

CRYPTO 第一题

RSA Attack[已完成]

描述

这就是传说中的暴力美学么

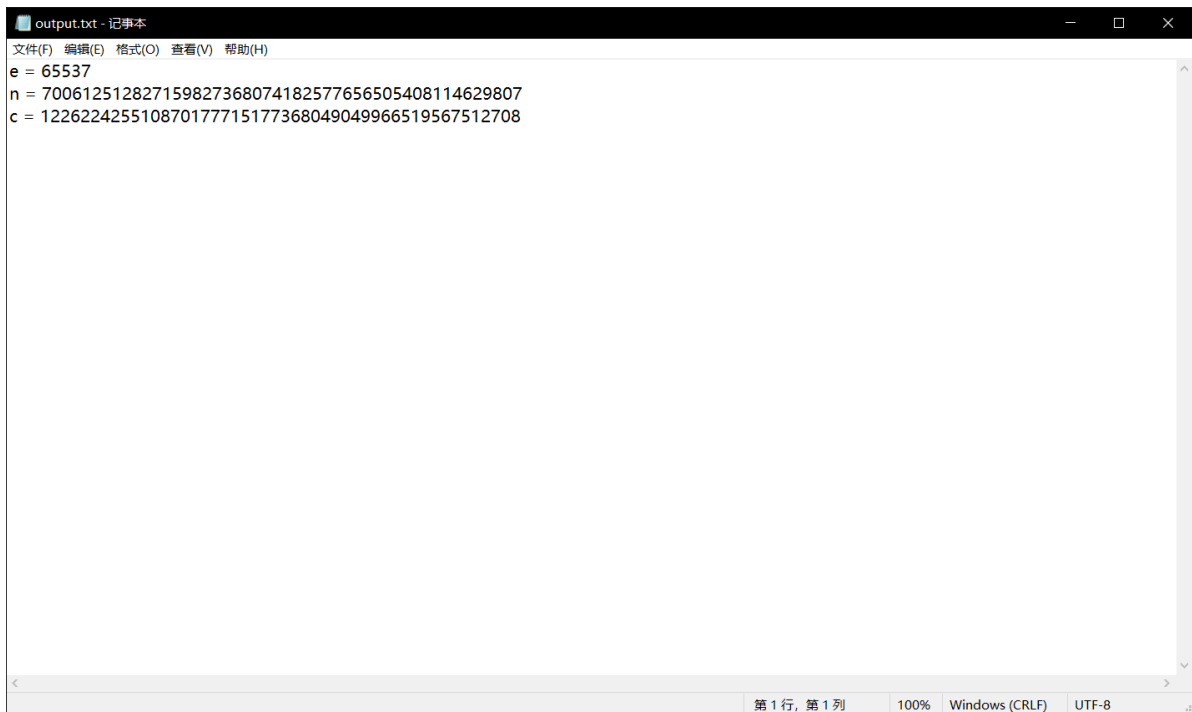
题目地址 <https://cmfj-1308188104.cos.ap-shanghai.myqcloud.com/Week2/RSA%20Attack.zip>

基准分数 150

当前分数 150

完成人数 128

打开链接,是一个压缩包,解压后有一个outout.txt和一个py代码



The screenshot shows a Notepad window titled "output.txt - 记事本". The menu bar includes "文件(F)", "编辑(E)", "格式(O)", "查看(V)", and "帮助(H)". The text content is as follows:

```
e = 65537
n = 700612512827159827368074182577656505408114629807
c = 122622425510870177715177368049049966519567512708
```

