Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ

Кафедра интеллектуальных информационных технологий

Лабораторная работа № 4

“Вариант 3”

Отчет

Выполнил:

студент гр. 221703

Веркович Е.В.

Проверил: Крищенович В.А.

Минск 2024

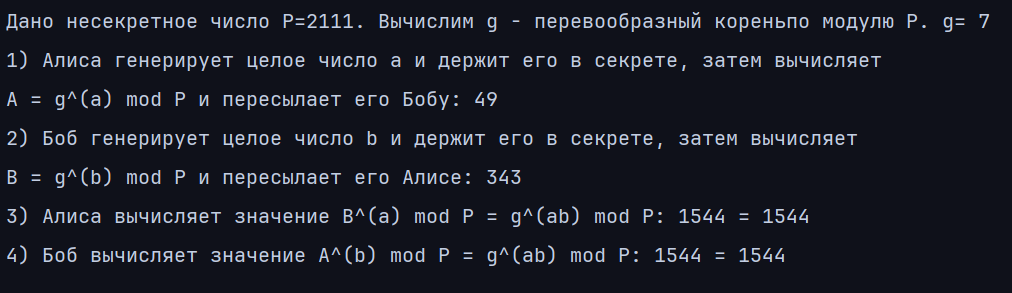
**Задание**

Для заданного простого P (в соответствии с вариантом) найти g – примитивный элемент конечного поля GF(P) и выполнить генерацию общего секрета. Для нахождения g воспользуйтесь методом перебора по возрастанию, возведения в степень по модулю P и проверки того факта, что все степени принимают значения от 0 до P - 1.

public class DiffiHelmanProtocol {  
 private final long P = 2111;  
 private long g;  
  
 public DiffiHelmanProtocol() {  
 this.g = (int) generateGFP(); *// Найдем первообразный корень g при инициализации* }  
  
 public long generateSecret(long x){  
 return powMod(g,x,P);  
 };  
 public long generateSecret(long base, long x){  
 return powMod(base,x,P);  
 };  
  
 *// Функция для возведения числа в степень по модулю P* private long powMod(long base, long exp, long mod) {  
 long result = 1;  
 while (exp > 0) {  
 if (exp % 2 == 1) {  
 result = (result \* base) % mod;  
 }  
 base = (base \* base) % mod;  
 exp /= 2;  
 }  
 return result;  
 }  
  
 *// Проверка, является ли число g примитивным корнем по модулю P* private boolean isPrimitiveRoot(int g) {  
 *// Разложение P-1 = 2110 на простые множители: 2, 5, 211* int[] factors = {2, 5, 211};  
  
 *// Проверим для всех делителей P-1* for (int factor : factors) {  
 if (powMod(g, (P - 1) / factor, P) == 1) {  
 return false;  
 }  
 }  
 return true;  
 }  
  
 *// Метод для нахождения примитивного корня g по модулю P* private long generateGFP() {  
 for (int g = 2; g < P; g++) {  
 if (isPrimitiveRoot(g)) {  
 return g;  
 }  
 }  
 return -1; *// Если не найдено (что маловероятно для простого P)* }

**2) Описание шагов, выполняемых участниками протокола – Алисой и**

**Бобом для вычисления общего секрета.**



**Выводы:**

Выводы по протоколу Диффи-Хеллмана

1. Модель атакующего и оценки длины ключа:

- В контексте протокола Диффи-Хеллмана основная угроза исходит от атакующего, который имеет возможность перехватывать публичные ключи сторон, участвующих в обмене (Man-in-the-Middle или пассивное прослушивание). Атакующий не может напрямую вычислить секретный ключ, так как задача сводится к вычислению дискретного логарифма, что является сложной проблемой при достаточно большом простом числе P и правильном выборе примитивного корня g.

- Длина ключа определяется длиной числа P. Для устойчивости к современным атакам длина P обычно должна составлять не менее 2048 бит (что значительно больше числа 2111).

2. Возможные угрозы протоколу и предложения по защите:

-Атака "Человек посередине" (Man-in-the-Middle): Это одна из самых известных угроз для Диффи-Хеллмана. Атакующий может перехватывать публичные ключи сторон, заменяя их своими, и тем самым создать два отдельных сеанса обмена ключами, контролируя каждый из них. Без аутентификации сторон атакующий может незаметно расшифровать и перехватывать информацию.

- Защита: Введение механизма аутентификации, например, использование цифровых подписей или сертификатов для проверки подлинности сторон перед обменом ключами.

- Пассивное прослушивание: Атакующий может просто наблюдать за передачей публичных ключей, надеясь решить задачу дискретного логарифма и восстановить общий секретный ключ.

- Защита: Использование более крупных значений P и g, так как это увеличивает сложность вычисления дискретного логарифма. Как упоминалось ранее, стандартная длина для P должна быть не менее 2048 бит.