CISCN 2023 Writeup

队伍名: COMPASS-PAL

学校: 南方科技大学

成员:

1. 廖吉坤
2. 陈贲
3. 朱淳炜

1.签到卡 - MISC

题目描述

一个模拟打字机网站，输入python代码交互执行。打字机不能回退，只能全部重新输入。

思路

签到题难度应该不会高，考虑直接读文件

```
print(open('/flag').read())
```

拿到flag: `flag{dc0b5573-1204-4d34-8a71-331664c42d22}`

2.Sign_in_passwd - Crypto

题目描述

密码签到题，一眼base64换表

思路

所给的文件`flag.txt`中第一行为密文，第二行为编码表。第二行明显用url编码，先放到`CyberChef`里面得到`GHI3KLMNJOPQRSTUb=cdefghijklmnopWXYZ/12+406789VaqrstuvwxyzABCDEF5`，但是长度为65，猜测有一个字符被替换成了=，删掉末尾的`5`就跑出来了。

Exp:

```
import base64

str1 = "j2rXjx8yjd=YRZWyTIuwRdbyQdbqR3R9iZmsScutj2iqj3/tidj1jd=D"

string1 = "GHI3KLMNJOPQRSTUb=cdefghijklmnopWXYZ/12+406789VaqrstuvwxyzABCDEF"
string2 = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/"

print (base64.b64decode(str1.translate(str.maketrans(string1,string2))))
```