The status bar at the bottom indicates "第 1 行, 第 1 列", "100%", "Windows (CRLF)", and "UTF-8".

```

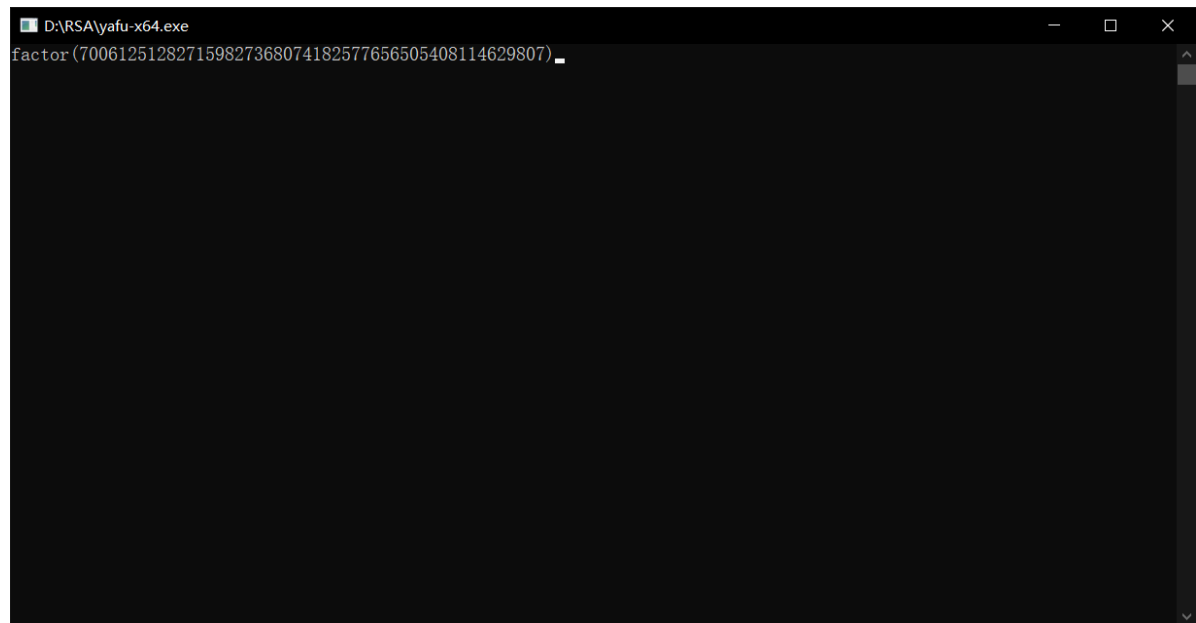
from Crypto.Util.number import getPrime
from libnum import s2n

from secret import flag

m = s2n(flag)
e = 65537
p = getPrime(80)
q = getPrime(80)
n = p * q
c = pow(m, e, n)
print("e =", e)
print("n =", n)
print("c =", c)

```

从代码中得知e,n,c三个数,求m,m即是flag,因为pq未知,先用yafu工具将n分解成两个因数



```

D:\RSA\yafu-x64.exe
factor(700612512827159827368074182577656505408114629807)

```

分解后结果如下

```

01/28/22 14:39:10 v1.34.5 @ DESKTOP-UNDL466, prp24 = 715800347513314032483037
01/28/22 14:39:10 v1.34.5 @ DESKTOP-UNDL466, prp24 = 978782023871716954857211

```

现在知道了p,q,e,c,然后再用脚本求出m即可

```

import libnum
from Crypto.Util.number import long_to_bytes
e = 65537
q = 715800347513314032483037
p = 978782023871716954857211
c = 122622425510870177715177368049049966519567512708
n = p*q
# n = int("",16)

# e = int("",16)

d = libnum.invmod(e, (p - 1) * (q - 1))
m = pow(c, d, n) # m 的十进制形式
string = long_to_bytes(m) # m明文
print(string) # 结果为 b' m ' 的形式

```

结果如下

```
b'hgame{SHorTeSt!fLAg}'
```

CRYPTO 第二题

RSA Attack 2[已完成]

