: ★★★★★ 3.0

题目来源: poxlove3

题目描述: “这个题目和我们刚刚做的那个好像啊但是为什么按照刚刚的方法做出来答案却不对呢”, 你奇怪的问了问小鱼, “可能是因为还有一些奇怪的加密方式在里面吧, 我们在仔细观察观察”。两个人 安安静静的坐下来开始思考, 很耐心的把自己可以想到的加密方式一种种的过了一遍, 十多分钟后两个人 异口同声的说“我想到了! ”。一种食物,格式为cyberpeace{小写的你解出的答案}

题目场景: 暂无

题目附件: 附件1

https://blog.csdn.net/qq_40730029

下载附件打开后是一长串摩斯码, 通过在线网站解码, 查看英文说明, 可以知道后面那些AB是我们需要再次解码的内容

加密时，明文中的每个字母都会转换成一组五个英文字母。其转换依靠下表：

A/a	aaaaa	H/h	aabbb	O/o	abbba	V/v	babab
B/b	aaaab	I/i	abaaa	P/p	abbbb	W/w	babba
C/c	aaaba	J/j	abaab	Q/q	baaaa	X/x	babbb
D/d	aaabb	K/k	ababa	R/r	baaab	Y/y	bbaaa
E/e	aabaa	L/l	ababb	S/s	baaba	Z/z	bbaab
F/f	aabab	M/m	abbaa	T/t	baabb		
G/g	aabba	N/n	abbab	U/u	babaa		

加密者需使用两种不同字体，分别代表A和B。准备好一篇包含相同AB字数的假信息后，按照密文格式化假信息，即依密文中每个字母是A还是B分别套用两种字体。

https://blog.csdn.net/qq_40730029

Bugku|培根密码加解密

```
ATTACKANDEFENCEWORLDISINTERESTING
attackanddefenceworldisinteresting
```

解密 加密

https://blog.csdn.net/qq_40730029

flag如下

```
cyberpeace{attackanddefenceworldisinteresting}
```

8.easy_RSA




题目：解答出来了上一个题目的你现在可是春风得意，你们走向了下一个题目所处的地方 你一看这个题目傻眼了，这明明是一个数学题啊!!! 可是你的数学并不好。扭头看向小鱼，小鱼哈哈一笑，让你在学校里面不好好听讲现在傻眼了吧~来我来! 三下五除二，小鱼便把这个题目轻轻松松的搞定了。flag格式为cyberpeace{小写的你解出的答案} ![在这里插入图片描述]


(<https://img-blog.csdnimg.cn/20200419125455677.png?x-oss->

[process=image/watermark,type_ZmFuZ3poZW5naGVpdGk,shadow_10,text_aHR0cHM6Ly9ibG9nLmNzZG4ubmV0L3FxXzQwNzlwMDI5,size_16,color_FFFFFFFF,t_70\)#####](process=image/watermark,type_ZmFuZ3poZW5naGVpdGk,shadow_10,text_aHR0cHM6Ly9ibG9nLmNzZG4ubmV0L3FxXzQwNzlwMDI5,size_16,color_FFFFFFFF,t_70)#####) 下载附件打开 ![在这里插入图片描述](<https://img-blog.csdnimg.cn/20200419125526544.png>)

9.easychallenge

题目：你们走到了一个冷冷清清的谜题前面，小鱼看着题目给的信息束手无策，丈二和尚摸不着头脑，你嘿嘿一笑，拿出了你随身带着的笔记本电脑，噼里啪啦的敲起来了键盘，清晰的函数逻辑和流程出现在了电脑屏幕上，你敲敲键盘，更改了几处地方，运行以后答案变出现在了电脑屏幕上。

easychallenge  8 最佳Writeup由Viking • ZERO_Nu1L提供  

难度系数：  3.0

题目来源： NJUPT_CTF

题目描述：你们走到了一个冷冷清清的谜题前面，小鱼看着题目给的信息束手无策，丈二和尚摸不着头脑，你嘿嘿一笑，拿出了你随身带着的笔记本电脑，噼里啪啦的敲起来了键盘，清晰的函数逻辑和流程出现在了电脑屏幕上，你敲敲键盘，更改了几处地方，运行以后答案变出现在了电脑屏幕上。

