

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Университет ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА № 2.1

по дисциплине

‘Информационная безопасность’

‘Атака на алгоритм шифрования RSA посредством метода Ферма’

Вариант №19

Выполнил:

Студент группы Р34111

Павлов Александр

Сергеевич

Преподаватель:

Маркина Т.А.



Санкт-Петербург, 2024

Цель работы

Изучить атаку на алгоритм шифрования RSA посредством метода Ферма.

Программные и аппаратные средства

Процессор: Intel Core i5-11400F 2.6GHz 12 ядер

Видеокарта: NVIDIA GeForce RTX 2060

Объем оперативной памяти: 32 GB

Язык программирования: Python

Задание

19	59046883376179	4044583	32279109612093 17838629182964 4165776716262 13093284635895 20048651313008 54626454832531 12801053743903 54675332003643 4544911979279 31928373564570 798945495513 19569174668782
----	----------------	---------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Ход работы

1. Вычисляем $n = \lfloor \sqrt{N} \rfloor + 1 = 7684197$
2. Вычисляем $t_1 = n + 1$
3. Возводим t_1 в квадрат
4. Вычисляем $w_1 = t_1^2 - N$
5. Проверяем, если w_1 – квадрат целого числа. Если нет – возвращаемся ко второму пункту $t_i = t_{i-1} + 1$
6. Вычисляем $p = t + \sqrt{w} = 7692977$
7. Вычисляем $q = t - \sqrt{w} = 7675427$
8. Вычисляем $\phi(N) = (p - 1)(q - 1) = 59046868007776$
9. Вычисляем $d = e^{-1} \bmod \phi(N) = 31944145322807$
10. Производим дешифрацию шифрблока $m = C^d \bmod N$
11. Переводим дешифрацию шифрблока в текстовый вид

Листинг разработанной программы с комментариями

```
from math import sqrt
```

```
N = 59046883376179
```

```

e = 4044583
C = "32279109612093
17838629182964
4165776716262
13093284635895
20048651313008
54626454832531
12801053743903
54675332003643
4544911979279
31928373564570
798945495513
19569174668782"

```

```

answer = ""

```

```

n = int(sqrt(N)) + 1

```

```

i = 0

```

```

while True:

```

```

    i += 1

```

```

    t = n + i

```

```

    w = t ** 2 - N

```

```

    if int(w ** 0.5) != w ** 0.5:

```

```

        continue

```

```

    sqrt_w = w ** 0.5

```

```

    break

```

```

p = t + sqrt_w

```

```

q = t - sqrt_w

```

```

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

```

```

d = pow(e, -1, int(phi))

```

```

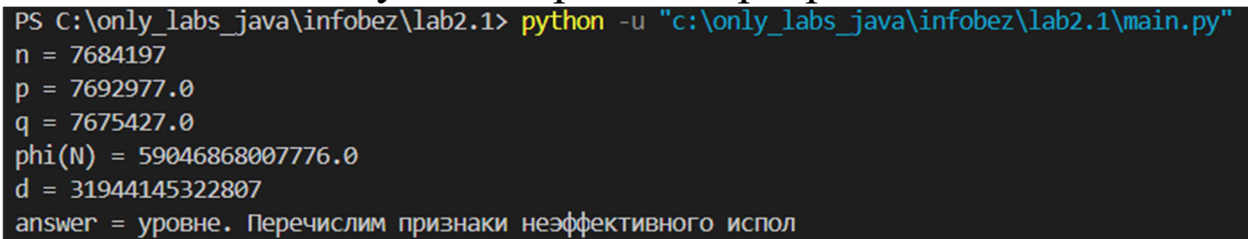
i = 0
for num in C.split("\n"):
    i += 1

    m = pow(int(num), d, N)
    part = m.to_bytes(4, byteorder='big').decode('cp1251')
    answer += part

print(f'n = {n}')
print(f'p = {p}')
print(f'q = {q}')
print(f'phi(N) = {phi}')
print(f'd = {d}')
print(f'answer = {answer}')

```

Результаты работы программы



```

PS C:\only_labs_java\infobez\lab2.1> python -u "c:\only_labs_java\infobez\lab2.1\main.py"
n = 7684197
p = 7692977.0
q = 7675427.0
phi(N) = 59046868007776.0
d = 31944145322807
answer = уровне. Перечислим признаки неэффективного испол

```

Выводы по работе

В ходе выполнения данной лабораторной работы я ознакомился с атакой на алгоритм шифрования RSA посредством метода Ферма.