[安洵杯2021] crypto little trick Writeup

原创

_bestkasscn 于 2021-11-29 21:13:32 发布 196 收藏

分类专栏： CTF 文章标签： 密码学

本文链接：https://blog.csdn.net/bestkasscn/article/details/121619769

[安洵杯2021] crypto little trick Writeup

题目：

```
from Crypto.Util.number import sieve_base, bytes_to_long, getPrime
import random
import gmpy2
import os

flag = b'D0g3{}'
flag = bytes_to_long(flag)
p = getPrime(1024)
q = getPrime(1024)
n = p * q
e = gmpy2.next_prime(bytes_to_long(os.urandom(3)))
c = gmpy2.powmod(flag,e,n)
print(p)
print(q)
print(c)

dp = ''
seeds = []
for i in range(0,len(dp)):
    seeds.append(random.randint(0,99))
print(seeds)

result = []
for j in range(0,len(dp)):
    random.seed(seeds[j])
    rands = []
    for k in range(0,4):
        rands.append(random.randint(0,99))
    result.append((~ord(dp[j])|rands[j%4]) & (ord(dp[j])|~rands[j%4]))
    del rands[j%4]
    print(rands)
print(result)

dq = ''
C = []
E = 0x10001
```

```
list_p = sieve_base[0:len(dq)]
list_q = sieve_base[len(dq):2*len(dq)]
for l in range(0,len(dq)):
    P = list_p[l]
    Q = list_q[l]
    C.append(pow(int(dq[l]),E,P*Q))
print(C)
#119494148343917708105807117614773529196380452025859574123211538859983094108015678321724495609785332508563534950
#957367289723559468197440246960403054020452985281797756117166991826626612422135797192886041925043855329391156291955066822268279533978514896151007690729926904044407542983781817530576308669792533266431

#125132685086281666800573404868585424815247082213724647473226016452471461555742194042617318063670311290694310562
#746442372293133509175379170933514423842462487594186286854028887049828613566072663640036114898823281310177406827049478153958964127866484011400391821374773362883518683538899757137598483532099590137741

#102382713154774882253317126410832900244888117100930337345359105734934095670569345281108450491431938367061222103
#030554661458192568932934292233898282526574260301185341276842652611925034062874089328323409383434479979163443506
#836638396592899163753687522351127758368557931478154764860266639165630670332197168080397798271140797924897991051366573235585952350072953406990940829202438122519224038535132599979820636694910636253737645266226451201277058645178371262666506516170412653674275505483042786498278203083483738854481117227949665777688420975606905681275047666950864081736942323849693035772584276891879134709550428336803 2

#[3, 0, 39, 78, 14, 49, 73, 83, 55, 48, 30, 28, 23, 16, 54, 23, 68, 7, 20, 8, 98, 68, 45, 36, 97, 13, 83, 68, 16, 59, 81, 26, 51, 45, 36, 60, 36, 94, 58, 11, 19, 33, 95, 12, 60, 38, 51, 95, 21, 3, 38, 72, 47, 80, 7, 20, 26, 80, 18, 43, 92, 4, 64, 93, 91, 12, 86, 63, 46, 73, 89, 5, 91, 17, 88, 94, 80, 42, 90, 14, 45, 53, 91, 16, 28, 81, 62, 63, 66, 20, 81, 3, 43, 99, 54, 22, 2, 27, 2, 62, 88, 99, 78, 25, 76, 49, 28, 96, 95, 57, 94, 53, 32, 58, 32, 72, 89, 15, 4, 78, 89, 74, 86, 45, 51, 