МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»**

Кафедра компьютерной алгебры и теории чисел

Разработка программного обеспечения, реализующего криптозащиту данных с использованием нескольких методов

АВТОРЕФЕРАТ МАГИСТЕРСКОЙ РАБОТЫ

студента 2 курса 227 группы

направление 02.04.01 — Математика и компьютерные науки

механико-математического факультета

Дадаева Арсена Исламовича

Научный руководитель
доцент, к.ф.-м.н. доцент В.В. Кривобок

Зав. кафедрой
зав. каф., к.ф.-м.н., доцент А.М. Водолазов

Саратов 2023

ВВЕДЕНИЕ

Тема. Тема данной магистерской работы - "Разработка программного обеспечения, реализующего криптозащиту данных с использованием нескольких методов".

Актуальность темы исследования. В современном мире, где данные играют все большую роль, защита конфиденциальности, целостности и доступности становится критически важной задачей. Криптография - это наука, которая занимается защитой данных от несанкционированного доступа, и представляет собой одну из основных технологий в области информационной безопасности.

Выбор темы данной работы обусловлен потребностью в изучении основных методов и алгоритмов криптозащиты данных, поиске способов и решений для создания надежных защитных систем, а также в исследовании существующих программных решений в данной области.

Цели и задачи. Цель данной работы - изучение базовых математических и алгоритмических основ криптографии и разработка программного обеспечения реализующего криптозащиту. Для достижения данной цели были поставлены следующие задачи:

- Изучение основных математических принципов криптографии и алгоритмов шифрования.
- Исследование наиболее распространенных методов криптозащиты данных.
- Разработка прототипа системы криптозащиты данных на основе существующего программного обеспечения.

Содержание работы. Работа состоит из 5 разделов. Первый раздел посвящен необходимым и основным понятиям из криптографии. Во втором разделе разбираются математические основы криптографии,в основном рассматриваются способы решения задачи дискретного логарифмирования. Третий раздел посвящен описанию симметричного блочного шифра AES. В нем подробно рассмотрены все основные шаги алгоритма и преобразования, которые выполняются в процессе шифрования. Четвертый раздел содержит краткую информацию об шифровании с открытым ключом, а также описывает алгоритм RSA, в котором подробно рассмотрены все этапы выполнения

данного алгоритма. В пятом разделе представлена программная реализация гибридной криптосистемы, которая использует алгоритмы AES и RSA для шифрования данных. Реализация выполнена на языке программирования Dart с использованием фреймворка Flutter.

Апробация. Доклад по теме исследования был представлен на кафедре компьютерной алгебры и теории чисел.

Основное содержание работы. Для начала необходимо ознакомиться с основными понятиями криптографии и изучить методы решения задачи дискретного логарифмирования.

Криптография - область науки, занимающаяся методами защиты информации от несанкционированного доступа. В ее основе лежат различные определения и понятия, например: криптосистемы, шифрование, расшифровка, ключи, алгоритмы, аутентификация и др. Путем сокрытия информации в зашифрованном виде и использования специальных методов, криптография обеспечивает конфиденциальность передачи данных и сохранение их целостности, а также предотвращение возможности подделки и изменения информации со стороны неавторизованных лиц.

Ключи шифрования – это специальные символьные последовательности, которые используются для защиты информации при ее передаче. Ключ шифрования может быть открытым или секретным, в зависимости от того, какая информация с ним связана. Шифрование с открытым ключом использует открытый ключ для кодирования информации, а шифрование с секретным ключом требует знания секретного ключа для декодирования информации.

Алгоритмы шифрования – это процедуры, которые задают правила шифрования и дешифрования информации. Алгоритмы шифрования могут быть симметричными (требуют наличия одинаковых секретных ключей для шифрования и дешифрования) или асимметричными (требуют наличия открытого и секретного ключей для шифрования и дешифрования информации) [1].

