2017第二届广东省强网杯线上赛--SimpleMath(DFS)

原创

前方是否可导？ 于 2020-07-30 22:29:26 发布 476 收藏 1

分类专栏： MD5

本文链接：https://blog.csdn.net/weixin_44110537/article/details/107702274

encrypt

```
from Crypto.Util.number import *
from hashlib import md5

flag = "xxx"
assert len(flag) == 15
pad = bytes_to_long(md5(flag).digest())

hack = 0

for char in flag:
 hack *= ord(char)
 hack += pad

print hack
# hack = 280098481791453837177137197730537158171743673148935867304957882116
# flag = "flag{" + flag + "}"
```

decrypt

首先,可以将得到的hack 进行分解.

由于得到md5是128位的数据,可以将所有满足条件的md5值求出.(采用二进制枚举)

然后从后面开始枚举每一个flag中字符,通过深搜的办法把所有可能的值找出来并配合遍历第一个flag的可能取值来对可能的值进行筛选,然后再打印出来.

```python
#print(len(str(pow(2, 127))))
import copy
import hashlib
import Crypto.Util.number
li=[2,2,19,31,59,97,127,3727,44948980991,1753609692783577883,556795634058750798159011]
hack=2800984817914538371771371977305371581717436731489358673049578821 16
md=[]
Try=1
for i in range(2048):
    Try=1
    s=bin(i)[2:].zfill(11)
    for j in range(len(s)):
        if s[j]=='1':
            Try*=li[j]
    if len(hex(Try)[2:])==32:
        md.append(Try)
def is_factor(n):
    li=[]
    for i in range(33,127):
        if n%i==0:
            li.append(i)
    return li
flag=[]
def dfs(count,n,value):
    global flag
    if count==14:
        s=''
        temp=copy.deepcopy(flag)
        temp.reverse()
        for i in temp:
            s+=chr(i)
        for i in range(33,127):
            x=chr(i)+s
            if value==Crypto.Util.number.bytes_to_long(hashlib.md5(x.encode()).digest()):
                print(x)
                break
        return
    li=is_factor(n-1)
    for i in range(len(li)):
        flag.append(li[i])
        #print(flag)
        dfs(count+1,(n-1)//li[i],value)
        flag.pop()
for i in md:
    dfs(0,hack//i,i)
    flag=[]
```