描述

RSA 加密算法可安全啦，一般人可破解不了呢

可惜，再安全的算法也架不住使用的人“蠢”啊

小伙子我看你骨骼惊奇，下面这些人错误的使用了 RSA

快用你的密码学知识给他一个大嘴巴子，让他们长点教训

题目地址 <https://cmfj-1308188104.cos.ap-shanghai.myqcloud.com/Week2/RSA%20Attack%202.zip>

基准分数 250

当前分数 250

完成人数 188

解压得2个文件

```
task.py ×
1 import re
2 from math import ceil
3 from Crypto.Util.number import getPrime
4 from libnum import s2n
5 from secret import flag
6
7 flag_parts = list(map(s2n, re.findall(rf"{{{ceil(len(flag) / 3)}}}", flag)))
8
9 print("# task1")
10 m = flag_parts[0]
11 e = 65537
12 p = getPrime(1024)
13 q = getPrime(1024)
14 r = getPrime(1024)
15 n1 = p * q
16 c1 = pow(m, e, n1)
17 n2 = r * q
18 c2 = pow(m, e, n2)
19 print("e =", e)
20 print("n1 =", n1)
21 print("c1 =", c1)
22 print("n2 =", n2)
23 print("c2 =", c2)
24
25 print("# task2")
26 m = flag_parts[1]
27 e = 7
28 p = getPrime(1024)
29 q = getPrime(1024)
30 n = p * q
```

```
output.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
# task1
e = 65537
n1 =
146115456051079508275810051653276947828231886031517681697314314
183613062311149850377759174614339253080543969708096908040739858
353764646298606097102921813686006186265904984918504045034434142
414554873044483448923378774224657157091542386535051416059041849
853118737634957613457221552894578896860197466632937201068742273
236992882777942922089571724465234205963911148915595378110294731
501236416241081036765167544494928051266425527512783096348467776
360421141359905162459075173773201900914007292773076367248905921
552564379965661609954567430182250138519375938860861291313515829
58811003596445806061492952513851932238563627194553
c1 =
965075803554932988664271816439183802328812013694203741320763105
376036912584995031647672348468111310423680858101990670067065306
237596121664884353679987689532305437801346923070145524106271337
770666947677115752724993307387122132705797012726237073550669419
110046308257408484535063515678066777681017211510981429273346928
022971149411064556225001287399141306136081722471075032423079692
908380267160214143720516748000734987068685104675254411687005690
312116824966036851568223828884335112144637268090397158532937141
122654075952730052331573980701136378212002956719295192733955673
315234274064519957670199895100508623561838510479
n2 =
209374787251099838030791854504496165674645969613487274538172490
351100475855801428235512895771459581271215867928785093860851784
521711124558904294744577972192028270308842622730613347524934967
```

可以看出flag是由这三个字符串拼成的,分别解码即可

Web 第一题

webpack-engine[已完成]

描述

webpack packs the web.

(请使用 Chrome 浏览器打开)