Алгоритмы шифрования данных включают в себя множество математических инструментов, базирующихся на абстрактной алгебре, теории чисел, линейной алгебре, а также других областях, таких как теория поля, комбина-

торика, теория вероятностей и квантовые алгоритмы. Все эти инструменты относятся к дисциплине дискретной математики [2].

Криптография использует математические понятия, такие как множества, отношения и операции над множествами, эквивалентность множеств, мощность множеств, функции, целые и действительные числа, простые числа, а также алгоритмы для нахождения НОД, НОК чисел и разложения многочленов на множители. Для работы с матрицами используются алгоритмы и операции.

Некоторые основные алгебраические структуры, используемые в криптографии, включают группоиды, моноиды, полугруппы, группы, частичные порядки, кольца и поля.

Одним из главных вызовов, с которым сталкиваются в криптологии, является проблема разложения больших целых чисел на множители.

Алгоритм RSA является примером тесной связи между криптографией и дискретной математикой. Данный алгоритм позволяет использовать систему шифрования с открытым ключом, при которой шифрование может быть выполнено каждым, а расшифровка невозможна без специального закрытого ключа. Эти ключи в совокупности составляют открытый ключ.

Один из чисел, используемых в данной системе, является произведением двух простых чисел p и q, а другое число, обозначаемое как E, должно быть относительно простым с произведением $(p-1)(q-1)$ в силу технических ограничений. Другими словами, НОД между E и $(p-1)(q-1)$ должен равняться 1.

Основой системы RSA является то, что проще найти произведение двух больших простых чисел, чем выполнить обратную операцию. Также существуют другие фундаментальные теоретико-числовые процедуры, обладающие свойствами однонаправленности или функции с замком, такие как возведение в степень с большим конечным полем.

В поле вещественных чисел нахождение степени g^x с заданной точностью не является гораздо более простой операцией, чем нахождение обратной операции $\log_g x$ с заданной точностью. Тем не менее, для конечной группы $(Z/nZ)^*$ или F_q^*, в которой умножение является групповой операцией, суще-

ствует метод повторного возведения в квадрат, который позволяет вычислить g^x для больших x за полиномиальное по $\log x$ время.

Предположим, что задан элемент y, который может быть представлен как g^x (считается, что основание g задано), как найти степень элемента g, в которой он равен y, т.е. как вычислить $\log_g y$? Этот вопрос называется «задачей дискретного логарифмирования».

Определение 2.5. Пусть $(G, \cdot)$ - конечная циклическая группа порядка m, g - образующий элемент G и $y \in G$. Дискретным логарифмом (показателем) элемента y группы G по основанию g называется число $x \in \{0, 1, \ldots, m-1\}$, являющееся решением уравнения:

$$g^x = y$$

Дискретный логарифм элемента y по основанию g обозначается $\log_g y$. Если групповая операция задана в аддитивной форме, то данное уравнение записывается в виде $xg = y$ [3].

Для криптографических протоколов наиболее важными являются следующие циклические группы:

1. Мультипликативная группа $\mathbb{Z}_p^*$ кольца вычетов по простому модулю p.

2. Мультипликативная группа $GF(q)^*$ конечного поля из $q = p^n$ элементов.

3. Циклическая подгруппа точек эллиптической кривой $E(GF(q))$ над полем $GF(q)$.

Далее рассмотрим алгоритмы, предназначенные для решения задачи дискретного логарифмирования, которая состоит в нахождении степени элемента b, в которой он равен заданному элементу y.

В произвольной мультипликативной группе. Статья Buchmann J., Jacobson M. J., Teske E. «On Some Computational Problems in Finite Abelian Groups» [7] посвящена разрешимости и решению задачи дискретного логарифмирования в произвольной конечной абелевой группе. В данном алгоритме используется таблица, представляющая собой коллекцию $O(\sqrt{|\langle g \rangle|})$ пар элементов, и для ее выполнения требуется $O(\sqrt{|\langle g \rangle|})$ умножений. Несмотря на это, данный алгоритм является медленным и неэффективным в практиче-

ском применении. Для решения задачи дискретного логарифмирования для конкретных групп существуют свои более эффективные алгоритмы.

