2022年 HSC-1th中CRYPTO的AFFINE

原创

沐一 · 林 于 2022-02-28 09:36:47 发布 18 收藏

分类专栏： CTF 密码学 文章标签： ctf

本文链接：https://blog.csdn.net/xiao__1bai/article/details/123174322


CTF 同时被 2 个专栏收录

167 篇文章 6 订阅

订阅专栏


密码学

51 篇文章 1 订阅

订阅专栏

2022年 HSC-1th中CRYPTO的AFFINE

照例下载附件，是 py 文件：

```
# -*- coding: utf-8 -*-
import string
import hashlib

letter=string.ascii_letters+string.digits
#abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789

def encrypt(m, c, a, b):
    for i in range(len(m)):
        ch=m[i]
        t=(letter.index(ch) * a + b) % 62
        c.append(letter[t])
    d = ''.join(c)
    print(d)

m =
c = []
a =
b =

assert ("flag" in m)

print("加密后的密文为：")
Cipher = encrypt(m, c, a, b)
flag = hashlib.md5("".join(str(m)).encode("utf8")).hexdigest()
#print(flag)
"""
加密后的密文为：
xGJ13kkRK9QDfORQomFOf9NZs9LKVZvGqVIsVO9NOkorv
"""
```

.

.

可以看到 `a` 和 `b` 都不知道，一开始以为是简单的爆破类型，由于被 `assert ("flag" in m)` 迷惑住了，以为 `flag` 是开头四个，对应密文也是`开头`四个，结果怎么都爆破不出来，后来尝试着把密文重新加密，发现有奇效，然后配合爆破就爆破出来了。

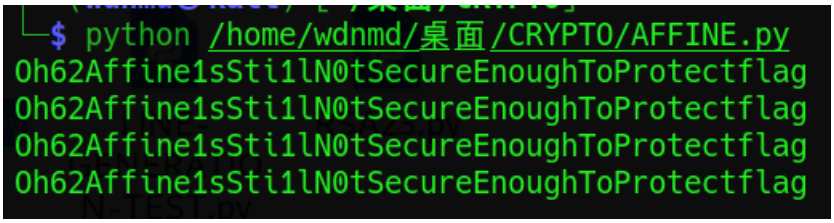
.

.

附上最终脚本，flag 竟然是最后四个，看来以后遇到 `assert ("flag" in m)` 要留个心眼了：

```
import string
import hashlib

letter=string.ascii_letters+string.digits
n = 'flag'
c = []
m='xGJ13kkRK9QDfORQomFOf9NZs9LKVZvGqVIsVO9NOkorv'
def encrypt(m, c, a, b):
 for i in range(len(m)):
  ch=m[i]
  t=(letter.index(ch) * a + b) % 62
  c.append(letter[t])
 d = ''.join(c)
 if 'flag' in d:
  print(d)
```



.

.

最终结果 `md5` 加密即可。

.

.

解毕！

敬礼！