题目场景： 暂无

题目附件： 附件1 https://blog.csdn.net/qq_40730029

下载附件打开后看到是一个.pyc文件，百度后发现可以将pyc文件反编译回py文件，在kali里安装反编译软件(使用pop3命令可以看如何配置rsactftool)

配置rsactftool工具链接：https://blog.csdn.net/qq_40730029/article/details/105937669

软件安装命令如下：

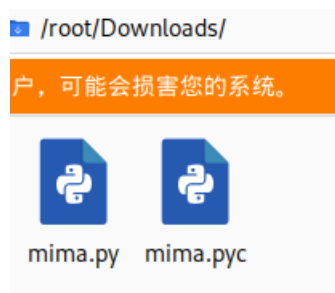
```
pip3 install uncompyl6
```

进入附件所在目录，操作如下命令（下载的附件名字过长，我将它重命名了）：

```
uncompyl6 -o . mima.pyc
```

```
root@kali:~/Downloads# uncompyl6 -o . mima.pyc
mima.pyc --
# Successfully decompiled file
```

路径下出现了.py文件



查看mima.py文件（使用cat命令）

```

import base64

def encode1(ans):
    s = ''
    for i in ans:
        x = ord(i) ^ 36
        x = x + 25
        s += chr(x)

    return s

def encode2(ans):
    s = ''
    for i in ans:
        x = ord(i) + 36
        x = x ^ 36
        s += chr(x)

    return s

def encode3(ans):
    return base64.b32encode(ans)

flag = ''
print 'Please Input your flag:'
flag = raw_input()
final = 'UC7KOWVXWVNKNIC2XCXKHKK2W5NLBKNOUOSK3LNNVWV3E'

```

运行文件，命令如下：

```
python mima.py //python3运行会报错，存在不兼容
```

运行之后输入都不正确，去看代码，发现flag要通过final的值得到，我们需要将代码反着走一次

```

root@kali:~/Downloads# python mima.py
Please Input your flag:
a
wrong
root@kali:~/Downloads# python mima.py
Please Input your flag:
UC7KOWVXWVNKNIC2XCXKHKK2W5NLBKNOUOSK3LNNVWV3E
wrong

```

解密脚本

```

import base64
final = 'UC7K0wVxwVNKNIC2XCXKHKk2W5NLBKN0UOSK3LNNVwW3E=== '

def decode1(ans):
    s = ''
    for i in ans:
        x = ord(i) - 25
        x = x ^ 36
        s += chr(x)
    return s

def decode2(ans):
    s = ''
    for i in ans:
        x = ord(i) ^ 36
        x = x - 36
        s += chr(x)
    return s

def decode3(ans):
    return base64.b32decode(ans)

print decode1(decode2(decode3(final )))

```

运行解密脚本，得到flag

```

root@kali:~/Downloads# python flag.py
cyberpeace{interestinghhhhh}

```

flag如下:

```
cyberpeace{interestinghhhhh}
```

10.Normal_RSA

题目：你和小鱼走啊走走啊走，走到下一个题目一看你又一愣，怎么还是一个数学题啊 小鱼又一笑，hhh数学在密码学里面很重要的！现在知道吃亏了吧！你哼一声不服气，我知道数学 很重要了！但是工具也很重要，你看我拿工具把他解出来！你打开电脑折腾了一会还真的把答案 做了出来，小鱼有些吃惊，向你投过来一个赞叹的目光

Normal_RSA
👍 22 最佳Writeup由露思提供
WP
建议

难度系数: ★★★★★ 5.0

题目来源: PCTF

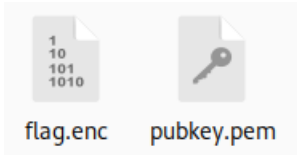
题目描述: 你和小鱼走啊走走啊走，走到下一个题目一看你又一愣，怎么还是一个数学题啊 小鱼又一笑，hhh数学在密码学里面很重要的！现在知道吃亏了吧！你哼一声不服气，我知道数学 很重要了！但是工具也很重要，你看我拿工具把他解出来！你打开电脑折腾了一会还真的把答案 做了出来，小鱼有些吃惊，向你投过来一个赞叹的目光

