МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский Авиационный Институт»
(Национальный Исследовательский Университет)

Институт: №8 «Информационные технологии и прикладная математика»
Кафедра: 806 «Вычислительная математика и программирование»

Лабораторная работа № 4
по курсу «Криптография»

Группа: М8О-308Б-21

Студент(ка): Т. Ж. Караев

Преподаватель: А. В. Борисов

Оценка:

Дата: 23.05.2025

Москва, 2025

ОГЛАВЛЕНИЕ

1 Тема

Эллиптические кривые.

2 Задание

Подобрать такую эллиптическую кривую, порядок точки которой полным перебором находится за 10 минут на ПК. Упомянуть в отчёте результаты замеров работы программы, характеристики вычислителя. Также указать какие алгоритмы и/или теоремы существуют для облегчения и ускорения решения задачи полного перебора. Рассмотреть для случая конечного простого поля $\mathbb{Z}_p$.

3 Теория

Мы рассматриваем эллиптическую кривую над конечным полем Z_p, где p — простое число. Каноническая форма кривой:

$$E: y = x^3 + ax + b \; mod \; p$$

Порядком точки p называется наименьшее положительное число, такое что $np = 0$. Чтобы его найти, необходимо выполнять последовательное сложение $p, 2p, 3p \; ... np$ до тех пор, пока не получится нейтральный элемент 0.

4 Ход лабораторной работы

В рамках лабораторной работы была поставлена задача подобрать такие параметры эллиптической кривой, при которых порядок одной из точек кривой можно было бы найти полным перебором за время порядка 10 минут на обычном персональном компьютере.

В процессе выполнения работы стало ясно, что задача не столько на вычисление порядка точки, сколько на демонстрацию сложности подбора параметров для кривых, при которых перебор занимает какое-то определённое, но довольно значительное время — 10 минут.

Я пытался найти эти значения, перебирая простые числа для параметра p и большие значения параметров a, b. Я даже пробовал случайный подбор некоторых параметров, и тем не менее значения не были найдены.

В то же время другие люди каким-то чудесным образом находили эти параметры просто из головы и сразу использовали их в коде. Я предполагаю, что существуют математические методы для сужения диапазона поиска или примерного прогноза порядка точки для тройки параметров, однако в конечном счёте, чтобы добиться долгого вычисления на конкретной машине (например, на среднестатистическом компьютере обычного человека), всё равно придётся перебирать эти параметры и проверять, сколько времени занимает поиск порядка точки.

Также существуют некоторые алгоритмы, облегчающие вычисление порядка точки. Например, алгоритм Шуфа является первым детерминированным полиномиальным алгоритмом для этой задачи. Алгоритм использует теорему Хассе, которая ограничивает возможное

количество точек на кривой, и применяет китайскую теорему об остатках для восстановления точного значения. Он позволяет заранее определить порядок группы точек на кривой, что существенно сокращает объём перебора при поиске точки с нужным порядком.

Или алгоритм Полларда. Он является стохастическим методом для решения задачи дискретного логарифма в группах, включая группы точек эллиптических кривых. Он основан на идее случайного блуждания и использует эффект "дня рождения" для обнаружения коллизий, что позволяет находить порядок точки значительно быстрее, чем при полном переборе. Этот алгоритм также позволяет ускорить процесс нахождения порядка точки.

Код программы для перебора параметров и вычисления порядка точки:

```
import random
import time

import sympy


def is_valid_curve(a, b, p):
    return (4 * pow(a, 3, p) + 27 * pow(b, 2, p)) % p != 0


def find_point_on_curve(a, b, p):
    while True:
        x = random.randint(0, p - 1)
        rhs = (pow(x, 3, p) + a * x + b) % p
        if sympy.legendre_symbol(rhs, p) == 1:
            for y in range(p):
                if (y * y) % p == rhs:
                    return (x, y)


def point_add(P, Q, a, p):
    if P is None:
        return Q
    if Q is None:
        return P

    x1, y1 = P
    x2, y2 = Q

    if x1 == x2 and y1 != y2:
```

```
        return None

    if P == Q:
        m = (3 * x1 * x1 + a) * sympy.mod_inverse(2 * y1, p)
    else:
        m = (y2 - y1) * sympy.mod_inverse(x2 - x1, p)

    m %= p

    x3 = (m * m - x1 - x2) % p
    y3 = (m * (x1 - x3) - y1) % p
    return (x3, y3)


def scalar_mult(k, P, a, p):
    R = None
    Q = P
    while k:
        if k & 1:
            R = point_add(R, Q, a, p)
        Q = point_add(Q, Q, a, p)
        k >>= 1
    return R


def find_order(P, a, p):
    Q = P
    order = 1

    while Q is not None:
        Q = point_add(Q, P, a, p)
        order += 1

    return order


def main():

    for p in sympy.primerange(50_000, 1000_000_000):
        for a in range(100_000, 1000_000):
            for b in range(100_000, 1000_000):
                P = find_point_on_curve(a, b, p)

                print(f'p = {p}, a = {a}, b = {b}, точка P = {P}')

                start = time.time()
                order = find_order(P, a, p)
                duration = time.time() - start

                print(f'Порядок точки: {order}, время: {duration:.2f}
сек')

                if 600 <= duration <= 700:
                    print(
                        'Параметры для эллиптической кривой: \n'
                        'p = {p}, a = {a}, b = {b} \n'
                        'Точка: {P} \n'
                        'Порядок: {order}'
```

```
                    )
                    break

if __name__ == '__main__':
    main()
```

Вывод программы:

```
sempaitakoo@desktoptakoo:~/cryptography_labs/lab4/src$ uv run main.py
p = 50021, a = 100000, b = 100000, точка P = (4376, 9181)
Порядок точки: 50431, время: 0.12 сек
p = 50021, a = 100000, b = 100001, точка P = (48675, 9798)
Порядок точки: 24953, время: 0.06 сек
p = 50021, a = 100000, b = 100002, точка P = (44906, 23109)
Порядок точки: 1777, время: 0.01 сек
p = 50021, a = 100000, b = 100003, точка P = (32793, 2518)
Порядок точки: 10002, время: 0.03 сек
p = 50021, a = 100000, b = 100004, точка P = (21504, 5477)
Порядок точки: 50359, время: 0.12 сек
p = 50021, a = 100000, b = 100005, точка P = (29084, 179)

...
```

5 Выводы

В ходе выполнения работы было выяснено, что задача подбора параметров эллиптической кривой, при которых порядок точки можно найти перебор за заданное время, на практике оказывается крайне трудоёмкой задачей. Было выяснено, что защита, основанная на эллиптических кривых, достаточно надёжна и опирается на высокую вычислительную сложность.

6 Список используемой литературы

- https://en.wikipedia.org/wiki/Schoof%27s_algorithm
- https://en.wikipedia.org/wiki/Pollard%27s_rho_algorithm