65, 13, 75, 95, 42, 20, 77, 34, 66, 56, 20, 26, 18, 28, 11, 88, 62, 72, 27, 74, 42, 63, 76, 82, 97, 75, 92, 1, 5, 20, 78, 46, 85, 81, 54, 64, 87, 37, 91, 38, 39, 1, 90, 61, 28, 13, 60, 37, 90, 87, 15, 78, 91, 99, 58, 62, 73, 70, 56, 82, 5, 19, 54, 76, 88, 4, 3, 55, 3, 3, 22, 85, 67, 98, 28, 32, 42, 48, 96, 69, 3, 83, 48, 26, 20, 45, 16, 45, 47, 92, 0, 54, 4, 73, 8, 31, 38, 3, 10, 84, 60, 59, 69, 64, 91, 98, 73, 81, 98, 9, 70, 44, 44, 24, 95, 83, 49, 31, 19, 89, 18, 20, 78, 86, 95, 83, 23, 42, 51, 95, 80, 48, 46, 88, 7, 47, 64, 55, 4, 62, 37, 71, 75, 98, 67, 98, 58, 66, 70, 24, 58, 56, 44, 11, 78, 1, 78, 89, 97, 83, 72, 98, 12, 41, 33, 14, 40, 27, 5, 18, 35, 25, 31, 69, 97, 84, 47, 25, 90, 78, 15, 72, 71]

#[54, 36, 60] [84, 42, 25] [20, 38, 39] [81, 9, 92] [70, 65, 94] [6, 11, 75] [27, 50, 46] [49, 85, 8] [95, 14, 73] [54, 71, 30] [53, 28, 65] [11, 13, 59] [94, 89, 8] [36, 41, 44] [91, 13, 48] [92, 94, 89] [94, 74, 90] [32, 65, 7] [90, 68, 90] [22, 96, 12] [83, 35, 5] [74, 74, 90] [27, 48, 33] [32, 98, 95] [80, 37, 84] [25, 68, 84] [49, 85, 37] [74, 94, 74] [48, 41, 44] [22, 94, 2] [50, 45, 38] [74, 20, 20] [50, 16, 82] [27, 8, 33] [32, 98, 91] [30, 57, 26] [98, 95, 91] [54, 28, 43] [58, 20, 94] [45, 55, 92] [78, 52, 51] [57, 81, 27] [76, 51, 53] [47, 65, 66] [57, 26, 80] [63, 72, 6] [24, 50, 82] [76, 51, 99] [68, 63, 47] [23, 36, 60] [63, 42, 6] [7, 59, 98] [43, 45, 34] [27, 70, 95] [32, 15, 7] [90, 68, 76] [20, 20, 60] [27, 70, 95] [18, 66, 19] [3, 69, 14] [56, 55, 58] [23, 39, 15] [47, 63, 92] [91, 49, 56] [17, 68, 16] [47, 66, 14] [79, 3, 31] [44, 29, 90] [39, 58, 85] [27, 56, 46] [8, 60, 14] [62, 74, 79] [17, 68, 16] [52, 96, 28] [39, 18, 62] [54, 12, 28] [54, 70, 95] [63, 27, 22] [20, 9, 58] [10, 70, 65] [48, 8, 33] [61, 45, 71] [8, 17, 16] [36, 48, 41] [13, 59, 17] [50, 55, 38] [92, 17, 23] [44, 29, 90] [43, 24, 44] [90, 76, 90] [50, 45, 38] [23, 54, 36] [69, 14, 46] [40, 17, 24] [91, 13, 48] [95, 14, 2] [94, 5, 8] [64, 95, 19] [95, 94, 8] [92, 17, 97] [18, 90, 62] [40, 17, 24] [81, 9, 73] [37, 92, 84] [95, 20, 29] [6, 11, 75] [11, 13, 17] [37, 90, 39] [51, 99, 53] [4, 1, 51] [54, 12, 43] [61, 89, 45] [21, 30, 90] [58, 64, 94] [7, 21, 90] [7, 59, 98] [60, 99, 14] [96, 73, 15] [23, 10, 15] [81, 9, 92] [60, 99, 14] [85, 11, 12] [79, 3, 31] [27, 48, 8] [50, 16, 82] [41, 84, 44] [25, 68, 84] [45, 43, 4] [51, 99, 53] [63, 27, 22] [90, 68, 90] [79, 32, 24] [58, 84, 89] [7, 24, 44] [96, 55, 52] [90, 68, 76] [20, 20, 60] [18, 33, 19] [11, 13, 17] [45, 55, 92] [18, 90, 62] [92, 97, 23] [7, 59, 34] [64, 70, 95] [51, 11, 12] [63, 27, 22] [44, 29, 48] [37, 95, 20] [48, 50, 96] [19, 37, 84] [45, 43, 76] [42, 56, 55] [84, 76, 25] [62, 79, 94] [90, 68, 90] [81, 9, 92] [39, 58, 85] [19, 10, 90] [50, 45, 38] [91, 13, 55] [63, 40, 92] [14, 83, 54] [68, 9, 84] [8, 17, 68] [42, 72, 6] [20, 19, 39] [13, 84, 