题目场景: 暂无

题目附件: 附件1

https://blog.csdn.net/qq_40730029

下载附件打开



题目是rsa, 使用openssl解密.pem中参数,得到e和modulus的值

命令如下(要进入到路径中使用openssl):

```
openssl rsa -text -pubin -modulus -in pubkey.pem
```

```
root@kali:~/Desktop/a# openssl rsa -text -pubin -modulus -in pubkey.pem
RSA Public-Key: (256 bit)
Modulus:
e 00:c2:63:6a:e5:c3:d8:e4:3f:fb:97:ab:09:02:8f:
  1a:ac:6c:0b:f6:cd:3d:70:eb:ca:28:1b:ff:e9:7f:
  be:30:dd
Exponent: 65537 (0x10001)
Modulus=C2636AE5C3D8E43FFB97AB09028F1AAC6C0BF6CD3D70EBCA281BFFE97FBE30DD
writing RSA key
-----BEGIN PUBLIC KEY-----
MDwwDQYJKoZIhvcNAQEBBQADKwAwKAIhAMJjauXD20Q/+5erCQKPGqsC/bNPXDr
yigb/+l/vjDdAgMBAAE=
-----END PUBLIC KEY-----
root@kali:~/Desktop/a#
```

通过python将模数转换成十进制

```
root@kali:/# python
Python 2.7.18 (default, Apr 20 2020, 20:30:41)
[GCC 9.3.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> a=0xC2636AE5C3D8E43FFB97AB09028F1AAC6C0BF6CD3D70EBCA281BFFE97FBE30DD
>>> print a
87924348264132406875276140514499937145050893665602592992418171647042491658461
>>>
```

结果:

```
87924348264132406875276140514499937145050893665602592992418171647042491658461
```

在线分解大素数得到p和q的值

网站: <http://www.factordb.com/>

搜索	顺序	报告结果	因子表	状态	资料下载	登
87924348264132406875276140514499937145050893665602592992418171647042491658461 <input type="button" value="分解!"/> (?)						
结果: p q						
状态 (?)	数字	数				
FF	77 (显示)	8792434826 ... 61	= 275127860351348928173285174381581152299	<39>	.319576316814478949870590164193048041239	<39>

https://blog.csdn.net/qq_40730029

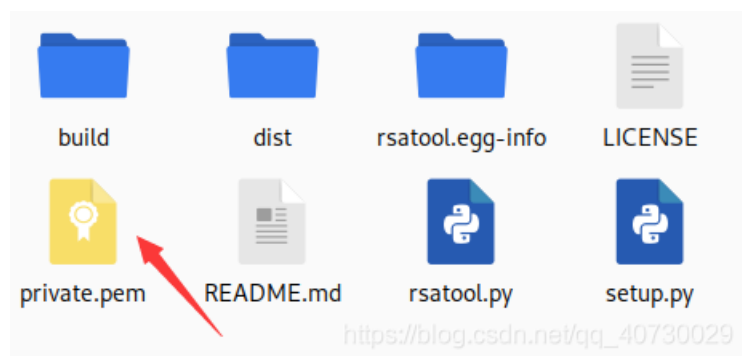
结果:

```
275127860351348928173285174381581152299
319576316814478949870590164193048041239
```

生成私钥文件

```
python rsatool.py -out private.pem -e 65537 -p 275127860351348928173285174381581152299 -q 319576316814478949870590164193048041239
```

```
root@kali:~/rsatool# python3 rsatool.py -o private.pem -e 65537 -p 275127860351
348928173285174381581152299 -q 319576316814478949870590164193048041239
Using (p, q) to initialise RSA instance
n =
c2636ae5c3d8e43ffb97ab09028f1aac6c0bf6cd3d70ebca281bffe97f3e30dd
e = 65537 (0x10001)
d =
1806799bd44ce649122b78b43060c786f8b77fb1593e0842da063ba0d8728bf1
p = 275127860351348928173285174381581152299 (0xcefbb2cf7e18a98ebcd36e3e7c3b02b
)
q = 319576316814478949870590164193048041239 (0xf06c28e91c8922b9c236e23560c09717
)
Saving PEM as private.pem
rsatool.py:98: DeprecationWarning: encodestring() is a deprecated alias since 3
.1, use encodebytes()
return (PEM_TEMPLATE % base64.encodestring(self.to_der()).decode()).encode()
root@kali:~/rsatool#
```



