

# 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 = "GHI3KLMNJOPQRSTUbcdefghijklmnopWXYZ/12+406789VagrstuvwxyzABCDEF"
string2 = "ABCDEFGHGIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/"

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

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

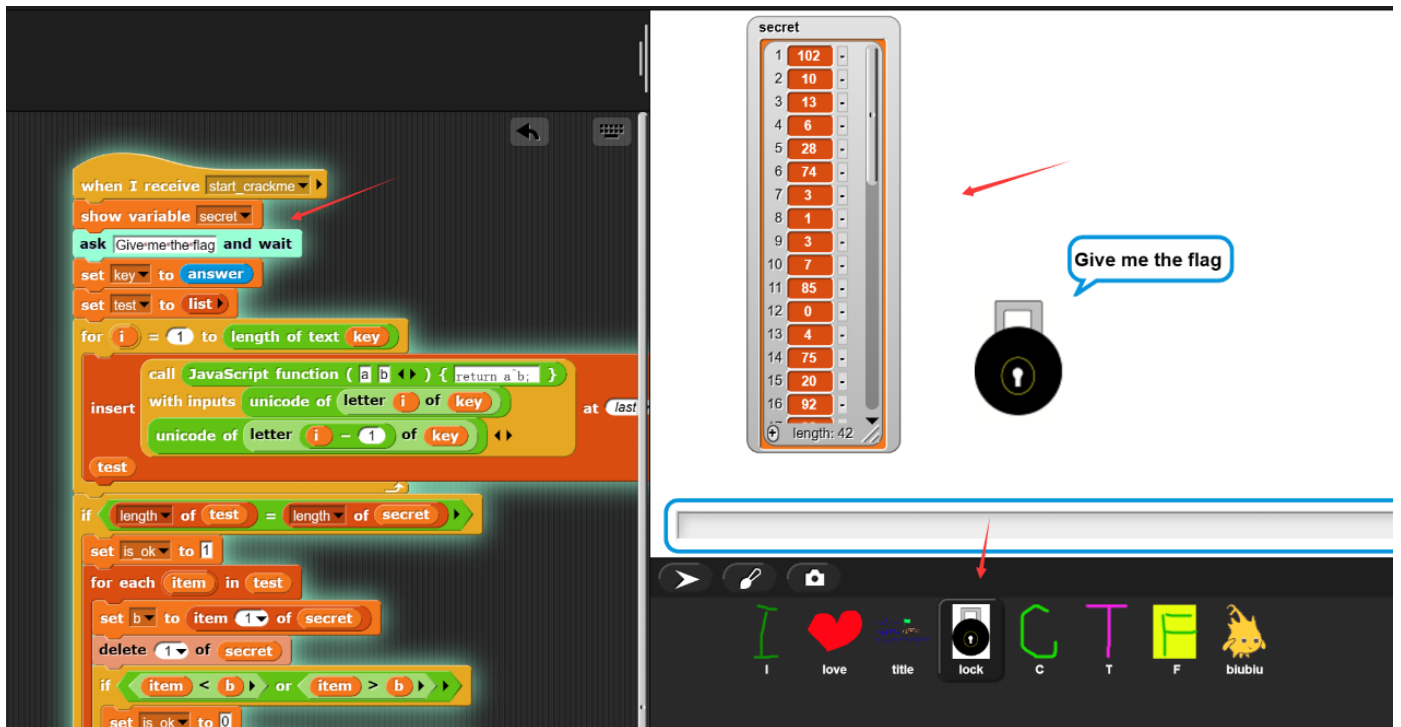
## 3.Baby\_re - Reverse

### 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：

```
>>> curl -d "name=哈哈&school=安徽大学&phone=150XXXXXXX" http://39.106.48.123:38953/api/login
{
  "message": "success",
  "data": {
    "id": "f5d095c8-b294-4537-b8eb-1ebb3b4e1810"
  }
}
>>> curl -d "id=f5d095c8-b294-4537-b8eb-1ebb3b4e1810&publicKey=04032090CDE12742CBAC9910F4C7E0C5206EASBFD9E0486F16C93AA6C9A76A90F8D0C271377272E4073D9A8ECCCE3D307CCF5AA472E71329CF9C191B6A0F751524" http://39.106.48.123:38953/api/allkey
{
  "message": "success",
  "data": {
    "publicKey": "049d1ab7fba1bc91bbfd2da9b9c764979bd258b8319d99d41142e05b8eb2ffda53111c0aed7df00432c53a23d8dc2f9c4ca2b34e1b54e28038c0b66f4734580c98",
    "privateKey": "27ca786c24f4d48326013356e321137bb780e078c002f4f265d7495f30776865",
    "randomString": "d4eeb8d5958b73bbe1f07b9d06a9ccf",
    "id": "f5d095c8-b294-4537-b8eb-1ebb3b4e1810"
  }
}
>>> curl -d "id=f5d095c8-b294-4537-b8eb-1ebb3b4e1810" http://39.106.48.123:38953/api/search
{
  "message": "success",
  "data": {
    "id": "f5d095c8-b294-4537-b8eb-1ebb3b4e1810",
    "name": "哈哈",
    "school": "安徽大学",
    "phone": "150XXXXXXX",
    "publicKey": "049d1ab7fba1bc91bbfd2da9b9c764979bd258b8319d99d41142e05b8eb2ffda53111c0aed7df00432c53a23d8dc2f9c4ca2b34e1b54e28038c0b66f4734580c98",
    "privateKey": "03ef3ec3021078a6d3fd4c381bcb6e2f581aefed235302873f1ef4c57950393",
    "randomString": "d4eeb8d5958b73bbe1f07b9d06a9ccf"
  }
