1、EzSignIn

打开虚拟机

nc 47.102.130.35 32119 得到flag

2、SignIn(MISC)

用画图改变图片的高度，得到flag

3、ezRSA

通过分析运算：

```
n=p*q

leak1=pow(p,q,n)

leak2=pow(q,p,n)
```

得到p,q就是leak1，leak2

d=phi=(p-1)*(q-1)

下面通过通用代码处理rsa问题：

```
# -*- coding: utf-8 -*-
def gcd(a, b):
    if b == 0:
        return a
    else:
        return gcd(b, a % b)


def ext_gcd(a, b):
    if b == 0:
        x1 = 1
        y1 = 0
        x = x1
        y = y1
        r = a
        return r, x, y
    else:
        r, x1, y1 = ext_gcd(b, a % b)
        x = y1
        y = x1 - a // b * y1
        return r, x, y
import time
def exp_mode(base, exponent, n):
    bin_array = bin(exponent)[2:][::-1]
    r = len(bin_array)
    base_array = []

    pre_base = base
    base_array.append(pre_base)

    for _ in range(r - 1):
        next_base = (pre_base * pre_base) % n
        base_array.append(next_base)
        pre_base = next_base
```

```
    a_w_b = __multi(base_array, bin_array, n)
    return a_w_b % n

def __multi(array, bin_array, n):
    result = 1
    for index in range(len(array)):
        a = array[index]
        if not int(bin_array[index]):
            continue
        result *= a
        result = result % n # 加快连乘的速度
    return result
import time

# 生成公钥私钥，p、q为两个超大质数
def gen_key(p, q):
    n = p * q
    fy = (p - 1) * (q - 1)      # 计算与n互质的整数个数 欧拉函数
    e = 65537                    # 选取e    一般选取65537，因为65537展开成二进制的时候，1比较少，0很多，所以加密速度很快
    # generate d
    a = e
    b = fy
    r, x, y = ext_gcd(a, b)
    # 计算出的x不能是负数，如果是负数，说明p、q、e选取失败，不过可以把x加上fy，使x为正数，才能计算。
    if x < 0:
        x = x + fy
    d = x
    # 返回：    公钥     私钥
    return    (n, e), (n, d)

# 加密 m是被加密的信息 加密成为c
def encrypt(m, pubkey):
    n = pubkey[0]
    e = pubkey[1]

    c = exp_mode(m, e, n)
    return c

# 解密 c是密文，解密为明文m
def decrypt(c, selfkey):
    n = selfkey[0]
    d = selfkey[1]

    m = exp_mode(c, d, n)
    return m


if __name__ == "__main__":
    #'''公钥私钥中用到的两个大质数p,q，都是1024位'''
    p =14912717007361127196818257675129033155901844180572531042609541283758922767075754074392986585365039983910283
    q =11612299271467091538130991696749043648902000117288064416717991546702179489292797727208059664178556911913425
    #'''生成公钥私钥'''
    pubkey, selfkey = gen_key(p, q)
    #'''需要被加密的信息转化成数字，长度小于秘钥n的长度，如果信息长度大于n的长度，那么分段进行加密，分段解密即可。'''
    m =10529481867532520034258056773864074017027019578041866245400647840230251661652999709715919620810933437191661
    print("待加密信息-->%s" % m)
    #'''信息加密，m被加密的信息，c是加密后的信息'''
    c = encrypt(m, pubkey)
    print("被加密后的密文-->%s" % c)
    #'''信息解密'''
    d = decrypt(c, selfkey)
    print("被解密后的明文-->%s" % d)
    #pubkey,selfkey都为元组，pudkey[0]=selfkey[0]=n,pubkey[1]=e,selfkey[1]=d
    print("直接解密结果为")
    print(pow(m,selfkey[1],p*q))
```

直接取最后得到的直接结果：
22300522817737044001082695383721958821501141507949255067191412827214153791352322382693802972815

通过long_to_bytes()得到flag

4、ezIDA

直接拖到ida内部查看

主界面发现flag

5、签到

关注公众号发送消息得到flag