25] [20, 9, 65] [55, 80, 32] [11, 59, 17] [25, 68, 84] [30, 57, 26] [9, 61, 84] [20, 65, 58] [14, 18, 54] [96, 1, 73] [9, 92, 73] [8, 68, 16] [40, 20, 24] [58, 20, 64] [17, 97, 23] [27, 56, 46] [90, 29, 13] [96, 55, 47] [48, 50, 96] [62, 79, 94] [67, 78, 51] [91, 13, 55] [95, 20, 29] [39, 90, 62] [23, 10, 15] [23, 54, 36] [95, 14, 73] [23, 36, 60] [23, 54, 60] [95, 14, 2] [61, 10, 90] [7, 97, 41] [35, 83, 5] [11, 13, 59] [21, 30, 90] [63, 27, 22] [54, 13, 30] [37, 90, 39] [9, 16, 60] [23, 36, 60] [49, 85, 37] [54, 13, 71] [20, 20, 60] [90, 76, 90] [27, 48, 33] [36, 48, 41] [48, 8, 33] [35, 45, 34] [42, 56, 58] [84, 75, 42] [13, 55, 48] [23, 39, 15] [27, 50, 46] [22, 96, 12] [11, 39, 68] [63, 72, 6] [23, 54, 60] [57, 42, 57] [91, 3, 0] [30, 26, 80] [22, 93, 2] [68, 9, 16]
```

```
[63, 40, 92] [8, 68, 16] [35, 83, 5] [27, 50, 56] [45, 55, 38] [35, 35, 5] [46, 37, 86] [90, 29, 45] [54, 86, 17] [40, 86, 17] [71, 83, 99] [76, 51, 99] [85, 8, 37] [6, 11, 75] [1, 11, 68] [67, 78, 52] [60, 99, 14] [18, 33, 19] [90, 68, 90] [81, 9, 92] [3, 83, 31] [76, 99, 53] [49, 85, 37] [92, 94, 89] [2, 27, 22] [24, 16, 82] [76, 51, 53] [27, 54, 70] [13, 71, 30] [88, 58, 85] [39, 18, 62] [32, 15, 65] [43, 45, 34] [47, 40, 92] [9, 95, 73] [23, 10, 39] [17, 97, 23] [68, 61, 84] [32, 62, 98] [45, 43, 4] [83, 35, 5] [7, 97, 41] [35, 83, 5] [58, 20, 64] [43, 24, 44] [90, 45, 13] [71, 83, 99] [58, 20, 64] [55, 47, 52] [40, 86, 17] [45, 55, 46] [81, 9, 92] [84, 76, 25] [81, 92, 73] [8, 60, 14] [19, 80, 37] [85, 8, 37] [7, 98, 34] [35, 83, 5] [47, 65, 66] [23, 16, 91] [57, 81, 27] [10, 70, 94] [45, 87, 3] [70, 95, 19] [62, 79, 94] [18, 66, 19] [54, 75, 74] [92, 84, 21] [1, 39, 68] [68, 9, 60] [19, 80, 37] [91, 3, 0] [35, 45, 34] [37, 92, 21] [20, 9, 65] [9, 92, 73] [96, 73, 15] [7, 59, 34] [32, 62, 0]

#[-38, -121, -40, -125, -51, -29, -2, -21, -59, -54, -51, -40, -105, -5, -4, -50, -127, -56, -124, -128, -23, -104, -63, -112, -34, -115, -58, -99, -24, -102, -1, -5, -34, -3, -104, -103, -21, -62, -121, -24, -115, -9, -87, -56, -39, -30, -34, -4, -33, -5, -114, -21, -19, -7, -119, -107, -115, -6, -25, -27, -32, -62, -28, -20, -60, -121, -102, -10, -112, -7, -85, -110, -62, -100, -110, -29, -41, -55, -113, -112, -45, -106, -125, -25, -57, -27, -83, -2, -51, -118, -2, -10, -50, -40, -1, -82, -111, -113, -50, -48, -23, -33, -112, -38, -29, -26, -4, -40, -123, -4, -44, -120, -63, -38, -41, -22, -50, -50, -17, -122, -61, -5, -100, -22, -44, -47, -125, -125, -127, -55, -117, -100, -2, -26, -32, -111, -123, -118, -16, -24, -20, -40, -92, -40, -102, -49, -99, -45, -59, -98, -49, -13, -62, -128, -121, -114, -112, -13, -3, -4, -26, -35, -15, -35, -8, -18, -125, -14, -6, -60, -113, -104, -120, -64, -104, -55, -104, -41, -34, -106, -105, -2, -28, -14, -58, -128, -3, -1, -17, -38, -18, -12, -59, -4, -19, -82, -40, -122, -18, -42, -53, -60, -113