МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
ФАКУЛЬТЕТ ЭЛЕКТРОННО-ИНФОРМАЦИОННЫХ СИСТЕМ
Кафедра интеллектуальных информационных технологий

Отчет
по дисциплине
«Современные методы защиты информации»
по лабораторной работе № 3
«Атака на алгоритм шифрования RSA»

Выполнил:
студент 4 курса
группы ИИ-22
Полиенко В.Э.
Проверила:
Хацкевич А.С.

Брест 2024

Цель: Изучить атаку на алгоритм RSA посредством метода Ферма

Вариант	Модуль, N	Экспонента, e	Блок зашифрованного текста, C
1	99595193774911	1908299	75790643190143 36869061035180 38422576553598 68899435645717 16193161920958 98487458352335 34167725433806 96613844267045 26583768908805 73052827576371 94695336463618 69092596694070

Ход работы

```
import java.math.BigInteger;

public class MethodFermat {
    private BigInteger N;
    private BigInteger e;

    MethodFermat(BigInteger N, BigInteger e) {
        this.N = N;
        this.e = e;
    }

    public BigInteger sqrt(BigInteger x) {
        if (x.compareTo(BigInteger.ZERO) < 0) {
            throw new ArithmeticException("Cannot compute square root of a
negative number");
        }
        if (x.equals(BigInteger.ZERO) || x.equals(BigInteger.ONE)) {
            return x;
        }
        BigInteger a = x;
        BigInteger b = x.shiftRight(1);

        while (b.compareTo(a) < 0) {
            a = b;
            b = x.divide(b).add(b).shiftRight(1); // (x / b + b) / 2
        }

        return a;
    }

    public void checkSqrt(BigInteger C) {
        BigInteger sqrtResult = sqrt(N);
        BigInteger square = sqrtResult.multiply(sqrtResult);

        System.out.println("Корень: " + sqrtResult + " Квадрат корня: " + square +
" N: " + N);

        if (square.equals(N)) {
            System.out.println(sqrtResult + " является точным квадратным корнем
числа " + N);
        } else {
            System.out.println(sqrtResult + " НЕ является точным квадратным корнем
числа " + N);
            BigInteger w1 = square.subtract(N).abs();

            while (!w1.equals(sqrt(w1).multiply(sqrt(w1)))) {
                sqrtResult = sqrtResult.add(BigInteger.ONE);
                square = sqrtResult.multiply(sqrtResult);
                w1 = square.subtract(N).abs();
```

```
                System.out.println(sqrtResult + " " + square + " - " + N + "
Разность: " + w1 + " квадрат: " + sqrt(w1).multiply(sqrt(w1)) + " Корень: " +
sqrt(w1));
            }

            BigInteger p = sqrtResult.add(sqrt(w1));
            BigInteger q = sqrtResult.subtract(sqrt(w1));
            BigInteger Composition =
q.subtract(BigInteger.ONE).multiply(p.subtract(BigInteger.ONE));
            System.out.println(sqrtResult + " " + "p: " + p + " q: " + q + " q*p =
" + Composition);

            BigInteger inverse = e.modInverse(Composition);
            System.out.println("Обратное к e: " + inverse);

            BigInteger decryptedMessage = C.modPow(inverse, N);
            System.out.println("Исходное сообщение: " + decryptedMessage);

        }
    }


}
```

Вывод программы:

```
Обработка C = 75790643190143
Корень: 9979739 Квадрат корня: 99595190508121 N: 99595193774911
9979739 НЕ является точным квадратным корнем числа 99595193774911
9979740 99595210467600 - 99595193774911 Разность: 16692689 квадрат: 16687225 Корень: 4085
9979741 99595230427081 - 99595193774911 Разность: 36652170 квадрат: 36650916 Корень: 6054
9979742 99595250386564 - 99595193774911 Разность: 56611653 квадрат: 56610576 Корень: 7524
9979743 99595270346049 - 99595193774911 Разность: 76571138 квадрат: 76562500 Корень: 8750
9979744 99595290305536 - 99595193774911 Разность: 96530625 квадрат: 96530625 Корень: 9825
9979744 p: 9989569 q: 9969919 q*p = 99595173815424
Обратное к e: 65973656360291
Исходное сообщение: 3303800608
Обработка C = 36869061035180
```


```
Обработка C = 75790643190143
Корень: 9979739 Квадрат корня: 99595190508121 N: 99595193774911
9979739 НЕ является точным квадратным корнем числа 99595193774911
9979740 99595210467600 - 99595193774911 Разность: 16692689 квадрат: 16687225 Корень: 4085
9979741 99595230427081 - 99595193774911 Разность: 36652170 квадрат: 36650916 Корень: 6054
9979742 99595250386564 - 99595193774911 Разность: 56611653 квадрат: 56610576 Корень: 7524
9979743 99595270346049 - 99595193774911 Разность: 76571138 квадрат: 76562500 Корень: 8750
9979744 99595290305536 - 99595193774911 Разность: 96530625 квадрат: 96530625 Корень: 9825
9979744 p: 9989569 q: 9969919 q*p = 99595173815424
Обратное к e: 65973656360291
Исходное сообщение: 3303800608
Обработка C = 36869061035180
```


Вывод: Изучил атаку на алгоритм RSA посредством метода Ферма