得到结果：`flag{8e4b2888-6148-4003-b725-3ff0d93a6ee4}`

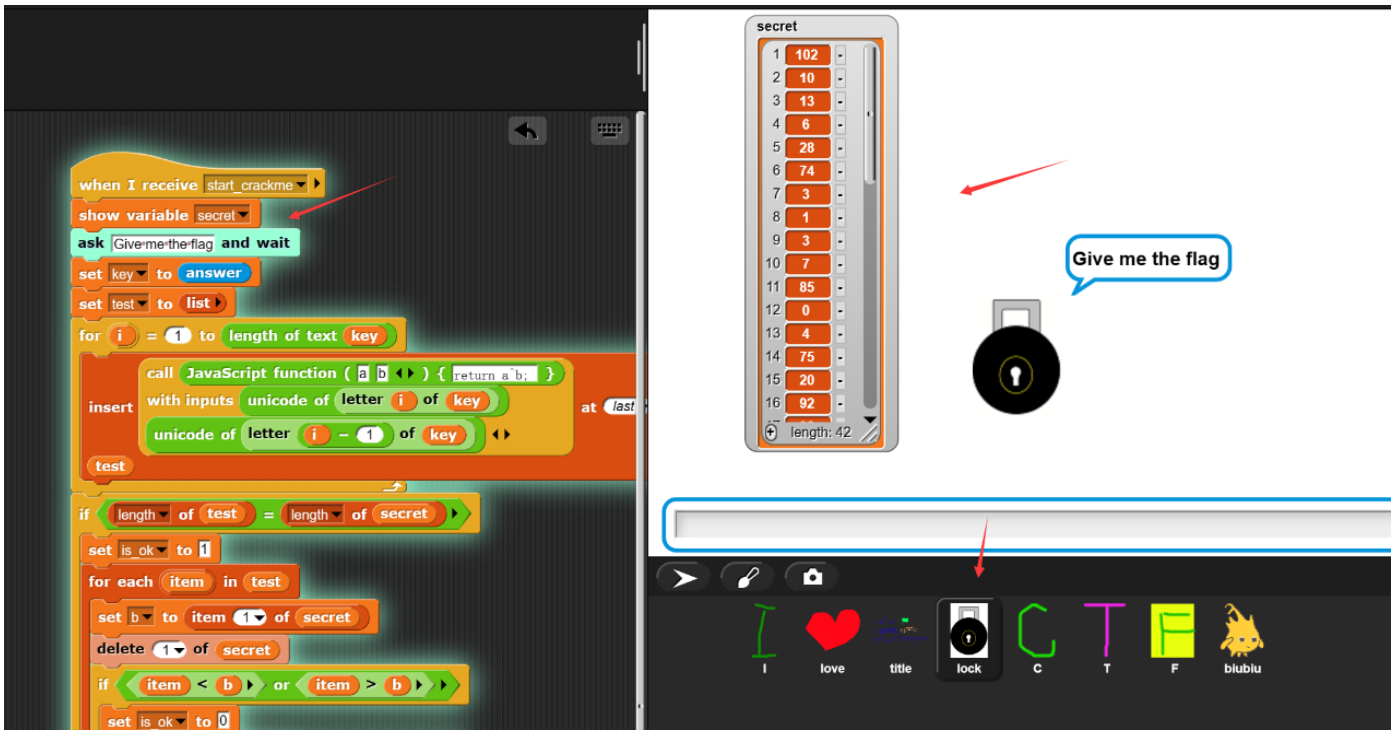
3.Baby_re - Reverse

题目描述

逆向签到题

思路

在snap网站中打开下发文件，在lock模块中输出secret，开启单步执行



右键选择secret后导出到本地

`102,10,13,6,28,74,3,1,3,7,85,0,4,75,20,92,92,8,28,25,81,83,7,28,76,88,9,0,29,73,0,86,4,87,87,82,84,85,4,85,87,30`

观察test的输入方式，可知加密方法为 $a[i] \oplus a[i-1]$

则将secret用如下脚本解密：

```
a=
[102,10,13,6,28,74,3,1,3,7,85,0,4,75,20,92,92,8,28,25,81,83,7,28,76,88,9,0,29,73,0,86,4
,87,87,82,84,85,4,85,87,30]
b=[0]*42
for i in range(0,42):
  b[i]=b[i-1]^a[i]
print("".join([chr(item) for item in b]))
```