题目地址 <https://ngin.hgame.potat0.cc>

基准分数 100

当前分数 100

完成人数 116

打开链接后显示

Webpack Engine

点击这个骚气的按钮

先试试看点击按钮(不过肯定是坑

点击启动炼金引擎

Webpack Engine

点击加载噩梦燃料

Webpack Engine

点击启动月球车

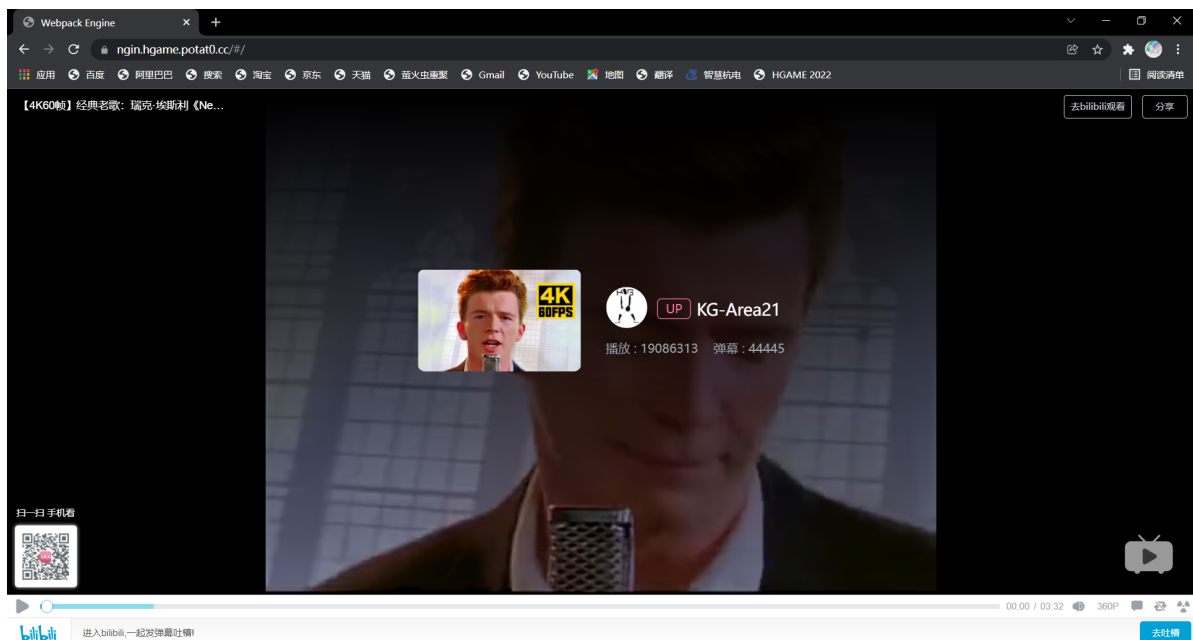
Webpack Engine

Webpack Engine

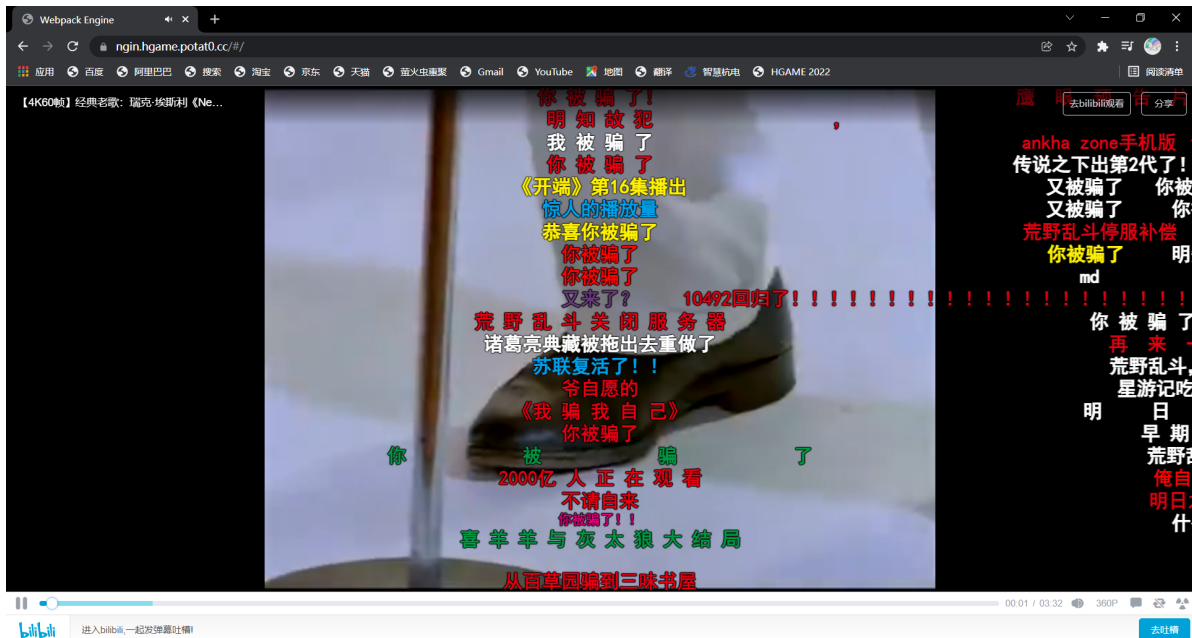
点击苏联解体

Webpack Engine

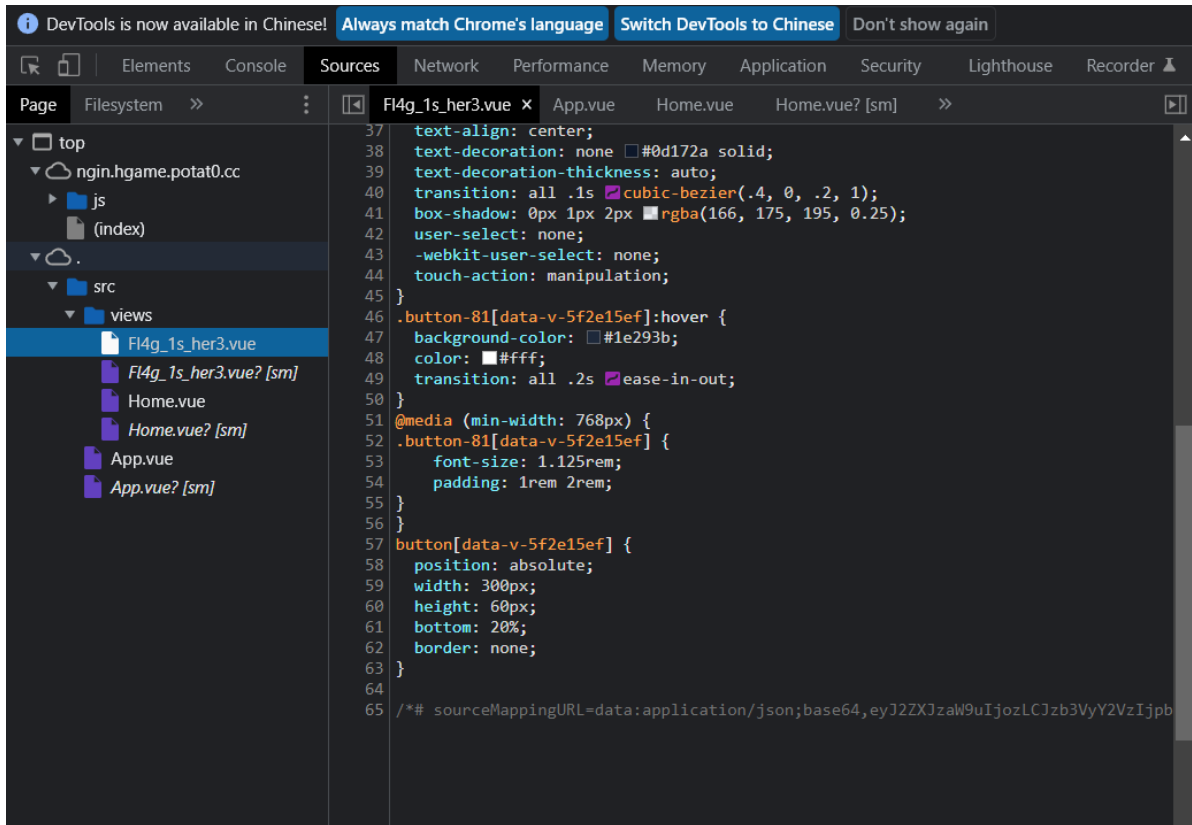
点击获取 flag



果然



弹幕很应景,返回链接,先打开F12看看,找到了有关flag的代码



右下方有提示base64,想到base64解码,然后把后面一长串字符串拿去解码,发现关键信息

gYm94LXNoYWRvdzogMHB4IDFweCAycHggcmdiYSgxNjYsIDE3NSwgMTk1LCAwLjI1KTtcbiAgdXNlcilzZWxlY3Q6IG5vbmU7XG4gIC13ZWJraXQtdXN1c
cilzZWxlY3Q6IG5vbmU7XG4gIHRvdWNoLWFjdGlvbjogbWluaXB1bGF0aW9uO1xufVxuXG4uYnV0dG9uLTgxOmhvdmVYIHtcbiAgYmFja2dyb3VuZC1jb
2xcvcjogIzFlMjkzYjtcbiAgY29sb3I6ICNmZmY7XG4gIHRyYW5zaXRpb246IGFsbCAuMnMgZWZzZS1pbilvdXQ7XG59XG5cbkBTZWRpYSAobWluLXdpZH
