WEEK 3

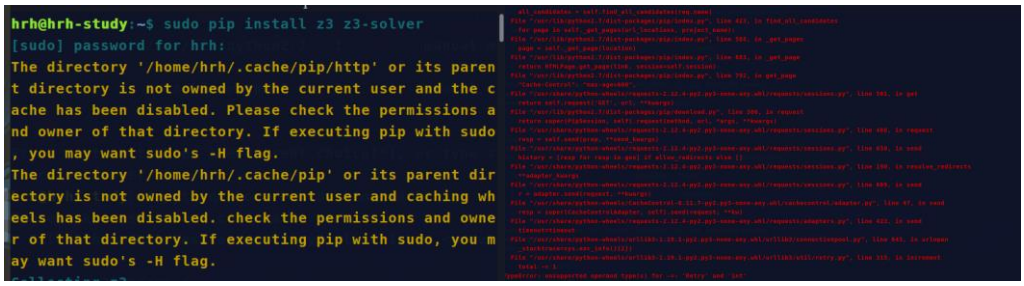
{Re}

[Math 简单]

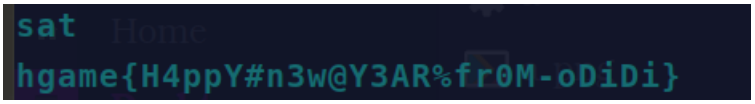
因为后来给出了题解的模板

```
wp_easy.py
from z3 import *

solver = Solver()
flag = [Int('flag%d' % i) for i in range(32)]

for i in flag:
    solver.add(i >= 32)
    solver.add(i <= 128)

solver.add(19*flag[8] + 33*flag[20] + ..... + 71*flag[0] == 145397)
solver.add(81*flag[10] + 54*flag[14] + .....  63*flag[22] == 127517)
solver.add(45*flag[30] + 94*flag[28] + .....  61*flag[18] == 141411)
solver.add(91*flag[20] + 9*flag[4] + 3 ..... *flag[0] == 117383)
solver.add(91*flag[21] + 92*flag[15] + ..... + 63*flag[10] == 156152)
# 省略
solver.add(3*flag[22] + 93*flag[8] + 7 ..... 31*flag[16] == 117383)
solver.add(90*flag[9] + 29*flag[4] + 3 ..... 56*flag[11] == 155741)
solver.add(85*flag[30] + 14*flag[10] + ..... + 32*flag[18] == 132804)
solver.add(45*flag[9] + 42*flag[18] + .....  + 79*flag[0] == 145568)
solver.add(40*flag[13] + 39*flag[10] + ..... 54*flag[30] == 130175)
solver.add(79*flag[0] + 15*flag[21] + .....  + 92*flag[30] == 171986)
solver.add(52*flag[12] + 40*flag[6] + .....  + 93*flag[0] == 151676)
solver.add(72*flag[0] + 8*flag[25] + 1 .....  49*flag[1] == 128223)
solver.add(89*flag[1] + 28*flag[24] + .....  + 18*flag[6] == 138403)

print('prepare okay')
check = solver.check()

print(check)

if check == sat:
    m = solver.model()
    s = []
    for i in range(32):
        s.append(chr(m[flag[i]].as_long()))
    print(''.join(s))
```

所以只需要先把方程全部输进去然后用 z3 处理就好。z3 我是放在 linux 下安装的，用的是 pip，但一开始用 sudo 有报错



后来先用了 su，然后再安装就好了。之后就没什么了，运行一下就能拿到 flag

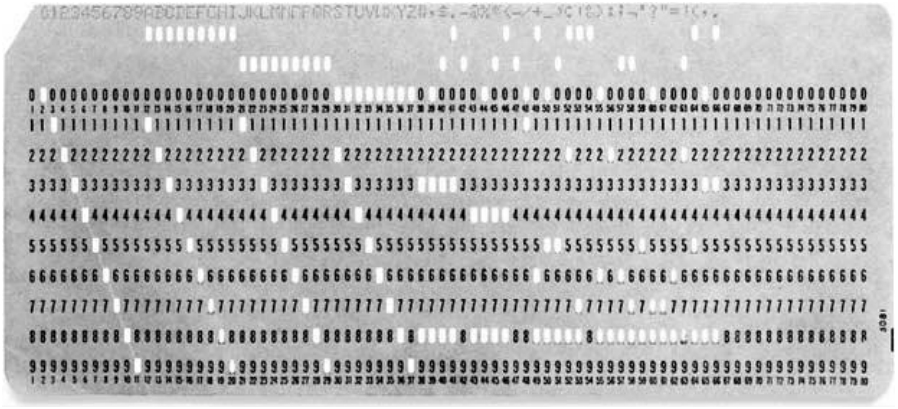


{Misc}

[旧时记忆]