}
>>> curl -d "id=f5d095c8-b294-4537-b8eb-1ebb3b4e1810" http://39.106.48.123:38953/api/quantum
{
  "message": "success",
  "data": {
    "id": "f5d095c8-b294-4537-b8eb-1ebb3b4e1810",
    "quantumString": "a6cddf13e4b18f1ea9ba1d2c01663e494f68aed3bc092118edf07b363620f52ceec8599f2c39a75683b6d5b8d03bda5dd664c29ed07134118aff4927bf0791e1723a21188542d9ce6dd98454ca5d048205717de316452b43993b14affe7e5bcf8f6d2752743461db735c68d0900btf"
  }
}
>>> curl -d "id=f5d095c8-b294-4537-b8eb-1ebb3b4e1810&quantumString=AF292C3BD43356CE98AAB70E761BA900" http://39.106.48.123:38953/api/check
{
  "message": "success",
  "data": "结果正确，请访问 /api/search 获取您的 flag"
}
>>> curl -d "id=f5d095c8-b294-4537-b8eb-1ebb3b4e1810" http://39.106.48.123:38953/api/search
{
  "message": "success",
  "data": {
    "id": "f5d095c8-b294-4537-b8eb-1ebb3b4e1810",
    "name": "哈哈",
    "school": "安徽大学",
    "phone": "150XXXXXXX",
    "publicKey": "049d1ab7fba1bc91bbfd2da9b9c764979bd258b8319d99d41142e05b8eb2ffda53111c0aed7df00432c53a23d8dc2f9c4ca2b34e1b54e28038c0b66f4734580c98",
    "privateKey": "03ef3ec3021078a6d3fd4c381bcb6e2f581aefed235302873f1ef4c57950393",
    "randomString": "d4eeb8d5958b73bbe1f07b9d06a9ccf",
    "quantumStringServer": "af292c3bd43356ce98a70e761ba900",
    "quantumStringUser": "AF292C3BD43356CE98AAB70E761BA900",
    "flag": "巴完成答题，结果正确:flag{50bd3611-75ac-49e0-9126-86b2bc7c9048}"
  }
}
```

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。

私钥(Private Key)	03EF3EC3021078A6D3FDD4C381BCB6E2F581AEFED235302873F1EF4C57950393
密文(Cipher Text)	A6CDDF13E64B18F1EA9BA1D2C01663E494F68AEDD3BC092118EDF07B363620F52CEEC0599F2C39A7568:
明文(Plain Text)	AF 29 2C 3B D4 33 56 CE 98 AA B7 0E 76 1B A9 0D

SM2 解密

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_filg_1000`

flag: `flag{c1f_filg_1000}`