В кольце вычетов по простому модулю. Рассмотрим уравнение

$$a^x \equiv b \mod p, \tag{1}$$

где p - простое, b не делится на p. Если a является образующим элементом группы $\mathbb{Z}/p\mathbb{Z}$, то уравнение (2.1) имеет решение при любых b. Такие числа a называются ещё первообразными корнями, и их количество равно $\phi(p-1)$, где ϕ — функция Эйлера. Решение уравнения (2.1) можно находить по формуле:

$$x = \sum_{i=1}^{p-1} (1 - a^i)^{-1} b^i \mod p.$$

Однако, сложность вычисления по этой формуле хуже, чем сложность перебора.

Следующий алгоритм имеет сложность $O(p^{\frac{1}{2}} \log p)$

Алгоритм

1. Присвоить $H := [p^{\frac{1}{2}}] + 1$
2. Найти $c \equiv a^H \mod p$
3. Составить таблицу значений $c^u \mod p, 1 \leq u \leq H$, и упорядочить её.
4. Составить таблицу значений $b \cdot a^v \mod p, 0 \leq v \leq H$, и упорядочить её.
5. Найти совпавшие элементы из первой и второй таблиц. Для них $c^u = b \cdot a^v \mod p$, откуда $a^{Hu-v} \equiv b \mod p$.
6. Выдать $x \equiv Hu - v \mod p - 1$.

Конец алгоритма

Алгоритмы с экспоненциальной сложностью.

1. Алгоритм Шенкса (алгоритм больших и малых шагов, baby-step giant-step)
2. Алгоритм Полига-Хеллмана — работает, если известно разложение числа $p-1 = \prod_{i=1}^{s} q_i^{\alpha_i}$ на простые множители. Сложность: $O(\sum_{i=0}^{s} \alpha_i (\log p + q_i))$. Если множители, на которые раскладывается $p-1$, достаточно маленькие, то алгоритм очень эффективен.
3. ρ-метод Полларда имеет эвристическую оценку сложности $O(p^{\frac{1}{2}})$.

Субэкспоненциальные алгоритмы. Данные алгоритмы имеют сложность $O(\exp(c(\log p \log\log p)^d))$ арифметических операций, где c и $0 \leq d < 1$ — некоторые константы. Эффективность алгоритма во многом зависит от близости c к 1 и d — к 0 [4].

1. Алгоритм Адлемана был представлен в 1979 году как первый субэкспоненциальный алгоритм решения задачи дискретного логарифмирования. Несмотря на это, на практике он не является достаточно эффективным. В данном алгоритме значение d равно $\frac{1}{2}$.
2. Алгоритм COS был предложен в 1986 году математиками Копперсмитом (Don Coppersmith), Одлыжко (Andrew Odlyzko) и Шреппелем (Richard Schroeppel). В этом алгоритме константа $c = 1$, $d = \frac{1}{2}$. В 1991 году с помощью Этого метода было проведено логарифмирование по модулю p 10^{58}. В 1997 году Вебер провел дискретное логарифмирование по модулю p 10^{85} с помощью некоторой версии данного алгоритма. Экспериментально показано, что при $p \leq 10^{90}$ алгоритм COS лучше решета числового поля.
3. Применение решета числового поля к задаче дискретного логарифмирования было осуществлено в период, идущий за периодом применения данного метода к факторизации чисел. Первоначальные концепции по данному вопросу появились в 1990-х годах. Алгоритм, предложенный в 1993 году Д. Гордоном, имел эвристическую сложность, измеряемую как $O(\exp 3^{3/2}(\log p \log\log p)^{\frac{1}{3}})$, однако его практическая применимость оказалась низкой. В последующем было разработано ряд улучшений данного алгоритма. Было доказано, что для значений p больших или равных 10^{100}, использование решета числового поля наиболее эффективно по сравнению с COS. Современные рекорды в задаче дискретного логарифмирования были установлены именно с использованием данного метода.

В произвольном конечном поле. Задача рассматривается в поле GF(q), где $q = p^n$, p — простое.