RoOiA3NjhweCkgelxuICAuYnV0dG9uLTgxIHtcbiAgICBmb250LXNpemU6IDEuMTI1cmVtO1xuICAuIHhZGRpbmc6IDFyZW0gMnJlbTtcbiAgfVxufVx
uXG5idXR0b24ge1xuICBwb3NpdGlvbjogYWJzb2x1dGU7XG4gIHdpZHRoOiAzMDBweDtcbiAgAGVpZ2h0OiA2MHB4O1xuICBib3R0b206IDIwJTtcbiAg
Ym9yZGVyOiBub251O1xufVxuPC9zdHlsZT5cbiJdLCJzb3VyY2VSb290IjoIn0=

清空

加密

解密

☐ 解密为UTF-8字节流

ACA, mCAAA;EACA, +BAAA;EACA, 8CAAA;EACA, iDAAA;EACA, iBAAA;EACA, yBAAA;EACA, OBAAA;AACA;AAEA;EACA, yBAAA;EACA, WAAA;EACA, +BAAA
:AACA;AAEA;AACA; IACA, mBAAA; IACA, kBAAA;AACA;AACA;AAEA;EACA, kBAAA;EACA, YAAA;EACA, YAAA;EACA, WAAA;EACA, YAAA;AACA", "source
sContent":["<template>\n <h1>{{filiililil4g}}</h1>\n</template>\n\n<script>\n\nexport default {\n data() {\n