用 `private.pem` 解密 `flag.enc`（在 `flag.enc` 目录中，`private.pem` 也要放到这里）

```
openssl rsautl -decrypt -in flag.enc -inkey private.pem
```

```
root@kali:~/Desktop/a# openssl rsautl -decrypt -in flag.enc -inkey private.pem
PCTF{256b_i5_m3dium}
root@kali:~/Desktop/a#
```

flag如下:

```
PCTF{256b_i5_m3dium}
```

11. 转轮机加密

题目：你俩继续往前走，来到了前面的下一个关卡，这个铺面墙上写了好多奇奇怪怪的英文字母，排列的的整整齐齐，店面前面还有一个大大的类似于土耳其旋转烤肉的架子，上面一圈圈的也刻着很多英文字母，你是一个小历史迷，对于二战时候的历史刚好特别熟悉，一拍大腿：“嗨呀！我知道是什么东西了！”。提示：托马斯·杰斐逊。flag，是字符串，小写。

转轮机加密

👍 28 最佳Writeup由Viking • ZER0_Nu1L提供

WP

建议

难度系数: ★★★★★ 6.0

题目来源: ISCC2017

题目描述: 你俩继续往前走, 来到了前面的下一个关卡, 这个铺面墙上写了好多奇奇怪怪的 英文字母, 排列的的整整齐齐, 店面前面还有一个大大的类似于土耳其旋转烤肉的架子, 上面一圈圈的 也刻着很多英文字母, 你是一个小历史迷, 对于二战时候的历史刚好特别熟悉, 一拍大腿: “嗨呀! 我知道 是什么东西了! ”。提示: 托马斯·杰斐逊。flag, 是字符串, 小写。

题目场景: 暂无

题目附件: 附件1

https://blog.csdn.net/qq_40730029

下载附件打开, 然后百度转轮机密码的含义和相关信息

a3b693cdec9e4d479285c519ce9c521d.txt - 记事本

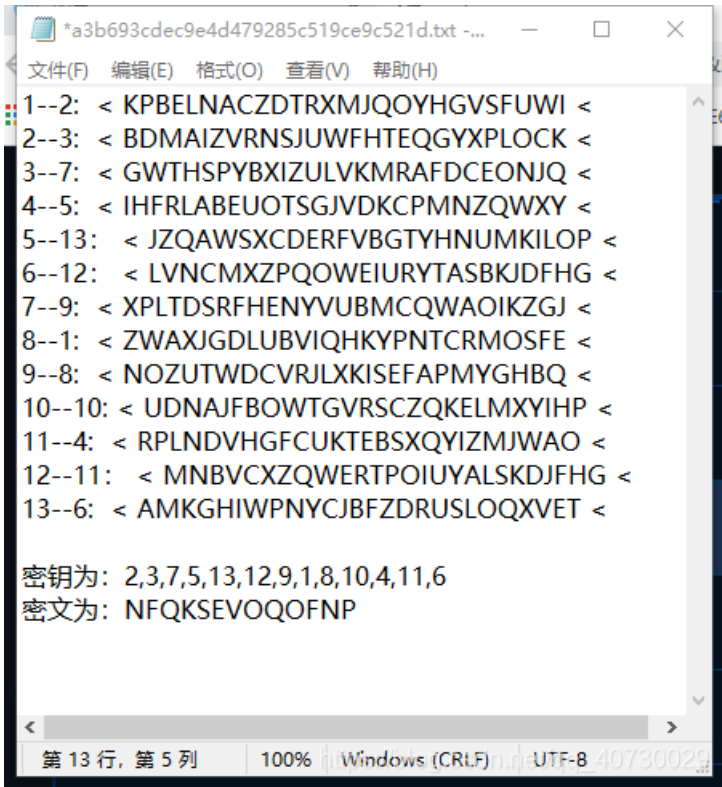
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

```
1: < ZWAXJGDLUBVIQHKYPNTCRMOSFE <
2: < KPBELNACZDTRXMJQOYHGVSFUWI <
3: < BDMAIZVRNSJUWFHTEQGYXPLOCK <
4: < RPLNDVHGFCUKTEBSXQYIZMJWAO <
5: < IHFRLABEUOTSGJVDKCPMNZQWXY <
6: < AMKGHIWPNYCBFZDRUSLOQXVET <
7: < GWTHSPYBXIZULVKMRAFDCEONJQ <
8: < NOZUTWDCVRJLXKISEFAPMYGHBQ <
9: < XPLTDSRFHENYVUBMCQWAOIKZGJ <
10: < UDNAJFBOWTGVRSCZQKELMXYIHP <
11: < MNBVCXZQWERTPOIUAYLSKDJFHG <
12: < LVNCMXZPQOWEIURYTASBKJDFHG <
13: < JZQAWSXCDEFVBGTYHNUMKILOP <
```