1. Алгоритм исчисления индексов (index-calculus) эффективен, если p невелико. В этом случае он имеет эвристическую оценку сложности $O(\exp c(\log p \log\log p)^{\frac{1}{2}})$.

2. Алгоритм Эль-Гамаля, появившийся в 1985 году, применим при $n = 2$ и имеет сложность $O(\exp c(\log p \log\log p)^{\frac{1}{2}})$ арифметических операций.
3. Алгоритм Копперсмита для решения задачи дискретного логарифмирования в конечном поле характеристики 2 стал первым алгоритмом субэкспоненциальной сложности с константой $d = \frac{1}{3}$ в оценках сложности. Он был представлен в 1984 году, что делает его более ранним методом, чем алгоритм решета числового поля.

В группе точек на эллиптической кривой.

Определение. Пусть K - поле характеристики, отличной от 2, 3, и $x^3 + ax + b$ (где $a, b \in K$) - кубический многочлен без кратных корней. Эллептическая кривая над K - это множество точек (x, y), $x, y \in K$,удовлетворяющих уравнению

$$y^2 = x^3 + ax + b, \tag{2}$$

вместе с единственным элементом, обозначаемым O и называемым «точка бесконечности» [7].

Если K - поле характеристики 2 , то эллиптическая кривая над K - это множество точек, удовлетворяющих уравнению либо типа

$$y^2 + cy = x^3 + ax + b \tag{3}$$

либо типа

$$^2+xy = x^3 + ax^2 + b \tag{4}$$

(здесь кубические многочлены в правых частях могут иметь кратные корни), вместе с «точкой в бесконечности» O.

Если K - поле характеристики 3 , то эллиптическая кривая над K - это множество точек, удовлетворяющих уравнению

$$y^2 = x^3 + ax^2 + bx + c \tag{5}$$

(где кубический многочлен справа не имеет кратных корней), вместе с «точкой в бесконечности» O.

Далее будут приведены краткие описания алгоритмов AES и RSA.

Алгоритм шифрования AES (Advanced Encryption Standard) является одним из наиболее распространенных симметричных блочных шифров. Его выбрали в качестве стандарта для защиты государственной информации в США, а также для широкого использования в различных приложениях.

Основные характеристики AES. AES работает с блоками данных размером 128 бит и производит шифрование на 128, 192 или 256 бит в зависимости от выбранного ключа шифрования. Все операции AES выполняются в поле Галуа, что обеспечивает высокую стойкость к атакам.

Алгоритм состоит из четырех основных операций: SubBytes, ShiftRows, MixColumns, и AddRoundKey. Они выполняются на каждом шаге циклически, пока не будет обработан весь блок данных.

Описание шагов AES

Шаг 1. AddRoundKey

На этом шаге ключ шифрования складывается по модулю 2 с блоком данных. В результате каждый байт блока данных смешивается с соответствующим байтом ключа шифрования.

Шаг 2. SubBytes

SubBytes заменяет каждый байт блока данных на новый байт, полученный из заранее определенной таблицы замен (S-блоков). Это обеспечивает дополнительную защиту против методов криптоанализа, которые основываются на частотном анализе данных.

Шаг 3. ShiftRows

ShiftRows выполняет циклический сдвиг строк блока данных на определенный количество байтов влево. Например, первая строка сдвигается на 1 байт, вторая – на 2 байта, а третья – на 3 байта. Это необходимо для того, чтобы предотвратить линейную зависимость между блоками данных.

Шаг 4. MixColumns

MixColumns выполняет линейное преобразование каждого столбца блока данных. Для этого применяется специальная матрица, которая умножается на каждый столбец.

Шаг 5. Повторение шагов

Выполняются также шаги 1-4 несколько раз, чтобы обеспечить лучшую защиту данных.

Ключ шифрования AES может быть выбран из 128, 192 или 256 бит. Чтобы обеспечить высокую криптостойкость, ключ должен создаваться случайным образом. Однако в реальной жизни это не всегда возможно, поэтому часто применяются методы, которые позволяют получить ключ из пароля пользователя.