我是出了两个 hint 才写出来的，直接百度搜存储器，历史，发现一个图片

看上去很像，发现是打孔卡片。然后就是去查它使用的方法，数字和字母很好找，但百度百科里没有直接给出特殊符号的值，只能去找图片对照。



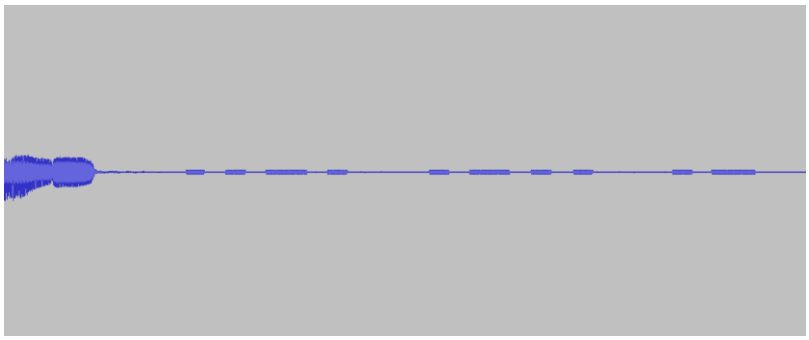
两个三孔特殊符号应该一个是_一个是%，最后 flag 为

```
12 11  0

A  J
B  K  S
C  L  T
D  M  U
E  N  V
F  O  W
G  P  X
H  Q  Y
I  R  Z

0LD_DAY5%M3MORY
```

[听听音乐？]

听到后面嘀嘀嘀的声音估计就是摩斯电码了，然后去找了个软件

Audacity，可以看见波形图



长的是 - ，短的是 . ,中间的间隔分成一个个字符，最后对照下来是

```
。。-。f
。-。。l
。-  a
--。g
---。。。
。----1
-t
。。--。-
。---j
。。-u
。。。。。5
-t
。。--。-
。。。。-4
。。--。-
。e
。-a
。。。s
-。--y
。。--。-
。--w
。-a
。。。-v
```

根据提示，字母全部大写，六个一组的应该是下划线，得到 flag

```
hgame{1T_JU5T_4_EASY_WAV}
```

{Crypto}

[babyRSA]

一开始怎么也跑不出来，后来发现 e 和 phi_n 不互质，查资料发现是先求出 e 和 phi_n 的最大公约数 gcd，这题是 4，然后把 e 先除 4，使得 e 和 phi_n 互质。然后用常规方法计算出 d，最后带入求出结果，但这个结果是原文的四次

方，然后开四次，最后转成字符。要用到一个 sympy 库，还是在 Linux 下安装比较方便。最后的脚本和 flag 如下

```python
from Crypto.Util.number import *
import sympy

def gcd(a,b):
    if a < b:
        a,b = b,a
    while b != 0:
        tem = a % b
        a = b
        b = tem
    return a

def invalidExponent(p,q,e,c):
    phiN = (p - 1) * (q - 1)
    n = p * q
    GCD = gcd(e, phiN)
    d = inverse(e//GCD,phiN)
    c = pow(c, d, n)
    plaintext = sympy.root(c, GCD)
    plaintext = long_to_bytes(plaintext)
    return plaintext


def main():
    e = 12
    p = 583800044303078033678069964607731236037903057890983844889520562066157682745 27
    q = 818595269757200606493800981936716128012005050291270765394576801554876696228 67
    c = 206087215323690202467878926681944491769659156726458690815919286163630886447291570510196171585626143608988384615185921752409380788006476576337410136447460

    plaintext = invalidExponent(p,q,e,c)
    print (plaintext)

main()
```

```
hgame{xxxxxxx}
```

[basicmath]

先去根据 hint 找到算法

奇素数

先来考虑当 p 是奇素数时的情况。

根据欧拉准则，当 a 是 $\bmod p$ 的平方剩余时有

$$a^{\frac{p-1}{2}} \equiv 1 \pmod{p}$$

首先设 $p-1=2^t \cdot s, (2 \nmid s)$

1° 当 $t=1$ 时，显然有

$$\sqrt{a} = \sqrt{a^{\frac{p-1}{2}} \cdot a} = a^{\frac{s+1}{2}}$$

所以 $x = a^{\frac{s+1}{2}}$ 就是方程的一个解了（因为 s 是奇数，所以 $\frac{s+1}{2}$ 是整数可以直接计算）

发现题目是符合这种条件的特殊情况，用这样的方法试一下，得到一个解发现，拿去转成字符串发现是乱码。。。只能询问下出题人，得知应有两个解，flag 是另一个解。但一开始不知道怎么找第二个解，用一个特例去看看。取

p=19，a=7，易知 8 和 11 是两个解，但直接代入只能算出 11，算不出 8，最后发现把 a 取成-a 即可得到 8。代回原来的方程，得到两个解

```
232828321890052373500842932806925222465025676596844604612122594734846587450748673989604439470234591202189447038449025024582
```

flag 是上面那个。转成字符串

```
>>> long_to_bytes(2328283218900523735008429328069252224650256765)
'hgame{easy_Crypto!}'
```