, -40, -126, -15, -63, -40, -124, -114, -58, -26, -35, -26, -8, -48, -112, -52, -11, -117, -52, -32, -21, -38, -124, -13, -103, -6, -30, -33, -28, -31, -1, -97, -59, -64, -28, -1, -40, -2, -10, -26, -24, -3, -50, -113, -125, -122, -124, -5, -50, -62, -11, -8, -88, -109, -7, -31, -105, -54, -28, -8, -62, -58, -101, -58, -53, -124, -18, -124, -17, -109, -52, -45, -40, -109, -85, -7, -108, -121, -58, -49, -91, -102, -8, -10, -17, -55, -19, -11, -116, -47, -120, -121, -23, -99, -19, -51, -36, -110, -126, -29, -110, -9, -97, -54, -83, -86]

#[1, 0, 7789, 1, 17598, 20447, 15475, 23040, 41318, 23644, 53369, 19347, 66418, 5457, 0, 1, 14865, 97631, 6459, 36284, 79023, 1, 157348, 44667, 185701, 116445, 23809, 220877, 0, 1, 222082, 30333, 55446, 207442, 193806, 149389, 173229, 349031, 152205, 1, 149157, 196626, 1, 222532, 10255, 46268, 171536, 0, 351788, 152678, 0, 172225, 109296, 0, 579280, 634746, 1, 668942, 157973, 1, 17884, 662728, 759841, 450490, 0, 139520, 157015, 616114, 199878, 154091, 1, 937462, 675736, 53200, 495985, 307528, 1, 804492, 790322, 463560, 520991, 436782, 762888, 267227, 306436, 1051437, 384380, 505106, 729384, 1261978, 668266, 1258657, 913103, 935600, 1, 1, 401793, 769612, 484861, 1024896, 517254, 638872, 1139995, 700201, 308216, 333502, 0, 0, 401082, 1514640, 667345, 1015119, 636720, 1011683, 795560, 783924, 1269039, 5333, 0, 368271, 1700344, 1, 383167, 7540, 1490472, 1484752, 918665, 312560, 688665, 967404, 922857, 624126, 889856, 1, 848912, 1426397, 1291770, 1669069, 0, 1709762, 130116, 1711413, 1336912, 2080992, 820169, 903313, 515984, 2211283, 684372, 2773063, 391284, 1934269, 107761, 885543, 0, 2551314, 2229565, 1392777, 616280, 1368347, 154512, 1, 1668051, 0, 2453671, 2240909, 2661062, 2880183, 1376799, 0, 2252003, 1, 17666, 1, 2563626, 251045, 1593956, 2215158, 0, 93160, 0, 2463412, 654734, 1, 3341062, 3704395, 3841103, 609968, 2297131, 1942751, 3671207, 1, 1209611, 3163864, 3054774, 1055188, 1, 4284662, 3647599, 247779, 0, 176021, 3478840, 783050, 4613736, 2422927, 280158, 2473573, 2218037, 936624, 2118304, 353989, 3466709, 4737392, 2637048, 4570953, 1473551, 0, 0, 4780148, 3299784, 592717, 538363, 2068893, 814922, 2183138, 2011758, 2296545, 5075424, 1814196, 974225, 669506, 2756080, 5729359, 4599677, 5737886, 3947814, 4852062, 1571349, 4123825, 2319244, 4260764, 1266852, 1, 3739921, 1, 5948390, 1, 2761119, 2203699, 1664472, 3182598, 6269365, 5344900, 454610, 495499, 6407607, 1, 1, 476694, 4339987, 5642199, 1131185, 4092110, 2802555, 0, 5323448, 1103156, 2954018, 1, 1860057, 128891, 2586833, 6636077, 3136169, 1, 3280730, 6970001, 1874791, 48335, 6229468, 6384918, 5412112, 1, 7231540, 7886316, 2501899, 8047283, 2971582, 354078, 401999, 6427168, 4839680, 1, 44050, 3319427, 0, 1, 1452967, 4620879, 5525420, 5295860, 643415, 5594621, 951449, 1996797, 2561796, 6707895, 7072739]
```