Преимущества AES

- Высокая криптостойкость и устойчивость к различным атакам.

- Простота реализации и эффективность работы на устройствах с ограниченными ресурсами.

- Широкое распространение и использование в различных сферах, в том числе банковской сфере, Интернете, мобильных приложениях и др.

Недостатки AES

- Нет абсолютной гарантии безопасности, так как метод криптоанализа может появиться в будущем и нарушить защиту данных.

- Выполнение нескольких операций в обработке блока данных может привести к увеличению времени шифрования в сравнении с другими алгоритмами.

Алгоритм RSA (Rivest, Shamir, Adleman) является наиболее известным алгоритмом шифрования с открытым ключом. Он используется для защиты информации во многих сферах, включая электронную почту, банковские операции и коммерческие транзакции.

RSA работает на основе математических принципов, связанных с разложением больших чисел на простые множители. Он использует два ключа: открытый ключ для шифрования и закрытый ключ для расшифровки. Длина ключа обычно составляет 1024 или 2048 бит.

Важным преимуществом RSA является возможность подписывать и проверять электронные документы. Это обеспечивает аутентификацию и целостность информации.

Описание алгоритма RSA

Шаг 1. Генерация ключей

Первый шаг заключается в генерации пары ключей: открытого и закрытого. Для этого выбираются два случайных простых числа, которые перемножаются. Они образуют модуль N, который будет использоваться при шифровании данных. Затем выбирается число e, которое является открытым ключом, и такое, что число $(e,\ \ (p-1)(q-1))$ равно единице. Закрытый ключ d вычисляется как обратный элемент к числу e по модулю $(p-1)(q-1)$.

Шаг 2. Шифрование

На этом шаге исходное сообщение M преобразуется в целое число m, которое меньше модуля N. Затем выполнение преобразования:

$$C = m^e \bmod N$$

Результатом является зашифрованное сообщение C.

Шаг 3. Расшифрование

Зашифрованное сообщение C представляет собой целое число, которое должно быть расшифровано закрытым ключом d. Для этого выполняется преобразование:

$$M = C^d \bmod N$$

Результатом является исходное сообщение M.

Преимущества и недостатки RSA

Преимущества RSA:

- Широкое распространение и использование в различных сферах.

- Высокая степень криптографической защиты.

- Возможность подписывать и проверять электронные документы.

Недостатки RSA:

- Вычислительная сложность для больших чисел может привести к увеличению времени выполнения операций шифрования и расшифрования.

- Более длинный ключ требует большей вычислительной мощности, что может быть проблематично для некоторых устройств с ограниченными ресурсами.

ЗАКЛЮЧЕНИЕ

В ходе проведенного исследования было выявлено, что криптография - это область науки, посвященная защите конфиденциальных данных от неправомерного доступа, и является ключевой технологией в области информационной безопасности. В данном исследовании было сфокусировано внимание на программных средствах криптозащиты данных, которые включают в себя методы шифрования, аутентификации, цифровой подписи, хеширования, контроля доступа и т.д. Были изучены основные математические принципы криптографии и алгоритмы шифрования, а также рассмотрены наиболее распространенные методы криптозащиты данных.

После выполнения проекта, связанного с разработкой алгоритмов AES и RSA на языке Dart, а также создания графического интерфейса, достигнуты поставленные цели. В рамках исследования были созданы необходимые функции и алгоритмы, позволяющие безопасно обрабатывать и защищать цифровые данные. Проверка приложения подтвердила его стабильность, безопасность и готовность к использованию.

Однако, следует отметить, что дальнейшее развитие данного приложения может потребовать улучшения и оптимизации с целью повышения его эффективности. Возможным направлением является расширение поддержки более широкого спектра размеров блоков шифрования, а также реализация новых алгоритмов шифрования. Кроме того, необходимо продолжить работу над удобством использования приложения и повышением уровня его безопасности.

