Федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики»
Факультет программной инженерии и компьютерных технологий

Отчет по лабораторной работе № 1
“Атака на алгоритм шифрования RSA посредством метода Ферма”

по дисциплине Информационная безопасность

Вариант 10

Студент группы № P34151 Шипулин Павел Андреевич

Преподаватель Маркина Татьяна Анатольевна

Санкт-Петербург
2024

Цель работы

Изучить атаку на алгоритм шифрования RSA посредством метода Ферма.

Вариант задания

№ варианта	**Модуль, N**	**Экспонента, e**	**Блок зашифрованного текста, C**
10	77027476849549	2936957	18937689886043 6667195679130 53238895771820 6189192838687 48623327840257 47264919314001 42510070950746 16878504505970 22744978157662 23644842894223 71614018816334 24651499733229

Ход работы

1. Ознакомиться с теорией, изложенной в [3]. («Взлом алгоритма RSA при неудачном выборе параметров криптосистемы»).
2. Получить вариант у преподавателя
3. Используя разложение модуля на простые числа методом Ферма и полученные исходные данные, определить следующие показатели:
    a. Множители модуля (p и q).
    b. Значение функции Эйлера для данного модуля $\varphi(N)$.
    c. Обратное значение экспоненты по модулю $\varphi(N)$.

4. Дешифровать зашифрованный текст, исходный текст должен быть фразой на русском языке.
5. Результаты и промежуточные вычисления оформить в виде отчета.

Листинг программ

Ссылка на репозиторий:
https://github.com/PashcalE2/IS/tree/main/cryptography/second_part

Файл lab1.py

```python
import math


def lab1(N: int, e: int, C: str):
    print("Рассчет параметров")
    n = int(math.sqrt(N) + 1)
    print(f"n = int(sqrt({N})) + 1 = {n}")

    t = n
    while True:
        # Перебор t >= n
        t += 1
        sub = t ** 2 - N

        sqrt_sub = int(math.sqrt(sub))
        if sqrt_sub ** 2 == sub:
            break

    p = t + sqrt_sub
    print(f"p = {t} + {sqrt_sub} = {p}")

    q = t - sqrt_sub
    print(f"q = {t} - {sqrt_sub} = {q}")

    phi = round((p - 1) * (q - 1))
    print(f"ϕ(N) = {p - 1} * {q - 1} = {phi}")

    d = pow(e, -1, phi)
    print(f"d = {e}^(-1) mod {phi} = {d}")

    print("Дешифровка")
    result = ""
    for i, c in enumerate(C.split()):
        num_block = pow(int(c), d, N)
        print(f"num_block_{i} = {c}^{d} mod {N} = {num_block}")

        text_block = num_block.to_bytes(4, byteorder="big").decode("cp1251")
        print(f"text_block = {text_block}")

        result += text_block

    print(f"Результат = {result}")
```

```
if __name__ == "__main__":
    """
    Вариант 10
    """

    _N = 77027476849549
    _e = 2936957
    _C = """
    18937689886043
    6667195679130
    53238895771820
    6189192838687
    48623327840257
    47264919314001
    42510070950746
    16878504505970
    22744978157662
    23644842894223
    71614018816334
    24651499733229
    """

    lab1(_N, _e, _C)
```

Выполнение

Результат выполнения программы

Рассчет параметров

n = int(sqrt(77027476849549)) + 1 = 8776530

p = 8776535 + 9474 = 8786009

q = 8776535 - 9474 = 8767061

ф(N) = 8786008 * 8767060 = 77027459296480

d = 2936957^(-1) mod 77027459296480 = 8540915045653

Дешифровка

num_block_0 = 18937689886043^8540915045653 mod 77027476849549 = 4075692279

text_block = то ч

num_block_1 = 6667195679130^8540915045653 mod 77027476849549 = 3908168686

text_block = исло

num_block_2 = 53238895771820^8540915045653 mod 77027476849549 = 552592880

text_block = пер

num_block_3 = 6189192838687^8540915045653 mod 77027476849549 = 3856982242

text_block = едав

num_block_4 = 48623327840257^8540915045653 mod 77027476849549 = 3773164795

text_block = аемы

num_block_5 = 47264919314001^8540915045653 mod 77027476849549 = 4112578792

text_block = х ши

num_block_6 = 42510070950746^8540915045653 mod 77027476849549 = 4042189550

text_block = роко

num_block_7 = 16878504505970^8540915045653 mod 77027476849549 = 3806722528

text_block = веща

num_block_8 = 22744978157662^8540915045653 mod 77027476849549 = 4075154428

text_block = тель

num_block_9 = 23644842894223^8540915045653 mod 77027476849549 = 3992712480

text_block = ных

num_block_10 = 71614018816334^8540915045653 mod 77027476849549 = 4024494821

text_block = паке

num_block_11 = 24651499733229^8540915045653 mod 77027476849549 = 4075741791

text_block = тов_

Результат = то число передаваемых широковещательных пакетов_

Вывод

Узнал и проверил, что при неудачном значении N в методе шифрования RSA, можно легко найти секретный ключ d с помощью простого перебора чисел $t \geq n = \lceil \sqrt{N} + 1 \rceil$ и дальнейшей проверки на полный квадрат выражения $t^2 - N$.