ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет программной инженерии и компьютерной техники

Дисциплина «Информационная безопасность»

Лабораторная работа №2.2

«Атака на алгоритм шифрования RSA методом повторного шифрования»

Вариант: 4

Учебно-методическое пособие: Криптографические системы с секретным и открытым ключом: учебное пособие. / А.А. Ожиганов; УНИВЕРСИТЕТ ИТМО. — Санкт-Петербург, 2015

Автор: Калинин Даниил Дмитриевич

Группа: P34141

Преподаватель: Маркина Татьяна Анатольевна

г. Санкт-Петербург

2024

Содержание

Цель работы

Изучить атаку на алгоритм шифрования RSA посредством повторного шифрования.

Порядок выполнения работы

- Ознакомьтесь с теорией в [3], рассмотренной в подразделе («Атака повторным шифрованием»);
- Получите вариант задания у преподавателя;
- По полученным исходным данным, используя метод перешифрования, определите порядок числа *e* в конечном поле $Z_{\varphi(N)}$
- Используя значение порядка экспоненты, получите исходный текст методом перешифрования;
- Результаты и промежуточные вычисления оформите в виде отчета.

Вариант

Вариант	**Модуль, *N***	**Экспонента, *e***	**Блок зашифрованного текста, *C***
4	489740760623	892627	237434928568 89382477865 257542914775 153947910848 219678068406 166466311168 49516725114 55375254449 370796045103 322927050068 196366079994 39243100230 299525662956

Выполнение работы

Код

```
def re_encryption_method(N, e, C):
    """Определение порядка числа e"""
    print("-- Метод повторного шифрования --")

    # Выбираем первую часть, на которой будем искать степерь повтороного
кодирования
    raw_parts = C.split("\n")
    y = 0
    for i in range(len(raw_parts)):
        if raw_parts[i].strip() != "":
            y = int(raw_parts[i].strip())
            break

    # Выполняем повторное шифрование
    y_i = y
    i = 1
    while True:
        y_i = pow(y_i, e, N)
        i+=1

        if (y_i == y):
            break

    print(f'y_{i} = y = {y_i}')
    return i - 2 #Порядок числа e


def decode(N, e, C, m):
    """Декодирует полученное сообщение в текст"""

    print("-- Дешифрование сообщения --")

    # Раздаляем закодированное сообщение на части и подготавливаем их
    raw_parts = C.split("\n")

    parts = []
    for i in range(len(raw_parts)):
        if raw_parts[i].strip() != "":
            parts.append(int(raw_parts[i].strip()))

    # Декодируем каждую часть
    original_message = ""
    for part in parts:
        int_decoded_part = pow(part, pow(e, m), N)
```

```
        decoded_part = int_decoded_part.to_bytes(4,
byteorder='big').decode('cp1251')
        original_message += decoded_part
        print(f'Декодирована часть {part} -----> y_{m+1} = {int_decoded_part}
-----> {decoded_part}')

    return original_message

if __name__ == '__main__':
    # Описание варианта
    N = 489740760623
    e = 892627
    C = """
        237434928568
        89382477865
        257542914775
        153947910848
        219678068406
        166466311168
        49516725114
        55375254449
        370796045103
        322927050068
        196366079994
        39243100230
        299525662956
        """

    print("-- Исходные данные --")
    print(f'N = {N}')
    print(f'e = {e}')
    print(f'C = \"{C}\"')
    print()

    # Определяем порядок числа e
    m = re_encryption_method(N, e, C)
    print(f'x = y_{m + 1}')
    print(f'Порядок числа e = {m}')
    print()

    # Декодируем сообщение
    original_message = decode(N, e, C, m)
    print(f'\nОригинальное сообщение - \"{original_message}\"')
```

Результаты работы программы

```
C:\Python310\python.exe "D:\Учеба\4 курс\7 семестр\(ИБ) Информационная безопасность\information-security-labs\lab_2_2\lab_2.2.py"
-- Исходные данные --
N = 489740760623
e = 892627
C = "
        237434928568
        89382477865
        257542914775
        153947910848
        219678068406
        166466311168
        49516725114
        55375254449
        370796045103
        322927050068
        196366079994
        39243100230
        299525662956
        "

-- Метод повторного шифрования --
y_95461 = y = 237434928568
x = y_95460
Порядок числа e = 95459

-- Дешифрование сообщения --
Декодирована часть 237434928568 -----> y_95460 = 4025414123 -----> посл
Декодирована часть 89382477865 -----> y_95460 = 3856985826 -----> едов
Декодирована часть 257542914775 -----> y_95460 = 3774014955 -----> ател
Декодирована часть 153947910848 -----> y_95460 = 4243451633 -----> ьнос
Декодирована часть 219678068406 -----> y_95460 = 4076609770 -----> ть к
Декодирована часть 166466311168 -----> y_95460 = 3773100256 -----> адра
Декодирована часть 49516725114 -----> y_95460 = 690020896 -----> ) в
Декодирована часть 55375254449 -----> y_95460 = 1165256805 -----> Ethe
Декодирована часть 370796045103 -----> y_95460 = 1919837556 -----> rnet
Декодирована часть 322927050068 -----> y_95460 = 552135656 ----->  или
Декодирована часть 196366079994 -----> y_95460 = 552214766 ----->  к о
Декодирована часть 39243100230 -----> y_95460 = 4176011754 -----> шибк
Декодирована часть 299525662956 -----> y_95460 = 3773571167 -----> ам _

Оригинальное сообщение - "последовательность кадра) в Ethernet или к ошибкам _"

Process finished with exit code 0
```

Вывод

В ходе лабораторной работы была совершена атака на алгоритм шифрования RSA посредством повторного шифрования, в следствии чего было декодировано исходное сообщение. В процессе выполнения был изучен алгоритм совершения атаки на алгоритм шифрования RSA посредством повторного шифрования.