В целом, данное исследование позволило реализовать важные алгоритмы шифрования и создать интуитивно понятный графический интерфейс для работы с цифровыми данными.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Нестеренко, А.Ю. Теоретико-числовые методы в криптографии: учебное пособие / Ю.А. Нестеренко. — Москва: гос. ин-т. электроники и математики. 2012. - 224 с.

2 Белов, А.Г. Исследование алгоритма дискретного логарифмирования Адлемана / А.Г. Белов // Труды Российской национальной конференции «Математика, информатика, физика». — М.: Издательство Физико-математического факультета МГУ, 2016. - С. 297-301.

3 Авдошин, С.М. Дискретная математика. Модулярная алгебра, криптография, кодирование / С.М. Авдошин, А.А. Набебин. — М.: ДМК Пресс, 2017. - 354 с.

4 Rose, J.S. Computational problems in abstract algebra / J.S. Rose. — NYC.: Dover Publications, 1995, - 448 p.

5 Кормен, Т. Алгоритмы. Построение и анализ / Т. Кормен, Ч. Лейзер, Р. Ривест, К. Штайн. — М.: МЦНМО, 2006. - 1312 с.

6 Маховенко, Е.Б. Теоретико-числовые методы в криптографии: учебное пособие / Е.Б. Маховенко. — М.: Гелиос АРВ, 2006. - 320 с.

7 Коблиц, Н. Курс теории чисел и криптографии / Н. Коблиц. — Москва: ТВПб, 2001. - 254 с.

8 Крон, Р.В. Линейная алгебра: учебное пособие для студентов вузов сельскохозяйственных, инженерно-технических и экономических направлений / Р.В. Крон, С.В. Попова, Н.Б. Смирнова, Е.В. Долгих. – М., 2015. - 115 с.

9 Ященко, В.В. Введение в криптографию / В.В. Ященко. — 4-е изд., доп. — М.: МЦНМО, 2012. — 348 с.

10 Салий, В.Н. Криптографические методы и средства защиты информации: учебное пособие / В.Н. Салий. — Саратов: СГУ, 2012. - 40 с.

11 Василенко, О. Н. Теоретико-числовые алгоритмы в криптографии / О.Н. Василенко. — Москва: МЦНМО, 2003. - 328 с.

12 Коробейников, А. Г. Математические основы криптологии. Учебное пособие / А. Г. Коробейников, Ю.А.Гатчин. — СПб: СПб ГУ ИТМО, 2004. - 106 с

13 Paar, C. Understanding Cryptography: A Textbook for Students and Practitioners / C. Pear, J. Pelzl. — NYC.: Springer, 2010, - 464 p.

14 Коган, И.А. Криптография. Введение в математическую теорию защиты информации / И.А. Коган, Ю.А. Романенко. — М.: МЦНМО, 2010. - 288 с.

15 Болотов, А.А. Элементарное введение в эллиптическую криптографию: Алгебраические и алгоритмические основы / А.А. Болотов, С.Б. Гашков, А.Б. Фролов, А.А. Часовских. — М.: КомКнига, 2006. - 328 с.

16 Зубов, А.Ю. Криптографические методы защиты информации. Совершенные шифры / А.Ю. Зубов. — М.: Гелиос АРВ, 2005. - 192 с.

17 Hankerson, D. Guide to Elliptic Curve Cryptography / D. Hankerson, A. Menezes, S. Vanstone. — NYC.: Springer, 2004, - 352 p.

18 Игнашев, А.Е. Простые числа и числа Ферма: учебное пособие / А.Е. Игнашев. — М.: Физматлит, 2002. - 240 с.

19 Рацеев, С.М. Элементы криптографии. Часть 1 / С.М. Рацеев. — Ульяновск: УлГУ, 2012. -112 с.

20 Саломаа, А. Криптография с открытым ключом / А. Саломаа. — М.: Мир, 1996. - 318 с.

21 Payne, R. Beginning App Development with Flutter: Create Cross-Platform Mobile Apps / R. Payne. — Apress: 1st ed. edition, 2019, - 336 p.