Научно-исследовательская практика. Cryptohack: Lack of Entropy (Firebird Internal CTF)

Винников Кирилл, Затирахин Кирилл

Июль 2023

Условие задачи

Дана программа, шифрующая исходное сообщение с помощью алгоритма RSA. Также дан файл с выводом представленной программы. Конкретно: модуль по которому ведутся вычисления n, экспонента шифрования e, зашифрованное сообщение c
Необходимо расшифровать исходное сообщение.
Ниже представлен фрагмент исходного кода задачи, где генерируются простые p и q

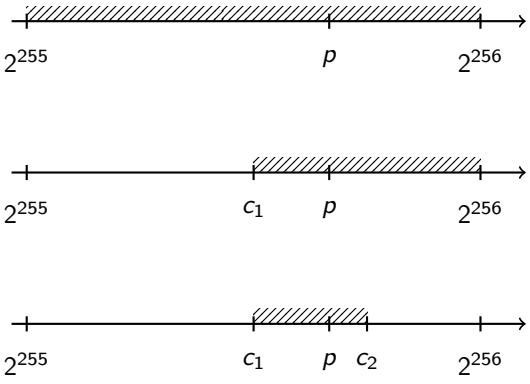
```
#e = 65537
while True:
    p = random.getrandbits(256)
    q = int(gmpy2.digits(p, 3))
    # Ensure that they are prime
    if not gmpy2.is_prime(p): continue
    if not gmpy2.is_prime(q): continue
    # Ensure that d exists
    if (p-1) % e == 0: continue
    if (q-1) % e == 0: continue
    break
```

Метод решения

Число $\boldsymbol{n}$ зависит от простых чисел $\boldsymbol{p}$ и $\boldsymbol{q}$. В свою очередь, число $\boldsymbol{q}$ напрямую зависит от $\boldsymbol{p}$. Число $\boldsymbol{p}$ генерируется длиной 256 бит, то оно находится в промежутке от 2^{255} до 2^{256}.

Таким образом, задача сводится к поиску такого простого числа $\boldsymbol{p}$, что полученный модуль совпадет с исходным. Если это произойдет, можно легко вычислить функцию Эйлера φ и найти обратный эелемент по отношению к экспоненте $\boldsymbol{e}$ и расшифровать сообщение.

Визуализация



Решение (фрагмент кода)

```
first = gmpy2.next_prime(2**255)
last = gmpy2.next_prime(2**256)
p = -1
q = -1
while (first <= last) and (p == -1):
    p0 = gmpy2.next_prime((first+last)//2)
    q0 = int(gmpy2.digits(p0, 3))
    N=p0*q0
    if (N == n):
        p = p0
        q = q0
    else:
        if (N<n):
            first = p0 +1
        else:
            last = p0 -1
```

Вывод и проверка результата

```
p= 108620945043711001039798448113818516279249705471113441037537257405725491373069
q= 112220201022020222100120210102110122002101001002212110110100021000102102200001012202100121122111101021200100201212022020112011202202021212112000000021110001102221

m= 19895241581069760683884624774316151662118492279731665153746489481848283504352617916863 7

m= b'firebird{b1n4ry_s34rch_f0r_th3_w1n!}'
```