密钥为: 2,3,7,5,13,12,9,1,8,10,4,11,6

密文为: NFQKSEVOQOFNPdn.net/qq_40730029

根据密钥将第一条密码替换为第二条, 第二条替换为第三条, 以此类推, 第十三条替换为第六条



轮转机密码密文对应的是每行字符串的第一列，所以我们将每一行的第一个字母对应到密文，此时是将字符向左转，将所有列的字符串写出，根据提示信息是指第二次大战期间，找到第18列字符串与题目信息相关，于是判定为flag

密文为: NFQKSEVOQOFNP

NACZDTRXMJQOYHGVSFUWIKPBEL
 FHTEQGYXPLOCKBDMAIZVRNSJUW
 QGWTHSPYBXIZULVKMRAFDCEONJ
 KCPMNZQWXYIHFRLABEUOTSGJVD
 SXCDERFVBGTYHNUMKILOPJZQAW
 EIURYTASBKJDFHGLVNCMXZPQOW
 VUBMCQWAOIKZGJXPLTDSRFHENY
 OSFEZWAXJGDLUBVIQHKYPNTRM
 QNOZUTWDCVRJLXKISEFAPMYGHB
 OWTGVRSCZQKELMXYIHPUDNAJFB
 FCUKTEBSXQYIZMJWAOPLNDVHG
 NBVCXZQWERTPOIUAYALSKDJFHGM
 PNYCBFZDRUSLOQXVETAMKGHIW

第一列: **NFQKSEVOQOFNP** 第二列: AHGCXIUSNWCBN
 第三列: CTWPCUBFOTUVY 第四列: ZETMDRMEZGKCC
 第五列: DQHNEYZUVTXJ 第六列: TGSZRTQWTREZB
 第七列: RYPQFAWAWSBQF 第八列: XXYWVSAXDCSWZ
 第九列: MPBXBBOJCZXED 第十列: JLXYGKIGVQRR
 第十一列: QOITJKDRKYTU 第十二列: OCZHYDZLJEIPS
 第十三列: YKUFHFGULLZOL 第十四列: HBLRNHJBXMMIO
 第十五列: GDVLUGXVKXJUQ 第十六列: VMKAMPLPIIYWYX
 第十七列: SAMBKVLQSIAAV 第十八列: **FIREINTHEHOLE**
 第十九列: UZAULCDKFRST 第二十列: WVFOOMSYAUPKA
 第二十一列: IRDTPXRPPDLDM 第二十二列: KNCSJZFNMMNJK
 第二十三列: PSEGZPHTYADFG 第二十四列: BJOJQCEGJVHH
 第二十五列: EUNVAONRHFHGI 第二十六列: LWJDWWYMBBGMW

```
fireinthehole //该字符串直接提交就对了，前面不用加flag{}或cyberpeace{}
```