return {\n filiililil4g: 'YUdkaGJXVjdSREJ1ZEY5bU1ISTVaWFJmTWw5RGJFOXpNMTlUTUhWeVkyVmZiVUJ3ZlE9PQo=' \n }\n }\n}\n</script>\n\n<style>\nhtml, body {\n height: 100%;\n margin: 0;\n padding: 0;\n overflow:
hidden;\n}\n</style>\n\n<style scoped>\n.home {\n height: 100%;\n position: relative;\n display: flex;\n flex-
direction: column;\n justify-content: center;\n align-items: center;\n}\n\n.player {\n width: 100%;\n height:

复制

再把YUdkaGJXVjdSREJ1ZEY5bU1ISTVaWFJmTWw5RGJFOXpNMTlUTUhWeVkyVmZiVUJ3ZlE9PQo=拿去解码

YUdkaGJXVjdSREJ1ZEY5bU1ISTVaWFJmTWw5RGJFOXpNMTlUTUhWeVkyVmZiVUJ3ZlE9PQo=

清空

加密

解密

☐ 解密为UTF-8字节流

aGdhbWV7RDBudF9mMHI5ZXRfMI9DbE9zM19TMHVyY2VfbUBwfQ==

复制

再把aGdhbWV7RDBudF9mMHI5ZXRfMI9DbE9zM19TMHVyY2VfbUBwfQ==拿去解码

aGdhbWV7RDBudF9mMHI5ZXRfMl9DbE9zMl9TMHVyY2VfbUBwfQ==

清空

加密

解密

☐ 解密为UTF-8字节流

hgame{D0nt_f0r9et_2_C10s3_S0urce_m@p}

复制

得到flag