题目后面的代码可以忽略掉，核心就是已知上面的rsa部分。已知p,q,c,我们要求d还需要e，题目没给出e，但是通过e的算法

```
e = gmpy2.next_prime(bytes_to_long(os.urandom(3)))
```

可知e的范围大致小于10000000，所以直接写脚本爆破e就能拿flag

exp：

```python
import gmpy2
from Crypto.Util.number import *

p = 11949414834391770810580711761477352919638045202585957412321153885998309410801567832172449560978533250856353495095736728972355946819744024696040305402045298528179775611716699182662661242213579719288604192504385532939115629195506682226827953397851489615100769072992690404440754298378181753057630866979253326643 1

q = 12513268508628166680057340486858542481524708221372464747322601645247146155574219404261731806367031129069431056274644237229313350917537917093351442384246248759418628685402888704982861356607266364003611489882328131017740682704947815395896412786648401140039182137477336288351868353889975713759848353209959013774 1

c = 10238271315477488225331712641083290024488811710093033734535910573493409567056934528110845049143193836706122210303055466145819256893293429223389828252657426030118534127684265261192503406287408932832340938343447997791634435068366383965928991637536875223511277583685579314781547648602666391656306703321971680803977982711407979248979910513665732355859523500729534069909408292024381225192240385351325999798206366949106362537376452662264512012770586451783712626665065161704126536742755054830427864982782030834837388544811172279496657776884209756069056812750476669508640817369423238496930357725842768918791347095504283368032 1

e = 2
while True:
    e = gmpy2.next_prime(e)
    print(e)
    try:
        d = gmpy2.invert(e, (p - 1) * (q - 1))
        res = long_to_bytes(pow(c, d, p * q))
        if b'D0g3' in res:
            print(res)
            break
    except:
        pass
```

```
D0g3{Welc0me_t0_iSOON_4nd_have_4_go0d_time}
```

这题大概要跑二十多分钟才能跑出来，我跑了几分钟就没跑了导致这题没出，真是太菜了。。