得到结果：`flag{12307bbf-9e91-4e61-a900-dd26a6d0ea4c}`

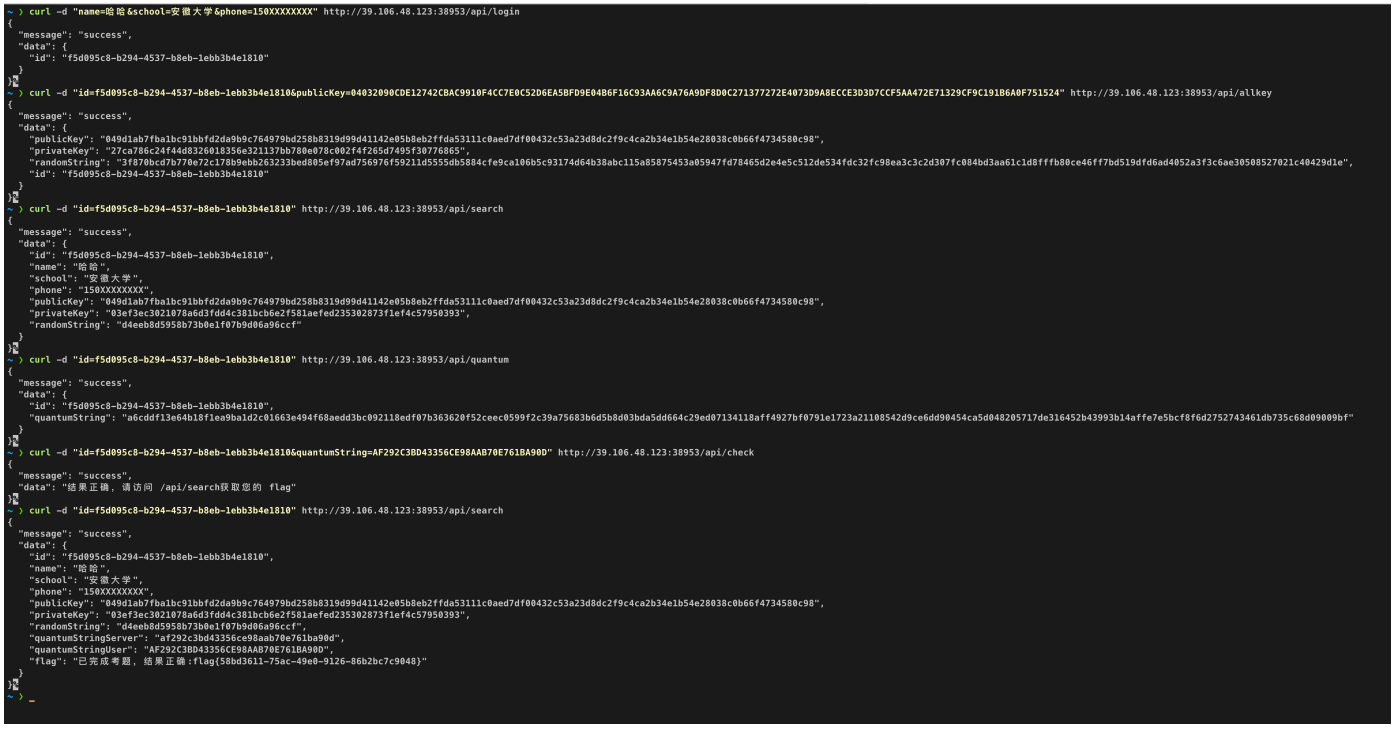
4.基于国密SM2算法的密钥分发 - Crypto

题目描述

情景题，按题面做就行了，但是实际上有个小漏洞。

思路

参照题面一步一步做，但是卡在随机串解密上（工具寄了）。但是神奇的是，可以直接访问/api/search查到密钥。于是这道题就这么轻易的完成了。实际操作图：



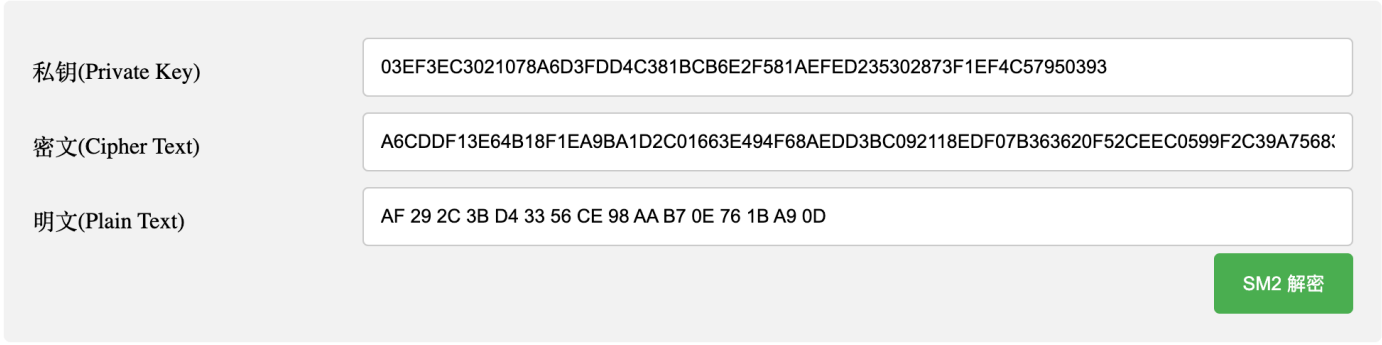
解密：

SM2 在线解密工具

SM2是一种公开密钥加密标准，由国家密码管理局于2010年12月17日发布，相关标准为“GM/T 0003-2012 《SM2椭圆曲线公钥密码算法》”。2016年，成为中国国家密码标准（GB/T 32918-2016）。

密文顺序为C1C3C2.因加入了随机数，每次加密的结果并不一样。
类似链接:https://the-x.cn/cryptography/Sm2.aspx

更新日志：
2022-04-19:发现不能解密超过35字节明文加密的数据。已修复。



得到结果：`flag{58bd3611-75ac-49e0-9126-86b2bc7c9048}`

5. 被加密的生产流量 - MISC

题目描述

流量审计题，挺吃经验和直觉的

思路

标题说是生产流量，故直接找modbus协议的流量。通过wireshark筛选`tcp.stream eq 0`，读取数据流中内容，每一段第一位字符拼起来得到编码`MMYWMX3GNEYWOXZRGAYDA`，猜测是base32，解码得到`c1f_f1lg_1000`

flag：`flag{c1f_f1lg_1000}`